

## Logical architecture components (SharePoint Server 2010)

**Updated: 2009-11-12**

[This article is pre-release documentation and is subject to change in future releases.]

There are a variety of ways you can arrange the components in a logical architecture design. Each of the components presents different opportunities for sharing and isolation. Before you begin your logical architecture design:

Know what your sharing and isolation goals are.

Evaluate the tradeoffs for each choice.

Each section in this article describes a particular logical architecture component and discusses the following considerations for that component: capacity, sharing and isolation, configurable items, administration, and planning recommendations.

In this article:

[Server farms](#)

[Service applications](#)

[Application pools](#)

[Web applications](#)

[Zones](#)

[Policy for a Web application](#)

[Content databases](#)

[Site collections](#)

[Sites](#)

[Host-named site collections](#)

[My Sites](#)

### Server farms

A server farm represents the top-level element of a design. Individual server farms provide physical isolation.

Several criteria that are determined by your organization might affect the number of server farms that are required, including:

Heavy use of services might warrant one or more dedicated services farms.

Separate operational divisions of responsibility.

Dedicated funding sources.

Separate datacenter locations.

Industry requirements for physical isolation between sites.

However, you can satisfy many isolation requirements on a single server farm. For example, you can use different Internet Information Services (IIS) application pools with different process identities to achieve isolation at the process level for both sites and service applications.

In addition to isolation requirements that might require more than one server farm, an organization might implement multiple server farms to satisfy performance and scale goals, licensing requirements, or a publishing environment.

### Service applications

A service application provides a resource that can be shared across sites within a farm or, in some cases, across multiple farms.

In SharePoint Server 2010, services are no longer contained within a Shared Services Provider (SSP). Instead, the infrastructure for hosting services moves into Microsoft SharePoint Foundation 2010 and the configuration of service offerings is much more flexible. Individual services can be configured independently and third-parties can add services to the platform.

You can deploy only the services that are needed to a farm. Services that are deployed are called service applications.

Service applications are associated with Web applications. Each service application can be configured differently:

- Web applications can be configured to use only the services that are needed, rather than the entire set of services that are deployed.

- You can deploy multiple instances of the same service in a farm and assign unique names to the resulting service applications.

- You can share service applications across multiple Web applications within the same farm.

- Some service applications can be shared across farms.

### Capacity

There is no recommended limit for the number of service applications in a single farm.

### Sharing and isolation

Service applications can be shared in two ways:

- Sharing the service application and the service data. This is the default behavior for services that are shared across Web applications. For example, search results are shared across Web applications that consume the same search application.

- Sharing only the service application, but isolating the data by deploying the service in partitioned mode. In a hosted environment, you can deploy service applications in partitioned mode by using Windows PowerShell. Each tenant's data is stored in a separate partition in the database for the service. A tenant's subscription ID is used to map the tenant's service data to

their sites. For example, if you deploy the search service in partitioned mode, each tenant will only see search results for their own content.

 **Note:**

Not all service applications support partitioning.

Conversely, service applications can be isolated in two ways:

Deploying multiple service applications in separate application pools to achieve process isolation of services and service data. For example, a finance team might warrant a separate and dedicated Business Data Connectivity application.

Deploying services in partitioned mode. This approach works well in hosted environments in which tenants will never share service data. However, it might not be practical in environments where there is a mixture of needs for shared and isolated service data.

If needed, you can additionally isolate service applications by deploying them to separate application pools to achieve process isolation. However, application pools are a limited resource and farm performance is affected if too many application pools are used. For more information, see [Application pools](#) in this article.

### Configurable items

The following table displays configurable items that contribute to isolation and sharing.

Item	Description
Default group	By default, all service applications are included in the default group. You can add and remove service applications from the default group at any time, including when you create one.  When you create a Web application, you can select the default group or you can create a custom group of services. You create a custom group of services by selecting only the service applications that you want the Web application to use.
Connection (proxy)	When you create a service application, a connection for the service application is created at the same time. A connection is a virtual entity that connects Web applications to service applications. Some service applications, such as the Managed Metadata Service, store settings in the connections. In Windows PowerShell, connections are called proxies.
Service application permissions	You can delegate management of service applications to other users by granting them permissions to one or more of the service applications.
Trusted My Site host locations	In organizations where multiple User Profile Service applications are deployed, this feature ensures that users create a My Site in the location that is intended for their profile. This feature prevents users from creating multiple My Sites across an organization.

### Administration

Configuration and management of service applications can be delegated to administrators who specialize in managing individual services, such as search, user profiles, and managed metadata.

In a hosted environment, tenants can manage some of the service settings for their organization.

## Planning recommendations

Configure service applications either to share resources across multiple Web applications or isolate content.

For example, multiple sites that reside in different Web application and application pools can be unified by sharing services in the default proxy group to provide taxonomy, content type, and profile sharing across an intranet. This provides for personalization and enterprise-wide standardization across many sites and applications. This choice provides an example of balancing process isolation (by implementing separate Web applications and application pools) with the business need to share information and leverage profile data across the applications.

You can also configure service applications to enhance your overall isolation goals. For example, using a dedicated set of services for partner collaboration ensures that partner users cannot search on or access sensitive information within your intranet environment. You can configure individual services to further isolate content between site collections. For example, you can:

- Limit search scopes to the individual site collections.

- Configure the User Profile service to only display users that are part of the same organizational unit in Active Directory Domain Services (AD DS).

- Use the Stsadm command-line tool to configure the People Picker to display only users that are members of the site collection.

When you design your services strategy for an organization, consider the ways in which you can configure the individual services to enhance your overall content sharing or isolation goals.

When you design a services strategy for a hosting environment, determine which services will be available and partitioned.

## Application pools

In Internet Information Services (IIS) 7.0, an application pool is a group of one or more URLs that are served by a worker process or set of worker processes.

When you create site collections and services in SharePoint 2010 Products, you select an application pool to use or you can create a new application pool. Each application pool has its own worker process and can have a separate identity (security account) which prevents two processes from interacting.

## Capacity

The memory overhead of an application pool is 30-50 megabytes (MB) plus any memory for the applications running in the application pool process space. The various application demands usually quickly drives the memory usage of an application pool to 800 MB or larger. The limit for the number of application pools is influenced by the available memory on the system. That is, the number of application pools is dictated by the following two factors:

- Available addressable memory.

- The amount of memory consumed by applications running in the application pool.

The general guideline for acceptable performance is to use eight or fewer application pools.

## Sharing and isolation

IIS application pools provide a way for multiple sites to run on the same server computer but still have their own worker processes and identity. This can help to prevent an exploit on one site that enables the attacker to inject malicious code that can attack sites in different application pools. More importantly, this strategy isolates code that introduces memory issues or other issues so that the problematic code does not affect all applications.

**Configurable items**

Using a separate application pool identity for each application pool is recommended, if needed, for security and reasons of isolation.

**Administration**

If separate identities are used for each application pool, each identity will have to be maintained.

**Planning recommendations**

Practically speaking, consider using a dedicated application pool for each of the following reasons:

- To separate authenticated content from content that is primarily anonymous.

- To isolate applications that store passwords for and interact with external business applications, for example, Business Data Connectivity connections.

**Web applications**

A Web application is an IIS Web site that is created and used by SharePoint 2010 Products. A Web application can be extended up to four times to create four additional zones in SharePoint 2010 Products, resulting in up to five IIS Web sites that are associated with a single Web application, each IIS Web site associated with a different zone. You can assign each Web application a unique domain name. For more information, see [Zones](#) in this article.

**Sharing and isolation**

Each Web application has a unique domain name, which helps to prevent cross-site scripting attacks.

**Configurable items**

The following table displays configurable items that contribute to isolation and sharing.

Item	Description
Service applications	Service applications are associated with Web applications. When you create a Web application, you can select the default proxy group (default set of service applications), or you can specify a custom set of service applications for the Web application. All sites within a Web application consume services from the same service applications. A service application can provide services for more than one Web application, thus sharing content and profile data across the Web applications.
Zones	Within a Web application, you can create up to five zones. Use zones to enforce different access and policy conditions for large groups of users.
Policy for Web application	Create a policy to enforce permissions across one or more zones in a Web application. A policy can be created for a specific user or user group. For more information, see <a href="#">Policy for a Web application</a> in this article.

**Administration**

Ongoing administration of Web applications is not significant.

**Planning recommendations**

Generally speaking, use dedicated Web applications to:

Separate content available to anonymous users from content available to authenticated users.

Isolate users. For example, you can ensure that partners do not have access to intranet content by placing partner sites in a separate Web application.

Enforce permissions through the use of policies. For example, you can create a policy to explicitly deny write access to one or more groups of users. Policies for a Web application are enforced regardless of permissions configured on individual sites or documents within the Web application.

Optimize database performance. Applications achieve better performance if they are placed in content databases with other applications with similar data characteristics. For example, the data characteristics of My Sites typically include a large number of sites that are small in size. In contrast, team sites typically encompass a smaller number of very large sites. By placing these two different types of sites in separate Web applications, the resulting databases are composed of data with similar characteristics, which optimizes database performance.

Optimize manageability. Because creating separate Web applications results in separate sites and databases, you can implement different limits for each site's Recycle Bin, expiration, and size, and negotiate different service-level agreements. For example, you might allow more time to restore sites that are not critical to your business.

## Zones

Zones represent different logical paths (URLs) of gaining access to the same Web application. Within each Web application, you can create up to five zones using one of the available zone names: Default, Intranet, Internet, Custom, or Extranet. Each name can only be selected once per Web application. Each zone is represented by a different Web site in IIS.

The Default zone is the zone that is first created when a Web application is created. The other zones are created by extending a Web application.

### Capacity

You can create up to five zones within a Web application. Typically, zones are coordinated across Web applications so that zones of the same name are configured for the same users.

### Sharing and isolation

Zones provide a method of partitioning users by:

**Authentication type:** Each zone can be configured to use a different authentication provider, enabling you to share the same content across partner companies.

**Network zone:** Each zone can be configured to accommodate users entering from a different network zone, such as an extranet or the Internet.

**Policy permissions:** You can explicitly allow or deny read or write access to content per zone based on a user account or a group account.

### Configurable items

The following table displays configurable items that contribute to isolation and sharing.

Item	Description
------	-------------

Authentication provider	Each zone can be configured to use a different authentication provider.
Anonymous access	Turn anonymous access on or off per zone.
Secure Sockets Layer (SSL)	Turn SSL on or off per zone.
Public URL and alternate access mapping	Specify the domain name users will type to access content in the Web application. Alternatively, use alternate access mapping to map user-friendly or zone-appropriate URLs to the default URL (server name and port) for each zone. Alternate access mapping provides support for off-box termination of SSL. Off-box termination of SSL is when a proxy server terminates an SSL request and then forwards the request to a Web server by using HTTP. In this case, alternate access mappings can be configured to return these requests using SSL, thus maintaining secure communication between the client and the proxy server.
Policy for Web application	Create a unique set of policies for each zone within the Web application. If you have a special group of users that require exceptions to your overall security policy, consider using a separate zone to accommodate these users.

### Administration

If you use alternate access mapping, consider that all public URLs require Domain Name System (DNS) entries to map the public URLs to the IP address of the load balancer used for the farm.

### Planning recommendations

When you design zones, several key decisions are critical to the success of the deployment. These decisions include design and configuration decisions for the following zones:

- The Default zone

- Zones for external access

The following sections describe some of the planning recommendations and requirements for zones, including the default zone.

Administrative e-mail is sent with links from the Default zone. This includes e-mail to owners of sites that are approaching quota limits. Consequently, users who receive administrative e-mail messages and alerts must be able to access links through the Default zone. This is especially important for site owners.

Host-named site collections are only available through the Default zone. All users who are intended to access host-named site collections must have access through the Default zone.

The Default zone must be the most secure zone. This is because when a user request cannot be associated with a zone, the authentication and policies of the Default zone are applied.

In an extranet environment, the design of zones is critical for two reasons:

- User requests can be initiated from several different networks, such as the internal network, a partner company network, or the Internet.

Users consume content across multiple Web applications. For example, an intranet environment might include sites that are hosted in several different Web applications. Additionally, employees might have access to both the intranet content and to partner collaboration content.

In an extranet environment, ensure that the following design principles are followed:

Configure zones across multiple Web applications to mirror each other. The configuration of authentication and the intended users should be the same. However, the policies associated with zones can differ across Web applications. For example, ensure that the Intranet zone is used for the same employees across all Web applications. In other words, do not configure the Intranet zone for internal employees in one Web application and remote employees in another.

Configure alternate access mappings appropriately and accurately for each zone and each resource.

### Policy for a Web application

A policy for a Web application enforces permissions on all content within a Web application, enabling you to set security policy for users at the Web application level. The permissions in a policy override all other security settings that are configured for sites and content.

You can configure policy based on users or user groups in AD DS, but not SharePoint groups. A policy can be defined for the Web application in general or just for a specific zone.

#### Capacity

There are no capacity restrictions that apply to policies for Web applications.

#### Sharing and isolation

A policy for a Web application provides a method of setting permissions based on users and the zone that they access content through.

For example, by using a policy, you can:

Allow Help desk staff access to all content.

Deny write access to partners or vendors.

Deny access to secure data to a group of users regardless of how site owners configure permissions.

Ensure that the crawl account has access to crawl all content.

#### Configurable items

The following table displays configurable items that contribute to isolation and sharing.

Item	Description
User policy	Create a policy that applies to users or user groups: The policy can be applied to all zones or one zone.

Anonymous policy	<p>You can enter user names, group names, or e-mail addresses.</p> <p>Specify the permissions that you want to apply to the policy.</p> <p>You can modify the default permission levels or create new permission levels by clicking <i>Permission policy</i> when you create the policy in Central Administration.</p> <hr/> <p>If anonymous access is enabled for the Web application or one or more zones, then you can create a policy that applies to all anonymous users. The default policy settings are:</p> <p>None: No policy</p> <p>Deny write: No write access</p> <p>Deny all: No access</p> <p>Anonymous user policy levels cannot be modified.</p>
Permission policy	<p>Edit the specific permissions associated with one of the default permission levels, or create a new permission policy level. Additionally, you can specify the particular permissions that allowed or denied for site collections and sites.</p> <p>After creating a new permission policy level, you can create a user policy that uses the permission policy.</p>

### Administration

Ongoing administration of policies for Web applications is not significant.

### Planning recommendations

Because policies are managed centrally, consider using policies to manage large groups of users, rather than individual users.

### Content databases

By default, all content for a Web application is stored in one content database. You can separate content into multiple content databases at the site collection level. A content database can include one or more site collections. A single site collection cannot span multiple databases. Backing up and restoring sites takes place at the content database level.

### Capacity

The guideline for acceptable performance is to implement 100 or fewer content databases per Web application.

### Sharing and isolation

Planning for databases enables you to either optimize for efficiency (multiple site collections sharing a database) or isolation (one database per site collection).

Achieve scale efficiency by managing databases to the maximum target size. In this case, you configure database settings to add new site collections to existing databases until the maximum number of site collections has been reached. You calculate the maximum number of site collections by estimating the average or maximum size of site collections divided into the maximum target size for the database. This approach works well when you expect a large number of small site collections, such as My Sites.

Achieve isolation of content between teams or projects by limiting a database to one site collection. This approach enables you to independently manage the content of individual teams. For example, you can independently manage each team's database for backup, recovery, and migration. This approach provides

the opportunity to implement different service-level agreements for different teams or projects. This approach also enables you to manage content to the lifecycle of a project. For example, you can archive a database when a project is completed.

### Configurable items

The following table displays configurable items that contribute to isolation and sharing.

Item	Description
Database server	Specify which SQL Server computer a content database is created on.
Failover server	You can choose to associate a content database with a specific failover server that is used in conjunction with SQL Server database mirroring.
Capacity settings	You can specify the number of sites that can be created before a warning event is generated and the maximum number of sites that can be created in each database.

### Administration

A manageable database administration plan balances the number of databases with the resources required to manage the databases.

Administration of databases includes:

- Creating new databases for new team sites or site collections that require dedicated databases.

- Monitoring database sizes and creating new databases when target sizes are approached.

- Backing up and restoring databases.

### Planning recommendations

Choose one of the following two approaches:

- Establish target sizes for content databases with appropriate size-warning thresholds. Create new databases when size-warning thresholds are reached. With this approach, site collections are automatically added to the available database or databases, based on target sizes alone.

- Associate site collections with specific content databases. This approach enables you to place one or more site collections in a dedicated database that can be managed independently from other databases.

If you want to associate site collections to specific content databases, you can use the following methods to accomplish this:

- Use Windows PowerShell to create a site collection in a new database.

- Apply the following database capacity settings on the Manage Content Database Settings page on the SharePoint Central Administration Web site:

  - Number of sites before a warning event is generated = 0

  - Maximum number of sites that can be created in this database = 1

Add a group of site collections to a dedicated database by performing the following steps:

Add a content database for the Web application and ensure that the database status is set to Ready.

Set the status of all other databases to Offline. While content databases are offline, new site collections cannot be created. However, existing site collections in offline databases are still accessible for both read and write operations.

Create the site collections. They are automatically added to the online database.

Set the status of all other databases back to Ready.

## Site collections

A site collection is a set of Web sites that have the same owner and share administration settings. Each site collection contains a top-level Web site and can contain one or more subsites.

### Capacity

The recommended guideline for acceptable performance is to implement fewer than 50,000 site collections per content database; however, performance can be affected at about 10,000 site collections. Scaling out by distributing site collections across multiple database servers provides additional storage capacity and throughput.

### Sharing and isolation

Site collections introduce several sharing and isolation opportunities that affect permissions, navigation, and feature deployment.

The following items can be shared within a site collection and cannot be shared across site collections (except items that are stored in a file system, such as features in the `_layouts` directory):

- Master pages

- Page layouts

- Images

- Site templates

Additionally, permissions and navigation are isolated at the site collection level in the following ways:

- Subsites within a site collection can inherit permissions from the top-level site.

- Site collections cannot inherit permissions from other site collections.

- There is no built-in navigation from one site collection to another.

Finally, SharePoint Server 2010 aggregates search results across site collections based on a user's permissions, regardless of the number of site collections or databases (depending on search scopes).

It is important to note that although permissions are enforced on individual sites, the sites are still vulnerable to cross-site scripting attacks from other sites within the domain.

### Configurable items

The following table displays configurable items that contribute to isolation and sharing.

Item	Description
Site collection administrator	You can specify one user to be the primary site collection administrator and one user to be the secondary site collection administrator. In Central Administration, you cannot enter more than one account for these roles, nor can you enter a group account for these roles.
Site template	A site template determines which lists and features will be available on your new site. Many aspects of a site can be customized after creation. However, the site template cannot be changed once the site is created.
Quota template	You can apply a quota template to limit resources used for a site collection. The following templates are provided: <ul style="list-style-type: none"> <li>Personal Site (100 MB)</li> <li>Team Sites (2,000 MB)</li> </ul>

The following table displays configurable items within a site collection that contribute to isolation and sharing. These settings are available after you create the site collection using the settings in the previous table.

Item	Description
Site collection administrators	You can specify multiple user accounts to be site collection administrators. You cannot add group accounts.
Permission level	Add user and group accounts to site collections and specify permission levels for each.

### Administration

Site collection creation does not require DNS entries (unless you are creating host-named site collections) and can be easily automated or delegated to users. You can create site collections for your team sites centrally, or you can allow users to create their own site collections by using Self-Service Site Management.

Using a dedicated database for a site collection provides the ability to perform backup and recovery at the site collection level.

### Planning recommendations

Site collections bridge logical architecture and information architecture. When you design your site collections, consider the following two design tasks:

- Design consistent URLs across your organization.

- Create logical divisions of content.

Unless you are using host-named site collections, each Web application must have a single root-level site collection. This provides a single URL path into the sites that reside in the Web application. This is also a requirement if you are implementing multiple zones within a Web application. For more information, see [Host-named site collections](#) in this article.

Many organizations plan to implement multiple site collections within a Web application for use by different teams or divisions within the organization. Common design goals include the following:

- Maintain a separate and independent site collection for each team.

- Create a unique URL for each team.

- Isolate content between teams.

To satisfy these goals, you can use managed paths to incorporate multiple top-level site collections within a Web application. By defining managed paths, you can specify which paths in the URL namespace of a Web application are used for site collections. You can specify that one site collection or more than one site collection exists at a distinct path below the root site. Without managed paths, all sites created below the root site collection are part of the root site collection.

You can create the following two types of managed paths:

**Explicit inclusion:** A site collection with the explicit URL that you assign. An explicit inclusion is applied to only one site collection. You can associate each of these site collections with a different content database if you want to manage growth and to provide the opportunity to back up and restore these sites separately. An example URL for a site collection created by using this method is `http://fabrikam/hr`. The limit on site collections created with an explicit inclusion is approximately 100 site collections within a Web application; however, 20 is a good operational maximum. If your organization requires a greater number of site collections, use a wildcard inclusion instead.

**Wildcard inclusion:** A path that is added to the URL. This path indicates that all sites that are specified directly after the path name are unique site collections. This option is typically used to support Self-Service Site Management, such as My Sites or sites created for partner collaboration. Example URLs for site collections created by using this method are `http://partnerweb/sites/project1` and `http://partnerweb/sites/project2`. In these examples, "http://partnerweb" represents the root-level site collection and "/sites" represents the wildcard inclusion.

## Sites

A site consists one or more related Web pages and other items (such as lists, libraries, and documents) that are hosted inside a site collection.

### Capacity

The guideline for acceptable performance is to implement fewer than 250,000 sites per site collection. You can create a very large total number of Web sites by nesting the subsites. However, a large number of nested subsites can greatly affect the time it takes to upgrade sites. 5,000 sites within a site collection is a good operational target.

### Sharing and isolation

Sites include built-in navigation from one subsite to another within a site collection. There is no built-in navigation from one site collection to another.

As with site collections, separate sites are vulnerable to cross-site scripting attacks from other sites within the domain.

### Configurable elements

From within each site, you can add user or group accounts to the Owners group for that site.

## Administration

You can use a variety of tools to back up and restore individual sites.

### Host-named site collections

Host-named site collections are an option if you want to create multiple root-level site collections within a Web application. For example, administrators for hosting organizations use host-named site collections to create multiple domain-named sites.

There is no special mode, such as host header mode, that is required to create host-named site collections. You create host-named site collections by using Windows PowerShell. Additionally, by using Windows PowerShell, you can use managed paths with host-named site collections (**New-SPManagedPath - HostHeader**).

Host-named site collections give you more control over URLs. However, host-named site collections are only available through the Default zone. User accounts that are configured to authenticate through other zones cannot access host-named site collections.

In SharePoint 2010 Products, host-named site collections support off-box SSL termination. However, only the protocol scheme can be changed off-box (http:// or https://). The reverse proxy server cannot change the host name or the port number (except to switch from the default SSL port to the default HTTP port).

## Capacity

You can create up to 100,000 host-named site collections within a single IIS Web site.

## Sharing and isolation

The independent domain names that result from host-named site collections help prevent cross-site scripting attacks between two sites.

## Administration

Administrative tasks for host-named site collections include the following:

- Add host-named site collections by using Windows PowerShell.

- Each host-named site collection requires a separate DNS entry.

### My Sites

My Sites are special SharePoint sites that are personalized for each user. My Sites are enabled by default as part of the User Profile service, and every user in an organization can create a unique My Site. For information about capacity, sharing and isolation, and administration, see [Sites](#) earlier in this article.