

Infrastructure Planning   
and Design

Windows Server® 2008 and Windows Server 2008 R2

Active Directory® Domain Services

Version 2.2

Published: February 2008

Updated: November 2011

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

Copyright © 2011 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, Forefront, Outlook, SharePoint, Windows, Windows NT, Windows PowerShell, and Windows Server 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](#_Toc300222924)

[Introduction to Active Directory Domain Services Planning and Design 2](#_Toc300222925)

[Step 1: Determine the Number of Forests 6](#_Toc300222926)

[Step 2: Determine the Number of Domains 9](#_Toc300222927)

[Step 3: Assign Domain Names 12](#_Toc300222928)

[Step 4: Select the Forest Root Domain 14](#_Toc300222929)

[Step A1: Design the OU Structure 16](#_Toc300222930)

[Step B1: Determine Domain Controller Placement 18](#_Toc300222931)

[Step B2: Determine the Number of Domain Controllers 20](#_Toc300222932)

[Step B3: Determine Global Catalog Placement 22](#_Toc300222933)

[Step B4: Determine Operations Master Role Placement 26](#_Toc300222934)

[Step C1: Create the Site Design 28](#_Toc300222935)

[Step C2: Create the Site Link Design 30](#_Toc300222936)

[Step C3: Create the Site Link Bridge Design 33](#_Toc300222937)

[Step D1: Determine Domain Controller Configuration 35](#_Toc300222938)

[Conclusion 38](#_Toc300222939)

[Appendix A: Design Job Aids 39](#_Toc300222940)

[Appendix B: IPD in Microsoft Operations Framework 4.0 41](#_Toc300222941)

[Appendix C: Active Directory Domain Services in Microsoft Infrastructure Optimization 42](#_Toc300222942)

[Version History 43](#_Toc300222943)

[Acknowledgments 44](#_Toc300222944)

# 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. It is assumed that the reader has a basic understanding of the technologies discussed in these guides. It is the intent of these guides to define business requirements, then align those business requirements to product capabilities, and design the appropriate infrastructure.

## 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 Active Directory® Domain 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 Active Directory Domain Services Planning and Design

Active Directory Domain Services (AD DS) controls the core security of the Windows® network environment. The directory service is responsible for authenticating user and computer accounts within the AD DS infrastructure. In addition, the directory service provides a mechanism for centralized, delegated administration of resources within the forest.

To develop and implement a successful design of AD DS, numerous questions must be answered and many decisions and strategies must be determined. Considerations for performance, security, manageability, scalability, and many other criteria must be addressed if the design is to be successful.

The purpose of this guide is to assist designers in the decision-making process by providing a clear and concise path for designing the AD DS infrastructure, given the relative context. This guide relies on best practices and real-world experience to offer considerations and alternatives at each point in the design.

This guide, when used in conjunction with product documentation, will help companies confidently plan an AD DS implementation. Appendix A, “Design Job Aids,” includes a sample job aid for recording the decisions made during the design process.

## 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.

AD DS in the Windows Server 2008 R2 operating system includes many new features that help improve AD DS manageability, supportability, and performance, including the following:

* Active Directory Recycle Bin
* Active Directory module for Windows PowerShell® and Windows PowerShell cmdlets
* Active Directory Administrative Center
* Active Directory Best Practices Analyzer
* Active Directory Web Services
* Authentication mechanism assurance
* Offline domain join
* Managed Service Accounts
* Active Directory Management Pack

For more details on the changes, see <http://technet.microsoft.com/en-us/library/dd378796.aspx>.

## Assumptions

To limit the scope of material in this guide, the following assumptions have been made:

* The decision to implement AD DS has already been made. This guide does not address the business or technical case to make a directory choice.
* 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 in configuration.
* The reader has familiarity with the Microsoft infrastructure and directory services. This guide does not attempt to educate the reader on the features and capabilities of Microsoft products. The product documentation covers that information.

## Active Directory Domain Services Design Process

This guide focuses on addressing the critical design decisions faced by most organizations when implementing Active Directory Domain Services (AD DS) in Windows Server 2008 or Windows Server 2008 R2.

This guide’s goal is to address the most common scenarios, decisions, activities, options, tasks, and outcomes that most organizations will encountered. It does not attempt to address every possible scenario or permutation of a scenario. Readers who think their situation is unique should consider hiring a design consultant to address their needs.

This guide addresses the following decisions and/or activities that need to occur in preparing for AD DS planning. The following 13 steps represent the most critical design elements in a well-planned AD DS implementation:

* Step 1: Determine the Number of Forests
* Step 2: Determine the Number of Domains
* Step 3: Assign Domain Names
* Step 4: Select the Forest Root Domain
* Step A1: Design the OU Structure
* Step B1: Determine Domain Controller Placement
* Step B2: Determine the Number of Domain Controllers
* Step B3: Determine Global Catalog Placement
* Step B4: Determine Operations Master Role Placement
* Step C1: Create the Site Design
* Step C2: Create the Site Link Design
* Step C3: Create the Site Link Bridge Design
* Step D1: Determine Domain Controller Configuration

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

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

In many cases, the sequence in which the decisions are made or the tasks are accomplished is significant to the design process. The critical path of the design process is the path that orders decisions in series, as one task must be completed before another task starts.

The critical path for AD DS design is illustrated in the flow chart in Figure 1. For the purposes of this document, the steps will be performed in a sequential path, moving from top to bottom of the diagram. Some process flows in this path can be performed either in parallel or sequentially in any order. For example, both A and B must be completed; however, they can be performed at the same time, A can be performed before B, or vice versa.

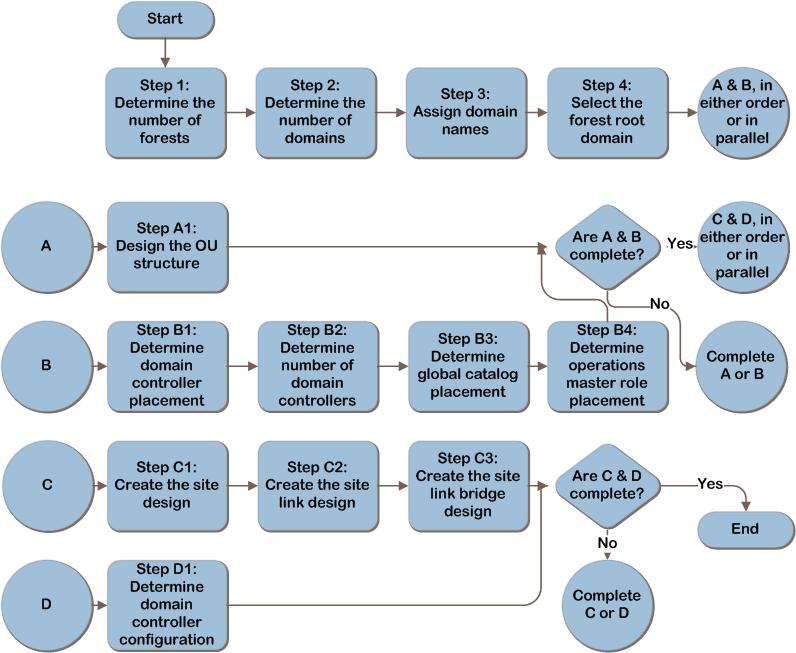


Figure 1. Critical path and process flow for AD DS design

## Information Collection

Various types of information will be needed during the planning process. The following information is required for designing the AD DS infrastructure.

* **Needed for designing the OU structure of each domain (A1)**
* The current administrative model used in the organization. This lists who is responsible for managing the resources of the environment. Another way of looking at it would be to ask “**Who** does **what** to **whom**?”
* Group Policy deployment requirements and strategies
* **Needed for domain controller placement (B1)**
* The number of users per physical location (for example, corporate office, branch office, satellite office)
* The number of computers per physical location
* **Needed for creating a site design (C1)**
* Physical location map
* Network link speeds and available bandwidth between locations
* TCP/IP subnets used in each physical location
* Domains represented in each physical location
* Domain controllers (per domain) in each physical location
* **Needed for creating a site link design (C2):** Replication convergence goals for the following:
* Configuration and Schema
* Domain
* Global Catalog
* Application Partitions

## Applicable Scenarios

This guide addresses considerations that are related to planning and designing the necessary components for a successful AD DS infrastructure:

* Production corporate intranets
* Centralized facilities (hub locations)
* Branch offices (satellite locations)
* National networks
* Global networks

## Out of Scope

This document is designed to guide the architect through the process of designing the core implementation for AD DS. Its scope has therefore been limited so that it does not cover the following areas:

* Migration from implementations earlier than Windows Server 2008. There are, however, some design considerations involving Windows 2000 server components.

When migrating, it is still important to validate the current design, and this guide will assist with that process.

* Migration from, coexistence with, or interoperation with third-party directory services.
* Active Directory Lightweight Directory Service (AD LDS), formerly known as Active Directory Application Mode (ADAM), is a lightweight implementation of AD DS, sometimes set up for use by individual applications.
* Federated implementations in which multiple corporations are joined together.
* Multi-tenant considerations in which multiple companies are hosted within a forest.

# Step 1: Determine the Number of Forests

Every AD DS implementation will have at least one forest. The first step in AD DS design is to determine whether one or multiple forests are required to meet the organization’s objectives. If multiple forests are required, then the total number of forests needs to be determined.

Getting this decision correct in the beginning is important. As planning progresses, the assumptions that are driven by this design decision will make changing the configuration more difficult. It is considerably more difficult to collapse forests once they have been established than it is to add additional forests later.

## Option 1: Single Forest

When considering the overall design of AD DS, a single forest implementation is the default.

A best practice is to start with a single forest and let business requirements justify any additional forests.

For extremely large directories, replication could become an issue. Whereas domains are used to partition the directory data and control replication of domain-centric information, forest-wide information—which includes configuration data, schema, and global catalog data—must be replicated.

## Option 2: Multiple Forests

The following requirements will dictate a design with multiple forests:

* **Multiple schemas.** Everything in the forest shares a common schema. Conflicts between applications or administration of the schema can introduce the need for an additional forest.
* **Resource forests.** Some organizations may require multiple forests for isolation reasons, but need to share a common resource, for example Microsoft Exchange Server 2000 and later. A separate forest can be created to host the shared resources, and forest-level trusts can be used to provide the authentication and authorization paths. A test environment could be created as a resource forest.
* **Forest administrator distrust.** Some organizations have an internal structure that includes more than one IT team. When each IT team wants to control the forest while denying the other IT staff control, implementing multiple forests are means to that end. This is a common scenario when companies merge, in government agencies, and at universities.
* **Legal regulations or geo-political reasons for application and/or data access.** All domains in a single forest have automatic, two-way Kerberos trusts so that data and applications can be accessed easily. When working with some countries or regions, legal requirements may dictate the separation of data and applications. Multiple forests provide this separation.

Implementing multiple forests increases the cost of managing the environment. Additional hardware and software are required to maintain and support multiple forests, and additional staff may also be required.

If information sharing across forests is required, then cross-forest trusts are necessary. These trusts support Kerberos in Windows Server 2003 and Windows Server 2008 environments.

Global catalogs do not replicate across forest boundaries. To obtain a unified view across multiple forests, directory synchronization software, such as [Identity Lifecycle Manager 2007](http://www.microsoft.com/windowsserver/ilm2007/default.mspx) or [Forefront® Identity Manager 2010](http://www.microsoft.com/forefront/identitymanager/en/us/default.aspx) must be implemented. Implementing such technologies increases the administrative burden of multiple forests.

### How Many Forests?

When the need for multiple forests is confirmed, the exact number of required forests must be determined. Iterate through the forest decision until all of the business requirements have been addressed and the total number of forests required has been identified.

## Evaluating the Characteristics

This table can assist in determining the complexity, cost, and security relative to results of decisions made in this task.

Table 1. Evaluate the Characteristics of Required Forests

|  |  |  |
| --- | --- | --- |
| Complexity | | |
| One forest | A single forest is the entry point for a deployment of AD DS; complexity cannot be reduced. | Low |
| Multiple forests | Second and subsequent forests add to the overall complexity of the environment. | High |
| Cost | | |
| One forest | A single forest is the most inexpensive choice because it requires less hardware, software, and administrative support. | Low |
| Multiple forests | Hardware, software, and administrative considerations increase the cost for each forest that is added to the design. | High |
| Security | | |
| One forest | The forest is the security boundary, and the forest administrator has access to all resources within the forest. | → |
| Multiple forests | Security responsibilities are granted to the administrator of each forest. The division of security responsibilities among multiple administrators could be a better overall rating for security. | **↑** |

## 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 forest design decisions:

* **Are there any acquisition or divestiture plans in the near future?** If the company might be acquired in the near future, it may be prudent to discuss design details with the acquiring company, rather than design a directory that could be discarded once the acquisition is complete.

If the company is acquiring a new business, requirements around that acquisition should be considered during the design phase. For example, unique administration requirements might be introduced during the acquisition.

If a business unit is going to be divested, a separate forest might make the transition easier and simpler.

* **Are there any impending separation requirements?** Pending or known compliance regulations might introduce separation requirements.

## Tasks and Considerations

For each forest in the environment, it’s important to consider time synchronization. Kerberos depends on the time of domain controllers, servers, and clients being synchronized within minutes of one another; otherwise, Kerberos authentication will fail. Time is one of the considerations used for assessing the health state of the directory. Active Directory relies on the domain controller that runs the primary domain controller (PDC) emulator role in the root domain to keep the *master time* for all domains in the forest. There are two options for establishing the time for that domain controller.

The time can be set to synchronize with either an internal source or an external source to the organization. If an internal source is used, it can be synchronized with a time server that is on the Internet. Also, the time source and domain controller can use authentication to ensure a reliable time. If an external time source is used, no authentication is provided.

Manually setting and updating the time is not recommended. The AD DS environment relies too heavily on the time, and serious problems can occur if the time is not set properly.

## Step Summary

A single forest is ideal. It is easier to manage as well as being cheaper to implement, maintain, and support. Multiple forests are necessary if legal, schema, administrative, or application requirements dictate the decision.

## Additional Reading

* Creating a Forest Design: <http://technet.microsoft.com/en-us/library/cc753859(WS.10).aspx>
* Knowledge Base article 816042 “How to configure an authoritative time server in Windows Server”: <http://support.microsoft.com/kb/816042/>

# Step 2: Determine the Number of Domains

The second step in AD DS design is to determine the number of domains that are required to meet the organization’s objectives. Because each forest is unique and separated from the other forests, the number of domains in each forest must be considered independent of the other forests.

The addition or removal of a domain after the initial design has been implemented is not always simple. Migration of computers, users, data, and applications could make the modification to the number of domains a complex task.

## Option 1: Single Domain

The design will need to have at least one domain. If there are multiple forests, then there will need to be one domain per forest, minimum. A single domain model has the following advantages and benefits:

* A single domain is the least expensive option. Additional domains increase the cost of hardware, software, and administration.
* A single domain is easier to manage. Management overhead and the related costs increase with additional domains.
* A single domain is easier to recover in the event of a disaster.

## Option 2: Multiple Domains

Any of the following requirements will lead to a design with multiple domains:

* In environments that consist of a combined total of 100,000 user or computer objects, tests should be performed in the lab to ensure that the replication load does not overwhelm the replication topology for the domain. Multiple domains may be required to reduce the overall domain replication load.
* If AD DS has a large number of frequently changing attributes, it may be useful to break the environment into multiple domains to control the replication within the domains. Testing should be done in a lab to determine if multiple domains reduce the replication traffic in a significant way.
* The compression algorithm used to replicate directory service changes across slow links is highly efficient. However, if slow links still cause issues for replication, a separate domain might be necessary. This scenario can be challenging when there are numerous changes occurring to directory service objects on a regular basis.
* An existing Microsoft directory, running on an earlier operating system level, needs to be preserved. To do so, the environment can be separated into its own domain.

Note   Windows Server 2008 supports fine-grained password policies. This new technology supports multiple password policies in the same domain. Windows 2000 or Windows Server 2003 domains will support only one password policy per domain.

By choosing to have multiple domains, the cost of managing the environment is affected in the following ways:

* Additional staff may be necessary to maintain the domains, each of which will have its own administrator group.
* More staff might be necessary to manage multiple domains, which involves a more complex set of management requirements.
* Additional hardware and software must be acquired to instantiate the domain.
* Group Policy settings that need to be applied to the domain or OUs in domains across the forest will need to be applied separately in each domain.

### How Many Domains?

Once the need for multiple domains has been identified, the exact number of domains per forest is determined. A separate domain will be added to address each of the considerations that have been identified.

## Evaluating the Characteristics

This table can assist in assessing the complexity and cost of one domain versus multiple domains.

Table 2. Evaluate the Characteristics of Required Domains

|  |  |  |
| --- | --- | --- |
| Complexity | | |
| One domain | A single-domain directory is the least complex environment. | Low |
| Multiple domains | Complexity increases with the addition of each domain. However, just adding another domain does not add as much complexity as it does for cost and manageability. | High |
| Cost | | |
| One domain | The cost to set up and operate a single domain is the lowest possible. | Low |
| Multiple domains | Setup costs rise with each additional domain because of the requirements of installing and configuring each domain controller, not to mention the hardware and software cost for each domain controller. | High |

## 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 domain design decisions:

* **Is there a need to separate a business unit because of legal requirements?** Some companies and many governmental, university, or military environments require that some users and computers exist in a separate domain. If such a policy exists, it should be re-evaluated as the domain is no longer the security boundary it was in Windows NT® 4.0 and previous versions. If the policy is around isolation requirements, a separate forest will be required.
* **Are there different administrative units that need to be autonomous?** In most cases, using delegation at the OU level within a single domain can provide autonomy to administrative units. However, politics, corporate structure, administrative controls, and other factors might cause a need for additional domains instead.

## Step Summary

A single domain is the default configuration for each forest. Add domains only as necessary to solve technical and business concerns that can’t be solved within a single domain. Additional domains cost more to install and increase the hardware and software needed to run the domain controllers in each domain.

Remember to record the decisions made in the job aid in Appendix A of this guide.

**Important!**   The number of domains will need to be determined per forest.

## Additional Reading

* Creating a Domain Design: <http://technet.microsoft.com/en-us/library/cc754645(WS.10).aspx>
* AD DS: Fine-Grained Password Policies: <http://technet2.microsoft.com/windowsserver2008/en/library/056a73ef-5c9e-44d7-acc1-4f0bade6cd751033.mspx?mfr=true>

# Step 3: Assign Domain Names

The third step in AD DS design is to assign names to each of the domains. There are two names to assign: the Domain Name System (DNS) name and the Network Basic Input/Output System (NetBIOS) name. Although Windows Server 2008 uses DNS for name resolution instead of the Windows Internet Name Service (WINS) NetBIOS name resolution method that is used in Windows NT 4.0–based networks, most organizations still require WINS since there are applications that require it.

## Task 1: Assign the NetBIOS Name

The NetBIOS naming activity does not lend itself to a list of specific options. However, there are considerations that need to be addressed when designing the namespace.

NetBIOS names are the names that users most often see when prompted for a domain name in Windows. Each domain requires a NetBIOS name to be assigned. The NetBIOS name must be unique on the network or name resolution conflicts will result.

The same NetBIOS name can be used in different corporations to represent the same entity; for example, the NetBIOS name *CORP* is often used as the name for the internal corporate network. In this case, if two companies merge and both have used the NetBIOS name *CORP,* there will be a conflict when the two networks are integrated.

Name resolution conflicts can be avoided by using a NetBIOS name that is more likely to be unique across corporations, such as *CONTOSOCORP* for a corporation named Contoso. Use a name that will be unique and independent of existing regional or organizational names within the corporation.

## Task 2: Assign the DNS Name

Similar to NetBIOS naming, the DNS naming activity does not lend itself to a list of specific options. However, there are considerations that need to be addressed when designing the namespace.

As with NetBIOS, several rules apply to DNS naming. The “Additional Reading” section later in this section provides links to resources about these rules.

The DNS names of AD DS domains include two parts: a host name and a network name. When concatenated, these names create a non-ambiguous name for a resource. The host name is the name of the AD DS domain.

First, determine the network name. A best practice is to match the registered Internet domain name for the business. Doing so will ensure that the name is unique across the Internet and is not in conflict with other corporations’ external name ownerships; this reduces the risk of name conflicts during mergers and acquisitions.

Second, select the host name for the domain. The default naming scheme will make the NetBIOS name and DNS the same, such as the NetBIOS name *CONTOSOCORP* and the DNS name *contosocorp.com*. For ease of tracking DNS names and NetBIOS names within interfaces and when troubleshooting network-related or AD DS–related issues, it is a good idea to keep these names the same. However, doing so is not required.

To ensure uniqueness among companies, don’t duplicate existing corporations’ registered Internet DNS domain names. It is a best practice to register all top-level domain names (also known as network names), which are being used, both internally and externally, with Internet Network Information Center (InterNIC) to ensure global DNS uniqueness of the name on the Internet.

## 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 domain naming decisions:

* **We plan to use our current Internet domain name. Are there any groups or applications that require a different DNS namespace, perhaps for identity reasons?** A separate domain would be required to support a different DNS name within the forest.
* **Are there any planned mergers or acquisitions?** Changes in the corporate structure can affect the naming structure.

## Step Summary

Domain names should be kept simple and should be consistent with the Internet DNS namespace. Both the NetBIOS name and DNS name need to be considered for consistency, manageability, and complexity.

Once implemented, this decision is difficult to change because all computers, applications, and scripts would need to be updated to represent the new name. Also, users are exposed to the domain name at logon and when using domain-related applications. A change to the domain name would be confusing and disruptive.

**Important!**   Run through the decision process to determine the name of every domain in every forest and record the decision in the job aid in Appendix A.

## Additional Reading

* Knowledge Base article 909264 “Naming conventions in Active Directory for computers, domains, sites, and OUs”: <http://support.microsoft.com/kb/909264>
* Namespace planning for DNS: <http://technet.microsoft.com/en-us/library/cc759036(WS.10).aspx>

# Step 4: Select the Forest Root Domain

The first domain deployed in an AD DS forest is called the forest root domain. This domain remains the forest root domain for the life cycle of the AD DS deployment. It cannot be changed without redeploying the entire forest.

The forest root domain contains the Enterprise Admins and Schema Admins groups. These administrator groups are used to manage forest-level operations, such as the addition and removal of domains and changes to the schema.

A domain that exists in the design can be selected as the forest root, or a dedicated forest root can be selected. Once the forest root domain has been established, it cannot be changed without rebuilding the forest.

## Option 1: Use a Planned Domain

When the domain design for a forest indicates a single domain, then this single domain is the forest root domain. This one domain will host all users, groups, computers, and the forest root groups.

If multiple domains exist in the design, one of the domains can be selected to be the forest root domain in addition to managing the users and resources of the domain. The selected domain will define the forest namespace and will need to be the first domain deployed in the environment. Although it will also manage users and resources, it will always maintain its unique status as the domain containing the Enterprise Admins and Schema Admins groups.

## Option 2: Dedicated Forest Root Domain

A dedicated forest root domain, also known as an empty forest root, may be added to the existing domain structure to specifically manage the forest level functions. When selected, this domain does not contain any user accounts or resources other than the service administrator accounts for the forest root domain, and it does not represent any region in the domain structure. All domains become children of this domain.

A dedicated forest root is generally chosen for the following reasons:

* Operational separation of forest service administrators from domain service administrators.
* Protection from operational changes in other domains.
* Serves as a neutral root so that no region appears to be subordinate to another region.

Note that a dedicated forest root presents the following disadvantages:

* Additional overhead is involved in monitoring and maintenance of the separate domain controllers. The dedicated root forest domain controllers must be available in order to allow users to gain access to resources in other domains.
* Additional hops are inserted in the trust path to domain controllers in order to allow users to gain access to resources in other domains, unless shortcut trusts are implemented.

It should be noted, however, that the forest level functions are not protected from a rogue administrator manipulating the AD DS database in such a way as to compromise the integrity and security of the directory. This means that while an empty forest root may separate functional administrative groups, it does not grant any additional security to the forest from rogue administrators.

## Evaluating the Characteristics

This table can assist in assessing the cost of using a planned domain versus empty root domain.

Table 3. Evaluate the Characteristics of Required Forest Root Domain

|  |  |  |
| --- | --- | --- |
| Cost | | |
| Use a planned domain | No additional costs are required as a planned domain is being used as the forest root. | Low |
| Empty root domain | Dedicating an empty root domain to host the forest root will incur extra hardware and software costs for the computers to run the domain and maintain its availability. | High |

## 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 question has been known to affect forest root placement decisions:

* **Are any mergers or acquisitions planned?** Changes in the corporate structure could affect the placement of a forest root.

## Step Summary

The identity of the forest root domain has been determined at this point. Either a planned domain has been chosen or a new domain has been added to the design as the forest root.

# Step A1: Design the OU Structure

Objects in the directory are organized by using organizational units (OUs). The design for the OUs will have two primary factors: the delegation of the administration of directory objects and the application of Group Policy objects (GPOs). Fundamentally, the OU design should be a reflection of how the objects in the domain are managed.

Changing the OU design is not difficult, but it can be complex since access control lists need to be carefully manipulated. Once delegation and Group Policy have been established, redesigning the OUs to which the configurations have been applied will take time.

Since OUs serve the dual roles of administration delegation and the application of Group Policy, it will be necessary to go through the design process for OUs twice: once for delegation and then a second time with an eye toward Group Policy usage.

## Task 1: Design OU Configuration for Delegation of Administration

OUs can be used to delegate the administration of objects, such as users or computers, to a designated group. Although it is possible to delegate permissions to an individual, it is a best practice to use groups because as people change in the organization, it is easier to update membership in the delegation groups than to update the permissions on objects in the directory. Delegation by means of an OU involves the following tasks:

* Identify or create administrative groups to which rights will be delegated.
* Place the individuals or groups to which rights will be delegated into the OU. Create the OUs to which the administrative groups will have authority.
* Assign the object rights to be delegated to the administrative group within each OU.
* Create/place the objects to be controlled within the OU.

When identifying the groups to which administrative tasks will be delegated, try to be as specific as possible about the minimum amount of control that is required. For example, if a group needs just the ability to update users’ telephone information, the group should not be granted full control.

## Task 2: Design OU Configuration for Group Policy Application

OUs can be created to apply Group Policy settings to a specific subset of computers or users. By default, all objects in an OU will receive the settings contained in an applied GPO.

With the OU design complete from a delegation (or operations) perspective, the next step is to revise the OU design to account for any unique circumstances that Group Policy settings may introduce. For example, from a delegation perspective, an OU may be established called “Workstations” to delegate permissions to manage all workstations. When Group Policy considerations are applied, there may be a need for a desktop OU and a mobile OU to reflect the different policy needs for desktops and notebooks. In this case, these desktop and mobile OUs may be created as sub-OUs inside the workstations or OU, or the Workstations OU may be replaced by these two individual OUs.

Identify groups of users or machines to which a GPO needs to be applied. Then, examine the current OU design for the domain. Re-use existing OUs if possible and create new OUs only if necessary. If new OUs are created to support GPOs, then make sure to review the object delegation in the previous task to ensure that the object administration and operation model is up to date.

There are many filtering and targeting options for Group Policy application. Security filtering, Windows Management Instrumentation (WMI) filtering, and Group Policy preference targeting can all be used to target which objects receive which GPOs. Use these techniques as a last resort in lieu of using the default Group Policy application and precedence. Filtered Group Policy security is very difficult to troubleshoot and manage and can cause a slight performance degradation for client logons.

## Step Summary

The OU structure needs to be defined for each domain in the design. At the end of this decision, the OU design should have identified the following:

* OUs to be created, based on one of two design criteria: delegation of administration or Group Policy application.
* Which objects need to be located in each OU.
* Administration (Delegation) groups to be created and mapped to OUs.
* Object rights to be granted to each group in each OU.
* Which GPOs need to be created and to which OUs they should be linked.

## Additional Reading

Best Practice Active Directory Design for Managing Windows Networks: [www.microsoft.com/technet/prodtechnol/windows2000serv/technologies/activedirectory/plan/bpaddsgn.mspx](http://www.microsoft.com/technet/prodtechnol/windows2000serv/technologies/activedirectory/plan/bpaddsgn.mspx)

# Step B1: Determine Domain Controller Placement

Typically, a network topology will have a few physical locations (large campuses or office buildings and data centers) that are considered hubs, in which there are concentrations of users, computers, or network connectivity. These hubs may connect to a number of smaller satellite locations, such as branch or home offices, to which the hubs provide network or computing resources. Satellite locations typically do not provide services to other satellite offices.

In this step, decide where domain controller resources will be placed for each domain in each forest. Step B2 will address how many domain controllers to place in each location for each domain.

In order to reduce cost and complexity and to increase manageability, it is better to place domain controllers in as few locations as possible and where they will have the best utilization and highest value impact for the organization.

One additional consideration is that each domain should have a domain controller in at least two geographically dispersed locations to allow for business continuity in the event that one location experiences a catastrophic event.

All domain controllers need to be physically secured. If physical security is not available at a location, a full domain controller should not be placed at that location; however, a Read-Only Domain Controller could be placed in a location where physical security is a concern. See “Task 2: Determine Type of Domain Controller Placed in Location” in the next section, “Determine Number of Domain Controllers.”

The decision about domain controller placement can be changed easily at any time.

## Task 1: Hub Locations

Hub locations provide computing and networking services to many users within the organization. Hub locations may provide these resources to users in the hub, as well as to one or more satellite locations.

Since hub locations are a central point, they are ideal candidates for having the highest value impact. A hub may be the aggregation point for several satellite locations, so having the hub host the domain controller is less expensive than having each individual location host its own domain controller.

Determine which hub locations will host domain controllers for which domains and record the decisions in the job aid.

## Task 2: Satellite Locations

Satellite locations are connected to the overall network through hubs. In most cases, a satellite location has fewer users and computers than a hub. The clients in a satellite location can use resources locally, can use resources in the hub, or can use the hub to access network resources located in other parts of the network. Several considerations can indicate the need to place a domain controller in a satellite site.

Domain controllers need to be managed. Place a domain controller in a particular location only if the domain controller can be managed locally or managed remotely by use of a secure connection.

Communication with a domain controller is essential to authentication when accessing network resources. Therefore, if the WAN link from the satellite office to the hub is unreliable and cannot be cost-effectively updated, consider placing a domain controller in the satellite office to accommodate client authentication.

Another factor for considering the placement of a domain controller in a satellite office is whether WAN link bandwidth is available for both routine network traffic and authentication. In some cases, satellite-office network traffic over a WAN link might need to use the majority of the available bandwidth for an application or service. In that case, a local domain controller in the satellite office might be necessary.

Another consideration for placing a domain controller in a satellite office is to accommodate services and resources that might reside in the satellite office. Services such as DNS and Distributed File System (DFS), as well as resources such as mail and databases, could benefit from having a domain controller on the same network instead of having to cross a WAN link for authentication and management of the directory.

Site autonomy is sometimes a reason for placing domain controllers in a location. For example, if a company has a manufacturing facility in a remote location and the equipment on the manufacturing line requires authentication to work, then placing a domain controller in this satellite location allows manufacturing to continue regardless of whether or not the WAN is available.

Determine which satellite locations will host domain controllers for which domains, and record the decisions in the job aid.

## 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 question has been known to affect domain controller placement decisions:

* **Are there any users who travel frequently to satellite locations and who require high-performance logon and directory services in those locations?** A domain controller may be required in some satellite locations to provide a local experience for traveling users.

## Tasks and Considerations

When placing domain controllers in hubs and satellites, it may be necessary to control which domain controllers register site location records within DNS. For example, if a domain controller fails in a satellite site, the clients should contact a domain controller in the nearest hub rather than a domain controller located in another satellite site. This is particularly true if the WAN link between the two is not reliable. Identify if this is a concern for the planned environment. Additional information can be found in the “Additional Reading” section.

## Step Summary

Place domain controllers in hub and satellite locations when appropriate. Most hub locations require one or more domain controllers. Satellite offices might require a domain controller depending on WAN link characteristics, number of clients, and resources.

Remember to repeat this decision process for every domain in every forest.

## Additional Reading

* Planning Domain Controller Placement: <http://technet.microsoft.com/en-us/library/cc754920(WS.10).aspx>
* Knowledge Base article 247811 “How Domain Controllers are Located in Windows”: <http://support.microsoft.com/kb/247811/>
* Knowledge Base article 306602 “How to optimize the location of a domain controller or global catalog that resides outside of a client’s site”: <http://support.microsoft.com/?id=306602>

# Step B2: Determine the Number of Domain Controllers

A previous section of this guide addressed the need to determine the physical placement of domain controllers. A related decision is to determine the number of domain controllers for each location.

There are many deciding factors around how many domain controllers to have for each domain. Decisions are based on performance of authentications, access to resources, replication, and cost.

## Task 1: Determine Number of Domain Controllers

For each domain in each location identified in Step B1, the minimum number of domain controllers required needs to be identified. The table below describes the minimum number of domain controllers required, based on number of users.

Table 4. Minimum Number of Domain Controllers

|  |  |
| --- | --- |
| User per domain in a site | Minimum number of domain controllers required per domain in a site |
| 1–499 | One – Single Processor |
| 500–999 | One – Dual Processors/Cores |
| 1,000–2,999 | Two – Dual Processors/Cores |
| 3,000–10,000 | Two – Quad Processors/Cores |

For workloads greater than 10,000 users in a site, additional testing should be performed with user workloads to determine the need for additional hardware. Previous guidance stated an extra quad processor system for every additional 5,000 users. However, for authentication-only workloads, this will be overkill for most environments.

If only one domain controller per location exists, consideration should be made for the need to span the WAN to communicate with a domain controller for authentication and access to resources in the event of failure of the local domain controller.

All domain controllers within a domain must be fully aware of all information related to the domain. This is handled by replication of the AD DS database between domain controllers. This replication occurs within AD DS sites and across site boundaries. If the number of replication partners in a given site reaches 15 or more, an additional domain controller should be added to the site. Another domain controller should be added for each additional 15 replication partners.

Review all applications that rely on AD DS data. Some applications, such as Exchange Server, require additional domain controllers in order to function correctly. Evaluate the need for additional domain controllers based on the expected loads and requirements of the applications.

## Task 2: Determine Type of Domain Controller Placed in Location

For each domain controller identified, determine whether that domain controller will be a write-able or a read-only domain controller (RODC). The full domain controller should only be placed in locations where the physical security of the domain controller can be ensured.

The primary reason to use an RODC is for locations with poor physical security. Since the RODC is read-only, nothing on the RODC can be changed and replicated back to the write-able domain controllers. RODCs require upstream access to a full domain controller for authentication purposes. By default, none of the hashes for passwords are replicated to the RODC. The RODC forwards the request for logon to a writeable domain controller. It’s possible to configure the environment so that the full domain controller replicates the requested hash back to the RODC for caching. It should be noted that if this occurs and the RODC is compromised, only the hashes replicated to the RODC need to be reset.

The functionality provided by the RODC may be affected if the WAN is down or a full domain controller is not available to service requests from the RODC.

Determine which domain controllers will be writable and which will be read-only, and record the decisions in the job aid.

## Step Summary

A minimum of two domain controllers is needed to provide fault tolerance for a domain. Based on previously described business requirements, domain controllers can be placed in physical locations to provide local authentication. Additional domain controllers may be required based on user authentication and application requirements. The use of RODC servers can increase security dramatically and also can increase performance. The cost for adding these servers in the correct scenarios is minimal and should be considered.

The decision to add or remove domain controllers can be changed at any time.

# Step B3: Determine Global Catalog Placement

Global catalog services facilitate the lookup of information to all domains in the forest, specifically to domains outside of the current domain. The catalog is a subset of information from each domain that is replicated to every global catalog server in the forest. Applications such as Exchange Server rely heavily on the global catalog for relevant information. The global catalog is also used during the logon process to enumerate universal group memberships.

All global catalog services physically reside on one or more domain controllers. There is no way to separate global catalog functionality from a domain controller.

The decision needs to be made as to which domain controllers in the forest will host global catalog services.

## Task 1: Determine Global Catalog Locations and Counts

If a forest consists of only one domain, then all domain controllers should be configured as global catalog servers. The subset of data that would be replicated to all global catalogs will already be replicated through the normal domain replication process. There will be no additional requirements for disk space usage, CPU usage, or replication traffic.

If a forest contains multiple domains, then typically each domain controller should not be a global catalog server because of the increase in storage requirements and the additional replication overhead.

In a multi-domain forest environment, a subset of the domain controllers in the environment will be configured to run as global catalog servers. Because all global catalogs replicate a subset of all objects in each domain, placement of the global catalog needs to be carefully considered with respect to the increased bandwidth overhead introduced by the additional traffic. In addition, there are increased hardware requirements for storing global catalog data. Figure 4 illustrates the decision tree flow for deciding where to place global catalogs in the environment.



Figure 2. Decision tree flow for placement of global catalog servers in the environment

### Are There Any Applications That Need a Global Catalog Server Running at the Location?

Certain applications, such as Exchange Server, Microsoft Message Queuing (MSMQ), and applications that use distributed COM (DCOM), rely heavily on global catalog servers. These applications tend to perform better when they have a local global catalog available to improve query times.

There may be some application restrictions around whether a RODC can be used as a global catalog. Exchange Server does not support global catalogs running on an RODC; however, Microsoft Outlook® messaging and collaboration clients can use an RODC global catalog for Address Book lookups.

### Is the Number of Users at the Location Greater Than 100?

Global catalog servers should be placed at any location that has more than 100 users in order to reduce WAN traffic and to prevent productivity loss in case of WAN link failures.

### Is the WAN Link 100 Percent Available?

Consider placing a global catalog server in a location in which the WAN link is not sufficiently reliable to ensure user authentication, or else configure universal group membership caching.

### Do Many Roaming Users Work at the Location?

Roaming users need to contact a global catalog server whenever they log on for the first time at any location. Therefore, a global catalog server should be placed at locations that include many roaming users. Often, too many logons over the WAN link can cause significant WAN traffic and cause performance degradation and production loss.

### Can Universal Group Membership Caching Suffice?

Universal group membership caching is an option for locations that include fewer than 100 users and do not include many roaming users or applications that require a global catalog server. Universal group membership caching can be enabled on domain controllers that are running Windows Server 2003 or Windows Server 2008.

When a user logs on to the network, the global catalog server is contacted to enumerate universal group membership for that user. Over slow links, this process can take a significant amount of time or, in the event of a failure to contact the global catalog server, can result in denial in the logon process. Universal group membership caching can be used to address this problem. Universal group membership caching is available by default on domain controllers that are running Windows Server 2003 or Windows Server 2008. The feature must be enabled on a per-site basis.

### How Many Global Catalogs?

Once it has been determined that global catalog servers are required in a location, the next question is how many global catalog servers are required. In most cases, one or two global catalog servers will suffice in each location. Application requirements, such as Exchange Server requirements, may increase the number of global catalog servers required.

Record which domain controllers will be configured as global catalogs.

## 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 question has been known to affect global catalog placement decisions:

* **Are there any users who travel frequently to satellite locations and who require high-performance logon and directory services in those locations?** A global catalog server may be required in some satellite locations to provide a local experience for traveling users.

## Step Summary

Configure domain controllers as global catalog servers only when there is a technical reason to do so. Exceptions may be made when a population of traveling users requires high-performance global catalog services in sites outside the users’ domains.

Keep the number of global catalog servers to a minimum to reduce cost, management, and complexity of configuration and maintenance.

The design of the global catalog servers must be repeated for every forest.

## Additional Reading

Planning Domain Controller Placement: <http://technet.microsoft.com/en-us/library/cc754920(WS.10).aspx>

# Step B4: Determine Operations Master Role Placement

The next step is to decide the placement of the operations master roles (also known as flexible single master operations, or FSMO) for the forest and each domain. Although each domain controller within AD DS can authenticate accounts and write to the directory database, some functions are dedicated to a single domain controller. Operations master roles exist on designated domain controllers and control specific functions of the domain and forest.

Each domain has three operations master roles:

* **Primary domain controller (PDC) emulator operations master.** This role processes all replication requests from Windows NT 4.0 backup domain controllers (BDCs) and processes all password updates for clients that are not running AD DS client software. This is also the default domain controller used for updating Group Policy.
* **Relative ID (RID) operations master.** This role allocates RIDs to all domain controllers in order to ensure that all security principals have a unique security ID (SID).
* **Infrastructure operations master.** This role maintains a list of the security principals from other domains that have membership in groups within the operations master’s domain.

There are also two operations master roles for each forest:

* **Schema operations master.** This role allows changes to the schema.
* **Domain naming operations master.** This role is responsible for additions and removal of domains, sites, and domain-based DFS configurations to and from the forest.

As a general guideline, keep the operations roles on as few domain controllers as possible to simplify tracking the role locations. If the load on the operation master justifies a move, place the RID and PDC emulator roles on separate domain controllers in the same site. The domain controllers should be direct replication partners.

In general, the infrastructure master should never be placed on a global catalog server. If an infrastructure master is placed on a global catalog server, it will not correctly identify outdated security principals from other domains. The exception is in domains in which all domain controllers are global catalog servers or in a single-domain forest. In these cases, the infrastructure master has all the information it needs.

The schema and domain naming masters are rarely used and should be tightly controlled; keep them together on the same domain controller that hosts the global catalog. Certain operations, such as creating grand-child domains, use the domain naming master and will fail if the role is not on a global catalog server.

Place these domain controllers in a location that has the most users for that domain and that has a highly reliable network. Operations master role placement can be modified easily.

All operations master roles should be placed on domain controllers that are readily available to all other domain controllers in the environment. Domain controllers that are unable to communicate with the domain controllers hosting the operations master roles can experience failures.

## Task 1: Operations Master Role Placement

In a single domain forest, leave the five roles on a single server. There is no benefit to separating the roles.

In the forest root domain of multi-domain forests, leave all the operations master roles on the same domain controller, provided that all domain controllers in the forest root domain are also global catalog servers. There is no benefit to separating the roles.

If some of the forest root domain controllers are not configured as global catalog servers, then move the infrastructure master role to a domain controller that is not a global catalog server and ensure that the server is never configured as such. The infrastructure master role should not reside on a global catalog server unless all domain controllers in the domain are global catalog servers.

In all other domains, the three domain-specific operations master roles can reside on the first domain controller for that domain. Do not place the infrastructure master role on a domain controller that is also a global catalog server.

## Step Summary

Operations master roles should be placed strategically to ensure the complete and proper functioning of all directory services, from both an authentication and a management standpoint. Operations master role server placement must be decided for five roles in the root domain and three roles for all other domains in the forest. This process must be completed for every forest.

## Tasks and Considerations

For each operations master role, designate a domain controller that can host the operations master roles. The standby operations master domain controller should be a direct replication partner of the actual operations master role holder in case the standby can assume the role in the event the actual role holder fails. The new operations master role holder will then have the most up-to-date information regarding AD DS.

## Additional Reading

* Knowledge Base article 223346 “FSMO placement and optimization on Active Directory domain controllers”: <http://support.microsoft.com/kb/223346>
* Knowledge Base article 197132 “Windows 2000 Active Directory FSMO roles”: <http://support.microsoft.com/kb/197132>
* Knowledge Base article 324801 “How to view and transfer FSMO roles in Windows Server 2003”: <http://support.microsoft.com/kb/324801>

# Step C1: Create the Site Design

The site design is the mapping of the physical network to the logical site construct within AD DS. A site within AD DS is a logical collection of one or more well-connected TCP/IP subnets. Sites are used to control directory replication by setting a schedule for inter-site replication. Sites also are used to direct client systems to network resources that are AD DS–aware, and thus can be logically placed closest to these resources.

The following decisions need to be made:

* Should a physical location be directly correlated to a site?
* Can a physical location be grouped with other locations into a site?

Once the sites have been identified, the final tasks will be to map the TCP/IP subnets represented in a specific location to the corresponding site. The site design can be changed later if necessary.

## Task 1: Create a Site for the Location

A site should be defined for any physical location in which domain controllers are being placed, as well as any physical location that contains resources or services that rely on site topology information to direct the client to the nearest requested resource.

For example, if numerous physical locations need to access file resources, these resources can be configured within a Distributed File System (DFS) environment. After placing the DFS servers that contain the resources in the physical locations, a site can be configured for each location. When a client accesses the DFS-based resource, the local DFS resource will be accessed, reducing WAN traffic and increasing performance for the resource access.

Finally, sites can be created to control which domain controllers handle authentication traffic for applications that have extremely high authentication requirements. Large Microsoft SharePoint® portal environments can generate significant domain controller/global catalog traffic. By creating a site specifically for the SharePoint servers and assigning specific domain controller/global catalogs to the site, administrators can control the authentication traffic of the portal solution.

For each site identified, record the site name and the IP subnets that are assigned to that site.

## Task 2: Associate Location to Nearest Defined Site

For any remaining physical locations that have not been associated with a site within AD DS, associate the subnets in that location to an existing site. The site selected should include a location that has the greatest WAN speed and available bandwidth to the location being configured. This approach will help direct client traffic generated within the location to the site having the greatest capacity to handle the additional traffic.

Record the assignment of the additional subnet information to the selected site.

## Step Summary

Each physical location should be examined and a decision should be made as to whether the location should be a new site within the directory or should be associated to another site. The subnets within each location should be assigned to the site in which they belong. Each domain controller should also be assigned to the proper site.

The site design needs to be completed for each forest.

## Additional Reading

Best Practices for Active Directory Design and Deployment: <http://technet.microsoft.com/en-us/library/bb877991.aspx>

# Step C2: Create the Site Link Design

Site links are used to connect the defined sites in AD DS. The site links reflect the inter-site connectivity and method used to transfer replication traffic. All sites must be connected with site links if the domain controllers at each site are to replicate. By default, all sites belong to the default-first-site-link, with replication scheduled to occur every 180 minutes, each day of the week.

Site links can be created at any time and sites can be added or removed easily. However, there could be an impact on replication because of latency issues when reconfiguring sites, site links, and scheduling associated with the site links.

## Task 1: Determine the Site Link Design

AD DS automatically creates the default-first-site-link. When all sites in the design are connected and have the same connectivity and availability to one another, a single site link can be used to represent the links between the sites. This full mesh design assumes that all sites are well connected and that there is no need to design specific links between sites. This approach simplifies the design by eliminating the need to design site links, as well as automatically configuring the site link structure.

Since the connectivity and availability of the links are identical, the replication schedule, interval, and cost will be configured identically. This choice is only useful when all of the sites are connected by WAN connections with identical available bandwidth and latency.

If sites are connected with physical network links that have different costs of usage, availability, speed, or available bandwidth, there may be a need for different replication schedules. A new site link would need to be created to account for these differences.

Site links use a cost algorithm to influence which path replication traffic will use to flow between sites. A preferred connection would be configured at a lower cost than a less-preferred connection. The replication system uses the link with the lowest cost. If there is a dollar cost to using a link, the link might be assigned a higher cost value as well.

The replication of traffic across the link is controlled by the availability schedule and how frequently the link is set to replicate. For example, a link can be configured to replicate every 30 minutes during the hours of 2:00 A.M. to 4:00 A.M., Monday through Friday.

Site scheduling can specify intervals as brief as every 15 minutes, ranges from any time of the 24 hour clock, and any combination of days of the week.

When assigning the replication schedules and intervals, care should be taken to ensure that any replication goals required by the organization are met. Replication goals can be defined such that all changes are recorded in a set period of time for the following:

* **Configuration and schema convergence.** Any changes to configuration or schema are replicated to all domain controllers in the forest.
* **Domain convergence.** All domain changes are replicated to all domain controllers in the domain.
* **Global catalog convergence.** All global catalog changes are replicated to all global catalogs in the forest.
* **Application partition convergence.** All application partition changes are replicated to all domain controllers hosting the affected application partition.

When defining the replication schedules and intervals, ensure that all replication goals are met for worst case scenarios. That is, can a change originating in one site replicate within the time frame with the site that is the greatest number of hops away from the originating site? If it is not possible to meet the goal, then the interval and schedule need to be updated or the goal needs to be redefined.

Consider a topology that consists of five sites (A–F), consisting of a single forest with four domains. The sites are connected to one another through direct site links with the replication schedule configured for 24 hours and the interval set at the default of 3 hours. For the purposes of this example, there are two replication goals: one for schema and configuration convergence to be completed in 6 hours and one for global catalog convergence in 4 hours.



Figure 3. Configuration and schema convergence

Because of the number of hops involved, it is not possible for a change introduced in site E or F to converge at site C within 6 hours; it would take a minimum of 9 hours (3 hops). Either the replication goal would need to be updated or the interval would need to be set to another value.



Figure 4. Global catalog convergence

In the example of global catalog convergence, there are four global catalog servers in sites A, B, D, and E. If a change is introduced in site D, then all sites will be updated within the 4-hour replication goal. However, if the change is introduced in sites A, B, or E, then it will not be possible to meet the goal of 4 hours as it will take a minimum of 6 hours (2 hops) to reach all sites. Again, the replication goals would need to be updated or the interval time frame changed on the site links.

Associate all sites with similar links to the new link and remove the sites from the default-first-site-link.

For each site link identified, record the name of the site link, the cost associated with using the link, and the link’s replication schedule and interval. For each site, record the link that is used to connect it to other sites. A site may have multiple site links associated with it.

## Step Summary

Links between all sites should be defined through the use of one or more site links. If any sites are disconnected from the others, the Knowledge Consistency Checker (KCC) will generate an error message. The site links control the replication of the directory database between domain controllers in different sites and, if multiple paths are available, control which path is preferred.

The site link design can be changed. However, changing the site links may have an impact on the performance of directory changes until all updates converge.

## Additional Reading

* Creating a Site Link Bridge Design: <http://technet.microsoft.com/en-us/library/cc753638(WS.10).aspx>
* Appendix B: Do Not Use a Lag Site as a Disaster Recovery Strategy: <http://technet.microsoft.com/en-us/library/dd835581(WS.10).aspx>

# Step C3: Create the Site Link Bridge Design

A site link bridge enables transitivity between site links. Each site link in a bridge needs to have a site in common in order for replication to flow correctly across the bridge. The site link bridge design can be changed, but it should be done carefully to ensure that the replication of AD DS is not compromised or stopped.

## Option 1: Default Behavior

If the network is fully routed and there is no need to control the AD DS replication flow, then leave the transitivity enabled for all site links by leaving the **Bridge All Site Links** option enabled. This is the default state.

By allowing all transitivity across all sites, any domain controller in a site can create a direct replication partner with another domain controller in another site. This simplifies replication in that there is no need to restrict or define which sites a domain controller can use to search for replication partners.

This may become an issue with larger implementations that are based on a hub-and-spoke model. By bridging all site links, there is no control on which domain controller is considered part of the hub site when it comes to replication.

## Option 2: Custom Site Link Bridge

If a network is not fully routed, disable the **Bridge All Site Links** option for the IP transport and configure site link bridges to map to the physical network connections. Additionally, if the IP network is fully routed but there are too many routes that the KCC should not consider, creating a custom site link bridge topology and disabling the automatic transitivity of site links will eliminate confusion. The KCC, by default, will consider all possible connections and bridges for replication.

Site link bridges can also be used to control replication flow of AD DS. The two most common reasons for creating site link bridges are to control replication for failover of a hub-and-spoke network design and to control replication through a firewall. If AD DS replication flow is to be controlled through the design of site link bridges, then disable the **Bridge All Site Links** option for the IP transport.

By configuring two site link bridges for replication of AD DS between two sites, replication will succeed even if one link fails. This is necessary because the disabling of **Bridge All Site Links** will negate the KCC and Intersite Topology Generator (ISTG) from helping with the bridging of site links in the case of a failure of any aspect of the topology.

If replication traffic passes through a firewall and the firewall is configured to allow connections from specific domain controllers, then site link bridges need to be configured to match this environment. A site link bridge is created for each side of the firewall, and the site links connecting each site are associated to the site link bridge on the links’ side of the firewall. The site link that connects the two sites through the firewall will not be placed in a bridge. If a domain controller that is allowed to communicate through the firewall fails, its replication partners will attempt to set up new replication partners only with domain controllers in sites that are part of the bridge.

It should be noted that the robustness of AD DS replication can be reduced by the choices being made. For example, if all domain controllers in the hub of a hub-and-spoke design fail, then the satellite sites will become disconnected from the replication topology because all their potential partners have been removed from the network.

Likewise, if the domain controllers that can communicate across the firewall fail, then replication will update only those changes that are made on either side of the firewall. Modifications will not cross the firewall until the failed domain controllers are brought back online.

## Evaluating the Characteristics

This table can assist in comparing the complexity of a default site link bridge versus custom site link bridge.

Table 5. Evaluate the Characteristics of Site Link Bridges

|  |  |  |
| --- | --- | --- |
| Complexity | | |
| Default site link bridge | Using the default configuration means a less complex implementation. | Low |
| Custom site link bridge | Customizing the configuration increases the complexity of the environment. | High |

## Step Summary

Site link bridges should be configured and the default configuration changed only when the network requires such modifications. All sites should be interconnected with one another, either directly or through the bridge.

The decision to establish or change site link bridges can be changed.

Record any additional site link bridges that are created and the associated site links with that bridge.

## Additional Reading

Creating a Site Link Bridge Design: <http://technet.microsoft.com/en-us/library/cc753638(WS.10).aspx>

# Step D1: Determine Domain Controller Configuration

Once the number of domain controllers has been identified, the final step is to determine the disk space, memory, processor, and the network requirements for each domain controller.

## Task 1: Identify Minimum Disk Space Requirements for Each Domain Controller

For each domain controller, plan to allocate at a minimum the following amount of space:

* 500 MB for AD DS transaction logs.
* 500 MB for the drive containing the SYSVOL share.
* 1.5 GB to 2 GB for the Windows Server 2008 operating system files.
* 0.4 GB of storage for every 1,000 users in the directory for the NTDS.dit drive.

For example, for a forest with two domains (domain A, domain B), with 10,000 and 5,000 users respectively, provide a minimum of 4 GB of disk space for each domain controller that hosts domain A and a minimum of 2 GB of disk space for each domain controller that hosts domain B.

Domain controllers running as global catalog servers will need additional disk space allocated if the forest contains more than one domain. For a given global catalog server, the additional space requirement is 50 percent of the recommended disk space for each additional domain outside of the global catalog server’s own domain. In the earlier example, Domain A required 4 GB of disk space and Domain B required 2 GB of disk space. For a global catalog server in Domain A, an additional 1 GB would be needed (Domain B’s 2 GB / 2), for a total of 5 GB of storage. For a global catalog server in Domain B, an additional 2 GB will be needed (Domain A’s 4 GB / 2), for a total of 4 GB of disk space.

Finally, if any applications are using the directory to store data in an application partition, the storage requirements for each application partition will need to be added to the domain controller disk requirements.

Identifying capacity requirements is one element in planning the disk configuration. The second element is performance planning. The disk subsystem needs to be configured to read and write data at a rate that meets business expectations for performance. Some form of RAID can be used to provide fault tolerance for the data.

For smaller sites, a single disk may meet both the capacity and performance requirements. For larger sites, the log, OS, and database files may need to be placed on separate volumes in order to meet the performance requirements. Test the configuration to ensure that the disk subsystem is not a bottle neck with the expected load. Additional disk spindles may be required if performance is lacking in the original capacity-based disk configuration.

Record the drive configuration information for each server.

## Task 2: Identify Memory Requirements for Each Domain Controller

The following table gives a conservative estimate of the minimum required memory allocation for a domain controller. The table assumes that the domain controllers are hosting only AD DS and DNS.

Table 6. Minimum Required Memory Allocation

|  |  |
| --- | --- |
| User per domain in a site | Minimum memory requirements per domain controller |
| 1–499 | 512 MB |
| 500–999 | 1 GB |
| > 1,000 | 2 GB |

Although this table lists the minimum, additional memory can improve the performance of the directory. AD DS will attempt to cache the database in memory. This reduces disk access and improves performance. This cache is limited by the virtual address space and the amount of physical RAM on the server.

If there is an existing infrastructure, measure the performance of the domain controllers to determine if the existing memory is sufficient for the environment. If this is a new deployment, begin with 2 GB of RAM. Test the configuration with the expected loads and add memory as required.

To determine whether more RAM is needed for the server, monitor the percentage of AD DS operations being satisfied from the cache by using the Reliability and Performance Monitor. Examine the lsass.exe instance (for Active Directory Domain Services) or Directory instance (for Active Directory Lightweight Directory Services) of the Database\Database Cache % Hit performance counter. A low value indicates that many operations are not being satisfied from the cache; adding more RAM might improve the cache hit rate and the performance of AD DS. You should examine the counter after AD DS has been running for some time under a normal workload. The cache starts out empty when the AD DS service is restarted or the machine is rebooted, so the initial hit rate is low.

The use of the Database Cache % Hit counter is the preferred way to assess the amount of RAM a server needs. Alternatively, a guideline is that when the RAM on a server is twice the physical size of the AD DS database on disk, it likely gives enough room for caching the entire database in memory. However, in many scenarios this is an overestimation because the actual portion of the database most frequently used is only a fraction of the entire database.

## Task 3: Determine Processor Requirements

A 32-bit server running the standard edition of Windows Server 2008 can only address 4 GB of RAM. If there is an expected need to grow the RAM in a server beyond 4 GB, then move to a 64-bit architecture. By moving to a 64-bit version of Windows Server 2008, or Windows Server 2008 R2, future expandability of the system is protected as well as future proofing the hardware from future market decisions around 32-bit.

Note   Windows Server 2008 R2 is available only in a 64-bit version.

The general recommendation is that for sites with less than 500 users, start with a single processor; for sites with less than 10,000 users, start with dual processors or cores and then scale from there. This assumes that the primary work of the directory is user authentication.

If the servers handle additional requests, such as Exchange Server, then monitor the performance of the system and adjust the number of processors and/or cores as required. If there are existing domain controllers in the environment, then performance monitoring the existing boxes can be useful for getting a baseline on the required hardware.

Record the number of processors and cores and chosen architecture for each domain controller.

## Task 4: Identify Network Requirements for Each Domain Controller

Many corporate networks run at either 100-megabit or gigabit connectivity to the servers. Typically, a single network adapter is sufficient to handle all the network traffic to and from the server.

Placing multiple network adapters in a domain controller can cause a variety of issues, ranging from replication failures to authentication failures, and is generally not recommended.

## Tasks and Considerations

AD DS is optimized for read-heavy scenarios, that is, where the workload consists of more query operations than update operations. The most important performance tuning step is to ensure that the server has sufficient RAM to be able to cache the most frequently used portion of the database in memory. By monitoring the Database Cache % Hit on the server, a determination of whether additional memory is required can be made. The percentage of hits will be low if the directory service was just recently started.

In scenarios where the directory is write-heavy, then optimize the disk subsystem for performance. Using hardware RAID controllers, low-latency high RPM disks, and battery-backed write caches on the controller can help improve performance. Because most of the workload consists of writes, the cache does not provide as much benefit as it does in the read-heavy scenarios.

## Step Summary

The proper physical configuration of domain controllers is essential to the proper operation of AD DS. Critical elements include the disk subsystem, memory, processor, and network adapters. Hardware can be reconfigured as needed but doing so may require outages.

## Additional Reading

* *Performance Tuning Guidelines for Windows Server 2008*: <http://www.microsoft.com/whdc/system/sysperf/Perf_tun_srv.mspx>
* “Active Directory Performance for 64-bit Versions of Windows Server 2003”: <http://www.microsoft.com/downloads/details.aspx?FamilyID=52e7c3bd-570a-475c-96e0-316dc821e3e7>

# Conclusion

This guide has focused on summarizing the critical design decisions, activities, and tasks required to enable a successful design of AD DS in Windows Server 2008 or Windows Server 2008 R2. It has addressed the technical aspects, service characteristics, and business requirements needed to complete a comprehensive review of the decision-making process. When used in conjunction with product documentation, this guide can help organizations confidently plan AD DS implementation.

## Additional Reading

In addition to the product documentation, the web link that follows contains supplemental information on the product concepts, features, and capabilities addressed in this guide:

*Windows Server 2003 Deployment Kit: Designing and Deploying Directory and Security Services:* <http://go.microsoft.com/fwlink/?LinkID=15308>

# Appendix A: Design Job Aids

This job aid summarizes the decision and task results. Additional job aids included with the *Windows Server 2003 Deployment Kit* can be found at <http://www.microsoft.com/downloads/details.aspx?familyid=6cde6ee7-5df1-4394-92ed-2147c3a9ebbe&displaylang=en>.

1. How many forests?

Number \_\_\_\_

Names \_\_\_\_\_\_\_\_

1. How many domains per forest?

Forest 1 \_\_\_\_

Forest N \_\_\_\_

1. Assign domain names (DNS and NetBIOS) for each domain (add columns for additional forests).

Table A-1. Domain and Forest

|  |  |  |
| --- | --- | --- |
| Domain | Forest 1 | Forest N |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| N |  |  |

1. List the forest root domain for each forest.

Forest 1 \_\_\_\_\_

Forest N \_\_\_\_\_\_

1. Design the OU structure for each domain.

Table A-2. Forest Root Domain and OU Name

|  |  |  |  |
| --- | --- | --- | --- |
| OU name | Domain | Purpose | Controls |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. Determine the domain controller placement for each domain.

Sketch out the domain controller placement.

1. Determine the number of domain controllers for each location.

Table A-3. Domain Location and Number of Controllers

|  |  |
| --- | --- |
| Location | Number of domain controllers |
|  |  |
|  |  |
|  |  |

1. Plan global catalog server placement for each forest.

Sketch out the global catalog server placement.

1. Plan the operations master role placement for each forest and domain.

Sketch out operations master role placement.

1. Create a site design.

Sketch out site design.

1. Create a site link design.

Sketch out site links.

1. Establish the physical server design.

Table A-4. Server Name and Configuration Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| Server name | Processor | Disk size/ configuration | Memory |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Appendix B: IPD in Microsoft Operations Framework 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 B-1. The architecture of Microsoft Operations Framework (MOF) 4.0

# Appendix C: Active Directory Domain 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 [www.microsoft.com/infrastructure](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 Infrastructure Optimization Model was to develop a simple way to use a maturity framework that is flexible and 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 will help move an organization to the Rationalized level. AD DS provides the administrator with the mechanism for user and machine authentication within the organization. AD DS begins to move the organization to the Standardized level, while providing the infrastructure for additional services required in the Rationalized and Dynamic levels. This guide will assist you in planning and designing the infrastructure for an AD DS implementation.

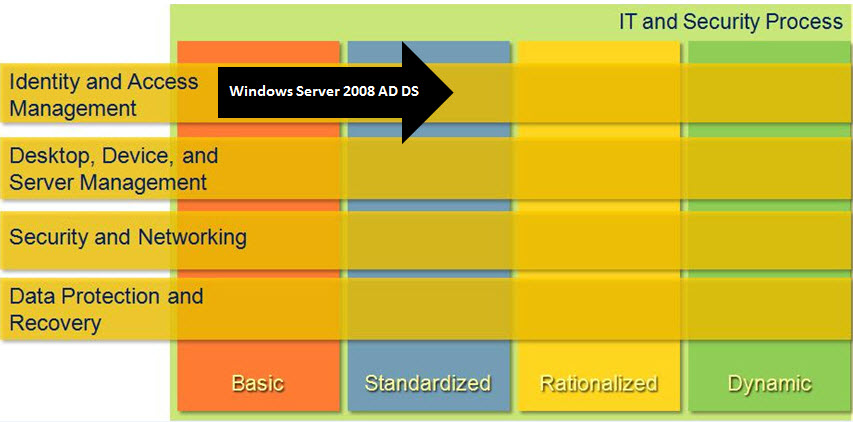


Figure C-1. Mapping of Active Directory Domain Services technology into the Core Infrastructure Optimization Model

# Version History

|  |  |  |
| --- | --- | --- |
| Version | Description | Date |
| 2.2 | Minor updates to this guide to reflect current IPD template. | November 2011 |
| 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. | July 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 Active Directory Domain 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*
* Charles Denny – *Microsoft*
* Michael Kaczmarek – *Microsoft*
* Robin Maher – *Microsoft*
* Derek Melber – *Studio B*
* Lee Oommen – *Microsoft*
* Fergus Stewart – *Microsoft*
* Melissa Stowe– *Microsoft*

**Reviewers:**

* Robert DeLuca – *Microsoft*
* Dave Field – *Studio B*
* Lex Liao – *Microsoft*
* Greg Myers – *Studio B*

**Editors:**

* Laurie Dunham – *Xtreme Consulting Group*
* Janet Majure – *Studio B*
* Lisa Pere – *Studio B*
* Ruth Preston – *Volt Technical Services*
* Pat Rytkonen – *Volt Technical Services*

## Feedback

Please direct questions and comments about this guide to [ipdfdbk@microsoft.com](mailto:ipdfdbk@microsoft.com?subject=IPD%20-%20Active%20Directory%20Domain%20Services).