

# Windows Deployment Services Overview

Article 08/31/2016

Applies To: Windows Server 2012 R2, Windows Server 2012

Windows Deployment Services (WDS) enables you to deploy Windows operating systems over the network, which means that you do not have to install each operating system directly from a CD or DVD.

For information about what is new or included in your version of Windows Deployment Services, see [What's New in Windows Deployment Services in Windows Server 2012](#).

For step-by-step guidance on how to use the Windows Deployment Services, see [Windows Deployment Services Getting Started Guide for Windows Server 2012](#).



## Benefits of Windows Deployment Services

1. Allows network-based installation of Windows operating systems, which reduces the complexity and cost when compared to manual installations.
2. Supports deploying images for mixed environments including Windows 7 and Windows Server 2008 R2 through Windows 8.1 and Windows Server 2012 R2.
3. Uses standard Windows Setup technologies including Windows Preinstallation Environment (Windows PE), .wim files, and image-based setup.
4. Transmits data and images by using multicast functionality.
5. Allows you to create images of a reference computer using the Image Capture Wizard, which is an alternative to the ImageX tool.
6. Allows you to add driver packages to the server and configure them to be deployed to client computers along with the install image.

## Prerequisites for installing Windows Deployment Services

The following are requirements for installing this role, depending on whether you choose the default installation (both Deployment Server and Transport Server), or only the Transport Server role service.

 Expand table

Deployment Server and Transport Server	Transport Server
<ul style="list-style-type: none"> <li>• <b>Active Directory Domain Services (AD DS).</b> A Windows Deployment Services server must be a member of an Active Directory Domain Services (AD DS) domain or a domain controller for an AD DS domain.</li> </ul> <p>The AD DS domain and forest versions are irrelevant; all domain and forest configurations support Windows Deployment Services.</p> <p> Expand table</p> <div data-bbox="297 1087 691 1325"> <p> <b>Note</b></p> <p>AD DS is not required if the WDS server is configured in Standalone mode.</p> </div> <ul style="list-style-type: none"> <li>• <b>DHCP.</b> You must have a working Dynamic Host Configuration Protocol (DHCP) server with an active scope on the network because Windows Deployment Services uses PXE, which relies on DHCP for IP addressing.</li> <li>• <b>DNS.</b> You must have a working Domain Name System (DNS) server on the network before you can run Windows Deployment Services.</li> <li>• <b>NTFS volume.</b> The server running Windows Deployment Services requires an NTFS file system volume for the image store.</li> </ul>	<ul style="list-style-type: none"> <li>• For Windows Server 2008: The only prerequisite is that you must be a member of the Local Administrators group on the server to install Transport Server. Also note that a PXE provider is not installed with Transport Server, so you must create a custom PXE provider to network boot.</li> <li>• For Windows Server 2008 R2: You must be a member of the Local Administrators group to install Transport Server. In addition, if you are using Transport Server to network boot, your environment must contain DHCP (Windows Server 2008 R2 contains a PXE provider, which allows you to network boot).</li> <li>• For Windows Server 2012 and Windows Server® 2012 R2, there are no additional prerequisites.</li> </ul>

**Deployment Server and Transport Server    Transport Server**

- **Credentials.** To install the role, you must be a member of the Local Administrators group on the server.

To initialize the server, you must be a member of the Domain Admins group.

**ⓘ Note**

To initialize the WDS server in Standalone mode, you need not be a member of the Domain Users group.

For more information, see [Required Permissions](#) .

## Tools for managing Windows Deployment Services

The following are tools that you can use to manage your server:

1. **Windows Deployment Services MMC snap-in** - Provides an easy way to manage images, computers, and common server settings. You can perform almost all tasks from the MMC snap-in (for example, you cannot prestage client computers). Note that the snap-in is not available when you install only the Transport Server role service.
2. **WDSUTIL command-line tool** - enables you to manage the full functionality of the server. WDSUTIL also enables you to script common tasks.
3. **Windows PowerShell cmdlets for WDS** – enables you to manage and monitor the WDS server using Windows PowerShell scripts.

# Deploying and configuring WDS in a multi-computer environment

It is not common to split a WDS server's functionality among different computers.

## Installing the WDS role on virtual machines

You can run Windows Deployment Services on Hyper-V virtual machines. Depending on the capacity of your server running Hyper-V, you can install, configure and use Windows Deployment Services. You can also create a virtual machine to act as a client machine when building and testing your deployment images. This is especially useful for building your custom images. Note, however, that the performance will often be degraded, particularly during the Trivial File Transfer Protocol (TFTP) download phase. This phase is very resource-intensive and may fail if insufficient resources are available on your server running Hyper-V.

### Warning

WDS cannot be run on a virtual machine in Windows Azure.

## Support for the WDS role in a clustered environment

The Windows Deployment Services server service is not cluster-aware. However, you can run multiple WDS servers in a network to provide fault-tolerance and load balancing.

## Considerations for managing this role remotely

You can manage Windows Deployment Services remotely by installing the Windows Deployment Services server role on a local server and then using that server to connect to the remote server in the data center or you can use the Windows PowerShell cmdlets for Windows Deployment Services to manage a remote server.

### Note

Windows Deployment Service cannot be managed using the Remote Server Administration Tools.

## Managing the WDS role using the Server Core installation option

You cannot use Windows Deployment Services with the Server Core installation option.

## Configuring the WDS role for high availability

WDS is designed to automatically fail over to other WDS servers. We recommend that you deploy two WDS servers at a minimum and then additional servers to increase redundancy for higher availability.

### Note

Installing images from WDS over a wide area network is not recommended.

## Known issues

When using the Dynamic Driver Provisioning (DDP) feature on a WDS server running Windows Server 2012 if you attempt to add a driver to a Windows 8.1 or Windows Server 2012 R2 boot.wim you may receive the following error when running the "Add driver packages to image" wizard:

**An error occurred while trying to execute command. Error Code: 0xc0000135**

This is a known issue related to DISM version included with Windows Server 2012 that is used by WDS to service images. This version of DISM supports servicing Windows Server 2012 and earlier boot images but does not support later versions of boot images. You can resolve this issue by upgrading your WDS server to Windows Server 2012 R2 which contains a DISM version that supports servicing Windows 8.1 and Windows Server 2012 R2 boot.wim's or you can install the Windows 8.1 ADK and manually add the driver to the boot.wim. For instructions for how to add drivers, see [Add and Remove Drivers Offline Using DISM](#)

# Common usage scenarios

The following are common scenarios for Windows Deployment Services.

## Scenario one: The small business

Fabrikam, Inc. is a manufacturer of towels with custom designs. It is a small business with a single office. Monica Brink, Fabrikam's resident IT professional, is responsible for maintaining the IT infrastructure for the company, which consists of 25 client computers running Windows XP SP2 Professional and a single server running Windows Server 2003 with SP2. The server functions as a file print server, Web server, Exchange server, Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP) server, and domain controller. The computers are linked by a 100-MBps Ethernet connection.

Monica is given the task of moving all of the client computers to the Windows 7 (or Windows 8.1) operating system and upgrading the single server to Windows Server 2008 R2, Windows Server 2012, or Windows Server 2012 R2. It is important that she makes the switch as quickly as possible so that office workers can continue their work.

In the past, she deployed a new operating system one computer at a time. This took her around 45 minutes per computer (almost 19 hours to set up the operating system on all the client computers). For almost three days, Monica was unavailable to work on anything else. Then she would spend almost as much time installing the applications on each computer.

Monica is the only IT professional at Fabrikam, which means that she also must help teach users about the new operating system. Therefore, it is important that she minimizes the amount of time she spends on deployment. To accomplish this, Monica chooses to use Windows Deployment Services because she can:

- Save time by running several installations simultaneously.
- Use a custom install image with preinstalled applications.
- Create an image by using the Windows Deployment Services Image Capture Wizard.

To begin, Monica does the following:

1. Upgrades her server to Windows Server 2008 R2, or Windows Server 2012 or Windows Server 2012 R2.

2. Installs the Windows Deployment Services server role.
3. Adds the Boot.wim from the product DVD (which contains a Windows PE image, Setup.exe, and supporting files) by using the Windows Deployment Services MMC snap-in.
4. Adds the Install.wim from the product DVD to the Windows Deployment Services server by using the MMC snap-in.
5. Uses the MMC snap-in to create a capture image from the boot image she added in step 3. This image contains Windows PE and a wizard that will capture her custom image into a .wim file.

All users at Fabrikam have the same desktop hardware, which was purchased from a single vendor. To deploy a standard image to the computers, Monica does the following:

1. Boots a reference computer from the network and installs the Install.wim onto it, which contains the standard version of Windows.
2. Installs Microsoft Office, the company's towel-design application, and the latest drivers from the manufacturer's site.
3. Uses Sysprep to generalize the operating system.
4. Reboots the computer into the capture image.
5. Uses the Image Capture Wizard to recapture the operating system and upload it directly to the Windows Deployment Services server.

Now, Monica is ready to install the new operating systems. She does not need to migrate any user data, because all of the employees store their user data on a server (rather than on their hard disks). She reboots a client computer and then presses F12 to perform a network boot. This boots her into the Boot.wim file, which guides her through the installation process. She selects the disk partition and image she wants, and then the installation begins. While waiting for the image to be applied to the first computer, Monica boots another computer and starts the same process on that one.

## Scenario two: The medium-sized business

Northwind Traders is a shipping firm with three offices: a central office in Tooth City, and branch offices in the towns of Brushville and Flosston. Ron Gable is one of six IT staff

members at Northwind Traders. His responsibility is maintaining the 250 client computers used by the company's employees. These are mostly desktop computers, but the sales force uses laptops for customer presentations. There are 200 computers in the central office in Tooth City, and 25 each in the Brushville and Flosston offices. Each site has an internal network running at 100 MB per second (MBps), and the branch sites are connected to the Tooth City office by a T1 line. Ron has three Windows Deployment Services servers at the Tooth City office and one in each of the branch offices, which are administered remotely.

Ron's supervisor has tasked him with deploying Windows 7 (or Windows 8.1) to the whole company. Previously, this would have involved many expensive trips to Brushville and Flosston, and it would have taken Ron several weeks to complete. He wants to use Windows Deployment Services to deploy Windows remotely; however, company policy dictates that there can be only one DHCP server on the corporate network, and this server is located at the Tooth City office. Remotely deploying images to the 50 computers at the branch offices would cause immense congestion on the connection.

Ron chooses to use Windows Deployment Services because with unattended setup, he can:

- Deploy Windows to computers at the branch sites without being physically present there.
- Use his existing replication solution to deliver images to the branch site servers.
- Use the network boot referral system to minimize network traffic between the branch sites and the central office.

Ron prestages each client computer and assigns the appropriate branch office server as the referral server for each. This ensures that the client downloads files and images from the local server, which minimizes traffic on the line between the offices.

Ron has two standard operating system configurations — one for the desktop computers and one for laptops that contains the sales presentations and drivers for projectors. Therefore, he builds two images: one with the desktop configuration, and one with the laptop configuration (with no applications). He stores all the user data on one of the servers, so he can deploy Windows without preserving any existing data on the client computers.

Ron uses Windows System Image Manager (Windows SIM) to author two image unattend files — one for the desktop computers and one for the laptops. These files automate the installation, so Ron does not need to be present at each computer during the installation.



They also automatically install Microsoft Office and the line-of-business application that the company uses for package tracking. He uses the Windows Deployment Services management tools to associate the unattend files with the images.

Next, Ron configures Windows Deployment Services so when a computer is restarted, it will boot from the network automatically and deploy the appropriate image (without requiring the users to press F12). After the image is applied to each computer, the computer is automatically joined to the corporate domain and restarted. He also configures the deployment so that when the computer restarts, the computer will not network boot unless F12 is pressed. This way, the computer then boots to the hard disk drive and finishes the installation process. This prevents a boot loop, in which the computer would continue booting into Setup. When the installation is completed, the computer is ready for the user to log on.

## Scenario three: The large enterprise

Shu Ito is the network architect for Wide World Importers, a large enterprise with 5,000 employees in offices all over the world. The major employee centers are in the United States and Germany, and there are 13 branch offices in other countries. Shu has five servers available to him in the U.S. hub, two in the German hub, and one in each of the branch offices. The servers at the hubs are connected to the corporate Ethernet on 1 Gbps network interface cards (NICs); the other computers are on 100-Mbps NICs. The hubs are connected by T3 lines, and the other sites are connected by T1 lines. All of the servers are hired on two-year leases.

Wide World Importers is replacing the accounting department's 200 computers with computers running Windows 7 (or Windows 8.1). Shu would also like to deploy a Windows Server 2008 R2, Windows Server 2012 or Windows Server 2012 R2 image to any newly leased servers in the U.S. office. The servers in the German office and the branch sites are the responsibility of the local administrators. Currently, deployments at Wide World Importers are done by using RIS, and Shu wants to ensure that the existing computer building processes are preserved with the move to Windows Deployment Services. In addition, it is important that each computer is deployed with an operating system in a language that is appropriate for the users in that country or region.

Shu chooses to use Windows Deployment Services because it enables him to do the following:

- Use appropriate language packs to reduce the required number of images.

- Manage all of his Windows Deployment Services servers from a single computer.
- Use multicast deployments to preserve bandwidth while deploying images to many computers concurrently.
- Write scripts to automate common management tasks.

Shu upgrades his servers to Windows Server 2008 R2, Windows Server 2012 or Windows Server 2012 and creates his images. The vast majority of his deployments will be in English or German, so he creates Windows 7 (or Windows 8.1) image in each language. Other languages will be installed by using external language packs, and applications will be downloaded by using Systems Management Server (SMS). After creating the images, Shu uploads the images and language packs to the Windows Deployment Services server.

Shu authors unattend files with Windows SIM. He then uses File Replication Service (FRS) to copy the images, language packs, and unattend files to the Windows Deployment Services servers around the world. Of the accounting computers used by Wide World Importers, 150 are in the U.S. office, 30 are in the German office, and the remaining 20 are scattered around the world. Shu uses multicasting to deploy to the 150 computers in the U.S. office simultaneously. To do this, he creates a multicast transmission for the relevant image on his Windows Deployment Services server.

To preserve the state and data on the previous computers, Shu uses the User State Migration Tool (USMT) to save all of the data and user configurations to a shared folder on the primary Windows Deployment Services server. Then he sets up each computer to boot from its local Windows Deployment Services server and to start automated setup by using the unattend files. The computers in the U.S. office will automatically join the multicast transmission, while the computers in the other offices will deploy using unicasting. When the installation is completed, Shu runs a task with USMT to migrate the user data to each computer.

When the lease on a server expires and the server is replaced, Shu can use Windows Deployment Services to deploy his Windows Server images in the same way that he performed the RIS deployment.

## Scenario Four: A Custom Deployment Using Transport Server

John Woods is the server maintenance engineer at the A. Datum Corporation data center. He is responsible for maintaining the 300 servers used by A. Datum Corporation's major customers. One of these customers is Adventure Works.

Adventure Works uses 40 servers to run a career Web site (which is backed by a database) for circus performers. After the release of a popular film about circus life, Adventure Works expects an increase in the use of their Web site. They order 10 additional servers to handle the anticipated traffic.

John wants to deploy operating systems to these servers by using Windows Deployment Services. He does not have AD DS running in this environment; he wants to store the configuration information for his computers in a SQL Server database. As a result, he cannot use the standard Windows Deployment Services solution. In addition, he wants to partition the disks in a standard configuration and also copy data (some for database servers, some for Web servers) before the unattended setup begins. John chooses to use Windows Deployment Services because he can:

- Write a plug-in that reads configuration data for the computers from a data store other than AD DS (the data store is typically a database or a flat file).
- Write scripts (to run in Windows PE) that perform preinstallation tasks and then call Setup to install the operating system.

John creates 10 computer accounts in his database for his 10 new servers, and he populates them with the required information. He installs the Windows Deployment Services server role on his server (choosing to install only the Transport Server role service). He then writes a PXE provider (a plug-in that reads information from the database and passes it to Windows Deployment Services) and registers it with the server. He creates a custom boot image that contains Windows PE along with startup scripts to partition the disks and copy the data. Then he uses ImageX to capture one of his existing servers as an install image.

After performing these initial tasks, John connects his servers to the network and boots them. They boot into Windows PE by using the configuration stored in the database. His scripts run to prepare each computer for deployment, and the scripts end by running ImageX to apply the operating system image on each computer.

## Additional references

For information about getting started with Windows Deployment Services, see [Windows Deployment Services Getting Started Guide for Windows Server 2012](#).

For more information about Windows PowerShell cmdlets for WDS, see [Windows Deployment Services Cmdlets in Windows PowerShell](#) .