

Infrastructure Planning and Design

Windows® Deployment Services

Version 2.1

Published: February 2008

Updated: July 2010

For the latest information, please see [www.microsoft.com/IPD](http://www.microsoft.com/IPD)

Copyright © 2010 Microsoft Corporation. This documentation is licensed to you under the Creative Commons Attribution License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/3.0/us/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA. When using this documentation, provide the following attribution: Infrastructure Planning and Design is provided with permission from Microsoft Corporation.

This documentation is provided to you for informational purposes only, and is provided to you entirely "AS IS". Your use of the documentation cannot be understood as substituting for customized service and information that might be developed by Microsoft Corporation for a particular user based upon that user’s particular environment. To the extent permitted by law, MICROSOFT MAKES NO WARRANTY OF ANY KIND, DISCLAIMS ALL EXPRESS, IMPLIED AND STATUTORY WARRANTIES, AND ASSUMES NO LIABILITY TO YOU FOR ANY DAMAGES OF ANY TYPE IN CONNECTION WITH THESE MATERIALS OR ANY INTELLECTUAL PROPERTY IN THEM.

Microsoft may have patents, patent applications, trademarks, or other intellectual property rights covering subject matter within this documentation. Except as provided in a separate agreement from Microsoft, your use of this document does not give you any license to these patents, trademarks or other intellectual property.

Information in this document, including URL and other Internet website references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, email addresses, logos, people, places and events depicted herein are fictitious.

Microsoft, Active Directory, Hyper-V, SQL Server, Windows, Windows Server, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries and regions.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

You have no obligation to give Microsoft any suggestions, comments or other feedback (“Feedback”) relating to the documentation. However, if you do provide any Feedback to Microsoft then you provide to Microsoft, without charge, the right to use, share and commercialize your Feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software or service that includes the Feedback. You will not give Feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your Feedback in them.

Contents

[The Planning and Design Series Approach 1](#_Toc242174062)

[Introduction to Windows Deployment Services Planning and Design 2](#_Toc242174063)

[Windows Deployment Services in Microsoft Infrastructure Optimization 3](#_Toc242174064)

[Windows Deployment Services Design Process 4](#_Toc242174065)

[Step 1: Determine the Number of Windows Deployment Services Instances Required 7](#_Toc242174066)

[Step 2: Is There an Existing Windows Deployment Services or RIS Infrastructure? 9](#_Toc242174067)

[Step 3: Select Between Full Windows Deployment Services or Transport Server Role 12](#_Toc242174068)

[Step 4: Determine the Server Resource Requirements 14](#_Toc242174069)

[Step 5: Determine the File Share Fault-Tolerance and Consistency Mechanism 23](#_Toc242174070)

[Step 6: Determine the Client Windows Deployment Services Discovery Method 29](#_Toc242174071)

[Dependencies 32](#_Toc242174072)

[Conclusion 32](#_Toc242174073)

[Appendix A: Job Aids 35](#_Toc242174074)

[Appendix B: Server Performance Analyzing and Scaling 37](#_Toc242174075)

[Version History 48](#_Toc242174076)

[Acknowledgments 49](#_Toc242174077)

# The Planning and Design Series Approach

This guide is one in a series of planning and design guides that clarify and streamline the planning and design process for Microsoft® infrastructure technologies.

Each guide in the series addresses a unique infrastructure technology or scenario. These guides include the following topics:

* Defining the technical decision flow (flow chart) through the planning process.
* Describing the decisions to be made and the commonly available options to consider in making the decisions.
* Relating the decisions and options to the business in terms of cost, complexity, and other characteristics.
* Framing the decision in terms of additional questions to the business to ensure a comprehensive understanding of the appropriate business landscape.

The guides in this series are intended to complement and augment the product documentation.

## Benefits of Using This Guide

Using this guide will help an organization to plan the best architecture for the business and to deliver the most cost-effective Windows® Deployment Services technology.

Benefits for Business Stakeholders/Decision Makers:

* Most cost-effective design solution for an implementation. Infrastructure Planning and Design (IPD) eliminates over-architecting and overspending by precisely matching the technology solution to the business needs.
* Alignment between the business and IT from the beginning of the design process to the end.

Benefits for Infrastructure Stakeholders/Decision Makers:

* Authoritative guidance. Microsoft is the best source for guidance about the design of Microsoft products.
* Business validation questions to ensure the solution meets the requirements of both business and infrastructure stakeholders.
* High integrity design criteria that includes product limitations.
* Fault-tolerant infrastructure, where necessary.
* Proportionate system and network availability to meet business requirements. Infrastructure that is sized appropriately to meet business requirements.

**Benefits for** Consultants or Partners:

* Rapid readiness for consulting engagements.
* Planning and design template to standardize design and peer reviews.
* A “leave-behind” for pre- and post-sales visits to customer sites.
* General classroom instruction/preparation.

Benefits for the Entire Organization:

Using this guide should result in a design that will be sized, configured, and appropriately placed to deliver a solution for achieving stated business requirements, while considering the performance, capacity, manageability, and fault tolerance of the system.

# Introduction to Windows Deployment Services Planning and Design

Windows Deployment Services is the updated and redesigned version of Remote Installation Services (RIS). Windows Deployment Services assists with the rapid adoption and deployment of Windows operating systems. It can be used to set up new computers through a network-based installation without the IT professional having to be physically present at each computer and without having to install directly from CD or DVD media.

The purpose of this guide is to present the infrastructure planning process for Windows Deployment Services by providing a clear and concise workflow of the decisions and tasks required.

This guide, when used in conjunction with product documentation, enables companies to confidently plan the Windows Deployment Services infrastructure.

## What’s New in Windows Server 2008 R2

This guide’s design process is valid for both Windows Server® 2008 and Windows Server 2008 R2 environments.

Windows Server 2008 R2 introduces new functionality and enhancements to Windows Deployment Services, including the following:

* Dynamic driver provisioning
* Virtual hard disk (VHD) deployment
* Additional multicasting functionality
* Pre-Boot Execution Environment (PXE) provider for Transport Server
* Additional Extensible Firmware Interface (EFI) functionality

For more details on the changes, see [http://technet.microsoft.com/en-us/library/dd391932(WS.10).aspx](http://technet.microsoft.com/en-us/library/dd391932%28WS.10%29.aspx).

## Assumptions

This guide makes the following assumptions so as to better focus the scope of the material presented:

* This design is for use in a production environment. It is expected that a test environment will also be created to mirror the production environment configuration.
* The reader is familiar with common desktop deployment technologies and networking components including Windows Deployment Services, Dynamic Host Configuration Protocol (DHCP), Domain Name System (DNS), and Active Directory® Domain Services (AD DS). This guide does not attempt to educate the reader on the features and capabilities of these or other Microsoft products.

## Feedback

Please direct questions and comments about this guide to satfdbk@microsoft.com.

We value your feedback on the usefulness of this guide. Please complete the following **Solution Accelerators Satisfaction Survey**, available at <http://go.microsoft.com/fwlink/?LinkID=132579>, and help us build better guidance and tools.

# Windows Deployment Services in Microsoft Infrastructure Optimization

The Infrastructure Optimization (IO) Model at Microsoft groups IT processes and technologies across a continuum of organizational maturity. (For more information, see <http://www.microsoft.com/infrastructure>.) The model was developed by industry analysts, the Massachusetts Institute of Technology (MIT) Center for Information Systems Research (CISR), and Microsoft's own experiences with its enterprise customers. A key goal for Microsoft in creating the IO Model was to develop a simple way to use a maturity framework that is flexible and that can easily be applied as the benchmark for technical capability and business value.

IO is structured around three information technology models: Core Infrastructure Optimization, Application Platform Optimization, and Business Productivity Infrastructure Optimization. According to the Core Infrastructure Optimization Model, having administrator-controlled automated physical or virtual application distribution helps move an organization to the Standardized maturity level. Windows Deployment Services gives the administrator control over operating system deployment by providing a mechanism to deploy operating systems with an increased level of customization while using a common base image for all installation scenarios. On the path to the Standardized level, organizations can use Windows Deployment Services to enable Zero Touch Installation (ZTI) for operating system deployments, making it possible to manage operating system refreshes and new installations from a central location.

 

Figure 1. Mapping of Windows Deployment Services technology into Core Infrastructure Optimization Model

# IPD in Microsoft Operations Framework (MOF 4.0)

Microsoft Operations Framework (MOF) offers integrated best practices, principles, and activities to assist an organization in achieving reliable solutions and services. MOF provides guidance to help individuals and organizations create, operate, and support technology services, while helping to ensure the investment in technology delivers expected business value at an acceptable level of risk. MOF’s question-based guidance helps to determine what is needed for an organization now, as well as providing activities that will keep the organization running efficiently and effectively in the future.

Use MOF with IPD guides to ensure that people and process considerations are addressed when changes to an organization’s technology services are being planned.

* Use the Plan Phase to maintain focus on meeting business needs, consider business requirements and constraints, and align business strategy with the technology strategy. IPD helps to define an architecture that delivers the right solution as determined in the Plan Phase.
* Use the Deliver Phase to build solutions and deploy updated technology. In this phase, IPD helps IT pros design their technology infrastructures.
* Use the Operate Phase to plan for operations, service monitoring and control, as well as troubleshooting. The appropriate infrastructure, built with the help of IPD guides, can increase the efficiency and effectiveness of operating activities.
* Use the Manage Layer to work effectively and efficiently to make decisions that are in compliance with management objectives. The full value of sound architectural practices embodied in IPD will help deliver value to the top levels of a business.



Figure 2. The architecture of Microsoft Operations Framework (MOF) 4.0

# Windows Deployment Services Design Process

The goal of this guide is to address the most common scenarios, decisions, activities, options, tasks, and outcomes. Windows Deployment Services architecture has well-defined requirements and supported configurations; it is highly recommended that a Product Support Services (PSS) support review be performed for all Microsoft product implementations.

## Decisions

The six steps below represent the critical design decisions and activities in a successful, well-planned Windows Deployment Services implementation:

* Step 1: Determine the Number of Windows Deployment Services Instances Required
* Step 2: Is There an Existing Windows Deployment Services or RIS Infrastructure?
* Step 3: Select Between Full Windows Deployment Services or Transport Server Role
* Step 4: Determine the Server Resource Requirements
* Step 5: Determine the File Share Fault-Tolerance and Consistency Mechanism
* Step 6: Determine the Client Windows Deployment Services Discovery Method

Some of these items represent decisions that must be made. Where this is the case, a corresponding list of common options is presented.

Other items in this list represent tasks that must be carried out. These items are addressed because their presence is significant to completing the infrastructure design.

## Decision Flow

The following figure provides a graphical overview of the steps in designing a Windows Deployment Services infrastructure.



Figure 3. Windows Deployment Services decision flow

## Information Collection

The following physical location characteristics (Step 1) are required for designing a Windows Deployment Services infrastructure:

* Geographic location
* Logical site name
* AD DS site code
* Network link speed
* Network multicast capabilities and multicast Internet Protocol (IP) scheme
* Network link available bandwidth and utilization
* Network connectivity map or diagram (showing the connections between sites)
* Number of clients
* DHCP and AD DS provided

## Applicable Scenarios

This guide addresses considerations that are related to planning and designing the necessary components for a successful Windows Deployment Services infrastructure:

* Planning for Windows Image (WIM)-based and VHD-based image deployment via Windows Deployment Services
* Planning for WIM-based and VHD-based image storage
* Planning for satellite deployments
* Planning bandwidth considerations
* Planning WIM-based and VHD-based image storage fault tolerance and image consistency

## Out of Scope

Windows Deployment Services is utilized in the Microsoft Deployment Toolkit (MDT) that, like the IPD guides, is available from the Solution Accelerators team. This Windows Deployment Services guide can be used for planning the architecture of Windows Deployment Services and be used within MDT. However, MDT itself is out of the scope of this guide. MDT unifies the tools and processes required for desktop and server deployment into a common deployment console and collection of guidance.

MDT and the included end-to-end guidance reduce deployment time, standardize desktop and server images, limit service disruptions, and reduce post-deployment help desk costs. MDT is available at <http://www.microsoft.com/downloads/details.aspx?familyid=3bd8561f-77ac-4400-a0c1-fe871c461a89&displaylang=en&tm>.

# Step 1: Determine the Number of Windows Deployment Services Instances Required

This step provides guidance on identifying the locations that require a Windows Deployment Services instance. Each Windows Deployment Services instance consists of a Windows Deployment Services server with access to an image storage system. The number of instances identified determines the number of times the design process will be applied. User or business requirements may drive multiple instances of Windows Deployment Services within a single physical location.

## Task 1: Identify Locations Requiring Access to a Windows Deployment Services Instance

For every location in the environment that requires image deployments to a client, access to at least one Windows Deployment Services instance will be required. If the clients are separated by a wide area network (WAN) from the planned Windows Deployment Services instance, ensure that the WAN provides low latency and enough available bandwidth for Windows Deployment Services to function properly.

For best performance, plan to place the Windows Deployment Services servers as close to the clients as possible. For small satellite offices, a Windows Deployment Services server in the hub can be used provided that the WAN link has sufficient bandwidth and low latency. The decision to associate a satellite to a hub Windows Deployment Services server may need to be re-evaluated once the final design is finished.

Identify each location within the organization that will require image deployments, as well as identify any locations that already contain Windows Deployment Services 2003 or RIS infrastructure. Record this information in Table A-1 “Instances Inventory” in Appendix A: “Job Aids.”

## Task 2: Determine the Need for Multiple Windows Deployment Services Installations in a Single Location

Although a single Windows Deployment Services instance may be sufficient to meet the image deployment requirements of a location, additional requirements may force the architect to plan for multiple Windows Deployment Services instances within a single physical location.

Additional Windows Deployment Services instances may be required for the following reasons:

* **Isolated network.** There may be isolated networks within the location that require image deployments. Training labs, for example, may be separated from the organization’s network so that services deployed within the lab do not affect the overall networking infrastructure of the organization. A separate Windows Deployment Services deployment within the lab can be used to quickly redeploy lab workstations.
* **Low bandwidth availability or high latency.** If clients are separated from the Windows Deployment Services servers by segments that have low available bandwidth or have high latency, another Windows Deployment Services deployment may be necessary to handle the clients separated from the initial Windows Deployment Services servers by the problematic segment. Networks with a latency of 5 milliseconds or higher will be severely affected.

For each location identified in task 1, determine the number of Windows Deployment Services deployments required for the location. Record this information in Table A-1 in Appendix A.

## Step Summary

At this point, the location and number of Windows Deployment Services instances has been identified and recorded in Table A-1 in Appendix A. For each of these instances, the architect will need to repeat this step through the planning process.

# Step 2: Is There an Existing Windows Deployment Services or RIS Infrastructure?

The next step is to identify whether a new Windows Deployment Services 2008 instance will be necessary or whether an existing Windows Deployment Services 2003 or Remote Installation Services (RIS) infrastructure will be upgraded for each Windows Deployment Services instance identified in Step 1. In locations requiring a Windows Deployment Services instance with no legacy infrastructure, a new Windows Deployment Services 2008 instance will be planned. However, locations with existing legacy infrastructures need to be evaluated for replacement or upgrade.

Use Table A-2 “New or Upgrade Installation Inventory” in Appendix A to record locations in the existing infrastructure. Also record whether the decision was made to upgrade the existing infrastructure or to put a new installation in place.

## Option 1: New Windows Deployment Services 2008 Installation

If the location does not have existing infrastructure for image deployment, then a new Windows Deployment Services 2008 instance will be planned.

For locations with legacy Windows Deployment Services or RIS infrastructures, a new Windows Deployment Services 2008 instance can be planned as well, rather than upgrading the existing infrastructure to Windows Deployment Services 2008. Once the new Windows Deployment Services 2008 instance is deployed, the legacy infrastructure can be decommissioned.

A new Windows Deployment Services instance may be required for the following reasons:

* **Legacy hardware issues.** The existing hardware used for the infrastructure does not support Windows Server 2008.
* **Restructure or simplification of architecture.** The legacy infrastructure may be overly complex or have architectural design issues. It could be more cost-effective to plan a new deployment than to fix the issues with the current design.

## Option 2: Upgrade Existing Windows Deployment Services 2003 or RIS Infrastructure

In order to upgrade legacy infrastructure to Windows Deployment Services 2008, the infrastructure will have to be configured with Windows Deployment Services 2003 in native mode. Windows Deployment Services 2003 operates in three different modes, depending upon the method of installation and the type of image formats that are installed on the server.

The three modes are:

* **Legacy**. Supports RIS images and OSChooser only.
* **Mixed**. Supports RIS images and OSChooser, as well as WIM-based images and Windows Preinstallation Environment (Windows PE).
* **Native**. Supports only WIM-based images and Windows PE.

The process to move a Windows Deployment Services 2003 server to native mode consists of converting or replacing all images with WIM-based images. Once Windows Deployment Services 2003 is placed in native mode, it no longer supports RIS images or the OSChooser, which is used by clients to determine which RIS image to deploy.

Legacy RIS servers cannot be directly upgraded to Windows Deployment Services 2008. Windows Server 2003 or Windows Server 2000-based RIS infrastructures must first be upgraded to Windows Deployment Services 2003. Mixed mode is enabled on the server, and then the images are migrated to the WIM-based format. Finally, the server is configured for native mode and upgraded to Windows Server 2008.

## Evaluating the Characteristics

Table 1. Evaluate Characteristics

|  |  |  |
| --- | --- | --- |
| Complexity | Justification | Rating |
| New installation | A new installation can be designed for the needs of the organization existing today. | Medium |
| Upgrade an installation | The existing design, which is inherited through the upgrade, may not be the most efficient design for the organization.  | Low |
| Cost | Justification | Rating |
| New installation | A new installation can enable planning for new hardware that can handle large numbers of clients with smaller number of servers, lowering the cost of operations. | Medium |
| Upgrade an installation | Operational, planning, and hardware upgrade costs may be much more expensive. All design choices made previously for the legacy infrastructure are inherited. | Low |
| Performance | Justification | Rating |
| New installation | The infrastructure can be planned to handle the requirements of the organization existing today. | ↑ |
| Upgrade an installation | Existing hardware may be stressed with the addition of Windows Deployment Services 2008, reducing the number of clients the service can effectively address. | ↓ |

## Validating with the Business

In addition to evaluating the decision in this step against IT-related criteria, the effect of the decision on the business should also be validated. Ask the business the following question before making this design decision:

* Does the existing legacy infrastructure meet the needs of the organization?The legacy architecture may no longer meet the organization’s requirements. It may not have been carefully designed originally. Or it may no longer handle the requirements for image delivery required by the organization today. If the current infrastructure does not meet the needs of the organization, a new instance should be created to meet the expected requirements.

## Tasks and Considerations

The Windows Image (WIM) format is supported by Windows Deployment Services 2003 in mixed or native mode as well as Windows Deployment Services 2008. As new Windows Deployment Services 2008 servers are put into production, the WIM format-based images used on existing Windows Deployment Services 2003 servers can be used with Windows Deployment Services 2008. The new Windows Deployment Services 2008 server can use an existing remote image store containing the existing WIM-based and VHD-based images.

If Windows Deployment Services 2003 is using RIS-based images (RIPREP or RISETUP) or if Windows Server 2003 or Windows 2000 Server RIS is being used, ensure that an equivalent image is offered in WIM format for Windows Deployment Services 2008. Otherwise, clients will get inconsistent offers for images.

## Step Summary

For each location within the environment that has been identified for Windows Deployment Services, a determination has been made whether to deploy a new Windows Deployment Services 2008 instance or to upgrade the existing Windows Deployment Services 2003 or RIS infrastructure. This information was recorded in Table A-2 in Appendix A.

For information on how to upgrade a legacy Windows Deployment Services 2003 and RIS infrastructure to Windows Deployment Services 2008, see the product documentation for Windows Deployment Services 2008 included in its help file.

## Additional Reading

* Windows Deployment Services Deployment Guide: [http://technet.microsoft.com/en-us/library/cc770667(WS.10).aspx](http://technet.microsoft.com/en-us/library/cc770667%28WS.10%29.aspx)
* Windows Deployment Services Getting Started Guide: [http://technet.microsoft.com/en-us/library/cc771670(WS.10).aspx](http://technet.microsoft.com/en-us/library/cc771670%28WS.10%29.aspx)
* Windows Deployment Services TechCenter page: [http://technet.microsoft.com/en-us/library/cc772106(WS.10).aspx](http://technet.microsoft.com/en-us/library/cc772106%28WS.10%29.aspx)

# Step 3: Select Between Full Windows Deployment Services or Transport Server Role

In this step, for each instance, determine whether a full Windows Deployment Services server will be deployed, or only the Transport Server role. This information will be used in determining the server requirements later on in the process. Record these decisions in Table A-3 “Full Windows Deployment Services or Transport Server Roll Inventory” in Appendix A: “Job Aids.”

## Option 1: Full Windows Deployment Services

This option provides the full functionality of Windows Deployment Services. This option requires that AD DS, DHCP, and DNS be available in the environment.

Features provided by a full Windows Deployment Services instantiation include:

* PXE boot services.
* Microsoft Management Console (MMC) tools.
* The ability for the client to select which image to install from a presented list.
* Both unicast and multicast deployments.

## Option 2: Transport Server Only

This option provides only the core networking components required for creating and managing a multicast stream. A multicast stream allows multiple clients to tune into a stream of data without requiring the data to be sent individually to each client on a separate unicast stream. The Transport Server does not require AD DS, DHCP, or DNS.

This option should be selected when AD DS, DHCP, or DNS are not available in the environment. For example, a data center that blocks DHCP within the server room may use this method to deploy server images.

Windows Deployment Services in Windows Server 2008 R2 includes PXE startup.
Windows Deployment Services in Windows Server 2008 without R2 does not support PXE startup. Because of this, all machines that must be imaged must be manually booted using a custom boot image that is tied to the server and the multicast stream.
This adds a level of cost and complexity to the deployment process around boot image management.

## Evaluating the Characteristics

Table 2. Evaluate Characteristics

| Complexity | Justification | Rating |
| --- | --- | --- |
| Full Windows Deployment Services | Requires AD DS, DHCP, and DNS. | Medium |
| Transport Server only | No requirements other than network multicast support. | Low |

## Validating with the Business

In addition to evaluating the decision in this step against IT-related criteria, the effect of the decision on the business should also be validated. The following questions have been known to affect determining server requirements:

* **Does the organization have the expertise to create and manage custom boot images?** Creating and managing custom boot images takes additional planning and expertise to successfully accomplish. Ensure that appropriate training is in place if necessary.
* **Are there any business policies in place that prohibit the use of DHCP within a data center?** DHCP is required for a full Windows Deployment Services server; it is not required for just the Transport Server role.

## Step Summary

For each instance, it was determined whether a full Windows Deployment Services server will be deployed, or just the Transport Server role will be instantiated in the location. This decision can affect the networking requirements in that the Transport Server role can only provide a multicast stream of data. This information was recorded in Table A-3 in Appendix A.

## Additional Reading

* Windows Deployment Services Deployment Guide: [http://technet.microsoft.com/en-us/library/cc770667(WS.10).aspx](http://technet.microsoft.com/en-us/library/cc770667%28WS.10%29.aspx)
* Windows Deployment Services Getting Started Guide: [http://technet.microsoft.com/en-us/library/cc771670(WS.10).aspx](http://technet.microsoft.com/en-us/library/cc771670%28WS.10%29.aspx)
* Windows Deployment Services TechCenter page: [http://technet.microsoft.com/en-us/library/cc772106(WS.10).aspx](http://technet.microsoft.com/en-us/library/cc772106%28WS.10%29.aspx)

# Step 4: Determine the Server Resource Requirements

This step determines the size and number of Windows Deployment Services servers required for each Windows Deployment Services instance. Deployment requirements are identified for the instances, and the servers are then scaled, both up and out, to meet those requirements.

## Task 1: Identify Deployment Requirements for Each Windows Deployment Services Instance

In order to determine the number of servers and the hardware configuration of the servers, several key pieces of information must be gathered for each Windows Deployment Services instance:

* **Total number of computers.** Identify the total number of clients that are to be supported. This information informs the peak number of simultaneous imaging requests that should be handled by the infrastructure. The worst case scenario is that all clients are imaged at once.
* **Image deployment speed.** Identify the targeted amount of time that an image deployment should take. As a baseline, one computer can take 25 minutes from PXE boot to functioning desktop for a standard Windows Vista® install, not including applications. The time to stream (or distribute) an image can be minimized by increasing the size of the hardware and the number of servers.
* **Size and number of images.** Identify the total number of images available in the location and the size of each image. This information factors in to the disk capacity requirements and memory requirements of the server.

Record the information gathered in this step in Table A-4 “Instance Requirements” in Appendix A: “Job Aids.”

## Task 2: Determine Whether Virtualization Will Be Used

For each instance, determine whether the Windows Deployment Services infrastructure will be physical or virtual.

### Option 1: Physical Hardware

Using physical hardware can provide greater choices and flexibility in the type of hardware utilized in the Windows Deployment Services infrastructure. Both x86 and x64 architectures are available as well as multi-processor support. Physical hardware instantiations can potentially handle more clients than a virtualized environment due to the lack of the overhead introduced by the virtual services.

### Option 2: Virtual Machine

Virtualization introduces flexibility into an environment by allowing virtual machines (VMs) to be moved easily between hosts. Services that may not be compatible with each other can both be run on the same host due to the isolation provided by the VM. However, virtualization overhead may affect the number of clients that can be supported by a
given VM.

Additionally, the type of virtualization environment can have a large impact. Microsoft Hyper-V™ environments give better performance than Microsoft Virtual Server 2005 R2. Hyper-V supports 64-bit environments, multiple central processing units (CPUs), and lower virtualization overhead versus Virtual Server 2005 R2.

The scaling of the VMs will be discussed in this guide. However, additional planning is required for properly provisioning the physical machines that will host the VMs. The IPD guide for *Windows Server Virtualization* provides the planning process for server virtualization and is available at [www.microsoft.com/IPD](http://www.microsoft.com/ipd).

Testing is required to determine the correct client-to-virtualized Windows Deployment Services server load ratio.

## Evaluating the Characteristics

Table 3. Evaluate Characteristics

|  |  |  |
| --- | --- | --- |
| Complexity | Justification | Rating |
| Physical hardware | The unique characteristics of the physical hardware need to be addressed. This, in addition to the setup of the hardware, makes this more complex than a virtual machine. | Medium |
| Virtual machine | The creation of a single virtual machine is less complex due to standardized virtualized hardware. | Low |
| Cost | Justification | Rating |
| Physical hardware | Additional capital expenses as well as time investment in setting up the physical hardware. | High |
| Virtual machine | Use of virtual machines can reduce capital costs because large servers can safely share loads. | Low |
| Security | Justification | Security |
| Physical hardware | If using physical hardware for the Windows Deployment Services infrastructure, control of the entire physical server can be maintained. | Physical hardware |
| Virtual machine | A virtual machine will run on a virtual server host. Determining what runs on this host in addition to Windows Deployment Services and who owns the infrastructure may be outside the control of Windows Deployment Services owners. In addition, the VM’s virtual hard disk may be stored on a file system that the Windows Deployment Services owner has no control over. | Virtual machine |

## Validating with the Business

In addition to evaluating the decision in this step against IT-related criteria, the effect of the decision on the business should also be validated. Ask the business the following question prior to making this design decision:

* Does the organization have a business strategy for virtualization? If the organization is making an investment in virtualization, it may make sense to investigate the appropriateness of hosting Windows Deployment Services in virtual machines. If there is no strategy around virtualization, the additional overhead for planning the virtualization infrastructure might not be worth the additional resource investment just for Windows Deployment Services.

Record whether the Windows Deployment Services infrastructure will be physical or virtual in Table A-5 “Physical or Virtual Instance Requirement” in Appendix A.

## Task 3: Determine Image Storage Location

While the boot image files are always stored locally to the Windows Deployment Services server, a decision needs to be made whether to store image files locally or on a remote file server. This needs to be decided for each instance. The image files may be WIM-based or VHD-based. New for Windows Server 2008 R2, VHD-based images can be used to deploy images of Windows Server 2008 R2 to a physical (not virtual) computer using Windows Deployment Services.

Note   It is assumed, regardless of the location chosen, that the disk system will be configured
to provide fault tolerance at the hardware level through the use of redundant array of independent disks (RAID).

### Option 1: Local Storage

In this option, images are stored locally on the same machine as Windows Deployment Services. The disk storage can be provided by a local disk or through a storage area network (SAN). The images are accessed through a share named REMINST created during installation.

Note   SAN usage as a local store has not been tested by the product group. While there are no anticipated issues with this configuration, testing should be performed prior to rolling out this configuration in production.

### Option 2: Remote File Share

In this option, images can be stored on a remote file share. The remote file share can be hosted on another file server or a network-attached storage system. Additional configuration is required to present the image group that is available remotely. See the product documentation for additional guidance.

This option is useful for reducing the amount of traffic on a WAN when clients in a satellite location access a Windows Deployment Services server in a hub location. The WIM-based or VDH-based image copy can be directed to a local file share within the satellite site.

If this option is used, it is important that there is sufficient network bandwidth between the remote file share and both the Windows Deployment Services server and clients.

### Evaluating the Characteristics

Table 4. Evaluate Characteristics

|  |  |  |
| --- | --- | --- |
| Complexity | Justification | Rating |
| Local storage | This is the installation default. | Low |
| Remote file share | Additional configuration is required in order for Windows Deployment Services to refer clients to the remote file share. | Medium |
| Security | Justification | Rating |
| Local storage | The system is self-contained. | → |
| Remote file share | Ownership of the remote file share may reside outside the Windows Deployment Services owner’s control.  | ↓ |

For each instance, determine whether the image files will be stored locally or on a remote file server, and record the decision in Table A-5 in Appendix A.

## Task 4: Scale the Servers

For each instance, the WDS servers and remote file servers, if used, need to be scaled to handle the expected load.

If there is an existing Windows Deployment Services 2003 or RIS infrastructure deployed within the organization, it can be monitored for performance to obtain a baseline for how the servers should be scaled.

In instances where there is no existing infrastructure or the existing infrastructure is unsuitable, then start with one of the tested configurations listed below for Windows Deployment Services 2008 and monitor the performance.

In either case, adjust the configuration as necessary to handle the load.

This information was gathered from the “Optimizing Performance and Scalability” chapter in the *Windows Deployment Services Deployment Guide,* available at [http://technet.microsoft.com/en-us/library/cc732088(WS.10).aspx](http://technet.microsoft.com/en-us/library/cc732088%28WS.10%29.aspx).

Note   The guidance provided here assumes that the Windows Deployment Services server is not sharing resources with other services or applications. Windows Deployment Services can be co-located with other services, but scaling the additional workload requirements for the other services is beyond the scope of this document.

The next two tables show the approximate time under various circumstances that it took for all clients to apply an install image. This includes the initial PXE boot, download of the Boot.wim, download of a standard Windows Vista image, and application of the image.

The next table shows the start-to-finish times for applying an image using multicast.

Table 5. Start-to-Finish Times for Image-Apply Phase Using Multicast

| Checkpoint | 25 clients | 100 clients | 300 clients |
| --- | --- | --- | --- |
| Start | 0 | 0 | 0 |
| TFTP Start | 0:23 | 0:21 | 0:23 |
| TFTP Finish | 1:02 | 2:40 | 7:16 |
| MC Start | 3:04 | 3:55 | 8:18 |
| MC Finish | 6:06 | 7:54 | 12:30 |
| Desktop | 19:47 | 22:40 | 27:40 |

The next table shows the start-to-finish times for applying an image using server message block (SMB).

Table 6. Start-to-Finish Times for the Image-Apply Phase Using SMB

| **Checkpoint** | **25 clients** | **100 clients** | **300 clients** |
| --- | --- | --- | --- |
| Start | 0 | 0 | 0 |
| TFTP Start | 0:21 | 0:22 | 0:20 |
| TFTP Finish | 0:58 | 2:40 | 7:13 |
| SMB Start | 3:14 | 4:38 | 8:29 |
| SMB Finish | 13:36 | 38:15 | 1:47:58 |
| Desktop | 20:59 | 45:37 | 1:55:15 |

The next table shows the configurations of the server, client, and image files.

Table 7. Configurations Used in Testing.

| **Item** | **Configuration** |
| --- | --- |
| Server | Dual processor Xenon 5150 2.67 GHz8 GB of random access memory (RAM)Windows Server 2008 64 bit1-gigabit network adapter |
| Client | Windows Vista capable x86100-megabit network adapter |
| Images | Windows Server 2008 RTM x86 boot image ~128 MBWindows Server 2008 RTM x86 Install Image ~1.32 GB  |

For best performance in large organizations, it is recommended that Windows Deployment Services be deployed to its own server. However, if Windows Deployment Services will be co-located with other services, keep the following guidelines in mind:

* Do not place Windows Deployment Services on the same computer that is network-intensive or CPU-intensive—for example, Exchange Server or Microsoft SQL Server®. The high traffic levels of Windows Deployment Services can degrade the performance of these products and vice versa.
* Windows Deployment Services cannot be hosted on a computer that only has a wireless network connection.

Note   If DHCP and Windows Deployment Services are co-located together, two additional configuration changes are required to ensure operability:

* Use the management tools, WDSUTIL or the Windows Deployment Server MMC, to configure Windows Deployment Services to stop listening on the DHCP ports.
* All active DHCP scopes are updated to include the optional 60-client identifier, which is configured with the value of PXE Client. This option allows DHCP clients to learn about the Windows Deployment Services server from normal DHCP requests.

### CPU

Windows Deployment Services is primarily input/output (I/O) bound by the network and the speed that the image data file can be read from the disk. The information provided above can be used as a baseline for determining the architecture and number of CPUs.

If additional services are placed on the same server as Windows Deployment Services, then the type, number, and speed of processors may be adjusted to handle the additional load. Additional performance monitoring and testing may be required to determine if the processor is creating a bottleneck.

The following are general guidelines when selecting new hardware:

* **64-bit processors**. Select 64-bit hardware for the benefit of additional address space.
* **Multi-core and multiple processors.** Research data has shown that two CPUs are not as fast as one CPU that is twice as fast. Because it is not always possible to get a CPU that is twice as fast, doubling the number of CPUs is preferred, but it does not guarantee twice the performance.
* **Do not use CPU speed as a comparison.** This can be a misleading indicator of performance, particularly across manufacturers and generations of processors.
* **L2 or L3 cache.** Larger L2 or L3 processor caches generally provide better performance and often play a bigger role than raw CPU frequency.

Record the number of CPUs, architecture, and speed expected to be required for handling the required loads in Table A-5 in Appendix A.

Additional information on CPU scaling can be found in Appendix B: “Server Performance Analyzing and Scaling” of this guide.

### Memory

Windows Deployment Services attempts to cache the operating system image files in memory after the initial client request for the image. This decreases the response time for additional requests of the image as the server does not read the image from disk again. Increasing the memory capacity of the server to allow for more images to be cached can improve the performance of the server.

Use the size and number of images required for a location to help determine how much RAM should be allocated to the server above the base requirements for the operating system. Frequency of image deployment can also be used to tune this number. If an image isn’t frequently deployed, then adding additional RAM to cache that image may not be cost effective.

Due to limitations on the physical amount of RAM that an x86 system can reference, it may be beneficial to run the Windows Deployment Services server on an x64 system.

Record the amount of RAM required for the server in Table A-5 in Appendix A.

### Network

For each server, the size and number of network adapters should be determined. The available bandwidth and the latency of the network between the clients and the location of the images have the greatest impact on the performance of the infrastructure.

Windows Deployment Services performs best using a 1-GB per second network adapter. In tests performed by Microsoft, a Windows Deployment Services server using a 100-MB per second network adapter could deploy up to 10 images in under an hour, independent of the server RAM, disk speed, or processor speed. By contrast, a server using a 1-gigabit network adapter could simultaneously image 75 clients in 45 minutes.

The following are general guidelines for selecting new hardware:

* **64-bit capability.** Adapters that are 64-bit capable can perform direct memory access (DMA) operations to and from high physical locations (above 4 GB). If the driver does not support DMA above 4 GB, the system double-buffers the I/O to a physical address space of less than 4 GB.
* **Copper versus fiber adapters.** Copper adapters generally have the same performance as their fiber counterparts. Certain environments may favor one over the other. For example, in environments with high electrical noise, fiber would be a better choice.
* **Dual-port or quad-port adapters.** These can be useful for servers with limited Peripheral Component Interconnect (PCI) slots. However, using two single-port network adapters usually yields better performance than using a single dual-port network adapter for the same workload.
* **Interrupt moderation.** Some adapters can moderate how frequently they interrupt the host processors. Moderating interrupts can often result in a reduction in CPU load on the host but if it is not performed intelligently, the CPU savings might cause increases in latency.
* **Offload capability.** Offload-capable adapters offer CPU savings that translate into improved performance. For Windows Deployment Services and file servers, look for adapters that provide checksum offload, segmentation offload (GSO), and Transmission Control Protocol (TCP) offload engine (TOE).
* **PCI bus.** PCI bus limitations can be a major factor in limiting performance, particularly for multi-port adapters. It is important to consider placing them in a high-performing PCI slot that provides adequate bandwidth. In general, Peripheral Component Interconnect Express (PCI-E) adapters provide higher bandwidth than Peripheral Component Interconnect Exchange (PCI-X) adapters.

Determine the type of network adapter required for the load to be placed against the system. If multiple network adapters are being used in the server to increase the bandwidth, ensure that the network switching hardware is compatible with the configuration.

For each server, record the type and number of network adapters being used in Table
A-5 in Appendix A.

### Disk

The disk capacity and performance requirements should be recorded for each server (both Windows Deployment Services and remote file servers, if any), which is part of the infrastructure. The boot image files tend to be small, on the order of 50 MB. If a Windows Deployment Services server stores the image files remotely, then the amount of local disk space required will be negligible.

The required disk capacity is determined by adding the size of each image required for the location and allowing some overhead for future growth.

Disk performance has the second greatest impact on the performance of the infrastructure. The disk subsystem is scaled to handle the expected number of I/Os per second (IOps) generated by the client requests. The capacity of each spindle, numbers of spindles, speed of the spindles, and RAID configuration of the spindles all have an effect upon the number of IOps that can be handled at a given time. The choice to use unicast versus multicast streaming can also affect the performance requirements of the disk system. A multicast stream will require less performance around IOps than a unicast SMB stream handling the same number of clients.

Note   Determining the IOps generated by a client can be difficult. It is possible to measure another system with a similar workload, such as a file server, to get an idea of the required IOps. The IOps derived can be used with the calculations provided in Appendix B: “Server Performance Analyzing and Scaling” to determine the RAID configuration and number of spindles necessary to meet the performance requirement.

In addition to the disk performance and capacity, if the images are stored remotely, then the server or network-attached storage system should be tested to ensure that it can provide images at the required performance level. If the remote file share is utilized by more than one Windows Deployment Services server, then scale the CPU, memory, disk performance, and network in order to handle the highest number of expected simultaneous image deployments. The throughput from the storage device to Windows Deployment Services should be consistent. If degradation occurs, image deployment times will increase or deployment may fail.

For each server, record the required disk configuration in Table A-5 in Appendix A.

For more information on calculating disk performance, see Appendix B: “Server Performance Analyzing and Scaling” of this guide.

## Task 5: Determine Number of Servers for Location

Now that the server sizing has been finished, the number of servers for each location needs to be determined.

It is possible that a single server will meet the requirements for each location with regard to the number of clients and the ability to stream the images within the required time.

However, additional servers may be necessary for the following reasons:

* **Cost of hardware.** The capital costs for a single server to handle the load may be too high. Sharing the load across multiple servers within the location may be more cost effective. To identify the number of servers required, determine the size of a server that can handle a subset of the client population in the required time frame, and then scale out the server to handle the entire client population.
* **Unable to meet time expectations.** By reducing the number of client requests that the server is required to handle, the time it takes to deploy an image can be reduced. Performance tests can be run against the server hardware to determine the number of clients that can be serviced and still meet the time expectation. Additional servers are then added to the environment to handle the clients.
* **Availability requirements.** Windows Deployment Services does not support any form of fault tolerance, including server clusters. An additional server can be added to increase the availability of the system in some cases. While multiple servers can be used to cover the same sets of clients, there is no explicit load balancing capability available for the Windows Deployment Services server. In instances where the client is pre-staged in AD DS, that client will fail to find the redundant server.

For each location, identify the total number of servers required to meet the organization requirements for image deployment and record this in Table A-5 in Appendix A.

## Tasks and Considerations

For a client to receive an image from a full Windows Deployment Services server, a two-stage process occurs.

During the first stage, the client uses Trivial File Transfer Protocol (TFTP) to download the boot image from the Windows Deployment Services server. This image is kept as small as possible to reduce the amount of time required to transfer the information. Network latency can have a significant impact on the TFTP protocol, increasing the time required for download.

The second stage involves transferring the operating system image file to the client. This can be accomplished in two ways: unicast or multicast.

A client using unicast will transfer the image file data using SMB protocols. Windows Deployment Services will either deliver the file, if it is stored locally, or return the network path of the file on the remote file server. Since these image files tend to be quite large, as the number of client requests over SMB increases, the amount of available bandwidth can become constrained. This can be especially true of satellite offices using Windows Deployment Services over a WAN link.

If multicast is being used, each client requesting the same image will tune into the single multicast stream. This reduces the amount of bandwidth taken up by the clients, as well as server load, as each client is not requesting its own stream of data. The clients will receive the image file from the Windows Deployment Services server. The Windows Deployment Services server may need to pull the file data from a remote file server prior to serving it up to the clients. In order for multicast to function properly, the network switching hardware must be capable of transmitting multicast packets. Most modern managed switches and routing equipment support multicasting. However, it is important to test the equipment for performance and compatibility.

With Windows Server 2008 R2, there is the capability for multiple stream transfer, meaning there can be multiple bands to broadcast images to clients. This optimizes the transfer rate per connected client, but it does require a Windows 7 boot image. The second significant capability introduced is Client Auto Removal, where slower clients can be dropped back to unicast, or even dropped completely, allowing the remaining clients to complete the image transfer more reliably. This is supported with Windows 7 or Windows Vista SP1 or later boot images.

If the images are stored remotely, ensure that there is sufficient bandwidth between the remote location and Windows Deployment Services and the clients. If degradation occurs, image deployment times will increase or the deployment may fail.

## Step Summary

For each location, the deployment requirements have been identified as well as deciding whether to use physical hardware or virtual machines for the Windows Deployment Services infrastructure. The image storage location has been identified. And finally, the size and number of servers has been determined for each location. All of the data collected in this step was recorded in Table A-4 and Table A-5 in Appendix A.

# Step 5: Determine the File Share Fault-Tolerance and Consistency Mechanism

This section determines the mechanisms used to provide fault tolerance to the WIM-based and VHD-based operating system images within the Windows Deployment Services infrastructure. In addition, mechanisms for keeping images consistent are also determined.

## Decision 1: Determine Image Storage System Fault Tolerance

In order to increase the availability of the infrastructure, the share through which the WIM- based and VHD-based images are accessed can be made fault tolerant.
The shares that are made fault tolerant include the REMINST share on the Windows Deployment Services server and any shares used on remote file servers. Determine the method for making all shares used in the infrastructure fault tolerant.

### Option 1: Distributed File System

Distributed File System (DFS) can be used to provide a fault-tolerant method for accessing file shares. DFS allows the administrator to define a file namespace and provide multiple targets for folders contained within the namespace. When a client attempts to access a DFS-enabled share, the request is handled by the nearest DFS server hosting that particular share. This can be used to control which server will provide the client with the install image. This is particularly useful for controlling bandwidth usage on WANs between clients in satellite sites and remote Windows Deployment Services servers in the hub.

The process is listed below and then illustrated in the following diagram:

1. The client in virtual local area network (VLAN) 1 requests the Windows PE image from the Windows Deployment Services server.
2. Once the client boot process has started with the Windows PE image, it attempts to transfer the installation image that has been mapped to a DFS share. In the VLAN 1 scenario, this means that the client accesses File Server 1 to load the installation image.
3. Clients in VLAN 2 and VLAN 3 use their respective file servers to access the installation image that they require.



Figure 4. Example of DFS usage when using unicast

If the operating system images are stored locally, DFS can still be used to present a unified namespace with failover capabilities. For example, consider a case where two Windows Deployment Services servers are present: WDS1 and WDS2. If a client receives a boot image from WDS1 and if the file share later becomes unavailable on WDS1 for some reason, the client can still receive the image from WDS2.

### Option 2: Server Clustering

Server clustering can increase the fault tolerance of a single content storage system file share. The file share becomes a clustered resource running on a cluster with two or more computers. If the computer hosting the file share fails, the file share moves to a remaining active node.

The most practical approach to using server clustering is when an existing file server cluster is already implemented within the environment and can meet the expected capacity and performance requirements introduced by Windows Deployment Services.

Note   Server clustering cannot be used to provide fault tolerance when the content is locally stored with the Windows Deployment Services system because Windows Deployment Services is not cluster-aware. Server clustering, therefore, is not supported.

Although a share hosted in a server cluster can become part of a DFS namespace, the content of the share cannot be replicated using Distributed File System Replication
(DFS-R).

## Evaluating the Characteristics

Table 8. Evaluate Characteristics

| Complexity | Justification | Rating |
| --- | --- | --- |
| DFS | Configuring the DFS namespace can be moderately difficult. Microsoft provides guidance for implementing this form of file protection using DFS. | Medium |
| Server clustering | Server clustering tends to be extremely complex to set up due to the interaction between networks, shared storage, and specialized hardware and software configurations. | High |
| Cost | Justification | Rating |
| DFS | If using existing file servers, then DFS is fairly low in cost as it is built into the operating system. | Low |
| Server clustering | Server clustering is costly due to the requirements of additional servers and shared storage. | High |
| Scalability | Justification | Rating |
| DFS | DFS allows for multiple copies of the target share to be accessible at the same time through a single namespace. | High |
| Scalability | Justification | Rating |

## Validating with the Business

In addition to evaluating the decision in this step against IT-related criteria, the effect of the decision on the business should also be validated. The following questions have been known to affect fault-tolerant design decisions:

* **Does the organization have a strategy around DFS or server clustering?** Is there a current strategy in place around DFS or server clustering that should be considered? Is there a need to deviate from any standardized strategy?
* **How important is it to the organization to have Windows Deployment Services always available?** Is Windows Deployment Services considered important enough to ensure the fault tolerance of the system? The cost of putting fault tolerance in place may outweigh the benefit received from a non-mission–critical application.

After deciding whether to use DFS or server clustering for the fault-tolerance method, record this information in Table A-6 “Fault-Tolerance Method” in Appendix A: "Job Aids."

## Decision 2: WIM-Based and VHD-Based Image Consistency

The mechanism for managing images is determined in this step. Although the file system or remote share has been made fault tolerant, it’s possible that images that are shared across systems may become inconsistent with respect to each other. This step identifies the method for maintaining and managing the consistency of the images.

### Option 1: Manual Copy/Manage Image Locally

Images are managed locally at each server. In order to share images with other Windows Deployment Services servers, the images are manually copied to the target machines.

### Option 2: DFS with Replication

The built-in DFS Replication (DFS-R), provided in Windows Server 2008, can be used to keep all targets within the DFS tree in sync with each other.

Note   File Replication Service, the predecessor for DFS-R, is not supported by Windows Deployment Services.

Capitalizing on both DFS namespaces and replication provides the following benefits:

* **Load balancing.** Clients can be directed to computers other than the Windows Deployment Services server to perform the image download.
* **Simplified administration.** Management operations on images can be centrally managed and easily propagated to other distribution points.

It is important to note that DFS-R is not supported on a server cluster, although DFS is supported.

### Option 3: Third-Party Replication

Third-party replication systems can be used to provide the image consistency.

## Evaluating the Characteristics

Table 9. Evaluate Characteristics

| Complexity | Justification | Rating |
| --- | --- | --- |
| Manual copy/manage image locally | Manual copy or managing images at each Windows Deployment Services instance is extremely complex to coordinate, particularly if a standard image is required for all sites. | High |
| DFS with replication | Configuring the DFS with replication can be moderately difficult. Microsoft provides guidance for implementing this form of file protection using DFS-R. | Medium |
| Third-party replication | Third-party replication systems will vary on the complexity and knowledge required for implementation and operation. | High |

Table 9. Evaluate Characteristics (Continued)

| Cost | Justification | Rating |
| --- | --- | --- |
| Manual copy/manage image locally | Manual copy or managing images at each Windows Deployment Services instance is extremely costly and prone to errors. | High |
| DFS with replication | If using existing file servers, then DFS is fairly low in cost as it is built into the operating system, although upgrading to Windows Server 2003 R2 or later is required for DFS-R. | Low |
| Third-party replication | Depending upon the licensing costs, third-party replication systems could be costly. In addition, another skill set must be learned in order to appropriately manage the system. | High |
| Fault Tolerance | Justification | Rating |
| Manual copy/manage image locally | Although this method can increase the fault tolerance of the system by providing duplicate copies of the images, it’s an extremely poor option due to the human factor. | ↓ |
| DFS with replication | When replication is combined with DFS namespaces, an extremely fault-tolerant system can be provided. | ↑ |
| Third-party replication | Automatic replication can ensure that the most up-to-date content is available. | ↑ |
| Performance | Justification | Rating |
| Manual copy/manage image locally | This option does not perform well as it requires manual intervention. | ↓ |
| DFS with replication | The design of DFS-R allows for a high level of performance in replicating data on networks. | ↑ |
| Third-party replication | The replication method implemented can have an effect on the performance of the infrastructure as the number of nodes increases. | → |
| Scalability | Justification | Rating |
| Manual copy/manage image locally | This option does not scale well. | ↓ |
| DFS with replication | This option can scale to a large number of servers. | ↑ |
| Third-party replication | Depending upon the third-party application, this can scale. | → |

## Validating with the Business

In addition to evaluating the decision in this step against IT-related criteria, the effect of the decision on the business should also be validated. Ask the business the following question prior to making this design decision:

* **Does the organization have a data replication architecture that can be utilized?** Can an existing data replication method be utilized rather than introducing a new one specific for Windows Deployment Services?

After deciding which method to use to ensure image consistency, record the chosen method in Table A-6 in Appendix A.

## Tasks and Considerations

If replication is being used to distribute images to multiple Windows Deployment Services servers, then the refresh policy of Boot Configuration Data (BCD) stores must be configured on each of the Windows Deployment Services servers.

The BCD refresh setting causes the server to regenerate BCD stores in the \Tmp directory for all Windows PE boot images. The frequency with which it does this operation is controlled by the refresh period configuration. This setting is required in replication scenarios so that changes made to boot images (add, remove, rename, and so on) on the master server are reflected in the boot menus that clients receive from remote servers. This time interval should be set to an appropriate value based on the frequency of image updates. If changes to boot images do not occur very often or if it is acceptable to have a long delay between a modification and when clients at remote sites see the changes, then set this to a higher value. If changes to boot images are made often, or if booting clients are expected to immediately pick up the changes, then set this to a lower value. However, be careful in setting a low value. BCD generation causes CPU and disk overhead on the Windows Deployment Services server, and configuring the value to too small a window can harm performance on the server. A good default value is 30 minutes.

Additionally, the \MGMT and \TMP directories on each Windows Deployment Services server must be excluded from the replication. These folders contain server-specific data.

## Step Summary

For each share utilized in the infrastructure, the mechanism for ensuring fault tolerance is determined. Finally, image consistency among the servers is addressed. All of the data collected in this step was recorded in Table A-6 in Appendix A.

## Additional Reading

“Working with Images” in Windows Deployment Services Deployment Guide:
[http://technet.microsoft.com/en-us/library/cc731843(WS.10).aspx](http://technet.microsoft.com/en-us/library/cc731843%28WS.10%29.aspx)

# Step 6: Determine the Client Windows Deployment Services Discovery Method

For each new Windows Deployment Services instance, determine the method used by clients to discover the Windows Deployment Services servers. Clients discover Windows Deployment Services servers through a PXE boot request, which is a modified DHCP request that is broadcast on the network. When the Windows Deployment Services server and the PXE client reside on the same network segment, no additional changes to the infrastructure are required. The broadcast will be heard by the Windows Deployment Services server.

On networks where the clients and the Windows Deployment Services server are located on separate subnets, a mechanism for discovering the Windows Deployment Services server is required. Clients can discover Windows Deployment Services servers either through network boot referrals or through IP helper updates.

## Option 1: Using Network Boot Referrals

Network boot referrals use DHCP options 66 and 67 configured on the DHCP server to notify the PXE client where to download the network boot program (NBP). The DHCP options are configured for active scopes on the DHCP server and hold the following values:

* 66 = Boot server host name (set to the Windows Deployment Services server name)
* 67 = Boot file name (set to the boot file name that the clients attempt to download and execute)

Using network boot referrals has the following drawbacks:

* Using DHCP options is not as reliable as updating the IP helper tables. In testing, Microsoft has observed some issues (mainly with older PXE ROMs) whereby clients do not correctly parse the DHCP options returned from the DHCP server. The result is that booting clients see a “TFTP Failed” error message. Generally, this problem occurs when the PXE ROM ignores the boot server host name value and instead attempts to download the NBP directly from the DHCP server (which likely does not have the file in question).
* If multiple network boot servers are available to service client requests, specifying the explicit network boot server’s name as part of the DHCP scope may prevent load-balancing from occurring.
* Clients may be directed to a network boot server that is not available. Because the client does not have to contact a network boot server directly to determine the appropriate network boot file to download, the DHCP server may instruct clients to download a nonexistent boot file or direct them to a server that is not available on the network.
* Clients may bypass the network boot server’s answer settings. Many network boot servers on the market today have an on/off mechanism that controls whether certain (or any) client requests are answered. Following the PXE standard, client computers contact the network boot server directly to obtain the path and file name of the NBP. Using DHCP options 66 and 67 may cause the client to bypass this communication with the network boot server completely and therefore circumvent or ignore the network boot server’s settings regarding answering clients.

## Option 2: Using IP Helper Updates

IP helper updates involve configuring router and switching hardware to forward DHCP and PXE boot requests from the network segment where the client is located to the DHCP and Windows Deployment Services server’s segment. Configuration of the IP helpers should follow these guidelines:

* All DHCP broadcasts on User Datagram Protocol (UDP) port 67 by client computers are to be forwarded directly to both the DHCP server and the Windows Deployment Services PXE server. This must not be a rebroadcast of the packet.
* All traffic from the client computers to UDP port 4011 on the Windows Deployment Services PXE server is routed appropriately. This directed traffic must not be blocked by a firewall.

## Evaluating the Characteristics

Table 10. Evaluate Characteristics

| Complexity | Justification | Rating |
| --- | --- | --- |
| Network boot referral | Configuring network boot referrals can be complex because each DHCP server has to be associated with a Windows Deployment Services server. | High |
| IP helper updates | This option reduces complexity because IP helper tables on the routers are updated with the IP address of the Windows Deployment Services servers. | Low |
| Fault Tolerance | Justification | Rating |
| Network boot referral | This option provides no fault tolerance; if the Windows Deployment Services server being referred is down, the client will fail to complete the PXE boot process. | ↓ |
| IP helper updates | This option provides fault tolerance provided that there is more than one Windows Deployment Services server that can answer the client. | ↑ |
| Scalability | Justification | Rating |
| Network boot referral | This configuration allows the DHCP server to refer the client to only one Windows Deployment Services server. | ↓ |
| IP helper updates | This option allows any Windows Deployment Services server in the location to answer a client request. | ↑ |
|  |  |  |

Clients can discover Windows Deployment Services servers either through network boot referrals or through IP helper updates. After deciding which method to use in this step, record this information in Table A-7 “Identify Client Discovery Method” in Appendix A: "Job Aids."

## Step Summary

In locations where the clients and Windows Deployment Services servers are separated by a router, a mechanism for discovering the servers must be determined. The method for Windows Deployment Services discovery should be recorded in Table A-7 in Appendix A so that it can be implemented at deployment.

## Additional Reading

“Microsoft Product Support Services (PSS) support boundaries for network booting Microsoft Windows Preinstallation Environment (Windows PE) 2.0”: <http://support.microsoft.com/kb/926172>

# Dependencies

A complete Windows Deployment Services installation requires the following infrastructure:

* Windows Deployment Services server that is a member of an AD DS domain. This can be provided by a domain controller in a local site or one that is in a remote location.
* An active and available DHCP server on the network. DHCP provides information for a PXE client to find the Windows Deployment Services server. DHCP is a critical component for clients to boot from the network to facilitate diagnostics and operating system deployments.
* An active DNS server on the network. DNS allows booting clients to resolve the names of Windows Deployment Services servers to IP addresses.
* An NTFS file system partition, either local or available through a remote share, for storing images.
* High network bandwidth for deploying images with a healthy and stable networking infrastructure.
* Internet Protocol version 4 (IPv4) is used on the network; networks running Internet Protocol version 6 (IPv6) only are not supported.

The following protocols are required to be usable on the network:

* Dynamic Host Configuration Protocol:
* UDP 67
* UDP 68, if Server Authorization is enabled
* Pre-Boot Execution Environment:
* UDP 4011
* Trivial File Transfer Protocol:
* UDP 69
* Random UDP Endport on client
* Remote Procedure Call:
* TCP 135 (Remote Procedure call [RPC] End Point Mapper Port)
* TCP 5040 (Default for Windows Deployment Services)
* Server Message Block:
* TCP 137 – TCP 139

# Conclusion

This guide summarizes the critical design decisions, activities, and tasks required to enable a successful design of a Windows Deployment Services 2008 infrastructure.

It addresses the technical aspects, service characteristics, and business requirements required to complete a comprehensive review of the decision-making process.

This guide, when used in conjunction with product documentation, allows organizations to confidently plan the implementation of Windows Deployment Services technologies.

## Additional Reading

In addition to the product documentation, the following web links contain supplemental information on product concepts, features, and capabilities addressed in this guide:

 For a newsgroup about Windows Deployment Services, see “Setup Deployment” on TechNet Forums/Windows Server 2008: <http://go.microsoft.com/fwlink/?LinkId=87628>

 Automated Installation Kit (AIK) for Windows Vista SPI and Windows Server 2008: <http://go.microsoft.com/fwlink/?LinkId=81030>

 Windows Automated Installation Kit (AIK): <http://go.microsoft.com/fwlink/?LinkID=53552>

 Windows Deployment Services Getting Started Guide: <http://go.microsoft.com/fwlink/?LinkID=84628>

 “Sysprep Technical Reference”: <http://go.microsoft.com/fwlink/?LinkId=87732>

## Feedback

Please direct questions and comments about this guide to satfdbk@microsoft.com.

We value your feedback on the usefulness of this guide. Please complete the following **Solution Accelerators Satisfaction Survey**, available at <http://go.microsoft.com/fwlink/?LinkID=132579>, and help us build better guidance and tools.

# Appendix A: Job Aids

Use the job aids that follow to record the information required for the organization to implement Windows Deployment Services.

Step 1. Use this job aid to record Windows Deployment Services instance locations and the number of instances required.

Table A-1. Instances Inventory

| Location | Number of instances required |
| --- | --- |
|   |   |
|   |   |
| Total number of instances required: |

Step 2. Use this job aid to record whether a new installation will be put into place or if an upgrade of existing infrastructure is to be used for each location.

Table A-2. New or Upgrade Installation Inventory

| Location | Upgrade or new? |
| --- | --- |
|   |   |
|   |   |

Step 3. Use this job aid to record for each instance whether a full Windows Deployment

Table A-3. Full Windows Deployment Services or Transport Server Role Inventory

| Instance | Full Windows Deployment Services or Transport Server role only? |
| --- | --- |
|   |   |
|   |   |

Step 4. Use this job aid to record the server resource requirements.

Table A-4. Instance Requirements

| Instance | Total number of computers | Image deployment speed | Size and number of images |
| --- | --- | --- | --- |
|   |   |   |   |
|   |   |   |   |

Step 4. Use this job aid to record the physical and virtual instance requirements.

Table A-5. Physical or Virtual Instance Requirements

| Instance | Physical/Virtual | WIM-based storage location | Server requirements | Number of servers needed |
| --- | --- | --- | --- | --- |
|  |  |  | CPU:Memory:Disk Size & Config:Network:  |  |
|  |  |  | CPU:Memory:Disk Size & Config:Network:  |  |

Step 5. Use this job aid to record the fault-tolerance and consistency method for each share used in the infrastructure.

Table A-6. Fault-Tolerance Method

| Share name/location | Fault-tolerance method | Consistency method |
| --- | --- | --- |
|   |   |   |
|   |   |   |

Step 6. Use this job aid to record the Windows Deployment Services client discovery method.

Table A-7. Identify Client Discovery Method

| Windows Deployment Services server name | Network boot referral or IP helper? | Devices needing configuration for discovery method |
| --- | --- | --- |
|   |   |   |
|   |   |   |

# Appendix B: Server Performance Analyzing and Scaling

The following information identifies important monitoring counters used for capacity planning and performance monitoring of a system.

## Processor Utilization

Over-committing CPU resources can adversely affect all the workloads on the same server, causing significant performance issues for a larger number of users. Because CPU resource use patterns can vary significantly, no single metric or unit can quantify total resource requirements. At the highest level, measurements can be taken to see how the processor is utilized within the entire system and whether threads are being delayed. The following table lists the performance counters for capturing the overall average processor utilization and the number of threads waiting in the processor Ready Queue over the measurement interval.

Table B-1. Performance Monitor Counters for Processor Utilization

| Object | Counter | Instance |
| --- | --- | --- |
| Processor | % Processor Time | \_Total |
| System | Processor Queue Length | N/A |

### Processor\Percent Processor Time

As a general rule, processors that are running for sustained periods at greater than 90 percent busy are running at their CPU capacity limits. Processors running regularly in the 75–90 percent range are near their capacity constraints and should be closely monitored. Processors regularly reporting 20 percent or less utilization can make good candidates for consolidation.

For response-oriented workloads, sustained periods of utilization above 80 percent should be investigated closely as this can affect the responsiveness of the system. For throughput-oriented workloads, extended periods of high utilization are seldom a concern, except as a capacity constraint.

Unique hardware factors in multiprocessor configurations and the use of Hyper-threaded logical processors raise difficult interpretation issues that are beyond the scope of this document. Additionally, comparing results between 32-bit and 64-bit versions of the processor are not as straightforward as comparing performance characteristics across like hardware and processor families. A discussion of these topics can be found in Chapter 6, “Advanced Performance Topics” in the *Microsoft Windows Server 2003 Performance Guide*.

### System\Processor Queue Length

The Processor Queue Length can be used to identify if processor contention, or high CPU-utilization, is caused by the processor capacity being insufficient to handle the workloads assigned to it. The Processor Queue Length shows the number of threads that are delayed in the processor Ready Queue and are waiting to be scheduled for execution. The value listed is the last observed value at the time the measurement was taken.

On a machine with a single processor, observations where the queue length is greater than 5 is a warning sign that there is frequently more work available than the processor can handle readily. When this number is greater than 10, then it is an extremely strong indicator that the processor is at capacity, particularly when coupled with high CPU utilization.

On systems with multiprocessors, divide the queue length by the number of physical processors. On a multiprocessor system configured using hard processor affinity (that is, processes are assigned to specific CPU cores), which have large values for the queue length, can indicate that the configuration is unbalanced.

Although Processor Queue Length typically is not used for capacity planning, it can be used to identify if systems within the environment are truly capable of running the loads or if additional processors or faster processors should be purchased for future servers.

## Memory Utilization

In order to sufficiently cover memory utilization on a server, both physical and virtual memory usage needs to be monitored. Low memory conditions can lead to performance problems, such as excessive paging when physical memory is low, to catastrophic failures, such as widespread application failures or system crashes when virtual memory becomes exhausted.

Table B-2. Performance Monitor Counters for Memory Utilization

| Object | Counter | Instance |
| --- | --- | --- |
| Memory | Pages/sec | N/A |
| Memory  | Available Mbytes | N/A |
| Memory | Pool Paged Bytes | N/A |
| Memory  | Pool Paged Resident Bytes | N/A |
| Memory  | Transition Faults/sec | N/A |
| Memory | Committed Bytes | N/A |
| Process | Working Set | <Process Name> |

### Memory\Pages/sec

As physical RAM becomes scarce, the virtual memory manager will free up RAM by transferring the information in a memory page to a cache on the disk. Excessive paging to disk might consume too much of the available disk bandwidth and slow down applications attempting to access their files on the same disk or disks. The Pages/Sec counter tracks the total paging rates, both read and writes, to disk.

For capacity planning, watch for upward trends in this counter. Excessive paging can usually be reduced by adding additional memory. Add memory when paging operations absorbs more than 20–50 percent of the total disk I/O bandwidth. Because disk bandwidth is finite, capacity used for paging operations is unavailable for application-oriented file operations. The Total Disk I/O Bandwidth is a ratio of the Pages/sec and the Physical Disk\Disk Transfers/sec for all disks in the system:

Memory\Pages/sec ÷ Physical Disk (\_Total)\Disk Transfers/sec

### Memory\Available Mbytes

The Available Mbytes displays the amount of physical memory, in megabytes, that is immediately available for allocation to a process or for system use. The percent Available Megabytes can be used to indicate if additional memory is required. Add memory if this value drops consistently below 10 percent. To calculate the percent of Available Megabytes:

(Memory\Available Mbytes ÷ System RAM in Megabytes) \* 100

This is the primary indicator to determine whether the supply of physical memory is ample. When memory is scarce, Pages/sec is a better indicator of memory contention. Downward trends can indicate a need for additional memory. Counters are available for Available Bytes and Available Kbytes

### Memory\Pool Paged Bytes and Memory\Pool Paged Resident Bytes

The Pool Paged Bytes is the size, in bytes, of the paged pool, an area of system memory used by the operating system for objects that can be written to disk when they are not being used.

The Pool Paged Resident Bytes is the size, in bytes, of the nonpaged pool, an area of system memory used by the operating system for objects that can never be written to disk, but must remain in physical memory as long as they are allocated.

A ratio of Pool Paged Bytes to Pool Paged Resident Bytes can be calculated by:

Memory\Pool Paged Bytes ÷ Memory\Pool Paged Resident Bytes

This ratio can be used as a memory contention index to help in planning capacity. As this approaches zero, additional memory needs to be added to the system to allow both the Nonpaged pool and Page pool to grow.

The size returned by the Pool Paged Resident Bytes can be used for planning additional TCP connections. Status information for each TCP connection is stored in the Nonpaged pool. By adding memory, additional space can be allocated to the Nonpaged pool to handle additional TCP connections.

### Memory\Transition Faults/sec

The Transition Faults counter returns the number of soft or transition faults during the sampling interval. Transition faults occur when a trimmed page on the Standby list is re-referenced. The page is then returned to the working set. It is important to note that the page was never saved to disk.

An upward trend is an indicator that there may be a developing memory shortage. High rates of transition faults on their own do not indicate a performance problem. However, if the Available Megabytes is at or near its minimum threshold value, usually 10 percent, then it indicates that the operating system has to work to maintain a pool of available pages.

### Memory\Committed Bytes

The Committed Bytes measures the amount of committed virtual memory. Committed memory is allocated memory that the system must reserve space for either in physical RAM or on the paging file so that this memory can be addressed by threads running in the associated process.

A memory contention index called the Committed Bytes:RAM can be calculated to aid in capacity planning and performance. When the ratio is greater than 1, virtual memory exceeds the size of RAM and some memory management will be necessary. As the Committed Bytes:RAM ratio grows above 1.5, paging to disk will usually increase up to a limit imposed by the bandwidth of the paging disks. Memory should be added when the ratio exceeds 1.5. The Committed Bytes:RAM is calculated by:

 Memory\Committed Bytes ÷ System RAM in Bytes

### Process\Working Set

The Working Set counter shows the amount of memory allocated to a given process that can be addressed without causing a page fault to occur. To see how much RAM is allocated overall across all process address spaces, use the \_Total instance of Working Set. Watch for upward trends for important applications.

Some server applications, such as IIS, Exchange Server, and SQL Server, manage their own process working sets. To measure their working sets, application-specific counters need to be used.

## Disk Storage Requirements

The process of planning for storage requirements is divided into capacity requirements and disk performance. Although a total capacity requirement can be determined, the performance requirements, as well as fault tolerance requirements, of the system will have an impact on the implementation of the storage subsystem. For example, a single drive could provide enough storage space, but the performance of that single disk may not meet the performance needs of the system.

Due to this, both capacity and performance requirements need to be met, which may alter the decision around the size, speed, and configuration of the drives in the storage subsystem.

### Disk Space Capacity

The amount of storage capacity required can be calculated based on OS requirements as well as any application-specific data that needs to be stored on the system.

### Disk Performance

Disk performance is typically expressed as a value of IOps, measured over some period of time during peak usage.

To determine the number of disks needed to meet a system’s IOps requirement, the IOps of a given drive needs to be determined. To further complicate matters, IOps are very dependent upon the access pattern. For example, a drive will typically have a higher IOps rating for sequential reads than it will for random writes. For this reason, it is normal to calculate a “worst case” IOps measurement based on short random input/output operations.

To calculate the IOps of a drive, information about the drive needs to be collected. The following table lists the information required that normally can be found in the manufacturer’s datasheet about the drive.

Table B-3. Information Required for Calculating IOps

| Required information | Description |
| --- | --- |
| Spindle Rotational Speed (RPM) | The spindle speed expressed as RPM. |
| Average Read Seek Time (ms) | The average seek time for reads. |
| Average Write Seek Time (ms) | The average seek time for writes. |

The first step to calculating the IOps is to determine the Average Seek Time in milliseconds that the drive is capable of doing. There is an assumption that there will be a 50/50 mix of read and write operations. If the ratio of read and write operations is modified, then the Average Seek Time will need to be adjusted. For example, if a drive has an Average Read of 4.7 ms and an Average Write of 5.3 ms, the Average Seek Time for this drive will be 5.0 ms:

5.0ms = (4.7ms + 5.3ms) ÷ 2

Next, the I/O Latency needs to be calculated. This is calculated by adding the Average Seek Time to the Average Latency. The following table lists the Average Latency of common spindle speeds of drives on the market today.

Table B-4. Average Latency Based on Spindle Rotational Speeds

| Spindle Rotational Speed (rpm) | Average Latency (ms) |
| --- | --- |
| 4,200 | 7.2 |
| 5,400 | 5.6 |
| 7,200 | 4.2 |
| 10,000 | 3.0 |
| 15,000 | 2.0 |

The example drive has a spindle speed of 10,000 RPM. So this drive has an I/O Latency of 8.0 ms:

8.0 ms = 5.0ms + 3.0ms

A drive can only perform one I/O operation at a time. To calculate the number of I/Os that can be performed in a millisecond, 1 is divided by the I/O Latency. Finally, this value is converted to I/O per Second by multiplying by 1000. The IOps calculation for the example drive evaluates to 125 IOps:

125 IOps = (1 I/O ÷ 8.0ms) \* 1000 ms/sec

## Storage Requirements

To determine storage requirements, additional information is needed to be collected around the system being considered. Some of this information is easily identified and self explanatory, while other information may be more difficult to identify due to lack of quantifiable data. All of the following calculations are for a single server; although if shared storage systems are being considered, then the information can be scaled up based on the number of systems sharing that storage. The following table shows the information that needs to be collected.

Table B-5. Information Required for Calculating Storage Requirements

| Required information | Description | Example |
| --- | --- | --- |
| # Users Per Server | Total number of users hosted by that server. | 700 |
| % Concurrent Users | The percentage of users connected to the server during peak times. | 80% |
| IOps per User Required | The number of IOps generated by a user. | 0.5 |
| Storage Capacity in Gigabytes | The calculated disk storage capacity needed. | 450 |
| % Buffer Factor (for growth) | The percentage of disk storage growth allowed by the system. | 20% |
| Read % of IOps | The percentage of IOps that are Read operations. | 50% |
| Write % of IOps | The percentage of IOps that are Write operations. | 50% |
| Disk Size (GB) | The drive size being considered for the storage system. | 146 |
| Calculated Drive IOps | The calculated IOps of the drive being considered for the storage system. | 125 |

The information in the table above is fairly self explanatory with the exception of the IOps per user required. This is a measurement of the number of IOps that a single user will generate on the system. Most venders do not have this information for their applications unless the application is extremely I/O intensive. This information may be calculated from observation of a running system, but it is fraught with a number of challenges and is beyond the scope of this guide. For the purpose of this example, this guide will use the value used with Exchange Server, which is 0.5 IOps per user.

Based on the information in Table B-5, there are a number of derived values that need to be calculated. The following table lists these values.

Table B-6. Derived Information Required for Calculating Storage Requirements

| Required information | Description | Example |
| --- | --- | --- |
| # of Concurrent Users | The number of concurrent users calculated by calculating the percentage of concurrent users and the number of users per server. | 560 |
| IOps per Server Required | The number of IOps generated by each user multiplied by the number of concurrent users. | 280 |
| Total Storage Requirements | The Storage Capacity increased by the percentage of growth allowed. | 540 |
| Number of Read IOps | The percentage of Reads and the IOps per Server Required. | 140 |
| Number of Write IOps | The percentage of Reads and the IOps per Server Required. | 140 |
| Drive Size Actual (GB) | After formatting, drives capacity is roughly 10 percent less than the advertised total capacity of the disk. This value adjusts for this loss of space. This value is calculated by taking 90 percent of the Disk Size (GB). | 132 |

RAID 0+1 and RAID 5 are the most common drive configurations for storing data in a redundant manner on a system. However, these two RAID systems have different IOps calculations due to how they operate.

### RAID 0+1 Calculations

To calculate the number of drives necessary to meet the storage requirements with RAID 0+1, divide the Total Storage Requirements by the Drive Size Actual. Round the result up and multiply by 2. For this example, 10 drives are needed for RAID 0+1 to meet the storage requirements:

10 = ROUNDUP(540÷132)\*2

To calculate the number of drives required to meet the performance requirements with RAID 0+1, multiply the Number of Write IOps by 2 and add the Number of Read IOps. Divide this total by the Calculated Drive IOps and round the result up. For this example, 4 drives are needed for RAID 0+1 to meet the performance requirements:

4 = ROUNDUP(((140\*2)+140)÷125)

Although RAID 0+1 can meet the performance requirements with just 4 disks, 10 disks are required to meet the capacity requirements.

### RAID 5 Calculations

To calculate the number of drives necessary to meet the storage requirements with RAID 5, the Total Storage Requirements needs to be multiplied by 1.2 to adjust for parity storage requirements. This value is then divided by the Drive Actual Size and then rounded up. For this example, 5 drives are needed for RAID 5 to meet the storage requirements:

5 = ROUNDUP((540\*1.2)÷132)

To calculate the number of drives required to meet the performance requirements with RAID 5, multiply the Number of Write IOps by 4 and add the Number of Read IOps. Divide this total by the Calculated Drive IOps and round the result up. For this example, 6 drives are needed for RAID 5 to meet the performance requirements:

6 = ROUNDUP(((140\*4)+140)÷125)

Although RAID 5 can meet the storage requirements with just 5 disks, 6 disks are required to meet the capacity requirements.

### RAID 0+1 versus RAID 5 Calculations

As can be seen in this example, RAID 5 looks to be the better choice when using 10K 146-GB drives. However, it is important to look at different types of drives when doing these calculations. For example, if a drive that has 300 GB is substituted for the 146-GB drives and if all other characteristics remain the same, then the choices drastically change.

Using 300-GB drives, RAID 0+1 requires just 4 drives to meet both capacity and performance characteristics. RAID 5 will require 3 drives to meet capacity but 6 drives to meet performance requirements. By changing the size of the drive, the best choice changed as well.

## Storage Monitoring

IOps are used to help characterize the performance requirements of the system. However, once the system is up, additional performance monitoring can be utilized to determine if the disk subsystem is slowing the system down.

Table B-7. Performance Monitor Counters for Disk Performance

| Object | Counter | Instance |
| --- | --- | --- |
| Physical Disk | % Idle Time | <All Instances> |
| Physical Disk | Disk Transfers/sec | <All Instances> |
| Physical Disk | Avg. Disk secs/Transfers | <All Instances> |
| Physical Disk | Split IO/sec | <All Instances> |

### Physical Disk\Percent Idle Time

The Percent Idle Time counter is the percent of time that a disk was idle during the sampling interval. An idle time of less than 20 percent indicates that the disk may be overloaded.

Physical Disk(n)\Disk utilization can be derived by subtracting the Physical Disk(n)\Percent Idle Time from 100 percent.

### Physical Disk\Disk Transfers/sec

The Disk Transfers/sec is the number I/O request Packets (IRP) that has been completed during the sampling interval. A disk can only handle one I/O operation at a time. So the number of physical disks attached to the computer serves as an upper bound on the sustainable disk I/O rate. Where disk arrays are used, divide the Disk Transfers/sec by the number of disks in the array to estimate individual disk I/O rates.

The Physical Disk(n)\Average Disk Service Time/Transfer can be calculated by taking the Physical Disk(n)\Disk Utilization and dividing it by the Physical Disk(n)\Disk Transfers/sec for each physical disk. This indicates how fast the drive responds to a request. If this climbs above what is specified for the disk, it can indicate that the subsystem is overloaded.

### Physical Disk\Avg. Disk secs/transfers

The Avg. Disk secs/transfer is the overall average of the response time for physical disk requests during the sampling interval. This includes both the time the request was serviced by the device and the time it spent waiting in the queue. If this climbs to over 15–25 disk IOps per disk, then the poor disk response time should be investigated.

The Physical Disk(n)\Average Disk Queue Time/Transfer can be calculated by taking the Physical Disk(n)\Avg. Disk secs/Transfer and subtracting the Physical Disk(n)\Avg. Disk Service Time/Transfer. The Average Disk Queue Time/Transfer indicates the amount of time a request spent waiting in the disk queue for servicing. A high queue time can indicate a poorly responding disk subsystem or specifically a poorly responding physical disk.

### Physical Disk\Split Input Output per Second

Split IOps is the rate physical disk requests were split into multiple disk requests during the sampling interval. A large number of split IOps indicates that the disk is fragmented and performance is being affected. The percentage of Split I/Os can be calculated by the following formula where “n” is a specific disk:

 (Physical Disk(n)\Split IO/sec ÷ Physical Disk(n)\Disk Transfers/sec) \* 100

If this percentage is greater than 10 to 20 percent, check to see whether the disk needs to be defragmented.

## Network Performance

Most workloads require access to production networks to ensure communication with other applications and services and to communicate with users. Network requirements include elements such as throughput—that is, the total amount of traffic that passes a given point on a network connection per unit of time.

Other network requirements include the presence of multiple network connections.

Table B-8. Performance Monitor Counters for Network Performance

| Object | Counter | Instance |
| --- | --- | --- |
| Network Interface | Bytes Total/sec | (Specific network adapters) |
| Network Interface | Current Bandwidth | (Specific network adapters) |
| IPv4 & IPv6 | Datagrams/sec | N/A |
| TCPv4 & TCPv6 | Connections Established | N/A |
| TCPv4 & TCPv6 | Segments Received/sec | N/A |

### Network Interface\Bytes Total/sec and Network Interface\Current Bandwidth

This Bytes Total/sec is the number of bytes transmitted and received over the specified interface per second The Current Bandwidth counter reflects the actual performance level of the network adaptor, not its rated capacity. If a gigabit network adapter card on a segment is forced to revert to a lower speed, the Current Bandwidth counter will reflect the shift from 1 Gps to 100 Mpbs.

Using these two values, the Network interface utilization can be calculated for each interface, designated as “n” with the following equation:

 (Network Interface(n)\Bytes Total/sec ÷ Network Interface(n)\Current Bandwidth) \*100

If Percent Busy for a given adapter exceeds 90 percent, then additional network resources will be necessary. Generally, the maximum achievable bandwidth on a switched link should be close to 90–95 percent of the Current Bandwidth counter.

### IPv4 & IPv6\Datagrams/sec

These counters show the total number of IP datagrams transmitted and received per second during the sampling interval. By generating a baseline around this counter, a trending and forecasting analysis of the network usage can be performed.

### TCPv4 & TCPv6\Connections Established

The Connections Established counter is the total number of TCP connections in the ESTABLISHED state at the end of the measurement interval. The number of TCP connections that can be established is constrained by the size of the Nonpaged pool. When the Nonpaged pool is depleted, no new connections can be established.

Trending and forecasting analysis on this counter can be done to ensure that the system is properly scaled to handle future growth. The server can be tuned using the TCP registry entries like *MaxHashTableSize* and *NumTcTablePartitions* based on the number of network users seen on average.

### TCPv4 & TCPv6\Segments Received/sec

The Segments Received/sec is the number of TCP segments received across established connections, averaged over the sampling interval. The average number of segments received per connection can be calculated. This can be used to forecast future load as the number of users grows. The following formula can be used to calculate the average number of segments received per connection:

TCPvn\Segments Received/sec ÷ TCPvn\Connections Established/sec

## Windows Server-Based File Servers

For Windows Server-based file servers, additional performance counters are available in addition to the ones mentioned previously, which can be used to monitor the performance of the system. The following table lists the most important counters.

Table B-9. Performance Monitor Counters for File Server Performance

| Object | Counter | Instance |
| --- | --- | --- |
| Server | Work Item Shortages | N/A |
| Server Work Queues | Available Threads | <All Instances> |
| Server Work Queues | Queue Length | <All Instances> |

### Server\Work Item Shortages

The Work Item Shortages counter indicates the number of times a shortage of work items caused the file server to reject a client request. This error usually results in session termination. This is the primary indicator that the File Server service is short on resources.

SMB requests are stored in a work item and assigned to an available worker thread. If there are not enough available threads, the work item is placed in the Available Work Items queue. If this queue becomes depleted, then the server can no longer process SMB requests.

### Server Work Queues\Available Threads

The Available Threads reports the number of threads from the per-processor Server Work Queue that are available to process incoming SMB requests. When the number of available threads reaches 0, incoming SMB requests must be queued. This is the primary indicator for identifying if the number of work threads defined for the per-processor Server Work queues is a potential bottleneck.

The MaxThreadsPerQueue registry DWORD value located at HKLM\SYSTEM\CurrentControlSet\Services\lanmanserver\parameters controls the number of threads that are created in the per-processor thread pools. By default, the system will create 10 threads per-processor thread pool. This value should be increased when the following conditions hold true:

* Available Threads is at or near zero for a sustained period.
* Queue Length of waiting requests is greater than 5.
* Percent Processor Time for the associated processor instance is less than 80.

By increasing the available threads, additional work items can be handled; however, care should be taken not to overstress the system with the addition of the extra threads.

### Server Work Queues\Queue Length

The Queue Length counter reports the number of incoming SMB requests that are queued for processing, waiting for a worker thread to become available. There are separate per-processor Server Work queues to minimize inter-processor communication delays.

This is a primary indicator to determine whether client SMB requests are delayed for processing at the file server. It can also indicate that the per-processor Work Item queue is backed up because of a shortage of threads or processing resources.

If any of these queues grows beyond 5, then the underlying reason for this growth should be investigated. File server clients whose sessions are terminated because of a work item shortage must restore their sessions manually.

# Version History

| **Version** | **Description** | **Date** |
| --- | --- | --- |
| 2.1 | Minor updates to this guide to reflect current IPD template | July 2010 |
| 2.0 | Added “What’s New in Windows Server 2008 R2” section. Minor updates to this guide to reflect current IPD template.Added throughout that VHD-based images are now supported.In “Out of Scope” section, updated references to Microsoft Deployment Toolkit.Additional reading references updated throughout.In Step 3, reflected new availability of PXE startup in Windows Server 2008 R2.In Step 4, Task 5, revised text around multicasting.In Step 5, corrected statement implying DFS-N was necessarily for DFS-R. | October 2009 |
| 1.0 | First release. | February 2008 |

# Acknowledgments

The Solution Accelerators–Management and Infrastructure (SA-MI) team acknowledges and thanks the people who produced the Infrastructure Planning and Design Guide for Windows Deployment Services. The following people were either directly responsible for or made a substantial contribution to the writing, development, and testing of this guide.

Contributors:

* Jude Chosnyk – *GrandMasters*
* Joe Coulombe – *Microsoft*
* Charles Denny – *Microsoft*
* Michael Kaczmarek – *Microsoft*
* Robin Maher – *Microsoft*
* Daniel Nerenberg – *Studio B*
* Fergus Stewart – *Microsoft*
* Melissa Stowe– *Microsoft*

Reviewers:

* Dave Field – *Studio B*
* Otto Helweg – *Microsoft*
* Lex Liao – *Microsoft*
* Greg Myers – *Studio B*
* Jez Sadler – *Microsoft*

Editors:

* Vernon Barr *– Volt Technical Services*
* Laurie Dunham – *Microsoft*
* Janet Majure – *Studio B*
* Lisa Pere – *Studio B*
* Pat Rytkonen– *Volt Technical Services*