



**Microsoft SharePoint 2010
IT Professional Evaluation Guide**



This document is provided "as-is". Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

© 2010 Microsoft. All rights reserved.

Contents

Abstract.....	1
Introduction	1
Deployment and Upgrade	2
System Requirements	2
Browser Requirements	3
Installation.....	4
Upgrade from Windows SharePoint Services 3.0 and Office SharePoint Server 2007.....	6
Preparing for Upgrade	6
Upgrade Methods.....	7
Downtime Mitigation.....	8
Visual Upgrade	9
Upgrade Logging	9
Upgrade Scenario Example	10
Patch Management Improvements.....	12
SharePoint Central Administration Web Site	12
Service Applications.....	13
Business Connectivity Services.....	16
External Content Types	16
SharePoint Server and Office Integration.....	17
Managed Metadata Service	17
Managed Metadata Service Application.....	17
Example Managed Metadata Service Scenario	18
Search.....	19
Search Architecture	19
Query Architecture.....	19
Crawling Architecture	20
Search Administration.....	20

FAST Search for SharePoint.....	21
Service Applications Conclusion	21
Security	21
Secure Store Services.....	21
Managed Accounts.....	22
Claims-based Authentication	22
Health and Monitoring.....	23
Diagnostics.....	23
Usage and Health Data Collection	25
Reliability and Monitoring	26
Reporting	28
Remote BLOB (Binary Large Objects) Storage	29
Performance Controls.....	30
Resource Throttling	30
Controlling Large List Activities	31
Windows PowerShell Administration	33
Windows PowerShell cmdlet Examples.....	33
Business Continuity Management	34
Central Administration vs. Windows PowerShell.....	34
Granular Backup and Restore.....	35
Back Up a Site Collection	35
Export a Site or List.....	36
Recover Data from an Unattached Content Database.....	37
Scenario: Using Windows PowerShell to Script Backing Up All Site Collections Individually.....	38
Farm Backup and Restore	39
High Availability	39
Read-Only Databases	40

SQL Mirroring.....	40
SQL Snapshots.....	41
Search and Index Backup and Restore	42
Governance	42
Multi-tenancy and Hosting.....	43
Site Subscriptions.....	43
Administration	44
Branding	45
Themes	45
Master Pages.....	46
SharePoint Designer 2010	47
Conclusion	48
Resources.....	49

Abstract

This evaluation guide is designed to give IT Professionals an introduction and overview of the features in Microsoft® SharePoint® 2010 that are most pertinent to installing, managing, and configuring the SharePoint farm. It begins with a brief introduction and a summary of what's new, followed by a more in-depth discussion of the most relevant features.

The ultimate goal of this guide is to provide the IT pro with the understanding necessary for installing and evaluating SharePoint 2010. This guide is intended for the Windows Server® administrator, Windows SharePoint Services 3.0 and Office SharePoint Server 2007 administrator, Web server administrator, or any IT pro involved in server administration.

This evaluation guide is subject to change. For the latest information about SharePoint 2010, go to <http://www.microsoft.com/sharepoint>.

Introduction

Microsoft SharePoint 2010 is the business collaboration platform for the enterprise and the Internet. SharePoint 2010 can be deployed onsite (also called on-premises) or as a hosted service, such as SharePoint Online. It can also be deployed to physical machines or virtualized to support cost reduction, reduce operational burden, and provide server consolidation. Whether on-premises or hosted, virtualized or physically, deploying SharePoint Server requires the IT pro to be heavily involved. Microsoft SharePoint 2010 includes a large number of enhancements and new features for the administrators.

Specifically, SharePoint 2010 provides the IT pro with:

- **Increased productivity** by improving the administrative experience and by giving the administrator deeper operational insight. Increases in productivity are facilitated by a new, streamlined Central Administration Web site, new capabilities to manage and monitor the SharePoint farm, and Windows PowerShell™ support.
- **A scalable unified infrastructure** that includes better control over server resources (for example, to improve the performance and management of large lists) and data management and protection by using high availability. Also

included in SharePoint 2010 is a new scalable services architecture that allows SharePoint administrators to effectively manage and centralize services like Search, My Sites, and Taxonomy. The new services architecture is extensible to allow ISVs to build services and add them to the platform. Resource Throttling is implemented across the platform, from HTTP request, processor and memory usage, to list and document handling to ensure the performance and protect the availability of the server farm.

- **Flexible deployment options** by providing a quick and simple installation and configuration process and a predictable and consistent upgrade path from Windows SharePoint Services 3.0 and Office SharePoint Server 2007. Functionality has also been introduced that allows administrators to manage how users use SharePoint. For example, you can now safely allow site administrators to upload and run custom code within a controlled sandbox environment by using Sandboxed Solutions. Administrators also have more governance control: If you want to maintain a centralized SharePoint deployment versus a more decentralized approach, you can either block those deployments through Group Policy or track them by using Active Directory® marker support.

SharePoint 2010 has a large number of features geared toward IT pros, and a detailed explanation of every feature is beyond the scope of this document. This guide will describe some of the most compelling new and improved SharePoint 2010 IT pro features.

Deployment and Upgrade

System Requirements

To implement a SharePoint 2010 environment, your infrastructure must meet the following minimum requirements:

- Windows Server® 2008 64-bit operating system with Service Pack 2 (SP2) or later, or Windows Server 2008 R2
- Microsoft SQL Server® 2005 64-bit with SP3 and Cumulative Update 3 or later, or SQL Server 2008 64-bit with SP1 and Cumulative Update 2 or later, or SQL Server 2008 R2

For the complete and updated list of prerequisites for installing SharePoint 2010, see "Determine hardware and software requirements (SharePoint 2010)" (<http://go.microsoft.com/fwlink/?LinkId=166546>) on the Microsoft TechNet Web site.

Browser Requirements

SharePoint 2010 supports several commonly used Web browsers. Different browsers offer different levels of support and functionality. Browser support for SharePoint 2010 can be divided into three categories:

- Supported

A supported Web browser is a Web browser that is known to fully tested with all features and functionality to work with SharePoint Server 2010. If you encounter any issues, support can help you to resolve these issues.

- Supported with known limitations

A supported Web browser with known limitations works with most features and functionality, however if there is a feature or functionality that does not work or is disabled by design, documentation on how to resolve these issues is readily available.

- Not tested

A Web browser that is not tested means that its compatibility with SharePoint Server 2010 is untested, and there may be issues with using the particular Web browser.

SharePoint Server 2010 works best with up-to-date, standards-based Web browsers. The following are supported browser running on the Windows® operating system:

- Internet Explorer 7 32-bit
- Internet Explorer 8 32-bit

The following are supported browser options with known limitations:

- Internet Explorer 7 64-bit
- Internet Explorer 8 64-bit
- Firefox 3.6 32-bit on Windows operating systems
- Firefox 3.6 on non-Windows operating systems
- Safari 4.04 on non-Windows operating systems

SharePoint 2010 does not support Internet Explorer 6 for publishing site scenarios (such as Internet facing “.com” sites). The Web Content Management features built into SharePoint Server 2010 provide a deep level of control over the markup and styling of the reader experience. Page designers can use these features to help ensure that the pages they design are compatible with additional browsers, including Internet Explorer 6, for viewing content. However, it is the page designers’ responsibility to create pages that are compatible with the browsers that they want to support.

For full browser support information see <http://go.microsoft.com/fwlink/?LinkId=190341>

Installation

The installation process for SharePoint 2010 has been improved and streamlined in several key areas. To start with, the Microsoft SharePoint Products and Technologies 2010 Preparation Tool has been added to the installation process. Before installing SharePoint 2010, this prerequisite installer can be run to scan the system and ensure that all of the components required to deploy SharePoint Server have been installed. If the scan detects that some or all of these components haven’t been installed, the prerequisite installer will download and install the components needed for the SharePoint 2010 installation. The Microsoft SharePoint Products and Technologies 2010 Preparation Tool can optionally be implemented to leverage software on a local share for circumstances where Internet connectivity is not available.

After the prerequisite components have been installed, administrators have two options for installing SharePoint 2010: Use the graphical user interface (PSConfigUI) or script the installation by using configuration files and Windows PowerShell. PSConfigUI provides a simple step-by-step process where users can specify the configuration they want. Using Windows PowerShell is a bit more involved, but some IT Pros may find it suits their needs better than the PSConfigUI. By scripting the installation, you can ensure that all of your servers running SharePoint Server have identical configurations. Scripted installations are also advantageous from a disaster recovery perspective -if a server goes down, you can reinstall SharePoint Server quickly by using a previously created script. Both methods are fully supported.

After SharePoint 2010 has been installed, there is a new configuration wizard that has been designed to make the configuration process easier by guiding IT Professionals through the steps to configure a new farm as well as specific farm functionality.

The Farm Configuration Wizard, for example, walks administrators through the initial steps of the farm setup process, which includes specifying the accounts that each service will run under and the services that are required in the farm. It is also possible for additional wizards to be added by third-party vendors or by developers to assist with configuring other areas of SharePoint 2010 and if an administrator wants to fully customize the farm configuration, the Farm Configuration Wizard can be skipped in preference to configuring everything manually.

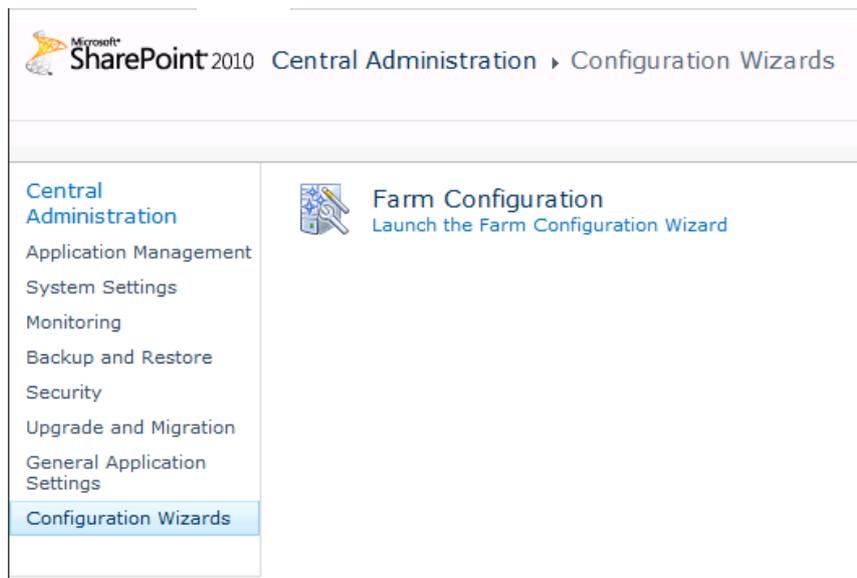


Figure 1: Configuration Wizards page

Another new aspect of the installation process is the addition of a farm passphrase. This is a password that is used to encrypt all communications and credentials across the farm. The farm passphrase is required before you can add a server to the SharePoint farm or remove a server from the farm. The passphrase can be changed through Windows PowerShell by the administrator.

In an effort to reduce administrator overhead, SharePoint 2010 introduces a new concept called managed accounts. A Managed Account is effectively an Active Directory user account whose credentials are managed by and contained within SharePoint. This allows SharePoint 2010 to securely manage its own service accounts and roll (or change) the passwords when required. When a password needs to be rolled, SharePoint 2010 will

respect Active Directory Domain Password Policies when creating a new password for the account.

Other installation and deployment changes allow network administrators to track and manage SharePoint 2010 servers installed and running on their network. During installation, SharePoint 2010 will write a “marker” to a preconfigured container in Active Directory. These markers make it possible for network administrators to track and manage SharePoint deployments across their infrastructure based on reporting from the data stored in Active Directory.

Administrators can also now utilize Windows Group Policy to block SharePoint from being installed on unauthorized servers. This gives administrators much tighter control over their SharePoint environment and significantly increases their ability to enforce established governance policies.

Upgrade from Windows SharePoint Services 3.0 and Office SharePoint Server 2007

Significant investment has been placed into upgrade scenarios for SharePoint 2010 to enable a predictable and smooth experience for the IT Professional. Investments have been made in all aspects of upgrade, from preparing to upgrade to customizing the post-upgrade environment.

Preparing for Upgrade

Upgrade preparation begins in Windows SharePoint Services 3.0 and Office SharePoint Server 2007 Service Pack 2, with the introduction of the **preupgradecheck** STSADM operation. This operation can be run on existing Windows SharePoint Services 3.0 or Office SharePoint Server 2007 farm to report on farm and server data, health and upgrade suitability. It identifies key information, such as:

- Servers and total amount of content
- Search configuration
- Alternate access mappings
- Features
- Site definitions
- Language packs

It also exposes potential issues, including:

- Large lists
- Data orphans
- CAML views and CAML content types
- Modified content databases that have improper schemas
- Missing upgrade dependencies

All of this is collected and compiled into a report that can be viewed using a browser. The `preupgradecheckprocess` is read-only; it does not make any changes to the existing SharePoint environment. Therefore, it can be run often, not only before upgrading but also to simply check the general health of the SharePoint environment.

Administrators can also attach SharePoint content databases to a SharePoint 2010 farm, and then run the **Test-SPContentDatabase** PowerShell cmdlet against the databases. This cmdlet will test the specified database against the specified Web application, identifying any current or potential issues, such as:

- Data orphans
- Missing site definitions
- Missing features
- Missing assemblies

This cmdlet is meant to complement the pre-upgrade checker report and is also read-only, preventing any disruption of the data while testing for issues.

Upgrade Methods

There are two primary methods for upgrading from Windows SharePoint Services 3.0 and Office SharePoint Server 2007 to SharePoint 2010: in-place and database attach. The in-place method is used to upgrade an existing Office SharePoint Server 2007 server to SharePoint 2010, and the process can be restarted as necessary if issues arise. The database attach method allows you to backup an Windows SharePoint Services 3.0 or Office SharePoint Server 2007 database and attach it to a SharePoint 2010 Web application. SharePoint Server then upgrades this database and makes it available through the Web application.

Additionally, Basic installations (or "single-click installations") can be upgraded by using an in-place upgrade, and then migrated from a Windows Internal Database deployment to take advantage of remote BLOB storage (RBS, covered later in this guide). This is useful for scenarios in which Windows SharePoint Services 3.0 has been installed with

Windows Internal Database, as SharePoint Foundation 2010 ships with SQL Server Express with a limit of 4GB per database. Large BLOB (binary large object) objects can be stored in SQL FILESTREAM on disk drives rather than inside the content database. Utilizing the BLOB storage capability in SharePoint 2010, the content database size can be kept under the size limitation.

Downtime Mitigation

Several features have been introduced in SharePoint 2010 to allow for faster upgrades with little to no server downtime.

The first feature is the use of read-only databases, which was originally made available in Windows SharePoint Services 3.0 and Office SharePoint Server 2007 Service Pack 2. During an upgrade or update, a content database in a backup farm can be marked as read-only within SQL Server, preventing any changes to the content while keeping the backup site live to provide read only access to users of the site. SharePoint 2010 recognizes that a content database has been made read-only from within SQL Server and deactivates all of areas of the user interface to provide a read only end user view. This allows users read access to the content while the upgrade takes place.

Another significant improvement to reduce the amount of time an upgrade takes is support for several databases-attach upgrades at the same time. Through the use of multiple Windows PowerShell sessions, multiple databases can be upgraded in parallel, which means the amount of data upgraded at one time is limited only by your SQL Server resources.

Finally, for customers who have such a large amount of content that completing the upgrade within a reasonable time window is not possible, there is another option. This option involves using alternate-access mapping redirections to direct traffic between a SharePoint 2010 farm and Windows SharePoint Services 3.0 or Office SharePoint Server 2007 farm based on the requested URL by using client-side 302 redirects. This allows a more gradual phased approach to upgrade and can allow the time required to upgrade very large content databases without incurring extended server or service downtime.

For more information on SharePoint 2010 upgrade see

<http://go.microsoft.com/fwlink/?LinkID=189356>

Visual Upgrade

To mitigate the impact of the upgrade on users, SharePoint 2010 supports Windows SharePoint Services 3.0 and Office SharePoint Server 2007 master pages and cascading style sheets. By default, after a content database is upgraded, the sites will be displayed with the Windows SharePoint Services 3.0 and Office SharePoint Server 2007 visuals, giving users the familiar look and feel they're accustomed to.

Visually, an upgraded site can then exist in one of three states: Windows SharePoint Services 3.0/Office SharePoint Server 2007, SharePoint 2010 preview mode, and SharePoint 2010. Using the SharePoint 2010 preview mode allows site administrators to preview how the site looks with the SharePoint 2010 user interface before committing to it. This setting is at the site level, allowing for a very granular, flexible experience.

Utilizing the gradual visual upgrade approach allows IT Professionals to take advantage of all of the platform management improvements while planning and managing the end user training and customization updates that may be required within their organization.

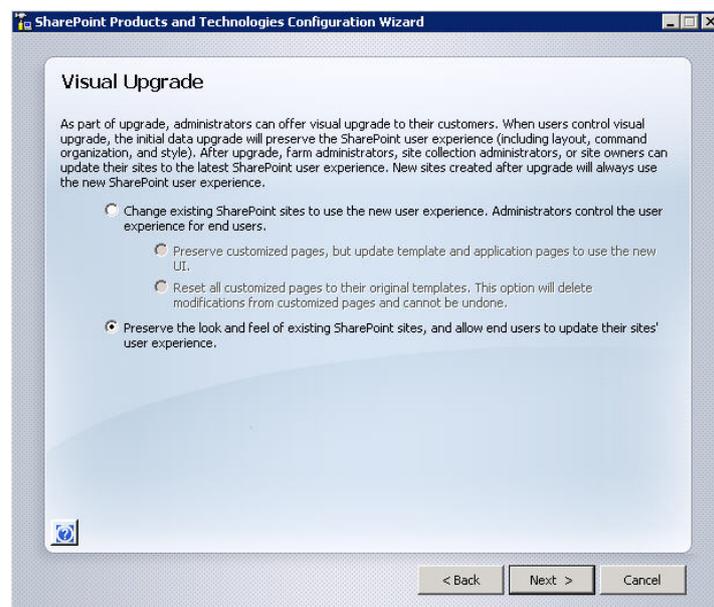


Figure 2: Visual Upgrade Options

Upgrade Logging

The logging capabilities have been expanded and standardized, allowing for easier, more consistent reporting on the upgrade process. This includes the creation of a

unique log for each upgrade. Also, an errors-only log is generated, which reduces the need to hunt through the full upgrade log to discover issues.

Upgrade Scenario Example

To better understand the upgrade process, consider the following example scenario. Randy has been tasked with upgrading his Office SharePoint Server 2007 farm to SharePoint 2010 while incurring as little downtime as possible. It will be acceptable for the data to be read-only during the upgrade. Randy will be deploying SharePoint 2010 on new hardware and will be performing the upgrade by using the database attach method.

To begin the process, Randy has tested and updated the server running Office SharePoint Server 2007 with Service Pack 2 and all applicable cumulative updates. Now he can run **Stsadm.exe -o preupgradecheck**. He uses the output to verify that his farm is ready to be upgraded. He does discover some issues where he is not following SharePoint best practices regarding large lists, and is able to examine these lists and find solutions. Because **preupgradecheck** is a read-only tool, it is safe for him to run it often, confirming his progress as he works to mitigate the issues that have been raised.

Now that Randy has his databases in good shape, he begins building SharePoint 2010 on his new hardware. After he has established this farm, configured all necessary service applications, and provisioned his required Web applications, he can begin the process of testing an upgrade. Because he has chosen the database attach method, he performs a backup of his database from production and restores it to his new server running SQL Server. On his new server farm, he takes advantage of the Windows PowerShell cmdlet **Test-SPContentDatabase** to report any potential issues with attaching the content database to his newly provisioned Web application. From the output of this command Randy finds he has not installed all of the necessary Features on the new SharePoint server that are referenced in the database he wants to attach. He installs the necessary Features, avoiding a failed upgrade.

With all of the necessary pieces now in place, Randy performs his first upgrade by using the Windows PowerShell cmdlet **Upgrade-SPContentDatabase**. After the upgrade concludes, he reviews the errors-only upgrade log and then the full upgrade log, which gives him both a focused picture of any issues that occurred and the line-by-line story that he can review to get a better understanding of what happened during the upgrade process.

With the upgrade cmdlet successfully completed, he now can open the site and review his content. The first thing he notices is that his site looks exactly as it did in the Office SharePoint Server 2007 farm, thanks to the Visual Upgrade features. To see what his site will look like with the SharePoint 2010 look and feel, he goes into Site Settings and selects the preview option. This gives Randy the opportunity to confirm that everything looks appropriate with the new visuals applied. If he is happy with the results, he can make the change permanent; if he feels he will need to make adjustments to the site, he can turn preview off and return to the Office SharePoint Server 2007 interface. The feature is set at the individual site level, allowing Randy to make very granular choices. He can also use Windows PowerShell to script the changing of the visuals.

Now Randy has finished the testing of database upgrade. He can start to upgrade the production content. After he finishes working with the various business content owners to validate the upgrade process and works through any issues, he is ready to plan his production upgrade schedule. His plan is to notify his users of the impending upgrade, and then set his Office SharePoint Server 2007 farm's content databases to read-only from within SQL Server. Not only will this set the data to read-only, preventing any data inconsistency during the upgrade window, it will trim the SharePoint user interface by taking away any New or Edit functionality for all users, which helps avoid confusion. After all databases are read-only, Randy performs a backup of the databases and restores them to his new server running SQL Server. With the databases in place he can then begin attaching the databases by using Windows PowerShell. During testing, he confirmed the new hardware was powerful enough to open multiple Windows PowerShell windows, each one upgrading a different database, all at the same time. This allows him to shorten his upgrade window. Randy can also check the progress of upgrade through these Windows PowerShell windows or from within SharePoint 2010's Central Administration interface.

With all of his company's SharePoint content upgraded and available, Randy then has DNS updated to resolve all of his Web applications to the new farm. His content owners confirm that the production upgrade was successful and switch their visuals to SharePoint 2010 as appropriate. Each content owner had previously determined the necessary course of action for moving to the new visuals and the time frame for making the change. Randy has set a hard date of 30 days before he will use Windows PowerShell to force all visuals to the SharePoint 2010 look and feel.

Randy has had a very successful upgrade to SharePoint 2010 because he utilized out-of-the-box SharePoint tools and performed lots of planning and testing. Because he worked out potential issues during the testing phase, he was able to accomplish the upgrade with minimal impact on his business users.

Patch Management Improvements

While upgrade is used to move from version to version, updating (also called "patching") uses the same principles to move from build to build. To allow control and flexibility of the update process, several capabilities are available. The UpdateManagement user interface allows for the monitoring of updates throughout the farm. There are also UpdateStatus health rules that will alert an administrator to inconsistencies within the farm. SharePoint 2010 also supports backward compatibility of updates. Administrators can apply updates to the servers in the farm without applying the updates to the databases for a temporary period. This allows the scheduling of downtime for the database upgrade portion of the update process, or the use of the downtime mitigation technique (using a read-only database to avoid an outage) while rolling in updates.

SharePoint Central Administration Web Site

The Central Administration Web site has been redesigned in SharePoint 2010 to provide a more familiar experience and make it easier for IT Professionals to find what they are looking for. The home page for Central Administration groups major functional areas together and lists many of the most commonly used tasks under each area. Each of the major areas (Application Management, Monitoring, and so on) is represented on the home page and can be accessed by clicking its name or by clicking the corresponding link from the navigation at the left side of the page.

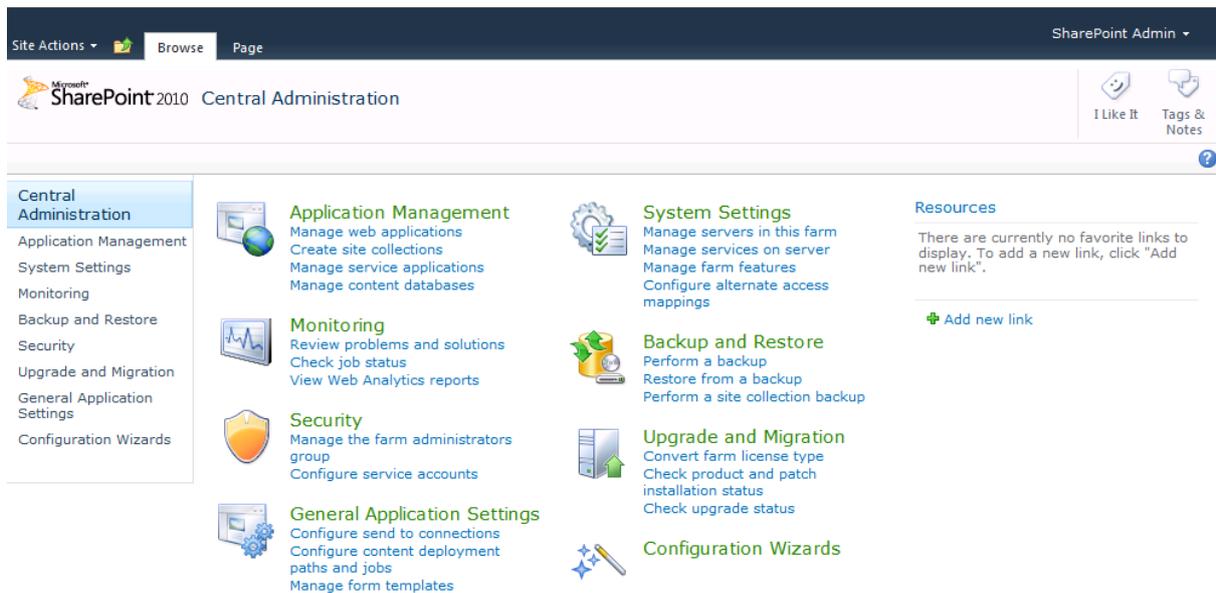


Figure 3:Central Administration

Although the layout has changed, IT Professionals will find many of the menus familiar, with some new options and functionality available. The Ribbon UI makes it easier for administrators to view or change details by making common configuration options a single click away. For example, in Office SharePoint Server 2007, many of the tasks related to Web applications required the user to click a different menu and reselect the Web application each time. Now, administrators can simply select a Web application from a list, and access all of the configuration options for managing that Web application with a single click by using the Ribbon UI.

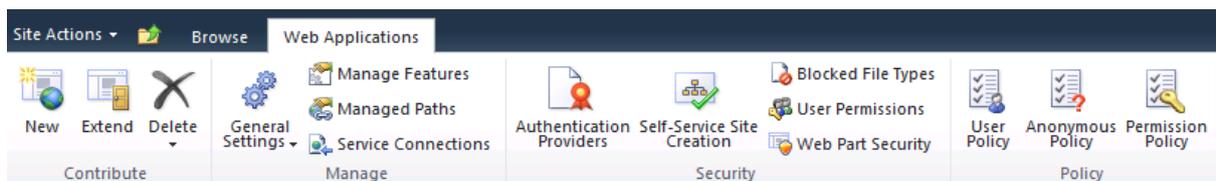


Figure 4:Ribbon Interface

Service Applications

SharePoint 2010 implements a new, more flexible, shared service model called Service Applications. This new architecture provides the foundation for all of the services that

can be shared within and between SharePoint farms. The service application architecture is also built into Microsoft SharePoint Foundation 2010, in contrast to the Shared Services Provider (SSP) architecture, which was available only in Office SharePoint Server 2007. The figures below show the Applications Management page in Central Administration and the Service Applications management page, respectively.

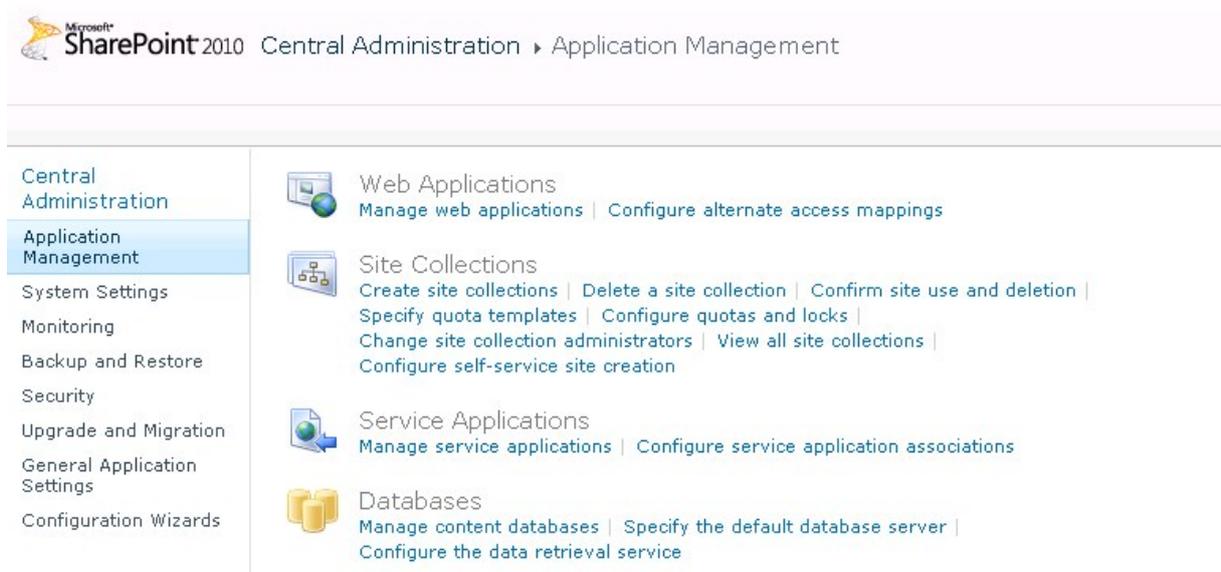


Figure 5: Application Management page

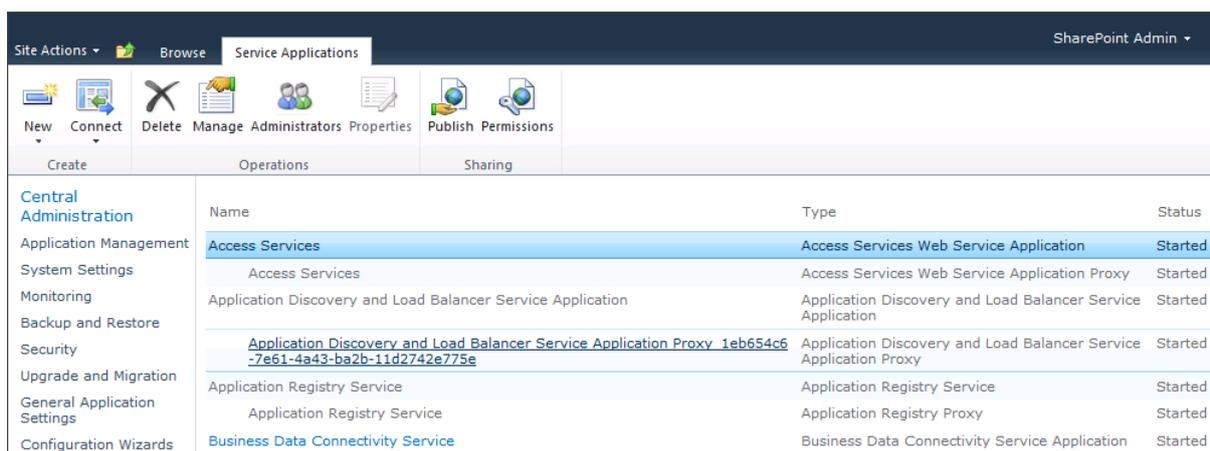


Figure 6: Manage Service Applications page

A key advantage of service applications in SharePoint 2010 is that they are very granular in terms of the services they provide. An administrator can pick and choose only those Service Applications required for the Web Application being configured, eliminating the

need for configuring unnecessary services and the overhead that they carry. In addition, the same service application can be configured differently in different Web applications; therefore, Web sites can be configured to use only the services that are needed, rather than the entire bank of available services. This granularity also extends to the way service applications use server resources such as IIS and SQL, and therefore allows making optimizing them for specific deployment scenarios more straightforward.

Several service applications can also be published, which enables them to be shared across server farms. This allows a very flexible deployment topology, in which some services can be run in a central farm and consumed from regional locations and others can be run locally.

The service application security model allows administrators deep control over service management. Farm administrators can specify administrators for specific service applications as well as set permissions for specific features within service applications.

The new services application architecture enables several new deployment topology options. A specific farm can connect to other farms to consume cross-farm services in a way that makes sense for the restrictions of network infrastructure and other deployment considerations. For example, in a large environment, an entire enterprise services farm (a farm that hosts the most commonly used cross-farm services) could be deployed. Or, a dedicated farm could be implemented and optimized to provide the services of a specific service application (Search, for instance), which could then be consumed by all other SharePoint farms in the organization.

Additional improvements for the service application model include:

- The architecture is extensible, allowing third-party companies to build and add services to the platform.
- Services are managed directly in Central Administration (rather than a separate administration site).
- Services can be monitored and managed remotely.
- Services can be managed and scripted by Windows PowerShell.
- Shared services communications take place over HTTP(S).
- Most services applications are built on the Windows Communications Framework. They have optimization built into their protocol, using binary streams instead of XML for data transfer. Test results show improvements in network throughput with this change.

Three examples of new service applications are described in the remainder of this section: Business Connectivity Services, the Managed Metadata Service, and Search.

Business Connectivity Services

Business Connectivity Services (BCS) provide the capabilities to connect SharePoint 2010 and Office 2010 client applications to external data sources (such as SQL, Oracle, SAP Web services, or custom data sets). BCS is an update to the Business Data Catalog introduced in Office SharePoint Server 2007 and introduces several new capabilities including full CRUD (Create, Rename, Update and Delete) operations on the external data, and the ability to create a new type of SharePoint list, called External Lists, which reference external data rather than data stored within SharePoint.

Configuration and administration is primarily accomplished through the service application pages within Central Administration, along with the Secure Store Service application used for deployments where BCS connects to external data sources, and requires per-user authentication. Because BCS is built on the service application architecture, it can be configured to run in multiple instances within the same farm, each configured independently, each managed by different administrators, and each connected to the same or different SharePoint web applications.

External Content Types

Business Data Connectivity is built around the concept of application models. An application model describes an external data source and is a collection of metadata that provides SharePoint with the information it needs to translate operations requested by SharePoint, or a client application, into requests that are specific to the external data.

At a basic level, an application model describes the methods SharePoint utilizes in order to execute to create, read, update and delete a record. It also defines what the data actually is - a customer or an order, for example, along with the actions a user can take on the data. The application model is defined in an XML document and contains descriptions of one or more External Content Types.

External content types are created by using Microsoft SharePoint® Designer 2010 or Microsoft Visual Studio® 2010, and also by importing an application model that contains one or more external content types into a BCS service application.

SharePoint Server and Office Integration

BCS data can be accessed through SharePoint sites using several BCS Web Parts, external lists and search, making it available for use in many scenarios and solutions. BCS data can also be inserted into Microsoft Word documents through content controls, synchronized into Outlook in the form of contacts and tasks, and taken offline using the Microsoft SharePoint Workspace client application (the replacement for the Microsoft Groove client application). External content types can also be consumed natively by Microsoft InfoPath 2010, Microsoft Access 2010, and other Office applications via custom code.

For more information on Business Connectivity Services see

<http://go.microsoft.com/fwlink/?LinkID=190217>

Managed Metadata Service

The managed metadata service publishes taxonomy and content type definitions for consumption within and between farms. It enables organizations to share a common taxonomy for tagging content inside and outside of SharePoint as well as allowing the wide reuse of content type definitions.

Managed Metadata Service Application

The service application is configured to use a specific site collection as the content type hub and will publish the content types defined in that site collection for reuse within and outside of the farm. The taxonomy hierarchy and nodes are stored in a term store that is configured within the service application. The term store stores all of the terms within the taxonomy and has a flexible set of configuration parameters to allow delegated administration of specific term sets and rich controls allow for defining synonyms within a term set.

Again, because the Managed Metadata service is built on the service application architecture it is possible to create multiple managed metadata service applications with different configurations and administrators. This provides the capability to share multiple term stores and content types from multiple site collections.

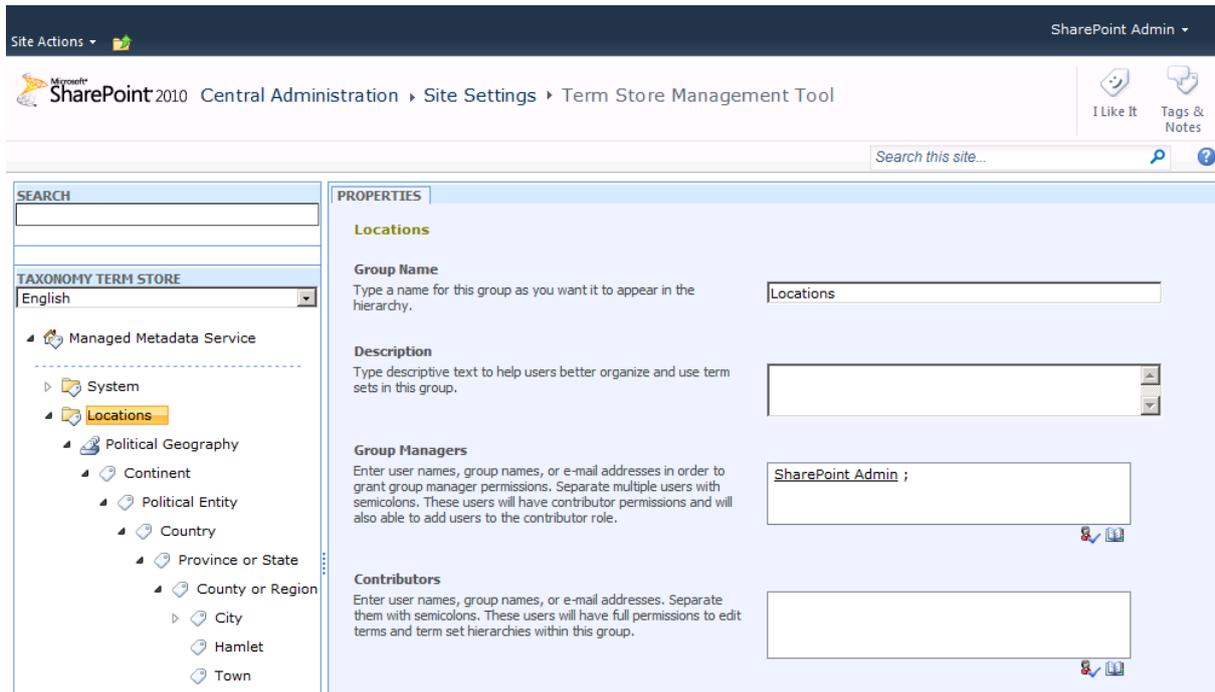


Figure 7:Manage Terms

Example Managed Metadata Service Scenario

Consider a company that has four departments: IT, HR, Products, and Legal. IT, HR, and Legal each have their own SharePoint site collection that serves as each department’s portal, while there are several product team site collections, one for each product team. All these site collections are contained in the same Web application. Two governance requirements have been specified for managing metadata: All documents that are created must include a core set of properties, and all keywords must be stored centrally.

To meet the first requirement, IT creates a content type called Document-Base at the root site collection of its department portal. IT adds columns to Document-Base for all of the required properties. Then IT creates a managed metadata service application and specifies the root site collection as the hub of the content types it is sharing. IT publishes the service and provides the service’s URL to all departments. A connection to IT’s managed metadata service is created. The second requirement — that all keywords be stored centrally — can be satisfied when each site collection connects to the service hub by specifying that managed metadata service as the default keyword store.

This scenario illustrates how the metadata service can be configured so that all departments have access to a centrally managed set of metadata defined in the

Document-Base content type, and all keywords can be centrally managed and stored in a single location (IT's term store), so all departments can use all of the keywords.

Search

SharePoint 2010 Search includes new features and a new architecture that enables a more scalable topology and high availability options. Search capability continues to be pervasive across the platform and is integrated everywhere - from the new social networking features through the Enterprise Content Management legal holds, and discovery.

Search in SharePoint 2010 has been developed as a set of service applications that provide highly configurable topology options to support many search deployment scenarios, both for SharePoint 2010 Search and FAST Search for SharePoint 2010.

Search Architecture

Search in SharePoint 2010 has been re-architected to allow greater redundancy within a single farm, and includes improvements to scaling search up and out. The query architecture and the crawling architecture can be scaled out separately, based on the needs of an organization, thus providing greater flexibility and robust redundancy.

From an administrator perspective, one of the most obvious changes is the support for multiple indexers. In Office SharePoint Server 2007, an SSP could only be configured to use a single indexer. With SharePoint Server 2010, administrator can scale out the number of crawl components by adding additional servers to the farm and configuring them as crawlers. This enables administrator to increase crawl frequency, volume, and performance by distributing the crawl load among several servers, along with providing indexer redundancy if a server fails.

Query Architecture

The query architecture is made up of three components: query servers, index partitions (which reside on query servers), and property databases. An index partition represents a portion of the entire index; therefore, the index is the aggregation of all index partitions. Partitioning the index allows different portions of the index to be spread across query servers. Administrators decide on the number and configuration of each of the partitions. At least one server in a farm must host the query role, and more query servers can be added to increase performance. Two or more query servers provide redundancy based on the configuration of index partitions. For example, a farm with three query

servers can be configured so that each query server has an index partition that represents one-third of the index. Redundancy for the query servers can be achieved by creating a second instance of each index partition on another query server. Deploying index partitions across query servers can help balance the query-processing load, provide redundancy, and increase query performance.

A query server receives a query and forwards the request to all other query servers to process (across all index partitions). The query server then merges the results to display to users.

Crawling Architecture

The crawl server hosts the crawling architecture, which includes crawlers, crawl databases, and property databases. The search architecture can be scaled out based on crawl volume and performance requirements. Each crawler is associated with a crawl database, and the crawled content and history are stored in the crawl database. Multiple crawlers can be used to crawl different content simultaneously. This improves performance and can also provide redundancy. Crawlers reside on crawl servers, populate index partitions, and propagate the partitions to query servers. Property information is stored in the property database. The number of property databases depends on the volume of content that is crawled and the amount of metadata that is associated with the content.

The crawl component must be hosted on at least one server in the farm. Two or more crawl servers provide redundancy based on how crawlers are associated with crawl databases. Additional crawl servers can be added to increase performance and to scale for capacity.

Search Administration

Search management has been streamlined by consolidating search administration to a single dashboard in Central Administration. In addition, many search-related configuration and administration tasks can be scripted with Windows PowerShell.

Search performance and functionality monitoring has also been improved through the built-in search analytics and reporting engine which provides administrators with very granular insight to every aspect of search to aid performance tuning and capacity planning.

In addition to the builtin tools,SharePoint 2010 also featuresrich support for System Center Operations Manager (SCOM) monitoring and alerting.

For more information, see the SharePoint Server 2010 Enterprise Search Evaluation Guide:

<http://go.microsoft.com/fwlink/?LinkID=189521>

FAST Search for SharePoint

FAST Search Server 2010 combines the power of FAST with SharePoint.FAST Search Server 2010 for SharePoint delivers a highly scalable search experience and platform for building search-driven applications with granular control of user experience and relevance, powerful content processing capabilities, and platform flexibility and scaling to handle the most challenging search deployments and applications.

For more information, see the FAST Search Server 2010 for SharePoint Evaluation Guide:

<http://go.microsoft.com/fwlink/?LinkID=189520>

Service Applications Conclusion

The Service application examples discussed above are only a small set of all the Service applications available in SharePoint Server 2010.These Service applications are a modular set of services that will provide SharePoint administrators with needed flexibility to provide only necessary functionality to Web applications within the farm or the enterprise. They can be scaled out to achieve maximum performance or combined into groups to maximize resources. Because each one can be assigned and administered separately, SharePoint administrators will be able to delegate as appropriate.

Security

SharePoint 2010introduces several new and improved security features, from a new security model in the form of Claims-basedAuthentication, to a new Single-Sign-On service application called Secure Store,right through to changes in the way SharePoint manages and controls the credentials for its own service accounts.

Secure Store Services

The new Secure Store Service is an authorization service that replaces Single Sign-On (SSO) in SharePoint Server 2007.Secure Store securely holds users' usernames and passwords for applications outside of SharePoint that SharePoint can integrate with.The

Secure Store enables SharePoint 2010 to connect to external data sources through BCS with an individual's credentials, and is programmatically accessible for developers to leverage in solutions built on top of SharePoint.

Managed Accounts

A Managed Account is effectively an Active Directory user account whose credentials are managed by SharePoint and used for service accounts. The introduction of Managed Accounts helps to reduce the load of managing various service accounts in SharePoint 2010. Much like managed accounts in Windows Server 2008, this new feature allows SharePoint to take control of all the service accounts the farm uses to run its various components. After SharePoint has control of these accounts, it can either manage passwords — automatically changing them as necessary following AD password policy — or it can notify the administrator when an accounts password is about to expire, allowing the change to be made manually. This functionality will greatly reduce the burden on the SharePoint farm administrator who is responsible for maintaining these various accounts and should prevent unnecessary downtime due to expired accounts.

Claims-based Authentication

SharePoint 2010 incorporates a new, more powerful and flexible authentication model called Claims Based Authentication that works with any corporate identity system, including Active Directory, LDAPv3-based directories, application-specific databases, as well as new user-centric identity models, such as Windows LiveID. This model uses claims-based authentication and a new product called the Windows Identity Foundation (WIF). Claims authentication utilizes the concept of an identity, and is based on standard protocols: WS-Federation, WS-Trust, and the Security Assertion Markup Language (SAML).

An identity is a set of information about a user, such as name, e-mail address, department, and other identifying factors. Identity drives very important aspects of an application, such as identifying who the user is (authentication), what permissions the user is granted (authorization), and how the application interacts with the user (personalization). Today, all applications work with identity in some form, but usually in their own unique ways. Leveraging claims based authentication will allow organizations to secure and share information in new and valuable ways, and no longer be tied to

security groups but able to use other attributes to authorize an individual's access to resources, such as their age and their location, for example.

Claims-based identity provides a common way for applications to acquire identity information from users inside their organization, from other organizations, and on the Internet. Identity information is contained in a security token. A token contains one or more claims about the user. Think of it as metadata about the user that stays with them throughout their session.

Claims-based authentication opens the door to great possibilities in SharePoint 2010 and is a deep and expansive topic. For more information on Claims-based authentication, see <http://go.microsoft.com/fwlink/?LinkID=190219>

Health and Monitoring

SharePoint 2010 includes a number of features that provide administrators with tools for monitoring the health and performance of their SharePoint farm. Providing SharePoint Administrators with deep operational insight into server performance and reliability has been a key investment area in this release, putting many new and improved tools into administrator's hands for server diagnostics, reliability and monitoring, and reporting.

Diagnostics

The Unified Logging Service (ULS) is used for diagnosing and tracing events and issues within SharePoint and has been improved in several areas, including new manageability controls, log file readability improvements, and the ability to work with ULS logs through Windows PowerShell scripting.

The ULS log is a significantly more comprehensive source of logging information than it was in SharePoint Server 2007. It is the single repository for SharePoint logging and trace data, as well as data from custom and third-party software through its extensible architecture. Due to the potential increase in log volume, there are several new features that provide administrators with control over both the events that are written to the log and log growth management. The primary method of controlling which events are logged is through Event throttling controls; these allow administrators to configure the severity of events that are captured from a wide range of sources. Event Log Flood Protection (EVFP) can also be enabled to suppress repeating events until conditions return to normal, thereby reducing the size of the log files.

To assist with log file management, log files are compressed by default using NTFS file compression. This reduces their size on disk by up to 50 percent. It is also possible to specify a custom file location for the logs, to define how long they are stored for before being automatically deleted, and the maximum size the log files are allowed to grow. This allows administrators the flexibility of having the data at hand when they need it most, along with the controls to prevent log files from overrunning server storage resources. SharePoint 2010 also includes several Windows PowerShell cmdlets for retrieving information and configuring the ULS.

Consider the following scenario: A user encounters an error and reports it to the SharePoint administrator. The administrator begins researching the error in an attempt to track down and fix the root cause. SharePoint 2010 introduces the concept of Correlation IDs to help with this process. Correlation IDs are identifiers that are internally associated with every request, and are displayed with error messages. The user wrote down the Correlation ID number and gave it to the administrator. Using Windows PowerShell, the administrator is able to track down the same correlation ID in the ULS log and discovers the cause of the error and make the necessary adjustments. This correlation between the ULS log files and the UI helps significantly decrease the amount of time an administrator needs to spend trying to find the cause of a problem and increases the speed at which he can troubleshoot and fix issues within SharePoint.

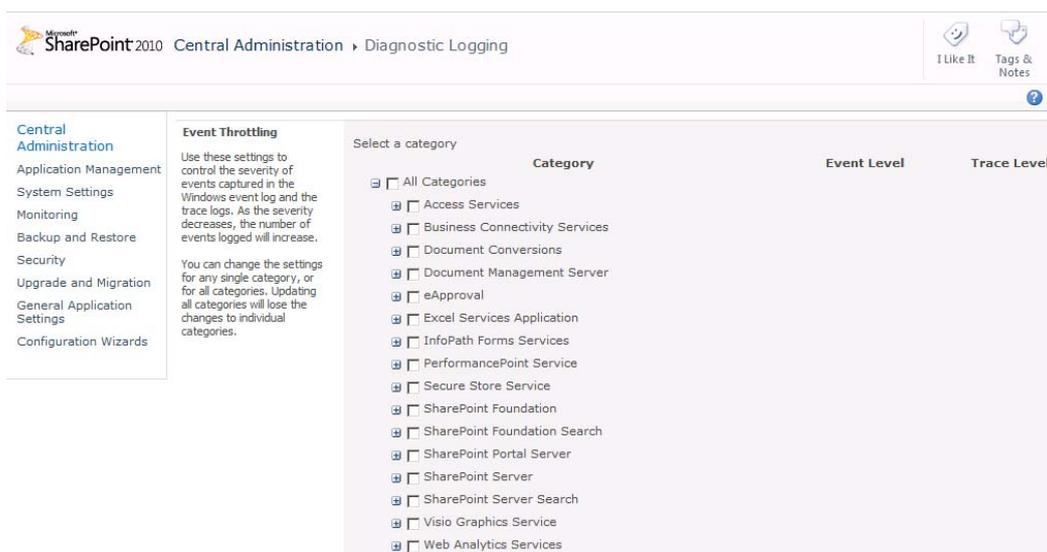


Figure 8:Diagnostic Logging

Usage and Health Data Collection

SharePoint 2010 can also log usage information such as feature usage and performance data to the new usage logging database and log files. This logging is done by the Usage and Health Data Collection service application, and is enabled by default when using the Farm Configuration Wizard during the initial farm set up. Administrators can use the public schema to read and query directly from the usage database to build custom reports.

The types of usage events that are logged are shown in the figure below and include: page requests, feature use, search query usage, site inventory usage, timer jobs, and rating usage.

The screenshot shows the SharePoint 2010 Central Administration console. The main heading is "Configure web analytics and health data collection". A warning message states: "Warning: this page is not encrypted for secure communication. User names, passwords, and any other information will be sent in clear text. For more information, contact your administrator." The page is divided into several sections:

- Usage Data Collection:** Includes a checkbox for "Enable usage data collection" (checked). Below it, under "Events to log:", there are checkboxes for Content Import Usage, Content Export Usage, Page Requests, Feature Use, Search Query Usage, Site Inventory Usage, Timer Jobs, and Rating Usage, all of which are checked.
- Usage Data Collection Settings:** Includes a text box for "Log file location" (C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\14\LOGS\), a "Maximum log file size" input set to 5 GB, and a "Log file location" label.
- Health Data Collection:** Includes a checkbox for "Enable health data collection" (checked) and a link to "Health Logging Schedule".
- Log Collection Schedule:** Includes a link to "Log Collection Schedule".

Figure 9: Usage Logging

Another great addition to the SharePoint administrator's tool belt is the developer dashboard. When enabled, the developer dashboard displays detailed trace information in the footer of every page, which provides both developers and administrators with a wealth of information on everything from the time the page took to be retrieved

from the SQL database stored procedures to the amount of time it took to render and display in the browser. This information, presented right on the page, can be invaluable for administrators working to troubleshoot performance issues, as well as developers working to debug and optimize their code. The developer dashboard is disabled by default and can be enabled for each Web application independently through the use of Windows PowerShell.

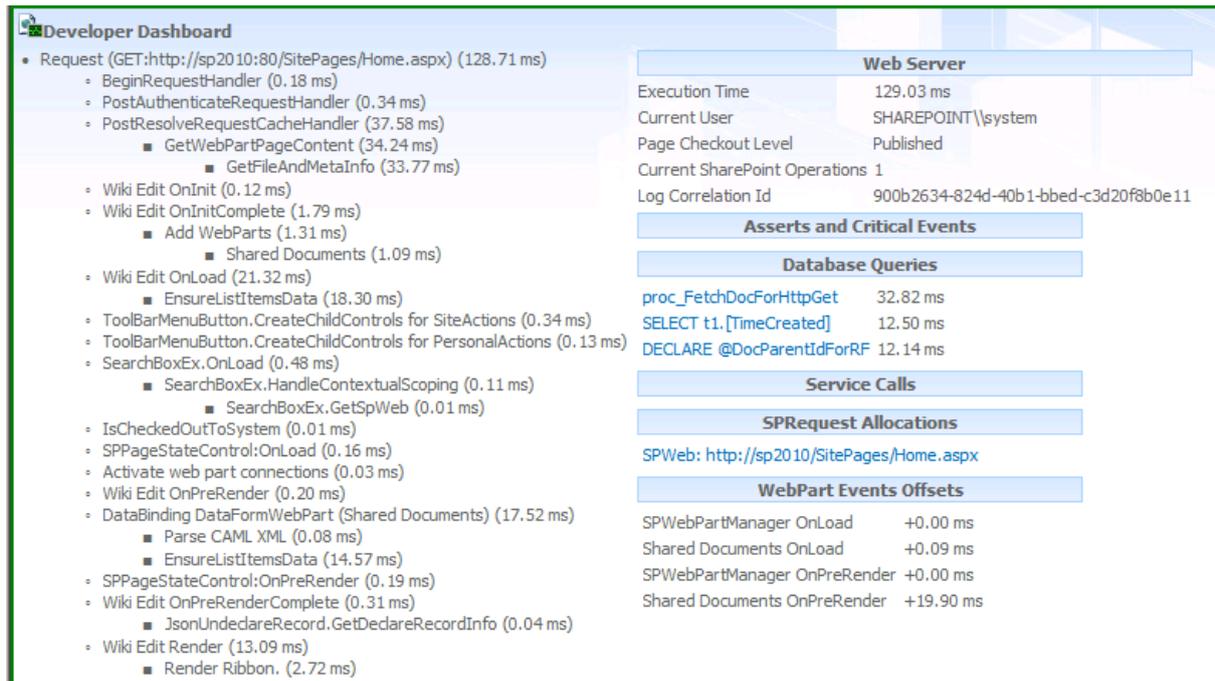


Figure 10: Developer Dashboard

Reliability and Monitoring

The SharePoint Health Analyzer is a new core component of SharePoint 2010 that is built on the principles established by the Best Practices Analyzer that shipped as a Command Line application for Office SharePoint Server 2007.

SharePoint Health Analyzer is tightly woven into SharePoint Central Administration. When the SharePoint Health Analyzer detects problems within the server farm, it prominently displays a message for the administrator on the home page of Central Administration. The administrator can then click a link that describes the problems detected in more detail. SharePoint 2010 includes a set of predefined rules that are used by the Health Analyzer to check the status of the farm's various

components. When issues are discovered, a variety of options exist for fixing the issue. These options include enabling automatic repair, links to additional online help content to help further diagnose and resolve issues, or manually repairing the issue.

Rules are defined in a typical SharePoint list format, meaning you can establish the same set of actions against a Health Rule as you can against a traditional list item. Custom rules can also be developed through APIs to create custom Health Rules that are specific to the administrator's deployment.

The status of a SharePoint farm's Health Analyzer rules are displayed in the Review Problems and Solutions link in the Monitoring section of Central Administration. The rules are also available in the SharePoint 2010 management pack for System Center Operation Manager. SharePoint 2010 ships with more than 50 predefined health rules to help ensure that the SharePoint environment is properly configured and healthy right from the start. An example of a rule definition is shown below.

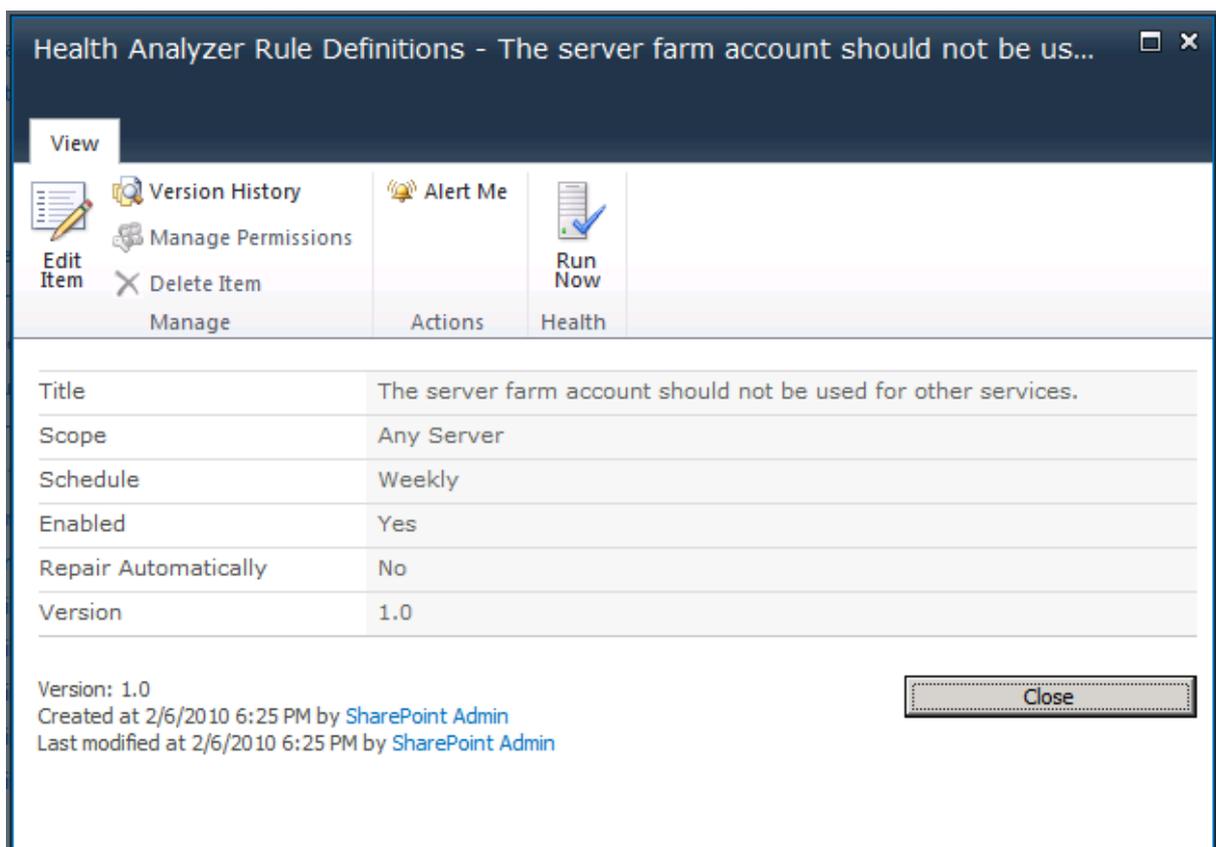


Figure 11:Health Rule

Reporting

Using the health and usage data collected, SharePoint 2010 can create health reports and usage reports that administrators can easily browse. This provides IT pros with a quick snapshot of the overall status of the server farm in an easy to read visual graph.

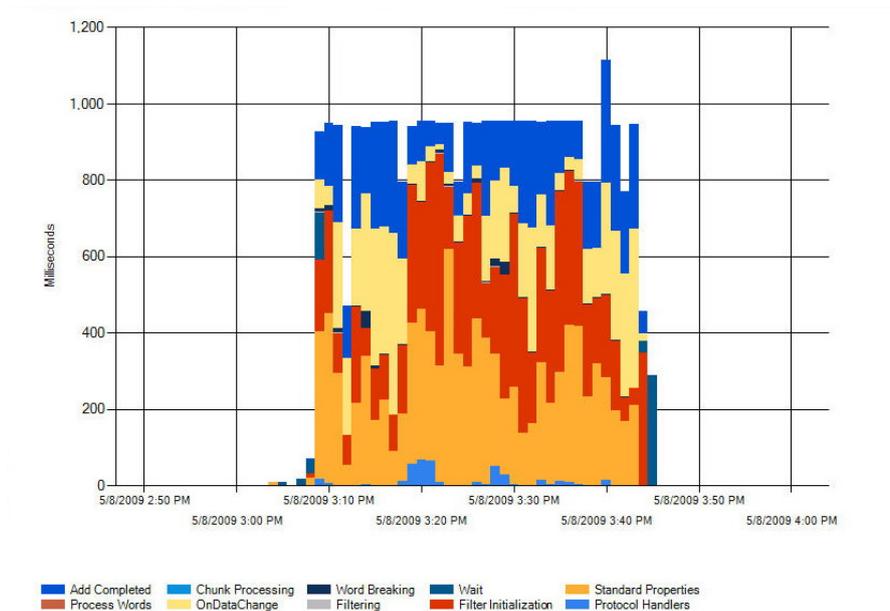


Figure 12:Crawl Activity Report

Web Analytics is also based on the health and usage data collection. Site administrators can use Web Analytics to analyze the trends of user visits to the site and identify the most popular content, which in turn can be utilized to improve the user experience based on the data collected. The report can also be exported into Microsoft Excel for customization and further analysis.

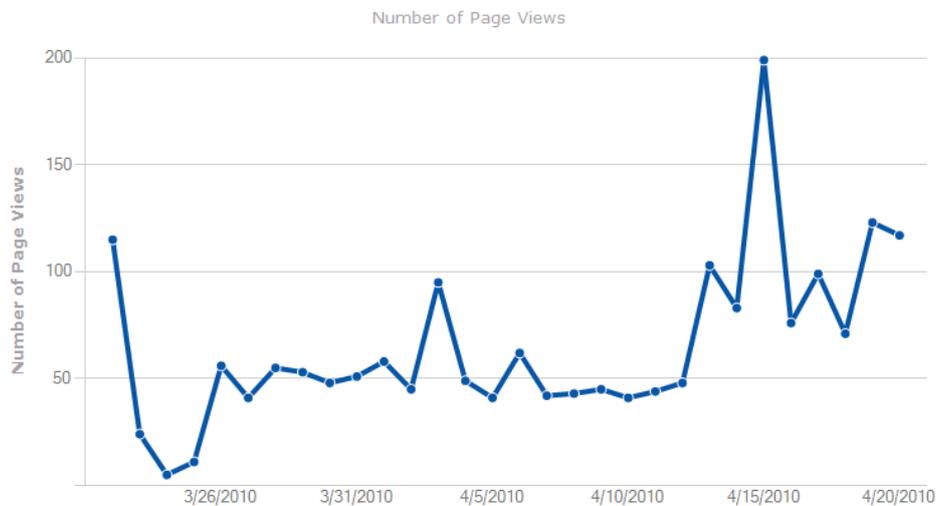


Figure 13:Web Analytics Report

Remote BLOB(Binary Large Objects) Storage

SharePoint 2010 introduces support for SQL Remote BLOB Store (RBS) as an alternative to storing large files directly in SQL. RBS is a framework API found in SQL Server 2008, which enables the storage of files in a location other than SharePoint 2010 content databases. RBS give SharePoint 2010 administrators the option of moving BLOBs (Binary Large Objects) out of SQL Server databases into another location that may provide less expensive storage or that better complies with retention and compliance policies.

The RBS Provider architecture in SharePoint 2010 is pluggable. As with External BLOB Store (EBS) in Windows SharePoint Services 3.0 and Office SharePoint Server 2007, independent software vendors will release RBS providers for use with SharePoint 2010. Microsoft also offers an RBS provider that utilizes the SQL FILESTREAM functionality.

If an RBS provider is installed on an established SharePoint 2010 farm, the BLOBs can be migrated from the SharePoint content database to the RBS provider. Migration of BLOBs is scoped at the content database level. BLOBs can also be migrated out of the remote BLOB store back into a content database. Since SharePoint 2010 supports multiple BLOB stores serving a single farm, BLOBs can also be migrated from one RBS to another. These actions are accomplished by using the Migrate operation of the RBS provider through Windows PowerShell.

Administrators can also control which BLOBs are stored remotely by configuring the `MinimumBlobStorageSize` setting of the RBS provider. This can be used to make sure that smaller documents remain stored in the content database, but larger files, such as video or other media files, are stored remotely.

For more information of how to implement RBS, see <http://go.microsoft.com/fwlink/?LinkID=190218>

Performance Controls

SharePoint 2010 offers several new performance features designed to protect the server from unexpected peaks in demand, and also to prevent, or control, certain user operations that can place a significant load on the server while it processes them. The two primary performance control features are resource throttling for web requests and large list management for lists containing thousands to millions of items.

Resource Throttling

Through Resource Throttling settings, SharePoint 2010 provides a way for administrators to determine the level at which the server will enter throttling mode. Every five seconds, a job runs that checks available server resources compared to configured levels. By default, server CPU usage, available memory, the number of requests in queue, and request wait time are monitored. After three unsuccessful checks, the server enters a throttling period and will remain in this state until a successful check is reported. Requests that were generated prior to the server's entering throttling mode will be completed; this helps prevent users from losing any current work when the server begins to throttle new requests. Any new HTTP GET and Search robot requests will generate a 503 error message and will be logged in the event viewer. While the server is in a throttling period, no new timer jobs will be started.

Throttling can be configured per Web application and can be enabled through the Central Administration user interface or by using a Windows PowerShell cmdlet. For more information on cmdlets, review the Windows PowerShell Administration section below. The following image is an example of the

Get-SPWebApplicationHttpThrottlingMonitor Windows PowerShell cmdlet.

```
PS C:\Users\Administrator> Get-SPWebApplicationHTTPThrottlingMonitor

cmdlet Get-SPWebApplicationHttpThrottlingMonitor at command pipeline position 1
Supply values for the following parameters:
Identity: http://intranet.contoso.com

Category           : Memory
Counter            : Available Mbytes
Instance           :
AssociatedHealthScoreCalculator : [1000.0,500.0,400.0,300.0,200.0,100.0,80.0,60.0,40.0,20.0]

Category           : ASP.NET
Counter            : Requests Current
Instance           :
AssociatedHealthScoreCalculator : [24.0,32.0,40.0,48.0,56.0,64.0,80.0,112.0,130.0,500.0]
```

Figure 14:Windows PowerShell Commands for Monitoring

Controlling Large List Activities

SharePoint 2010 has the ability to support millions of items within a list; however, as lists grow in size, it's easy for users to continue to treat the lists as though they only contain a few hundred items, and use default views such as the "All Items" view. When a list contains hundreds of thousands of items, this can start to put a measurable load on the SQL server every time the list is rendered, as it has to query all of the items in the list in order to sort them before rendering the first page of the "All Items" view. Depending on the overall load of the server at any given moment, this can begin to degrade the response times for other users. To maintain a consistent server response, SharePoint 2010 provides several different settings that can restrict the queries that can be run against a large list. These settings can be configured for each Web application and control the following:

- The number of results that can be returned in a user query. The default value is 5,000 items.
- The warning level for site administrators that their list is close to the threshold limit. This warning is displayed on the list settings page. The default value is 3,000 items.
- The number of query results that can be displayed for a site administrator using object model queries. This allows site administrators to access a larger data set through code than through the browser interface. The default value is 20,000 items.
- The maximum number of unique scopes in a list. The default value is 50,000 items.

- The number of lookup columns that each list can include. The default value is 6 lookup columns.

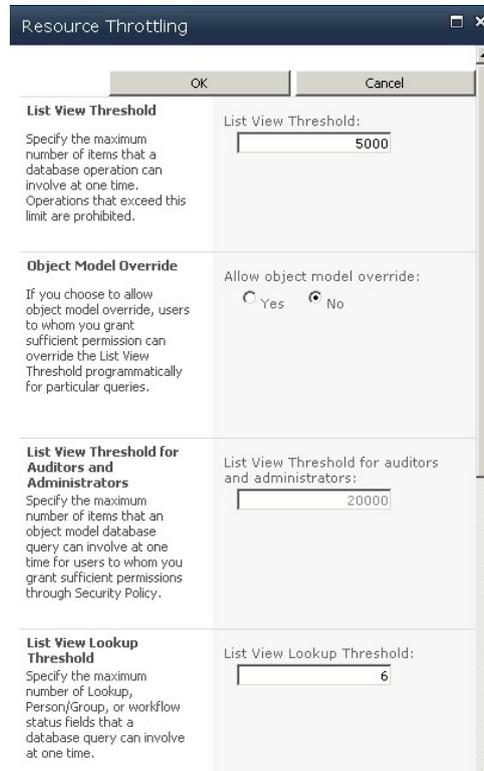


Figure 15:Resource Throttling

Although the settings described above apply to all lists within the Web application, additional settings are available and can be configured to override the Web application settings:

- Allowing object model override will allow for a query to be programmatically written against a list where the query threshold is not applied.
- Enabling a daily time window for large queries provides administrators a way to configure a specific window of time where the query thresholds are not applied.
- A Windows PowerShell command that will restrict specific lists from being queried through the object model (this would cancel out the object model override mentioned above for a specific list instance).

Windows PowerShell Administration

Windows PowerShell is a command-line interface and scripting language specifically designed for system administrators. Windows PowerShell support in SharePoint 2010 enables a new, more productive and flexible, way to manage and configure SharePoint 2010. Although the SharePoint command line tool STSADM.exe still ships with SharePoint 2010 and can be used for many administrative tasks, it is recommended that administrators use Windows PowerShell for the breadth of commands available, the flexibility it provides in terms of a rich scripting environment, and the performance gains when compared to STSADM. Windows PowerShell brings new tools and automation methods to greatly improve control, efficiency, and productivity for SharePoint 2010 administrators.

Windows PowerShell is built around the concept of cmdlets. A cmdlet is a single function that can be used separately or combined with other cmdlets to perform complex tasks and automate administration. Windows PowerShell provides a set of over 100 pre-built cmdlets, and SharePoint ships with over 500 SharePoint-specific cmdlets to allow command line control of nearly every aspect of SharePoint 2010.

Windows PowerShell cmdlet Examples

The following examples highlight some of the areas where SharePoint cmdlets can be utilized in order to accomplish an administrative task quickly and easily:

- Bulk site collection creation - Administrators can use Windows PowerShell to create hundreds of site collections by importing a pre-defined CSV file.
- Scheduled backup - Administrator can use Windows Task Scheduler with a backup script to setup daily backup jobs.
- Parallel upgrade multiple databases - Content database upgrade can be run in parallel by launching multiple instances of upgrade cmdlets in Windows PowerShell.
- Tenant administration creation - Administrators can use Windows PowerShell to create tenant administration sites.
- Bulk enable feature on sites - Features can be enabled on hundreds of sites with a single line of script.

For more information on Windows PowerShell script examples, see <http://go.microsoft.com/fwlink/?LinkId=190342>

Business Continuity Management

SharePoint 2010 provides a broad range of levels for performing backups, which include backing up the entire farm, farm configuration information, site collections, subsites, or lists. Backups can be performed by using Central Administration's backup pages or Windows PowerShell™. By using the backup and restore features available in SharePoint 2010, administrators have a powerful set of tools that allow them to manage their environment with minimal downtime and operational costs.

Central Administration vs. Windows PowerShell

SharePoint 2010 uses two different tools to configure backup and recovery. Central Administration provides a user interface where SharePoint Administrators are prompted via menu structures to select the information to be backed up.

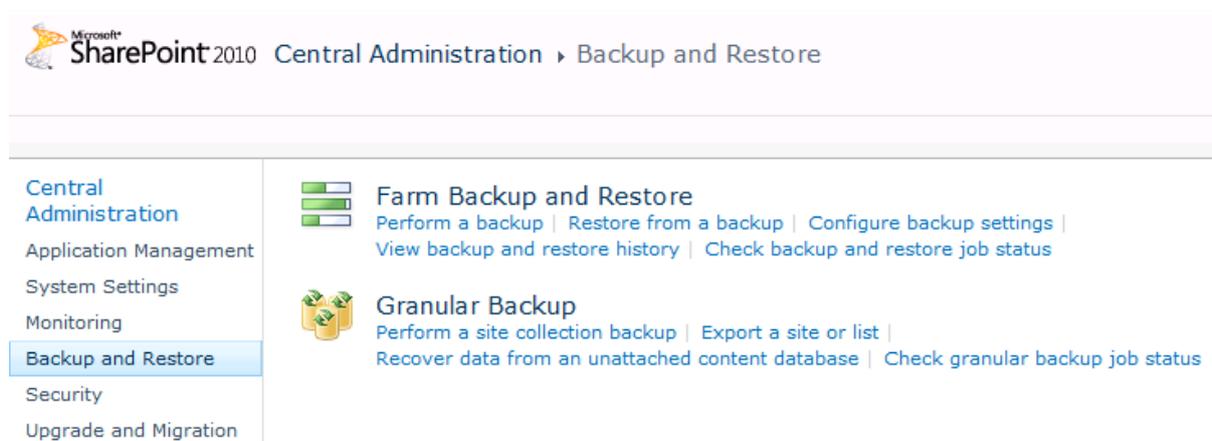


Figure 16: Backup and restore user interface

Windows PowerShell provides SharePoint administrators a way to perform backup and recovery with additional options such as file compression or working with SQL snapshots. Listed below are a few of the benefits available when utilizing Windows PowerShell for backups:

- Windows PowerShell scripts can be developed and scheduled (with Windows Task Scheduler), whereas Central Administration is used for single-use backups and restores.

- Windows PowerShell has the advantage of running against SQLsnapshots instead of the production database. One of the parameters of the Windows PowerShell backup cmdlet will generate a SQL snapshot, against which Windows PowerShell will run the action instead of the production database. This reduces the resource impact of the backup operation on the production environment.
- With Windows PowerShell, SharePoint administrators have more granular control of options for the backup or restore.

The remainder of this section will identify when additional features are made available through Windows PowerShell commands that are not available through the standard user interface.

For more information about Windows PowerShell, refer to the Windows PowerShell Administration section of this guide, and also see <http://go.microsoft.com/fwlink/?LinkId=156748h>

Granular Backup and Restore

SharePoint 2010 provides several new features that provide a granular level of backup for various components of site content. This includes content at the site, subsite, and list level. This level of granular backup provides SharePoint administrators with the ability to quickly and flexibly backup and restore content to meet user demand.

Back Up a Site Collection

Backing up a site collection will include the top level site and all subsites. When backing up the site collection through Central Administration, a SharePoint administrator identifies the site collection and provides a destination for the storage of the backup file.

The screenshot shows a dialog box for backing up a site collection. It has a 'Readiness' section with two status indicators: 'No site collection backup is in progress.' and 'Timer service is running.' Below this are two main sections: 'Site Collection' and 'File location:'. The 'Site Collection' section has a dropdown menu showing 'http://portal.company.com'. The 'File location:' section has a 'Filename:' input field, an 'Overwrite existing file' checkbox, and an example path: '\\backup\SharePoint\Site1.bak'. At the bottom right, there are two buttons: 'Start Backup' and 'Cancel'.

Figure 17:Backup a site collection

The same operation can be completed by using the **Backup-SPSite** Windows PowerShell cmdlet. Also worth noting is that additional parameters are available when using the Backup-SPSite cmdlet, including the ability to use a SQL snapshot.

```

SYNTAX
Backup-SPSite -Identity <SPSitePipeBind> -Path <String> [-AssignmentCollection <SPAssignmentCollection>] [-Force <SwitchParameter>] [-NoSiteLock <SwitchParameter>] [-UseSqlSnapshot <SwitchParameter>] [<CommonParameters>]
  
```

Figure 18:Windows PowerShell command for backing up a site

To restore the site collection, the **Restore-SPSite** Windows PowerShell command is used.

Export a Site or List

Through Central Administration a SharePoint Administrator can configure a backup of a subsite or list. As seen in the image below, after the site is selected, there are additional options for the site and for specific lists. From this page, the administrator can also choose to export security and select the different versions that will be exported with the site or list.

Readiness
 No export is in progress.
 Timer service is running.

<p>Site Collection</p> <p>Select a site or list to export. First select the site collection that contains the site, then select a site to export. To export a list, select the site collection and site that contain it, and then select the list.</p>	<p>Site Collection: <input type="text" value="http://portal.contoso.com"/></p> <p>Site: <input type="text" value="/"/></p> <p>List: <input type="text" value="Shared Documents"/></p>
<p>File location:</p> <p>Specify the destination for the export package.</p>	<p>Filename: <input type="text" value="d:\SP_Exports\SharedDocs.cmp"/></p> <p><input type="checkbox"/> Overwrite existing files</p> <p>Example: \\backup\SharePoint\export.cmp</p>
<p>Export Full Security</p> <p>Export full security of the site, including author, editors, created by times, and modified by times. This also includes all users in the sites.</p>	<p><input checked="" type="checkbox"/> Export full security</p>
<p>Export Versions</p> <p>Select the version history information to include for files and list items. You can include all versions, the last major version, the current version, or the last major and last minor versions.</p>	<p>Export versions: <input type="text" value="All Versions"/></p>

Figure 19:Export a site or list in the user interface

Similar to the site collection backup, Windows PowerShell provides some additional parameters, including the use of SQL snapshots, compression, and logging.

```
SYNTAX
Export-SPWeb [-Identity] <SPWebPipeBind> -Path <String> [-AssignmentCollection <SPAssignmentCollection>] [-CompressionSize <Int32>] [-Confirm <SwitchParameter>] [-Force <SwitchParameter>] [-HaltOnError <SwitchParameter>] [-HaltOnWarning <SwitchParameter>] [-IncludeUserSecurity <SwitchParameter>] [-IncludeVersions <LastMajor ; CurrentVersion ; LastMajorAndMinor ; All>] [-ItemUrl <String>] [-NoFileCompression <SwitchParameter>] [-NoLogFile <SwitchParameter>] [-UseSqlSnapshot <SwitchParameter>] [-WhatIf <SwitchParameter>] [-] [<CommonParameters>]
```

Figure 20:Options for exporting a site or list using Windows PowerShell

To restore the site or list, the **Import-SPWeb** Windows PowerShell command is used.

Recover Data from an Unattached Content Database

Also new to the SharePoint 2010 backup and restore functionality is the ability to restore site collections, sites, and lists from unattached content databases. A backup of a SharePoint content database can be restored to any server with SQL Server installed, and administrators can provide that SQL instance name to SharePoint in Central Administration's backup and restore interface. Administrators can then browse the contents of the database, then backup or export the content needed. This feature can be used to recover data without having to build a second farm for data recovery. The same level of backup granularity is available for both attached and unattached content databases, and the process for selecting the content and options for backup is the same as browsing an attached database. The images below show the process of browsing an unattached content database. Once the backup or export is finished, the exported content can be restored to the appropriate place in their SharePoint site, and the restored database can be deleted.

<p>Database Name and Authentication</p> <p>Specify the content database server and content database name to connect to.</p> <p>Use of Windows authentication is strongly recommended. To use SQL authentication, specify the credentials which will be used to connect to the database.</p>	<p>Database Server <input type="text" value="2010_SQL"/></p> <p>Database Name <input type="text"/></p> <p>Database authentication</p> <p><input checked="" type="radio"/> Windows authentication (recommended)</p> <p><input type="radio"/> SQL authentication</p> <p>Account <input type="text"/></p> <p>Password <input type="text"/></p>
<p>Operation to Perform</p> <p>Select an operation to perform on the content database you selected. You can browse the content of the content database, perform a site collection backup, or export a site or list.</p>	<p>Choose operation:</p> <p><input checked="" type="radio"/> Browse content</p> <p><input type="radio"/> Backup site collection</p> <p><input type="radio"/> Export site or list</p>

Figure 21: Recover data from an unattached database

Figure 22: Browse content from an unattached database

Scenario: Using Windows PowerShell to Script Backing Up All Site Collections Individually

Jennifer is an overly cautious SharePoint administrator who doesn't have complete faith in the database backups that are being performed each night by her database administrators. She would prefer that she also had her own backups, just in case. While reviewing at her options, Jennifer sees that she can do a full farm backup by using the user interface, but she doesn't want to VPN in every night at midnight to start the job —

so that option is out. She could setup a Windows PowerShell script to do the farm backup each night by using **Backup-SPFarm** and then schedule that by using Task Scheduler — but that backup is best served for disaster recovery, and that is what the database administrators are protecting. So she settles on using a Windows PowerShell script that will take advantage of the cmdlet **Backup-SPSite** to backup individual site collections. This will give her a great deal of flexibility for recovery. The script she used is:

```
Get-SPWebApplication | Get-SPSite -Limit all | ForEach-Object {$FilePath = "c:\backups\" + $_.Url.Replace("http://","").Replace("/","-") + ".bak"; Backup-SpSite -Identity $_ -path $FilePath}
```

The simplicity here illustrates the clear benefit of Windows PowerShell. This cmdlet loads all the Web applications in the farm, as well as all the site collections in the web applications, then loops the Backup-SPSite cmdlet through each site collection and saves the backup to the specified file path.

Farm Backup and Restore

SharePoint 2010 has built in capabilities for backing up the entire farm to provide recovery for a catastrophic failure. This interface can be accessed from within Central Administration for ad hoc backups and recovery, or it can be scripted by using the Windows PowerShell cmdlets **Backup-SPFarm** and **Restore-SPFarm**, respectively. A new addition to the catastrophic backup scenario is the capability to do a configuration-only backup.

The configuration-only backup outputs an XML file that can be used to restore all settings in the farm. This could be used for recovery of an existing farm, or to build a different farm entirely by using the same settings. This is possible because all settings that are backed up are not machine-specific.

High Availability

SharePoint 2010 provides rich integration with SQL Server that enables an administrator to leverage the high availability solutions provided by SQL Server. Changes include support for the Failover_Partner keyword, which enables SharePoint 2010 to natively interpret and respond to state changes in a database mirroring topology. In addition to native support for database mirroring, SharePoint 2010 also provides support for SQL Snapshots, read-only content databases, unattached database backup and restore, and configuration database backup and restore provide much more granular control and manageability of the SharePoint farm's data. Some of these features are explained

below. For more information on High Availability planning in SharePoint 2010, see <http://go.microsoft.com/fwlink/?LinkId=189518>

Read-Only Databases

SharePoint 2010 has the ability to recognize read-only SQL databases. Whenever SharePoint Server is connected to a SQL database that has been placed in read-only mode, SharePoint Server security trims all site collections in the database so that they are read-only to all users, removing all add and edit functionality from the sites, even for administrators. This feature can help mitigate downtime during upgrades and server maintenance. For instance, a secondary farm can be created and SQL log shipping can be used to keep it current with the production environment. DNS can then be updated to point to the secondary environment, where all content databases are set to read-only. The production farm can be upgraded or updated without users accessing it. After the upgrade or update is complete and the servers brought back online, DNS can be updated again to point back to the production farm. Using this approach will allow users to have full read access to their data during the upgrade process.

SQL Mirroring

SharePoint 2010 supports SQL mirroring. Farm administrator can configure the SQL environment to mirror a SharePoint content database first. When a new database is created, the SharePoint administrator will be prompted to enter the failover server. SharePoint then communicates with the witness server, and if issues are discovered, SharePoint will failover to the hot backup of the environment.

Database Information

Specify database connection settings for this content database. Use the **Database status** options to control whether or not new Site Collections can be created in the database. When the database status is set to **Ready**, the database is available for hosting new Site Collections. When the database status is set to **Offline**, no new Site Collections can be created.

Database server
serversp

SQL Server database name
WSS_Content_Portal

Database status
Ready

Database Read-Only
No

Database authentication
Windows authentication

Database Versioning and Upgrade

Use this section to check the version and upgrade status of this database. If the Current SharePoint Database Schema Version is less than the Maximum SharePoint Database Schema Version, the database should be upgraded as soon as possible.

Database Schema Versions

Microsoft.SharePoint.Upgrade.SPContentDatabaseSequence
Current Schema Version: 4.0.144.0, Maximum Schema Version: 4.0.144.0

Microsoft.SharePoint.Upgrade.SPContentDatabaseSequence2
Current Schema Version: 4.0.8.0, Maximum Schema Version: 4.0.8.0

Microsoft.SharePoint.Administration.SPContentDatabase
Current Schema Version: 14.0.4730.1010, Maximum Schema Version: 14.0.4730.1010

Failover Server

You can choose to associate a database with a specific failover server that is used in conjunction with SQL Server database mirroring.

Failover Database Server

Database Capacity Settings

Specify capacity settings for this database.

Number of sites before a warning event is generated
9000

Maximum number of sites that can be created in this database
15000

Figure 23:Configure a content database for failover database server

SQL Snapshots

As also mentioned earlier in this guide, SharePoint 2010 backup and export can take advantage of SQL snapshots. When a Windows PowerShell cmdlet is run that utilizes a SQL snapshot, a snapshot of the database is taken and the remaining commands are executed against the snapshot as opposed to the live production database. This method reduces the load on the production databases. It is important to note that SQL snapshots are only available when running SQL Server Enterprise.

The following sample SQL script can be used to create a snapshot for a SharePoint content database named WSS_Content:

```
CREATE DATABASE WSS_Backup_2010_4 ON
( NAME=WSS_Content, FILENAME = 'c:\WSS_Backup_2010_4.ss')
AS SNAPSHOT OF <WSS_Content>;
GO
```

For more information about SQL snapshots, refer to the following link: [Database Snapshots](http://go.microsoft.com/fwlink/?LinkId=156753) (<http://go.microsoft.com/fwlink/?LinkId=156753>).

Search and Index Backup and Restore

SharePoint 2010 has the ability to back up search and index databases using a point in time approach. This allows SharePoint Administrators the ability to restore the search administration, crawl, and query databases. SharePoint Server can then complete an incremental crawl based on all content that has been modified since the backup was taken. This will eliminate the requirement to complete a full crawl if the search database needs to be restored.

Governance

Governance planning is even more important to consider when deploying SharePoint 2010 due to the increased emphasis and availability of social computing features, which means there are more types of content to govern. In addition, SharePoint 2010 offers new capabilities to manage metadata at the enterprise level, which means that companies may need to consider establishing a new organizational role to plan and monitor metadata attributes across the organization. Even if a well-documented governance plan for your Office SharePoint Server 2007 environment is in place, it will likely need to be adapted to incorporate the SharePoint 2010 features that will be deployed. Here are some of the governance areas that are new to SharePoint 2010 for IT Professionals:

- SharePoint 2010 does a better job of handling lists with large amounts of data. Through a new feature, SharePoint 2010 can automatically restrict user queries of large lists via the Resource Monitoring interface. This is a policy setting and should be considered when defining a plan for overall governance since it can impact overall usability of the system.
- SharePoint 2010 introduces new capabilities for sharing metadata across multiple site collections and even server farms, which requires planning and control in order to leverage effectively. Organizations may need to consider adding a new role (or at least a new responsibility to an existing role) to manage and maintain the dictionary of shared metadata.
- SharePoint 2010 introduces Sandboxed Solutions, which enables the site collection administrator to directly upload customization elements such as web

parts. Within the governance plan, organizations should have a customization policies section that describes how to deal with the numerous ways to create solutions on top of SharePoint.

- A Group Policy object is introduced to disable the installation of SharePoint 2010 and related products. This can help to prevent rogue installation of SharePoint in an organization.
- Administrators can use Active Directory Markers (Service Connection Point, or SCP) to identify SharePoint 2010 servers in an organization. A container called Microsoft SharePoint Products can be created before beginning SharePoint 2010 deployments. When new SharePoint products get deployed, PSConfig adds a new service connection point into container. These SCPs can be tracked by ASDI Edit Tool or using Windows PowerShell scripts.

For more information on Governance in SharePoint 2010, see <http://go.microsoft.com/fwlink/?LinkID=189324>

Multi-tenancy and Hosting

SharePoint 2010 has the capability to isolate and separate data from different Web sites while sharing service application resources across these same sites. This capability is called *multi-tenancy*. Multi-tenancy of services creates a true hosting environment and makes it possible to share service resources across customers (tenants) while partitioning data based on *site subscriptions*. Site subscriptions group tenant data across all site collections owned by the tenant, and provide the ability to separate and group each tenant's data in an otherwise shared environment. Administrators can centrally deploy and manage features and services while giving tenants full control over the usage and experience.

Site Subscriptions

Multi-tenancy relies on site subscriptions and subscription IDs. Site collections for each tenant are grouped together by site subscription based on a common subscription ID. The subscription ID is used to map features and services to tenants and also to partition service data according to tenant requirements. The following key points provide additional perspective about the multi-tenancy capability:

- All site collections for a single site subscription must reside in the same Web application.
- Multiple site subscriptions can be hosted in a single Web application; multiple site subscriptions can share the same database.
- Administrators define which services are available and activated for each tenant. The subscription ID for a tenant is used to map the service to the site collections.
- Multiple tenants can share service data. For example, multiple branches of an organization can share profile data. In this case, the multiple branches share a data partition within the profile service.
- Service data can be partitioned for each tenant, ensuring that customer data is not exposed to other tenants. In this case, data for a single tenant is contained within a separate data partition for the service.

Administration

Administrators can host multiple tenants on the same farm and centrally manage the deployment of services and features. Tenants can manage the configuration of administrator-delegated features and control the functionality of their site collections. Administrative functionality is aligned with common hosting roles, as summarized in the following table:

Role	Description
Hosting company	Manages the farm-level settings and hardware Controls database configurations Installs all new approved features and solutions Can brand the Tenant Administrator pages
Hosted company administrator	Purchases space, features, and bandwidth from hosting company Controls the architecture of customer sites but not the content Reviews usage statistics
Hosted company	Owens site collection

	Installs or removes features and solutions Configures features and services Reviews usage statistics
--	--

Tenants administer their sites by using a site template called Tenant Administration, which is associated with a subscription ID. This site template is hidden and can only be granted to tenants by a farm administrator.

Branding

There have been a number of changes related to branding the user interface added to SharePoint 2010.

Themes

In previous versions of SharePoint Server and Windows® SharePoint Services, themes were a common way to change the look and feel of the site. Administrators could open the Site Settings page and select from one of several themes. Although this was a quick and easy way to make a site look different than the default site, often the themes available out of the box didn't match the corporate brand for a particular organization. In those cases, the options were either to create a custom theme, which required some development and modification of files on the server, or a company would just have to accept what was available.

SharePoint 2010 has changed the way themes work, making it easier to customize and create new themes. Users who have created a custom theme in Microsoft Office PowerPoint might notice that the interface to create themes looks very similar. It is even possible to import PowerPoint and other Office 2007 and 2010 client themes directly into SharePoint 2010. Additionally, themes can now be applied to all subsites from this interface. The new themes functionality in SharePoint 2010 makes it faster and easier than ever before for administrators to create and deploy a customized user interface.

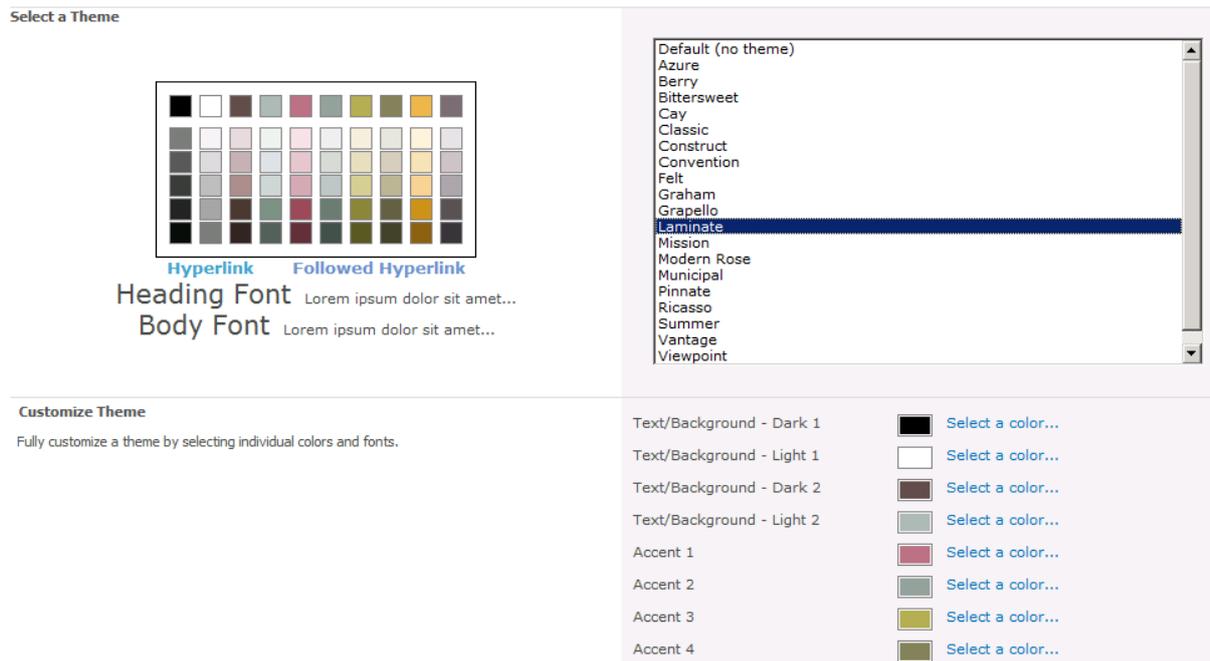


Figure 24: Theme in SharePoint 2010

Master Pages

SharePoint 2010 relies on master pages for implementing a custom look and feel to sites. Through the Web interface or by using SharePoint Designer, a site owner is able to apply branding to the site, independent of other sites. This allows the administration of look and feel to be delegated as appropriate. For branding of application and system pages, this role will fall to the server administrator.

Server administrators will be able to specify whether the system pages in the _Layouts folder are rendered using the site master pages provided by site owners or by default master pages available across the system. In Office SharePoint Server 2007, pages rendered from this directory all used the same master page, application.master. This presented a challenge to organizations that wanted to create a completely custom user interface, due to the fact that application.master is a system file, and there is no option to use a custom master page for this. The only options available were to modify the system file or to style the page using a custom theme. It should be noted that customizing application.master was not recommended because in the event that something unfortunate happened to this master page, none of the system settings for a site could be accessed. Not only does SharePoint 2010 add greater flexibility for how to

apply branding to these system pages, it also provides a fail-safe mechanism. If there is an error in the master page used for pages in the _Layouts directory, SharePoint will reference the default.master file so that system pages can still be accessed. Additionally, it is now possible to use Windows PowerShell to apply a custom master page to system error pages, login pages, confirmation pages, and other non-site-specific pages. By combining all of these options, creating a consistent brand across SharePoint is now easier than ever.

SharePoint Designer 2010

SharePoint Designer 2010 is a powerful tool for customizing SharePoint 2010 sites and adding functionality. With any tool deployed in an organization, applying an effective governance plan is key to ensuring responsible use of the tool. SharePoint 2010 gives administrators added control over how SharePoint Designer 2010 is used in each Web application. Administrators can control the following options:

- **Enable SharePoint Designer:** Controls whether SharePoint Designer can be used at all in the Web application.
- **Enable Detaching Pages from the Site Definition:** Determines whether pages can be detached from the site definition. (SharePoint Designer only allows customization of a page when the page is put into advanced editing mode. This will greatly reduce the number of pages that will be detached from the site definition.)
- **Enable Customizing Master Pages and Layout Pages:** Controls whether site administrators are allowed to customize master pages and layout pages.
- **Enable Managing of the Web Site URL Structure:** Specifies whether site administrators can manage the URL structure of their site.

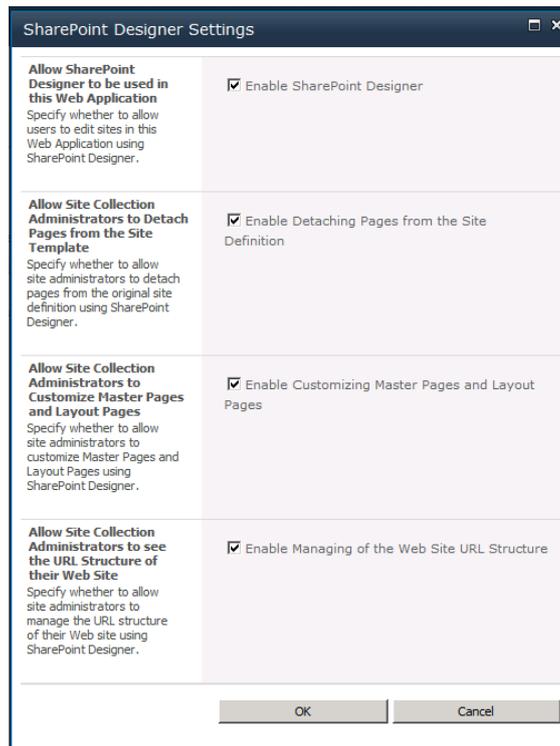


Figure 25:SharePoint Designer Settings

Conclusion

At its core, the principal uses for SharePoint®Products and Technologies (site provisioning, Web Parts, collaboration, document management, Web management, search) haven't changed. What has changed is the amount of out-of-the-box functionality that can be leveraged within an organization that chooses to implement SharePoint 2010 as a business solution.

From a business perspective, SharePoint 2010 solves the need for a centralized solution that can be used in conjunction with existing software and applications. By leveraging Microsoft Business Connectivity Services, SharePoint 2010 can natively connect to existing sources of information. With the multi-tenancy feature, organizations have more control with hosted environments (from the hosting side to the client side).

For an administrator, this release has made drastic improvements toward a solution that is not only manageable, but is also easy to maintain. The integration with Windows PowerShell allows for more administrative control, a more efficient command line tool, and improves productivity. The new Central Administration interface makes it easier to

locate and configure settings for the SharePoint environment. SharePoint 2010 has incorporated more performance controls, logging and reporting options, in addition to a much-desired improvement on the native disaster-recovery options. The revamped service application architecture ensures greater flexibility with regard to the shared services being used within the environment.

Resources

TechNet SharePoint 2010 TechCenter

<http://go.microsoft.com/fwlink/?LinkID=189311>

Getting Started Series for SharePoint 2010 IT Professionals

<http://go.microsoft.com/fwlink/?LinkID=164613>

Advanced Training Videos for SharePoint 2010 (Level 400)

<http://go.microsoft.com/fwlink/?LinkID=164614>

Microsoft Learning Resources for SharePoint 2010

<http://go.microsoft.com/fwlink/?LinkID=189312>

SharePoint 2010 Trial Download (Coming Soon)

<http://go.microsoft.com/fwlink/?LinkID=189313>

SharePoint 2010 Evaluation Virtual Machine Download

<http://go.microsoft.com/fwlink/?LinkID=189314>

SharePoint 2010 IT Professional Virtual Labs

<http://go.microsoft.com/fwlink/?LinkID=189315>

SharePoint 2010 TechNet Forum

<http://go.microsoft.com/fwlink/?LinkID=165422>