Healthcare Extensions for MDT 2010

Administrators Guide

Prepared for

Microsoft

Version 6.0.0.0 Baseline

First published

30 April 2010

Copyright

Disclaimer

At the time of writing this document, Web sites are referenced using active hyperlinks to the correct Web page. Due to the dynamic nature of Web sites, in time, these links may become invalid. Microsoft is not responsible for the content of external Internet sites.

TABLE OF CONTENTS

1 EXECUTIVE SUMMARY

Healthcare Extensions for Microsoft Deployment Toolkit (MDT) 2010 extends the public version of MDT 2010 to:

- Help healthcare organisations reduce the time required to develop a server and desktop build and the deployment infrastructure, by providing a packaged set of tasks common in operating system deployment scenarios
- Reduce errors associated with creating builds and produce higher-quality images for deployment through the inclusion of scripts in a preconfigured location that provides a tested platform to build upon

This guide contains step-by-step instructions for the installation and use of Healthcare Extensions for MDT 2010.

2 INTRODUCTION

Healthcare MDT 2008 provided a set of healthcare-specific enhancements that could be installed over the top of the publicly available version of MDT 2008, and could be used for the deployment of both Windows® XP and Windows Vista® operating systems.

In September 2009, Microsoft publicly released MDT 2010, which provides support for Windows® 7 and Windows Server® 2008 R2. Healthcare Extensions for MDT 2010 is the version of Healthcare MDT updated to work with MDT 2010 and to provide specific enhancements.

The supported operating systems for deployment using Healthcare Extensions for MDT 2010 are:

- Windows 7 (32-bit)
- Windows Vista SP2 (32-bit)
- Windows XP SP3 (32-bit)
- Windows Server 2008 R2 (64-bit)

2.1 Value Proposition

This guide explains how Healthcare Extensions for MDT 2010 can be used to:

- Produce a client or server image more quickly than with existing methods due to pre-populated task sequences that are already configured. These task sequences are not part of the stand-alone MDT 2010
- Provide a build environment to create operating system images, including automated installation of the required applications, drivers, hotfixes and additional configurations needed to create healthcare-customised images of the supported Windows operating systems
- Provide a Lite Touch deployment environment to deliver the Windows-based automated builds and images to healthcare organisations' desktop estates of between 10 and 500 seats
- Provide simple step-by-step guidance for using Healthcare Extensions for MDT 2010 in basic areas, such as adding applications and drivers to a build

2.2 Healthcare Extensions for MDT 2010 Overview

Microsoft has two implementations of the MDT 2010 methodology:

- Lite Touch Installation (LTI)
- Zero Touch Installation (ZTI)

Healthcare Extensions for MDT 2010 can be used to extend either of these implementations; however the Lite Touch version of MDT 2010 with the Healthcare Extensions for MDT 2010 extensions will be most useful for healthcare organisations that do not have a management infrastructure in place, but wish to take advantage of a solution for full, end-to-end deployments.

Healthcare Extensions for MDT 2010 overlays the public version of MDT 2010 and makes use of new MDT 2010 features, including the Information Center. This provides easy access to download and install all the components that MDT 2010 and Healthcare Extensions for MDT 2010 require, including:

- Windows Automated Installation Kit (WAIK)
- Application Compatibility Toolkit (ACT)
- Windows Vista Hardware Assistant (WVHA)
- Volume Activation tools
- User State Migration Tool (USMT)
- Microsoft Office Migration Tools

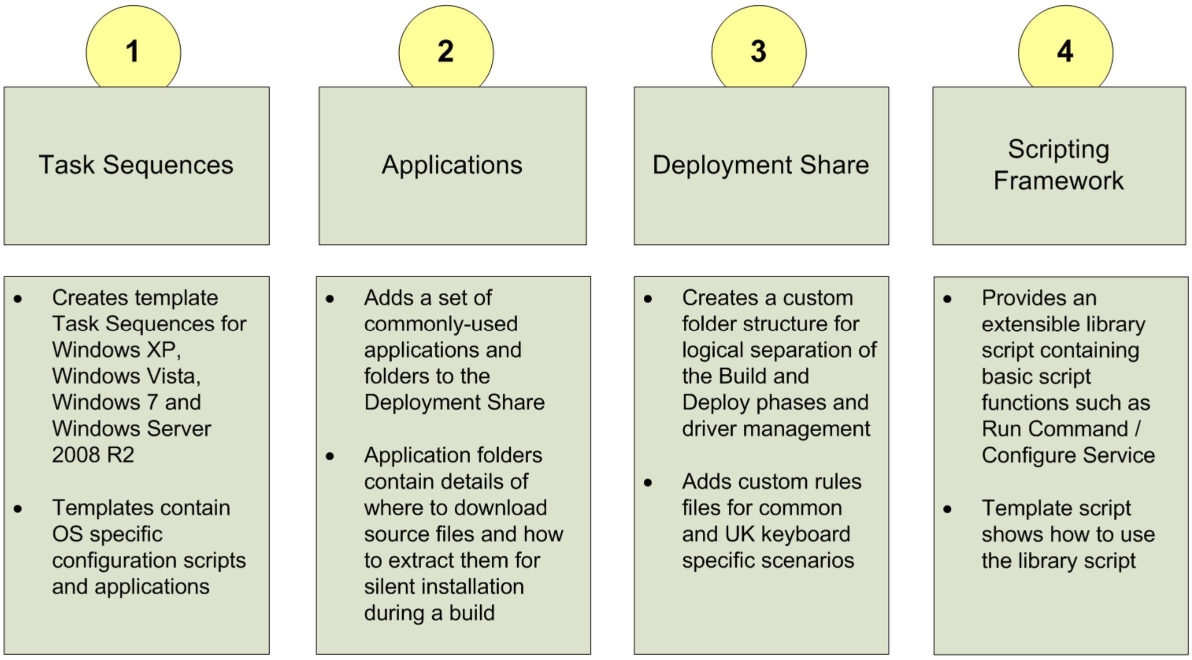
Healthcare Extensions for MDT 2010, together with MDT 2010, provides healthcare organisations with solutions to the problems faced during deployment, including:

- The pre-installation phases (for example, disk partitioning and formatting)
- The installation phase (disk imaging)
- The post-installation phases (for example, user state migration, application installation and customisation)

Healthcare Extensions for MDT 2010 simplifies building, customising and deploying Windows images for healthcare organisations.

2.2.1 Healthcare Extensions for MDT 2010 Enhancements

The enhancements that Healthcare Extensions for MDT 2010 provides are shown in Figure 1:



1 Task Sequences

- Creates template Task Sequences for Windows XP, Windows Vista, Windows 7 and Windows Server 2008 R2
- Templates contain OS specific configuration scripts and applications

2 Applications

- Adds a set of commonly-used applications and folders to the Deployment Share
- Application folders contain details of where to download source files and how to extract them for silent installation during a build

3 Deployment Share

- Creates a custom folder structure for logical separation of the Build and Deploy phases and driver management
- Adds custom rules files for common and UK keyboard specific scenarios

4 Scripting Framework

- Provides an extensible library script containing basic script functions such as Run Command / Configure Service
- Template script shows how to use the library script

Figure 1: Healthcare Extensions for MDT 2010 Enhancements

The following describes each of the Healthcare Extensions for MDT 2010 enhancements:

- **Task Sequences** – Healthcare Extensions for MDT 2010 adds a number of template Task Sequences for Windows XP, Windows Vista, Windows 7 and Windows 2008 R2. The pre-populated task sequence specifies the steps to create a build for a given operating system, and references a number of customisation scripts supplied with Healthcare Extensions for MDT 2010. The templates also contain the recommended base applications for a given operating system. For example, the relevant .NET Framework and Windows Installer versions. The custom applications and scripts referenced by the templates are initially disabled as the underlying application source files need to be downloaded before use
- **Applications** – Healthcare Extensions for MDT comes with pre-created scripts to silently install the most common applications used in builds, including .NET Framework, Windows Installer, Java™ and Adobe® Reader®. Each application comes with a folder structure and details of where to obtain the source files and how to integrate them into Healthcare Extensions for MDT 2010. These scripts and folders allow a generic build to be created very quickly
- **Deployment Share** – A number of folders are created in the Deployment Share to help keep builds, Task Sequences, drivers and operating system files organised. These folder structures also help to separate the files into Build and Deploy related groups. The folders are discussed in more detail in section 2.2.2
- **Custom Settings file –** The primary customisation point for Deployment Shares are rules, which are stored in a file called CustomSettings.ini. This file contains rules for installing the Windows operating systems on destination computers. Healthcare Extensions for MDT 2010 configures a UK-specific CustomSettings.ini file, with additional settings for automating the deployment process
- **Scripting Framework** – Healthcare Extensions for MDT 2010 scripts utilise a custom VBScript library script which contains pre-written code to perform common operations, such as running a program or configuring services. The script can be extended easily to support custom operations, and used as the basis for additional scripts

2.2.2 Changes from Healthcare MDT 2008

This section gives an overview of the more significant changes made since the release of Healthcare MDT 2008 for both the software installation and this document.

Software Installation

- Support has been added for Windows 7 and Windows Server 2008 R2
- Custom folder structures have been added to help separate the Build and Deploy processes
- The requirement of an existing Task Sequence and Operating System for installation has been removed
- Task Sequences are now installed as reusable templates, rather than overwriting an existing Task Sequence
- The number of applications in the Task Sequences templates has been reduced to the most common applications

- Pre-configured applications have been updated for the current versions of:
  - Microsoft App-V Client (formerly SoftGrid®)
  - Microsoft Office Communicator
  - Adobe Reader
  - .NET Framework
  - Sun™ Java Runtime
  - Microsoft System Center Configuration Manager 2007 Client
- Many seldom used server-based applications and scripts have been removed
- The installation now creates a separate folder structure for ease of upgrade or side-by-side installation of an already configured MDT 2010 Deployment Share

Administrator Guide

- Prescriptive guidance added for the installation of drivers that take advantage of the new custom folder structures and the selection profiles added to Healthcare Extensions for MDT 2010
- More guidance added relating to the separation of the Build and Deploy processes
- All diagrams and procedures updated to reflect MDT 2010
- More detailed guidance added for the creation of additional network Deployment Shares, USB and DVD media

2.2.3 Build and Deploy Terminology

When deploying a Windows operating system to any significant number of machines, it is recommended to do this in two phases. In Healthcare Extensions for MDT 2010, these phases are referred to as **Build** and **Deploy,** as shown in Figure 2. A *Lab* and *Production* separation is shown, whereby the creation of a Windows build image within the lab may require several iterations, and can cause significant network load. Where it is not possible or practical to create a separate lab environment, the Build and Deploy MDT servers can be one and the same. However, care should be taken to configure the network to reduce the impact of the build process on production users. It is also possible for the deployment mechanism to be a USB disk or DVD, rather than an MDT server.

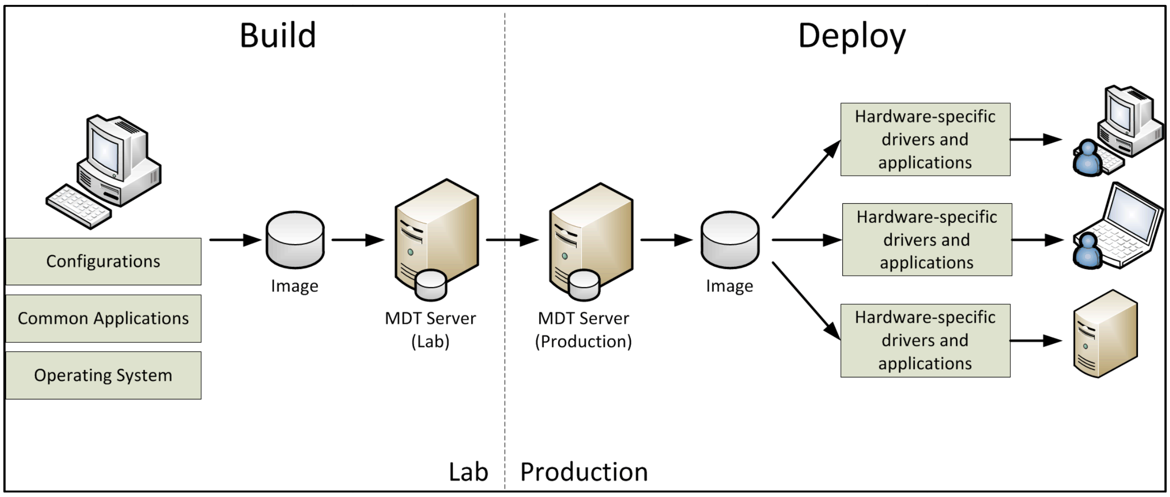

Build
Deploy
Configurations
Common Applications
Operating System
Image
MDT Server (Lab)
MDT Server (Production)
Image
Hardware-specific drivers and applications
Hardware-specific drivers and applications
Hardware-specific drivers and applications
Lab
Production

Figure 2: Build and Deploy Phases

2.2.3.1 Build

During the Build phase, the operating system is installed along with custom configurations and applications that will be common to all clients. All application installations and configurations should be scripted so that the Build process can be repeated without introducing human error. A common application list would include:

- Service Packs (SPs) and hotfixes
- .NET Framework 1.1 SP1 and .NET Framework 3.5 SP1
- Windows Installer
- Microsoft Office 2007
- Adobe Reader
- Sun Java Runtime

Although it is possible to perform a Build directly to an end-user machine, it is not recommended. Even with the small number of applications listed here, the process could be measured in hours rather than minutes. For this reason, the Build is performed once and then captured to an image. This image is then used as the basis for rapid mass deployment.

The Build image should be as small and generic as possible so that it can be used as the basis for deployment to a wide range of hardware types. A single modern driver can be 200MB in size, so only drivers needed to perform the Build, usually network and storage drivers, should be installed during this phase. Hardware-specific drivers will be dynamically installed during the Deploy phase.

2.2.3.2 Deploy

The image captured in the Build phase is applied to an end-user machine. Applying an image is a quick process that usually takes less than 10 minutes, depending on the speed of the network and target machine. However, as the captured Build image was generic, it is at this point that hardware-specific drivers and applications, such as hotkey button and power management utilities, must be installed. Generally, these hardware-specific files only take a few minutes to install, and do not significantly affect deployment time. Healthcare Extensions for MDT 2010 uses Selection Profiles and MDT variables to automatically determine which drivers should be installed on a particular hardware model.

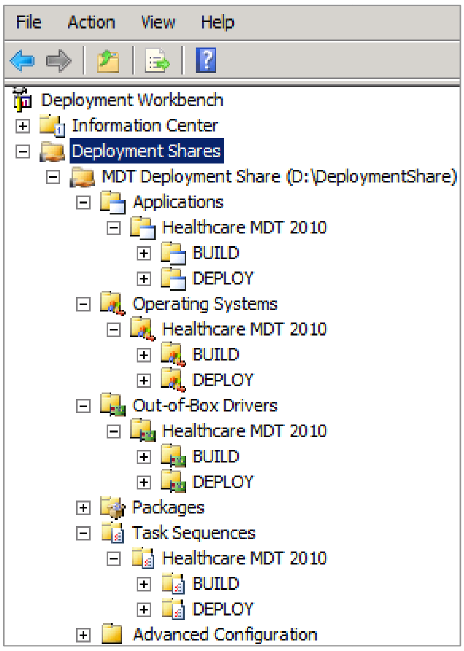
It is also possible to install additional applications at this point, such as business unit-specific programs, or applications that, for technical reasons, cannot be installed in the Build phase. Applications that require unique activation are a common example. The number of applications installed in the Deploy phase should be kept to a minimum so that deployment time is not adversely affected.

Figure 2 shows that the Deploy phase is run from an MDT server. Healthcare Extensions for MDT 2010 allows the IT Administrator to easily use USB disks and DVDs as well.

In healthcare organisations where operating system and software distribution mechanisms such as Configuration Manager are in use, Healthcare Extensions for MDT 2010 is still very useful. The image captured in the Build phase can be deployed directly by Configuration Manager.

2.2.3.3 Healthcare Extensions for MDT 2010 Folder Structure

To provide a visible structure for the Build and Deploy phases, Healthcare Extensions for MDT 2010 creates various folders in the Deployment Workbench. These allow the IT Administrator to keep in mind the purpose of the files, applications and drivers they add into the Deployment Workbench. The folder structure is shown in Figure 3. MDT Selection Profiles also reference these folders to support hardware-specific driver installation and easy creation of USB and DVD deployment media.



- Deployment Workbench
  - Information Center
  - Deployment Shares
    - MDT Deployment Share (D:\DeploymentShare)
      - Applications
        - Healthcare MDT 2010
          - BUILD
          - DEPLOY
      - Operating Systems
        - Healthcare MDT 2010
          - BUILD
          - DEPLOY
      - Out-of-Box Drivers
        - Healthcare MDT 2010
          - BUILD
          - DEPLOY
      - Packages
      - Task Sequences
        - Healthcare MDT 2010
          - BUILD
          - DEPLOY
      - Advanced Configuration

Figure 3: Healthcare Extensions for MDT 2010 Folder Structure

2.3 Knowledge Prerequisites

To effectively implement the recommendations made throughout this document, a number of knowledge-based and environmental infrastructure prerequisites should be in place. This section outlines the knowledge and skills required to use Healthcare Extensions for MDT 2010, while section 2.4 details the necessary infrastructure prerequisites. Section 2.3.1 details the prerequisite skills and knowledge, and section 2.3.2 details the suggested training resources and skill assessment.

2.3.1 Skills and Knowledge

The technical knowledge and minimum skills required to use this guide and Healthcare Extensions for MDT 2010 are:

- Familiarity with the MDT 2010 *Microsoft Deployment Toolkit Documentation Library* documentation, as installed with MDT 2010
- Experience of the installation and configuration of WAIK, and using the toolkit to:
  - Mount images
  - Modify Windows PE (WinPE)
- Familiarity with MDT 2010 to complete a bare-metal LTI build

2.3.2 Training and Assessment

Guidelines on the skill sets required in order to make best use of Healthcare Extensions for MDT 2010, as well as reference material and training resources, are detailed in APPENDIX A. However, all courses mentioned are optional and can be provided by a variety of certified training partners.

2.4 Infrastructure Prerequisites

Healthcare Extensions for MDT 2010 is designed to integrate into an existing development lab that includes servers and infrastructure for creating builds and testing deployment. If no such lab exists, Figure 4 illustrates the infrastructure prerequisites for implementing a development lab for use with Healthcare Extensions for MDT 2010. The development lab supports all the services and tools required to develop the client builds using two servers.

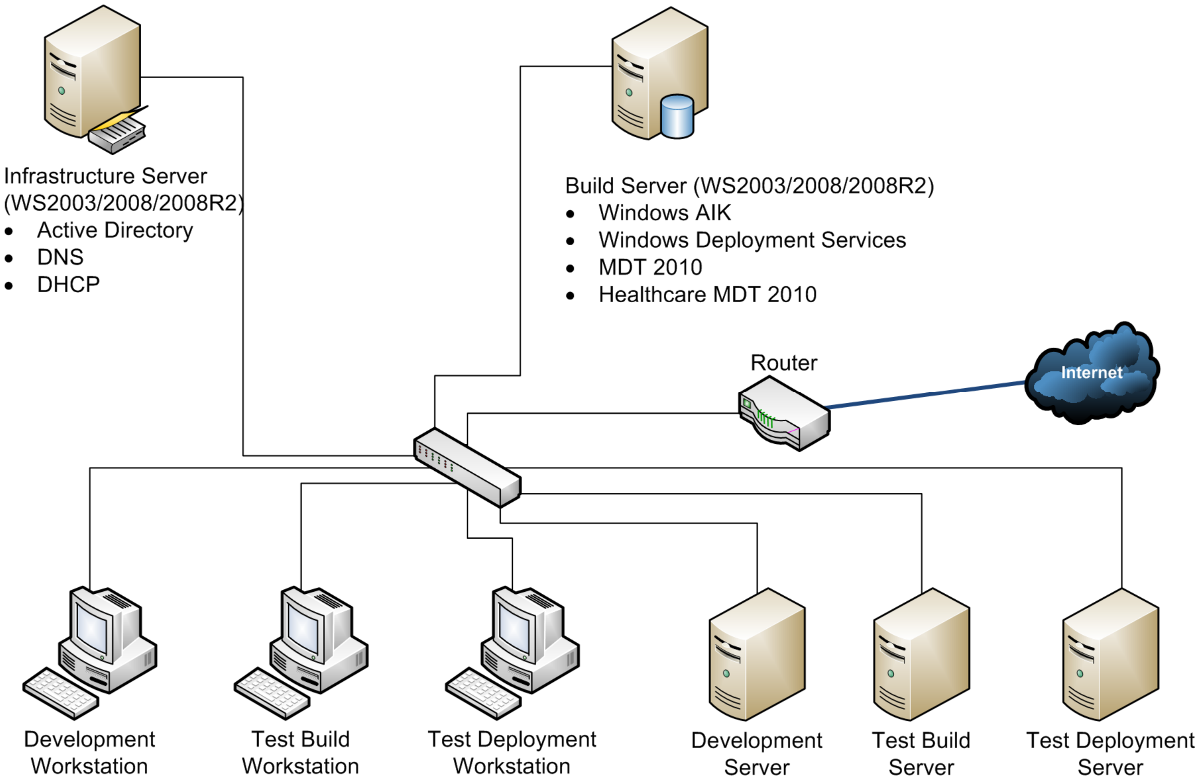


Figure 4: Infrastructure Prerequisites

> **Recommendation**
>
> It is recommended that healthcare organisations create a lab environment to develop the build. Once complete, the healthcare organisations should install a build server within the production environment to deploy the final image to target machines.
>
> **Note**
>
> For small healthcare organisations, or for Lab use, it is possible to place all the Infrastructure and Build Server roles on the same machine.

The servers above perform the roles outlined in Table 1:

Deployment Server Roles	Details
Infrastructure Server	Provides the required infrastructure services to the environment including Active Directory®, Domain Name System (DNS) and Dynamic Host Control Protocol (DHCP)
Build Server	Hosts all the tools and resources required to generate the initial build, including WAIK, Windows Deployment Server (WDS), MDT 2010 and Healthcare Extensions for MDT 2010

Table 1: Client Build Development Lab Servers

2.5 Audience

The guidance contained in this document is targeted at a variety of roles within healthcare IT organisations. Table 2 provides a reading guide for this document, illustrating the roles and the sections of the document that are likely to be of most interest. The structure of the sections referred to is described in section 3.1.

Role	Document Usage	Executive Summary	Introduction	Develop	Stabilise	Deploy
IT Manager	Review the relevant areas within the document to understand the justification and drivers, and to develop an understanding of the implementation requirements	✓	✓			
IT Architect	Review the relevant areas within the document against local architecture strategy and implementation plans	✓	✓	✓		
IT Professional/ Administrator	Detailed review and implementation of the guidance to meet local requirements	✓	✓	✓	✓	✓

Table 2: Document Audience

2.6 Assumptions

Use of this guide and Healthcare Extensions for MDT 2010 is based on the assumption that the healthcare organisation has the following in place:

- Windows Server 2003 SP2, Windows Server 2008 SP2, or Windows Server 2008 R2, with all critical updates applied
- Volume-licensed media for Windows 7, Windows Vista SP2, Windows XP SP3 and Windows Server 2008 R2
- License keys for Windows 7, Windows Vista SP2, Windows XP SP3 and Windows Server 2008 R2 (if using a Key Management Server (KMS) then license keys are only required for Windows XP)
- Administrative access to the server on which Healthcare Extensions for MDT 2010 is to be installed
- Access to the Healthcare Extensions for MDT 2010 setup files, the publicly-downloadable version of MDT 2010 and the WAIK (download locations and installation instructions for these can be found in section 4.1.1)

3 USING THIS DOCUMENT

This document is intended for use by healthcare organisations and IT Administrators who wish to use Healthcare Extensions for MDT 2010 to create and deploy the supported operating system builds. The document should be used to assist with the planning and implementation of Healthcare Extensions for MDT 2010 and as a reference guide for the most common tasks involved with its use.

3.1 Document Structure

This document contains three sections that deal with the project lifecycle, as illustrated in Figure 5:

- Develop
- Stabilise
- Deploy

Each section is based on the Microsoft IT Project Lifecycle as defined in the Microsoft Solutions Framework (MSF) Process Model, and the Microsoft Operations Framework (MOF). The IT Project Lifecycle is described in more detail in the *MSF Process Model White Paper*[1] and the *MOF Executive Overview*[2]. The MSF Process Model and MOF describe a high-level sequence of activities for building, deploying and managing IT solutions. Rather than prescribing a specific series of procedures, they are flexible enough to accommodate a broad range of IT projects.

As this document is intended to be a step-by-step guide for using Healthcare Extensions for MDT 2010, the three extra lifecycles stages (Envision, Plan and Operate) are not required or referred to within this guidance.

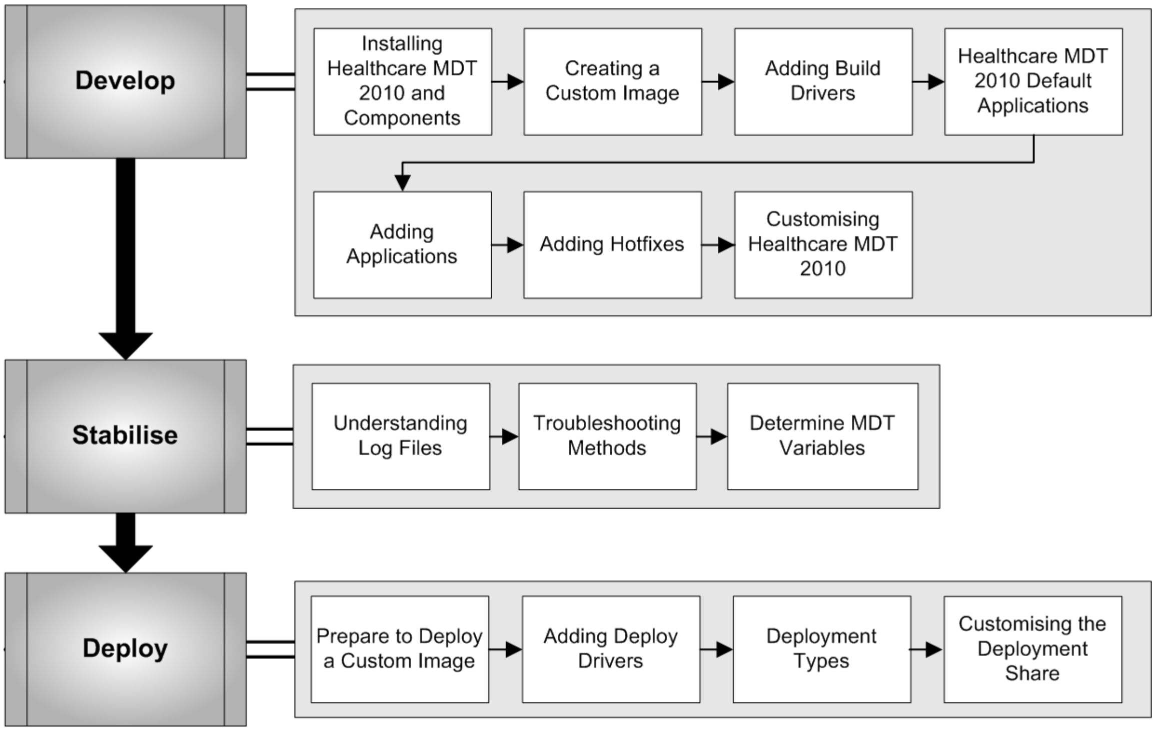


Figure 5: MSF Process Model Phases and Document Structure

[1] MSF Process Model White Paper: http://www.microsoft.com/downloads/details.aspx?FamilyID=e481cb0b-ac05-42a6-bab8-fc886956790e&DisplayLang=en

[2] MOF Executive Overview: http://www.microsoft.com/technet/itsolutions/cits/mo/mof/mofeo.mspx

4 Develop

During the Develop phase, the solution components are installed and configured to provide a development environment. The operating systems to be deployed can then be configured according to any previous planning and design. Further refinement of these components will continue into the stabilisation phase.

Figure 6 acts as a high-level checklist, illustrating the tasks that an IT Professional needs to perform when developing Healthcare Extensions for MDT 2010 within a healthcare organisation.

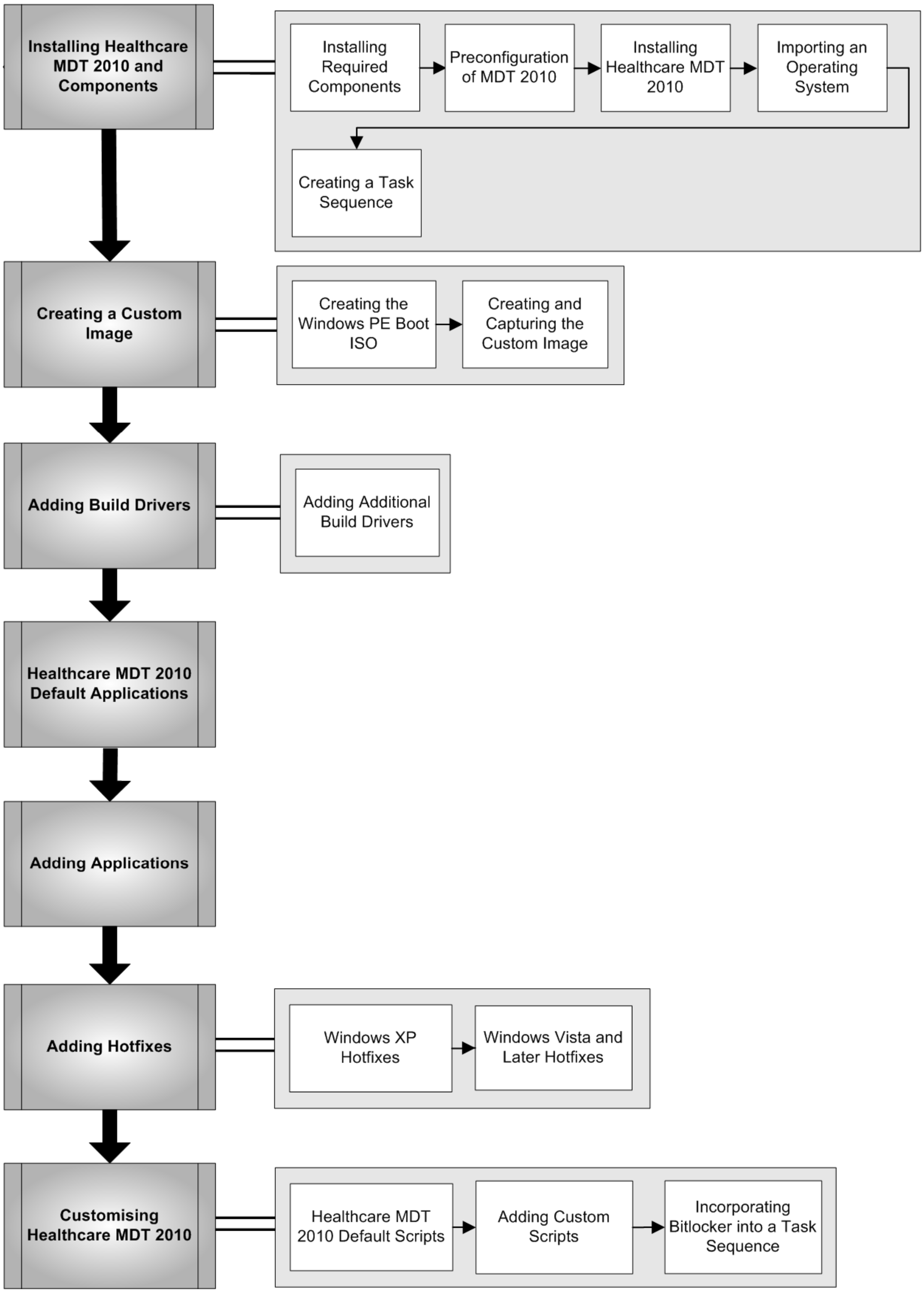

Installing Healthcare MDT 2010 and Components
Installing Required Components
Preconfiguration of MDT 2010
Installing Healthcare MDT 2010
Importing an Operating System
Creating a Task Sequence
Creating a Custom Image
Creating the Windows PE Boot ISO
Creating and Capturing the Custom Image
Adding Build Drivers
Adding Additional Build Drivers
Healthcare MDT 2010 Default Applications
Adding Applications
Adding Hotfixes
Windows XP Hotfixes
Windows Vista and Later Hotfixes
Customising Healthcare MDT 2010
Healthcare MDT 2010 Default Scripts
Adding Custom Scripts
Incorporating Bitlocker into a Task Sequence

Figure 6: Sequence for Developing Healthcare Extensions for MDT 2010

4.1 Installing Healthcare Extensions for MDT 2010 and Components

This section provides the details of the prerequisite components that Healthcare Extensions for MDT 2010 requires and the step-by-step guidance for installation.

> **Note**
>
> Many sections of this document refer to the 'Distribution folder'. This is the folder where all the MDT files, applications and drivers are located. By default, the folder will be C:\Distribution or D:\Distribution, and can be changed during the installation of MDT 2010. When giving file paths in this document, the Distribution folder will be referred to in the format of <Distribution>\<Folder>\<File>.

4.1.1 Installing Required Components

Before installing Healthcare Extensions for MDT 2010, the components listed in Table 3 must be installed on the build server.

> **Important**
>
> These components do not need to be reinstalled if they are already installed on the build server. More recent operating systems, such as Windows Server 2008, only require the download of the Windows Automated Installation Kit and MDT 2010.

Component	Location
Microsoft Core Extensible Markup Language (MSXML) Services 6.0	http://www.microsoft.com/downloads/details.aspx?FamilyID=993c0bcf-3bcf-4009-be21-27e85e1857b1&DisplayLang=en
Microsoft Management Console (MMC) 3.0	http://www.microsoft.com/downloads/details.aspx?FamilyID=4c84f80b-908d-4b5d-8aa8-27b962566d9f&DisplayLang=en
Microsoft® .NET Framework version 2.0	http://www.microsoft.com/downloads/details.aspx?FamilyID=0856eacb-4362-4b0d-8edd-aab15c5e04f5&DisplayLang=en
Windows Automated Installation Kit 2.0	http://www.microsoft.com/downloads/details.aspx?familyid=696DD665-9F76-4177-A811-39C26D3B3B34&displaylang=en
Windows PowerShell 1.0	http://www.microsoft.com/downloads/details.aspx?familyid=10EE29AF-7C3A-4057-8367-C9C1DAB6E2BF&displaylang=en
MDT 2010	http://www.microsoft.com/downloads/details.aspx?DisplayLang=en&FamilyID=3bd8561f-77ac-4400-a0c1-fe871c461a89

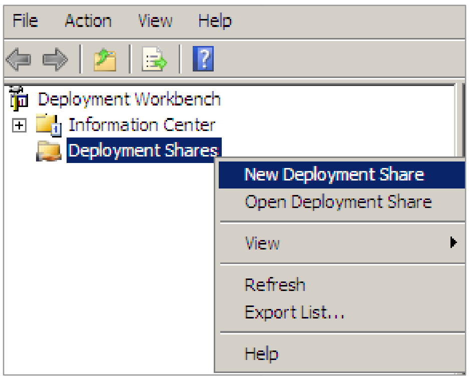
Table 3: Build Server Components for Healthcare Extensions for MDT 2010

4.1.2 Pre-configuration of MDT 2010

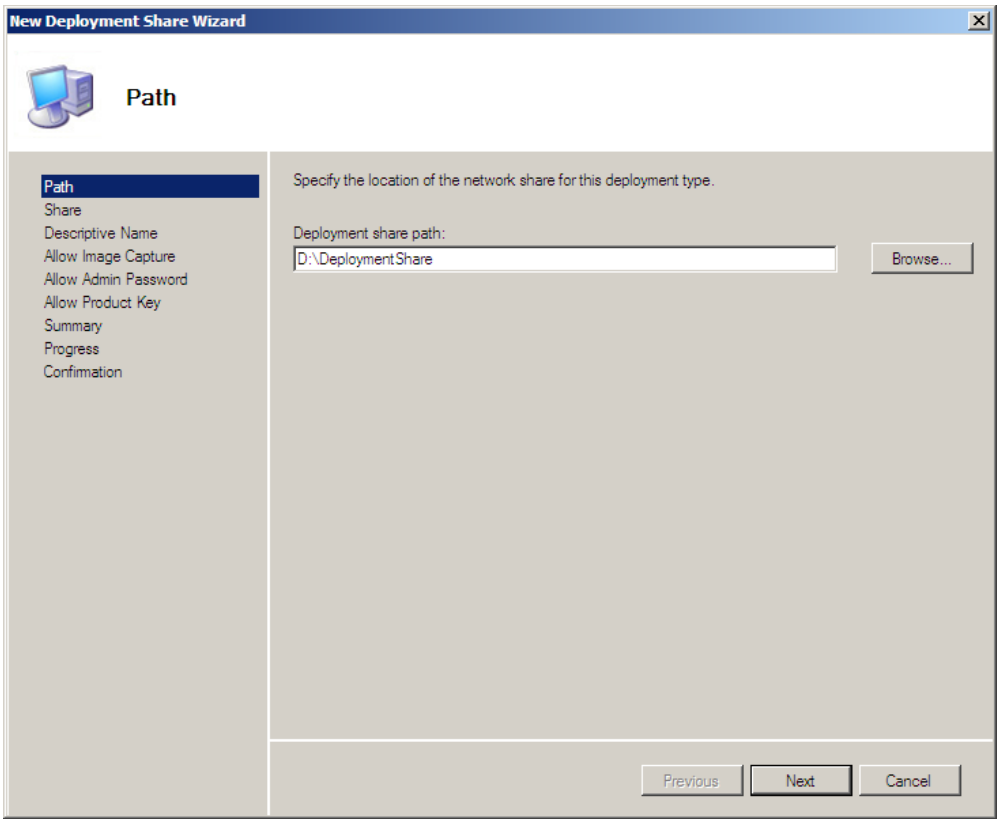
MDT 2010 needs to have a basic configuration in place prior to installing Healthcare Extensions for MDT 2010. A Deployment Share must first be created:

To create the Deployment Share:

1. In the Deployment Workbench, right-click **Deployment Shares** and select **New Deployment Share**:



2. Specify the path in **Deployment share path** and click **Next**:



Specify the location of the network share for this deployment type.

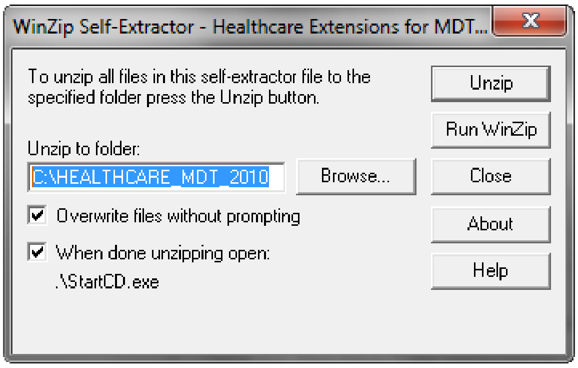
Deployment share path: D:\DeploymentShare

3. Accept the defaults on all remaining pages.

> **Important**
>
> The creation of the deployment share directory only sets permissions for the local Administrators group. To allow additional users to access the share, the NTFS and share permissions should be modified. As the Deployment Share is like any other file share this is done in the standard way.

4.1.3 Installing Healthcare Extensions for MDT 2010

1. Ensure that the Deployment Workbench is closed and run the Healthcare Extensions for MDT 2010 Installer.
2. The self-extracting executable will prompt for a location to extract the installation files. Type the destination folder or click **Browse** and select a folder from the **Select Folder** screen:



WinZip Self-Extractor - Healthcare Extensions for MDT...

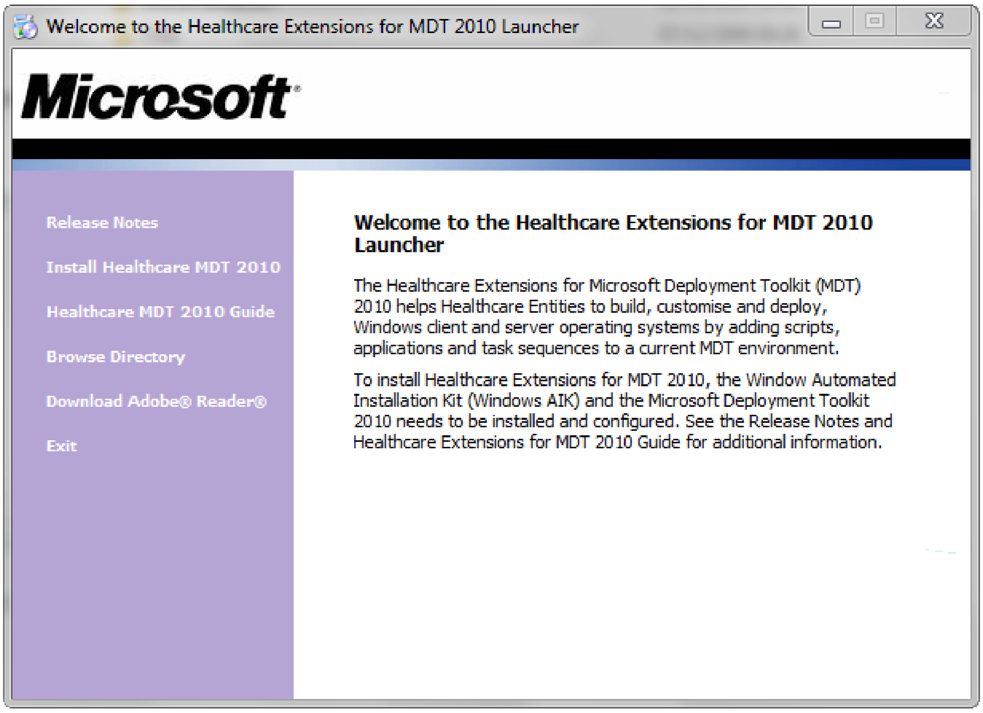
To unzip all files in this self-extractor file to the specified folder press the Unzip button.

Unzip to folder: C:\HEALTHCARE_MDT_2010

☑ Overwrite files without prompting

☑ When done unzipping open: .\StartCD.exe

3. Once the files have been extracted to a local folder, the Healthcare Extensions for MDT 2010 Launcher will start (if the installation does not start automatically, run the StartCD.exe file contained in the folder selected in the previous step):
4. From the Healthcare Extensions for MDT 2010 Launcher, click **Install Healthcare MDT 2010**. The Healthcare Extensions for MDT 2010 Installation wizard starts and displays the Welcome page.



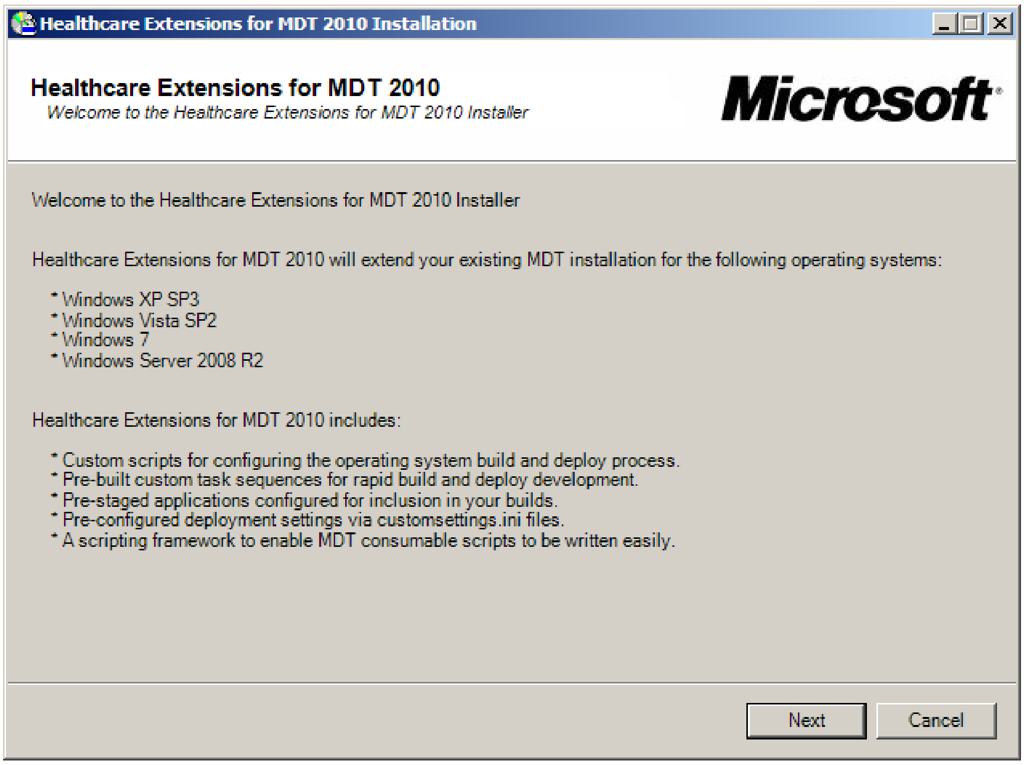
Welcome to the Healthcare Extensions for MDT 2010 Launcher

Welcome to the Healthcare Extensions for MDT 2010 Launcher

The Healthcare Extensions for Microsoft Deployment Toolkit (MDT) 2010 helps Healthcare Entities to build, customise and deploy, Windows client and server operating systems by adding scripts, applications and task sequences to a current MDT environment.

To install Healthcare Extensions for MDT 2010, the Window Automated Installation Kit (Windows AIK) and the Microsoft Deployment Toolkit 2010 needs to be installed and configured. See the Release Notes and Healthcare Extensions for MDT 2010 Guide for additional information.

5. Click **Next**:



Healthcare Extensions for MDT 2010 Installation

Healthcare Extensions for MDT 2010

Welcome to the Healthcare Extensions for MDT 2010 Installer

Welcome to the Healthcare Extensions for MDT 2010 Installer

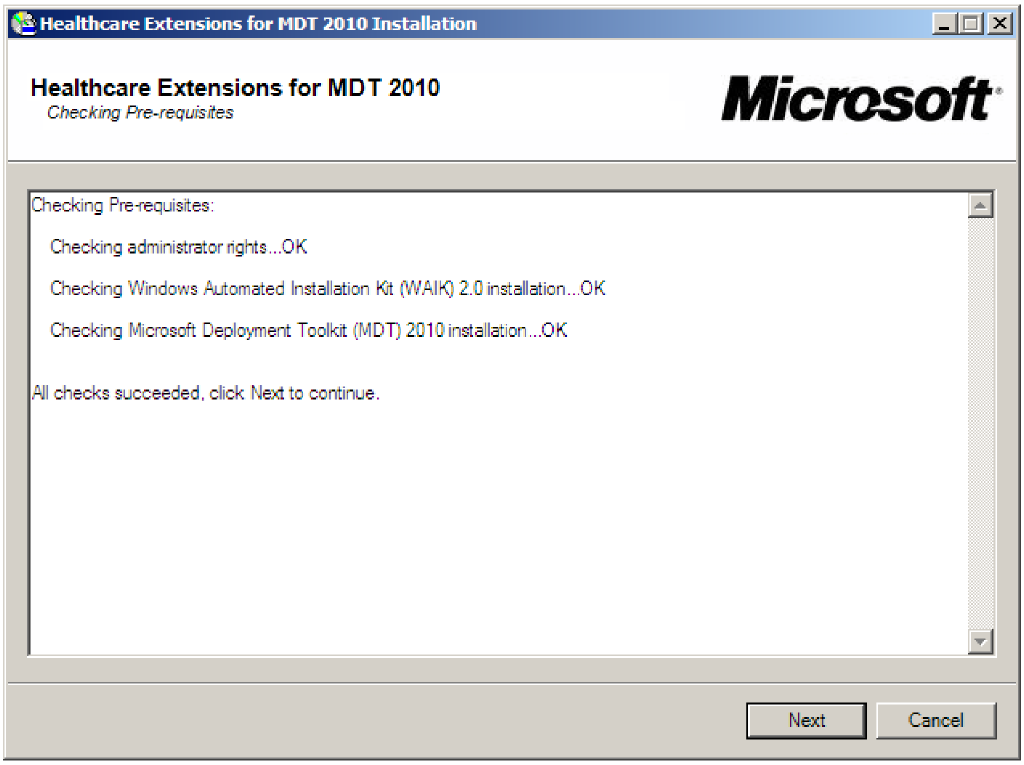
Healthcare Extensions for MDT 2010 will extend your existing MDT installation for the following operating systems:

- Windows XP SP3
- Windows Vista SP2
- Windows 7
- Windows Server 2008 R2

Healthcare Extensions for MDT 2010 includes:

- Custom scripts for configuring the operating system build and deploy process.
- Pre-built custom task sequences for rapid build and deploy development.
- Pre-staged applications configured for inclusion in your builds.
- Pre-configured deployment settings via customsettings.ini files.
- A scripting framework to enable MDT consumable scripts to be written easily.

6. On the **Checking Pre-requisites** page, ensure that all the checks have been passed. If any checks failed, fix the indicated problem and then re-run the installer. Once all the checks have passed, click **Next**:



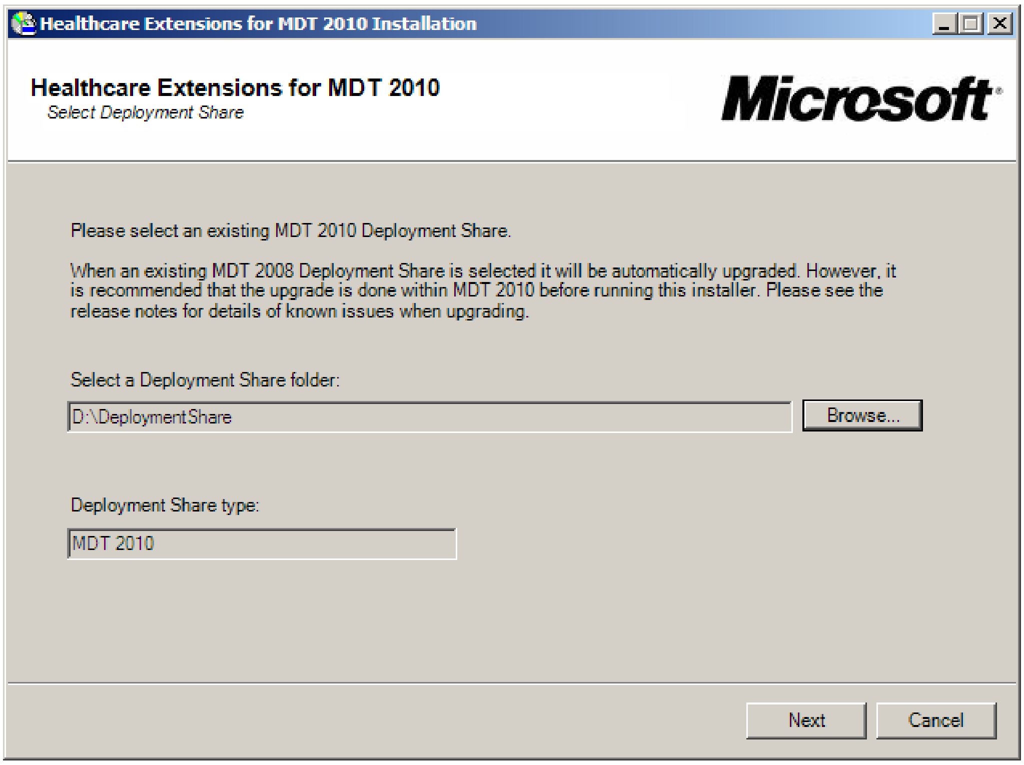
Healthcare Extensions for MDT 2010 Installation

Healthcare Extensions for MDT 2010

Checking Pre-requisites

```
Checking Pre-requisites:

  Checking administrator rights...OK

  Checking Windows Automated Installation Kit (WAIK) 2.0 installation...OK

  Checking Microsoft Deployment Toolkit (MDT) 2010 installation...OK

All checks succeeded, click Next to continue.
```

7. On the **Select Deployment Share** page, enter the path of the Deployment Share created in section 4.1.2 and click **Next**. The installation will now proceed:



Healthcare Extensions for MDT 2010 Installation
Healthcare Extensions for MDT 2010
Select Deployment Share
Microsoft
Browse...
Next
Cancel

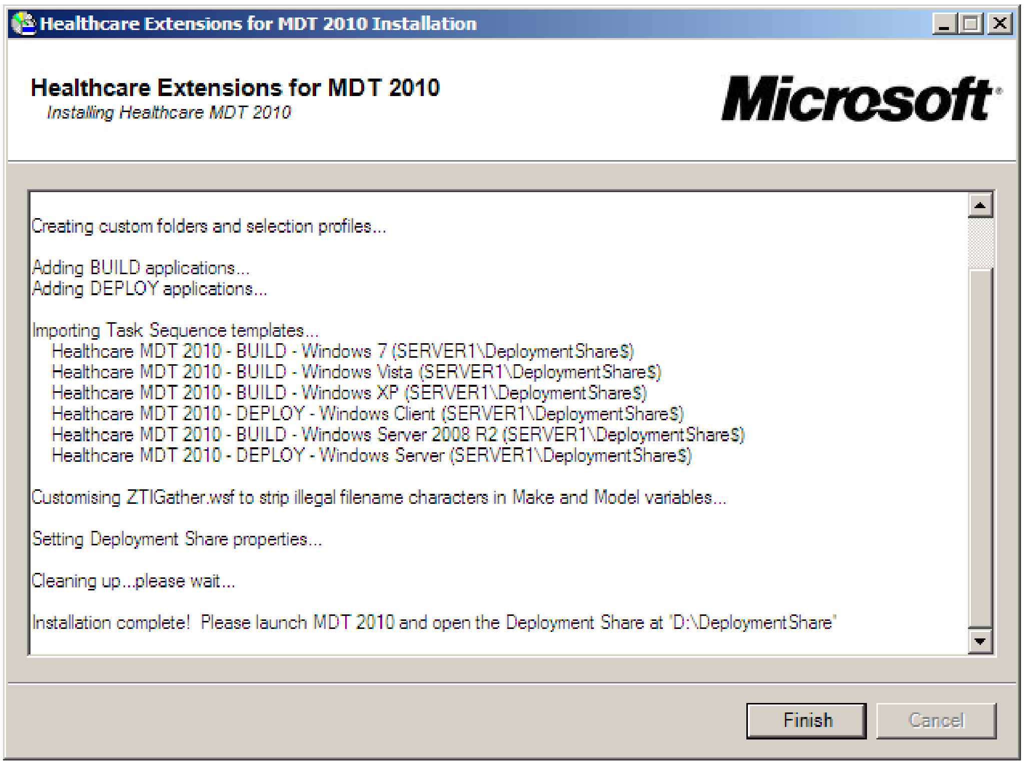
Please select an existing MDT 2010 Deployment Share.

When an existing MDT 2008 Deployment Share is selected it will be automatically upgraded. However, it is recommended that the upgrade is done within MDT 2010 before running this installer. Please see the release notes for details of known issues when upgrading.

Select a Deployment Share folder: D:\DeploymentShare

Deployment Share type: MDT 2010

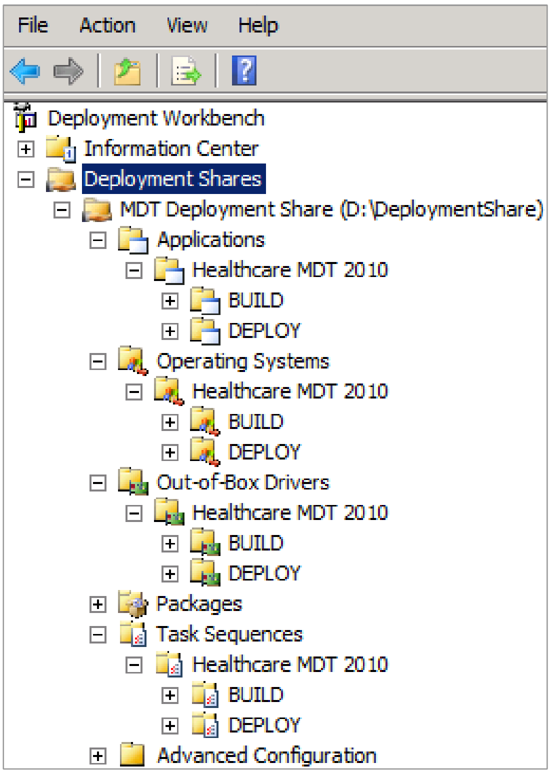
8. Once installation is complete, click **Finish**:



Healthcare Extensions for MDT 2010 Installation
Healthcare Extensions for MDT 2010
Installing Healthcare MDT 2010
Microsoft
Finish
Cancel

```
Creating custom folders and selection profiles...

Adding BUILD applications...
Adding DEPLOY applications...

Importing Task Sequence templates...
   Healthcare MDT 2010 - BUILD - Windows 7 (SERVER1\DeploymentShare$)
   Healthcare MDT 2010 - BUILD - Windows Vista (SERVER1\DeploymentShare$)
   Healthcare MDT 2010 - BUILD - Windows XP (SERVER1\DeploymentShare$)
   Healthcare MDT 2010 - DEPLOY - Windows Client (SERVER1\DeploymentShare$)
   Healthcare MDT 2010 - BUILD - Windows Server 2008 R2 (SERVER1\DeploymentShare$)
   Healthcare MDT 2010 - DEPLOY - Windows Server (SERVER1\DeploymentShare$)

Customising ZTIGather.wsf to strip illegal filename characters in Make and Model variables...

Setting Deployment Share properties...

Cleaning up...please wait...

Installation complete!  Please launch MDT 2010 and open the Deployment Share at 'D:\DeploymentShare'
```

9. Verify that the Healthcare Extensions for MDT 2010 custom folder structure has been created in the Deployment Workbench:

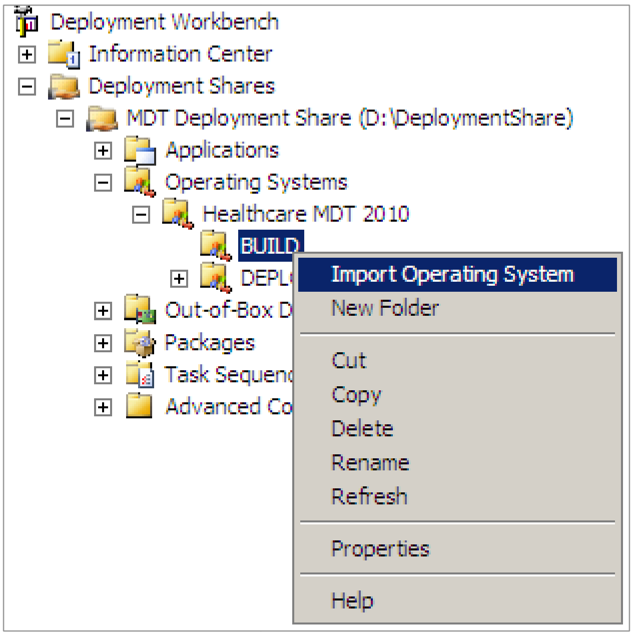


4.1.4 Importing an Operating System

Once Healthcare Extensions for MDT 2010 is installed, the first step should be to import all the operating system source files that will be required to import the operating system.

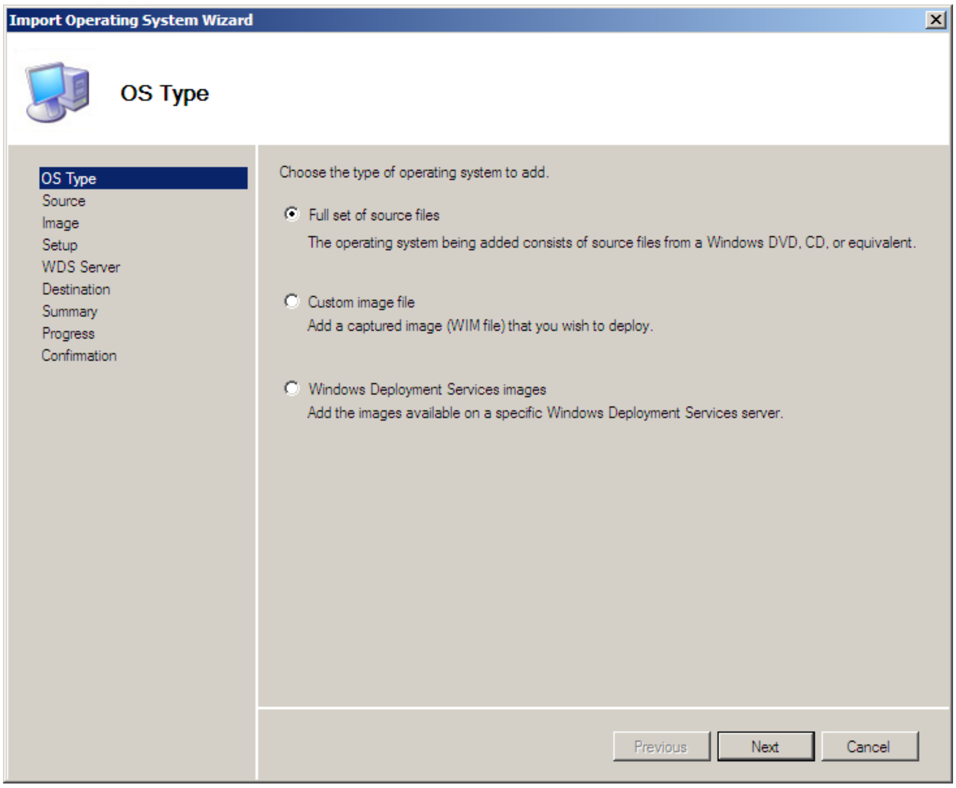
To import the operating system files:

1. In the Deployment Workbench, right-click **Operating Systems \ Healthcare MDT 2010 \ BUILD** and select **Import Operating System**:



The **Import Operating System Wizard** starts and displays the **OS Type** page.

2. Select **Full set of source files** and click **Next**:



Import Operating System Wizard

OS Type

Choose the type of operating system to add.

● Full set of source files

The operating system being added consists of source files from a Windows DVD, CD, or equivalent.

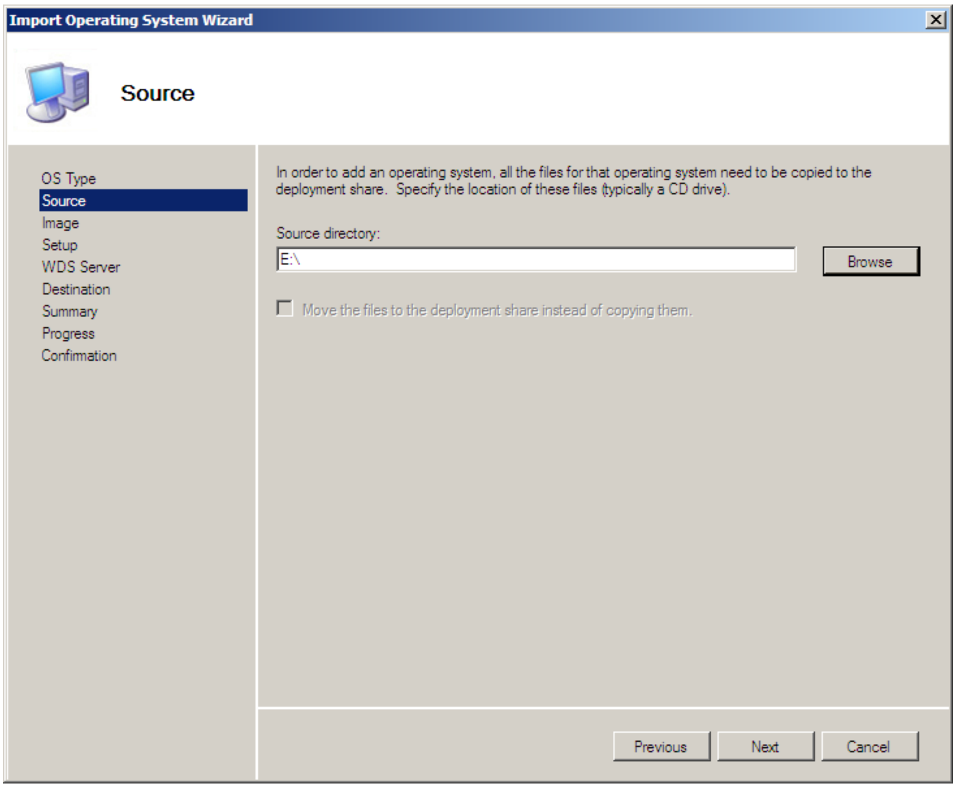
○ Custom image file

Add a captured image (WIM file) that you wish to deploy.

○ Windows Deployment Services images

Add the images available on a specific Windows Deployment Services server.

3. On the **Source** page, browse to the location of the operating system source files and click **Next**:



Import Operating System Wizard

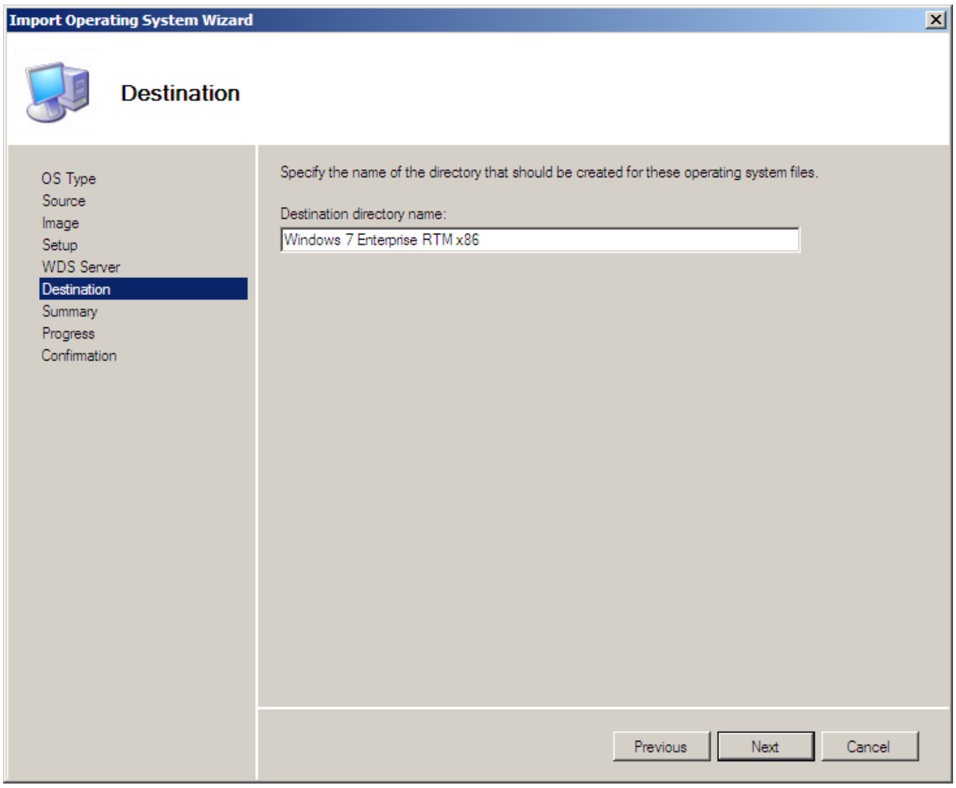
Source

In order to add an operating system, all the files for that operating system need to be copied to the deployment share. Specify the location of these files (typically a CD drive).

Source directory: E:\

□ Move the files to the deployment share instead of copying them.

4. On the **Destination** page, enter the destination directory name that will be created in the MDT 2010 distribution share to store the operating system files. It is recommended that the name is in the convention **<Operating System><Service Pack><Architecture>**. For example, *Windows Vista Enterprise SP2 x86* or *Windows 7 Enterprise RTM x86*. Click **Next**:

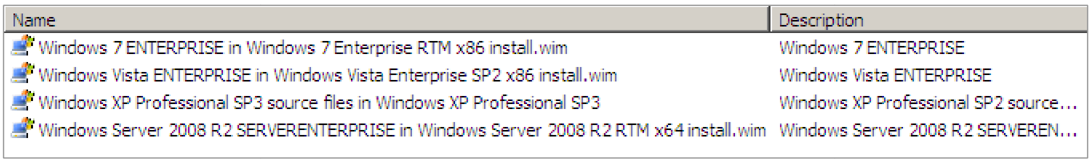


Specify the name of the directory that should be created for these operating system files.

Destination directory name: Windows 7 Enterprise RTM x86

5. Accept the defaults on the remaining pages and click **Finish**.

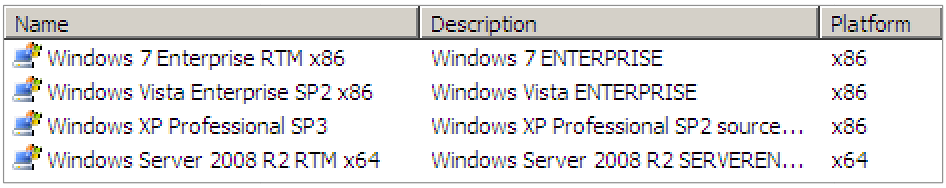
6. Repeat steps 1-5 for all the operating systems required. The operating system files will be copied to the MDT 2010 distribution share and will appear in the Deployment Workbench:



Name	Description
Windows 7 ENTERPRISE in Windows 7 Enterprise RTM x86 install.wim	Windows 7 ENTERPRISE
Windows Vista ENTERPRISE in Windows Vista Enterprise SP2 x86 install.wim	Windows Vista ENTERPRISE
Windows XP Professional SP3 source files in Windows XP Professional SP3	Windows XP Professional SP2 source…
Windows Server 2008 R2 SERVERENTERPRISE in Windows Server 2008 R2 RTM x64 install.wim	Windows Server 2008 R2 SERVEREN…

Note

As the default names can be unclear, it is recommended that each entry is renamed to use the same convention used for the source path in previous steps. The new names are shown in Figure 7:



Name	Description	Platform
Windows 7 Enterprise RTM x86	Windows 7 ENTERPRISE	x86
Windows Vista Enterprise SP2 x86	Windows Vista ENTERPRISE	x86
Windows XP Professional SP3	Windows XP Professional SP2 source…	x86
Windows Server 2008 R2 RTM x64	Windows Server 2008 R2 SERVEREN…	x64

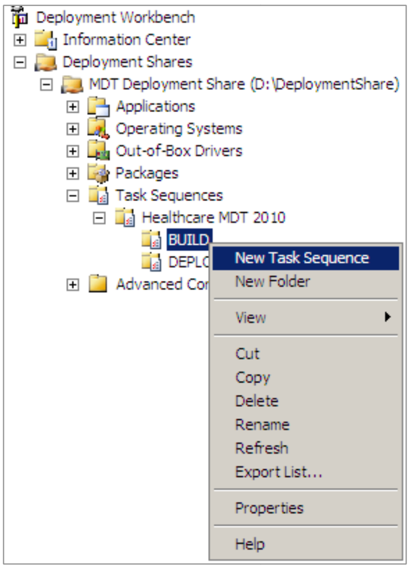
Figure 7: Deployment Workbench with Renamed Operating Systems

4.1.5 Creating a Task Sequence

After the operating system source files have been imported, a Task Sequence must be created which allows the installation of the operating system files imported in section 4.1.4.

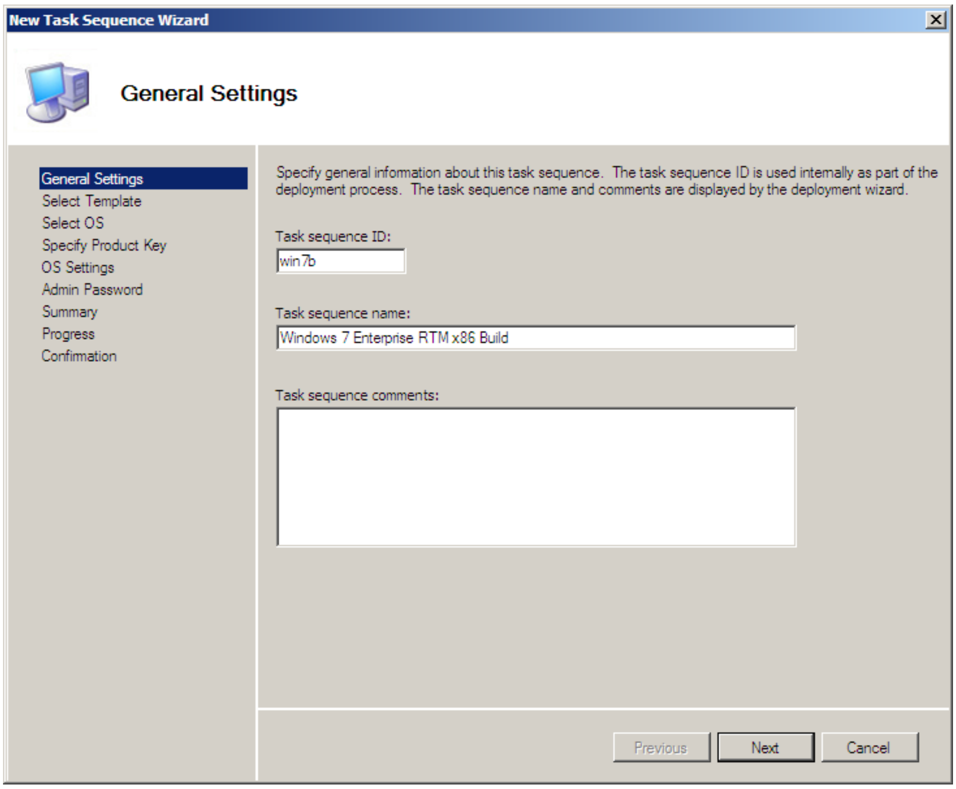
To create a Build Task Sequence:

1. In the Deployment Workbench, right-click **Task Sequences \ Healthcare MDT 2010 \ BUILD** and select **New Task Sequence**:



The **New Task Sequence Wizard** starts and displays the **General Settings** page.

2. Enter a **Task sequence ID**, **Task sequence name** and **Task sequence comments**. Click **Next**:



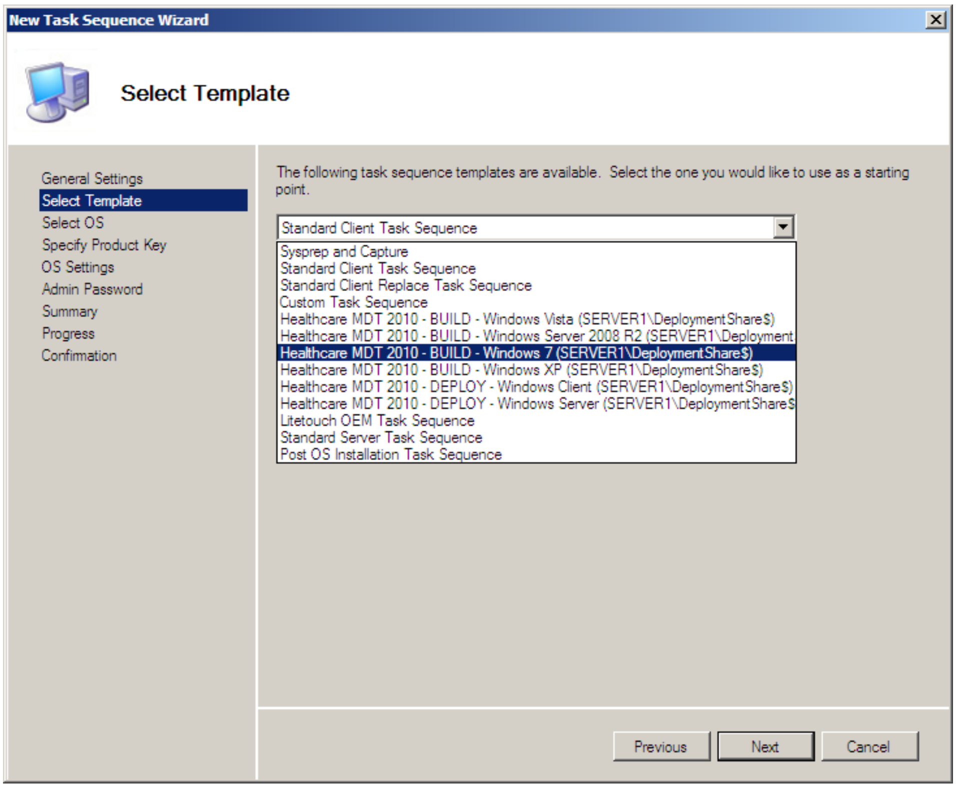
Specify general information about this task sequence. The task sequence ID is used internally as part of the deployment process. The task sequence name and comments are displayed by the deployment wizard.

Task sequence ID: win7b

Task sequence name: Windows 7 Enterprise RTM x86 Build

Task sequence comments:

3. On the **Select Template** page, select a corresponding Healthcare Extensions for MDT 2010 BUILD template and click **Next**:

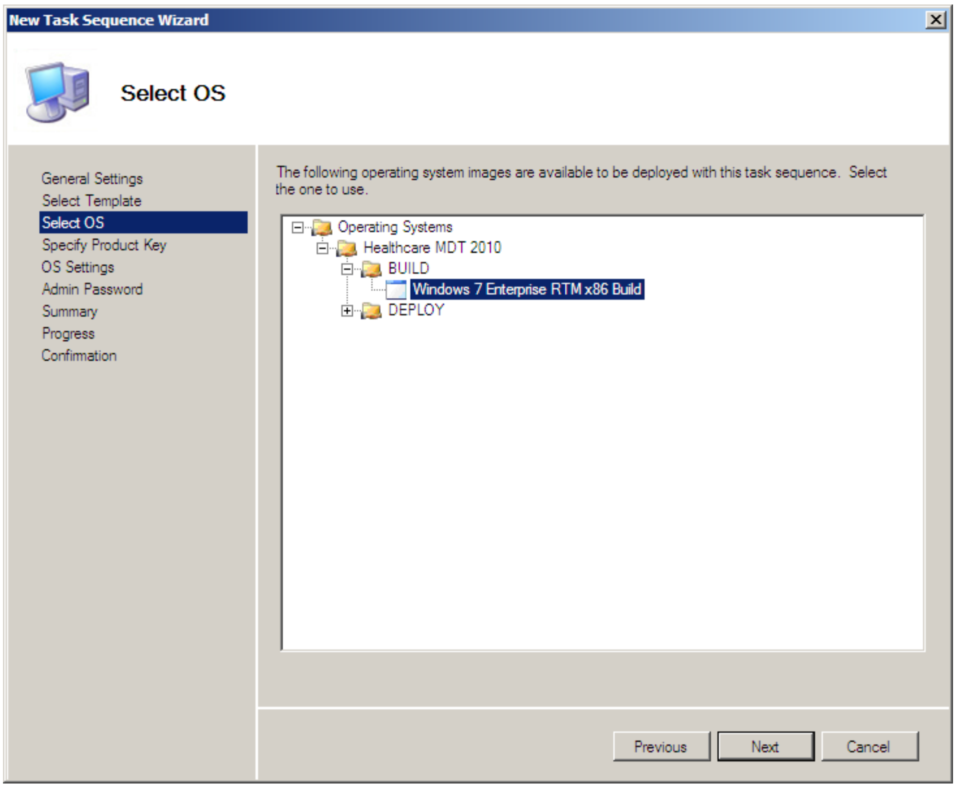


The following task sequence templates are available. Select the one you would like to use as a starting point.

Standard Client Task Sequence

- Sysprep and Capture
- Standard Client Task Sequence
- Standard Client Replace Task Sequence
- Custom Task Sequence
- Healthcare MDT 2010 - BUILD - Windows Vista (SERVER1\DeploymentShare$)
- Healthcare MDT 2010 - BUILD - Windows Server 2008 R2 (SERVER1\Deployment
- Healthcare MDT 2010 - BUILD - Windows 7 (SERVER1\DeploymentShare$)
- Healthcare MDT 2010 - BUILD - Windows XP (SERVER1\DeploymentShare$)
- Healthcare MDT 2010 - DEPLOY - Windows Client (SERVER1\DeploymentShare$)
- Healthcare MDT 2010 - DEPLOY - Windows Server (SERVER1\DeploymentShare$
- Litetouch OEM Task Sequence
- Standard Server Task Sequence
- Post OS Installation Task Sequence

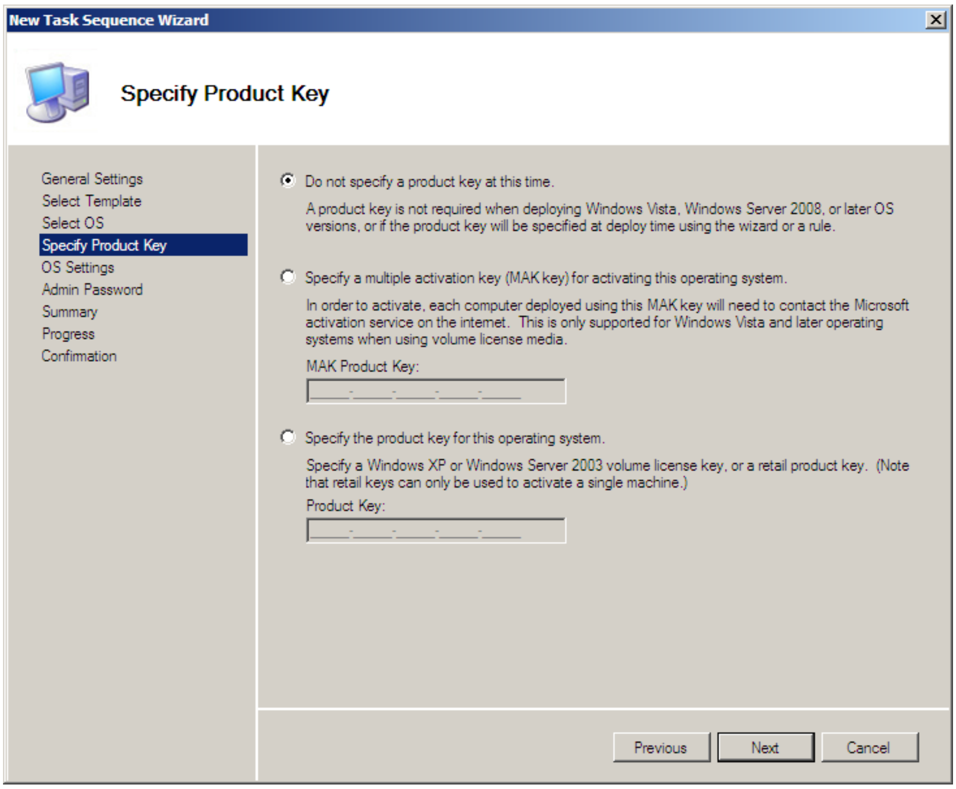
4. On the **Select OS** page, select the operating system previously imported and click **Next**:



The following operating system images are available to be deployed with this task sequence. Select the one to use.

- Operating Systems
  - Healthcare MDT 2010
    - BUILD
      - Windows 7 Enterprise RTM x86 Build
    - DEPLOY

5. On the **Specify Product Key** page, do one of the following:
   - If you are using Windows XP, enter the specific product key
   - If you are using Windows Vista, Windows 7 or Windows Server 2008, select **Do not specify a product key at this time.** Click Next:



Specify Product Key

● Do not specify a product key at this time.

A product key is not required when deploying Windows Vista, Windows Server 2008, or later OS versions, or if the product key will be specified at deploy time using the wizard or a rule.

○ Specify a multiple activation key (MAK key) for activating this operating system.

In order to activate, each computer deployed using this MAK key will need to contact the Microsoft activation service on the internet. This is only supported for Windows Vista and later operating systems when using volume license media.

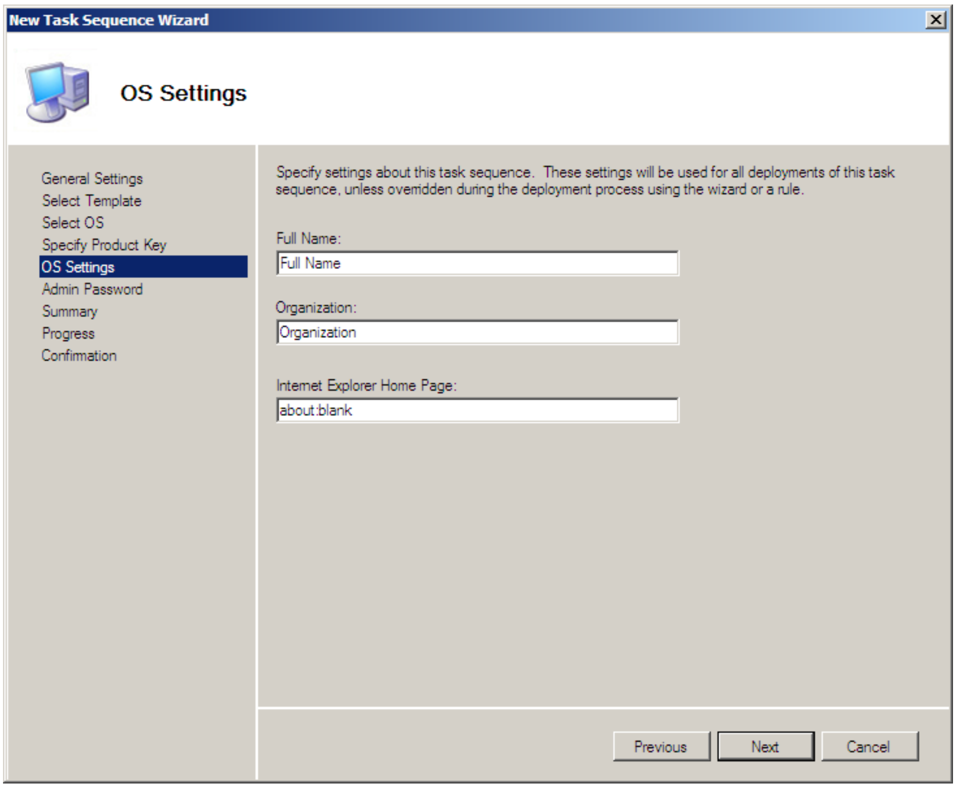
MAK Product Key:

○ Specify the product key for this operating system.

Specify a Windows XP or Windows Server 2003 volume license key, or a retail product key. (Note that retail keys can only be used to activate a single machine.)

Product Key:

6. On the **OS Settings** page, enter the **Full Name** (User N**ame), Organization and Internet Explorer Home Page** that you want to use for the build and click **Next**:



OS Settings

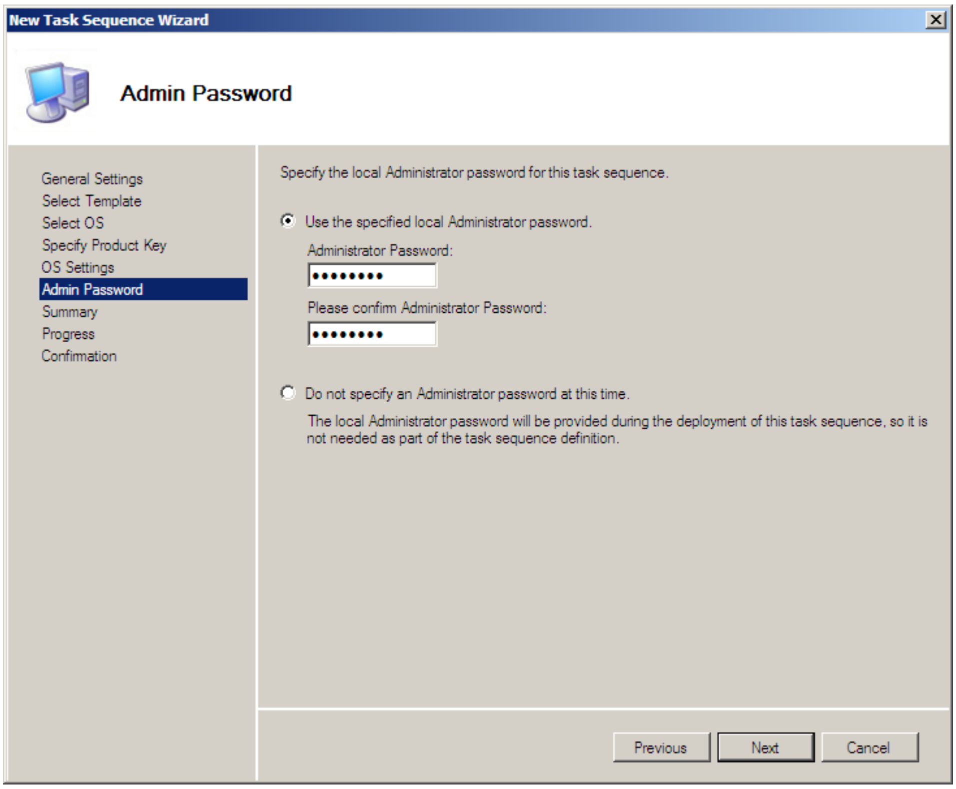
Specify settings about this task sequence. These settings will be used for all deployments of this task sequence, unless overridden during the deployment process using the wizard or a rule.

Full Name: Full Name

Organization: Organization

Internet Explorer Home Page: about:blank

7. On the **Admin Password** page, enter and confirm the local Administrator password that you want to be set on this Task Sequence, or click **Do not specify an Administrators password at this time**, and click **Next**:



New Task Sequence Wizard

Admin Password

Specify the local Administrator password for this task sequence.

● Use the specified local Administrator password.

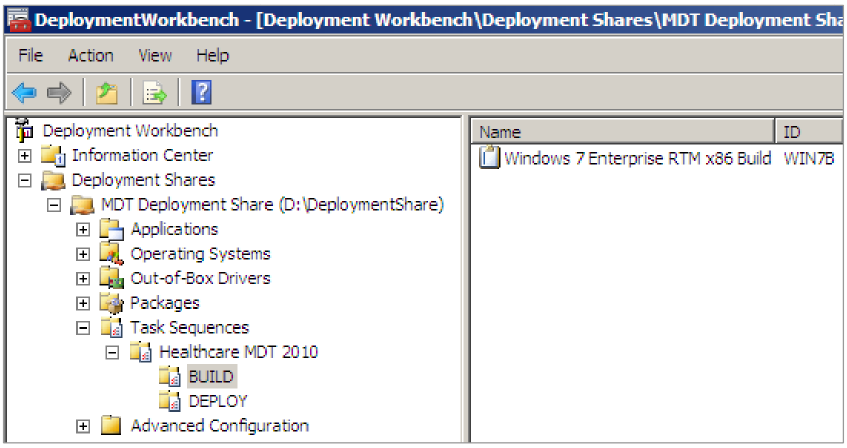
Administrator Password: ••••••••

Please confirm Administrator Password: ••••••••

○ Do not specify an Administrator password at this time.

The local Administrator password will be provided during the deployment of this task sequence, so it is not needed as part of the task sequence definition.

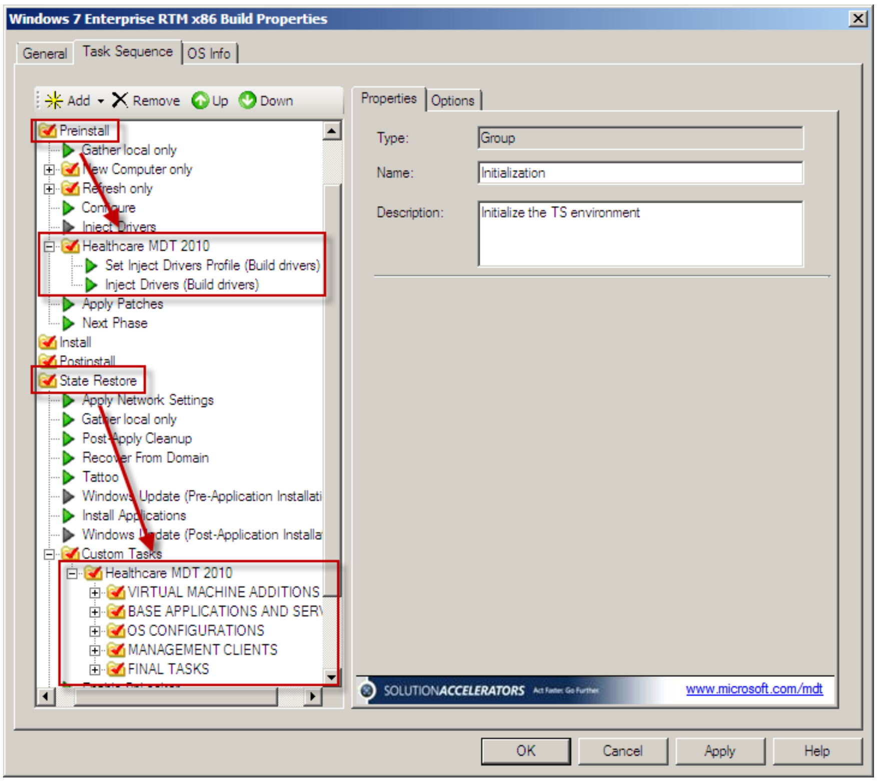
8. Accept the defaults on all remaining pages and click **Finish**. The task sequence will be created in the Deployment Workbench:



DeploymentWorkbench - [Deployment Workbench\Deployment Shares\MDT Deployment Sha

Name	ID
Windows 7 Enterprise RTM x86 Build	WIN7B

The custom Healthcare Extensions for MDT 2010 Task Sequence entries can be accessed by right-clicking the Task Sequence, selecting **Properties** and then the **Task Sequence** tab, as shown in Figure 8:

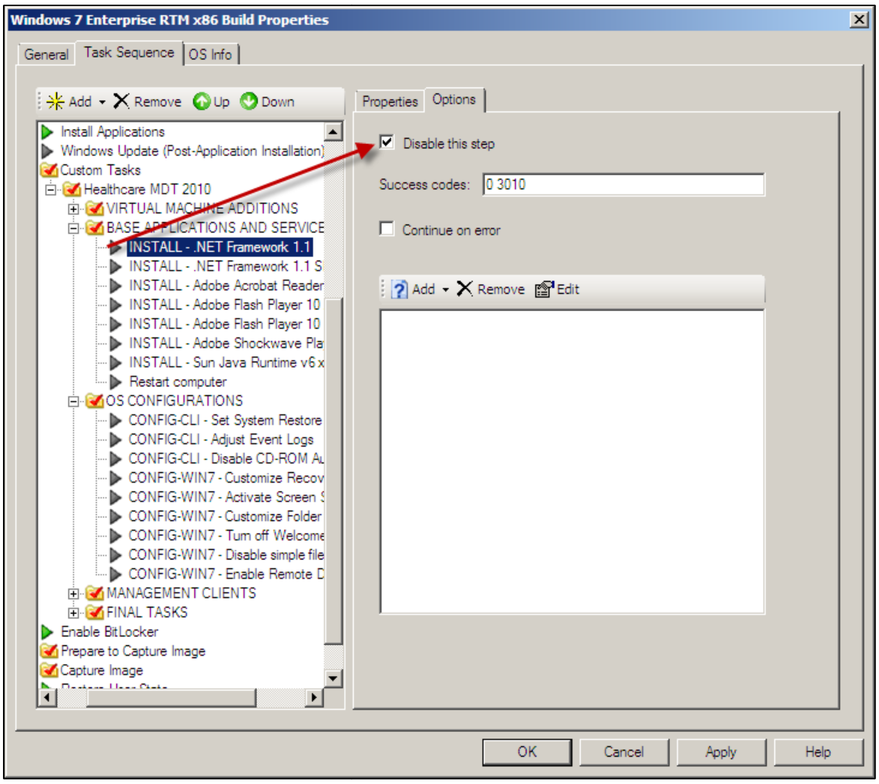


Preinstall
Gather local only
New Computer only
Refresh only
Configure
Inject Drivers
Healthcare MDT 2010
Set Inject Drivers Profile (Build drivers)
Inject Drivers (Build drivers)
Apply Patches
Next Phase
Install
Postinstall
State Restore
Apply Network Settings
Gather local only
Post-Apply Cleanup
Recover From Domain
Tattoo
Windows Update (Pre-Application Installati
Install Applications
Windows Update (Post-Application Installa
Custom Tasks
Healthcare MDT 2010
VIRTUAL MACHINE ADDITIONS
BASE APPLICATIONS AND SER\
OS CONFIGURATIONS
MANAGEMENT CLIENTS
FINAL TASKS

Type: Group
Name: Initialization
Description: Initialize the TS environment

Figure 8: Healthcare Extensions for MDT 2010 Windows 7 Task Sequence

By default, all Healthcare Extensions for MDT 2010 custom applications and scripts are disabled. Only those required by the IT Administrator should be re-enabled as shown in Figure 9:



Install Applications
Windows Update (Post-Application Installation)
Custom Tasks
Healthcare MDT 2010
VIRTUAL MACHINE ADDITIONS
BASE APPLICATIONS AND SERVICE
INSTALL - .NET Framework 1.1
INSTALL - .NET Framework 1.1 S
INSTALL - Adobe Acrobat Reader
INSTALL - Adobe Flash Player 10
INSTALL - Adobe Flash Player 10
INSTALL - Adobe Shockwave Pla
INSTALL - Sun Java Runtime v6 x
Restart computer
OS CONFIGURATIONS
CONFIG-CLI - Set System Restore
CONFIG-CLI - Adjust Event Logs
CONFIG-CLI - Disable CD-ROM Au
CONFIG-WIN7 - Customize Recov
CONFIG-WIN7 - Activate Screen S
CONFIG-WIN7 - Customize Folder
CONFIG-WIN7 - Turn off Welcome
CONFIG-WIN7 - Disable simple file
CONFIG-WIN7 - Enable Remote D
MANAGEMENT CLIENTS
FINAL TASKS
Enable BitLocker
Prepare to Capture Image
Capture Image

☑ Disable this step
Success codes: 0 3010
☐ Continue on error

Figure 9: Enabling Task Sequence Entries

4.2 Creating a Custom Image

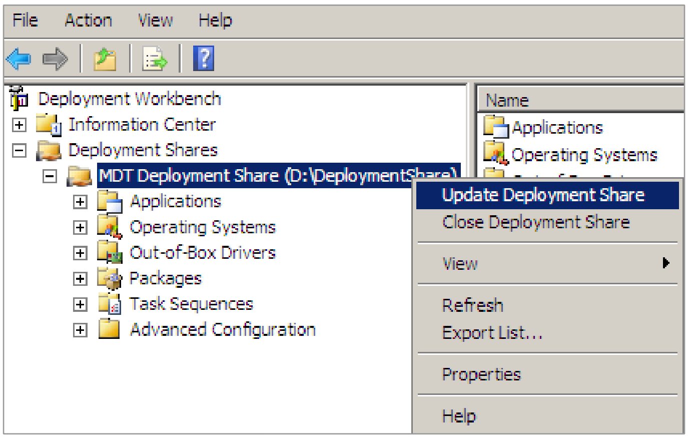
This section details how to create and capture a custom image. Once captured, an image can be deployed on a large scale using MDT 2010, or other deployment tools.

4.2.1 Creating the Windows PE Boot ISO File

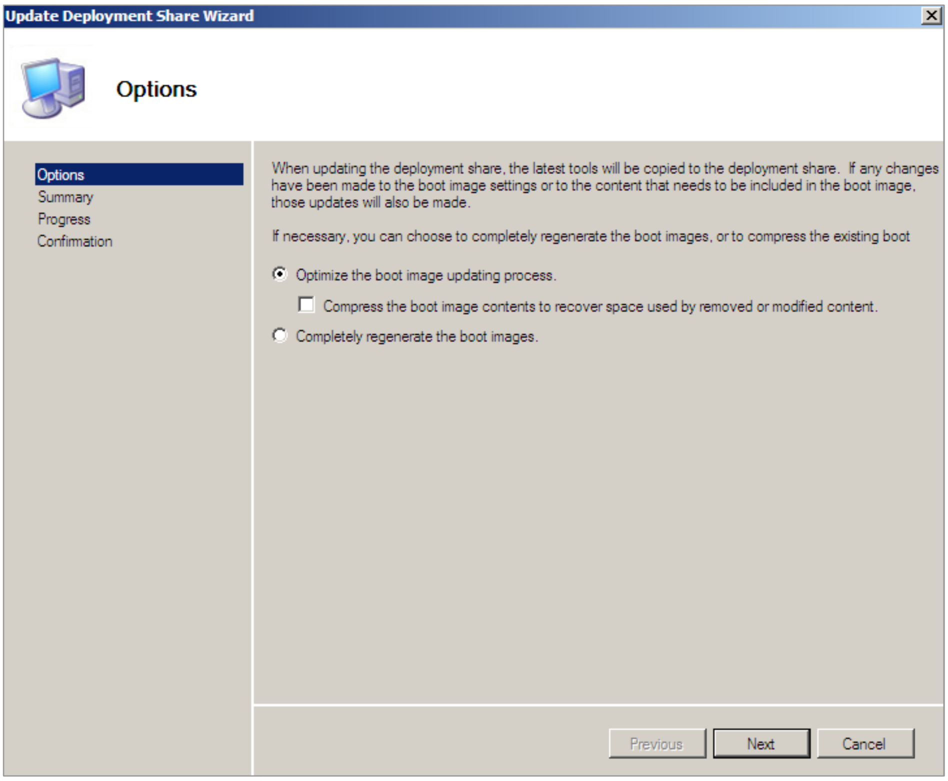
To initiate the build, a workstation must be started from a Windows PE MDT CD.

To create the ISO file for the CD:

1. In the Deployment Workbench, navigate to **Deployment Shares**, right-click the Deployment Share and select **Update Deployment Share**:



2. In the **Options** page, select **Optimize the boot image updating process** and click **Next**:



When updating the deployment share, the latest tools will be copied to the deployment share. If any changes have been made to the boot image settings or to the content that needs to be included in the boot image, those updates will also be made.

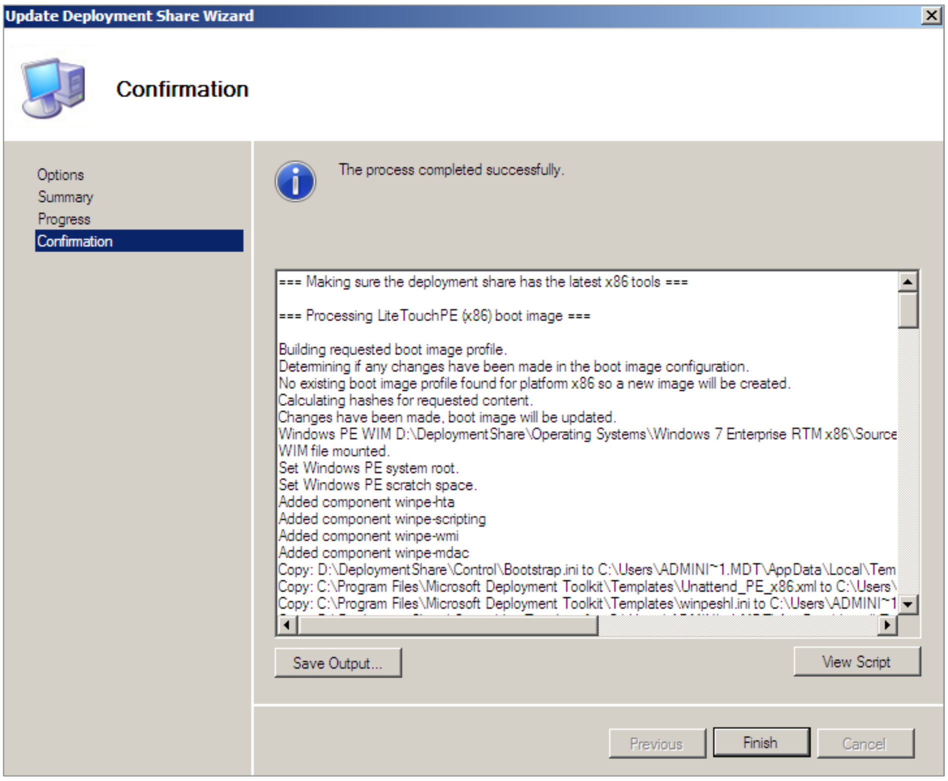
If necessary, you can choose to completely regenerate the boot images, or to compress the existing boot

● Optimize the boot image updating process.

□ Compress the boot image contents to recover space used by removed or modified content.

○ Completely regenerate the boot images.

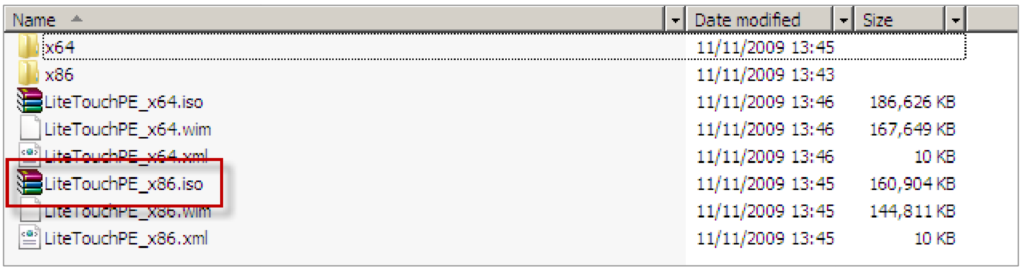
This will initiate a process to create the boot ISO files and may take several minutes:



The process completed successfully.

```
=== Making sure the deployment share has the latest x86 tools ===

=== Processing LiteTouchPE (x86) boot image ===

Building requested boot image profile.
Determining if any changes have been made in the boot image configuration.
No existing boot image profile found for platform x86 so a new image will be created.
Calculating hashes for requested content.
Changes have been made, boot image will be updated.
Windows PE WIM D:\DeploymentShare\Operating Systems\Windows 7 Enterprise RTM x86\Source
WIM file mounted.
Set Windows PE system root.
Set Windows PE scratch space.
Added component winpe-hta
Added component winpe-scripting
Added component winpe-wmi
Added component winpe-mdac
Copy: D:\DeploymentShare\Control\Bootstrap.ini to C:\Users\ADMINI~1.MDT\AppData\Local\Tem
Copy: C:\Program Files\Microsoft Deployment Toolkit\Templates\Unattend_PE_x86.xml to C:\Users\
Copy: C:\Program Files\Microsoft Deployment Toolkit\Templates\winpeshl.ini to C:\Users\ADMINI~1
```

3. Once the process has completed, browse to the **<Distribution>\Boot** folder:



Name	Date modified	Size
x64	11/11/2009 13:45	
x86	11/11/2009 13:43	
LiteTouchPE_x64.iso	11/11/2009 13:46	186,626 KB
LiteTouchPE_x64.wim	11/11/2009 13:46	167,649 KB
LiteTouchPE_x64.xml	11/11/2009 13:46	10 KB
LiteTouchPE_x86.iso	11/11/2009 13:45	160,904 KB
LiteTouchPE_x86.wim	11/11/2009 13:45	144,811 KB
LiteTouchPE_x86.xml	11/11/2009 13:45	10 KB

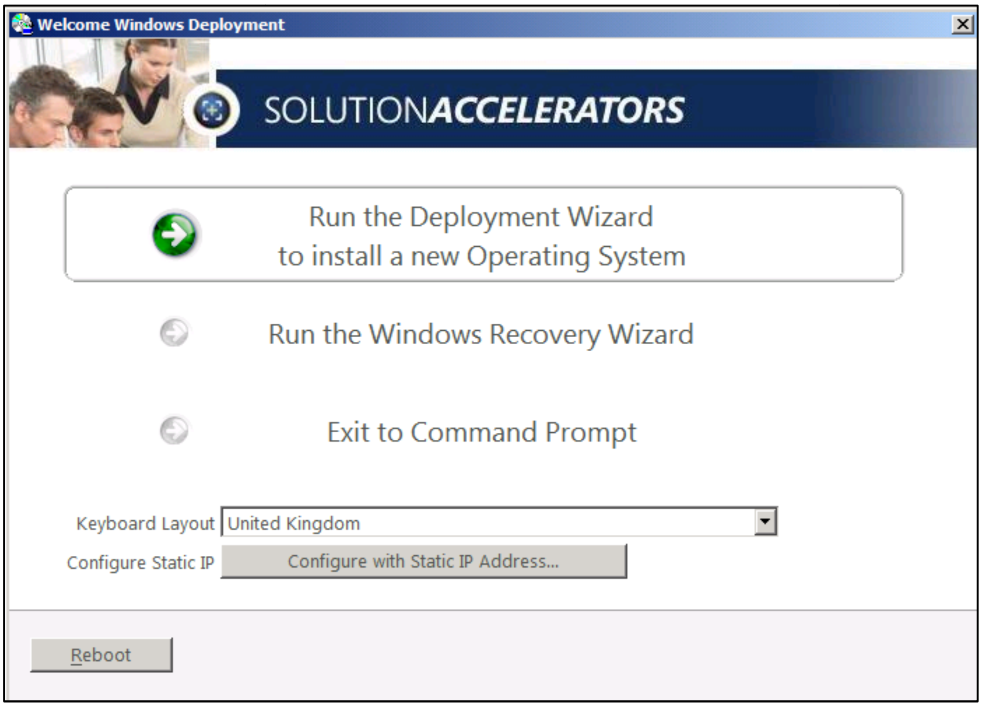
4. Create a bootable CD using the **LiteTouchPE_x86.iso** CD image file, or add the **LiteTouchPE_x86.wim** file to a WDS server boot section.

4.2.2 Creating and Capturing the Custom Image

This section lists the steps required to use the Windows PE CD and MDT 2010 to create and capture a build in Windows Image (WIM) format. Once an image has been captured, it can be imported into MDT 2010 for large-scale deployment, as detailed in section 6.1.

To create and capture a build in WIM format:

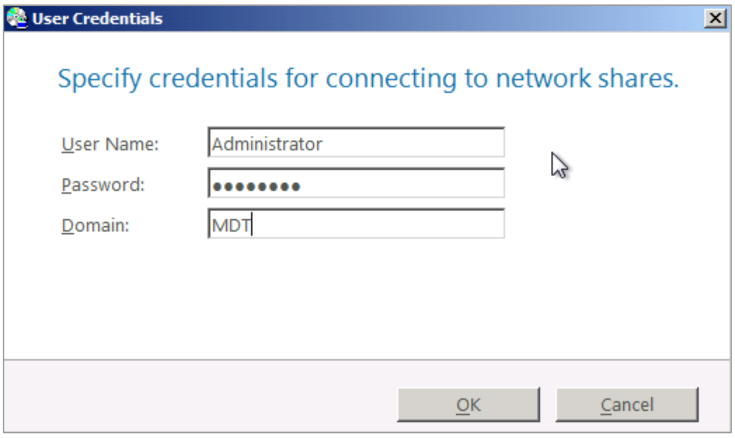
1. Start the target workstation from the Windows PE CD created in section 4.2.1 and select **Run the Deployment Wizard to install a new Operating System**:


Welcome Windows Deployment
SOLUTIONACCELERATORS
Run the Deployment Wizard to install a new Operating System
Run the Windows Recovery Wizard
Exit to Command Prompt
Configure with Static IP Address...
Reboot

Keyboard Layout: United Kingdom

Configure Static IP

2. Enter the credentials that will be used to access the Deployment Share and click **OK**:



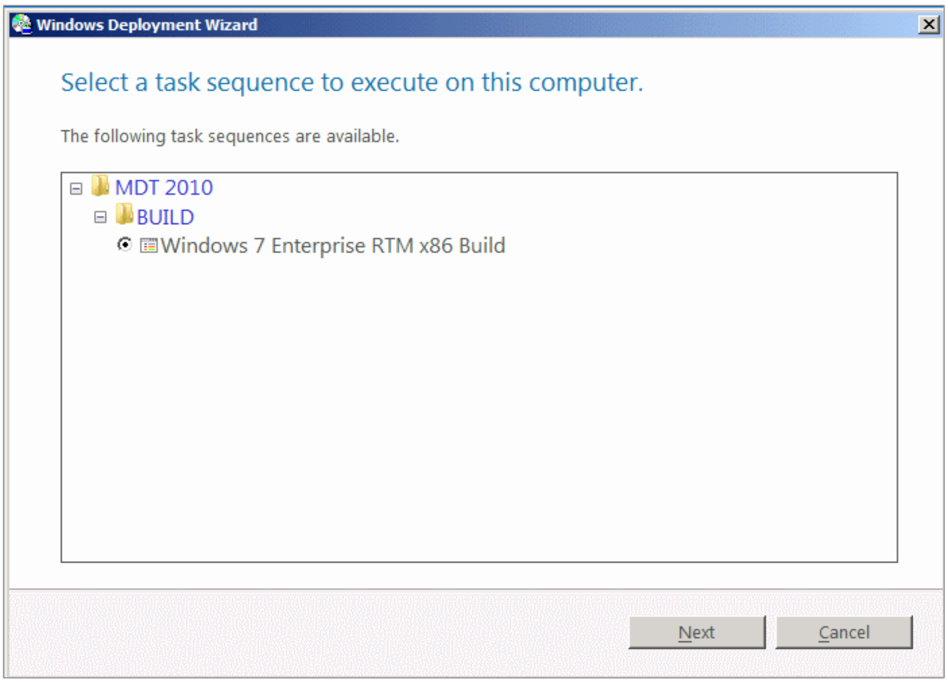
Specify credentials for connecting to network shares.

User Name: Administrator

Password: ●●●●●●●●

Domain: MDT

3. Select the Task Sequence to install and click **Next**:

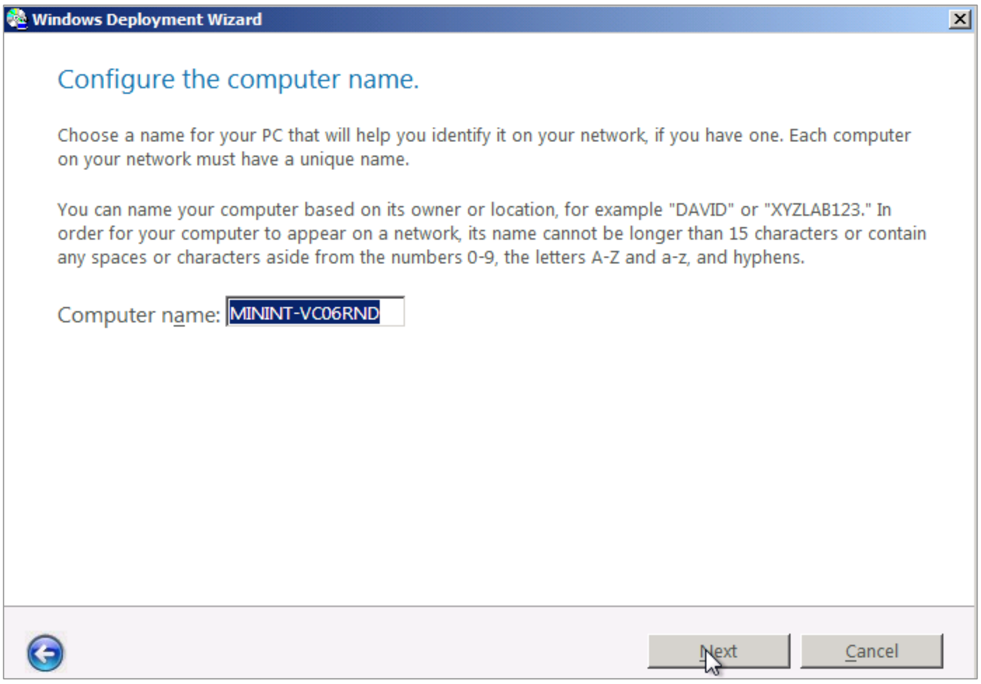


Select a task sequence to execute on this computer.

The following task sequences are available.

- MDT 2010
  - BUILD
    - ● Windows 7 Enterprise RTM x86 Build

4. Leave the **Computer name** as the default name and click **Next**:



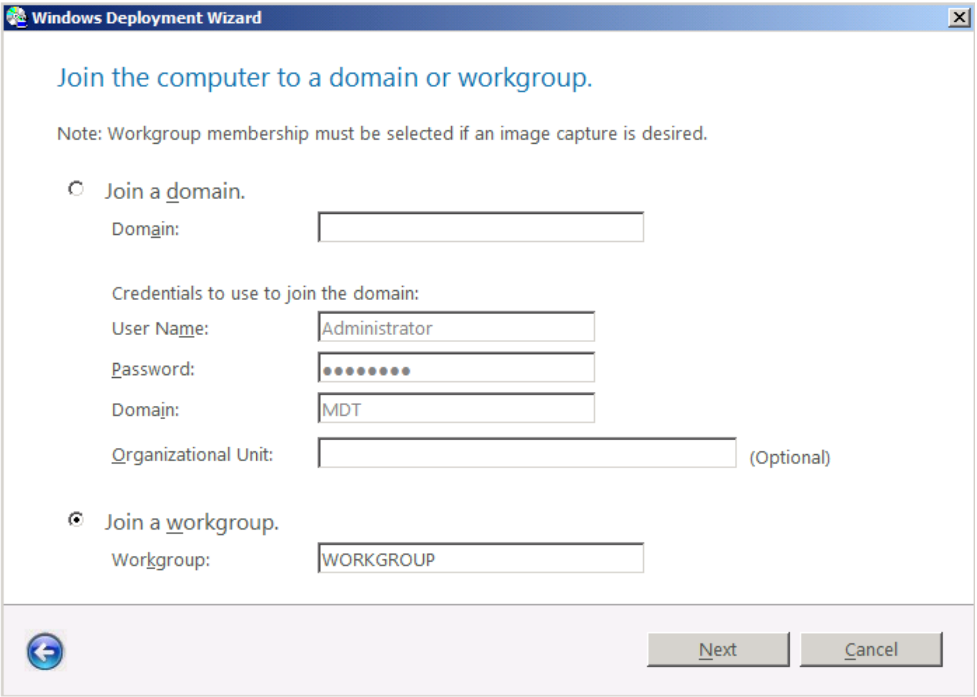
Configure the computer name.

Choose a name for your PC that will help you identify it on your network, if you have one. Each computer on your network must have a unique name.

You can name your computer based on its owner or location, for example "DAVID" or "XYZLAB123." In order for your computer to appear on a network, its name cannot be longer than 15 characters or contain any spaces or characters aside from the numbers 0-9, the letters A-Z and a-z, and hyphens.

Computer name: MININT-VC06RND

5. Ensure the computer is in a **Workgroup** named 'WORKGROUP**'** and click **Next**:



Windows Deployment Wizard

Join the computer to a domain or workgroup.

Note: Workgroup membership must be selected if an image capture is desired.

○ Join a domain.

Domain:

Credentials to use to join the domain:

User Name: Administrator

Password: ••••••••

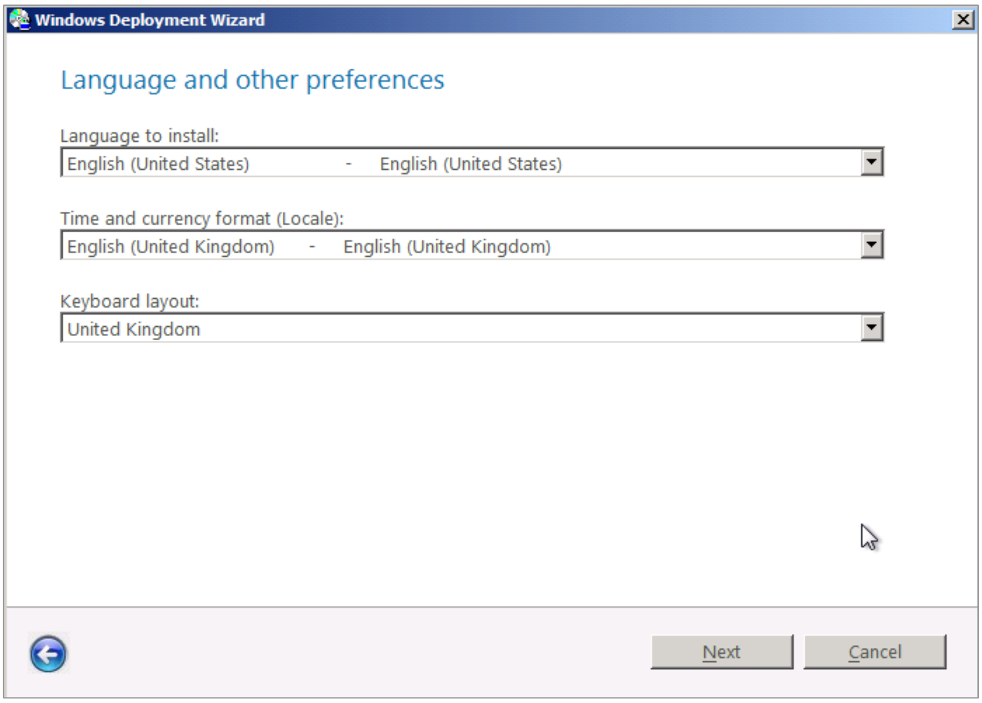
Domain: MDT

Organizational Unit: (Optional)

● Join a workgroup.

Workgroup: WORKGROUP

6. Ensure that the **Time and currency format (Locale)** and **Keyboard layout** settings are set **appropriately**. Click **Next**:



Windows Deployment Wizard

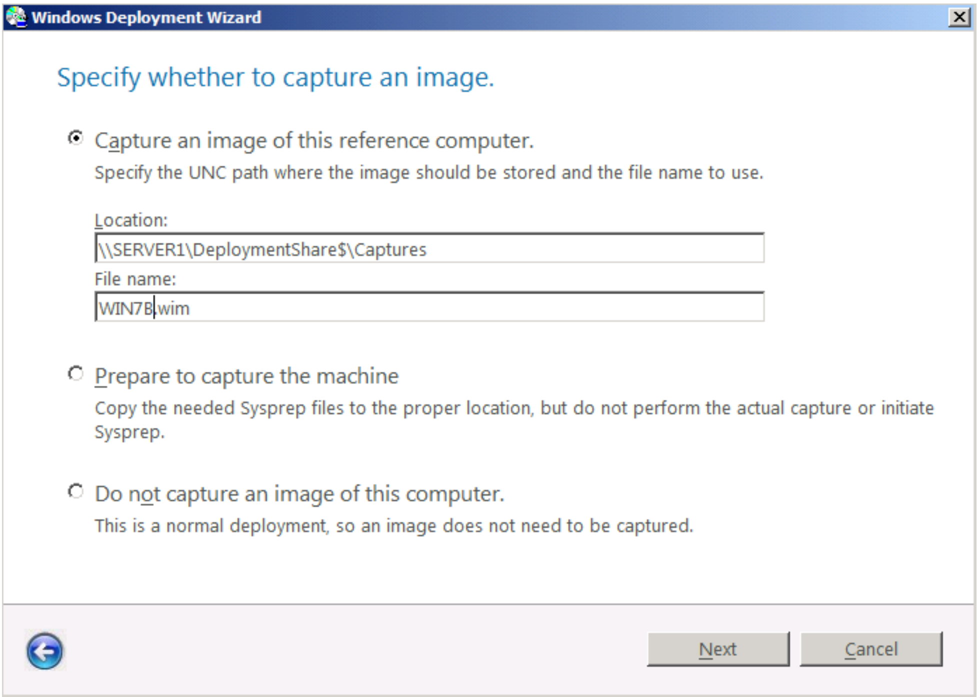
Language and other preferences

Language to install: English (United States) - English (United States)

Time and currency format (Locale): English (United Kingdom) - English (United Kingdom)

Keyboard layout: United Kingdom

7. Select **Capture an image of this reference computer** and keep the default values for the other settings. Click **Next**:



Specify whether to capture an image.

● Capture an image of this reference computer.
Specify the UNC path where the image should be stored and the file name to use.

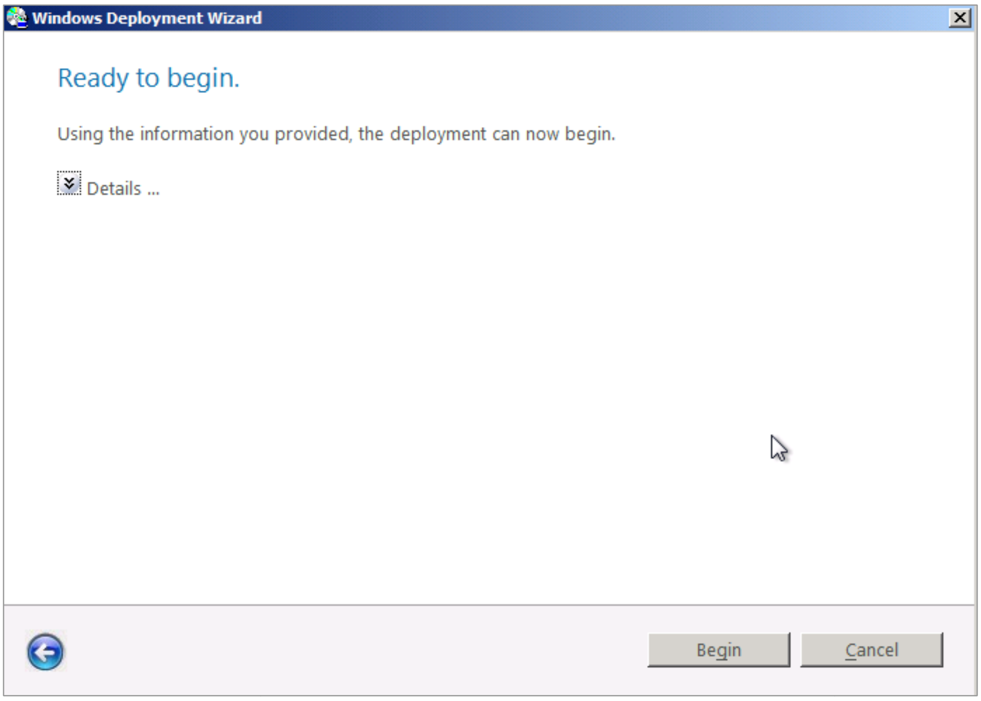
Location:
\\SERVER1\DeploymentShare$\Captures

File name:
WIN7B.wim

○ Prepare to capture the machine
Copy the needed Sysprep files to the proper location, but do not perform the actual capture or initiate Sysprep.

○ Do not capture an image of this computer.
This is a normal deployment, so an image does not need to be captured.

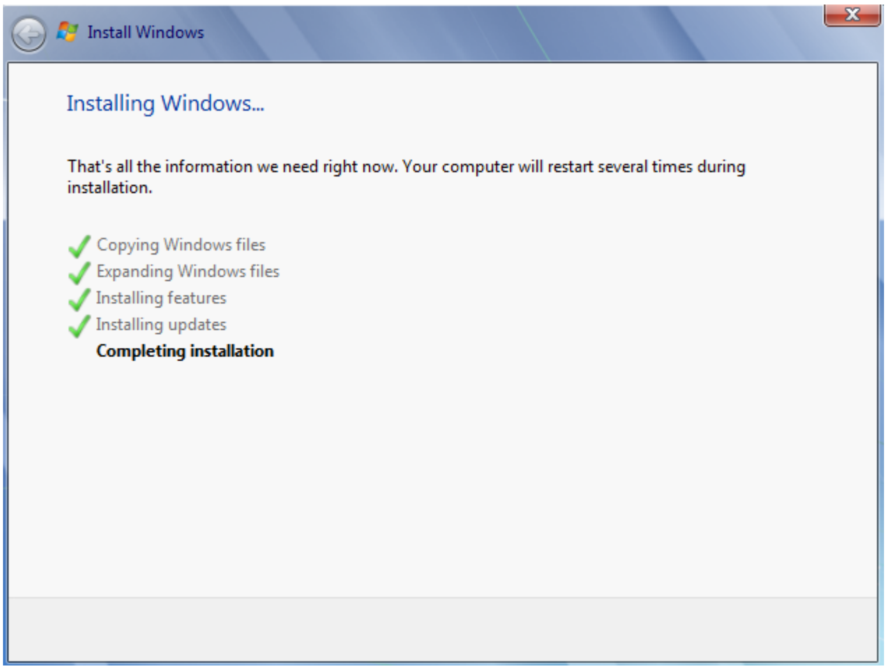
8. On the **Ready to begin** page, click **Begin**:



Ready to begin.

Using the information you provided, the deployment can now begin.

The setup program for the version of Windows being installed will appear:



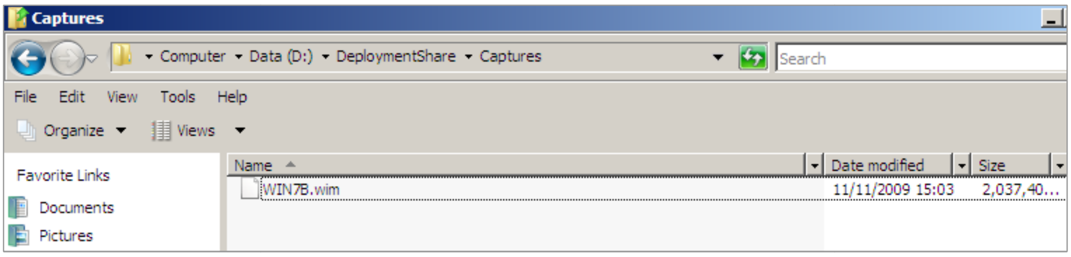
Installing Windows...

That's all the information we need right now. Your computer will restart several times during installation.

- ✓ Copying Windows files
- ✓ Expanding Windows files
- ✓ Installing features
- ✓ Installing updates
- **Completing installation**

The build process begins, and takes around thirty to sixty minutes, depending on the speed of the workstation and the number of applications selected.

Once the build is complete, an image of the machine is automatically captured in WIM format and placed in the **<Distribution>\Captures** folder:



Captures
Computer ▸ Data (D:) ▸ DeploymentShare ▸ Captures
Search
File Edit View Tools Help
Organize Views
Favorite Links
Documents
Pictures

Name	Date modified	Size
WIN7B.wim	11/11/2009 15:03	2,037,40...

Note

Once an image has been captured, it can be re-imported into MDT 2010 as a new operating system for large-scale deployment, as detailed in section 6.1.

4.3 Adding Build Drivers

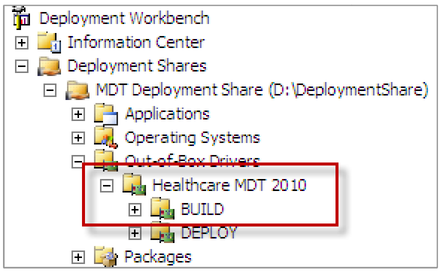
The Build image should be as small and generic as possible so that it can be used as the basis for deployment to a wide range of hardware types. A single modern driver can be up to 200MB in size, therefore only drivers (usually network and storage drivers) required to perform the Build should be installed during this phase. Hardware-specific drivers will be dynamically installed during the Deploy phase.

Before adding any drivers to the Build process, a full Build should be attempted, as detailed in section 4.2. If the Build completes successfully, **no additional Build drivers should be added**.

Usually, the only additional drivers required to complete the Build process are:

- Networking – the Build process requires access to the network
- Storage – the Build process requires access to the hard disk

Build drivers should be located in the **Out-of-Box Drivers \ Healthcare MDT 2010 \ BUILD** folder, as shown in Figure 10:

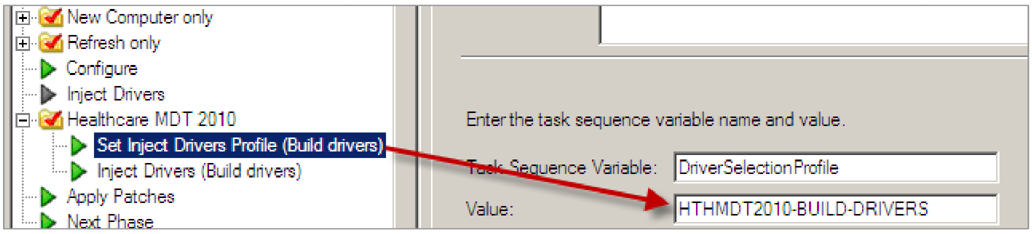


Deployment Workbench
Information Center
Deployment Shares
MDT Deployment Share (D:\DeploymentShare)
Applications
Operating Systems
Out-of-Box Drivers
Healthcare MDT 2010
BUILD
DEPLOY
Packages

Figure 10: Location for Build Drivers

The location for the Build Drivers has an associated Selection Profile, located in **Advanced Configuration \ Selection Profiles \ HTHMDT2010-BUILD-DRIVERS**.

This Selection Profile is then referenced in the Healthcare Extensions for MDT 2010 **BUILD** Task Sequence templates, as shown in Figure 11 and detailed in section 4.1.5:



New Computer only
Refresh only
Configure
Inject Drivers
Healthcare MDT 2010
Set Inject Drivers Profile (Build drivers)
Inject Drivers (Build drivers)
Apply Patches
Next Phase

Enter the task sequence variable name and value.

Task Sequence Variable: DriverSelectionProfile

Value: HTHMDT2010-BUILD-DRIVERS

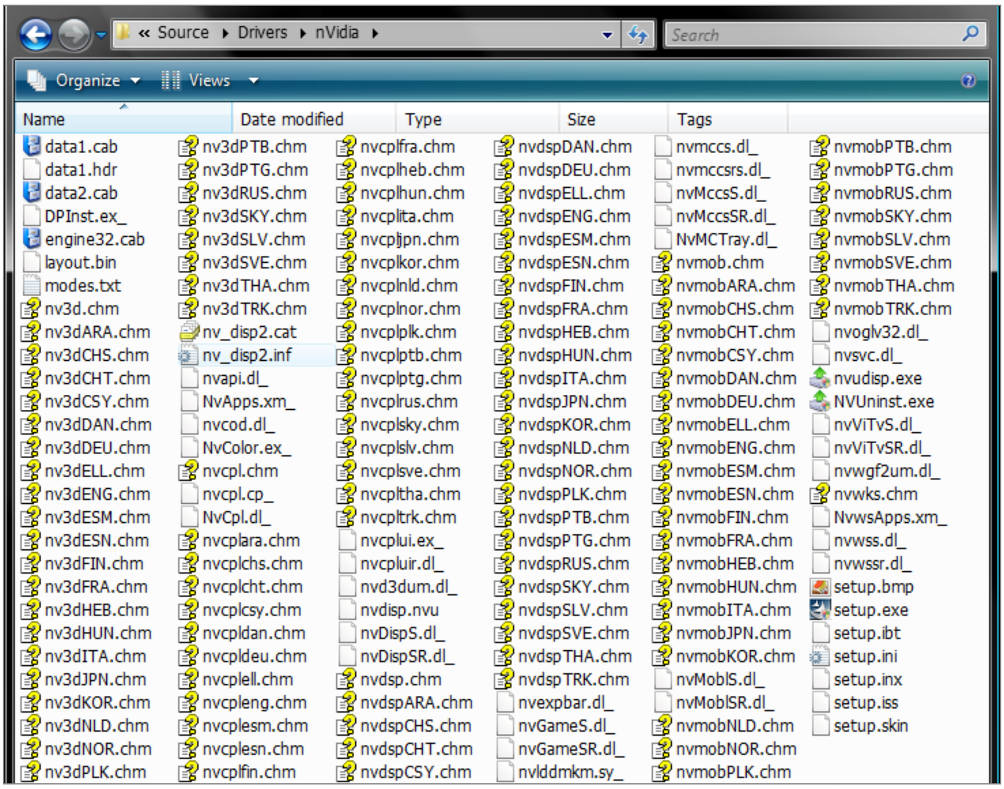
Figure 11: Build Drivers Task Sequence Properties

The combination of these drivers, Selection Profiles and Task Sequence properties ensure that only drivers located in the **Out-of-Box Drivers \ Healthcare MDT 2010 \ BUILD** folder are used during the Build process.

4.3.1 Adding Additional Build Drivers

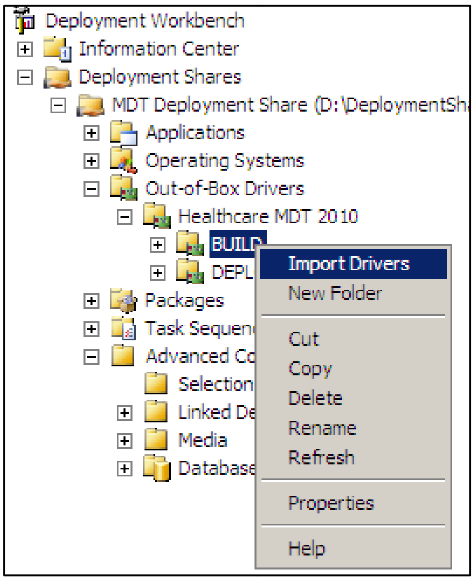
If the Build process is not initially successful and does require an additional driver, it can be added as follows:

1. Copy the extracted driver files to a temporary location on the MDT server:



data1.cab
data1.hdr
data2.cab
DPInst.ex_
engine32.cab
layout.bin
modes.txt
nv3d.chm
nv3dARA.chm
nv3dCHS.chm
nv3dCHT.chm
nv3dCSY.chm
nv3dDAN.chm
nv3dDEU.chm
nv3dELL.chm
nv3dENG.chm
nv3dESM.chm
nv3dESN.chm
nv3dFIN.chm
nv3dFRA.chm
nv3dHEB.chm
nv3dHUN.chm
nv3dITA.chm
nv3dJPN.chm
nv3dKOR.chm
nv3dNLD.chm
nv3dNOR.chm
nv3dPLK.chm
nv3dPTB.chm
nv3dPTG.chm
nv3dRUS.chm
nv3dSKY.chm
nv3dSLV.chm
nv3dSVE.chm
nv3dTHA.chm
nv3dTRK.chm
nv_disp2.cat
nv_disp2.inf
nvapi.dl_
NvApps.xm_
nvcod.dl_
NvColor.ex_
nvcpl.chm
nvcpl.cp_
NvCpl.dl_
nvcplara.chm
nvcplchs.chm
nvcplcht.chm
nvcplcsy.chm
nvcpldan.chm
nvcpldeu.chm
nvcplell.chm
nvcpleng.chm
nvcplesm.chm
nvcplesn.chm
nvcplfin.chm
nvcplfra.chm
nvcplheb.chm
nvcplhun.chm
nvcplita.chm
nvcpljpn.chm
nvcplkor.chm
nvcplnld.chm
nvcplnor.chm
nvcplplk.chm
nvcplptb.chm
nvcplptg.chm
nvcplrus.chm
nvcplsky.chm
nvcplslv.chm
nvcplsve.chm
nvcpltha.chm
nvcpltrk.chm
nvcplui.ex_
nvcpluir.dl_
nvd3dum.dl_
nvdisp.nvu
nvDispS.dl_
nvDispSR.dl_
nvdsp.chm
nvdspARA.chm
nvdspCHS.chm
nvdspCHT.chm
nvdspCSY.chm
nvdspDAN.chm
nvdspDEU.chm
nvdspELL.chm
nvdspENG.chm
nvdspESM.chm
nvdspESN.chm
nvdspFIN.chm
nvdspFRA.chm
nvdspHEB.chm
nvdspHUN.chm
nvdspITA.chm
nvdspJPN.chm
nvdspKOR.chm
nvdspNLD.chm
nvdspNOR.chm
nvdspPLK.chm
nvdspPTB.chm
nvdspPTG.chm
nvdspRUS.chm
nvdspSKY.chm
nvdspSLV.chm
nvdspSVE.chm
nvdspTHA.chm
nvdspTRK.chm
nvexpbar.dl_
nvGameS.dl_
nvGameSR.dl_
nvlddmkm.sy_
nvmccs.dl_
nvmccsrs.dl_
nvMccsS.dl_
nvMccsSR.dl_
NvMCTray.dl_
nvmob.chm
nvmobARA.chm
nvmobCHS.chm
nvmobCHT.chm
nvmobCSY.chm
nvmobDAN.chm
nvmobDEU.chm
nvmobELL.chm
nvmobENG.chm
nvmobESM.chm
nvmobESN.chm
nvmobFIN.chm
nvmobFRA.chm
nvmobHEB.chm
nvmobHUN.chm
nvmobITA.chm
nvmobJPN.chm
nvmobKOR.chm
nvMobIS.dl_
nvMobISR.dl_
nvmobNLD.chm
nvmobNOR.chm
nvmobPLK.chm
nvmobPTB.chm
nvmobPTG.chm
nvmobRUS.chm
nvmobSKY.chm
nvmobSLV.chm
nvmobSVE.chm
nvmobTHA.chm
nvmobTRK.chm
nvoglv32.dl_
nvsvc.dl_
nvudisp.exe
NVUninst.exe
nvViTvS.dl_
nvViTvSR.dl_
nvwgf2um.dl_
nvwks.chm
NvwsApps.xm_
nvwss.dl_
nvwssr.dl_
setup.bmp
setup.exe
setup.ibt
setup.ini
setup.inx
setup.iss
setup.skin

2. Right-click on the **Out-of-Box Drivers \ Healthcare MDT 2010 \ BUILD** folder and select **Import Drivers**:

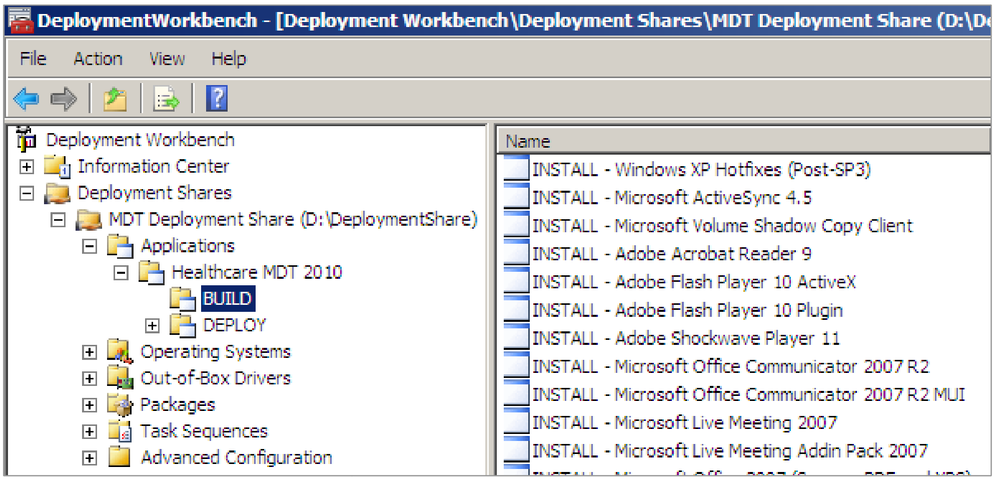


3. Accept the defaults on the remaining pages and click **Finish**. The newly imported drivers will appear.

4. Rebuild the Windows PE Boot ISO (see section 4.2.1) and start the Build process to ensure that the drivers have been correctly added.

4.4 Healthcare Extensions for MDT 2010 Default Applications

Healthcare Extensions for MDT 2010 is supplied preconfigured with a number of applications, as shown in Figure 12:


DeploymentWorkbench - [Deployment Workbench\Deployment Shares\MDT Deployment Share (D:\D
File
Action
View
Help

- Deployment Workbench
  - Information Center
  - Deployment Shares
    - MDT Deployment Share (D:\DeploymentShare)
      - Applications
        - Healthcare MDT 2010
          - BUILD
          - DEPLOY
      - Operating Systems
      - Out-of-Box Drivers
      - Packages
      - Task Sequences
      - Advanced Configuration

Name
INSTALL - Windows XP Hotfixes (Post-SP3)
INSTALL - Microsoft ActiveSync 4.5
INSTALL - Microsoft Volume Shadow Copy Client
INSTALL - Adobe Acrobat Reader 9
INSTALL - Adobe Flash Player 10 ActiveX
INSTALL - Adobe Flash Player 10 Plugin
INSTALL - Adobe Shockwave Player 11
INSTALL - Microsoft Office Communicator 2007 R2
INSTALL - Microsoft Office Communicator 2007 R2 MUI
INSTALL - Microsoft Live Meeting 2007
INSTALL - Microsoft Live Meeting Addin Pack 2007

Figure 12: Applications Screen in the Deployment Workbench

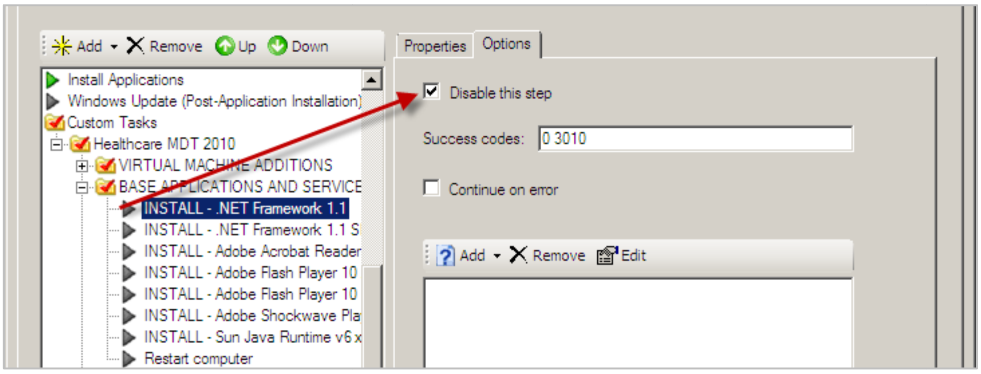
These applications point to source files contained in the **<Distribution>\Applications** folder, and are referenced in the Healthcare Extensions for MDT 2010 Task Sequence templates for each operating system.

> **Note**
>
> Not all applications made available through the installation of Healthcare Extensions for MDT 2010 are placed by default into the relevant Task Sequence. If required, these applications can be installed by adding the application installation via the Task Sequence properties.

Before an application can be used, the source files must be obtained and extracted. Instructions for doing this can be found in a README.txt file in each application folder in **<Distribution> \ Applications \ <APPLICATION>**. For example, for **<Distribution> \ Applications \ INSTALL - .NET Framework 1.1**, the README.txt file contains the following instructions:

```
.NET Framework v1.1

1. Obtain the installation redistributable file from the Microsoft Download
Website:

    http://www.microsoft.com/downloads/details.aspx?familyid=262D25E3-F589-4842-
8157-034D1E7CF3A3&displaylang=en

2. Extract the files from the downloaded file using the following command:

    dotnetfx.exe /t:<full path> /c

3. Copy the extracted files to the folder:

    <Distribution>\Applications\INSTALL - .NET Framework 1.1\Source
```

Once the source files have been installed, the application can be added (if not already present) and enabled in a Task Sequence, as shown in Figure 13:



- Install Applications
- Windows Update (Post-Application Installation)
- Custom Tasks
  - Healthcare MDT 2010
    - VIRTUAL MACHINE ADDITIONS
    - BASE APPLICATIONS AND SERVICE
      - INSTALL - .NET Framework 1.1
      - INSTALL - .NET Framework 1.1 S
      - INSTALL - Adobe Acrobat Reader
      - INSTALL - Adobe Flash Player 10
      - INSTALL - Adobe Flash Player 10
      - INSTALL - Adobe Shockwave Pla
      - INSTALL - Sun Java Runtime v6.x
      - Restart computer

☑ Disable this step

Success codes: 0 3010

☐ Continue on error

Figure 13: Enabling an Application

Table 4 lists the applications made available through Healthcare Extensions for MDT 2010, and indicates those that are placed by default within the Task Sequence for the relevant operating system:

Application Folder Name	Windows XP	Windows Vista	Windows 7	Windows Server 2008
INSTALL - .NET Framework 1.1	✓	✓	✓	✓
INSTALL - .NET Framework 1.1 SP1	✓	✓	✓	✓
INSTALL - .NET Framework 3.5 SP1	✓	✓		✓
INSTALL - Adobe Acrobat Reader 9	✓	✓	✓	
INSTALL - Adobe Flash Player 10 ActiveX				
INSTALL - Adobe Flash Player 10 Plugin				
INSTALL - Adobe Shockwave Player 11				
INSTALL - Microsoft ActiveSync 4.5				
INSTALL - Microsoft App-V 4.5 CU1 Client				
INSTALL - Microsoft CAPICOM 2.1.0.2				
INSTALL - Microsoft Live Meeting 2007 (nocache)				
INSTALL - Microsoft Live Meeting Addin Pack 2007 (nocache)				
INSTALL - Microsoft MSXML 6.0 SP1 x86				
INSTALL - Microsoft Office 2003 Primary Interop Assemblies		✓		
INSTALL - Microsoft Office 2007 (Save as PDF and XPS)				
INSTALL - Microsoft Office 2007 Primary Interop Assemblies		✓		
INSTALL - Microsoft Office Communicator 2007 R2				
INSTALL - Microsoft Office Communicator 2007 R2 MUI				
INSTALL - Microsoft SCCM 2007 Client	✓	✓	✓	
INSTALL - Microsoft SMS 2003 Client	✓	✓	✓	
INSTALL - Microsoft Virtual Machine Additions	✓	✓	✓	✓
INSTALL - Microsoft Visio Viewer				
INSTALL - Microsoft Volume Shadow Copy Client				

Application Folder Name	Windows XP	Windows Vista	Windows 7	Windows Server 2008
INSTALL - Microsoft VS2005 Tools for Office				
INSTALL - Microsoft VS2005 Tools for Office SE Runtime	✓	✓	✓	
INSTALL - Microsoft Windows Installer 3.1				
INSTALL - Microsoft Windows Search 4.0				
INSTALL - Sun Java Runtime v6 x86	✓	✓	✓	
INSTALL - Windows XP Hotfixes (Post-SP3)	✓			

Table 4: Application Installations

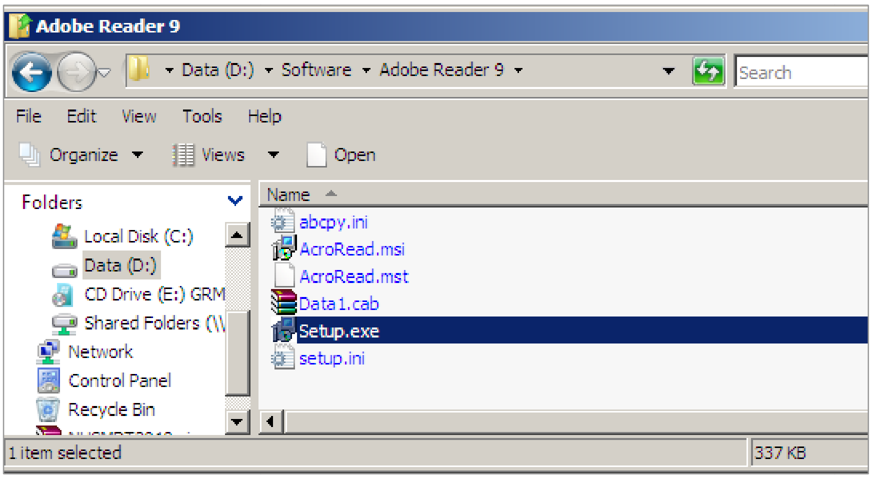
4.5 Adding Applications

To add a new application to Healthcare Extensions for MDT 2010:

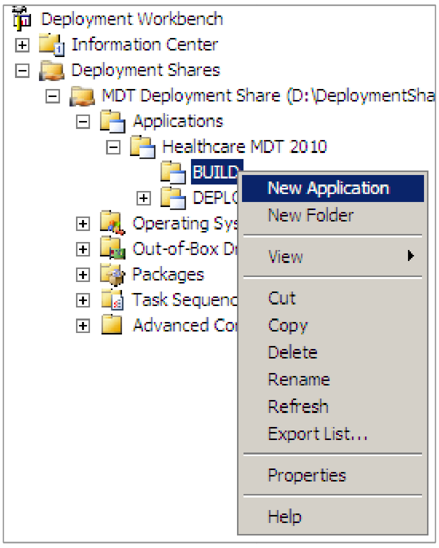
- The application source files or media must be available
- A set of command-line switches must be determined so that the application can be installed without the need for user interaction. For information on how to silently install applications, refer to the guidance provided in the document *Automated Build Application Integration* **{R1}**.

To add an application to Healthcare Extensions for MDT 2010:

1. Follow the instructions in APPENDIX B to obtain the source files for Adobe Reader and to customise its settings. At the end of this process, a folder with the source files will be available, as shown below. For this example, the folder used is **D:\Software \ Adobe Reader 9**:

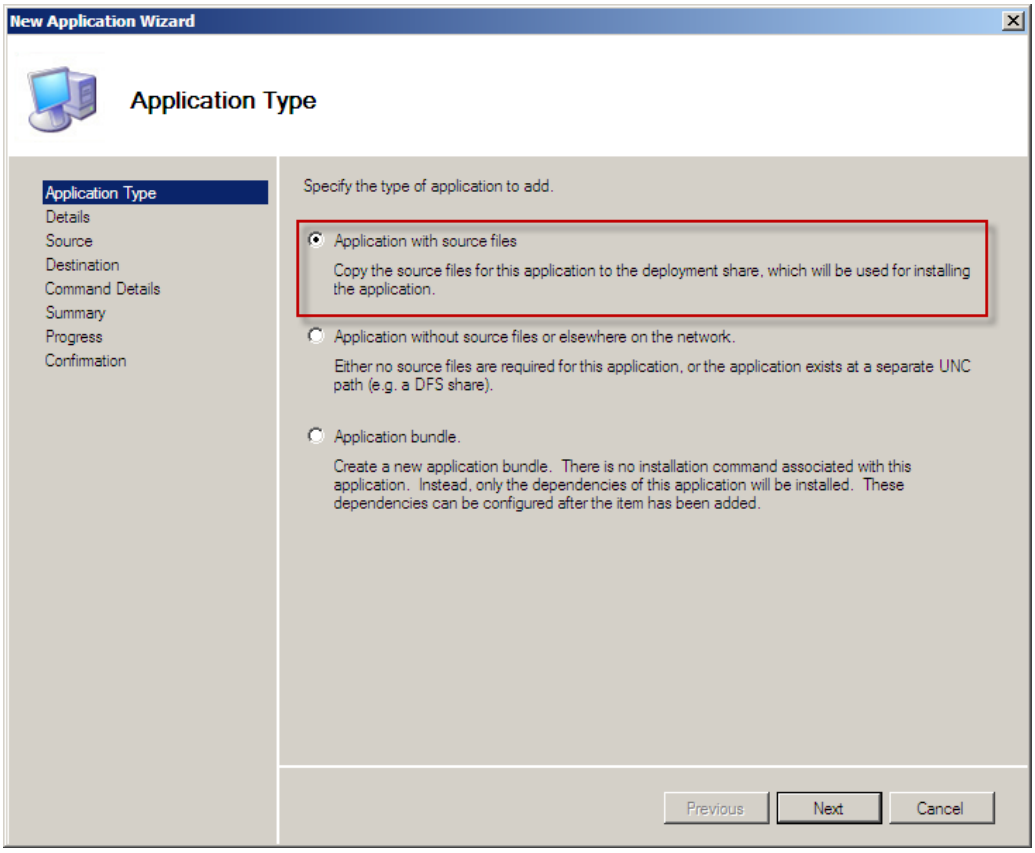


2. In the Deployment Workbench, right-click on the **Applications \ Healthcare MDT 2010 \ BUILD** folder and select **New Application**:



The **New Application Wizard** displays.

3. Select **Application with source files** and click **Next**:



Specify the type of application to add.

● Application with source files

Copy the source files for this application to the deployment share, which will be used for installing the application.

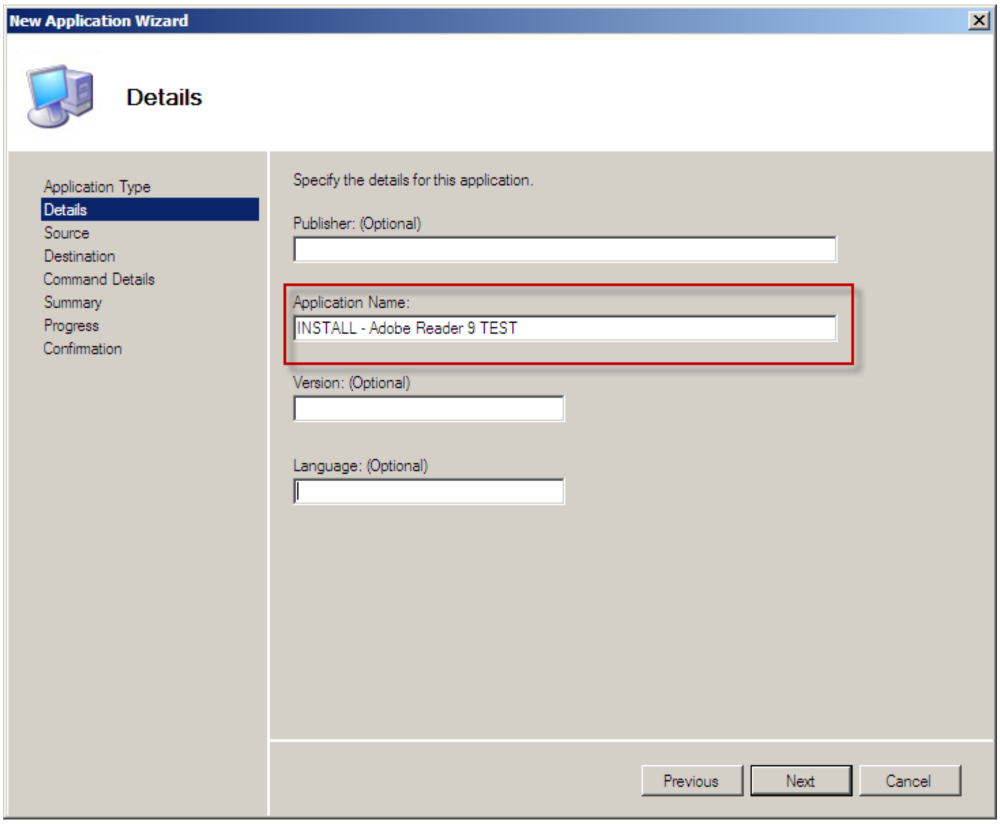
○ Application without source files or elsewhere on the network.

Either no source files are required for this application, or the application exists at a separate UNC path (e.g. a DFS share).

○ Application bundle.

Create a new application bundle. There is no installation command associated with this application. Instead, only the dependencies of this application will be installed. These dependencies can be configured after the item has been added.

4. Enter the details of the application as shown below. The convention in Healthcare Extensions for MDT 2010 is to use the ‘**INSTALL –** ‘ prefix. As there is already an existing Adobe Reader application supplied, the name should be ‘**INSTALL – Adobe Reader 9 TEST**’. Click **Next**:



Specify the details for this application.

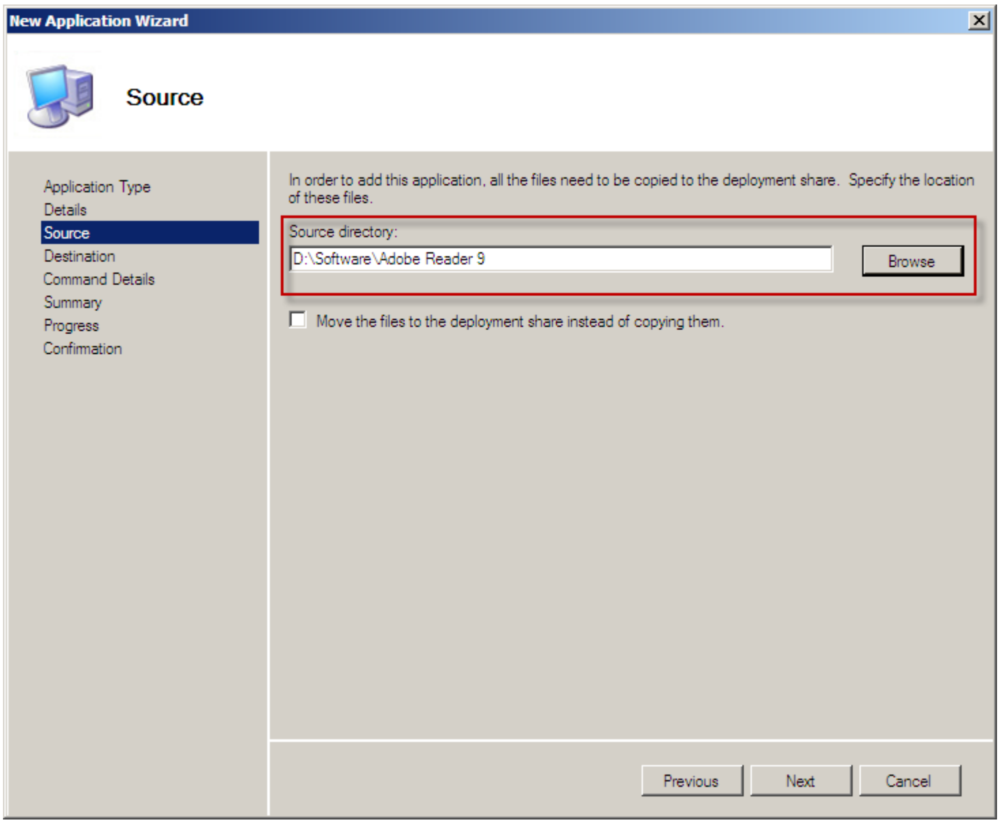
Publisher: (Optional)

Application Name: INSTALL - Adobe Reader 9 TEST

Version: (Optional)

Language: (Optional)

5. Enter the directory in which the application source files are located and click **Next**:

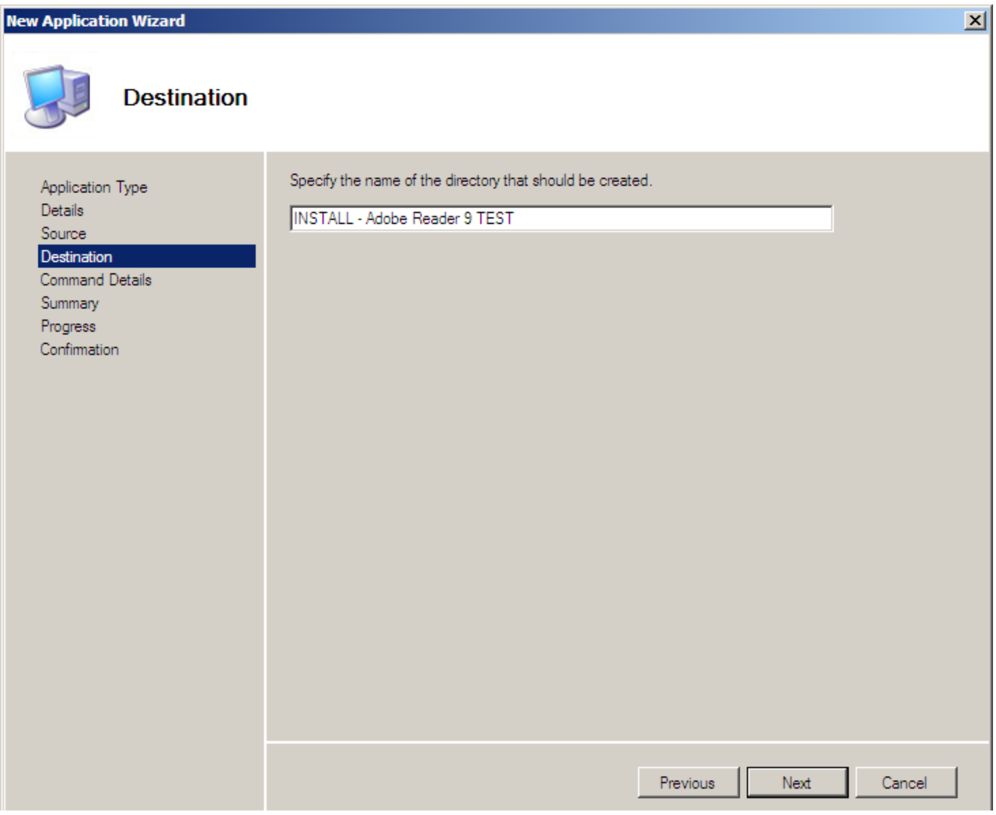


In order to add this application, all the files need to be copied to the deployment share. Specify the location of these files.

Source directory: D:\Software\Adobe Reader 9

☐ Move the files to the deployment share instead of copying them.

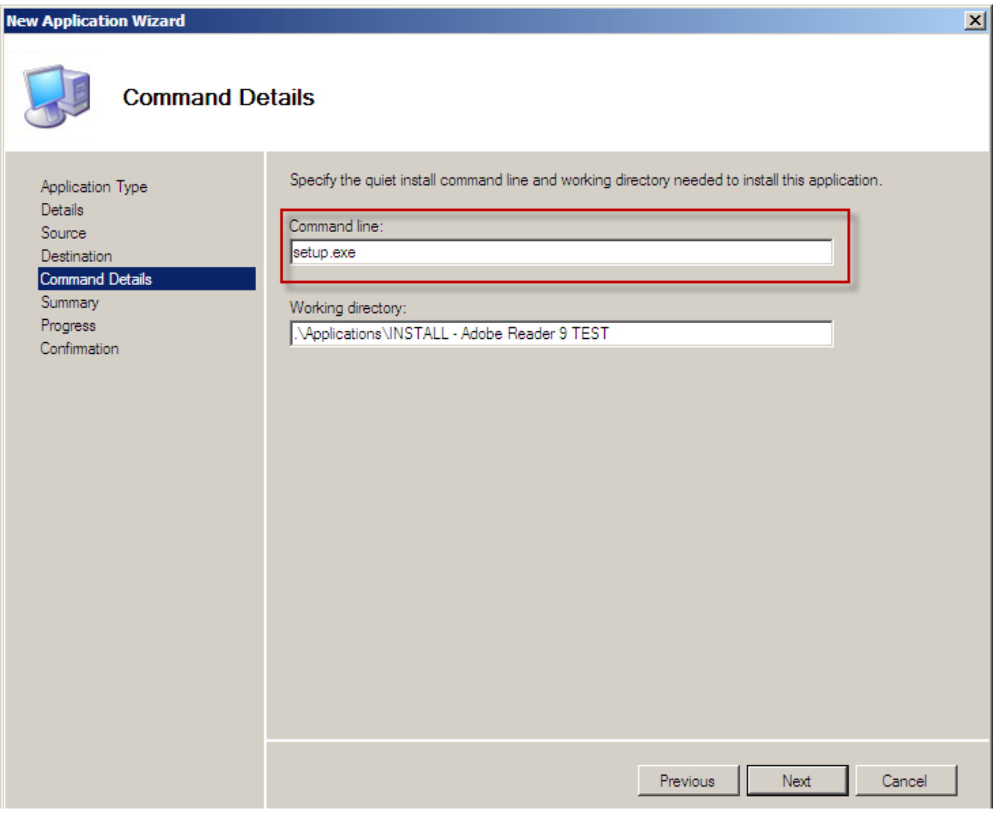
6. Enter the name of the directory that will be created for this application in the **<Distribution>\Applications** folder of MDT 2010 and click **Next**:



Specify the name of the directory that should be created.

INSTALL - Adobe Reader 9 TEST

7. In the **Command line**, enter the command that will be used to silently install the application. For Adobe Reader, this is ‘**setup.exe**’. The working directory is populated automatically. When finished, click **Next**:



Specify the quiet install command line and working directory needed to install this application.

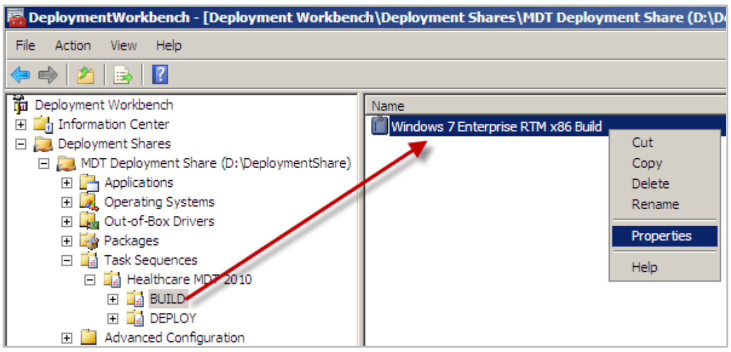
Command line: setup.exe

Working directory: .\Applications\INSTALL - Adobe Reader 9 TEST

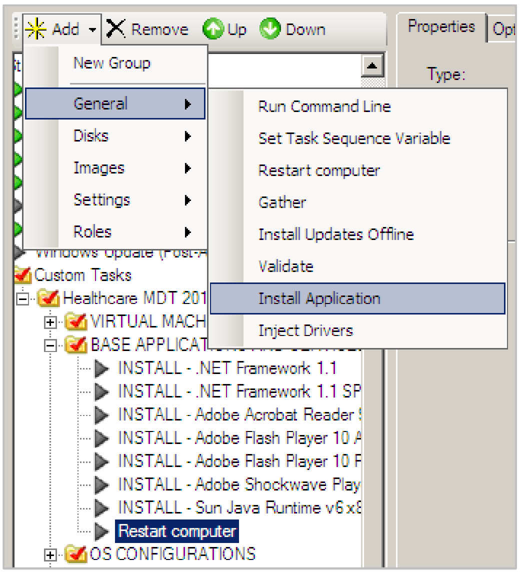
> **Note**
>
> The document *Automated Build Application Integration* **{R1}** includes information on the command line commands for other applications.

8. Accept the defaults on the remaining pages and click **Finish**. The application source files will be copied to the **<Distribution>\ Applications** folder and the new application will appear in the Deployment Workbench.

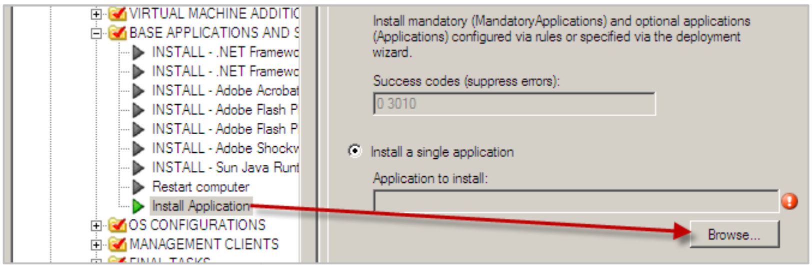
9. Select the Task Sequence to add the application to, right-click and select **Properties**:



10. On the Task Sequence tab, browse to the **BASE APPLICATIONS AND SERVICES** node and highlight the entry where the new application will be placed. Select **Add \ General \ Install Application**:



11. Select **Install a single application** and use the Browse button to select it:



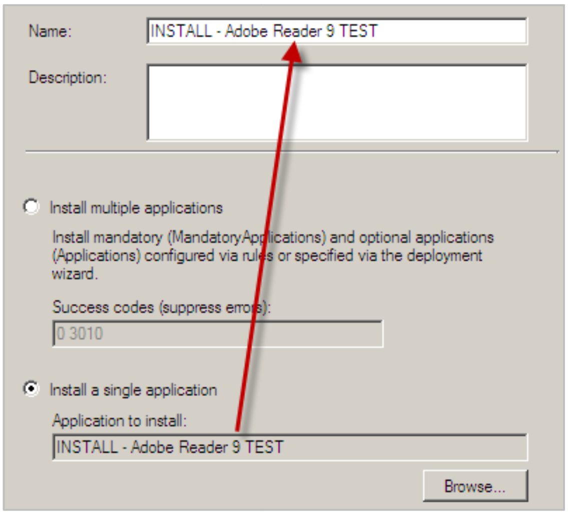
○ Install mandatory (MandatoryApplications) and optional applications (Applications) configured via rules or specified via the deployment wizard.

Success codes (suppress errors): 0 3010

● Install a single application

Application to install:

12. Change the **Name** of the application to match the full application name:



Name: INSTALL - Adobe Reader 9 TEST

Description:

○ Install multiple applications

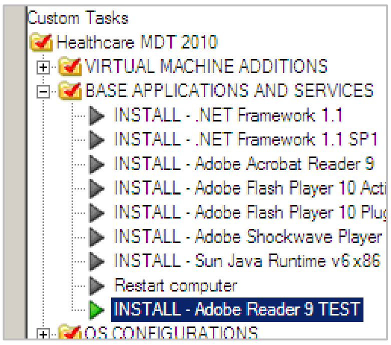
Install mandatory (MandatoryApplications) and optional applications (Applications) configured via rules or specified via the deployment wizard.

Success codes (suppress errors): 0 3010

● Install a single application

Application to install: INSTALL - Adobe Reader 9 TEST

13. Click **Apply**. and the new application will be shown in place:



```
Custom Tasks
  Healthcare MDT 2010
  ⊞ VIRTUAL MACHINE ADDITIONS
  ⊟ BASE APPLICATIONS AND SERVICES
      INSTALL - .NET Framework 1.1
      INSTALL - .NET Framework 1.1 SP1
      INSTALL - Adobe Acrobat Reader 9
      INSTALL - Adobe Flash Player 10 Acti
      INSTALL - Adobe Flash Player 10 Plug
      INSTALL - Adobe Shockwave Player
      INSTALL - Sun Java Runtime v6 x86
      Restart computer
      INSTALL - Adobe Reader 9 TEST
  ⊞ OS CONFIGURATIONS
```

14. Click **OK** to close the Task Sequence. The installation of the application is now complete.

4.6 Adding Hotfixes

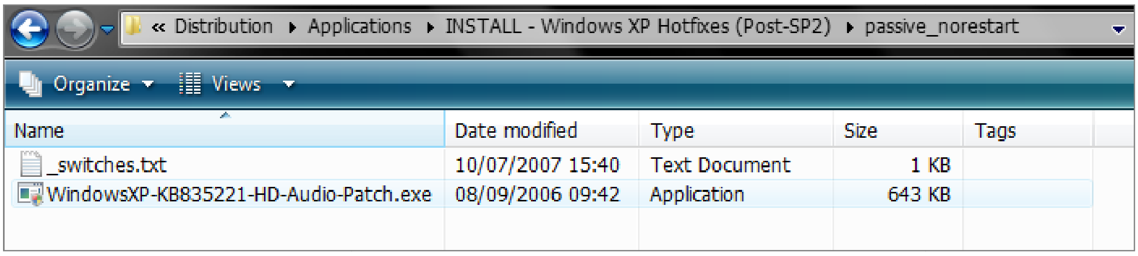
In Healthcare Extensions for MDT 2010, there are two ways of adding hotfixes to the build process. The method used depends on the build:

- For Windows XP or Windows Server 2003 use an application called 'INSTALL – Windows XP Hotfixes'
- For Windows Vista, Windows 7 or Windows Server 2008. – can be directly added into the Deployment Workbench

See sections 4.6.1 and 4.6.2 for more on the above methods.

4.6.1 Windows XP Hotfixes

When Healthcare Extensions for MDT 2010 is installed on a Windows XP task sequence, an application called 'INSTALL – Windows XP Hotfixes (Post-SP3)' is added by default. This application is responsible for installing Windows XP hotfixes placed in the **<Distribution> \ Applications \ INSTALL: Windows XP Hotfixes (Post-SP3) \ passive_norestart** folder, as shown in Figure 14:


« Distribution ▸ Applications ▸ INSTALL - Windows XP Hotfixes (Post-SP2) ▸ passive_norestart
Organize
Views

Name	Date modified	Type	Size	Tags
_switches.txt	10/07/2007 15:40	Text Document	1 KB	
WindowsXP-KB835221-HD-Audio-Patch.exe	08/09/2006 09:42	Application	643 KB	

Figure 14: Windows XP Hotfix Folder

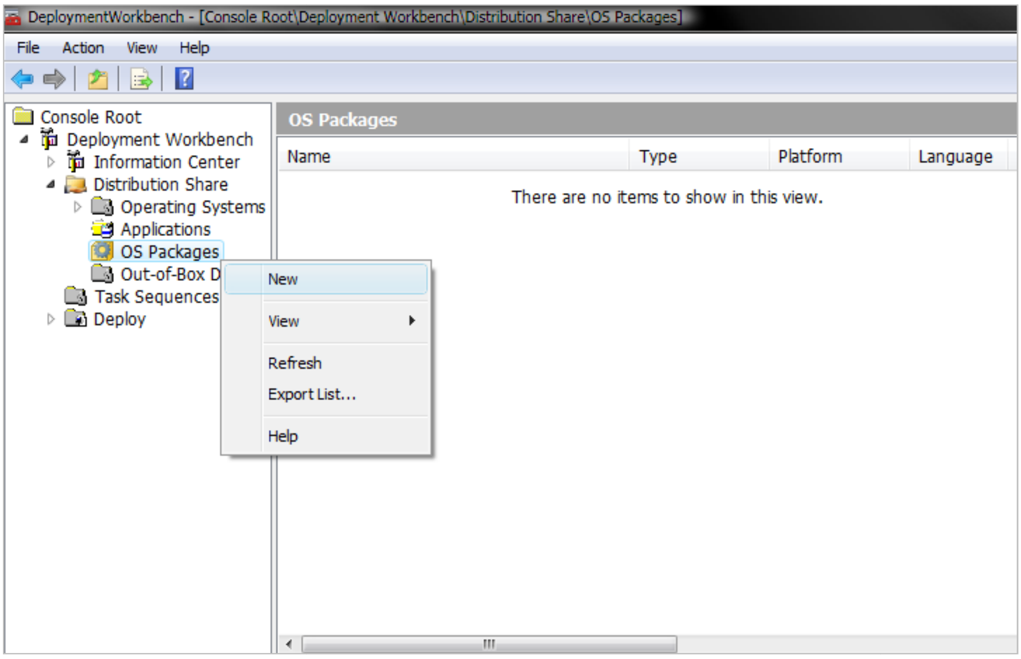
Any Windows XP hotfixes placed in this folder will be automatically installed *in alphabetical order* during the Windows XP build process.

4.6.2 Windows Vista and Later Hotfixes

Windows Vista and later hotfixes (in the MSU file format) can be added directly into the Deployment Workbench.

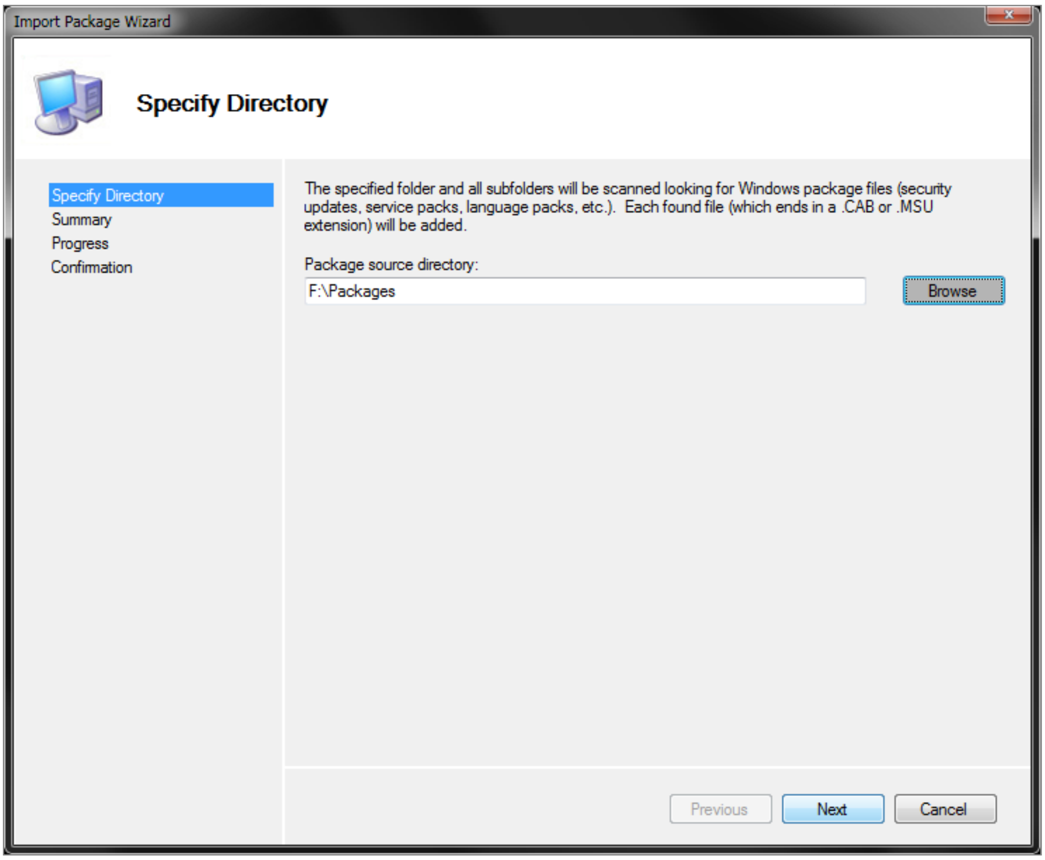
To add a hotfix to the Deployment Workbench:

1. In the Deployment Workbench, navigate to **Deployment Share \ Packages**.
2. Right-click **OS Packages** and select **New**:



There are no items to show in this view.

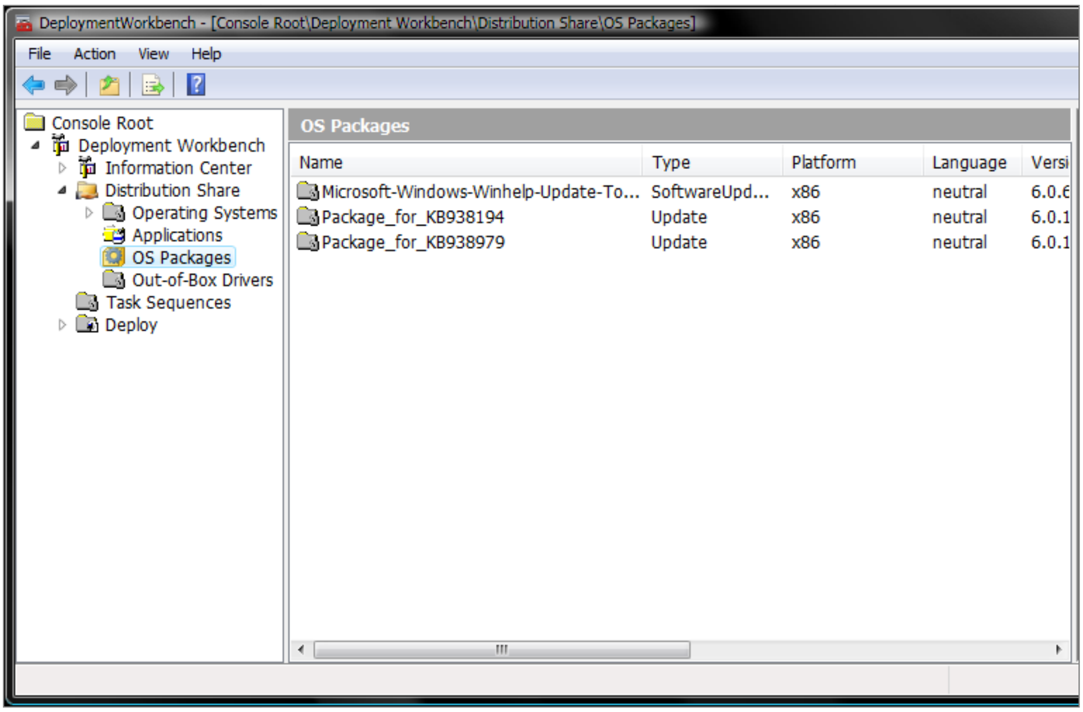
3. Enter the **Package source directory** in which one or more Windows Vista or Windows Server 2008 hotfixes are stored. In this example, the **F:\Packages** folder is used. Click **Next** followed by **Finish**:



The specified folder and all subfolders will be scanned looking for Windows package files (security updates, service packs, language packs, etc.). Each found file (which ends in a .CAB or .MSU extension) will be added.

Package source directory: F:\Packages

The hotfixes contained in the folder will be imported into the Deployment Workbench as shown below:



Name	Type	Platform	Language	Versi
Microsoft-Windows-Winhelp-Update-To...	SoftwareUpd...	x86	neutral	6.0.6
Package_for_KB938194	Update	x86	neutral	6.0.1
Package_for_KB938979	Update	x86	neutral	6.0.1

4.7 Customising Healthcare Extensions for MDT 2010

During an automated build (as well as during the installation of applications), an IT administrator may want to perform other operating system configurations. In Healthcare Extensions for MDT 2010, these configurations are performed using a set of scripts that are executed from a Task Sequence.

4.7.1 Healthcare Extensions for MDT 2010 Default Scripts

Healthcare Extensions for MDT 2010 comes supplied with a number of customisation scripts that are installed into the **<Distribution>\Applications\HTHMDT2010** folder. Where these scripts are installed varies depending on the operating system chosen. Brief descriptions of all the scripts are shown in Table 5:

Script Full Name	Script Location and File Name (Within <Distribution>\Applications\HTHMDT2010\)	Script Function
Generic Client Configuration Scripts (Windows XP and Windows Vista):		
CONFIG-CLI: Adjust Event Logs	GenericCLI\CFG-EventLog.vbs	Customises event logs to overwrite as needed
CONFIG-CLI: Disable CD-ROM Autorun	GenericCLI\CFG-NoAutoRun.vbs	Disables the CD autorun feature
CONFIG-CLI: Set Date Format	GenericCLI\CFG-SetDateFormat.exe	Configures DD-MMM-YYYY date format
CONFIG-CLI: Set Page File Size	GenericCLI\CFG-PageFileConfig.wsf	Configures the page file size
CONFIG-CLI: Set System Restore Percentage	GenericCLI\CFG-SystemRestorePercent.wsf	Configures the maximum percentage of disk space to use for system restore
Windows XP Configuration Scripts:		
CONFIG-XP: Activate Screen Saver	XP\CFG-SetScrSavr.VBS	Configures and enables the screen saver
CONFIG-XP: Apply Windows Firewall Settings	XP\CFG-firewall.cmd	Configures the Windows Firewall
CONFIG-XP: Close Start Menu	XP\CFG-CloseStartMenu.vbs	Ensures the Start menu is closed during the build
CONFIG-XP: Customise Start Menu Icons	XP\CFG-StartIcn.vbs	Configures Start menu icons
CONFIG-XP: Customise Folder View Settings	XP\CFG-SetFldr.vbs	Configures how folders appear
CONFIG-XP: Customise Recovery Options	XP\CFG-XPRecovr.vbs	Configures the Windows XP recovery options
CONFIG-XP: Customise Start Menu Options	XP\CFG-StartOpt.vbs	Configures aspects of the Start menu
CONFIG-XP: Disable Auto Update (Windows Update)	XP\CFG-XPupdate.vbs	Disables automatic updates
CONFIG-XP: Disable Internet Connection Wizard Icon	XP\CFG-RemovICW.vbs	Disables the Internet connection wizard
CONFIG-XP: Disable Unneeded Services	XP\CFG-ConfigServices.wsf	Configures which services to start
CONFIG-XP: Disable Welcome Screen	XP\CFG-nowelcm.vbs	Disables the Windows XP Welcome screen

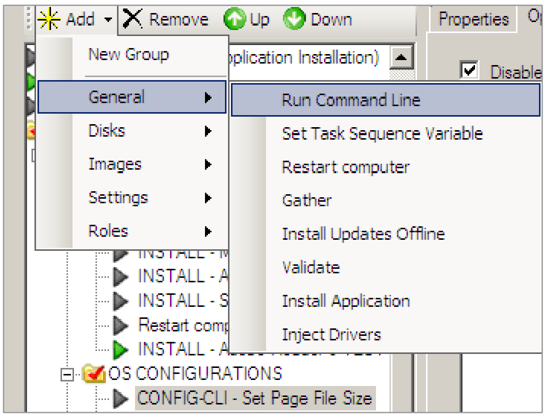
Script Full Name	Script Location and File Name (Within <Distribution>\Applications\HTHMDT2010\)	Script Function
CONFIG-XP: Disable Windows Tour Icon	XP\CFG-notour.vbs	Disables the Windows XP Tour Icon
CONFIG-XP: Enable the Quick Launch Bar	XP\CFG-showQL.vbs	Enables the quick launch bar
CONFIG-XP: Move Windows i386 Source Files	XP\CFG-updi386.vbs	Moves the location of the i386 folder
CONFIG-XP: Set boot.ini Timeout	XP\CFG-BootPro.vbs	Configures the boot.ini timeout **Note** This script should only be set to run after the SYSPREP process has completed.
Windows Vista Configuration Scripts:		
CONFIG-VISTA: Activate Screen Saver	Vista\CFG-SetScrSavr.VBS	Configures and enables the screen saver
CONFIG-VISTA: Configure Sidebar	Vista\CFG-SidebarState.wsf	Turns the Windows Vista sidebar on or off
CONFIG-VISTA: Customise Folder View Settings	Vista\CFG-SetFldr.wsf	Configures how folders appear
CONFIG-VISTA: Customise Recovery Options	Vista\CFG-Recovery.vbs	Configures the Windows Vista recovery options
CONFIG-VISTA: Disable simple file sharing	Vista\CFG-DisableSimpleFileSharing.wsf	Disables the file sharing wizard
CONFIG-VISTA: Disable Unneeded Services	Vista\CFG-ConfigServices.wsf	Configures which services to start
CONFIG-VISTA: Turn off Welcome Center	Vista\CFG-TurnOffWelcomeCenter.wsf	Disables the Welcome Center
CONFIG-VISTA: Turn on inactive tray icons	Vista\CFG-ShowInactiveTrayIcons.wsf	Ensures tray icons are always visible
Generic Server Configuration Scripts (Windows Server 2008):		
CONFIG-SVR: Autologon	GenericSVR\CONFIG-SVR-Autologon.wsf	Used to populate settings to allow reboots during a custom task sequence
CONFIG-SVR: Create OUs	GenericSVR\CONFIG-SVR-CreateOUs.wsf	Used to create an Organizational Unit structure
CONFIG-SVR: DHCP Server	GenericSVR\CONFIG-SVR-DHCP.wsf	Used to configure DHCP (DHCP must be installed before this script is executed)
CONFIG-SVR: DNS Server	GenericSVR\CONFIG-SVR-DNS.wsf	Used to configure DNS
CONFIG-SVR: Enable Remote Desktop	GenericSVR\CONFIG-SVR-EnableRemoteDesktop.wsf	Used to enable remote desktop

Script Full Name	Script Location and File Name (Within <Distribution>\Applications\HTHMDT2010\)	Script Function
CONFIG-SVR: Rename Computer	GenericSVR\CONFIG-SVR-RenameComputer.wsf	Used to rename the computer **Note** This should only be used on a computer that is NOT a Domain Controller (DC).
CONFIG-SVR: Set Administrator Password	GenericSVR\CONFIG-SVR-ResetAdminPass.wsf	Used to set the administrator password
CONFIG-SVR: WDS Server	GenericSVR\CONFIG-SVR-WDS.wsf	Used to configure the WDS server. WDS must be installed before this script is executed

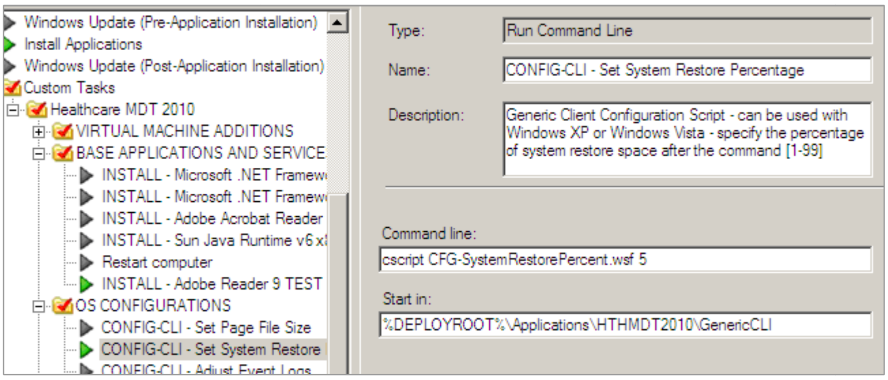
Table 5: Configuration Scripts

Configuration scripts not already present in a Task Sequence can be added as follows:

1. Open the Task Sequence **Properties**.
2. In the **OS CONFIGURATIONS** node, highlight the location in which the script should be added.
3. Select **Add \ General \ Run Command Line**:



4. Complete the **Command line** and **Start in** boxes. This will be different depending on the script, but will be similar to:
   - **Command line:** cscript.exe <scriptname> <parameters>
   - **Start in:** %DEPLOYROOT%\Applications\HTHMDT2010\<script folder>


Windows Update (Pre-Application Installation)
Install Applications
Windows Update (Post-Application Installation)
Custom Tasks
Healthcare MDT 2010
VIRTUAL MACHINE ADDITIONS
BASE APPLICATIONS AND SERVICE
INSTALL - Microsoft .NET Framew
INSTALL - Microsoft .NET Framew
INSTALL - Adobe Acrobat Reader
INSTALL - Sun Java Runtime v6 x
Restart computer
INSTALL - Adobe Reader 9 TEST
OS CONFIGURATIONS
CONFIG-CLI - Set Page File Size
CONFIG-CLI - Set System Restore
CONFIG-CLI - Adjust Event Logs

Type: Run Command Line

Name: CONFIG-CLI - Set System Restore Percentage

Description: Generic Client Configuration Script - can be used with Windows XP or Windows Vista - specify the percentage of system restore space after the command [1-99]

Command line: cscript CFG-SystemRestorePercent.wsf 5

Start in: %DEPLOYROOT%\Applications\HTHMDT2010\GenericCLI

5. Click **OK** to save the Task Sequence.

4.7.2 Adding Custom Scripts

To make writing custom scripts easier in this version of Healthcare Extensions for MDT 2010, a template script file and folder have been created. It is recommended that custom scripts are placed in the **<Distribution> \ Applications \ HTHMDT2010 \ Custom** folder. By default, this folder contains a single template VBScript named _Template.wsf.

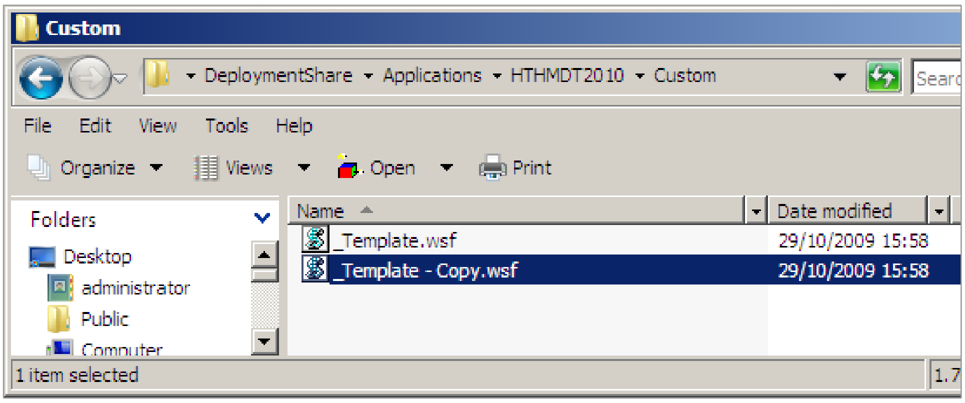
Use the _Template.wsf file by copying it and editing it in Notepad. The script has been preconfigured to make the EXTUtility scripting library available, which provides a number of simple functions to make scripting easier. Examples of EXTUtility use are provided within the _Template.wsf file.

Once a script has been completed, it can be added to a Task Sequence as a new 'CONFIG:' entry. To add the script, use the **Add \ General \ Run Command Line** menu from within the Task Sequence, and configure it with the following details:

- **Command line:** Enter 'cscript.exe <ScriptName>.wsf'
- **Start in:** Enter '%DEPLOYROOT% \ Applications \ HTHMDT2010 \ Custom'

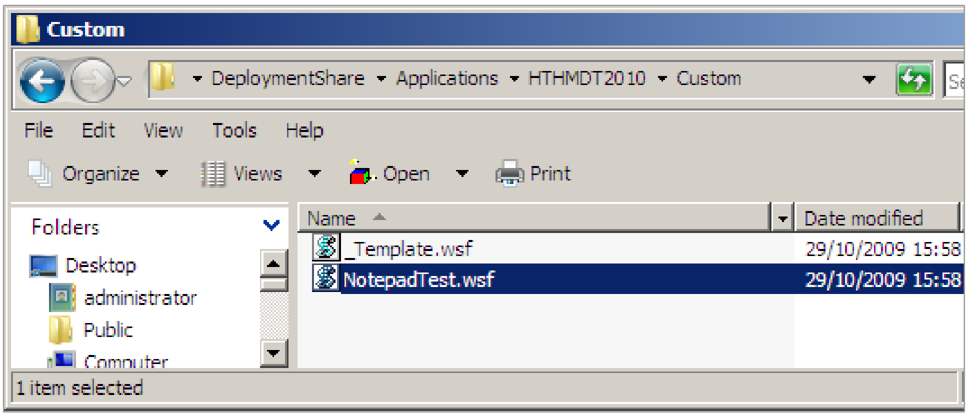
To add a custom script:

1. Make a copy of the **_Template.wsf** file located in the **<Distribution> \ Applications \ HTHMDT2010 \ Custom** folder:


Custom
DeploymentShare ▸ Applications ▸ HTHMDT2010 ▸ Custom
Searc
File Edit View Tools Help
Organize Views Open Print
Folders
Desktop
administrator
Public
Computer
1 item selected
1.7

Name	Date modified
_Template.wsf	29/10/2009 15:58
_Template - Copy.wsf	29/10/2009 15:58

2. Rename the script as appropriate; in this example **NotepadTest.wsf** will be used.

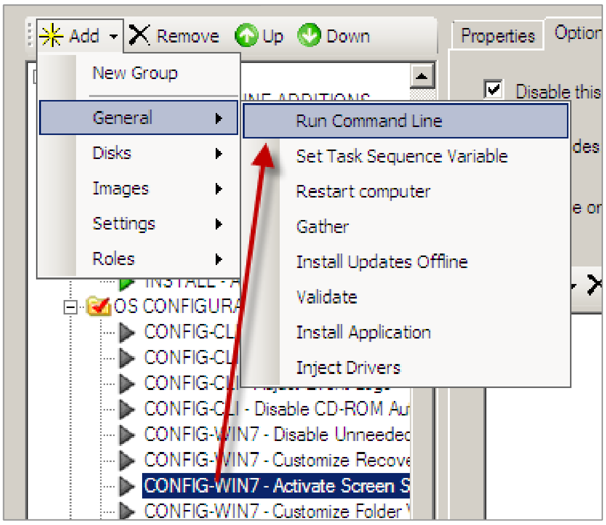

Custom
DeploymentShare ▸ Applications ▸ HTHMDT2010 ▸ Custom
Se
File Edit View Tools Help
Organize Views Open Print
Folders
Desktop
administrator
Public
Computer
1 item selected

Name	Date modified
_Template.wsf	29/10/2009 15:58
NotepadTest.wsf	29/10/2009 15:58

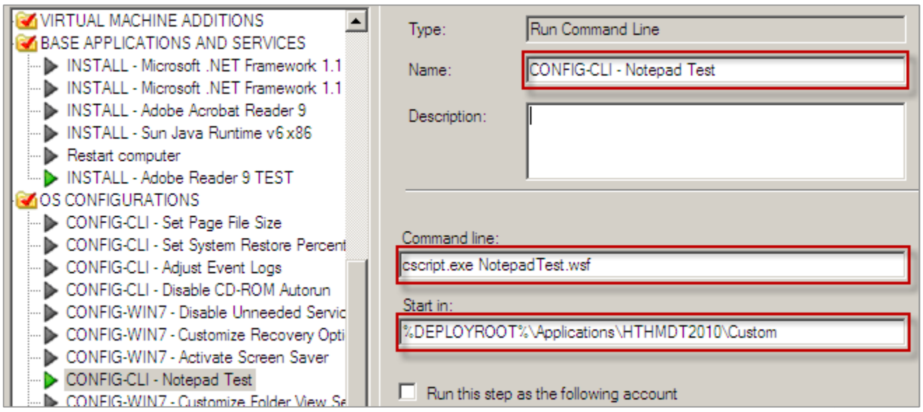
3. Edit the script file. An example for **NotepadTest.wsf** is:

```
<job id="Template">
     <script language="VBScript" src="..\..\..\Scripts\ZTIUtility.vbs"/>
     <script language="VBScript" src="..\Tools\EXTUtility.vbs"/>
     <script language="VBScript">

'//--------------------------------------------------------------------------
'//
'//  Global constant and variable declarations
'//
'//--------------------------------------------------------------------------

Option Explicit

'//--------------------------------------------------------------------------
'//  End declarations
'//--------------------------------------------------------------------------


'//--------------------------------------------------------------------------
'//  Main routine
'//--------------------------------------------------------------------------

' Run notepad and wait for it to exit before continuing
oEXTUtil.RunWait "notepad.exe"


     </script>
</job>
```

4. In the Task Sequence **Properties**, highlight the location to insert the script and select **Add \ General \ Run Command Line**:



5. Complete the following boxes:
   - **Name:** CONFIG-CLI – Notepad Test
   - **Command line:** cscript.exe NotepadTest.wsf
   - **Start in:** %DEPLOYROOT%\Applications\HTHMDT2010\Custom



VIRTUAL MACHINE ADDITIONS
BASE APPLICATIONS AND SERVICES
INSTALL - Microsoft .NET Framework 1.1
INSTALL - Microsoft .NET Framework 1.1
INSTALL - Adobe Acrobat Reader 9
INSTALL - Sun Java Runtime v6 x86
Restart computer
INSTALL - Adobe Reader 9 TEST
OS CONFIGURATIONS
CONFIG-CLI - Set Page File Size
CONFIG-CLI - Set System Restore Percent
CONFIG-CLI - Adjust Event Logs
CONFIG-CLI - Disable CD-ROM Autorun
CONFIG-WIN7 - Disable Unneeded Servic
CONFIG-WIN7 - Customize Recovery Opti
CONFIG-WIN7 - Activate Screen Saver
CONFIG-CLI - Notepad Test
CONFIG-WIN7 - Customize Folder View Se

Type: Run Command Line

Name: CONFIG-CLI - Notepad Test

Description:

Command line: cscript.exe NotepadTest.wsf

Start in: %DEPLOYROOT%\Applications\HTHMDT2010\Custom

☐ Run this step as the following account

6. Click **OK** to save the Task Sequence.

4.7.3 Incorporating BitLocker into a Task Sequence

MDT 2010 introduced an additional feature available during a Windows Vista or later deployment: the encryption of hard disks using BitLocker.

BitLocker can be enabled in the Task Sequence properties by selecting **Add > Disks > Enable BitLocker** from within the **Task Sequence** tab. The options allow for either the operating system drive, or another designated drive, to be encrypted. It also provides the ability to create a recovery key in Active Directory.

> **Important**
>
> The ability to encrypt a drive other than the operating system drive is only available on a Windows Vista client with Service Pack 1 installed, Windows 7, or a Windows Server 2008 server.

If a machine has more than one drive that requires BitLocker, for example both the operating system drive (the C: drive) and the data drive (the D: drive), two tasks should be added to the task sequence. This is because each task can only enable BitLocker for one drive.

To ensure BitLocker can be run successfully, the BitLocker Drive Preparation Tool[3] needs to be available to MDT 2010. This tool consists of a number of files which should be copied to the distribution share directory structure. The files required, and the location in which they should reside, are as follows:

- <Distribution>\Tools\X86
  - Bdehdcfg.exe
  - BdeCfgLog.dll
- <Distribution>\Tools\X86\en-us
  - DedHdCfg.exe.mui
  - BdeCfgLog.dll.mui

[3] Microsoft Help and Support – Description of the BitLocker Drive Preparation Tool **{R2 }**: http://support.microsoft.com/kb/930063

Recommendation

It is recommended that BitLocker is not used in conjunction with any other hard drive encryption product.

Note

To be able to back up the Trusted Platform Module (TPM) owner information, and BitLocker recover information, to Active Directory Domain Services, appropriate schema extensions and access control settings must be configured on the domain. For more information, read the Microsoft TechNet article *Configuring Active Directory to Back up Windows BitLocker Drive Encryption and Trusted Platform Module Recovery Information*[4].

[4] Microsoft TechNet – BitLocker Drive Encryption Configuration Guide: Backing Up BitLocker and TPM Recovery Information to Active Directory **{R3}:** http://technet2.microsoft.com/WindowsVista/en/library/3dbad515-5a32-4330-ad6f-d1fb6dfcdd411033.mspx?mfr=true

5 STABILISE

The Stabilise phase involves testing the solution components whose features are complete, and resolving and prioritising any issues that are found. Testing during this phase emphasises usage and operation of the solution components under realistic environmental conditions.

Figure 15 acts as a high-level checklist, illustrating the areas of Healthcare Extensions for MDT 2010 that a healthcare IT Professional is responsible for stabilising:

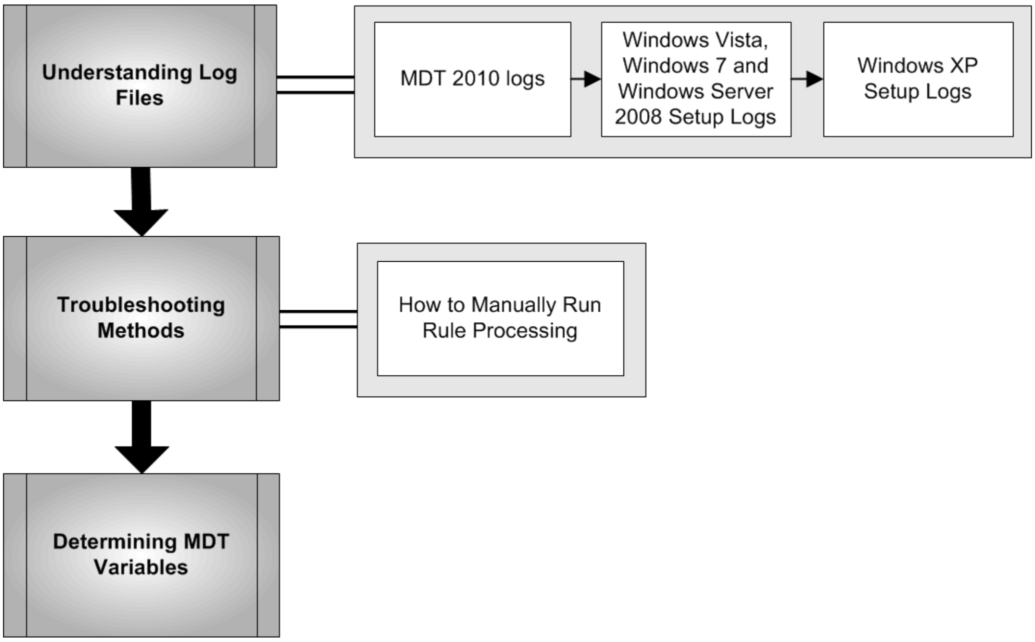

Understanding Log Files
MDT 2010 logs
Windows Vista, Windows 7 and Windows Server 2008 Setup Logs
Windows XP Setup Logs
Troubleshooting Methods
How to Manually Run Rule Processing
Determining MDT Variables

Figure 15: Sequence for Stabilising Healthcare Extensions for MDT 2010

5.1 Understanding Log Files

Before troubleshooting MDT 2010 and Healthcare Extensions for MDT 2010, there is a requirement to understand the many log files used during an operating system build and/or deployment. By understanding which log to refer to, and at what time, the build and deployment process can be appreciated from a troubleshooting perspective.

MDT 2010 scripts, as well as the Healthcare Extensions for MDT 2010 scripts EXTUtil.vbs and Install.vbs (found in the Windows XP Hotfix folder), automatically create log files. Each script creates a log file that matches the name of the script. For example, ZTIGather.wsf creates a log file named ZTIGather.log. Each script also updates a common log file called Bdd.log that aggregates the contents of logs created by all the MDT 2010 scripts. MDT 2010 logs are located in the C:\MININT\SMSOSD\OSDLOGS folder during the build/deployment process. They are moved at the completion of a deployment, and their destination is dependent on the deployment type, as described in Table 6:

Deployment Type	Destination
Lite Touch	C:\WINDOWS\TEMP\BDDLogs – unless the /debug:true option has been specified
Zero Touch	C:\TEMP\SMSOSD *or* C:\SMSOSD – unless the C:\MININT\Archive_OSD.SMS file is found

Table 6: Log File Destinations

The rest of this section details the log files that are created during the build and deployment processes, and provides examples of when they can be used for troubleshooting.

5.1.1 MDT 2010 Logs

MDT 2010 creates a number of log files during a build and deployment, as shown in Table 7:

Log File	Definition
Bdd.log	The aggregated MDT 2010 log file consists of entries from all MDT 2010 and Healthcare Extensions for MDT 2010 scripts. The Bdd.log file can be copied to a network location at the end of the deployment if the SLShare value is specified in CustomSettings.ini. The Bdd.log format is designed to be read by TRACE32, which is part of the SCCM 2007 Toolkit. It is recommended that this tool is used whenever possible to read the logs, as it makes finding errors easier. The SCCM 2007 Toolkit can be downloaded from: http://www.microsoft.com/downloads/details.aspx?FamilyID=948e477e-fd3b-4a09-9015-141683c7ad5f&DisplayLang=en
<Scriptname>.log	A log file created by each MDT 2010 script.
Wizard.log	Updated by MDT 2010 wizards.
DeployUpdates_platform.log	Created when deployment points are updated. Also used when updating Windows PE. This log is useful when troubleshooting Windows PE driver integration issues. This log is located in the %temp% folder.
SMSTS.log	Logs all of the transactions for the Task Sequence. This will be located in %temp% C:\Windows\System32\ccm\logs, or C:\SMSTSLog, depending on the situation.
WPEinit.log	Logs the Windows PE initialisation process. Useful for troubleshooting errors that arise when starting Windows PE.

Table 7: MDT 2010 Logs

5.1.2 Windows Vista, Windows 7 and Windows Server 2008 Setup Logs

Table 8 shows a subset of the setup logs that is most useful when troubleshooting deployment issues. For more detailed information about Windows Vista setup log files, see *Windows Vista setup log file locations*[5].

Log File	Definition
Setupapi.dev.log	Windows setup log, located in C:\Windows\inf. Useful for investigating failed driver installations.
Setupact.log	Windows setup log, located in C:\Windows\panther. Useful for investigating failed installations.
Setuperr.log	Windows setup log, located in C:\Windows\panther. Contains a list of errors that occurred during installation.
Netsetup.log	Windows setup log, located in C:\Windows\Debug. Useful for troubleshooting domain join issues.

Table 8: Windows Vista and Windows Server 2008 Setup Logs

5.1.3 Windows XP Setup Logs

Table 9 shows a subset of the setup logs that is most useful when troubleshooting deployment issues:

Log File	Definition
Setupapi.log	Windows setup log, located in C:\Windows. Records INF file installation actions. Useful for investigating failed driver installations.

[5] Microsoft Help and Support – Windows 7, Windows Server 2008 R2, and Windows Vista setup log file locations **{R4 }**: http://support.microsoft.com/kb/927521

Log File	Definition
Setupact.log	Windows setup log, located in C:\Windows. Lists installation actions.
Setuperr.log	Windows setup log, located in C:\Windows. Details installation errors.
Netsetup.log	Windows setup log, located in C:\WSndows\Debug. Useful for troubleshooting issues related to joining a domain.

Table 9: Windows XP and Windows Server 2003 Setup Logs

5.2 Troubleshooting Methods

During the troubleshooting process, it is critical that any changes made to the rules can be easily tested. Testing rules should follow a two-stage process:

1. Test on a client in the full operating system (OS) environment.
2. Launch the process and test within WinPE.

5.2.1 How to Manually Run Rule Processing

When creating rules within MDT 2010, it can take time to tune them to provide the required results. Each time a change is made, the entire build process needs to be restarted to see if the rules have the desired effect.

It is, however, possible to initially test MDT 2010 rule processing manually on a client in the full OS, without needing to completely rebuild a machine.

The rules that are applied to a particular computer are gathered together using a script called ZTIGather.wsf in conjunction with ZTIGather.xml. These scripts can be run directly without the need to launch the entire MDT 2010 process.

The steps below detail the process of configuring and troubleshooting rule processing within the Healthcare Extensions for MDT 2010. These can be run when the computer is running either the host operating system or WinPE.

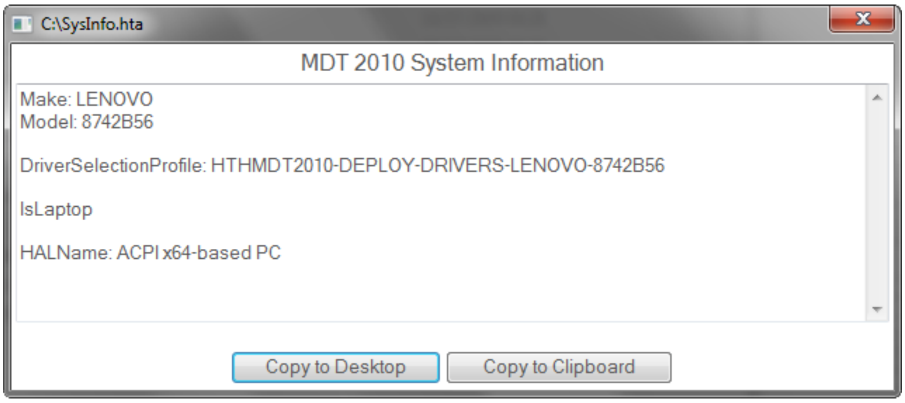
To manually run rule processing:

1. Copy **ZTIGather.wsf**, **ZTIGather.xml**, **ZTIUtility.vbs**, **CustomSettings.ini**, and any user exit scripts to a folder on the built workstation.
2. Remove the **C:\MININT** folder if it already exists.
3. Execute **cscript.exe ZTIGather.wsf /debug:true**.
4. Inspect the log **C:\MININT\SMSOSD\OSDLOGS\ZTIGather.log**.

5.3 Determining MDT Variables

During the MDT 2010 deployment process, information about the target machine is gathered using Windows Management Instrumentation (WMI), and made available using MDT variables. The information needs to be known so that automation functions can be utilised and logic can be incorporated into the Task Sequence to create rules. The most commonly used variables can be shown by using the **SysInfo** application, which is supplied with Healthcare Extensions for MDT 2010:

1. Copy **sysinfo.hta** to the machine that you want to identify. These files are installed by Healthcare Extensions for MDT 2010 in **<Distribution> \ Applications \ HTHMDT2010 \ Tools**.
2. Run **sysinfo.hta** on the machine:



```
Make: LENOVO
Model: 8742B56

DriverSelectionProfile: HTHMDT2010-DEPLOY-DRIVERS-LENOVO-8742B56

IsLaptop

HALName: ACPI x64-based PC
```

sysinfo.hta will display the following variables:

- Machine make
- Machine model
- Healthcare Extensions for MDT 2010 Driver Selection Profile (for more information on this, see section 6.2)
- Chassis type (Either IsDesktop, IsLaptop or IsServer)
- Hardware Abstraction Layer (HAL) name

3. Use the Copy buttons to copy the output of **sysinfo.hta** to a local file, or to the clipboard, for later use when configuring Task Sequence properties.

6 Deploy

During the Deploy phase, the core solution components are deployed for more widespread application and use, and the deployment is stabilised through ongoing monitoring. The solution is then transitioned to operations and support.

Figure 16 acts as a high-level checklist, illustrating the critical tasks that a healthcare IT Professional responsible for deploying Healthcare Extensions for MDT 2010 needs to perform

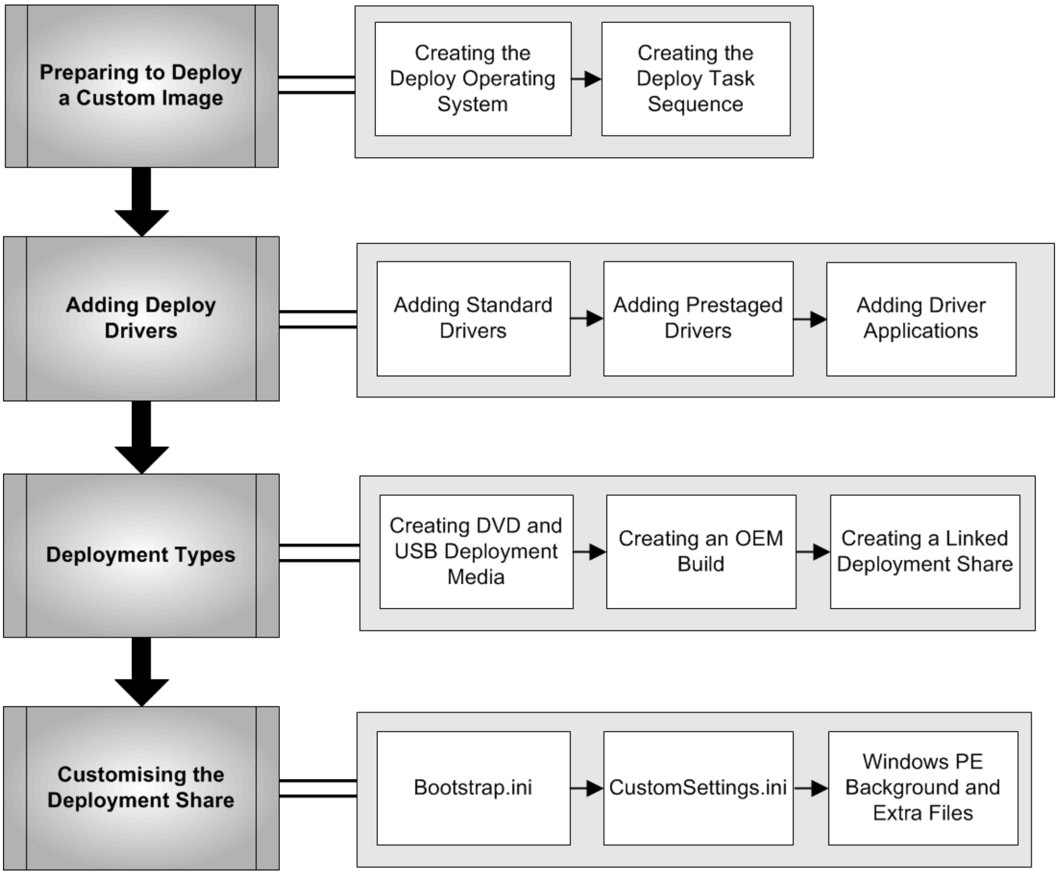

Preparing to Deploy a Custom Image
Creating the Deploy Operating System
Creating the Deploy Task Sequence
Adding Deploy Drivers
Adding Standard Drivers
Adding Prestaged Drivers
Adding Driver Applications
Deployment Types
Creating DVD and USB Deployment Media
Creating an OEM Build
Creating a Linked Deployment Share
Customising the Deployment Share
Bootstrap.ini
CustomSettings.ini
Windows PE Background and Extra Files

Figure 16: Sequence for Deploying Healthcare Extensions for MDT 2010

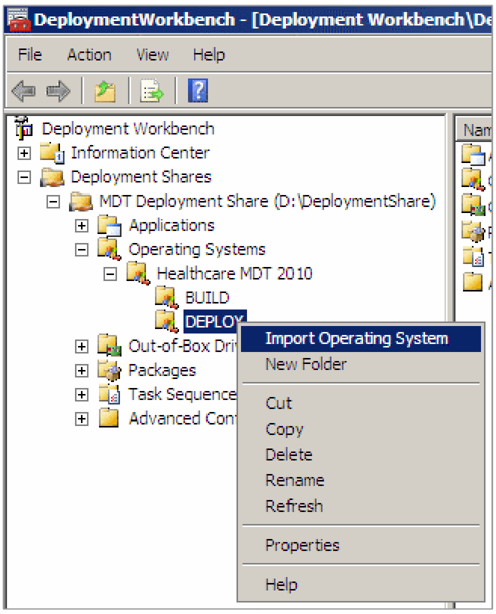
6.1 Preparing to Deploy a Custom Image

Once an image has been captured, it must be added as a new operating system and build.

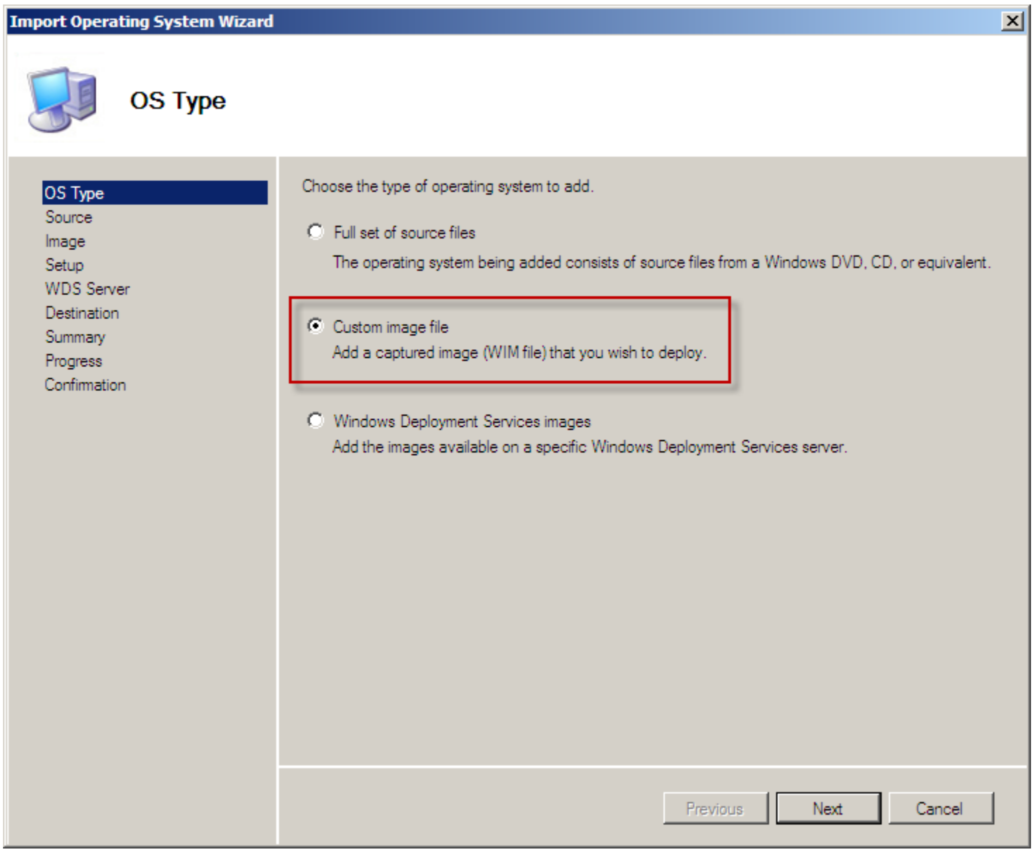
> **Note**
>
> A captured WIM image can be thought of as an operating system in itself. It can be used as the basis of another build process, and then captured again. In this way, it is possible to create a fairly simple build containing some basic healthcare organisation customisations, and to capture this build. This build can then be used as the basis for more specific builds, for example specific departmental builds.

6.1.1 Creating the Deploy Operating System

1. In the Deployment Workbench, right-click **Operating Systems \ Healthcare MDT 2010 \ DEPLOY** and select **Import Operating System**:



2. On the **OS Type** page, click **Custom image file** and click **Next**:



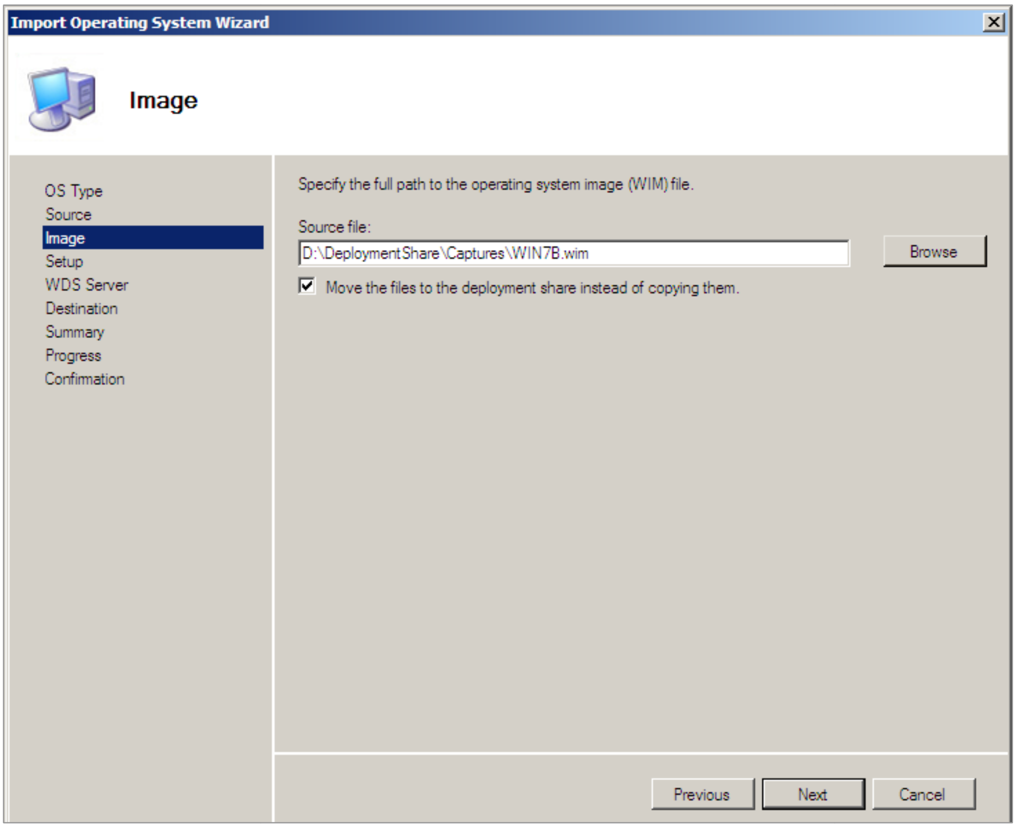
Choose the type of operating system to add.

○ Full set of source files
The operating system being added consists of source files from a Windows DVD, CD, or equivalent.

● Custom image file
Add a captured image (WIM file) that you wish to deploy.

○ Windows Deployment Services images
Add the images available on a specific Windows Deployment Services server.

3. On the **Image** page, select the image as captured in section 4.2.2. By default, this image will be located in **<Distribution>\Captures**:

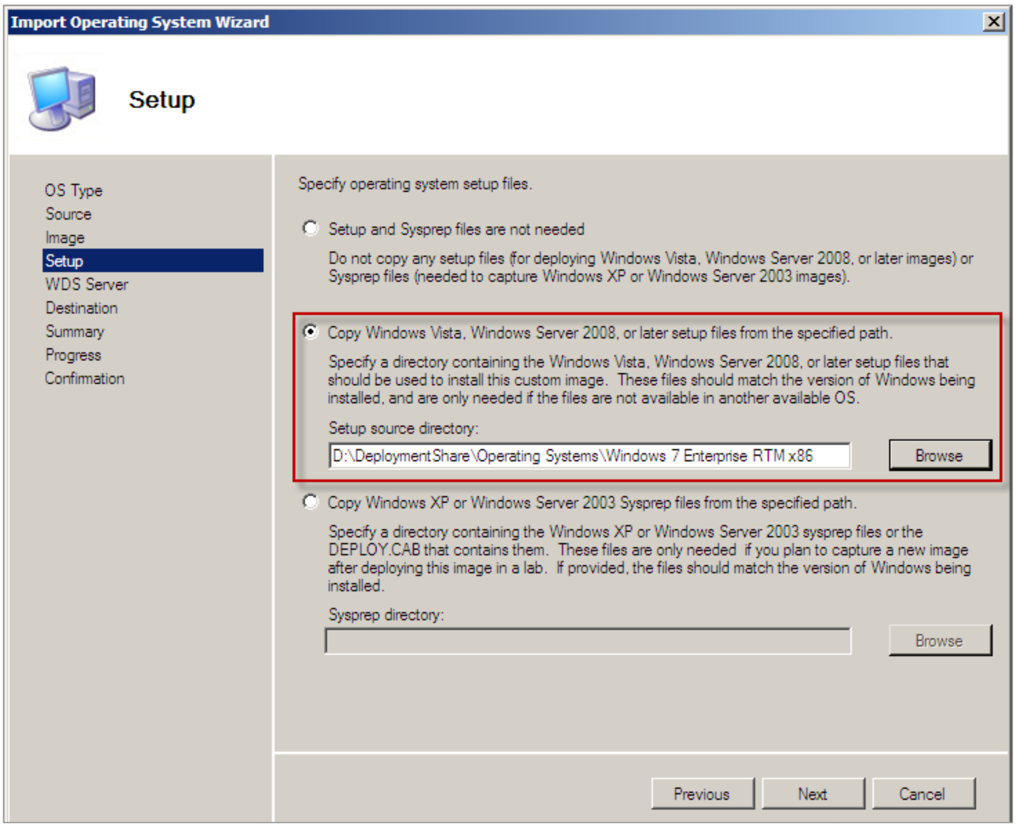


Specify the full path to the operating system image (WIM) file.

Source file: D:\DeploymentShare\Captures\WIN7B.wim

☑ Move the files to the deployment share instead of copying them.

4. On the **Setup** page, click **Copy Windows Vista…** (or **Copy Windows XP…** if using Windows XP) and browse to the setup source folder then click **Next**:



Specify operating system setup files.

○ Setup and Sysprep files are not needed

Do not copy any setup files (for deploying Windows Vista, Windows Server 2008, or later images) or Sysprep files (needed to capture Windows XP or Windows Server 2003 images).

● Copy Windows Vista, Windows Server 2008, or later setup files from the specified path.

Specify a directory containing the Windows Vista, Windows Server 2008, or later setup files that should be used to install this custom image. These files should match the version of Windows being installed, and are only needed if the files are not available in another available OS.

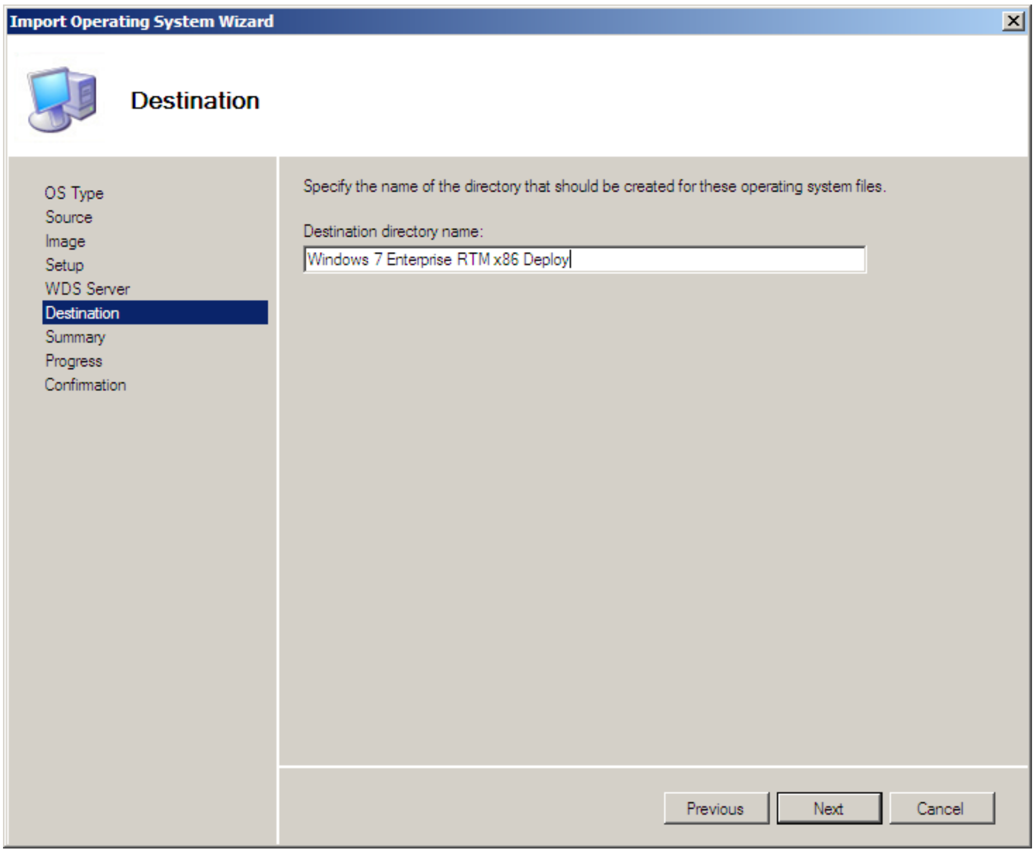
Setup source directory: D:\DeploymentShare\Operating Systems\Windows 7 Enterprise RTM x86

○ Copy Windows XP or Windows Server 2003 Sysprep files from the specified path.

Specify a directory containing the Windows XP or Windows Server 2003 sysprep files or the DEPLOY.CAB that contains them. These files are only needed if you plan to capture a new image after deploying this image in a lab. If provided, the files should match the version of Windows being installed.

Sysprep directory:

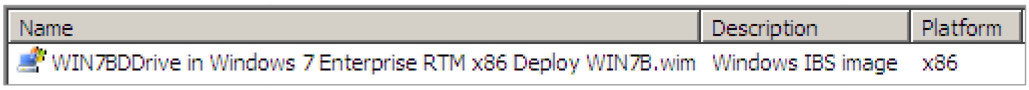
5. On the **Destination** page, enter the destination directory name that will be created in the MDT 2010 distribution share to store the operating system files. It is recommended that the name is in the convention **<Operating System><Service Pack><Architecture>**. For example 'Windows Vista Enterprise SP2 x86' or 'Windows 7 Enterprise RTM x86'. Click **Next**:



Specify the name of the directory that should be created for these operating system files.

Destination directory name: Windows 7 Enterprise RTM x86 Deploy

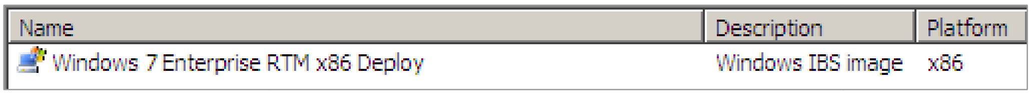
6. Accept the defaults for the remaining pages and click **Finish**. The operating system files will be copied to the MDT 2010 distribution share and will appear in the Deployment Workbench:



Name	Description	Platform
WIN7BDDrive in Windows 7 Enterprise RTM x86 Deploy WIN7B.wim	Windows IBS image	x86

Note

As the default names can be unclear, it is recommended that each entry is renamed to use the same convention used for the source path in previous steps. The new names are shown in Figure 17:



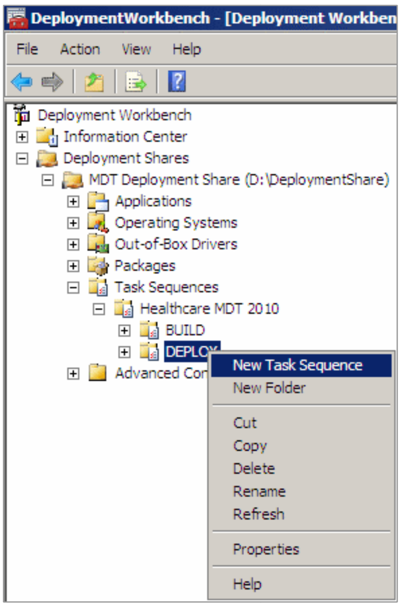
Name	Description	Platform
Windows 7 Enterprise RTM x86 Deploy	Windows IBS image	x86

Figure 17: Deployment Workbench with Renamed Operating Systems

6.1.2 Creating the Deploy Task Sequence

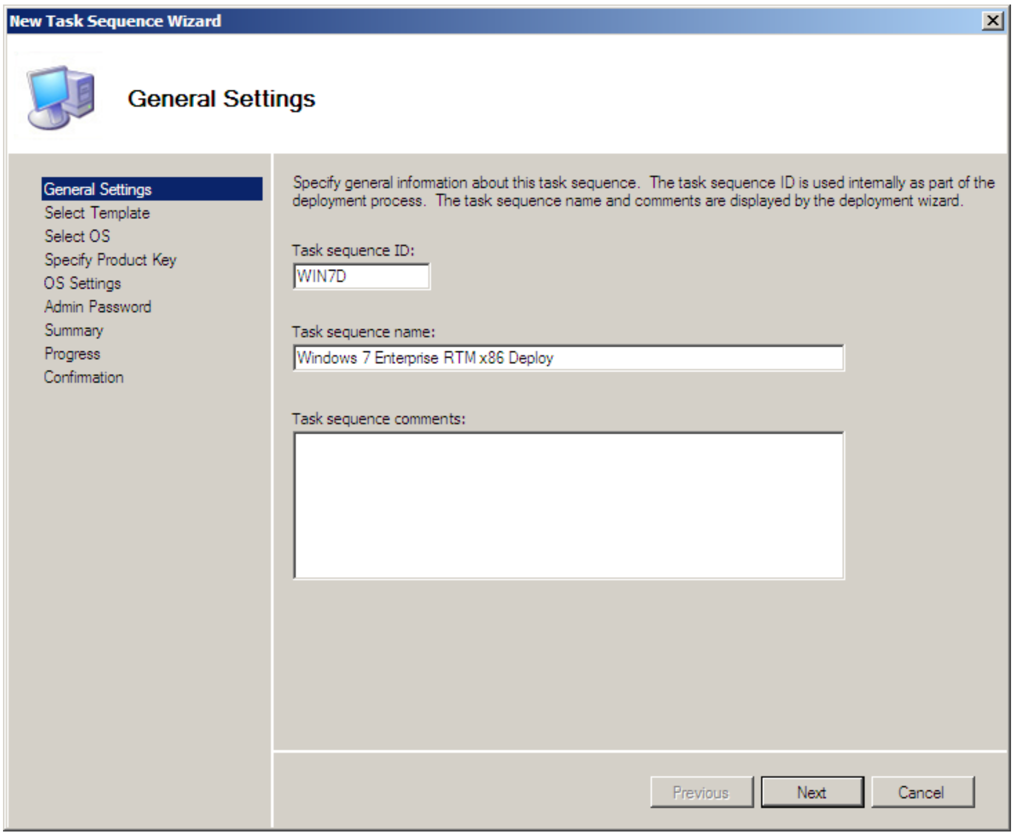
Once the captured WIM image has been added as a new operating system, create a new Task Sequence that uses the image as the basis for a build.

1. In the Deployment Workbench, right-click **Task Sequences \ Healthcare MDT 2010 \ DEPLOY** and select **New Task Sequence**:



The **New Task Sequence Wizard** starts and displays the **General Settings** page.

2. Type a **Task sequence ID**, **Task sequence name** and **Task sequence comments** and click **Next**:



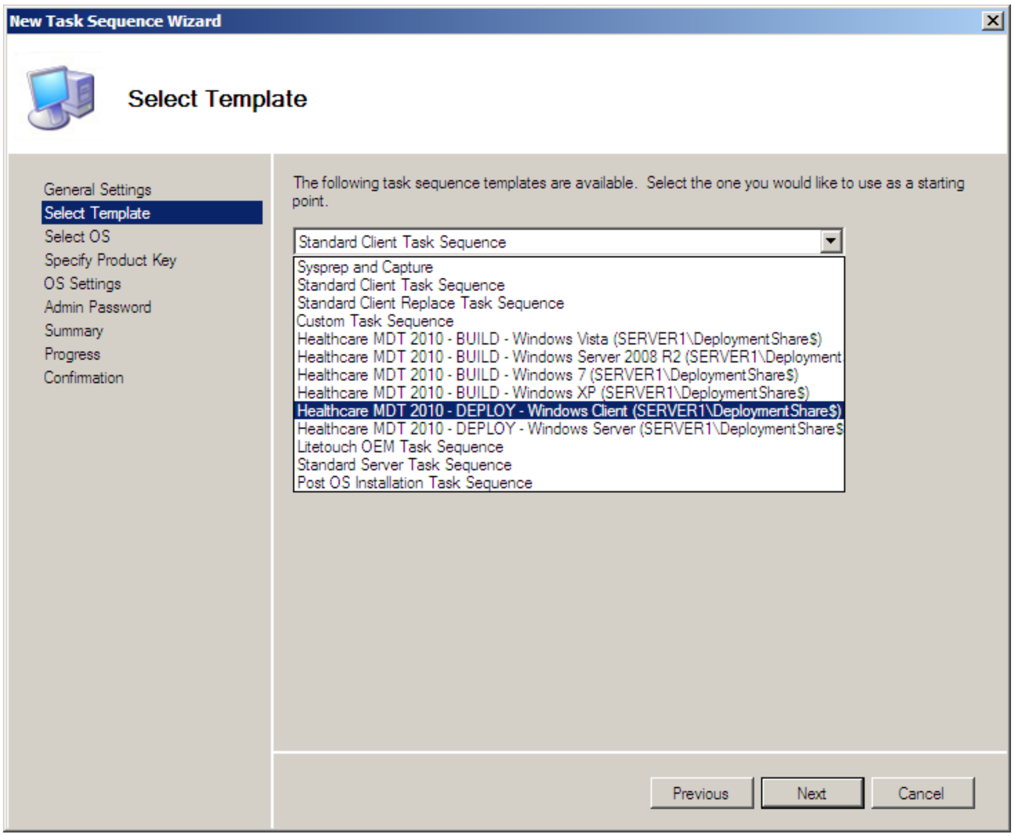
Specify general information about this task sequence. The task sequence ID is used internally as part of the deployment process. The task sequence name and comments are displayed by the deployment wizard.

Task sequence ID: WIN7D

Task sequence name: Windows 7 Enterprise RTM x86 Deploy

Task sequence comments:

3. On the **Select Template** page, select the Windows Client template of **Healthcare MDT 2010 – DEPLOY – Windows Client** and click **Next**:

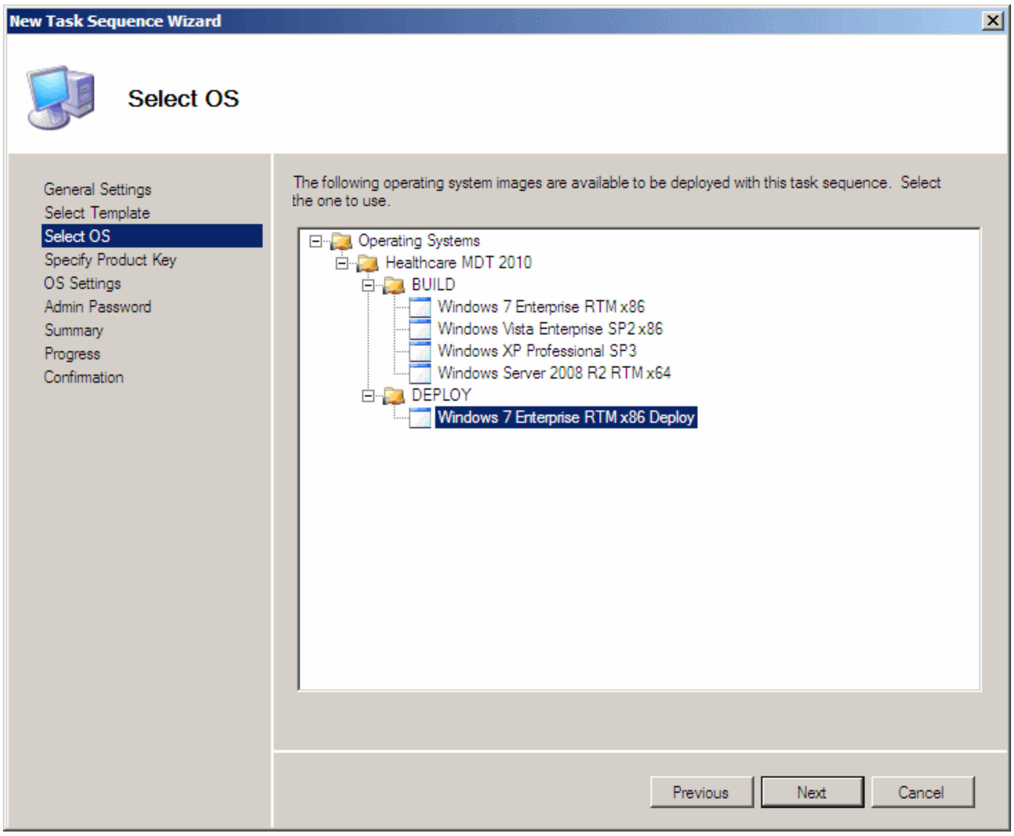


The following task sequence templates are available. Select the one you would like to use as a starting point.

Standard Client Task Sequence

Sysprep and Capture
Standard Client Task Sequence
Standard Client Replace Task Sequence
Custom Task Sequence
Healthcare MDT 2010 - BUILD - Windows Vista (SERVER1\DeploymentShare$)
Healthcare MDT 2010 - BUILD - Windows Server 2008 R2 (SERVER1\Deployment
Healthcare MDT 2010 - BUILD - Windows 7 (SERVER1\DeploymentShare$)
Healthcare MDT 2010 - BUILD - Windows XP (SERVER1\DeploymentShare$)
Healthcare MDT 2010 - DEPLOY - Windows Client (SERVER1\DeploymentShare$)
Healthcare MDT 2010 - DEPLOY - Windows Server (SERVER1\DeploymentShare$
Litetouch OEM Task Sequence
Standard Server Task Sequence
Post OS Installation Task Sequence

4. On the **Select OS** page, select the operating system previously imported and click **Next**:

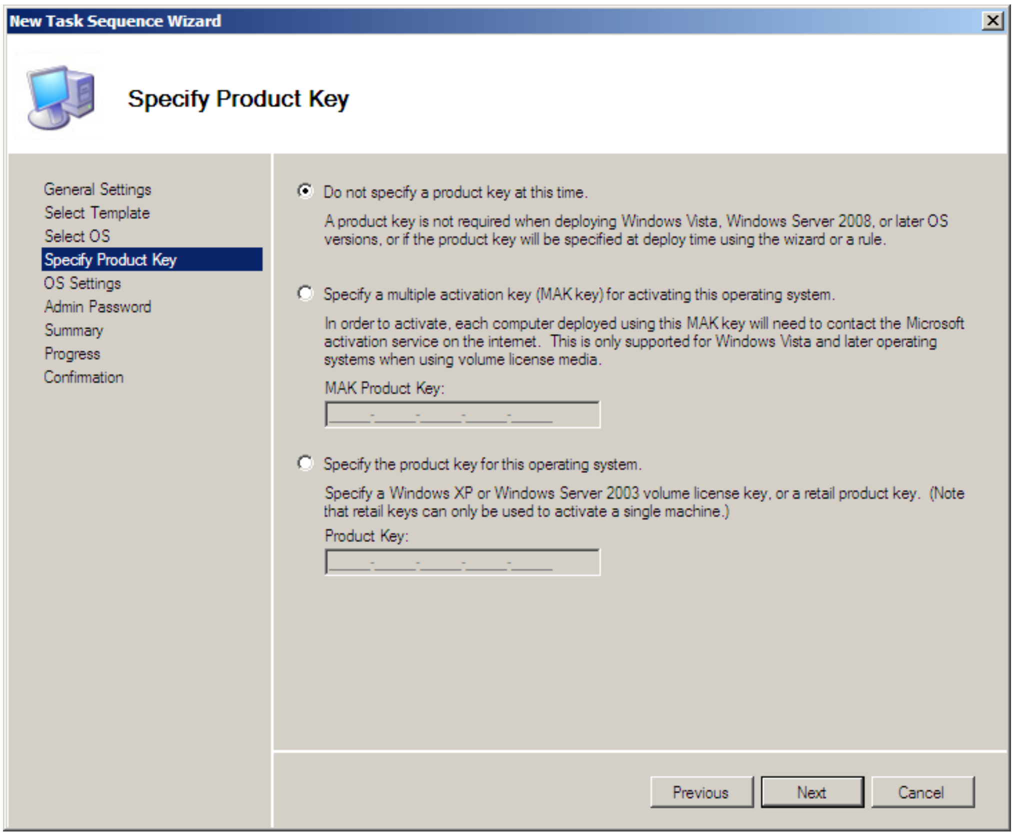


The following operating system images are available to be deployed with this task sequence. Select the one to use.

Operating Systems
Healthcare MDT 2010
BUILD
Windows 7 Enterprise RTM x86
Windows Vista Enterprise SP2 x86
Windows XP Professional SP3
Windows Server 2008 R2 RTM x64
DEPLOY
Windows 7 Enterprise RTM x86 Deploy

5. On the **Specify Product Key** page, do one of the following, and click **Next**:

   - If you are using a Key Management Server (KMS) with Windows Vista or later ,select **Do not specify a product key at this time**
   - If you are using a Multiple Activation Key (MAK) with Windows Vista or later, select **Specify a multiple…** and enter a product key
   - If you are using Windows XP, select **Specify the product key…** and enter a product key



New Task Sequence Wizard

Specify Product Key

● Do not specify a product key at this time.

A product key is not required when deploying Windows Vista, Windows Server 2008, or later OS versions, or if the product key will be specified at deploy time using the wizard or a rule.

○ Specify a multiple activation key (MAK key) for activating this operating system.

In order to activate, each computer deployed using this MAK key will need to contact the Microsoft activation service on the internet. This is only supported for Windows Vista and later operating systems when using volume license media.

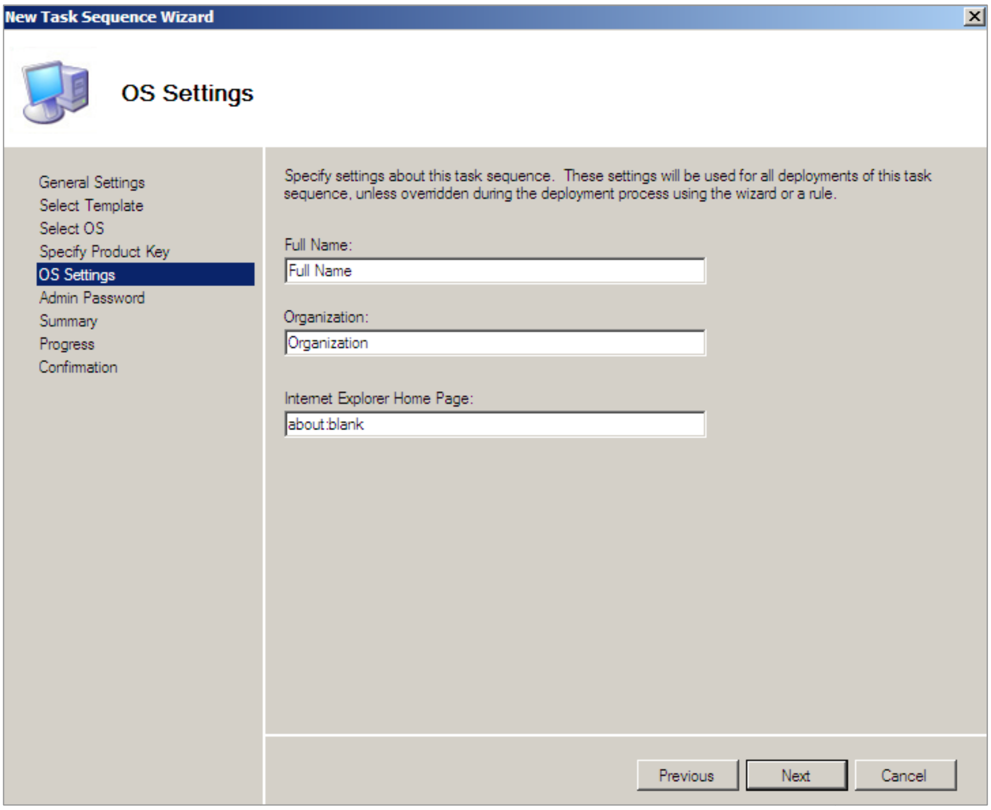
MAK Product Key:

○ Specify the product key for this operating system.

Specify a Windows XP or Windows Server 2003 volume license key, or a retail product key. (Note that retail keys can only be used to activate a single machine.)

Product Key:

6. On the **OS Settings** page, enter the **Full Name** (User Name), **Organization** and **Internet Explorer Home Page** that you want to use for the build and click **Next**:



New Task Sequence Wizard

OS Settings

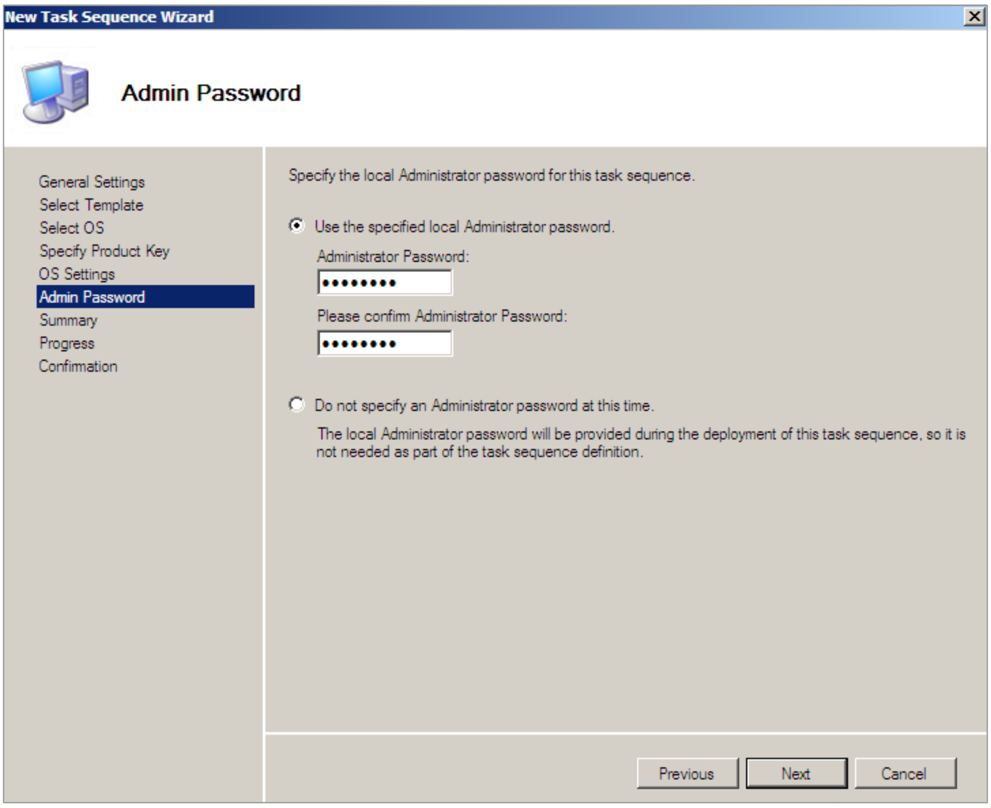
Specify settings about this task sequence. These settings will be used for all deployments of this task sequence, unless overridden during the deployment process using the wizard or a rule.

Full Name: Full Name

Organization: Organization

Internet Explorer Home Page: about:blank

7. On the **Admin Password** page, enter and confirm the local Administrator password that you want to be set on this Task Sequence, or click **Do not specify an Administrators password at this time**. Click **Next**:



New Task Sequence Wizard

Admin Password

Specify the local Administrator password for this task sequence.

● Use the specified local Administrator password.

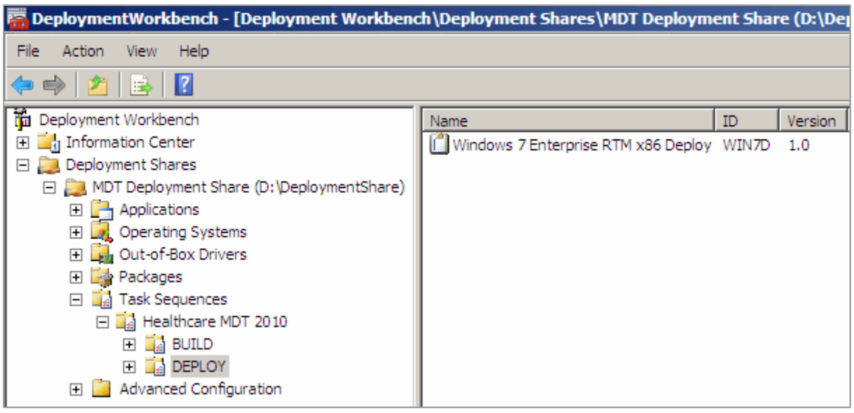
Administrator Password: ••••••••

Please confirm Administrator Password: ••••••••

○ Do not specify an Administrator password at this time.

The local Administrator password will be provided during the deployment of this task sequence, so it is not needed as part of the task sequence definition.

8. Accept the defaults on all remaining pages and click **Finish**. The task sequence will be created in the Deployment Workbench:

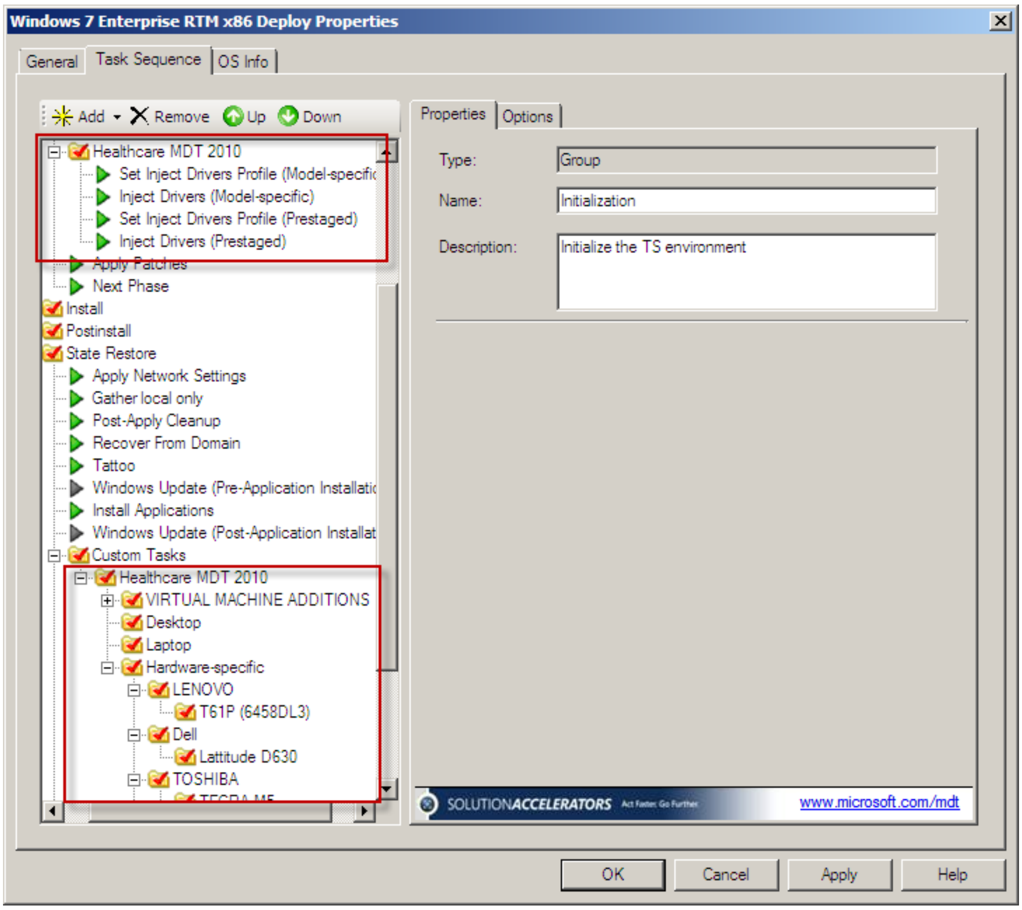


DeploymentWorkbench - [Deployment Workbench\Deployment Shares\MDT Deployment Share (D:\De
File
Action
View
Help

Deployment Workbench
- Information Center
- Deployment Shares
  - MDT Deployment Share (D:\DeploymentShare)
    - Applications
    - Operating Systems
    - Out-of-Box Drivers
    - Packages
    - Task Sequences
      - Healthcare MDT 2010
        - BUILD
        - DEPLOY
    - Advanced Configuration

Name	ID	Version
Windows 7 Enterprise RTM x86 Deploy	WIN7D	1.0

The custom Healthcare Extensions for MDT 2010 Task Sequence entries can be accessed by right-clicking the Task Sequence, selecting **Properties** and then the **Task Sequence** tab as shown in Figure 18:



- Healthcare MDT 2010
  - Set Inject Drivers Profile (Model-specific
  - Inject Drivers (Model-specific)
  - Set Inject Drivers Profile (Prestaged)
  - Inject Drivers (Prestaged)
- Apply Patches
- Next Phase
- Install
- Postinstall
- State Restore
  - Apply Network Settings
  - Gather local only
  - Post-Apply Cleanup
  - Recover From Domain
  - Tattoo
  - Windows Update (Pre-Application Installatic
  - Install Applications
  - Windows Update (Post-Application Installat
  - Custom Tasks
    - Healthcare MDT 2010
      - VIRTUAL MACHINE ADDITIONS
      - Desktop
      - Laptop
      - Hardware-specific
        - LENOVO
          - T61P (6458DL3)
        - Dell
          - Lattitude D630
        - TOSHIBA
          - TECRA M5

Type: Group

Name: Initialization

Description: Initialize the TS environment

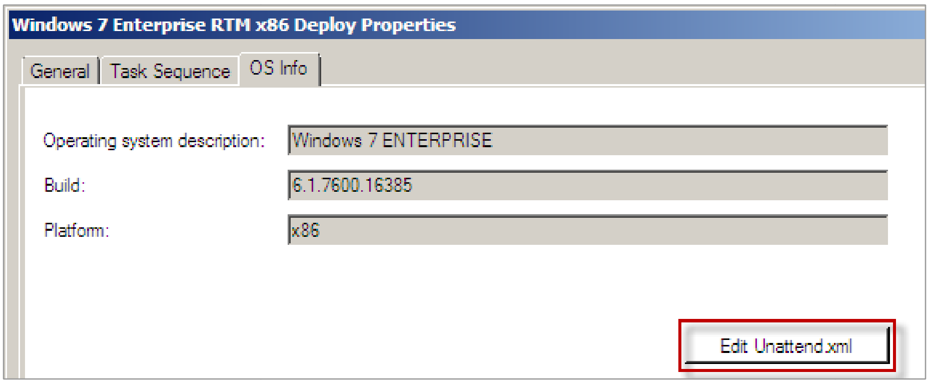
Figure 18: Healthcare Extensions for MDT 2010 Windows 7 Task Sequence

6.1.2.1 Windows Vista and Windows 7 Default Profiles

In Windows XP, any changes and customisations made to the Administrator account during the build are automatically applied to all users of that workstation once the machine is deployed. In Windows Vista and later, the Unattend.xml file must be manually edited to enable this functionality.

To manually edit the Unattend.xml file:

1. Select the Deploy Task Sequence created in section 6.1.2 and edit its properties.
2. Select the **OS Info** tab and click **Edit Unattend.xml** to launch **Windows System Image Manager**:

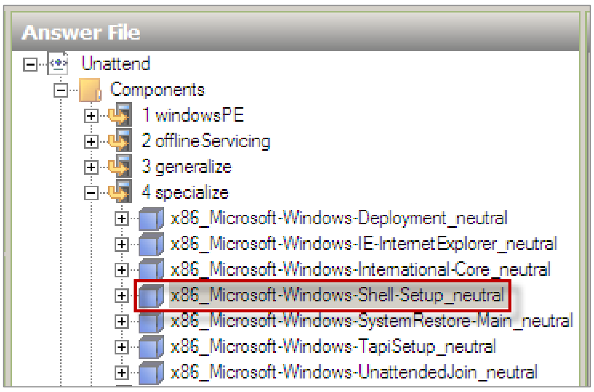


Operating system description: Windows 7 ENTERPRISE

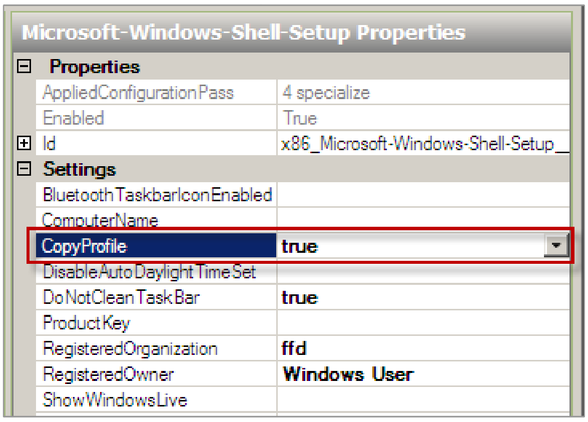
Build: 6.1.7600.16385

Platform: x86

3. In the **Answer File** pane, expand **4 specialize** and highlight **x86_Microsoft-Windows-Shell-Setup_neutral**:


Answer File
Unattend
Components
1 windowsPE
2 offlineServicing
3 generalize
4 specialize
x86_Microsoft-Windows-Deployment_neutral
x86_Microsoft-Windows-IE-InternetExplorer_neutral
x86_Microsoft-Windows-International-Core_neutral
x86_Microsoft-Windows-Shell-Setup_neutral
x86_Microsoft-Windows-SystemRestore-Main_neutral
x86_Microsoft-Windows-TapiSetup_neutral
x86_Microsoft-Windows-UnattendedJoin_neutral

4. In the **Microsoft-Windows-Shell-Setup Properties** pane, change the **CopyProfile** value to **true**:



Properties	
AppliedConfigurationPass	4 specialize
Enabled	True
Id	x86_Microsoft-Windows-Shell-Setup_
Settings	
BluetoothTaskbarIconEnabled	
ComputerName	
CopyProfile	true
DisableAutoDaylightTimeSet	
DoNotCleanTaskBar	true
ProductKey	
RegisteredOrganization	ffd
RegisteredOwner	Windows User
ShowWindowsLive	

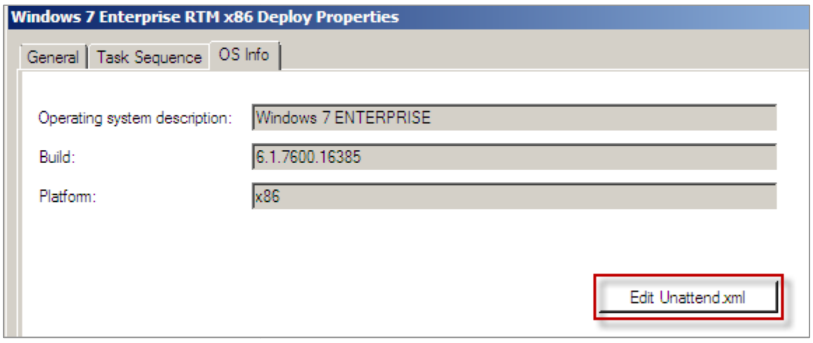
5. Close the **Windows System Image Manager** and save the changes.

6.1.2.2 Windows Vista and Windows 7 Display Resolution

By default, MDT 2010 sets the display resolution of deployed machines to 1024x768. This may not be appropriate. A preferred approach is to allow the machine to automatically determine the best display resolution to use.

To automatically determine the best display resolution:

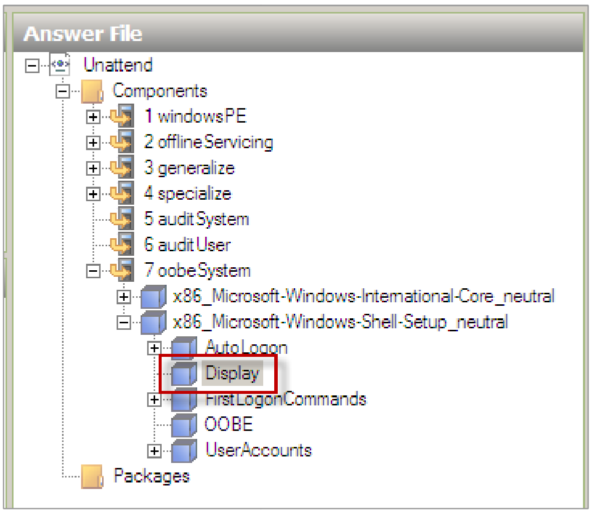
1. Select the Deploy Task Sequence created in section 6.1.2 and edit its properties.
2. Select the **OS Info** tab and click **Edit Unattend.xml** to launch **Windows System Image Manager**:



Windows 7 Enterprise RTM x86 Deploy Properties

Operating system description: Windows 7 ENTERPRISE
Build: 6.1.7600.16385
Platform: x86

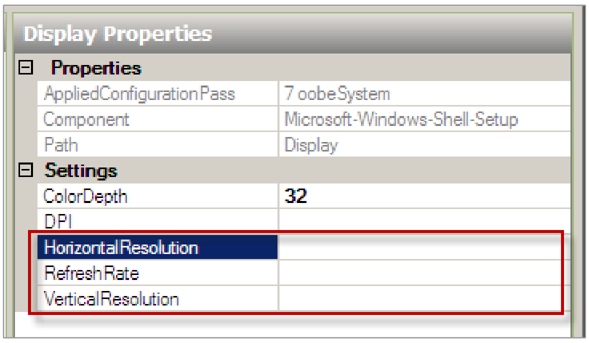
3. In the **Answer File** pane, expand **7 oobeSystem** and highlight **x86_Microsoft-Windows-Shell-Setup_neutral \ Display**:



Answer File

```
Unattend
  Components
    1 windowsPE
    2 offlineServicing
    3 generalize
    4 specialize
    5 auditSystem
    6 auditUser
    7 oobeSystem
      x86_Microsoft-Windows-International-Core_neutral
      x86_Microsoft-Windows-Shell-Setup_neutral
        AutoLogon
        Display
        FirstLogonCommands
        OOBE
        UserAccounts
  Packages
```

4. In the **Display Properties** pane, delete the contents of the **HorizontalResolution**, **RefreshRate** and **VerticalResolution** entries:



Display Properties

Properties	
AppliedConfigurationPass	7 oobeSystem
Component	Microsoft-Windows-Shell-Setup
Path	Display
Settings	
ColorDepth	32
DPI	
HorizontalResolution	
RefreshRate	
VerticalResolution	

5. Close the **Windows System Image Manager** and save the changes.

6.2 Adding Deploy Drivers

As detailed in section 4.3, the Build image contains very few drivers in-box, as these will be added as the image is deployed to a particular workstation. Deploy drivers can be split into three categories:

- **Standard drivers** – These are drivers that are specific to the hardware being deployed to, and are installed using plug-and-play detection
- **Prestaged drivers** – These are drivers that may not exist on the hardware being deployed to, but may be required in the future. For example, when a user plugs in a USB device
- **Driver applications** – These are hardware-specific applications that are not installed by plug-and-play. The most common examples are hotkey and on-screen-display power management applications

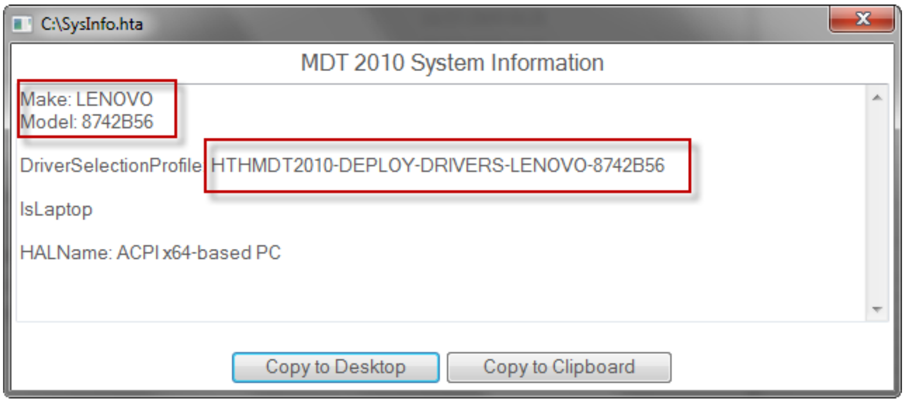
6.2.1 Adding Standard Drivers

Healthcare Extensions for MDT 2010 handles standard drivers on a model by model basis. The sequence for adding standard drivers is as follows:

1. Use SysInfo to determine Make and Model.
2. Create folder in Out-of-Box Drivers.
3. Add driver source files.
4. Create the driver Selection Profile.

To add a standard driver:

1. Run the SysInfo tool located in **<Distribution> \ Applications \ HTHMDT2010 \ Tools**. Record the Make, Model and DriverSelectionProfile:



MDT 2010 System Information

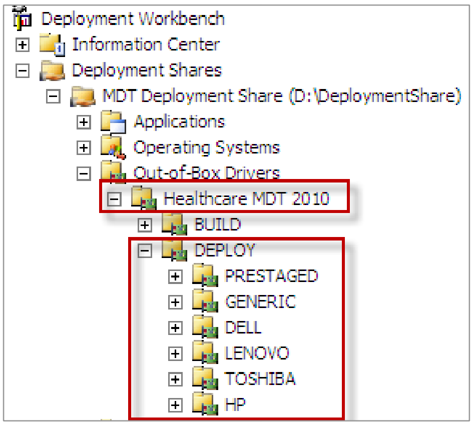
Make: LENOVO
Model: 8742B56

DriverSelectionProfile HTHMDT2010-DEPLOY-DRIVERS-LENOVO-8742B56

IsLaptop

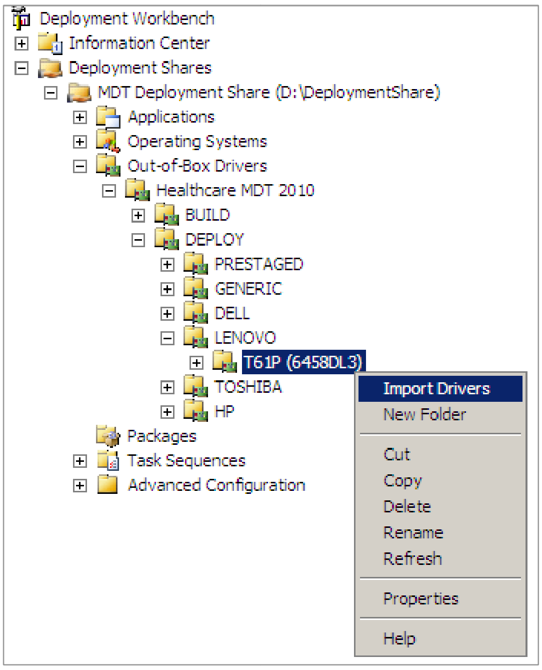
HALName: ACPI x64-based PC

2. Create a new make and model folder in the Deployment Workbench with the name **Out-of-Box Drivers Healthcare MDT 2010 \ DEPLOY \ <Make> \ <Model>**. If the model name is not very descriptive, use a more obvious name with the actual model name in brackets, as shown in the following T61P example:



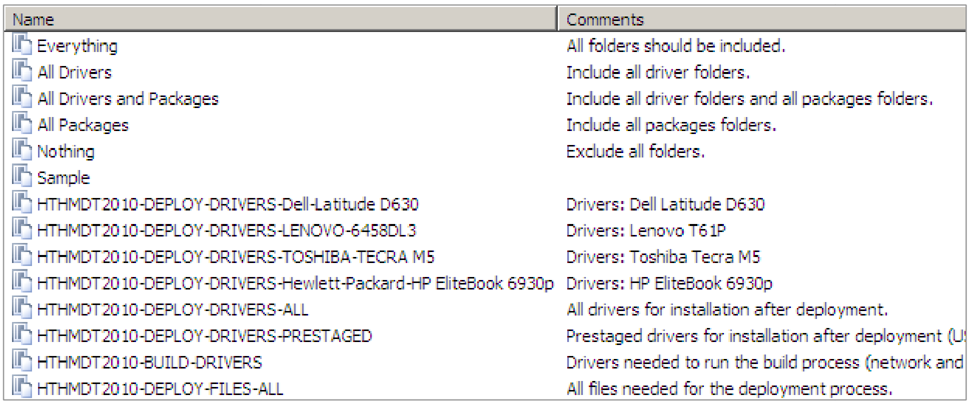
```
Deployment Workbench
⊞ Information Center
⊟ Deployment Shares
    ⊟ MDT Deployment Share (D:\DeploymentShare)
        ⊞ Applications
        ⊞ Operating Systems
        ⊟ Out-of-Box Drivers
            ⊟ Healthcare MDT 2010
                ⊞ BUILD
                ⊟ DEPLOY
                    ⊞ PRESTAGED
                    ⊞ GENERIC
                    ⊞ DELL
                    ⊞ LENOVO
                    ⊞ TOSHIBA
                    ⊞ HP
```

3. Right-click on the new folder and select **Import Drivers** and use the wizard to browse to the location containing all the drivers for the required hardware:



```
Deployment Workbench
⊞ Information Center
⊟ Deployment Shares
    ⊟ MDT Deployment Share (D:\DeploymentShare)
        ⊞ Applications
        ⊞ Operating Systems
        ⊟ Out-of-Box Drivers
            ⊟ Healthcare MDT 2010
                ⊞ BUILD
                ⊟ DEPLOY
                    ⊞ PRESTAGED
                    ⊞ GENERIC
                    ⊞ DELL
                    ⊟ LENOVO
                        ⊞ T61P (6458DL3)
                    ⊞ TOSHIBA
                    ⊞ HP
        Packages
        ⊞ Task Sequences
        ⊞ Advanced Configuration
```

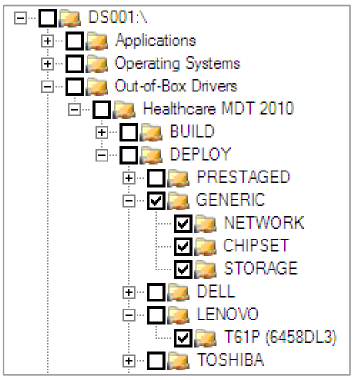
4. Open the **Advanced Configuration \ Selection Profiles** node in the Deployment Workbench:



Name	Comments
Everything	All folders should be included.
All Drivers	Include all driver folders.
All Drivers and Packages	Include all driver folders and all packages folders.
All Packages	Include all packages folders.
Nothing	Exclude all folders.
Sample	
HTHMDT2010-DEPLOY-DRIVERS-Dell-Latitude D630	Drivers: Dell Latitude D630
HTHMDT2010-DEPLOY-DRIVERS-LENOVO-6458DL3	Drivers: Lenovo T61P
HTHMDT2010-DEPLOY-DRIVERS-TOSHIBA-TECRA M5	Drivers: Toshiba Tecra M5
HTHMDT2010-DEPLOY-DRIVERS-Hewlett-Packard-HP EliteBook 6930p	Drivers: HP EliteBook 6930p
HTHMDT2010-DEPLOY-DRIVERS-ALL	All drivers for installation after deployment.
HTHMDT2010-DEPLOY-DRIVERS-PRESTAGED	Prestaged drivers for installation after deployment (U
HTHMDT2010-BUILD-DRIVERS	Drivers needed to run the build process (network and
HTHMDT2010-DEPLOY-FILES-ALL	All files needed for the deployment process.

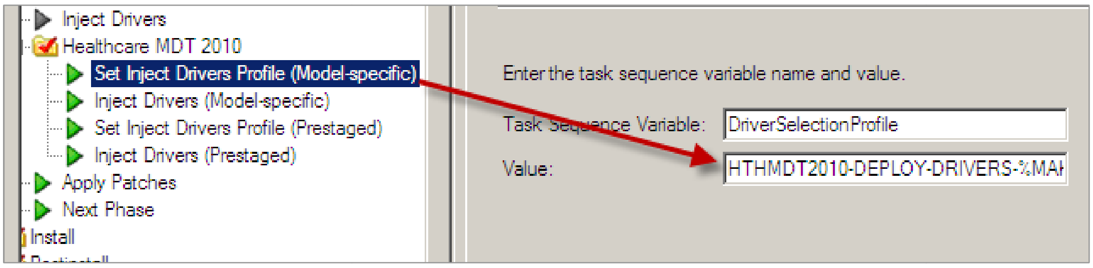
5. Create a new Selection Profile with the name shown for the **DriverSelectionProfile** in the SysInfo tool. Check the box that contains the new make and model folder created previously.

> **Note:**
>
> The name must be created **exactly** as shown in the SysInfo tool – it is recommended that the name is copied and pasted to avoid errors.



6. The driver has now been added, and will be deployed on the specified hardware.

The make and model driver deployment process works due to a customisation made in Healthcare Extensions for MDT 2010. The template Task Sequences have been preconfigured to use the Selection Profile in a way that caters for multiple hardware types. The Task Sequence customisation is shown in Figure 19, and is used whenever a Healthcare Extensions for MDT 2010 **DEPLOY** Task Sequence is used:


Inject Drivers
Healthcare MDT 2010
Set Inject Drivers Profile (Model-specific)
Inject Drivers (Model-specific)
Set Inject Drivers Profile (Prestaged)
Inject Drivers (Prestaged)
Apply Patches
Next Phase
Install

Enter the task sequence variable name and value.

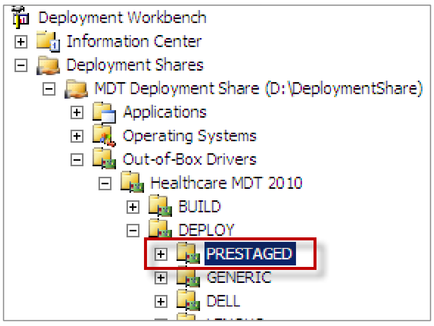
Task Sequence Variable: DriverSelectionProfile

Value: HTHMDT2010-DEPLOY-DRIVERS-%MAI

Figure 19: Healthcare Extensions for MDT 2010 Model-specific Task Sequence

6.2.2 Adding Prestaged Drivers

Right-click on the Out-of-Box Drivers Healthcare MDT 2010 \ DEPLOY \ PRESTAGED folder and select Import Drivers:



All drivers added to the PRESTAGED folder will be installed during deployment, regardless of the hardware used.

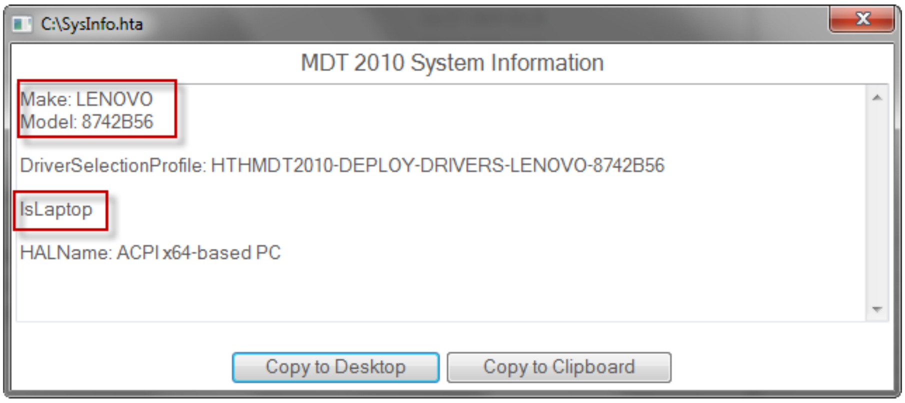
6.2.3 Adding Driver Applications

The sequence for adding driver applications is as follows:

1. Create folder in Out-of-Box Drivers
2. Add driver source files
3. Use SysInfo to determine and create the driver Selection Profile

To add driver applications:

1. Run the SysInfo tool located in **<Distribution> \ Applications \ HTHMDT2010 \ Tools**. Record the Make, Model and chassis type (IsLaptop, IsDesktop or IsServer):



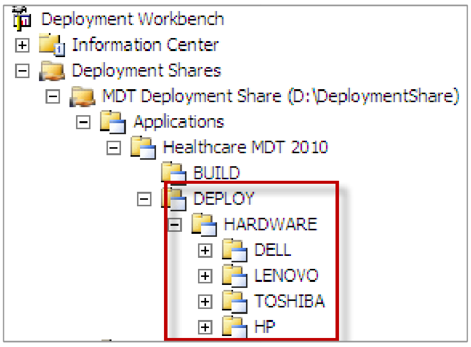
Make: LENOVO
Model: 8742B56

DriverSelectionProfile: HTHMDT2010-DEPLOY-DRIVERS-LENOVO-8742B56

IsLaptop

HALName: ACPI x64-based PC

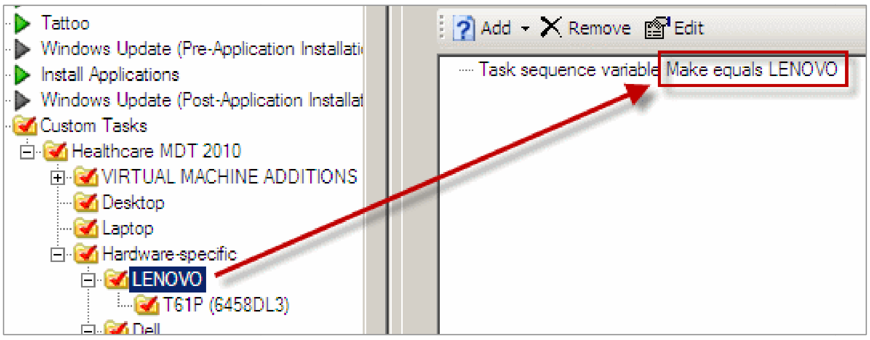
2. Create a new application in the Deployment Workbench, as detailed in section 4.4. The application should be added in the **Applications \ Healthcare MDT 2010 \ DEPLOY \ HARDWARE** node in a suitable sub-folder:



- Deployment Workbench
  - Information Center
  - Deployment Shares
    - MDT Deployment Share (D:\DeploymentShare)
      - Applications
        - Healthcare MDT 2010
          - BUILD
          - DEPLOY
            - HARDWARE
              - DELL
              - LENOVO
              - TOSHIBA
              - HP

3. Open the Task Sequence that will be used to deploy the operating system.

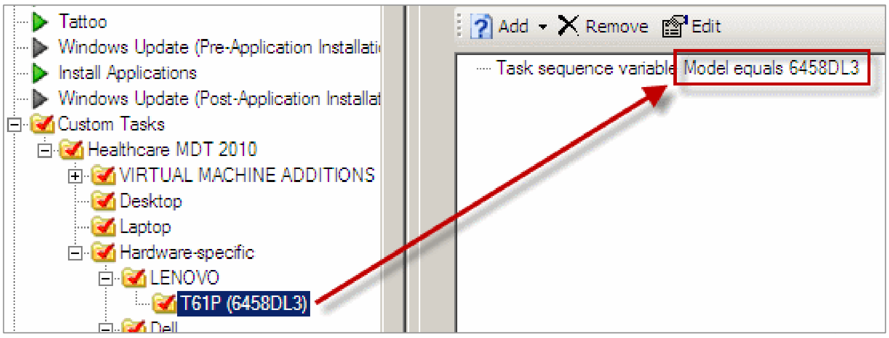
4. Add a folder corresponding to the hardware **make**, as determined in step 1. Add a condition to the folder of **Task Sequence variable Make equals <MAKE>**, where <MAKE> is the make determined in step 1. The make used in the Task Sequence condition must **exactly** match that shown in SysInfo:



- Tattoo
- Windows Update (Pre-Application Installati
- Install Applications
- Windows Update (Post-Application Installat
- Custom Tasks
  - Healthcare MDT 2010
    - VIRTUAL MACHINE ADDITIONS
    - Desktop
    - Laptop
    - Hardware-specific
      - LENOVO
        - T61P (6458DL3)
      - Dell

Task sequence variable Make equals LENOVO

5. Add a folder corresponding to the hardware **model**, as determined in step 1. Add a condition to the folder of **Task Sequence variable Model equals <MODEL>**, where <MODEL> is the model determined in step 1. The model used in the Task Sequence condition must **exactly** match that shown in SysInfo:



Task sequence variable Model equals 6458DL3

6. Add the driver application to this folder by clicking **Add \ General \ Install Application**.

7. Close the Task Sequence.

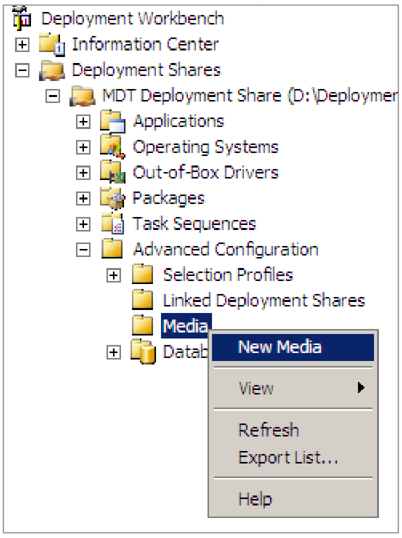
6.3 Deployment Types

In additional to the standard Deployment Share shown in previous sections, it is possible to create additional Deployment Shares:

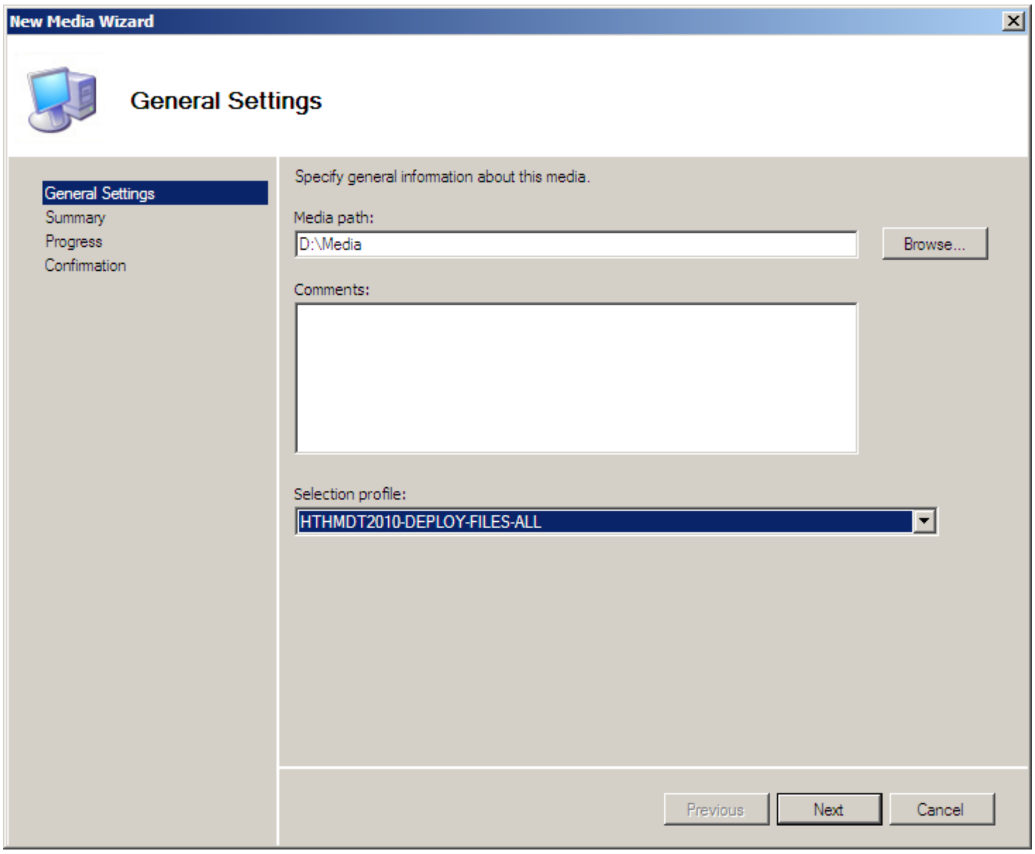
- **DVD or USB** – The captured WIM image and MDT 2010 configuration scripts are copied to a bootable DVD that can be used to deploy the image in remote locations, such as clinician surgeries
- **OEM** – This is an extended form of the DVD/USB build
- **Linked Deployment Share** – A subset of the standard Deployment Share is replicated to an additional network location

6.3.1 Creating DVD and USB Deployment Media

1. In the Deployment Workbench, right-click **Advanced Configuration \ Media** and select **New Media**:



2. Select an empty folder to be used as the media point and change the Selection profile to **HTHMDT2010-DEPLOY-FILES-ALL**. This will ensure that the media point only contains files needed for a deployment. Click **Next**:



New Media Wizard

General Settings

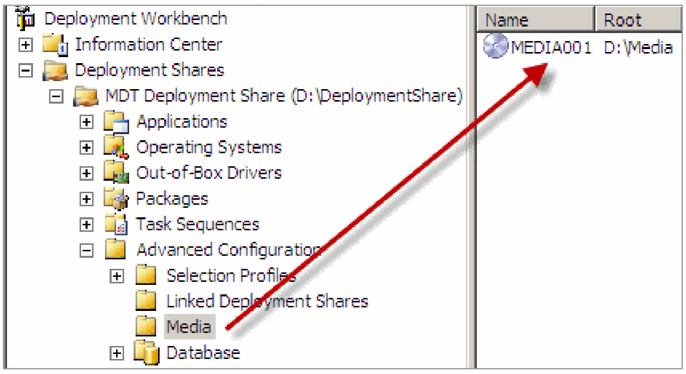
Specify general information about this media.

Media path: D:\Media

Comments:

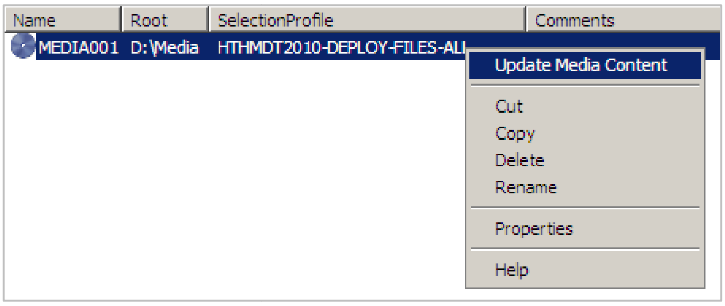
Selection profile: HTHMDT2010-DEPLOY-FILES-ALL

3. On the Summary page click **Finish**. A new media point will be shown in the Deployment Workbench:



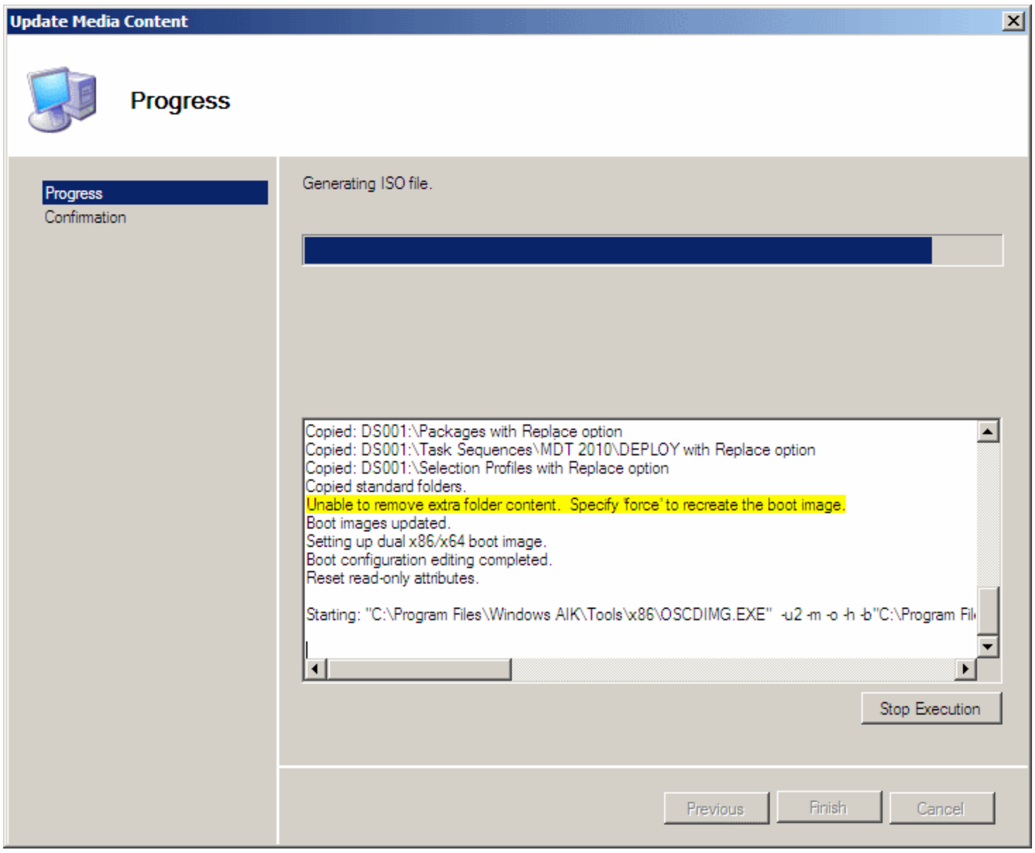
Name	Root
MEDIA001	D:\Media

4. Right-click on the newly created media point and select **Update Media Content**:



Name	Root	SelectionProfile	Comments
MEDIA001	D:\Media	HTHMDT2010-DEPLOY-FILES-ALL	

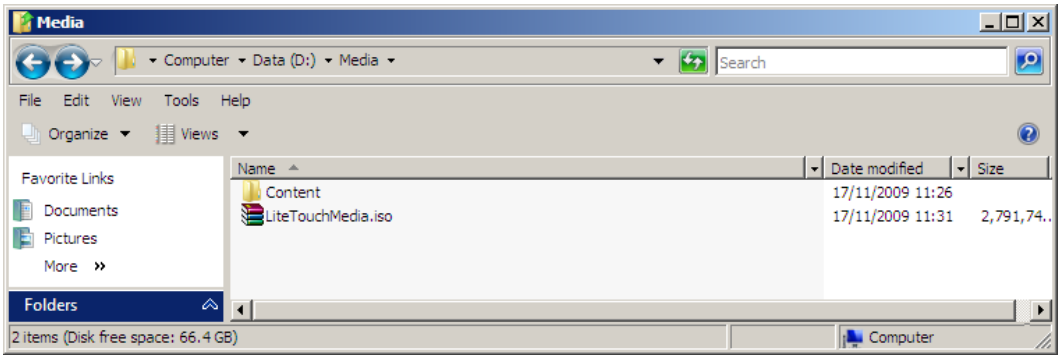
5. The deployment files will be created, which may take some time:



Generating ISO file.

```
Copied: DS001:\Packages with Replace option
Copied: DS001:\Task Sequences\MDT 2010\DEPLOY with Replace option
Copied: DS001:\Selection Profiles with Replace option
Copied standard folders.
Unable to remove extra folder content.  Specify 'force' to recreate the boot image.
Boot images updated.
Setting up dual x86/x64 boot image.
Boot configuration editing completed.
Reset read-only attributes.

Starting: "C:\Program Files\Windows AIK\Tools\x86\OSCDIMG.EXE"  -u2 -m -o -h -b"C:\Program Fil
```

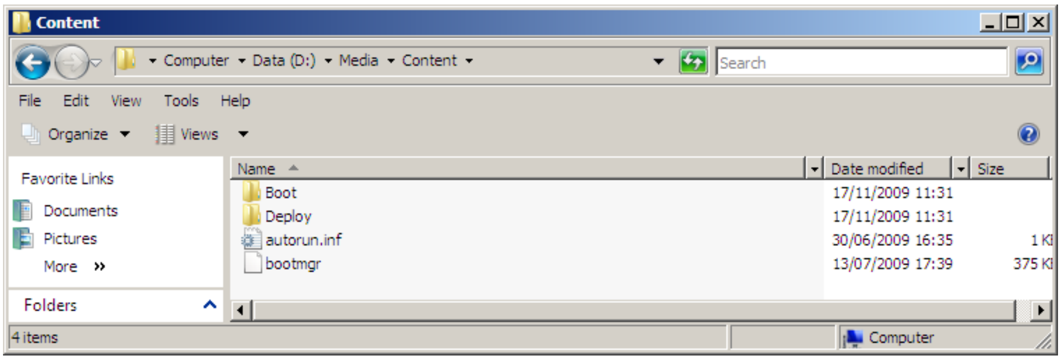
6. A **LiteTouchMedia.iso** file will be generated at the root of the media point. This can be burned to DVD using standard methods, and used as a standalone deployment DVD:



Name	Date modified	Size
Content	17/11/2009 11:26	
LiteTouchMedia.iso	17/11/2009 11:31	2,791,74..

2 items (Disk free space: 66.4 GB)

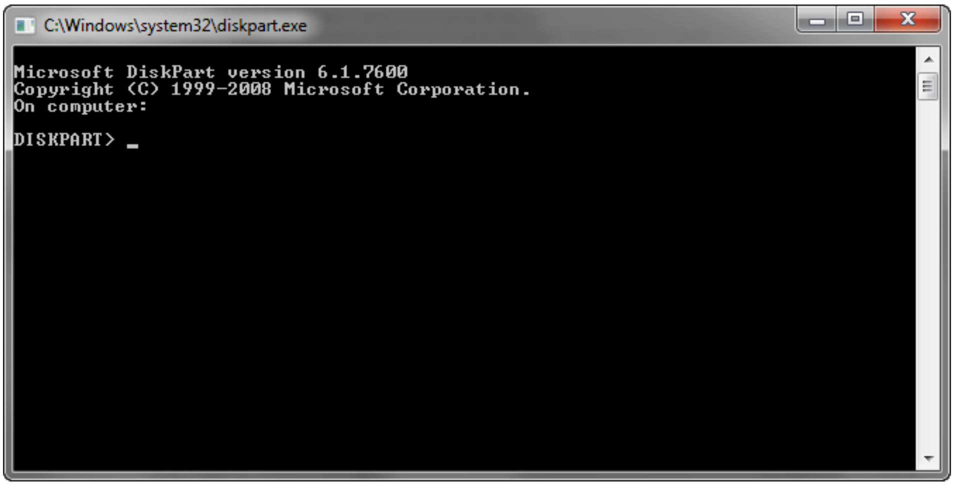
In the **Content** sub-folder of the media point USB deployment files will have been created:



Name	Date modified	Size
Boot	17/11/2009 11:31	
Deploy	17/11/2009 11:31	
autorun.inf	30/06/2009 16:35	1 KI
bootmgr	13/07/2009 17:39	375 KI

4 items

7. To use the USB deployment files, format and partition a USB device. Windows Vista or Windows 7 must be used to partition the USB device otherwise it will not be bootable. Once partitioned, run the **diskpart** utility from the command line:



C:\Windows\system32\diskpart.exe

```
Microsoft DiskPart version 6.1.7600
Copyright (C) 1999-2008 Microsoft Corporation.
On computer:

DISKPART> _
```

8. Identify the disk number of an inserted USB device by typing **list disk** (in this example the required disk number is **3**):

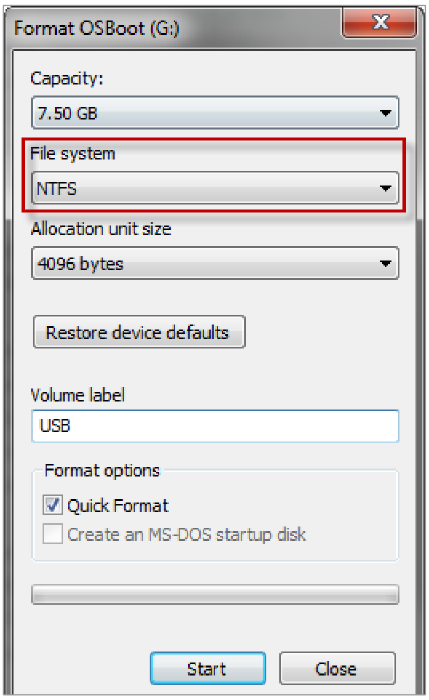
```
DISKPART> list disk

  Disk ###  Status         Size     Free     Dyn  Gpt
  --------  -------------  -------  -------  ---  ---
  Disk 0    Online           93 GB      0 B
  Disk 1    Online          298 GB      0 B
  Disk 3    Online         7681 MB      0 B
```

9. Select the disk, repartition it and make it active by following the example, substituting the disk number with that identified in the previous step.

> **Note:**
>
> This will delete all data on the USB device.

```
DISKPART> select disk 3

Disk 3 is now the selected disk.

DISKPART> clean

DiskPart succeeded in cleaning the disk.

DISKPART> create partition primary

DiskPart succeeded in creating the specified partition.

DISKPART> active

DiskPart marked the current partition as active.
```

10. Exit **diskpart**, and format the USB device in Windows Explorer using the NTFS file system:



Format OSBoot (G:)

Capacity: 7.50 GB

File system: NTFS

Allocation unit size: 4096 bytes

Volume label: USB

Format options

☑ Quick Format

☐ Create an MS-DOS startup disk

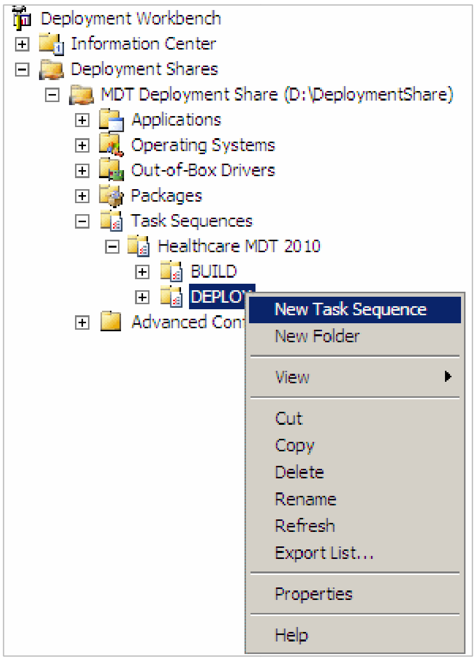
11. Copy the files from the **Content** folder to the root of the USB device. The USB device can now be used as standalone deployment media.

6.3.2 Creating an OEM Build

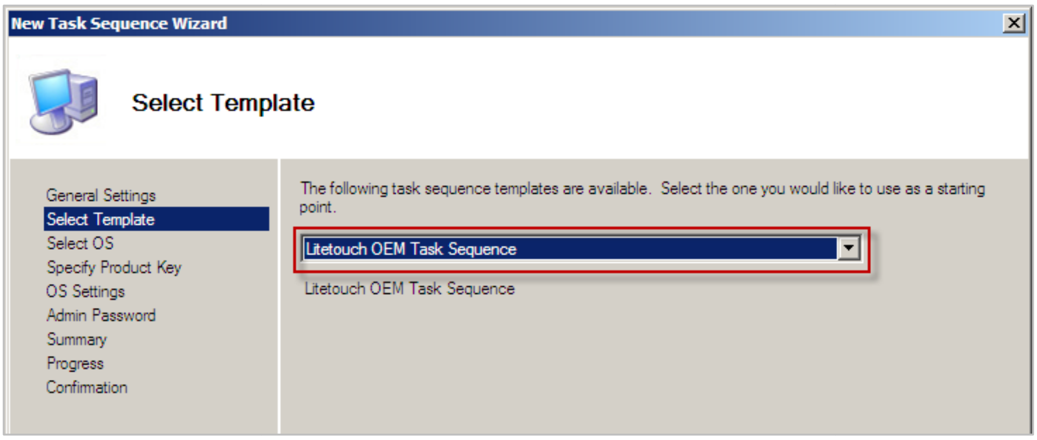
An OEM build is a special type of media build that is installed onto a workstation at an OEM.

To create an OEM build:

1. Create a media build as detailed in section 6.3.1.
2. In the Deployment Workbench, right-click the node **Task Sequences \ Healthcare MDT 2010 \ DEPLOY** and select **New Task Sequence**:



3. Enter a name and ID for the Task Sequence and click **Next**.
4. Select **Litetouch OEM Task Sequence** and click **Next**:



New Task Sequence Wizard

Select Template

The following task sequence templates are available. Select the one you would like to use as a starting point.

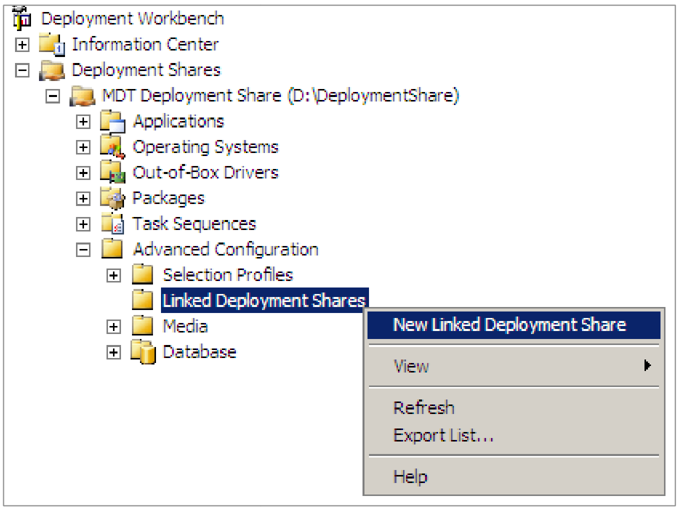
Litetouch OEM Task Sequence

Litetouch OEM Task Sequence

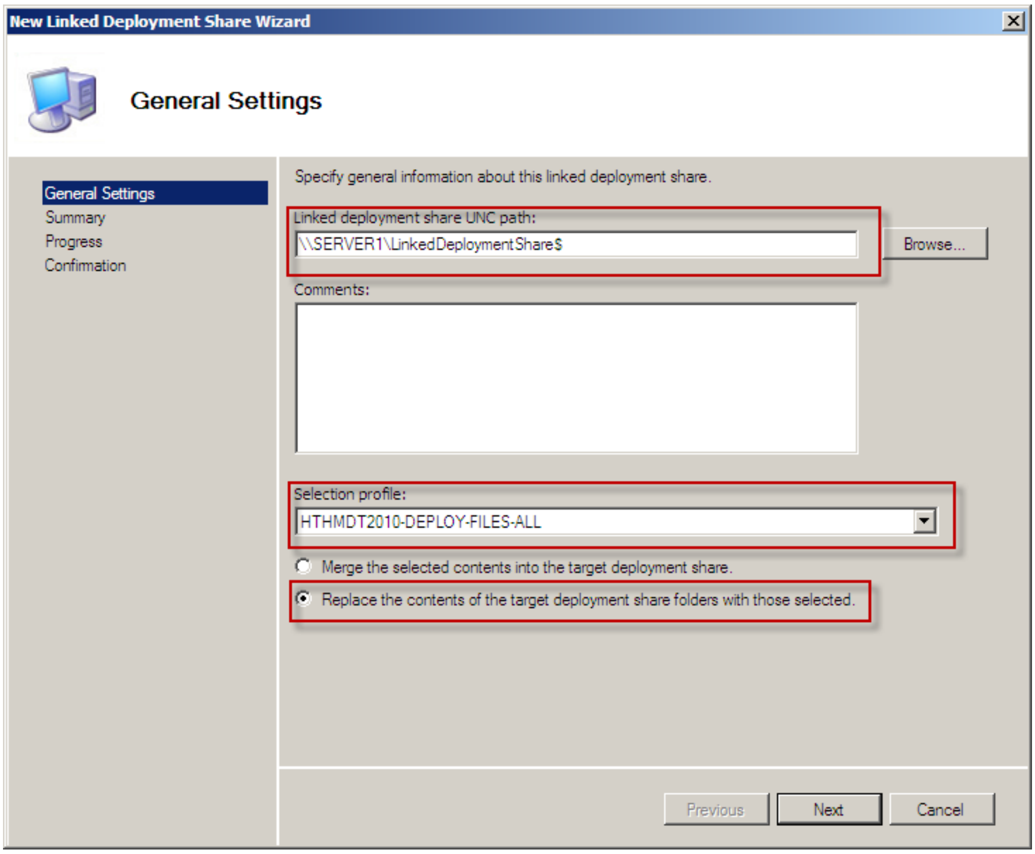
5. Accept the defaults on the remaining pages and click **Finish**.
6. Initiate a Build process using the standard procedure.
7. Select the newly created OEM Task Sequence when prompted.
8. Files will be copied, and the workstation will shut down when complete.
9. Send the workstation to the OEM for disk duplication.

6.3.3 Creating a Linked Deployment Share

1. In the Deployment Workbench, right-click **Advanced Configuration \ Linked Deployment Shares** and select **New Linked Deployment Share**:



2. Enter the UNC path of the new deployment share (this share must have previously been manually created). Change the Selection Profile to **HTHMDT2010-DEPLOY-FILES-ALL** and select **Replace the contents…** Click **Next**:


New Linked Deployment Share Wizard

General Settings

Specify general information about this linked deployment share.

Linked deployment share UNC path: \\SERVER1\LinkedDeploymentShare$

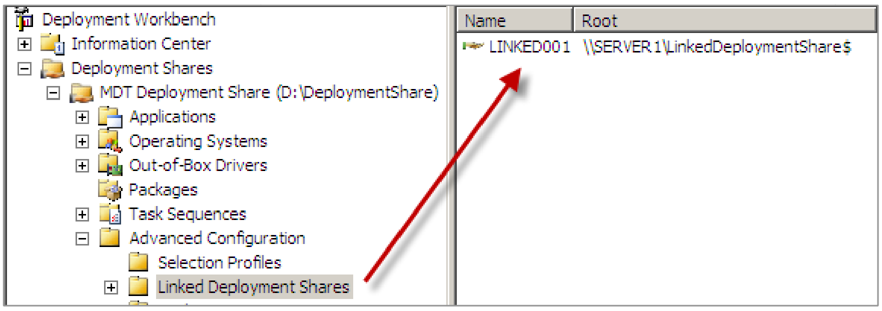
Comments:

Selection profile: HTHMDT2010-DEPLOY-FILES-ALL

○ Merge the selected contents into the target deployment share.

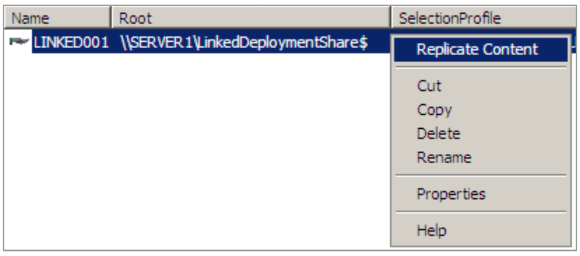
● Replace the contents of the target deployment share folders with those selected.

3. Click **Finish**. The new linked Deployment Share will be shown in the Deployment Workbench:



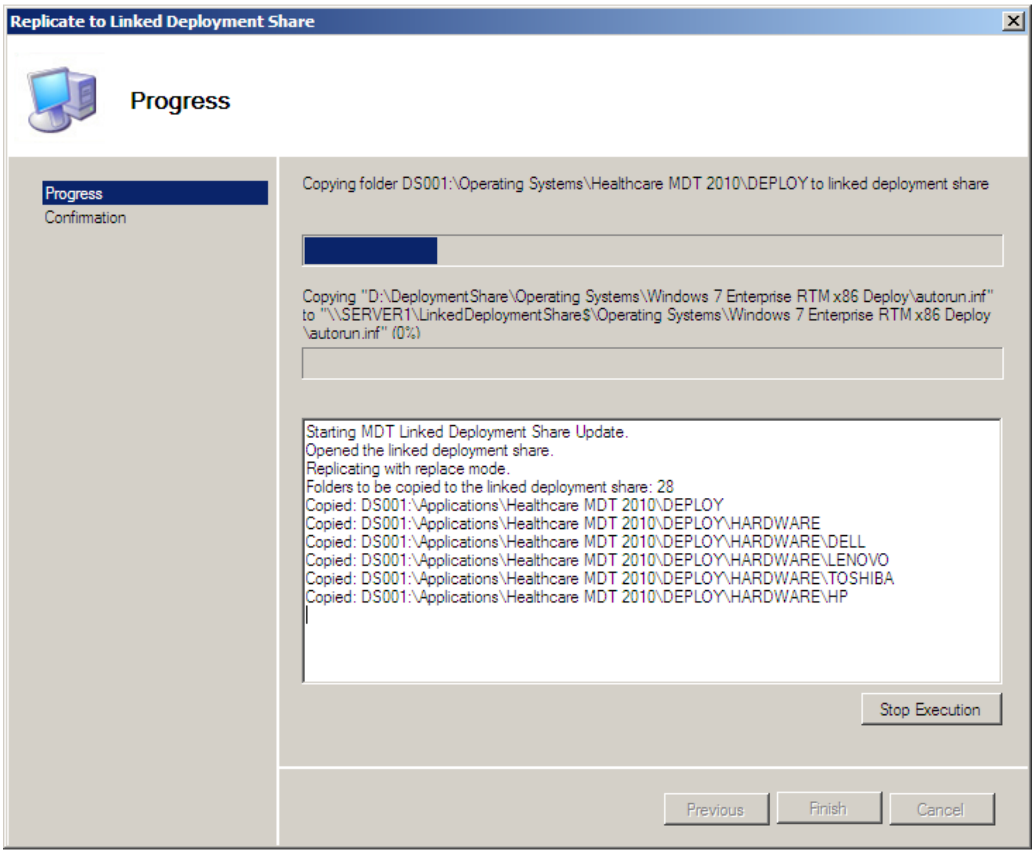
Name	Root
LINKED001	\\SERVER1\LinkedDeploymentShare$

4. Right-click on the linked Deployment Share and select **Replace Content**:



Name	Root	SelectionProfile
LINKED001	\\SERVER1\LinkedDeploymentShare$	

The content will be replicated which may take some time:



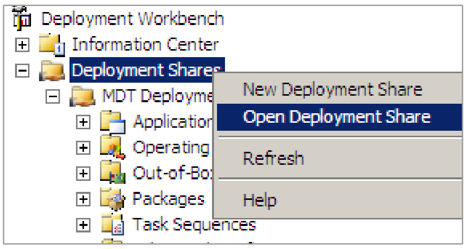
Copying folder DS001:\Operating Systems\Healthcare MDT 2010\DEPLOY to linked deployment share

Copying "D:\DeploymentShare\Operating Systems\Windows 7 Enterprise RTM x86 Deploy\autorun.inf" to "\\SERVER1\LinkedDeploymentShare$\Operating Systems\Windows 7 Enterprise RTM x86 Deploy\autorun.inf" (0%)

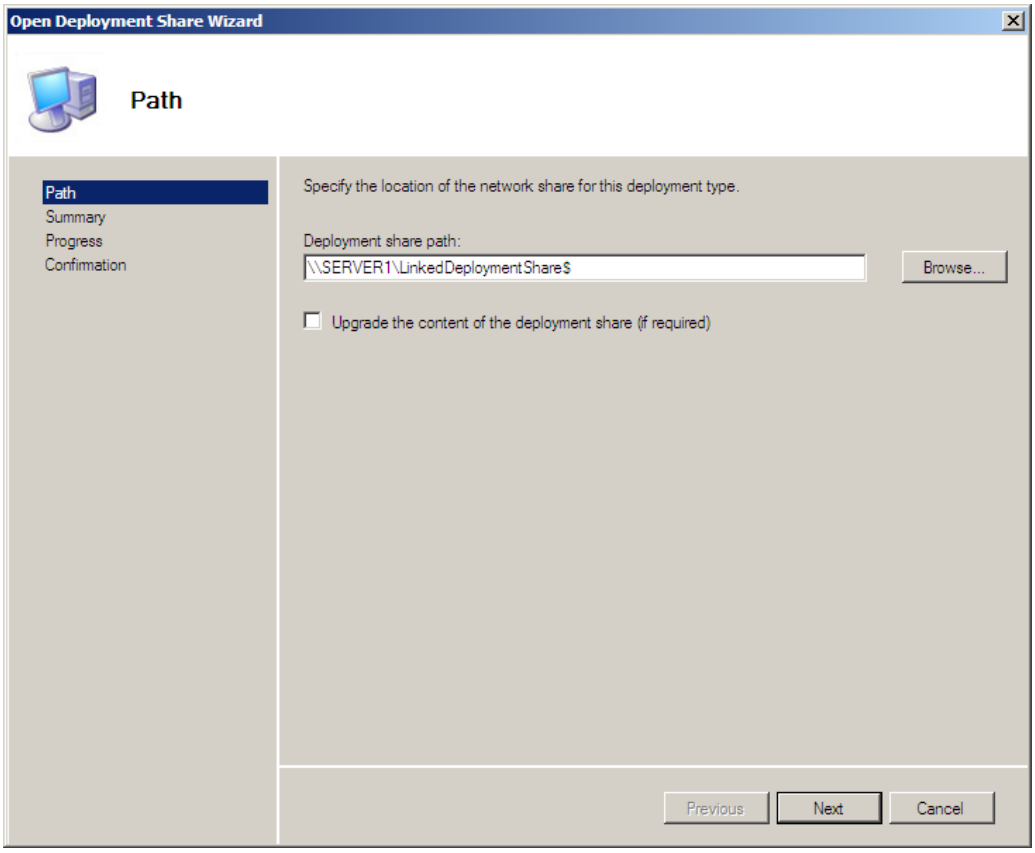
```
Starting MDT Linked Deployment Share Update.
Opened the linked deployment share.
Replicating with replace mode.
Folders to be copied to the linked deployment share: 28
Copied: DS001:\Applications\Healthcare MDT 2010\DEPLOY
Copied: DS001:\Applications\Healthcare MDT 2010\DEPLOY\HARDWARE
Copied: DS001:\Applications\Healthcare MDT 2010\DEPLOY\HARDWARE\DELL
Copied: DS001:\Applications\Healthcare MDT 2010\DEPLOY\HARDWARE\LENOVO
Copied: DS001:\Applications\Healthcare MDT 2010\DEPLOY\HARDWARE\TOSHIBA
Copied: DS001:\Applications\Healthcare MDT 2010\DEPLOY\HARDWARE\HP
```

5. Click **Finish** when the process has completed.

6. To customise the linked Deployment Share settings (for example, the background used in the Windows PE image), add the linked Deployment Share to the Deployment Workbench by right-clicking on **Deployment Shares** and selecting **Open Deployment Share**:



7. Enter the UNC path of the linked Deployment Share and click **Next**:



Open Deployment Share Wizard

Path

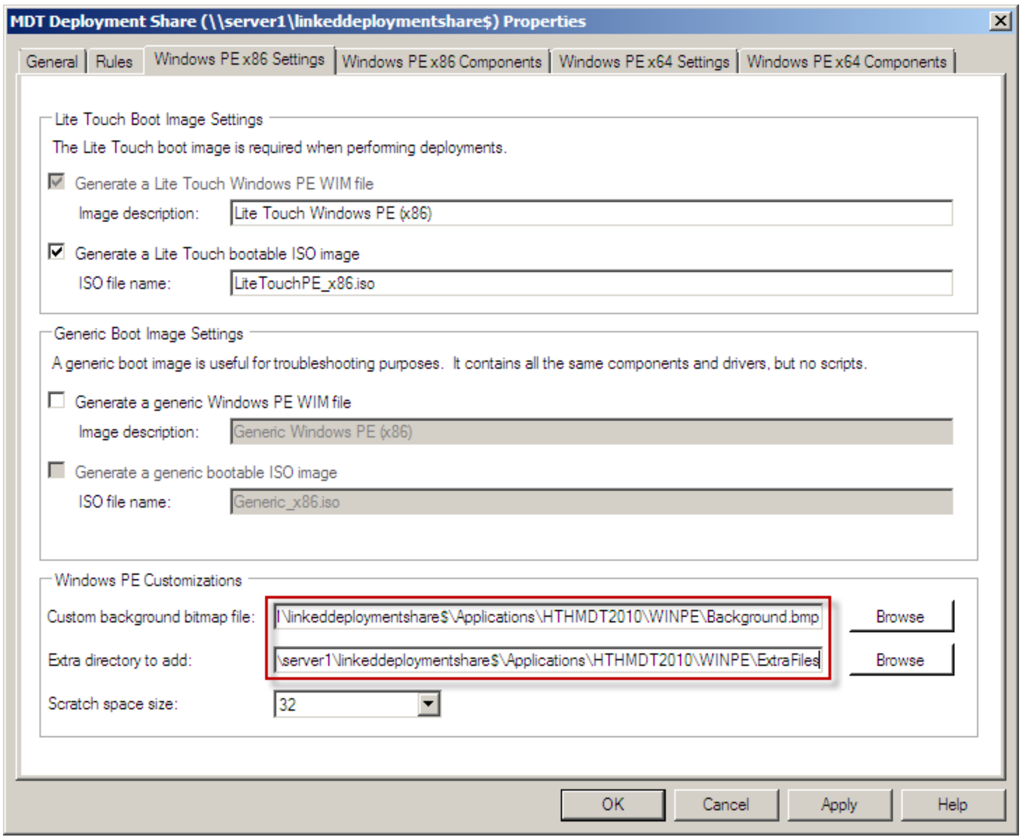
Specify the location of the network share for this deployment type.

Deployment share path: \\SERVER1\LinkedDeploymentShare$

☐ Upgrade the content of the deployment share (if required)

8. Click **Finish**.

9. Right-click the newly opened Deployment Share and select **Properties** to change Windows PE settings (see section 6.4 for additional details):



MDT Deployment Share (\\server1\linkeddeploymentshare$) Properties

Lite Touch Boot Image Settings

The Lite Touch boot image is required when performing deployments.

☑ Generate a Lite Touch Windows PE WIM file

Image description: Lite Touch Windows PE (x86)

☑ Generate a Lite Touch bootable ISO image

ISO file name: LiteTouchPE_x86.iso

Generic Boot Image Settings

A generic boot image is useful for troubleshooting purposes. It contains all the same components and drivers, but no scripts.

☐ Generate a generic Windows PE WIM file

Image description: Generic Windows PE (x86)

☐ Generate a generic bootable ISO image

ISO file name: Generic_x86.iso

Windows PE Customizations

Custom background bitmap file: l\linkeddeploymentshare$\Applications\HTHMDT2010\WINPE\Background.bmp

Extra directory to add: \server1\linkeddeploymentshare$\Applications\HTHMDT2010\WINPE\ExtraFiles

Scratch space size: 32

6.4 Customising the Deployment Share

There are three common customisations that can be made to a Deployment Share:

- Bootstrap.ini
- CustomSettings.ini
- Windows PE Background and Extra Files

All of the above can be altered by right-clicking on the Deployment Share and selecting **Properties** as shown in Figure 20:

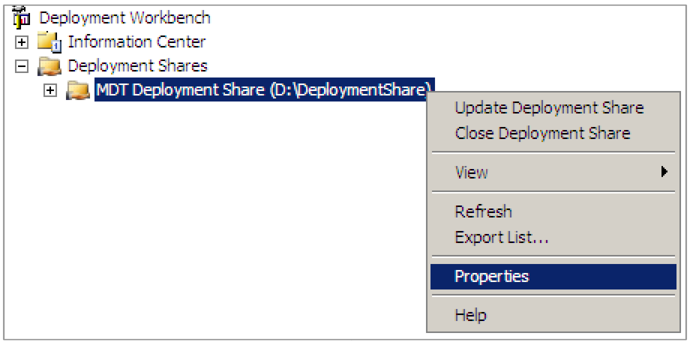
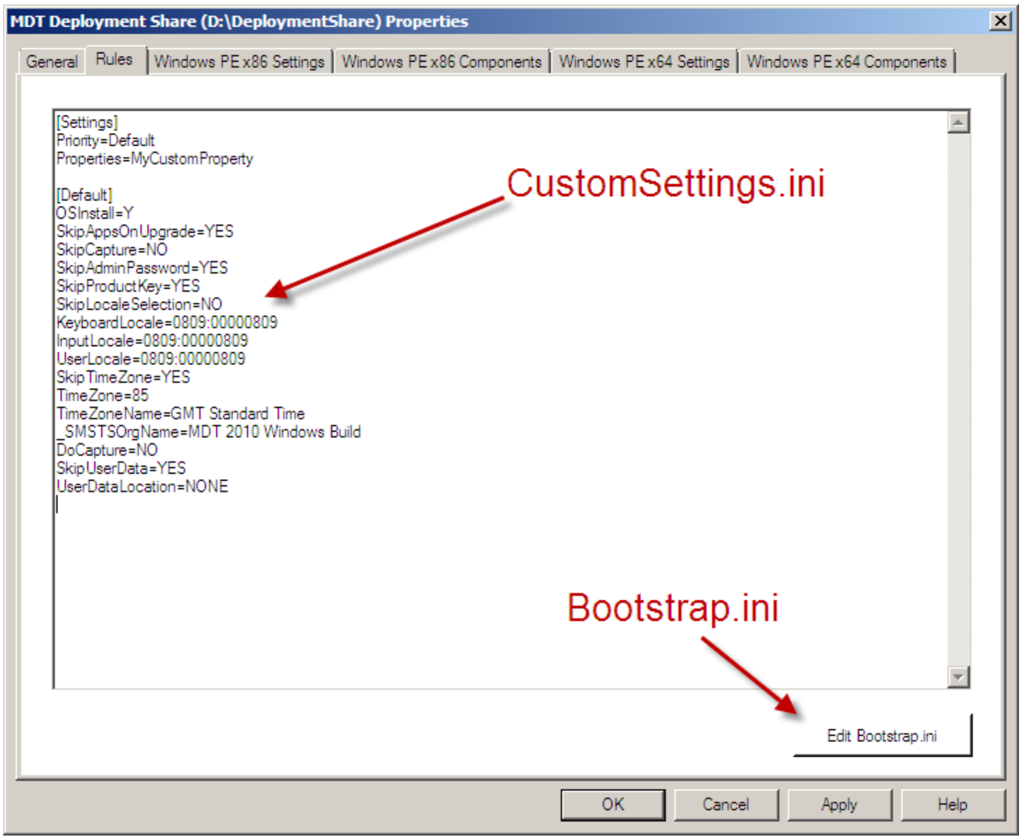


Figure 20: Deployment Share Properties

6.4.1 Bootstrap.ini

The Bootstrap.ini file is saved in the Windows PE image that is used to execute the Build and Deploy processes. The Bootstrap.ini file can be altered from the **Rules** tab within the Deployment Share properties as shown in Figure 21, or edited directly from **<Distribution> \ Control \ Bootstrap.ini**:



```
[Settings]
Priority=Default
Properties=MyCustomProperty

[Default]
OSInstall=Y
SkipAppsOnUpgrade=YES
SkipCapture=NO
SkipAdminPassword=YES
SkipProductKey=YES
SkipLocaleSelection=NO
KeyboardLocale=0809:00000809
InputLocale=0809:00000809
UserLocale=0809:00000809
SkipTimeZone=YES
TimeZone=85
TimeZoneName=GMT Standard Time
_SMSTSOrgName=MDT 2010 Windows Build
DoCapture=NO
SkipUserData=YES
UserDataLocation=NONE
```

Figure 21: Deployment Share Properties Rule Tab

The common customisation settings for Bootstrap.ini are shown in Table 10:

Setting	Values	Description
DeployRoot	\\<Server>\<DeploymentShareName>	The network path of the Deployment Share. This value will usually be set correctly by default.
SkipBDDWelcome	YES | NO	Set this to YES to skip the initial welcome screen prompt. A valid UserID, UserDomain, UserPassword and KeyboardLocalePE setting must also be given when using this option.
UserID	Active Directory User ID	Credentials for a user with permission to access the Deployment Share.
UserDomain	Active Directory Domain	
UserPassword	Active Directory Password	
KeyboardLocalePE	0809:00000809	The language code for the required keyboard locale. 0809 is for a United Kingdom locale.

Table 10: Bootstrap.ini Settings

After making any changes to Bootstrap.ini the Windows PE ISO must be recreated as detailed in section 4.2.1.

6.4.2 CustomSettings.ini

The CustomSettings.ini file is accessed via the network during the Build and Deploy processes. The CustomSettings.ini file can be altered from the **Rules** tab within the Deployment Share properties as shown in Figure 21, or edited directly from it **<Distribution> \ Control \ CustomSettings.ini**.

Common customisation settings for CustomSettings.ini are shown in Table 11. All available settings are listed in the Microsoft Deployment Toolkit Documentation Library which is installed with MDT 2010.

Setting	Values	Description
_SMSTSOrgName	Healthcare Extensions for MDT 2010 Build	The descriptive name shown during the build process.
SkipApplications	YES | NO	Optionally skip the Applications wizard page.
SkipProductKey	YES | NO	Optionally skip the Product Key wizard page.
SkipDomainMembership	YES | NO	Optionally skip the Domain wizard page.
JoinWorkgroup	Workgroup name	The default name of the workgroup to join.
SkipAdminPassword	YES | NO	Optionally skip the Admin Password wizard page.
AdminPassword	Workstation admin password	The admin password that is set on the deployed workstation.
SkipLocaleSelection	YES | NO	Optionally skip the Locale wizard page.
KeyboardLocale	0809:00000809	Keyboard locale.
InputLocale	0809:00000809	Input locale.
UserLocale	en-gb (Vista or later) 0809:00000809 (Windows XP)	User locale. This accepts different values for Windows XP, and Windows Vista or later.
SkipTimeZone	YES | NO	Optionally skip the Time Zone wizard page.
TimeZone	85	The time zone. 85 is London.
TimeZoneName	GMT Standard Time	The time zone name.
SkipCapture	YES | NO	Optionally skip the Image Capture wizard page.

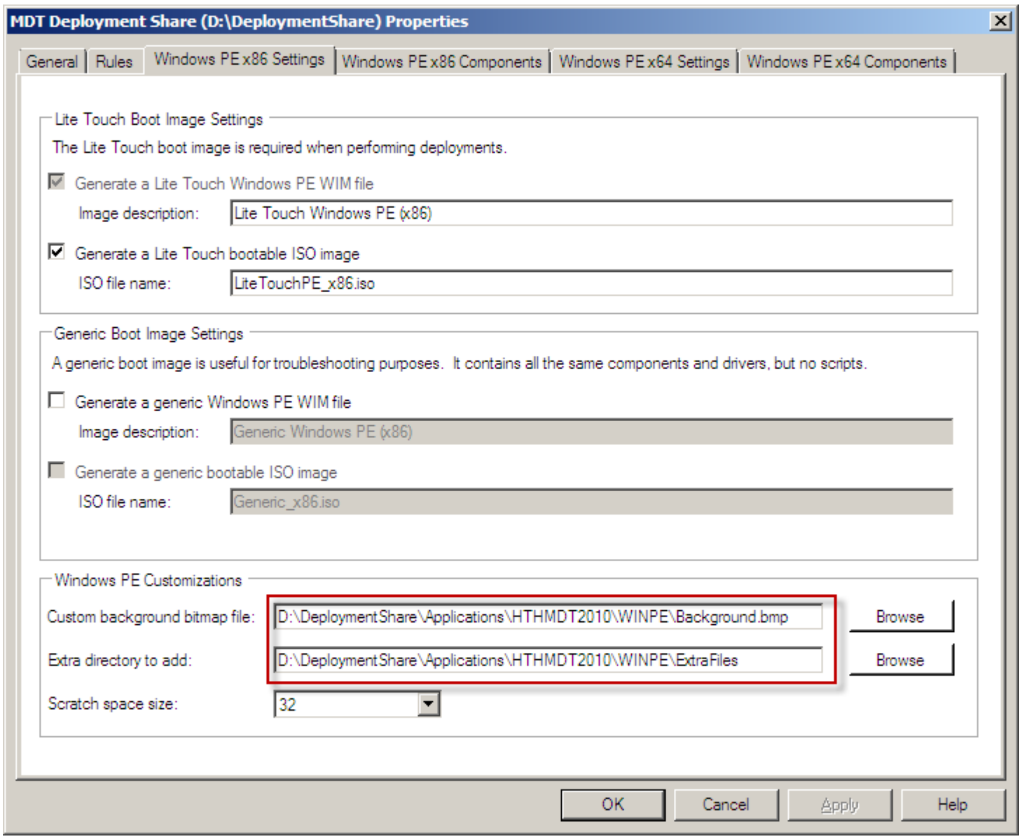
Setting	Values	Description
DoCapture	YES | NO	The default setting on the Image Capture wizard page.
SkipUserData	YES | NO	Optionally skip the User Data Migration wizard page.
UserDataLocation	NONE	The default user data storage location on the User Data Migration wizard page.
SkipBitLocker	YES | NO	Optionally skip the Bitlocker wizard page.

Table 11: CustomSettings.ini Settings

After making any changes to Bootstrap.ini, it is not necessary to recreate the Windows PE ISO as it is accessed dynamically via the network during the Build and Deploy processes.

6.4.3 Windows PE Background and Extra Files

A custom background image, and additional files and folders, can be added to the Windows PE image. Healthcare Extensions for MDT 2010 pre-configures these settings during installation, but they can be changed in the **Windows PE x86/x64 Settings** tabs as shown in Figure 22:


MDT Deployment Share (D:\DeploymentShare) Properties
General
Rules
Windows PE x86 Settings
Windows PE x86 Components
Windows PE x64 Settings
Windows PE x64 Components
OK
Cancel
Apply
Help
Browse
Browse

Lite Touch Boot Image Settings

The Lite Touch boot image is required when performing deployments.

☑ Generate a Lite Touch Windows PE WIM file

Image description: Lite Touch Windows PE (x86)

☑ Generate a Lite Touch bootable ISO image

ISO file name: LiteTouchPE_x86.iso

Generic Boot Image Settings

A generic boot image is useful for troubleshooting purposes. It contains all the same components and drivers, but no scripts.

☐ Generate a generic Windows PE WIM file

Image description: Generic Windows PE (x86)

☐ Generate a generic bootable ISO image

ISO file name: Generic_x86.iso

Windows PE Customizations

Custom background bitmap file: D:\DeploymentShare\Applications\HTHMDT2010\WINPE\Background.bmp

Extra directory to add: D:\DeploymentShare\Applications\HTHMDT2010\WINPE\ExtraFiles

Scratch space size: 32

Figure 22: Windows PE x86 Settings

After making changes to any of the Windows PE settings tabs, the Windows PE ISO must be recreated as detailed in section 4.2.1.

APPENDIX A SKILLS AND TRAINING RESOURCES

The tables in this Appendix provide details of the suggested training and skill assessment resources available. This list is not exhaustive; there are many third-party providers of such skills. The resources listed are those provided by Microsoft.

PART I Application Compatibility

For further information on Windows Application Compatibility see: http://technet.microsoft.com/en-us/desktopdeployment/bb414773.aspx

Skill or Technology Area	Resource Location	Description
Collecting and analysing the application inventory	http://technet.microsoft.com/en-gb/desktopdeployment/bb395313.aspx	Collect and analyse the application inventory to build your application portfolio including suggested processes and technologies that can be used.
Testing your mitigation strategies	http://technet.microsoft.com/en-gb/desktopdeployment/bb395314.aspx	Test mitigation strategies to create application mitigation packages. Provides the tools and methods for successfully testing and resolving any application compatibility issues.
Resolving any outstanding compatibility issues	http://technet.microsoft.com/en-gb/desktopdeployment/bb395315.aspx	Resolve outstanding compatibility issues to report compatibility mitigation. Describes how to use the appropriate tools for automating the testing process.
Deploying compatibility mitigation packages	http://technet.microsoft.com/en-gb/desktopdeployment/bb395316.aspx	Deploy compatibility mitigation packages with core application deployment or after core application deployment. Provides details of the tools and technologies that you can use to deploy the application fixes across the infrastructure.

Table 12: Application Compatibility Training Resources

PART II Imaging

For further information on imaging see: http://technet.microsoft.com/en-gb/desktopdeployment/bb395318.aspx

Skill or Technology Area	Resource Location	Description
Building a company-standard image for deployment	http://technet.microsoft.com/en-gb/desktopdeployment/bb395319.aspx	Plan for building a company-standard image that needs to include the applications that are used, and the features that are important, across the business functions.
Testing the image in a controlled pilot program	http://technet.microsoft.com/en-gb/desktopdeployment/bb395320.aspx	Prepare for deployment by testing to make sure the image meets the requirements for the environment. Provides details about how to plan, initiate, and manage an image pilot testing program.
Using the right tools and utilities to deploy the image	http://technet.microsoft.com/en-gb/desktopdeployment/bb395321.aspx	Decide which tools and technologies will be used to deploy the image. Provides information to help develop processes for deploying the company-standard image.
Managing the image lifecycle	http://technet.microsoft.com/en-gb/desktopdeployment/bb395322.aspx	Aids in understanding changes in the environment that will require the company-standard image to evolve, and will also help build cases and scenarios for updating the company-standard image to meet business demands.

Table 13: Imaging Training Resources

PART III Deployment Process – Lite Touch

For further information on the Lite Touch Deployment Process, see: http://technet.microsoft.com/en-us/library/bb456433.aspx

Skill or Technology Area	Resource Location	Description
Ensuring the appropriate Lite Touch Installation infrastructure exists	http://technet.microsoft.com/en-gb/desktopdeployment/bb395334.aspx	A Lite Touch Installation can be performed from within a network using Windows Deployment Services, or stand-alone using portable media created with the Deployment Workbench, such as DVDs or USB drives.
Preparing the deployment environment	http://technet.microsoft.com/en-gb/desktopdeployment/bb395335.aspx	Ensuring that that the deployment environment is properly prepared to run a Lite Touch Installation before deploying images to target computers.
Preparing the WDS server	http://technet.microsoft.com/en-gb/desktopdeployment/bb395336.aspx	How the image-installation process works through Windows Deployment Services when deploying to computers that Systems Management Server 2003 does not manage.
Deploying the OS images to the client computers	http://technet.microsoft.com/en-gb/desktopdeployment/bb395337.aspx	Understanding each deployment scenario (Upgrade Computer, Replace Computer, New Computer, Refresh Computer), and how they use different processes.

Table 14: Deployment Process Training Resources

PART IV 2007 Office System Deployment

For further information on 2007 Microsoft Office System Deployment, see: http://technet.microsoft.com/en-us/library/bb490141.aspx

Skill or Technology Area	Resource Location	Description
Choose and define your 2007 Office system editions	http://technet.microsoft.com/en-gb/desktopdeployment/bb395348.aspx	Clear guidance for the choice of 2007 Office system versions and deployment options.
Identify upgrade issues	http://technet.microsoft.com/en-gb/desktopdeployment/bb395349.aspx	Guidance on the primary issues faced when upgrading to the 2007 Office System, including feature installation states, settings migration, file conversion issues, and file co-existence with multiple versions of Microsoft Office.
Customize the 2007 Office system	http://technet.microsoft.com/en-gb/desktopdeployment/bb395351.aspx	Creating a customised 2007 Office system installation is a multi-step process. After creating and testing the deployment package, it can be inserted into the MDT 2010 imaging process so that the customised 2007 Office system configuration can be included automatically in all computer images.

Table 15: 2007 Office System Deployment Training Resources

PART V Supplemental Training Resources

Title	Link
Deployment TechCenter	http://technet.microsoft.com/en-us/desktopdeployment/default.aspx
The TechNet Script Center	http://www.microsoft.com/technet/scriptcenter/default.mspx
Script Center Script Repository	http://www.microsoft.com/technet/scriptcenter/scripts/default.mspx

Table 16: Supplemental Training Resources

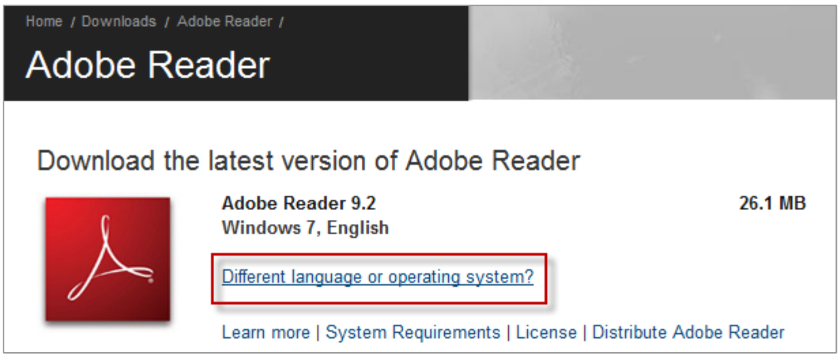
APPENDIX B ADOBE READER INSTALLATION GUIDE

This section details how to download, configure and add Adobe Reader to Healthcare Extensions for MDT 2010.

> **Note**
>
> The third-party applications shown in this Appendix are presented as examples of common applications in use by healthcare organisations. Any such references should not be considered an endorsement, and support for these products is not provided by Microsoft.

Downloading the Setup Files:

1. Go to the Adobe Reader download page[6] on the Adobe Web site.
2. Click **Different language or operating system?**



Home / Downloads / Adobe Reader /

Adobe Reader

Download the latest version of Adobe Reader

Adobe Reader 9.2 26.1 MB
Windows 7, English

Different language or operating system?

Learn more | System Requirements | License | Distribute Adobe Reader

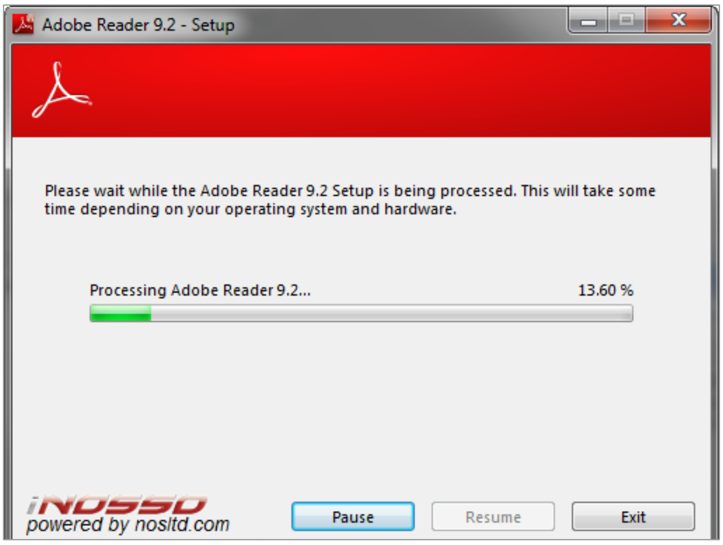
3. Specify the operating system, version and language required, and click **Continue**.
4. Follow the prompts on the next pages. A file named similar to AdbeRdr920_en_US.exe will be downloaded.

Extracting the Setup Files

In order to customise the setup of Adobe Reader, it is necessary to obtain the underlying .msi files.

1. Run the file downloaded in previous steps (for example, AdbeRdr920_en_US.exe). The installer will extract temporary files:

[6] Adobe Reader download page **{R5}:**
http://get.adobe.com/uk/reader/



Adobe Reader 9.2 - Setup

Please wait while the Adobe Reader 9.2 Setup is being processed. This will take some time depending on your operating system and hardware.

Processing Adobe Reader 9.2... 13.60 %

2. When the extraction completes, a prompt for selecting the destination folder appears. **Do not click Next**.

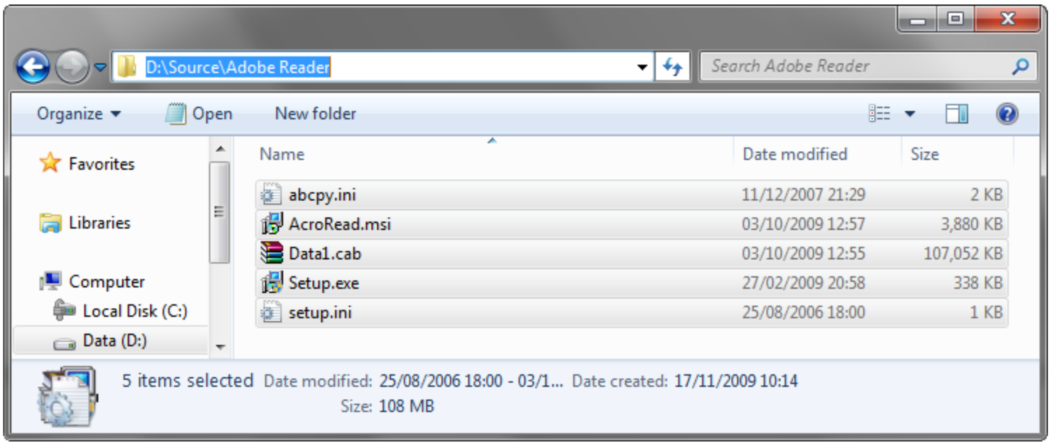
   Depending on the machine and operating system in use, the installer setup files are copied to one of the following temporary locations:

   - C:\Program Files\Adobe\Reader 9.0\Setup Files\
   - C:\Users\<user id>\AppData\Local\Temp\Adobe Reader
   - C:\Documents and Settings\Administrator\Local Settings\Temp\Adobe Reader
   - C:\Documents and Settings\Administrator\Local Settings\Temp\1\Adobe Reader

   The setup files consist of the following:

   - Abcpy.ini
   - AcroRead.msi
   - Data1.cab
   - Setup.exe
   - Setup.ini

3. Copy all of the setup files from the temporary location to the application source folder (in this example, the folder used is **D:\Source\Adobe Reader**):



D:\Source\Adobe Reader

Name	Date modified	Size
abcpy.ini	11/12/2007 21:29	2 KB
AcroRead.msi	03/10/2009 12:57	3,880 KB
Data1.cab	03/10/2009 12:55	107,052 KB
Setup.exe	27/02/2009 20:58	338 KB
setup.ini	25/08/2006 18:00	1 KB

5 items selected Date modified: 25/08/2006 18:00 - 03/1... Date created: 17/11/2009 10:14
Size: 108 MB

4. **Cancel** the Adobe Reader installation.

Customization Wizard

In order to customise an Adobe Reader installation, it is necessary to download the Adobe Customization Wizard. The current version can be downloaded from the *Adobe Customization Wizard 9* download page[7]:

After downloading the wizard, the setup customisation can be performed.

To perform the setup customisations:

1. Launch Adobe Customization Wizard 9 from the **Start** menu:
2. From the **File** menu, select **Open Package**.
3. Select the Adobe .msi file. In this example, the .msi file is located at D:\Source\Adobe Reader\AcroRead.msi.
4. Use the interface to configure the required customisations. The options chosen are unique to each healthcare organisation, but some basic recommendations are given in Table 17:

Installation Tab/Option	Configuration State
Installation Options	
Enable Optimization	Clear
Enable caching of installer files	Clear
Run Installation	Clear
If reboot required at the end of installation	Suppress reboot
End User License Agreement and Online Features	
Suppress display of End User License Agreement (EULA)	Selected
Online Features – Disable all updates	Selected
Disable Help > Purchase Adobe Acrobat	Selected
Toolbars and Document Status	
Disable Start Meeting	Selected
Do not show Beyond Reader at startup	Selected
Disable Adobe Online Services	Selected

Table 17: Adobe Reader Recommended Customisations

5. From the **File** menu, select **Save Package** and close the Adobe Customization Wizard. The following files are added to the D:\Source\Adobe Reader folder: AcroRead.mst, Setup.exe and Setup.ini.

When the Adobe Reader setup files have been customised, executing the newly-created Setup.exe file will result in a silent and automated installation. To add Adobe Reader as an application in the Deployment Workbench, follow the instructions in section 4.5. The silent installation command that should be used is ’setup.exe’.

[7] Adobe Customization Wizard 9 **{R6 }**: http://www.adobe.com/support/downloads/detail.jsp?ftpID=3993

APPENDIX C DOCUMENT INFORMATION

PART I Terms and Abbreviations

Abbreviation	Definition
ACT	Application Compatibility Toolkit
ASP	Active Server Pages
BDD	Business Desktop Deployment
DC	Domain Controller
DHCP	Dynamic Host Control Protocol
DNS	Domain Name System
EULA	End User License Agreement
FTP	File Transfer Protocol
HAL	Hardware Abstraction Layer
HTTP	Hypertext Transfer Protocol
ISO	International Standards Organisation
KMS	Key Management Server
LTI	Lite Touch Installation
MAK	Multiple Activation Key
MDT	Microsoft Deployment Toolkit
MMC	Microsoft Management Console
MSI	Microsoft Installer
MSXML	Microsoft Extensible Mark-up Language
OS	Operating System
OU	Organisational Unit
SMS	Systems Management Server 2003
SP	Service Pack
TPM	Trusted Platform Module
USB	Universal Serial Bus
USMT	User State Migration Tool
WAIK	Windows Automated Installation Kit
WDS	Windows Deployment Services
WIM	Windows Image
WinPE	Windows Pre-Execution Environment
WMI	Windows Management Instrumentation
WSUS	Windows Server Update Services
WU	Windows Update
WVHA	Windows Vista Hardware Assessment

Abbreviation	Definition
XML	Extensible Mark-up Language
ZTI	Zero Touch Installation

Table 18: Terms and Abbreviations

PART II References

Reference	Document	Version
R1.	Automated Build Application Integration http://www.microsoft.com/industry/healthcare/technology/hpo/desktop/desktop.aspx	3.0.0.0
R2.	Microsoft Help and Support: Description of the BitLocker Drive Preparation Tool http://support.microsoft.com/kb/930063	
R3.	Microsoft TechNet: BitLocker Drive Encryption Configuration Guide: Backing Up BitLocker and TPM Recovery Information to Active Directory http://technet2.microsoft.com/WindowsVista/en/library/3dbad515-5a32-4330-ad6f-d1fb6dfcdd411033.mspx?mfr=true	
R4.	Microsoft Help and Support: Windows 7, Windows Server 2008 R2, and Windows Vista setup log file locations http://support.microsoft.com/kb/927521	
R5.	Adobe: Download the latest version of Adobe Reader http://get.adobe.com/uk/reader/	
R6.	Adobe: Adobe Customization Wizard 9 http://www.adobe.com/support/downloads/detail.jsp?ftpID=3993	

Table 19: References