



Planning guide for sites and solutions for Microsoft SharePoint Server 2010, Part 2

Microsoft Corporation

Published: January 2011

Author: Microsoft Office System and Servers Team (itspdocs@microsoft.com)

Abstract

This book provides information about planning for site and solutions created by deploying Microsoft SharePoint Server 2010. Subjects include business intelligence, site creation, web content management, and enterprise search. The audiences for this book are business application specialists, line-of-business specialists, information architects, IT generalists, program managers, and infrastructure specialists who are planning a solution based on SharePoint Server 2010. This book is part of a set of four planning guides that provide comprehensive IT planning information for SharePoint Server.

For further information about planning sites and solutions for SharePoint Server, see [Planning guide for sites for Microsoft SharePoint Server 2010, Part 1](http://go.microsoft.com/fwlink/?LinkID=196150) (http://go.microsoft.com/fwlink/?LinkID=196150).

Subjects in Part 1 include site security, governance, and enterprise content management.

For information about planning the architecture of a SharePoint Server 2010 deployment, see [Planning guide for server farms and environments for Microsoft SharePoint Server 2010](http://go.microsoft.com/fwlink/?LinkID=189513) (http://go.microsoft.com/fwlink/?LinkID=189513).

For information about planning for capacity and performance in SharePoint Server 2010, see [Capacity planning for Microsoft SharePoint Server 2010](http://go.microsoft.com/fwlink/?LinkID=208221) (http://go.microsoft.com/fwlink/?LinkID=208221).

The content in this book is a copy of selected content in the [SharePoint Server 2010 technical library](http://go.microsoft.com/fwlink/?LinkId=181463) (http://go.microsoft.com/fwlink/?LinkId=181463) as of the publication date. For the most current content, see the technical library on the Web.

Microsoft®

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.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

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.

© 2011 Microsoft Corporation. All rights reserved.

Microsoft, Access, Active Directory, Backstage, Excel, Groove, Hotmail, InfoPath, Internet Explorer, Outlook, PerformancePoint, PowerPoint, SharePoint, Silverlight, Windows, Windows Live, Windows Mobile, Windows PowerShell, Windows Server, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Contents

Getting help.....	20
Technical diagrams (SharePoint Server 2010).....	21
Models	21
Tips for printing posters.....	33
Plan for sites and solutions (SharePoint Server 2010).....	34
Plan Web content management (SharePoint Server 2010)	37
Publishing features overview	39
About publishing sites	39
About publishing features.....	40
SharePoint Server Publishing Infrastructure features	40
Site templates	40
Groups and permission levels.....	40
Site settings.....	41
Navigation	42
Theme changes	42
Master pages and page layouts.....	42
Images and style sheets	43
Document libraries and lists.....	43
Content types.....	43
Columns	43
Web Parts	44
Page editing menu	44
Timer jobs	45
SharePoint Server Publishing features	46
Site settings.....	46
Regional settings	46
Document libraries and lists.....	46
Page editing menu	47
Other changes	48
Other publishing features	48
Plan Web pages	50
Web pages overview	50
Master pages	51
Page layouts	52
Content pages.....	53

Plan master pages	53
Plan page layouts	54
Plan content pages.....	55
Using page layouts to restrict authoring	57
Setting restrictions on field controls	58
Allowing or restricting Web Part zones	58
Web page planning worksheet.....	59
Plan Web page authoring (SharePoint Server 2010)	60
About planning Web page authoring	60
Plan ribbon authoring experience	61
Plan managed metadata	62
Plan reusable content.....	63
Plan dictionary customizations.....	64
Plan additional resources	64
Web page authoring planning worksheet.....	65
Plan content approval and scheduling.....	66
About planning content approval and content scheduling	66
Plan content approval.....	66
Plan content scheduling	67
Using content deployment with content approval and scheduling	67
Plan for caching and performance (SharePoint Server 2010).....	69
Disk-based BLOB caching	69
BLOB cache overview.....	70
Decide whether to use the BLOB cache	70
Store the BLOB cache	71
Enable the BLOB cache.....	71
Specify the size of the BLOB cache	71
Bit Rate Throttling.....	71
Bit Rate Throttling overview	72
Decide to use Bit Rate Throttling	72
Enable Bit Rate Throttling	72
Maximum upload file size	72
Maximum upload file size overview	72
Decide maximum upload file size	73
Configure the maximum upload file size	73
Plan for large Pages libraries (SharePoint Server 2010)	74
About large Pages libraries	74
Determine whether to use a large Pages library	75
Decide how to manage pages.....	75

Plan for navigation.....	76
Planning the Global Navigation and the Current Navigation menus	76
Planning other Web parts for navigation.....	76
Content deployment overview (SharePoint Server 2010)	78
What is content deployment?	78
About deployment paths and jobs	80
Content deployment paths	80
Content deployment jobs	80
About content deployment security	82
How content deployment works	83
Important considerations in content deployment	87
Plan content deployment (SharePoint Server 2010)	89
About planning content deployment.....	89
Determine whether to use content deployment.....	89
Determine how many server farms you need	90
Plan the export and import servers	91
Plan content deployment paths	91
Plan job scheduling	91
Plan for large jobs	92
Content deployment planning worksheet	93
Design content deployment topology	94
Elements of content deployment topologies	94
Typical content deployment topologies	94
Two-farm topology	95
Three-stage topology	96
Single-farm topology	96
Variations overview.....	98
Use and benefits of variations	98
Scenarios for using variations	99
Elements of variations	100
Understanding variations.....	100
Variation labels	101
Variation settings	102
Variations timer jobs	103
Understanding source variation and target variation site creation	103
Understanding site and page creation	104
Site creation	104
Page creation	104
Limitations of variations.....	106

Plan variations	107
About planning variations	107
Important items to consider when planning to use variations	108
Content approval.....	108
Site navigation	109
Content deployment.....	109
Web Parts	110
Multilingual sites.....	110
Determine the types of variations needed.....	110
Select the variation root site	111
Specify the source variation site.....	111
Plan target variation sites	111
Plan custom master pages, layout pages or style sheets	112
Plan custom content types.....	112
Decide how sites and pages will be created on target variation sites	113
Plan variations timer job scheduling.....	113
Variations planning worksheet	114
Plan information architecture for Web content management	115
General planning recommendations	115
Plan the structure of your site.....	116
Plan for social computing and collaboration.....	116
Plan for managed metadata	117
Plan for business intelligence and business data	117
Plan for search	118
Plan managed metadata (SharePoint Server 2010)	119
Managed metadata overview (SharePoint Server 2010)	120
Understanding managed metadata	120
Terms and term sets	120
Managed terms, enterprise keywords, and the term store	121
Working with managed metadata.....	121
Creating terms	121
Using terms.....	123
Entering terms.....	124
Entering enterprise keywords	124
Benefits of using managed metadata.....	125
More consistent use of terminology	125
Better search results	126
Dynamic	126
Managed metadata service application overview (SharePoint Server 2010).....	127

Managed metadata services	127
Managed metadata connections	127
Permissions for accessing a managed metadata service	128
Example scenario	130
Design	130
Permissions	132
Connection parameters	132
Managed metadata roles (SharePoint Server 2010)	135
Roles and capabilities	135
Plan terms and term sets (SharePoint Server 2010)	137
Plan: now or later	137
About planning managed metadata	138
Identify term sets	138
Identify term set owners	140
Determine term set groups	140
Define term sets	140
Identify the terms	141
Organize the terms	141
Identify who can add terms	142
Managed metadata planning worksheets	142
Plan to import managed metadata (SharePoint Server 2010)	143
About planning to import managed metadata	143
Locating existing data	144
Organizing the data into managed metadata	144
Cleaning up the data	145
Formatting the data to be imported	145
Importing the managed metadata	145
Merging terms	146
Plan to share terminology and content types (SharePoint Server 2010)	147
About planning managed metadata services and connections	147
Identify managed metadata services	148
Identify managed metadata connections	148
Determine service account permissions	149
Managed metadata services planning worksheet	151
Multilingual term sets (SharePoint Server 2010)	152
Defining terms	152
Using terms (tagging)	153
How terms are displayed	154
Recommendations	155

Business intelligence planning.....	156
Business intelligence basics.....	157
Choosing a business intelligence tool in SharePoint Server	158
Services in SharePoint Server for business intelligence.....	159
Excel 2010	159
Excel Services	159
Visio Services.....	159
PerformancePoint Services	159
Matching a tool with a broad scenario	160
SQL Server Reporting Services in SharePoint Server.....	161
PowerPivot for Excel 2010	161
Architecture for business intelligence in SharePoint Server 2010.....	162
Secure Store for Business Intelligence service applications	165
Secure Store Service	165
Data connection files	167
The Unattended Service Account	167
Data access from client and server	167
Excel Services and Visio Services	167
Excel Services	168
Visio Services.....	168
PerformancePoint Services.....	169
Summary of differences	169
Overview of documentation for SQL Server Reporting Services reports in SharePoint	171
Overview of Reporting Services in SharePoint integrated mode	171
Planning and architecture for Reporting Services in SharePoint integrated mode	172
Configuration for Reporting Services in SharePoint integrated mode	172
Overview of PowerPivot documentation (SharePoint Server 2010).....	173
Overview of PowerPivot for Excel and SharePoint	173
Planning and architecture for PowerPivot in SharePoint and Excel Services	173
Deployment for PowerPivot in Excel Services and SharePoint 2010 Products.....	174
Data warehousing, OLAP, and Analysis Services for SharePoint 2010	175
Overview of data warehousing, OLAP, and PowerPivot and relation to SharePoint 2010	175
Excel Services overview (SharePoint Server 2010).....	177
What is Excel Services?	177
Overview of Excel Services architecture	179
Excel Services Components	179

Performance and Scalability	180
Plan Excel Services data sources and external connections	182
Connections and Excel workbooks	182
Embedded and linked connections	182
Data providers.....	183
Authentication to external data	184
Integrated Windows authentication	185
Secure Store Service.....	185
None	186
Data connection libraries and managed connections	186
Reusing connections	186
Managing connections.....	186
Securing connections	187
Excel Services security and external data	187
Trusted file locations.....	187
Trusted data connection libraries	188
Trusted data providers.....	188
Unattended account	188
Plan Excel Services data providers (SharePoint Server 2010)	190
Data providers	190
Trusted data providers	190
Plan Excel Services authentication (SharePoint Server 2010)	192
About Excel Services security	192
Plan user authentication.....	193
Plan communication among servers	193
Plan external data authentication.....	194
Integrated Windows authentication.....	194
Secure Store Service authentication	195
None.....	195
Unattended service account	196
Security settings.....	196
File access method	197
Connection encryption	197
Trusted file locations	197
Trusted data providers	200
Trusted data connection libraries.....	200
View Only permissions.....	200
External data connections.....	201
.odc files	201
Managing .odc files	202

User-defined function assemblies.....	202
Excel Services capacity planning	203
High Performance Computing Services for Excel 2010	203
Plan for PerformancePoint Services (SharePoint Server 2010).....	204
PerformancePoint Services overview (SharePoint Server 2010)	206
PerformancePoint Services.....	206
New features and enhancements	206
Retired features	207
Overview of PerformancePoint Services architecture	208
PerformancePoint Services topology	208
PerformancePoint Services as a service application.....	209
Estimate performance and capacity requirements for PerformancePoint Services	210
Test farm characteristics	210
Test scenarios and processes.....	211
Hardware setting and topology.....	213
Test results	214
2M and 3M topologies	216
4M+ results for Unattended Service Account authentication	219
4M+ Results for per-user authentication	220
Recommendations	221
Analysis Services	223
Common bottlenecks and their causes	223
Performance monitoring	225
Client hardware and software requirements for PerformancePoint Dashboard Designer	227
Hardware requirements.....	227
Software requirements	227
Plan for importing PerformancePoint Server 2007 dashboard content to SharePoint Server 2010 (SharePoint Server 2010).....	228
Reports types not supported in PerformancePoint Services	228
Planning permissions and roles	228
Roles and permissions	229
Running the wizard.....	230
Post-migration tasks for the PerformancePoint dashboard author.....	232
Post-migration tasks for the dashboard author	232
Plan for PerformancePoint Services security (SharePoint Server 2010)	234
Authentication.....	234

Trusted Locations	234
Trusted data content libraries	235
Trusted Lists for Dashboard Content	236
Data source security	236
The Secure Store Service and Unattended Service Accounts	236
Claims-based authentication	237
Authorization and permissions in PerformancePoint Services (SharePoint Server 2010)	238
Planning permissions and roles	238
Roles and permissions	238
Planning for PerformancePoint data sources (PerformancePoint Services)	241
Tabular Data Sources	241
SharePoint Lists	241
Excel Services	242
SQL Server tables	242
Excel workbooks	242
Multidimensional Data Sources	242
Analysis Services	242
PowerPivot for Excel	243
Best practices for SQL Server 2005 and 2008 OLAP cube design and MDX querying	244
SQL Server 2008 enhancements for business intelligence	244
Best practices for Analysis Services	244
Overview of PerformancePoint Services components	246
Dashboard Designer	246
Web Parts	246
PerformancePoint Site collections	247
PerformancePoint Sites	247
Plan to customize PerformancePoint Services	248
Development scenarios for PerformancePoint Services	248
PerformancePoint Services and PowerPivot for Excel (white paper)	249
Plan for Visio Services (SharePoint Server 2010)	250
Visio Services overview (SharePoint Server 2010)	251
Use and benefits of Visio Services	251
Data sources supported by Visio Services	251
Published Visio drawings	252
Plan Visio Services deployment	253
Visio Services performance	253

Visio Graphics Service applications	254
Using a pilot deployment	254
Monitoring	255
Backup and recovery of data	255
Visio Professional 2010 and Visio Premium 2010 deployment	255
Plan Visio Services security (SharePoint Server 2010)	256
Web drawings that are not connected to data	256
Visio Web drawings that are connected to data	256
Visio Web drawings that are connected to SharePoint lists	257
Visio Web drawings that are connected to Excel Services	257
Visio Web drawings that are connected to SQL Server databases	257
Data authentication for Visio Services	259
Connecting to data hosted on SharePoint Server	260
Connecting to Excel workbooks	260
Connecting to SharePoint lists	260
Connecting to external data	261
Data connections	261
Windows authentication	263
Kerberos delegation	266
Secure Store	266
Unattended Service Account	267
SQL Server Authentication	268
Authentication against OLEDB/ODBC data sources	268
Data refresh	268
Visio Services resources	271
Documentation, references, and white papers	271
Blog posts	271
Video demonstrations	271
Plan for Business Intelligence Indexing Connector (SharePoint Server 2010)	272
Introduction to Business Intelligence Indexing Connector	273
Features of Business Intelligence Indexing Connector	273
Overview of Business Intelligence Indexing Connector search tab interface	275
Reports tab	275
Results description	276
Document thumbnail	276
Preview, duplicates, View in Browser	276
Refinement categories	277

Determine software requirements for Business Intelligence Indexing Connector	278
Software requirements for Business Intelligence Indexing Connector: back end	278
Installing software prerequisites	278
Software requirements for Business Intelligence Indexing Connector: front end	279
Install Microsoft Business Intelligence Indexing Connector — front end and back end	279
Overview of Business Intelligence Indexing Connector architecture.....	280
Setup for FAST Search Server 2010 for SharePoint (back end)	280
Setup for SharePoint Server 2010 (front end).....	281
Logical architecture	281
Understanding planning solutions and scenarios (white paper)	282
Business data and processes planning (SharePoint Server 2010)	283
Plan for Business Connectivity Services (SharePoint Server 2010)	284
Business Connectivity Services overview (SharePoint Server 2010)	285
Typical solutions based on Business Connectivity Services.....	285
Business Connectivity Services architecture	286
Business Connectivity Services security overview (SharePoint Server 2010)	290
About this article	290
Business Connectivity Services security architecture	290
Accessing external data from a Web browser	290
Accessing external data from an Office client application	292
Business Connectivity Services authentication overview	294
Configuring Business Connectivity Services for credentials authentication	294
Configuring Business Connectivity Services for claims-based authentication	300
Business Connectivity Service permissions overview	302
What can permissions be set on?	302
Special permissions on the Business Data Connectivity service	307
Common tasks and their related permissions.....	307
Securing Business Connectivity Services	308
Service account	308
Server to server communication	309
Applications that use FileBackedMetadataCatalog	309
Business Data Connectivity service administration overview (SharePoint Server 2010).....	310
The Business Data Connectivity service.....	310
What can be administered in the Business Data Connectivity service?	311
Plan Business Connectivity Services client integration (SharePoint Server 2010)	313
Prerequisites.....	313
Installing deployment packages	314

ClickOnce applications and trust-prompt behavior	314
Secure Store Service group mappings	316
Sign in as Different User	316
Security considerations	316
Secure communications.....	316
External list permissions	317
Outlook Web Access Web Parts.....	317
Client throttle limits.....	317
Diagnostic logging in Business Connectivity Services overview (SharePoint Server 2010)	321
Diagnostic logging in Business Connectivity Services	321
About Activity IDs	323
Diagnostic logging on servers	324
Diagnostic logging on Office 2010 clients	325
Example: using diagnostic logging	326
Plan to upgrade to Business Connectivity Services (SharePoint Server 2010)	328
The Business Data Catalog, Application Registry, and Business Data Connectivity service	328
How Business Connectivity Services upgrade works	329
Upgrading by using database attach.....	330
Solution-specific upgrade considerations.....	331
Models.....	331
Web Parts	332
Search.....	332
Single sign-on	333
Maintaining service databases on separate servers.....	333
Maintaining parent and child farm relationships	333
Plan InfoPath Forms Services (SharePoint Server 2010)	335
About forms in SharePoint Server 2010	336
InfoPath forms overview	336
Role of forms in SharePoint solutions	337
Types of InfoPath forms	337
InfoPath components	337
Web browser vs. Filler-only forms	338
Web browser forms.....	338
SharePoint list forms.....	338
External list forms	339
Form library forms.....	339
Workflow forms	339
Deploying forms	339
Publishing browser forms without code	340

Publishing browser forms with code	340
Form templates as sandboxed solutions	340
Administrator-approved form templates	340
Filling out forms	341
Browser vs. Filler forms	341
Offline form-filling	341
InfoPath Form Web Part	341
Plan a forms-driven application	342
Structure of a form-driven application	342
About planning a common form-driven application	343
Identifying the key piece of information	343
Using a list or a form library	344
Workflow	344
Additional data sources	345
Portals	345
Summary	346
Plan for user form templates (SharePoint Server 2010)	348
About user form templates	348
Browser-enabled user form templates	348
Plan external data access	349
Cross-domain access	349
InfoPath Forms Services Web service proxy	349
Authentication information in data connection files	350
Data connection library	350
Plan to upgrade form templates during an upgrade to SharePoint Server 2010	352
About upgrading forms during an upgrade to SharePoint Server 2010	352
Upgrade form templates during a database attach upgrade to SharePoint Server 2010	353
Export and import administrator-approved form template files between configuration databases	354
Update form template links to the server	354
Upgrade form templates during an in-place upgrade to SharePoint Server 2010	355
InfoPath 2010 Enhanced Integration with SharePoint Server 2010 and Its Implications When Designing Forms for Applications (white paper)	356
Plan workflows (SharePoint Server 2010)	357
Workflows overview (SharePoint Server 2010)	358
Workflow overview	358
Benefits of using workflows	359
Automating business processes	359
Workflows improve collaboration	360

Predefined workflows	360
Sample workflow scenario.....	362
Workflow types: Declarative and compiled	364
Workflow templates	364
Workflow associations.....	365
Office client interoperability	365
Choose a workflow authoring tool (SharePoint Server 2010)	367
Authoring workflows with Visual Studio 2010 and WF Workflow Designer	368
Authoring workflows with Microsoft SharePoint Designer 2010.....	370
Authoring tool comparison.....	373
Plan for approval and review processes in workflows (SharePoint Server 2010)	375
Workflow approval overview.....	375
How the Approval workflow works	375
Example — Manage the document approval process by using a workflow	376
Hybrid review model.....	377
Plan for workflow security and user management (SharePoint Server 2010)	378
List manager, administrator, and developer roles and responsibilities	378
Workflow developers.....	378
Site administrators	378
List administrators (anyone with Manage List or Web Designer permissions).....	379
Running workflows as an administrator	379
Workflow configuration settings.....	379
Required permissions to start a workflow	379
Central Administration settings	380
Enable user-defined workflows.....	380
Task notification for users without site access	380
Information disclosure in task and workflow history lists.....	381
Spoofing and tampering attacks in the task and workflow history lists	382
Security issues in the workflow history list.....	382
User-Impersonation Step type for declarative workflows	383
Approval Workflow: A Scenario (SharePoint Server 2010)	385
Authoring a workflow	385
Associating a workflow	385
Associating a workflow with a site	386
Starting a workflow	386
Interacting with a workflow	387
Summarizing the process.....	388
Approval workflow scenario.....	389
Access Services planning.....	390

Introduction to Access Services (SharePoint Server 2010)	391
Who should use Access Services?	391
Features of Access Services	391
Improving the reach and manageability of Access 2010 database applications with Access Services (white paper)	392
Plan site creation and maintenance (SharePoint Server 2010)	393
Plan process for creating sites (SharePoint Server 2010)	394
Determine who can create sites and a method for site creation	394
Plan for Self-Service Site Management	395
Plan for custom site creation processes	396
Worksheet	396
Plan site maintenance and management (SharePoint Server 2010)	397
Plan for site maintenance	397
Plan for managing site collections	398
Plan site collection quotas	398
Plan site use confirmation and deletion	399
Worksheet	399
Plan quota management (SharePoint Server 2010)	400
About planning quota management	400
Determine quota template settings	401
Determine recycle bin settings	401
Delete unused Web sites	402
Reporting and usage analysis overview	403
Overview	403
Reporting	403
Traffic reports	404
Search reports	407
Inventory reports	410
Web Analytics workflow	413
Web Analytics Web Part	413
Plan e-mail integration (SharePoint Server 2010)	414
Plan incoming e-mail (SharePoint Server 2010)	415
About incoming e-mail	415
Key decisions for planning incoming e-mail	415
Using a basic scenario	415
Using an advanced scenario	416
SharePoint Directory Management service	416

Incoming e-mail server display address	418
Safe e-mail server	418
E-mail drop folder	419
Configuration options and settings modes	419
Plan incoming e-mail worksheet.....	420
Plan outgoing e-mail (SharePoint Server 2010)	421
About outgoing e-mail	421
Key planning phases of outgoing e-mail	422
Outbound SMTP server	422
From and Reply-to addresses.....	422
Character set.....	423
Enterprise search planning (SharePoint Server 2010)	424
Gather information about the current search environment (SharePoint Server 2010)	425
Organization information	425
Topology information	425
Current search settings	425
Performance and usage reports.....	427
Determine the enterprise search team and stakeholders (SharePoint Server 2010).....	428
Plan for crawling and federation (SharePoint Server 2010)	429
Plan content sources	429
Plan to crawl different kinds of content	430
Plan content sources for business data	431
Crawl content on different schedules.....	431
Considerations for planning crawl schedules	432
Reasons to do a full crawl	432
Limit or increase the quantity of content that is crawled	434
Other considerations when planning content sources	435
Plan file-type inclusions and IFilters	436
Plan for authentication.....	436
Plan connectors.....	437
Plan to manage the impact of crawling	437
Plan crawl rules	438
Plan search settings that are managed at the farm level	439
Plan for federation	439
Plan authentication types for federation	441
Plan the topology for enterprise search (SharePoint Server 2010).....	443
Planning worksheets for SharePoint Server 2010.....	444

Planning worksheets by task..... 444

Planning worksheets by title..... 447

Getting help

Every effort has been made to ensure the accuracy of this book. This content is also available online in the Office System TechNet Library, so if you run into problems you can check for updates at:

<http://technet.microsoft.com/office>

If you do not find your answer in our online content, you can send an e-mail message to the Microsoft Office System and Servers content team at:

itspdocs@microsoft.com

If your question is about Microsoft Office products, and not about the content of this book, please search the Microsoft Help and Support Center or the Microsoft Knowledge Base at:

<http://support.microsoft.com>

Technical diagrams (SharePoint Server 2010)


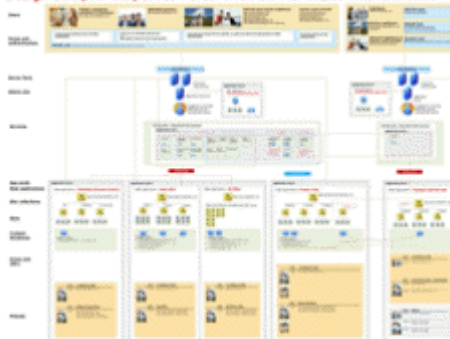
Many of these resources are visual representations of recommended solutions. They include poster-sized documents available in formats including Microsoft Office Visio 2007 or Microsoft Visio 2010 files (.vsd), PDF files, and XPS files. You might need extra software to view these files. See the following table for information about opening these files.

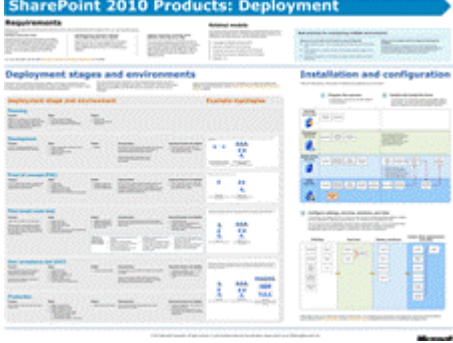


File type	Software
.vsd	Office Visio 2007, Microsoft Visio 2010, or the free Visio viewer (http://go.microsoft.com/fwlink/?LinkId=118761) If you use the Visio viewer, right-click the VSD link, click Save Target As , save the file to your computer, and then open the file from your computer.
.pdf	Any PDF viewer, such as Adobe Reader (http://go.microsoft.com/fwlink/?LinkId=134751)
.xps	Windows 7, Windows Vista, Windows XP with .NET Framework 3.0, or XPS Essentials Pack (http://go.microsoft.com/fwlink/?LinkId=134750)

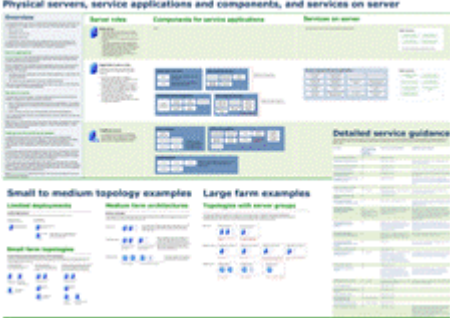

Models



Models are 34-by-44-inch posters that detail a specific technical area. These models are intended to be used with corresponding articles on TechNet. These models are created by using Office Visio 2007. You can modify the Visio files to illustrate how you plan to incorporate Microsoft SharePoint 2010 Products in your own environment.

Title	Description
Design Sample: Corporate Portal with Classic Authentication	Illustrate a typical corporate deployment, with the most common types of sites represented. The two samples differ only in the mode of authentication that is implemented. Use these design samples with the



Title	Description
<p>Design Sample: Corporate Portal with Classic Authentication</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkId=196969) PDF (http://go.microsoft.com/fwlink/?LinkId=196970) XPS (http://go.microsoft.com/fwlink/?LinkId=196971)</p> <p>Design Sample: Corporate Portal with Claims-based Authentication</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkId=196972) PDF (http://go.microsoft.com/fwlink/?LinkId=196973) XPS (http://go.microsoft.com/fwlink/?LinkId=196974)</p>	<p>following article: Design sample: Corporate deployment (SharePoint Server 2010)</p>
<p>SharePoint 2010 Products Deployment</p>	<p>Presents such deployment-related information as the different deployment stages and environments, plus a flowchart that illustrates the steps for installing and configuring SharePoint 2010 Products.</p>

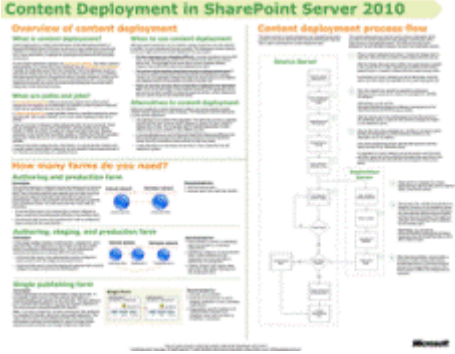

Title	Description
<p>SharePoint 2010 Products: Deployment</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkId=183024) PDF (http://go.microsoft.com/fwlink/?LinkId=183025) XPS (http://go.microsoft.com/fwlink/?LinkId=183026)</p>	
<p>Services in SharePoint 2010 Products</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkID=167090) PDF (http://go.microsoft.com/fwlink/?LinkID=167092) XPS (http://go.microsoft.com/fwlink/?LinkID=167091)</p>	<p>Describes and illustrates the services architecture, including common ways to deploy services in your overall solution design.</p> <p>Use this diagram with the following articles:</p> <ul style="list-style-type: none"> • Services architecture planning (SharePoint Foundation 2010) • Services architecture planning (SharePoint Server 2010)
<p>Cross-farm Services in SharePoint 2010 Products</p> 	<p>Illustrates how to deploy services across farms to provide centralized administration of services.</p> <p>Use this diagram with the following articles:</p> <ul style="list-style-type: none"> • Services architecture planning (SharePoint Foundation 2010) • Services architecture planning (SharePoint Server 2010)



Title	Description
<p>Visio (http://go.microsoft.com/fwlink/?LinkID=167093)</p> <p>PDF (http://go.microsoft.com/fwlink/?LinkID=167095)</p> <p>XPS (http://go.microsoft.com/fwlink/?LinkID=167094)</p>	
<p>Topologies for SharePoint Server 2010</p> <p>Topologies for SharePoint Server 2010 Physical servers, service applications and components, and services on server</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkID=167087)</p> <p>PDF (http://go.microsoft.com/fwlink/?LinkID=167089)</p> <p>XPS (http://go.microsoft.com/fwlink/?LinkID=167088)</p>	<p>Describes common ways to build and scale farm topologies, including planning which servers to start services on.</p>
<p>Extranet Topologies for SharePoint 2010 Products</p> <p>Extranet Topologies for SharePoint 2010 Products</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkId=187987)</p> <p>PDF (http://go.microsoft.com/fwlink/?LinkId=187988)</p> <p>XPS (http://go.microsoft.com/fwlink/?LinkId=187986)</p>	<p>Illustrates the specific extranet topologies that have been tested with SharePoint 2010 Products. Provides a comparison of ISA Server, Forefront TMG, Forefront UAG when used as a firewall or gateway product with SharePoint 2010 Products.</p>
<p>Hosting Environments in SharePoint 2010 Products</p>	<p>Summarizes the support for hosting environments and illustrates common hosting architectures.</p> <p>For more information on designing and</p>



Title	Description
<p>Hosting Environments in SharePoint 2010 Products</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkID=167084) PDF (http://go.microsoft.com/fwlink/?LinkID=167086) XPS (http://go.microsoft.com/fwlink/?LinkID=167085)</p>	<p>deploying hosting environments, see the following: White paper: SharePoint 2010 for hosters (SharePoint Server 2010).</p>
<p>Search Technologies for SharePoint 2010 Products</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkID=167731) PDF (http://go.microsoft.com/fwlink/?LinkID=167733) XPS (http://go.microsoft.com/fwlink/?LinkID=167732)</p>	<p>Compares and contrasts the search technologies that work with SharePoint Products 2010:</p> <ul style="list-style-type: none"> • SharePoint Foundation 2010 • Search Server 2010 Express • Search Server 2010 • SharePoint Server 2010 • FAST Search Server 2010 for SharePoint
<p>Search Environment Planning for Microsoft SharePoint Server 2010</p>	<p>Walks through primary architecture design decisions for search environments.</p>


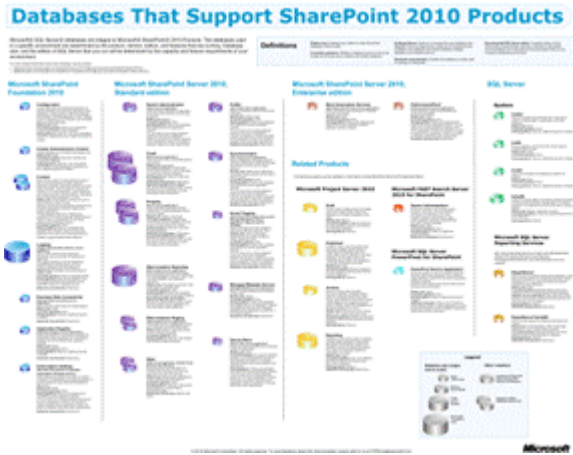
Title	Description
<p>Search Environment Planning for Microsoft SharePoint Server 2010</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkID=167734) PDF (http://go.microsoft.com/fwlink/?LinkID=167736) XPS (http://go.microsoft.com/fwlink/?LinkID=167735)</p>	
<p>Search Architectures for Microsoft SharePoint Server 2010</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkID=167737) PDF (http://go.microsoft.com/fwlink/?LinkID=167739) XPS (http://go.microsoft.com/fwlink/?LinkID=167738)</p>	<p>Details the physical and logical architecture components that make up a search system and illustrates common search architectures.</p>
<p>Design Search Architectures for Microsoft SharePoint Server 2010</p>	<p>Walks through the initial design steps to determine a basic design for a SharePoint Server 2010 search architecture.</p>


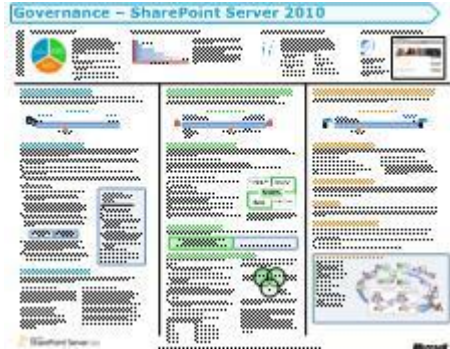
Title	Description
<p>Design Search Architectures for Microsoft SharePoint Server 2010</p>  <p>LinkID=167740) LinkID=167742) LinkID=167741)</p>	
<p>Microsoft Business Connectivity Services Model</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkId=165565) PDF (http://go.microsoft.com/fwlink/?LinkId=165566) XPS (http://go.microsoft.com/fwlink/?LinkId=165571)</p>	<p>Microsoft Business Connectivity Services are a set of services and features in Microsoft SharePoint Server 2010 and Microsoft SharePoint Foundation 2010 that support integrating data from external systems into solutions based on Microsoft SharePoint Server and Microsoft SharePoint Foundation. This model poster describes the architecture of Microsoft Business Connectivity Services in SharePoint Server 2010 and provides information about how to create solutions that are based on the service.</p> <p>Use this model with the following article: Business Connectivity Services overview (SharePoint Server 2010)</p>
<p>Content Deployment in SharePoint Server 2010</p>	<p>Describes the content deployment feature in SharePoint Server 2010. It includes information about the following:</p> <ul style="list-style-type: none"> • Overview of content deployment • Description of content deployment paths and jobs

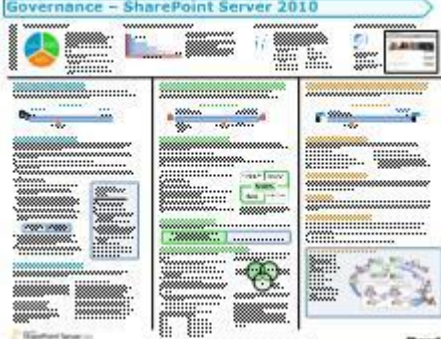
Title	Description
<p>Content Deployment in SharePoint Server 2010</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkID=179391&clcid=0x409)</p> <p>PDF (http://go.microsoft.com/fwlink/?LinkID=179523&clcid=0x409)</p> <p>XPS (http://go.microsoft.com/fwlink/?LinkID=179524&clcid=0x409)</p>	<ul style="list-style-type: none"> • When to use content deployment • Alternatives to content deployment • Illustrates common content deployment farm topologies • Illustrates and explains the overall content deployment process
<p>Microsoft SharePoint Server 2010 Upgrade Planning</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkId=167098)</p> <p>PDF (http://go.microsoft.com/fwlink/?LinkId=167099)</p> <p>XPS (http://go.microsoft.com/fwlink/?LinkId=167100)</p>	<p>Covers planning for an upgrade from Microsoft Office SharePoint Server 2007 to SharePoint Server 2010. It includes information about the following:</p> <ul style="list-style-type: none"> • Upgrade requirements: Hardware, operating system, and database • Upgrade process: specific steps to follow before, during, and after the upgrade <p>Use this model with the following article: Upgrading to SharePoint Server 2010</p>
<p>Microsoft SharePoint Server 2010 Upgrade Approaches</p>	<p>Helps you understand the in-place, database attach, and hybrid approaches to upgrading from Office SharePoint Server 2007 to SharePoint Server 2010.</p> <ul style="list-style-type: none"> • See the farm topologies before, during, and after upgrade

Title	Description
<p>Microsoft SharePoint 2010 Products Upgrade Approaches</p>  <p>Basic upgrade approach "building blocks"</p> <p>Hybrid approaches: combine building blocks</p> <p>Visio (http://go.microsoft.com/fwlink/?LinkId=167101) PDF (http://go.microsoft.com/fwlink/?LinkId=167102) XPS (http://go.microsoft.com/fwlink/?LinkId=167103)</p>	<ul style="list-style-type: none"> Compare the advantages of each type of upgrade approach <p>Use this model with the following articles:</p> <ul style="list-style-type: none"> Determine upgrade approach (SharePoint Server 2010) Upgrade process overview (SharePoint Server 2010)
<p>Microsoft SharePoint Server 2010 — Test Your Upgrade Process</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkId=167104) PDF (http://go.microsoft.com/fwlink/?LinkId=167105) XPS (http://go.microsoft.com/fwlink/?LinkId=167106)</p>	<p>Explains the methodology for testing the upgrade process before upgrading from Office SharePoint Server 2007 to SharePoint Server 2010.</p> <ul style="list-style-type: none"> Understand the goals for testing your upgrade process: customizations, hardware, timing, planning See specific steps to follow for testing your upgrade process <p>Use this model with the following article:</p> <p>Use a trial upgrade to find potential issues (SharePoint Server 2010)</p>
<p>Microsoft SharePoint Server 2010 — Services Upgrade</p>	<p>Covers upgrading services from Office SharePoint Server 2007 to SharePoint Server 2010.</p> <ul style="list-style-type: none"> Considerations for specific services: Personalization, Search, InfoPath Forms, Excel, Business Data

Title	Description
<p>Microsoft SharePoint Server 2010 Services Upgrade</p>  <p>Before you begin Review updated services infrastructure in SharePoint Server 2010. Review after upgrade models.</p> <p>Considerations for specific services Upgrade considerations for specific services.</p> <p>In-place upgrade with services Before upgrade, During upgrade (single SPS), After upgrade.</p> <p>Office SharePoint Server Search During upgrade (multiple SPS).</p> <p>WebPart Farm Services</p> <p>Excel Services</p> <p>Business Data Catalog</p> <p>Single Sign-On</p> <p>Database attach upgrade with services Before upgrade, During upgrade, After upgrade.</p> <p>SharePoint model 1 of 2</p> <p>Visio (http://go.microsoft.com/fwlink/?LinkId=167107) PDF (http://go.microsoft.com/fwlink/?LinkId=167108) XPS (http://go.microsoft.com/fwlink/?LinkId=167109)</p>	<p>Catalog, Single Sign-on</p> <ul style="list-style-type: none"> • In-place upgrade with services • Database attach upgrade with services
<p>Microsoft SharePoint Server 2010 — Upgrading Parent and Child Farms</p>  <p>SharePoint Server 2010 - Upgrading Parent and Child Farms</p> <p>Choose an upgrade approach Recommendation: Microsoft recommends upgrading a majority of the parent farm. Considerations and additional steps.</p> <p>Additional resources</p> <p>Process for upgrading parent and child farms Plan to upgrade for using a majority parent farm. Plan to upgrade for using a temporary parent farm.</p> <p>SharePoint model 2</p> <p>Visio (http://go.microsoft.com/fwlink/?LinkId=190984) PDF (http://go.microsoft.com/fwlink/?LinkId=190985) XPS (http://go.microsoft.com/fwlink/?LinkId=190986)</p>	<p>Covers the process for and considerations to keep in mind when you upgrade farms that share services (parent and child farms).</p>
<p>Getting started with business intelligence in SharePoint Server 2010</p>	<p>Covers an overview of business intelligence in SharePoint Server and provides you with the following</p>

Title	Description
<p>Business Intelligence in SharePoint Server 2010</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkId=167409) PDF (http://go.microsoft.com/fwlink/?LinkId=167170) XPS (http://go.microsoft.com/fwlink/?LinkId=167171)</p>	<p>information.</p> <ul style="list-style-type: none"> • An overview of each business intelligence service and when you might use the service. • Architecture for application of the business intelligence services and how they work together in a topology. • A list of possible data sources for each business intelligence service.
<p>Databases That Support SharePoint 2010 Products</p>  <p>Visio (http://go.microsoft.com/fwlink/?LinkId=187970) PDF (http://go.microsoft.com/fwlink/?LinkId=187969) XPS (http://go.microsoft.com/fwlink/?LinkId=187971)</p>	<p>Describes the Microsoft SQL Server databases on which SharePoint Server 2010 runs.</p>
<p>SharePoint 2010 Products: Virtualization Process</p>	<p>Provides guidance related to virtualization and the various stages of deployment, as well as requirements and examples.</p> <p>Use this diagram with the articles in the following chapters:</p> <ul style="list-style-type: none"> • Virtualization planning (SharePoint

Title	Description
<p>SharePoint 2010 Products: Virtualization Process</p>  <p>The diagram illustrates the virtualization process for SharePoint 2010 products. It shows a multi-tier architecture starting from a physical server layer, moving through a virtualization layer (Hypervisor) to multiple virtual machines (VMs). Each VM contains a specific SharePoint component, such as Search, Web Services, or Content Management. The process is divided into several stages, including planning, deployment, and management, with detailed steps and diagrams for each.</p> <p>Visio (http://go.microsoft.com/fwlink/?LinkId=195021) PDF (http://go.microsoft.com/fwlink/?LinkId=195022) XPS (http://go.microsoft.com/fwlink/?LinkId=195023)</p>	<p>Foundation 2010)</p> <ul style="list-style-type: none"> • Virtualization planning (SharePoint Server 2010)
<p>Governance for SharePoint Server 2010</p>  <p>The diagram provides a comprehensive overview of governance for SharePoint Server 2010. It is structured into several key areas: IT governance, information management governance, and application management governance. Each area is further broken down into specific components, processes, and roles. The diagram uses a mix of text, icons, and flowcharts to illustrate the relationships and flow of governance across the organization.</p> <p>Visio (http://go.microsoft.com/fwlink/?LinkId=200532) PDF (http://go.microsoft.com/fwlink/?LinkId=200533) XPS (http://go.microsoft.com/fwlink/?LinkId=200534)</p>	<p>Illustrates how to develop a governance plan that includes IT governance, information management governance, and application management governance.</p> <p>Use this diagram with the following articles:</p> <ul style="list-style-type: none"> • Governance overview (SharePoint Server 2010) • Governance features (SharePoint Server 2010)
<p>Duet Enterprise for Microsoft SharePoint and SAP Poster</p>	<p>Illustrates Duet Enterprise architecture for both the SAP and Microsoft environments, with detailed explanations of each area.</p>

Title	Description
 <p>Visio (http://go.microsoft.com/fwlink/?LinkID=208107&clcid=0x409)</p> <p>PDF (http://go.microsoft.com/fwlink/?LinkID=208108&clcid=0x409)</p> <p>XPS (http://go.microsoft.com/fwlink/?LinkId=208109&clcid=0x409)</p>	

Tips for printing posters

If you have a plotter, you can print these posters in their full size. If you don't have plotter, use the following steps to print on smaller paper.

► Print posters on smaller paper

1. Open the poster in Visio.
2. On the **File** menu, click **Page Setup**.
3. On the **Print Setup** tab, in the **Printer paper** section, select the size of paper you want to print on.
4. On the **Print Setup** tab, in the **Print zoom** section, click **Fit to**, and then enter **1 sheet across by 1 sheet down**.
5. On the **Page Size** tab, click **Size to fit drawing contents**, and then click **OK**.
6. On the **File** menu, click **Print**.

If you want to create posters that use the same symbols as these posters, you can download [Visio stencils for posters](http://www.microsoft.com/downloads/en/details.aspx?FamilyID=88e03d22-8f42-4c9d-94ef-d8e48322d677) (http://www.microsoft.com/downloads/en/details.aspx?FamilyID=88e03d22-8f42-4c9d-94ef-d8e48322d677).

Plan for sites and solutions (SharePoint Server 2010)

This section contains articles that will help you plan your Microsoft SharePoint Server 2010 site and solution components.

- [Fundamental site planning \(SharePoint Server 2010\)](#)

The articles in this section will guide you in planning sites that use SharePoint Server 2010 features.

- [Security planning for sites and content \(SharePoint Server 2010\)](#)

The articles in this section will help you plan permissions that control access to your sites and content.

- [Site and solution governance \(SharePoint Server 2010\)](#)

The articles in this section will help you plan how to set up your environment to host IT services and sandboxed solutions, and how to define the most appropriate information architecture for your business needs.

- [Plan for sandboxed solutions \(SharePoint Server 2010\)](#)

The articles in this section will help you plan sandboxed solutions that run in restricted execution environments within your enterprise.

- [Governance overview \(SharePoint Server 2010\)](#)

Introduces governance as an essential part of a successful Microsoft SharePoint Server 2010 deployment and explains why both information architecture and IT services are key components of a governance plan.

- [Plan for social computing and collaboration \(SharePoint Server 2010\)](#)

The articles in this section will guide you in planning solutions that implement social computing and collaboration capabilities in your enterprise.

- [Enterprise content management planning \(SharePoint Server 2010\)](#)

The articles in this section cover conceptual information about document management, records management, and managing digital assets.

- [Document management planning \(SharePoint Server 2010\)](#)

The articles in this section will help you plan document management solutions for your organization.

- [Records management planning \(SharePoint Server 2010\)](#)

The articles in this section describe records management in SharePoint Server 2010 and provide guidelines for planning your records management solution.

- [Plan digital asset libraries \(SharePoint Server 2010\)](#)

The articles in this section will guide you in planning solutions for sites that include digital assets such as video, audio, and images.

- [Plan Web content management \(SharePoint Server 2010\)](#)

The articles in this section will guide you in planning Web content management sites by using SharePoint Server 2010 features.

- [Plan managed metadata \(SharePoint Server 2010\)](#)

The articles in this section explain key concepts about managed metadata and provide guidance about how to use managed metadata in your SharePoint solution.

- [Business intelligence planning](#)

The articles in this section will guide you in planning Business Intelligence solutions on your enterprise data.

- [Business intelligence basics](#)

The articles in this section provide basic overview information about business intelligence capabilities in SharePoint Server 2010.

- [Plan for PerformancePoint Services \(SharePoint Server 2010\)](#)

The articles in this section help you plan your implementation of PerformancePoint Services and BI Dashboards.

- [Excel Services overview \(SharePoint Server 2010\)](#)

The articles in this section help you plan your implementation of Excel Services in your enterprise environment.

- [Plan for Visio Services \(SharePoint Server 2010\)](#)

The articles in this section help you plan your implementation of Visio Services.

- [Business data and processes planning \(SharePoint Server 2010\)](#)

The articles in this section will guide you in planning solutions that implement business processes on your enterprise's data.

- [Plan for Business Connectivity Services \(SharePoint Server 2010\)](#)

The articles in this section will help you plan solutions that connect your enterprise's external data to information workers who use SharePoint Web sites and Office 2010 applications.

- [Plan InfoPath Forms Services \(SharePoint Server 2010\)](#)

The articles in this section will help you plan solutions that use InfoPath forms to collect, customize, and validate business data used by workers to improve key business processes.

- [Plan workflows \(SharePoint Server 2010\)](#)

The articles in this section will help you plan and implement business processes in your SharePoint solutions.

- [Access Services planning](#)

The articles in this section help you plan your implementation of Access Services.

- [Plan site creation and maintenance \(SharePoint Server 2010\)](#)

The articles in this section contain guidance about how to create, maintain, and delete sites, as well as how to determine settings for quota templates and recycle bins.

- [Plan e-mail integration \(SharePoint Server 2010\)](#)

The articles in this section explain how to implement incoming e-mail and features that rely on outgoing e-mail.

See Also

[Plan for server farms and environments \(SharePoint Server 2010\)](#)

Plan Web content management (SharePoint Server 2010)

This section provides information that helps IT pros plan publishing sites by using Microsoft SharePoint Server 2010 features.

In this section:

- [Publishing features overview](#) describes the features that become available when publishing is enabled at the site collection and site levels for a nonpublishing site. This article also describes any dependencies between features, and it lists changes to the user interface that occur when publishing is enabled.
- [Plan Web pages](#) first introduces the elements of Web pages: master pages, page layouts, content pages, style sheets, Web Parts, Web Part zones, and server field controls. Next, this article provides guidance about how to plan each element of the Web pages in your publishing site.
- [Plan Web page authoring \(SharePoint Server 2010\)](#) describes the steps that are involved in planning how Web pages are authored.
- [Plan content approval and scheduling](#) contains general guidance about how to plan content approval and scheduling for use with SharePoint Server 2010 publishing sites.
- [Plan for caching and performance \(SharePoint Server 2010\)](#) provides information about how and when to use the BLOB cache, and it lists key considerations for planning to use it. This article also describes performance considerations for when to use Bit Rate Throttling, and it describes the limitations of upload file size restrictions.
- [Plan for large Pages libraries \(SharePoint Server 2010\)](#) describes the use of large Pages libraries in SharePoint Server 2010 publishing sites. Also, this article provides information to help you determine whether to use large Pages libraries with your publishing solution and information about how to plan for them.
- [Content deployment overview \(SharePoint Server 2010\)](#) provides an overview of the content deployment feature, describes how it works, and lists important considerations for using content deployment with your publishing solution.
- [Plan content deployment \(SharePoint Server 2010\)](#) discusses how to plan for using content deployment with your publishing solution.
- [Design content deployment topology](#) describes elements of topologies designed for content deployment and illustrates typical content deployment topologies.
- [Variations overview](#) provides an overview of the variations feature. It describes the elements of the variations feature, provides an overview of site and page creation for variation sites, lists some of the limitations of variations, and describes scenarios for using variations in SharePoint Server 2010

- [Plan variations](#) provides information about important items that you should consider when you are using variations in publishing sites, and it describes the tasks that are involved in planning a solution that uses variations in SharePoint Server 2010.

Publishing features overview

Publishing is the authoring and deploying of branded artifacts, content, custom assemblies, and configuration files across a Microsoft SharePoint Server 2010 farm. Publishing in SharePoint Server 2010 consists of two separate features. The SharePoint Server Publishing Infrastructure feature provides publishing functionality at the site collection level, and the SharePoint Server Publishing feature provides publishing functionality at the site level. The subset of features and functionality of each feature supports the goal of publishing as part of a Web content management solution.

This article only describes the features that become available when publishing is enabled at the site collection and site levels for a nonpublishing site. This article also describes any dependencies between features, and it lists changes to the user interface that occur when publishing is enabled. However, this article does not explain how to enable the publishing features, how to plan publishing sites, or how to convert nonpublishing sites to publishing sites.



Important:

Before you enable the publishing infrastructure and publishing features for a nonpublishing site, you should read this article to understand the specific publishing features that you want to use and to determine whether it is worth enabling the complete publishing infrastructure to get the benefit of only certain publishing features.

In this article:

- [About publishing sites](#)
- [About publishing features](#)
- [SharePoint Server Publishing Infrastructure features](#)
- [SharePoint Server Publishing features](#)
- [Other publishing features](#)

About publishing sites

The Publishing Portal site collection template and the Enterprise Wiki site collection template are the only two SharePoint Server 2010 site collection templates that are preconfigured to use the publishing features. Creating a site collection by using one of these two site collection templates automatically enables the publishing features for those site collections. By default, if the Publishing Portal site template is used, only the Publishing Site with Workflow site template and Enterprise Wiki site template are available to use to create a site within the site collection. A site collection administrator can enable other site templates for use within the site collection by using the Page Layout and Site Template Settings page.

Nonpublishing sites are all the other site templates that are available in SharePoint Server 2010, such as the Team Site and Document Workspace template. You can enable the SharePoint Server Publishing Infrastructure feature at the site collection level, and then enable the SharePoint Server

Publishing feature for the root site of the site collection and any sites below it in the site hierarchy. This enables all the publishing features that you typically get when you create a site by using a publishing site template, in addition to the standard features of the nonpublishing site. For a complete list of the site templates available in SharePoint Server 2010, see [Sites and site collections overview \(SharePoint Server 2010\)](#).

About publishing features

The SharePoint Server Publishing Infrastructure feature provides publishing functionality at the site collection level, and the SharePoint Server Publishing feature provides publishing functionality at the site level. The subset of features contained in each of these primary features is collectively known as “publishing features.” Publishing features are all the features that are part of a preconfigured publishing site or that are added when publishing is enabled at the site collection and site level. When publishing is enabled, all publishing features are automatically enabled. You cannot select individual publishing features, such as variations, to be enabled separately without enabling other publishing features. All publishing features are either active or inactive. However, even though you can decide to enable the publishing features, you do not have to use them all.

SharePoint Server Publishing Infrastructure features

This section describes the publishing features that are enabled when the SharePoint Server Publishing Infrastructure feature is enabled on a nonpublishing site collection.

Site templates

A site template is a predefined site configuration that determines, for example, the lists, files, Web Parts, Features, or settings with which to provision a new SharePoint site. When you enable the SharePoint Server Publishing Infrastructure feature, the following publishing site templates are added and are available to use when a new site is created:

- Publishing Site
- Publishing Site with Workflow
- Enterprise Wiki

For more information about the site templates that are available in SharePoint Server 2010, see [Sites and site collections overview \(SharePoint Server 2010\)](#) and [Site templates and definitions](#).

Groups and permission levels

SharePoint groups enable you to manage sets of users instead of individual users. The ability to view, configure, or manage a site is determined by the permission level that you assign to a user or group. When you create a site collection by using a nonpublishing site template, SharePoint Server 2010 automatically creates a standard set of groups and permission levels. When you enable the SharePoint Server Publishing Infrastructure feature, other groups and permission levels are added to the site

collection. These groups and permission levels enable you to assign users to specific publishing-related roles. For example, only users who have the Approve permission, or who are in the Approvers group can edit and approve pages, list items and documents for publishing.

The following groups are added to the site collection:

- Approvers
- Designers
- Hierarchy Managers
- Quick Deploy Users
- Restricted Readers
- Style Resource Readers

The following permission levels are added to the site collection:

- Approve
- Manage Hierarchy
- Restricted Read

By default, sites that are created below the site collection use the groups and permission levels from the parent site. For more information about groups and permission levels, see [Determine permission levels and groups \(SharePoint Server 2010\)](#).

Site settings

When you enable the SharePoint Server Publishing Infrastructure feature, the following changes are made to the Site Settings page:

- In the **Site Administration** section, the following links are added at both the site collection and the site level:
 - Content and structure
 - Content and structure logs
 - Searchable columns
- In the **Look and Feel** section, the **Quick launch** and **Top link bar** links are removed, and the **Navigation** link is added at both the site collection and the site level.
- In the **Site Collection Administration** section, the following links are added at the site collection level only:
 - Site collection navigation
 - Variations
 - Variation labels
 - Variation logs
 - Translatable columns
 - Suggested content browser locations

Navigation

In addition to the changes that are made to navigation links on the Site Settings page, the following navigation changes are made when you enable the SharePoint Server Publishing Infrastructure feature:

- The top link bar is replaced with the global navigation menu.
- Default settings for the global navigation menu and the Quick Launch menu are specified.

For more information about navigation, see [Site navigation overview \(SharePoint Server 2010\)](#).

Theme changes

Themes provide a quick and easy way to apply colors and fonts to sites in SharePoint Server 2010. Each site can apply a theme directly to itself. When you enable the SharePoint Server Publishing Infrastructure feature, the **Inherit Theme** and **Apply Theme** sections are added to the Site Theme page. These options allow a site administrator to specify whether a site should inherit the theme from the parent site or should use its own theme. These themes also allow the site administrator to specify whether to apply the selected theme only to the current site or to the current site and all sites below it in the site hierarchy. For more information about themes, see [Themes overview \(SharePoint Server 2010\)](#).

Master pages and page layouts

Master pages and page layouts dictate the overall behavior and appearance (look and feel) of a SharePoint site. Master pages contain controls that are shared across multiple page layouts, such as navigation, search, or language-preference for multilingual sites. Page layouts contain field controls and Web Parts. The top-level site for a site collection that is hosted on SharePoint Server 2010 has a special document library called the Master Page Gallery library. All page layouts and master pages are stored in this document library. When you enable the SharePoint Server Publishing Infrastructure feature, the following files and folders are added to the Master Page Gallery library:

- New master pages and page layouts, such as article pages and a Wiki page that is used by publishing sites, are added.
- A new folder is created in the Master Page Gallery library, and is named based on the language that was used for the SharePoint Server 2010 installation. For example, if the English version was installed, the folder name is **en-us**. This folder contains a folder named **Preview Images**, which contains the thumbnail preview images of the Publishing page layouts.



Note:

If other language packs have been installed, each language will have its own folder that contains a **Preview Images** folder in the Master Page Gallery library.

- A new folder named **Editing Menu** is created in the Master Page Gallery library, and it contains XML files that can be used to customize the page editing menus. For information about how to customize the page editing menus, see [How to: Customize Page Editing Toolbar Components](#).

For more information about master pages and page layouts, see [Page Layouts and Master Pages](#).

Images and style sheets

When you enable the SharePoint Server Publishing Infrastructure feature, the following items are added to the Style Library:

- Cascading style sheets for default styles and styles that can be customized.
- Images for user interface elements, such as bullets and arrows.
- Alternate preview images for the media player.
- XSL style sheets for applying styles to data-driven Web Parts such as Summary Links, Content Query and Table of Contents.

For information about how to customize styles, see [How to: Customize Styles](#).

Document libraries and lists

Different document libraries and lists are created for a site collection, depending on the site template that is used to create the site collection. When you enable the SharePoint Server Publishing Infrastructure feature, the following document libraries and lists are added to the root site of site collection:

- **Content and Structure Reports** This list is used to customize the queries that appear in the **View** list in the Site Content and Structure tool.
- **Reusable Content** This list contains HTML or text content that can be inserted into Web pages.
- **Site Collection Documents** This library stores documents that are used throughout the site collection.
- **Site Collection Images** This library stores images that are used throughout the site collection.

Content types

A content type defines the columns of a list item, a document, or a folder. When you enable the SharePoint Server Publishing Infrastructure feature, SharePoint Server 2010 adds, at the site collection level and at the list and library level, more content types that are used by sites within the site collection.

At the site collection level, publishing content types such as Page and Page Layout, and page layout content types such as Article Page and Enterprise Wiki Page are added. Two additional content types, Reusable HTML and Reusable Text, are added specifically for the Reusable Content list.

For more information about content types, see [Content type and workflow planning \(SharePoint Server 2010\)](#).

Columns

Metadata is information about a document that is used to categorize and classify your content. Each item of metadata that is associated with a content type is a column, which is a location in a list to store information. When you enable the SharePoint Server Publishing Infrastructure feature, the following columns are added:

- New page layout columns such as Byline and Page Content, and Publishing columns such as Article Date, Scheduling Start Date and Scheduling End Date are added at the site collection level. A custom Wiki Categories column that uses managed metadata for Wiki pages is also added at the site collection level.
- New columns for the Reusable Content list and the Content and Structure Reports list are added.

For more information about columns, see [Content type and workflow planning \(SharePoint Server 2010\)](#).

Web Parts

Web Parts are user interface elements that are used in pages on SharePoint sites to present information that is pulled from multiple data sources. When you enable the SharePoint Server Publishing Infrastructure feature, the following Web Parts are added at the site collection level and are available to use in all sites that are created within the site collection:

- Content Query Web Part
- Media Web Part
- Summary Links Web Part
- Table Of Contents Web Part

Page editing menu

The page editing ribbon is a panel of user interface elements that provide page information and ways to use the page. The user can use the page editing menu on the page editing ribbon to add text, images, and rich media to a page, check in the page to share a draft, and approve a pending version of the page for publishing. When you enable the SharePoint Server Publishing Infrastructure feature, the following changes are made to the page editing menu:

- The **Publish** tab is added to the master page.
- Under **Editing Tools**, on the **Format Text** tab, the following changes are made when a rich text field is selected:
 - The **Spelling** group and a **Spelling** button are added.
- Under **Editing Tools**, on the **Insert** tab, the following items are added when a link is selected:
 - In the **Media** group, the **From SharePoint** selection is added to the drop-down list of the **Picture** button.
 - In the **Links** group, the **From SharePoint** selection is added to the drop-down list of the **Link** button.
 - In the **Media** group, a **Video and Audio** button is added.
 - In the **Web Parts** group, **Media Web Part** is added to the **Media and Content** category in the Web Part selection menu when the **Web Part** button is clicked.
- Under **Link Tools**, on the **Format** tab, the following items are added:

- In the **Link** group, a **Select Link** button is added.
- In the **Properties** group, a **Bookmark** text box is added.
- Under **Picture Tools**, on the **Design** tab, the following items are added:
 - In the **Select** group, the **From SharePoint** selection is added to the **Change Picture** list.
 - The **Spacing** group is added.
 - In the **Spacing** group, the **Horizontal Space** and **Vertical Space** options are added.
- A **Media** tools menu is added to the page menu when a Media Web Part is selected.

Timer jobs

A timer job runs a specific Windows service for SharePoint Server 2010. The timer job contains a definition of the service to run and specifies how frequently the service is started. Each timer job has its own default schedule for when the job runs. You can change the frequency with which each job runs on the Job Definitions page on the Central Administration Web site.

When you enable the SharePoint Server Publishing Infrastructure feature, the following timer jobs are enabled on the server that hosts the Central Administration Web site:

- **Notification Timer Job** Sends e-mail to the item owner when an item is about to expire.
- **Scheduled Approval** Publishes approved pages according to the specified start date and time. By default, this timer job runs every minute.
- **Scheduled Unpublish** Unpublishes pages according to the specified end date and time. By default, this timer job runs every minute.
- **Variations Create Hierarchies Job Definition** Creates a complete variations hierarchy by creating all variation sites and pages from the source variation site, based on the variation labels. By default, this timer job runs once a day.
- **Variations Create Page Job Definition** Creates pages on the target variation sites when the **Automatic Creation** option has been disabled and a user manually creates a new page. By default, this timer job runs every hour.
- **Variations Create Site Job Definition** Creates variation sites when the **Automatic Creation** option has been disabled and a user manually creates a new variation site. By default, this timer job runs every five minutes.
- **Variations Propagate Page Job Definition** Updates pages on target variation sites after a page on the source variation site has been approved or after it has been manually submitted by a user. By default, this timer job runs every hour.
- **Variations Propagate Site Job Definition** Creates variation sites when the **Automatic Creation** option is enabled. By default, this timer job runs every five minutes.

For information about timer jobs, see [View timer job status \(SharePoint Server 2010\)](#).

SharePoint Server Publishing features

This section describes the publishing features that are enabled when the SharePoint Server Publishing feature is enabled on a nonpublishing site.

Site settings

When you enable the SharePoint Server Publishing feature, the following changes are made to the Site Settings page:

- In the **Galleries** section, the **Master pages** link is removed, and it is replaced with the **Master pages and page layouts** link at both the site collection and site level.
- In the **Site Administration** section, a **Site output cache** link is added at the site level only.
- In the **Look and Feel** section, the following links are added at both the site collection and site level:
 - **Master Page**
 - **Page layouts and site templates**
 - **Welcome Page**
- In the **Site Actions** section, the **Save site as template** link is removed at both the site collection and site level.
- In the **Site Collection Administration** section, the following links are added at the site collection level only:
 - **Site collection cache profiles**
 - **Site collection object cache**
 - **Site collection output cache**

Regional settings

When you enable the SharePoint Server Publishing feature, the **Subsite Settings** section is added to the Regional Settings page. This enables you specify whether all sites below the current site should inherit the regional settings set for the current site.

Document libraries and lists

When you enable the SharePoint Server Publishing feature, the following document libraries and lists are added:

- **Documents** This library stores documents that are used on pages in the site.
- **Images** This library stores images that are used on pages in the site.
- **Pages** This library stores pages that are created in the site.
- **Workflow Tasks** This list stores workflow tasks that are created in the site.



Note:

If you later disable the SharePoint Server Publishing feature, libraries or lists that contain content are not removed from the site, but empty libraries or lists are removed from the site.

In addition to the libraries and lists that are created, the following changes are made to document library settings:

- On the Document Library Settings page, in the **General Settings** section, a **Manage item scheduling** link is added.
- On the Versioning Settings page, the following changes are made:
 - The **Document Version History** option is set to **Create major and minor (draft) versions**. This option determines what versions are created when a file is edited in the Pages library.
 - The **Draft Item Security** option is set to **Only users who can edit items**. This option determines who can see draft items in the Pages library.
 - The **Require Check Out** option is set to **Yes**. This option requires documents to be checked out before they can be edited.

Page editing menu

When you enable the SharePoint Server Publishing feature, the following changes are made to the page editing menu:

- The **Publish** tab and **Publish** button are added to the master page.
- The following items are added to the **Publish** tab:
 - If the Publishing Approval Workflow feature is enabled for the site collection, and the Publishing Approval workflow template is associated with the document library, the following changes are made:
 - A **Workflows** group is added.
 - In the **Workflows** group, a **Start a Workflow** button is added.
 - In the **Workflows** group, a **Status** button is added.
 - In the **Workflows** group, a **View Tasks** button is added.
 - If the **Quick Deploy** job is enabled for the site collection's content deployment path, in the **Publishing** group, a **Quick Deploy** button is added.
 - If item scheduling is enabled for the document library, in the **Publishing** group, a **Schedule** button is added.
- On the **Page** tab, the following changes are made:
 - In the **Manage** group, the **Edit Properties** button is enabled.
 - In the **Manage** group, the **Rename Page** button is removed.
 - In the **Page Actions** group, the following items are added.
 - i. A **Preview** button is added.
 - ii. A **Page Layout** button is added.
 - iii. A drop-down list that contains page layouts is added to the **Page Layout** button.

- In the **Page Actions** group, a **Draft Check** button is added.
- Under **Editing Tools**, on the **Format Text** tab, the following changes are made:
 - The **Spelling** group and the **Spelling** button are added.
 - The **Layout** group and the **Text Layout** button are removed.
- Under **Editing Tools**, on the **Insert** tab, the following changes are made:
 - In the **Content** group, a **Reusable Content** button is added.
 - A customizable drop-down list is added to the **Reusable Content** button.
- If a page was created by using document conversions, a **Source Document Tools** menu is added to the page editing bar, and the following items are added:
 - Under **Source Document Tools**, a **Document** tab is added.
 - On the **Document** tab, a **View and Update** group is added.
 - In the **View and Update** group, a **View Document** button is added.
 - In the **View and Update** group, an **Update Page** button is added.

Other changes

When you enable the SharePoint Server Publishing feature, the following changes are made:

- Users can no longer create pages that have a space in the name. Spaces are automatically converted to a dash '-'.
- The **Manage Content and Structure** link is added to the **Site Actions** menu. This opens the Site Content and Structure tool for the entire site collection.

Other publishing features

When you enable both the SharePoint Server Publishing Infrastructure feature and the SharePoint Server Publishing feature, the following Publishing features are enabled:

- **Content deployment** You can use content deployment to deploy content from a source site collection to a destination site collection. Content deployment is administered at the farm level, on the Central Administration Web site. If your site uses content deployment and a farm administrator has enabled the Quick Deploy job, a **Quick Deploy** button will be added to the **Publishing** group on the **Publish** tab of the page editing menu. The Quick Deploy job enables users, such as authors and editors, to quickly deploy a Web page to the destination site collection. By default, a Quick Deploy job runs automatically every 15 minutes. When a user clicks the **Quick Deploy** button on a page, that page is included in the next automatically scheduled Quick Deploy job. For more information about content deployment and Quick Deploy jobs, see [Content deployment overview \(SharePoint Server 2010\)](#).
- **Variations** The variations feature in SharePoint Server 2010 makes content available to specific audiences on different sites by copying content from a source variation site to each target variation

site. When the SharePoint Server Publishing Infrastructure feature and the SharePoint Server Publishing feature are enabled, the **Variations**, **Variations Labels** and **Variations Logs** links are added to the **Site Collection Administration** section of the Site Settings page. If a variations hierarchy has been created, the **Variations** group is added to the page editing toolbar for all publishing pages on all sites within the site collection. However, the buttons in the **Variations** group are enabled only when a page is part of a source variation site. For more information about variations, see [Variations overview](#).

- **Object and output caching** The object cache reduces the amount of traffic between the Web server and the SQL database by storing objects—such as lists and libraries, site settings, and page layouts—in memory on the front-end Web server. The page output cache stores the rendered output of a page and uses cache profiles that specify how long items should be held in the cache. When the SharePoint Server Publishing Infrastructure feature and the SharePoint Server Publishing feature are enabled, links to configure these caches are added to the Site Settings page for the site and site collection. For more information about the object and page output caches, see [Cache settings operations \(SharePoint Server 2010\)](#).

See Also

[Plan Web content management \(SharePoint Server 2010\)](#)

Plan Web pages

When you plan to publish Web pages in Microsoft SharePoint Server 2010, you design the appearance of your published content, determine where authors can add content on pages, and control which authoring features authors can use. An effective plan for Web pages helps ensure that each type of content that your organization publishes is designed correctly and is available to achieve your publishing goals.

To help you understand your design options, this article first introduces the elements of publishing pages: master pages, page layouts, content pages, style sheets, Web Parts, Web Part zones, and server field controls. Next, this article contains guidance about how to plan each element of the Web pages in your publishing site. Because the design and configuration of page layouts helps restrict what authors can do on Web pages, this article includes guidance about how to use page layouts to restrict authoring. However, this article does not describe how to create master pages, page layouts, or content pages, nor does it describe how content authors create Web pages.

In this article:

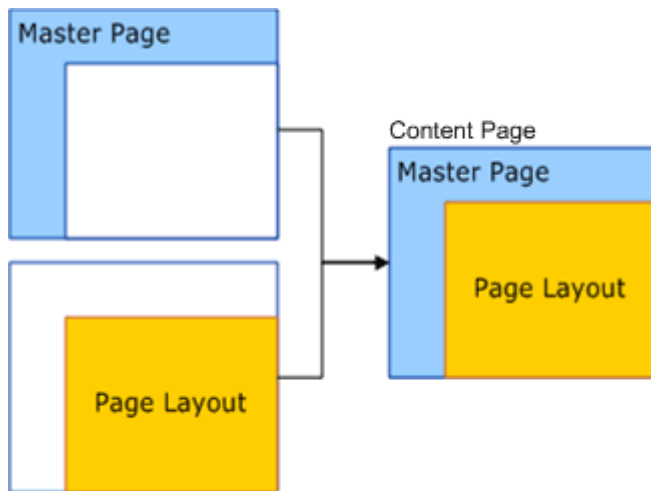
- [Web pages overview](#)
- [Plan master pages](#)
- [Plan page layouts](#)
- [Plan content pages](#)
- [Using page layouts to restrict authoring](#)
- [Web page planning worksheet](#)

Web pages overview

When a SharePoint Server 2010 site user opens a Web page in a SharePoint site, that page is rendered based on a set of elements that have each been planned and designed separately in the Web site. Separating elements of a page in this manner enables site planners and designers to treat different elements of the site in unique ways. For example, a site's branding and navigation can be planned and designed separately from the design of the site's content pages so that the branding can be applied across all site content and can be updated in one location. Similarly, the layout of pages can be designed separately from the content of pages so the same content can be displayed in different ways.

A Web page that is based on SharePoint Server 2010 is an ASP.NET file (.aspx) page that is dynamically rendered out of its constituent parts. The two primary parts of a Web page are the master page and the page layout. Master pages contain controls that are shared across multiple page layouts, such as navigation, search, or language-preference for multilingual sites. Page layouts contain field controls and Web Parts. When you create a Web page, content in the page is stored as list items in the Pages library. This Web page is referred to as a content page, because it contains the content that is

displayed to users when they view the page on the Web site. The following figure shows how page layouts and master pages work together to create the layout for a Web page.



The following sections describe master pages, page layouts, and content pages in more detail.

Master pages

A master page defines the outer frame of the Web page. It contains the elements that you want all pages in your site to share, and it provides a single place to control all those elements. Typically, a site uses a single master page, although large Internet sites might use more. For example, a corporate Web site that publicizes more than one product could use separate master pages so that the content for each product is branded correctly.



Note:

There are two kinds of master pages: site master pages and system master pages. The site master page is used on published Web pages in your site. The site master page is what site users and visitors see when they view published pages. The system master page provides the layout of pages in the site that is used by site designers and authors when they work with the site's user interface. The system master page is also used in some team site templates, such as the Enterprise Wiki and the Document Workspace site templates. This article primarily describes planning considerations for site master pages.

Master pages for all sites in a site collection are stored in the Master Page Gallery in the top-level site in the site collection. Because the Master Page Gallery is a SharePoint document library, master pages have all the features of documents in SharePoint Server 2010, such as versioning, auditing, workflow, check-in and check-out, and content approval.

Typically, master pages include the following elements:

- Branding elements, such as corporate logos and color schemes
- Shared navigation elements

- Shared features, such as search commands and Help commands
- Links to cascading style sheets. (Cascading style sheets control the page appearance, colors, and fonts.)

The publishing site templates that are included in SharePoint Server 2010 include site master pages that you can use as a starting point in your page design. To customize an existing master page or to create a new one, use Microsoft SharePoint Designer 2010 or Microsoft Visual Studio 2010. For more information, see [How to: Create a Minimal Master Page](#).

Page layouts

A page layout is an Active Server Pages (ASPX) page that defines a layout for a specific kind of content page. When a SharePoint site user opens a content page in a browser, the page layout that is associated with that page is first combined with the master page, which supplies the outer frame of the page, and then the contents of the page are inserted into the field controls on the page layout.

Because a page layout displays content that is stored in the columns of a content type, it must be designed for a particular content type. For example, a page layout that is associated with the Article Page content type would have several field controls, including the following:

- A Page Content field control to hold the contents of the Page Content column in the Article Page content type
- A Page Image field control to hold the image that is linked to from the Page Image column of the Article Page content type

Although a page layout must be designed for a single content type, a content type can be associated with multiple page layouts. For example, SharePoint Server 2010 includes two page layouts for the Article Page content type: one that displays the image on the left side of the page and another that displays the image on the right. For more information about content types, see [Content type and workflow planning \(SharePoint Server 2010\)](#).

Along with controls to display the contents of a page, a page layout can include other page elements, including the following:

- **Web Parts** A control that page authors can insert into a Web Part zone on a page and then configure.
- **Web Part zones** A specified area on a Web page that is a container for Web Parts.
- **Field controls** A control that is added directly to a page layout. For more information about field controls, see [Field Controls and Control Templates](#).
- **Cascading style sheets links** Cascading style sheets control the page appearance, colors, and fonts.

For example, a page layout for a business article could include a field control that displays a stock ticker. The stock ticker would be displayed together with other page content when that page layout is used.

Like master pages, page layouts for all sites in a site collection are stored in the Master Page Gallery in the top-level site in the site collection. Because the Master Page Gallery is a SharePoint library, page

layouts also have all the features of documents in SharePoint Server 2010, such as versioning and content approval. Publishing sites that you create by using SharePoint Server 2010 include page layouts that you can use as a starting point in your content page design. To customize an existing page layout or to create a new one, use Microsoft SharePoint Designer 2010 or Microsoft Visual Studio 2010.

Content pages

All content pages for a publishing site are stored in a single Pages library. Each item in a Pages library is a single Web page. Because the Pages library is a SharePoint library, the Web pages that it contains have all the features of documents in SharePoint Server 2010, such as versioning, auditing, workflow, check-in and check-out, and content approval.



Note:

Although all publishing pages in a site are in a single Pages library, Web solutions that are based on SharePoint Server 2010, such as intranet sites and Internet presence sites, typically consist of a hierarchy of sites, each with its own Pages library.

Authors create pages by selecting **New Page** on the **Site Actions** menu and edit them by selecting **Edit Page** on the **Site Actions** menu. When creating a new page, authors enter a name for the new page and then immediately begin authoring content on the page. To change the content type and page layout, authors select **Page Layout** in the **Page Actions** group on the **Page** tab of the page to be modified. To add content, select images, and do other editing tasks, authors use the **Format Text** and **Insert** tabs under **Editing Tools** on the page to be modified.

The columns that are associated with the content type for a Web page contain the HTML content for that page. They also contain links to images that appear with the page and a link to the page layout that is associated with the page.

Each column of content for a page is associated with a particular field control on the page layout that is associated with the page. For more information about field controls on page layouts, see [Page Layout Model](#).

Plan master pages

Master pages provide the shared framing elements of the page. These include the branding of the site, its navigation features, and other common elements such as search fields and Help commands. The site master page supplies the context of the page and should remain consistent as the user interacts with your site. To ensure that users have a consistent experience when they move from one page to another throughout a single site in a site collection, we recommend that you leave the site master page unchanged. To supply consistent branding and user interface, you can use the same site master page across all sites in your site collection.

You can change the master page that is used in other sites in your site hierarchy to change the branding in some sites. For example, an Internet presence site might consist of multiple sites that each present a different brand of products. You can change the site master page for each site in the site hierarchy to reflect the distinct product brand that each site presents.

Before you plan master pages, you should plan your site structure, as described in [Plan sites and site collections \(SharePoint Server 2010\)](#). To plan master pages, use the Master page data sheet in the [Web page planning worksheet](#).

Plan page layouts

A page layout defines a layout for a content page by providing field controls into which the contents of the content page are inserted. The field control displays the contents. Each page layout is associated with a particular content type, and multiple page layouts are often available for a single content type. For example, you can assign multiple page layouts to a single content type to provide alternate layouts for localized versions of content or to add or remove the display of certain fields and features from a page layout. You can create or customize a page layout, which includes adding new controls to display content together with additional controls such as Web Parts and server controls, by using Microsoft SharePoint Designer 2010 or Microsoft Visual Studio 2010.

SharePoint Server 2010 includes the following set of page layouts for each page content type.

- **Article Page** contains the following page layouts:

This page layout	Contains these page elements
Body only	A title and page content
Image on left	A title, page content, a page image on the left, and areas for a byline, article date, and image caption
Image on right	A title, page content, a page image on the right, and areas for a byline, article date, and image caption
Summary links	A title, page content, article date, byline, and a Summary Links Web Part in which authors can add a list of hyperlinks

- **Enterprise Wiki Page** contains a single page layout, Basic Page, which contains the content, the page rating, and the categories page elements.
- **Project Page** contains a single page layout, Basic Project Page, which contains the content, the page rating, the categories, , the page contact, and the task status page elements, and it contains a single link to the project Web page.
- **Redirect Page** contains a single page layout, Redirect, which contains a single hyperlink to which users who view the page are redirected.
- **Welcome Page** contains the following page layouts:

This page layout	Contains these page elements
Blank Web Parts page	A content area and multiple Web Part zones to which authors can add Web Parts
Splash	Only an image and two Summary Links Web Parts in which authors can add hyperlinks
Summary links	Content and image areas, together with two Summary Links Web Parts
Table of contents	Content and image areas, together with a Table of Contents Web Part to display a hyperlinked table of contents of the site

If you are using the page content types and layouts that are included with SharePoint Server 2010, there are no additional planning steps that you must follow. Authors can select page content types and associated layouts when they create new pages. However, if you add new fields to a page content type or if you create new custom content types for publishing pages, you should plan page layouts that reflect the new or changed content types.

You can also change a page layout by adding Microsoft ASP.NET 3.5 controls, such as Web Parts and Web Part zones, to the page. For example, you can add a Content Query Web Part, which displays a set of links that are returned by a configurable query, to a page layout. However, if you place a Web Part on a page layout outside a Web Part zone, you must configure the Web Part, and authors will be unable to change its configuration. For example, if you add a Content Query Web Part directly to a page layout, the query that you configure when the Web Part is added is permanently set and authors cannot modify it.

To plan page layouts for content types such as Article Pages, Enterprise Wiki Pages, Project Pages, and Welcome Pages, use the Page layouts data sheet tab in the [Web page planning worksheet](#).

Plan content pages

Each content page in SharePoint Server 2010 consists of text, images, and other content that is stored as an entry in a Pages library. Planning the content pages includes the following:

- Determining the page content types that meet your content needs
- Determining the columns to use for storing content, for each page content type.

SharePoint Server 2010 includes the following page content types:

- **Article Page** The most common content page type. This page is designed for general-purpose Web page content. It includes the following:
 - Columns for images and image captions
 - A column for page content

- Columns for links that appear with the page
- A column for the byline
- A column for the article date
- **Enterprise Wiki Page** The primary content page type for an Enterprise Wiki site. It includes the following:
 - A column for page content
 - Columns for ratings and number of ratings
 - A column for wiki categories
- **Project Page** A page to provide basic information that describes a project. This content type inherits from the Enterprise Wiki Page content type, instead of the Page content type. It includes the following:
 - A column for page content
 - Columns for ratings and number of ratings
 - A column for a link to the project Web page
 - A column for task status
 - A column for wiki categories
- **Redirect Page** A page to redirect the reader to another page. It includes a column for the redirect URL.
- **Welcome Page** Typically, the home page of a publishing site. It includes the following:
 - Columns for images to display
 - A column for page content
 - Columns for links that appear with the page

Additionally, because all these page content types inherit from the generic Page content type either directly, or through their parent content type, they all include the following:

- Columns for scheduling the page's start and end dates
- Columns for describing contact information for the author
- An image that appears with the page when it is listed in a table of contents or on another list
- Information for targeting audiences
- A column for comments about the page

When you plan content pages, we recommend that you use the page content types that are included in SharePoint Server 2010 as a starting point. The Article Page, Enterprise Wiki Page, Project Page, and Welcome Page content types are intended to be generally useful and to apply across various contexts. The primary content column in these content types is the Page Content column, which can hold HTML content. By using HTML and cascading style sheets to control the appearance of their content, authors and site designers might not have to design other content types. Also, by carefully selecting which layout to use for each kind of content, based on the Article Page, Enterprise Wiki Page, Project Page,

or Welcome Page, you can introduce more variety in your content presentation without introducing additional content types. For more information, see [Plan page layouts](#).

To plan content pages, use the corresponding data sheet tabs in the [Web page planning worksheet](#).

Using page layouts to restrict authoring

Depending on your publishing goals, you can restrict how much freedom authors have to format their Web page content or to add items such as images and hyperlinks to pages in your site. For example, in a highly controlled Internet presence site, you might want all formatting to be defined in cascading style sheets that are associated with your page layouts, and you might want to block writers from overriding style definitions by using inline formatting. In contrast, in a collaborative site, you might want to give authors full freedom to format their pages and add other page items, such as Web Parts that provide views of data. For example, in an intranet site that is used to collaborate on product specifications, you might want to enable authors to freely use styles, hyperlinks, images, and Web Parts to maximize their ability to communicate their ideas.

You can put restrictions on page layouts in the following ways:

- You can set properties on field controls that restrict what authors can do.
- You can remove Web Part zones to restrict authors from inserting and configuring Web Parts on their pages, or you can set restrictions on Web Part zones to limit how authors can use them.

The following table shows recommendations for restricting page layouts based on three levels of authoring environments:

Level of control	Typical site	Restriction recommendations
Tight	Internet presence	Strict limitations on editing field controls; other field control limitations, such as no hyperlinks from image field controls; Web Parts are put directly on the page layout and not in Web Part zones
Moderate	Enterprise intranet portal site	Moderate or no limitations on editing field controls; Web Part zones that contain Web Parts, but authors are restricted from adding/removing Web Parts
Loose	Divisional or team site or Enterprise Wiki	No limitations on editing field controls; Web Part zones allowed

Use the Page layouts data sheet tab in the [Web page planning worksheet](#) to record your decisions about restricting authoring features on content pages.

Setting restrictions on field controls

By opening your site in Microsoft SharePoint Designer 2010 or Microsoft Visual Studio 2010 you can edit the tags that are associated with field controls to restrict the kinds of SharePoint Server 2010 authoring features that authors can use when they edit pages in the browser window. For example, on field controls that are bound to columns of the Publishing HTML type, you can enable or restrict the following features:

- Setting fonts
- Inserting images
- Inserting tables
- Adding hyperlinks
- Adding text markup, such as bold and italic
- Adding Web Parts

You can set authoring restrictions on other column types. For example, on field controls that are bound to columns of the Publishing Image type, you can enable or restrict hyperlinks from images.

When you restrict an authoring feature on a page layout in Microsoft SharePoint Designer 2010 or Microsoft Visual Studio 2010, the related page editing commands in SharePoint Server 2010 become unavailable. For example, if you restrict table editing in a field control that contains content of the Publishing HTML type, table editing commands, such as **Insert Table** are unavailable under **Editing Tools** on the **Insert** tab.

Allowing or restricting Web Part zones

A Web Part is a server control that authors can insert in Web Part zones on pages. A Web Part zone is a specified area on a Web page that is a container for Web Parts. Web Parts display information based on their functionality, such as presenting site navigation links, list content, or database analytical information.

When a page layout includes one or more Web Part zones, the Web Part zones are available on pages that are using that layout, which enables authors to insert available Web Parts onto their content pages. If you enable authors to insert Web Parts on pages, you reduce your control over users' experience of the site. For example, an author could insert a Table of Contents Web Part onto a page that exposes parts of your site that you do not want users to move to from the current page.

You can restrict authors from adding Web Parts to pages by opening the associated page layouts in Microsoft SharePoint Designer 2010 or Microsoft Visual Studio 2010 and removing Web Part zones from them or by removing the HTML field controls. Similarly, when you design new page layouts, omit Web Part zones to limit authors' ability to add functionality to the pages that are associated with those page layouts.

You can also include Web Part zones in page layouts but restrict their usage. By setting a Web Part zone's properties, you can populate the Web Part zone with one or more Web Parts and enable authors to edit the properties of those Web Parts but not let them add other Web Parts to the Web Part zone.

Web page planning worksheet

Download an Excel version of the [Web page planning worksheet](#). Use this worksheet to record your decisions about what master pages your site needs, columns for specific page content types, and authoring restrictions on page layouts.

See Also

[Plan content approval and scheduling](#)

Plan Web page authoring (SharePoint Server 2010)

Web page authoring is the process by which authors add content to a publishing site such as a public-facing Internet site. Web page authoring is available on a site when you create a Microsoft SharePoint Server 2010 site by using one of the publishing site templates, or when the SharePoint Server Publishing Infrastructure feature is activated for a site collection and the SharePoint Server Publishing feature is activated for a site. For information about publishing site templates, see [Sites and site collections overview \(SharePoint Server 2010\)](#).

Before reading this article, you should read [Plan Web pages](#), which describes page layouts, field controls, and other elements of Web pages that are mentioned in this article. This article describes the steps involved in planning how Web pages are authored. This article does not describe how to author Web pages.

In this article:

- [About planning Web page authoring](#)
- [Plan ribbon authoring experience](#)
- [Plan managed metadata](#)
- [Plan reusable content](#)
- [Plan dictionary customizations](#)
- [Plan additional resources](#)
- [Web page authoring planning worksheet](#)

About planning Web page authoring

Planning Web pages involves understanding how Web pages are designed and deciding which elements belong on the Web pages for your site. Planning Web page authoring involves understanding how Web pages are created. SharePoint Server 2010 supports browser-based authoring. Content creators work directly in the Web browser by using SharePoint Server 2010 browser-based editing features such as the **Format Text** tab under **Editing Tools** on the ribbon.

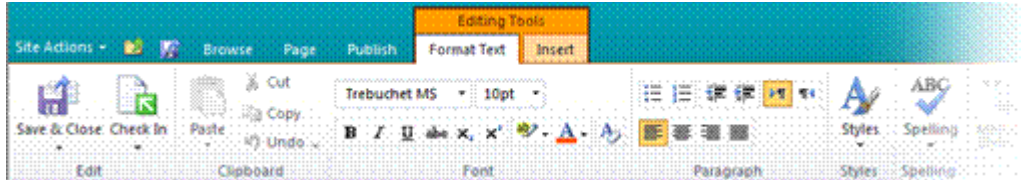
Planning browser-based authoring includes planning which resources, page layouts, supporting content, such as images and videos, and commands to hide from or show to authors and planning the editing experience in the field controls in which authors create content. It also includes planning for reusable content, planning dictionary customizations, and planning for additional resources that are needed by page authors.

A related set of planning considerations — planning how content will be approved and published — is discussed in [Plan content approval and scheduling](#).

Plan ribbon authoring experience

The ribbon contains UI elements that provide access to page editing commands and related tools, together with publishing and workflow commands, in addition to most other commands in SharePoint Server 2010. This ribbon is available to team members who have at least the Contribute permission level.

When a page is checked out for editing, and the pointer is positioned in the **Page Content** field, **Editing Tools** is displayed. **Editing Tools** contains the **Format Text** tab and the **Insert** tab, which contain the commands that authors use to format text and insert content elements, such as images, links, and reusable content. The following illustration shows the ribbon with the **Format Text** tab displayed:



Other contextual tabs or groups of tabs are displayed, based on the page element that is selected. For example, if you insert a table onto a page, **Table Tools** is displayed, and contains a **Layout** tab and a **Design** tab.

You can plan and implement new ribbon commands to provide added functionality for your content team. For example, if your organization has a process for creating and incorporating images in your documents that you want to automate, you can add a command to the **Page** tab on the ribbon.

You can customize the ribbon to provide additional features to authors or to restrict them from using some features.

- **Add buttons to provide new functionality** You can add new features to SharePoint Server 2010 and provide buttons on the ribbon to give authors access to the features. For example, if your publishing site is used to create highly technical content, you could add an equation editor feature and provide a button for authors to access it.
- **Add inline styles** You can replace the default styles that are available by adding or overriding styles in a style sheet. When you do this, authors can select the styles that are available for the current selection by using the **Styles** command on the contextual menu for the selected element. You can create custom styles for text, markup, images and the Media Web Part player. You can also upgrade styles from Microsoft Office SharePoint Server 2007 to Microsoft SharePoint Server 2010. For information about how to customize inline styles, see [How to: Customize Styles](#).
- **Add table styles** The ribbon includes a set of predefined table styles that can be customized to fit the styling of a single page. Each table style consists of a collection of cascading style sheets classes for each table tag. For example, you can customize the appearance of the first and last rows of a table, the odd and even rows, or the first and last column.

- **Customize image picker locations** In any field that includes a command to insert an image, you can add custom links to the list of default locations listed in the image picker dialog box. For more information, see [How to: Customize the asset picker](#).
- **Restrict access to editing features** As described in [Plan Web pages](#), you can restrict how much freedom authors have to format their Web page content or to add items such as images and hyperlinks to pages in your site by restricting access to editing features. By opening your site in Microsoft SharePoint Designer 2010 you can edit the tags that are associated with field controls to disable the ribbon buttons that authors can use when they edit pages. For example, you can disable the buttons that enable authors to:
 - Set fonts
 - Link to external addresses
 - Add headings to content
 - Make text bold, italic, or underlined
 - Add tables

For information about how to add, replace, and remove controls, groups and tabs on the ribbon, see [Customizing the Server Ribbon](#).

In addition to implementing a command as a menu command, you can also implement a command as a button on the Quick Access Toolbar. The Quick Access Toolbar makes frequently used commands available. To add buttons to the Quick Access Toolbar, you must edit the master page. The following illustration shows the Quick Access Toolbar:



When you plan for Web page authoring, consider whether you want to add or remove commands from the ribbon or the Quick Access Toolbar. Also consider the level of access you want content authors to have to editing features and what kinds of styles that you want to make available. What are the commands, and where should they be added? Make a list of any commands that are needed, the toolbar location where they should be added, whether a button is needed on the Quick Access Toolbar, and whether any additional locations have to be added to the image picker locations.

Plan managed metadata

As you plan your Web site, you must consider how managed metadata can help organize and display content pages and other data. Having a thorough and meaningful taxonomy for content authors to use is essential to building a successful site that requires minimal maintenance. Having the right set of terms for your site enables you to create rules which help organize pages into folders. Good content

organization simplifies the search for information and increases the query speed. For more information about managed metadata, see [Managed metadata overview \(SharePoint Server 2010\)](#).

When you create page layouts that authors will use to create new Web pages, you can add field controls such as text boxes and drop-down lists that contain a predetermined value or that restrict the kind of information authors are allowed to put on a page. You can also use managed metadata to add contextual information to a page, which enables you to do the following:

- Create a custom query that is made for the page.
- Display the appropriate navigation.
- Determine what related data fits best, and display it on the page.

For example, on a travel Web site, you could create a page layout for specific travel destinations that contains a managed keywords field for recreational activities and a Content Query Web Part. A page author who creates a page for a specific travel destination can select from a specified list of recreational activities for that destination. When a page reader views the page, the Content Query Web Part can display a list of other travel destinations that also contain those tags.

As you plan for Web page authoring, consider whether you want to add managed metadata to your page layouts for page authors to use. How will the metadata be used? What terms and term sets are needed? Who will own the term sets, and how will they be managed? For more information about how to plan managed metadata, see [Plan terms and term sets \(SharePoint Server 2010\)](#) and [Plan managed metadata \(SharePoint Server 2010\)](#).

Plan reusable content

The top-level site in a publishing site collection includes a Reusable Content list that is available to every site below it in the site hierarchy in which the SharePoint Server Publishing feature is activated. Reusable content items can be implemented as HTML or as text. By using the **Reusable Content** command on the **Insert** tab under **Editing Tools** on the ribbon, authors can select from a predefined list of content, or they can view a list of all available content and then insert it. For example, if your organization requires that specific marketing text be used when describing a particular product, you can create an item that contains the required description. When a user adds that reusable content item to a page, the text automatically is added to the page.

When you create a reusable content item in the Reusable Content list, you can specify that it be shown in the drop-down menu during page editing. You can also specify whether it can be automatically updated.

- **You specify that an item is automatically updated.** Authors cannot change the item after they insert it on a page. For example, you can implement a copyright statement or a company's name, address, and other contact information as an item that can be automatically updated. Doing this helps prevent authors from incorrectly using those items, and it ensures consistency across all Web pages where the items are used.

When an author inserts an automatically updated item on a page, the URL of the Reusable Content list item is inserted instead of the contents of the item. When a Web browser loads a page that

contains an automatically updated item, the Web browser replaces the URL with the contents of the item. Therefore, changes to automatically updated items in the Reusable Content list do not have to be propagated to pages that use them. They are immediately available the next time that a page is opened in a Web browser.

- **You do not specify that an item is automatically updated.** Authors can change the item after they insert it on a page. This is useful if you want to define the correct form for a block of content, but you want authors to provide the content itself. For example, in a site that provides product descriptions, in which you want each description to follow a particular tabular format, you could create a generic product description table item in the Reusable Content list, which authors could insert and then overwrite.

Plan dictionary customizations

The **Format Text** tab under **Editing Tools** includes a **Spelling** command that checks the spelling of content in all fields on a page that contains HTML content. The **Spelling** command indicates spelling errors and provides commands for fixing or ignoring them.

You can add a custom dictionary to your publishing site to prevent words that are unique to your content from being reported as spelling errors. For example, if your site includes unique product names, you can add them to the custom dictionary. Make a list of all product names, frequently used acronyms, and other words that you want to be included in a custom dictionary for your site.

Plan additional resources

When you create a publishing site, SharePoint Server 2010 creates the libraries that are listed in the following table. You can use these libraries to store additional resources that content creators can use.

Use this location	To store these items	That apply to this level in the site hierarchy
Master Page Gallery	Master pages and page layouts	Site collection
Documents	Documents used in page authoring	Current site
Site Collection Documents	Documents used in page authoring	Site collection
Images	Images used in page authoring	Current site
Site Collection Images	Images used in page authoring	Site collection
Style Library	Custom cascading style sheets and Extensible Stylesheet Language (XSL) styles	Site collection

When users insert an image or link into a page, the Select an Asset window enables them to browse the contents of the current site's lists and libraries, together with the Site Collection Documents library and the Site Collection Images library. You can also use the Suggested Content Browser Locations list to add links to other SharePoint Server 2010 libraries that contain resources to be included on Web pages. When a user inserts an image or link into a Web Part, the links are displayed in the **Suggested locations** menu of the Select an Asset window.

When you plan for Web page authoring, consider the kinds of additional resources that page authors might need. Think about who will create those resources, and where you want them to be stored. If some resources are located in other sites, make a list of what those resources are and where they are located so they can be added to the Suggested Content Browser Locations list.

Web page authoring planning worksheet

Download an Excel version of the [Web Page authoring planning worksheet](#). Use this worksheet to record your Web page authoring decisions for a type of content.

See Also

[Plan Web pages](#)

[Plan content approval and scheduling](#)

Plan content approval and scheduling

Content approval is the process by which authored content is approved or rejected for publication. *Content scheduling* is the process by which content is published and made available to readers according to a specified schedule. The Publishing feature in Microsoft SharePoint Server 2010 provides the ability to approve and schedule content for publishing.

This article contains general guidance about how to plan content approval and content scheduling for use with SharePoint Server 2010 publishing sites. However, this article does not describe how to configure settings for content approval or how to configure workflows.

In this article:

- [About planning content approval and content scheduling](#)
- [Plan content approval](#)
- [Plan content scheduling](#)
- [Using content deployment with content approval and scheduling](#)

About planning content approval and content scheduling

As you plan SharePoint Server 2010 publishing sites, you should plan how much control you want users to have over approving site content. For example, you might want to impose restrictions on how much control authors have over approving content they create. You have the option of giving users no control, simple moderation, or the ability to start a workflow after they submit content. When you plan publishing sites, you should also understand how the content scheduling process works.

Plan content approval

Content approval is the process by which users who have Approver permissions control the publication of content. Content approval is configured by using the **Content Approval** option on the Versioning Settings page in the library settings for the document library that contains content to be published.

When you plan for content approval, you must decide how you want content approval to work for your site and who can approve content for publishing. In SharePoint Server 2010, the control of content can fall within the following levels:

- **None** If content approval is not required for items in a document library, after an author submits content for publishing, it goes live immediately.
- **Simple moderation** Content must be manually approved by a member of the Approver group after an author submits it for publishing. The content is not visible to users who have Read permissions until it is approved.

- **Approval workflow** A workflow is used to run the approval process. Using a workflow makes the approval process more automated and takes advantage of the built-in workflow features, such as automatically sending e-mail to approvers, adding approval tasks to approvers' task lists, and letting authors track the status of the approval process. Users can also modify the Approval workflow template, or develop their own custom approval workflow by using Microsoft SharePoint Designer 2010 or Microsoft Visual Studio 2010.

By default, the publishing site templates are preconfigured to use one of the following categories. You can think of these categories as providing a range of restrictions over the approval of content, from least to most restrictive. The following table shows each category, the level of restriction, and the publishing site template that is automatically associated with each.

Category	Restriction level	Site template
None	None	Enterprise Wiki
Simple moderation	Low	Publishing Site
Approval workflow	Heavy	Publishing Site with Workflow

You can enable or disable publishing-related options for your site, such as requiring content approval or changing Approval workflow settings.

Plan content scheduling

Content scheduling is the process by which users who have at least Contributor permissions specify a schedule to publish content. If the **Content Approval** option is enabled for a document library, content must be approved before it is published. For more information about content approval, see [Versioning, content approval, and check-out planning \(SharePoint Server 2010\)](#).



Note:

Content scheduling is available only if the **Content Approval** option is enabled and if the **Document Version History** option is set to create major and minor (draft) versions.

Content can be scheduled to be published or unpublished at specified dates and times. The scheduled dates and times are initiated by timer jobs that continually check for pages and items in the document library or image library that are ready for publishing or unpublishing. You can change the frequency with which each job runs on the Job Definitions page on the Central Administration Web site.

Using content deployment with content approval and scheduling

Content deployment is a feature of SharePoint Server 2010 that you can use to copy content from a source site collection to a destination site collection. The content deployment feature is designed for

sites that use a multiple farm topology. A multiple farm topology consists of separate authoring, publishing, and possibly staging farms. If you are implementing a multiple farm topology, you must apply all the considerations that are outlined in this article for each authoring farm in your environment. For more information, see [Design content deployment topology](#) and [Technical diagrams \(SharePoint Server 2010\)](#).

If content deployment is used together with content approval and content scheduling for your SharePoint Server 2010 solution, all approval processes occur on the source server where the content is authored. When content is deployed to the target server, the publishing schedule that is associated with each piece of content is also deployed. For example, if a page is approved on the source server on Monday and is set to go live at midnight on Friday, the page is copied to the destination server the next time that a content deployment job runs. However, the page is not visible to users who have Reader or anonymous permissions until midnight on Friday.

See Also

[Plan Web content management \(SharePoint Server 2010\)](#)

[Plan Web pages](#)

Plan for caching and performance (SharePoint Server 2010)

Microsoft SharePoint Server 2010 provides a disk-based binary large object (BLOB) cache that reduces database load and increases browser performance for users. This article describes the BLOB cache, tells you how and when to use it, and lists key considerations for planning to use it. This article also contains information about when to use Bit Rate Throttling, an Internet Information Services (IIS) 7.0 extension that improves video performance for users when serving videos as part of managing digital assets in SharePoint Server 2010. Finally, this article also describes the limitations of upload file size restrictions, and lists considerations for adjusting the size limit for file transfers on the server.

For information about how to enable the BLOB cache, see [Configure cache settings for a Web application \(SharePoint Server 2010\)](#). For information about managing digital assets, see [Plan digital asset libraries \(SharePoint Server 2010\)](#).

In this article:

- [Disk-based BLOB caching](#)
 - [BLOB cache overview](#)
 - [Decide whether to use the BLOB cache](#)
 - [Store the BLOB cache](#)
 - [Enable the BLOB cache](#)
 - [Specify the size of the BLOB cache](#)
- [Bit Rate Throttling](#)
 - [Bit Rate Throttling overview](#)
 - [Decide to use Bit Rate Throttling](#)
 - [Enable Bit Rate Throttling](#)
- [Maximum upload file size](#)
 - [Maximum upload file size overview](#)
 - [Decide maximum upload file size](#)
 - [Configure the maximum upload file size](#)

Disk-based BLOB caching

This section describes the disk-based BLOB cache, and provides important information about how to plan to use the cache with a SharePoint deployment. It tells how to decide when to use the BLOB cache, where to store it, how to enable it, and how to configure the size of the cache in order to get the best performance for users.

BLOB cache overview

The disk-based BLOB cache controls the caching for binary large objects (BLOBs), such as frequently used image, audio, and video files, and other files that are used to display Web pages, such as .css and .js files. The BLOB cache is enabled on a front-end Web server and improves performance by retrieving BLOB files from the database and storing them in a directory on the front-end Web end server where they are served to users. This reduces the network traffic to and load on the database server.

The BLOB cache also provides features that support serving media files to users. One such feature is support for byte-range requests, which lets users select a later point in the video and immediately begin playback. Another feature is progressive caching, which starts serving the beginning of a large video file while the rest of the file is being cached. Video files are divided and retrieved in smaller sections to reduce the load between the front-end and back-end servers. An administrator can configure the size of the sections.

Decide whether to use the BLOB cache

When enabled, the BLOB cache caches various image, audio, and video files, together with .css and .js files. An administrator can modify the settings to add or remove file name extensions or file types to be cached. This functionality lets you either cache as many file types as possible, or to restrict the cache to certain kinds of files. For example, if you have an Internet-facing portal with read-only files such as .doc or .pdf files, you can specify that those files be cached so that they are displayed more quickly to users. If you have a collaboration site that contains files that are frequently updated, as well as media assets, you can specify that the cache is to store only audio or video types by including only file name extensions for those files in the cache settings.

Before you enable the BLOB cache, carefully consider the scenario in which you plan to use it. If your site will be used for heavy collaboration, enabling the BLOB cache might temporarily affect the performance of your site while the files to be cached are first written to the disk. After the files have been stored in the cache, site performance will improve, so take this into consideration when you decide whether or not to enable the cache. Base your decision to enable BLOB caching on the following criteria:

- For a publishing site for which most of the visitors are anonymous or where most of the files are static content, enable the BLOB cache for as many file types as possible.
- For other sites that contain lots of media assets that are read-only, or where only a small percentage of the media assets are updated, enable the BLOB cache for media files only.

There is one BLOB cache per Web application. If you plan to use the BLOB cache together with an asset library that you expect will be large, or together with a site that will receive lots of traffic, consider putting the site collection that contains the asset library into its own Web application so that it receives its own BLOB cache. This will ensure that other assets are not using up space in the BLOB cache that you want allocated to items in the asset library. It will also ensure that sites which receive lots of traffic do not prevent other sites which receive less traffic from benefitting from the BLOB cache.

Store the BLOB cache

When you enable the BLOB cache, you must specify a location on the front-end Web server where the files will be stored. By default, the cache will be created on the drive on which SharePoint is installed. Make sure that you put the BLOB cache on a drive that has sufficient disk space available in which to store the cache. Also, select a drive that will be used by as few processes as possible so that the BLOB cache process does not encounter conflicts when it tries to access the drive. If too many processes compete for disk access on the drive where the BLOB cache is located, BLOB cache performance and other processes will be adversely affected.

If you plan to use the BLOB cache in a scenario with heavy cache use, such as serving videos in a high traffic environment, and if you will use ULS logging, consider placing the BLOB cache on a separate physical drive from the ULS log — not on a separate partition. Storing the BLOB cache and the ULS log on the same drive can result in poor server performance. If you place the BLOB cache and the ULS log on the same physical drive, make sure that you closely monitor the disk queue length for any performance effect.

Each front-end Web server has its own local copy of the BLOB cache that is built as requests for files are received. If you use load balancing with multiple front-end Web servers, each server contains its own cache. When a file is requested by the first server, it is cached to that server only. If the next request for the same file comes from a second server, a second request is sent to the database server to retrieve the file to the cache on the second server.

Enable the BLOB cache

The BLOB cache is configured in the web.config file for each Web application and, by default, is not enabled. You must specifically enable the BLOB cache in order to get the performance advantage it provides. For information about how to enable the BLOB cache, see [Configure cache settings for a Web application \(SharePoint Server 2010\)](#).

Specify the size of the BLOB cache

When you decide how large to make the BLOB cache, you must consider the number and size of the files to determine the total size of the data to be stored in the cache. By default, the BLOB cache is set to 10 gigabytes (GB). Allow at least 20 percent more space on the drive than the size of the cache. For example, if you have 10 GB of content, set the size of the cache to 12 GB on a drive that has at least 15 GB of space. If the BLOB cache is too small, serving files to users slows, reducing the performance of your site.

Bit Rate Throttling

This section contains information about Bit Rate Throttling, describes when you should use it with the SharePoint solution, and explains how to enable it.

Bit Rate Throttling overview

Bit Rate Throttling is an IIS 7.0 extension that meters the download speeds of media file types and data between a server and a client computer. The encoded bit rates of media file types such as Windows Media Video (WMV), MPEG-4 (MP4), and Adobe Flash Video, are automatically detected, and the rate at which those files are delivered to the client over HTTP are controlled according to the Bit Rate Throttling configuration. For more information, see [Bit Rate Throttling](http://go.microsoft.com/fwlink/?LinkId=155151) (<http://go.microsoft.com/fwlink/?LinkId=155151>).

Decide to use Bit Rate Throttling

If you will make long-playing video assets available to users in SharePoint Server 2010, enable Bit Rate Throttling in IIS. Without Bit Rate Throttling, IIS will serve video files by using as much bandwidth as it can, which will result in increased network performance. When you enable Bit Rate Throttling in IIS, it will serve video files that use only as much bandwidth as is needed to support progressive downloading and viewing of videos. When the BLOB cache is also enabled, Bit Rate Throttling uses extension rules for files cached to disk. Files that are served from the BLOB cache by using Bit Rate Throttling are sent to the client based on a percentage of the compressed size using the encoded bit rate. For example, if the videos in your organization are smaller than 10 MB, you may decide not to use Bit Rate Throttling because it will affect how fast users can download videos to their local computers. However, if you are serving video files, enable Bit Rate Throttling to control the speed at which files are downloaded to client computers.



Note:

Bit rate throttling will not work correctly if you do not first enable the BLOB cache and configure it to cache the file types that you want to throttle.

Enable Bit Rate Throttling

In order to enable Bit Rate Throttling in IIS 7.0, you must install IIS Media Services 2.0. For information about how to install IIS Media Services 2.0, see [Bit Rate Throttling Readme](http://go.microsoft.com/fwlink/?LinkId=154962) (<http://go.microsoft.com/fwlink/?LinkId=154962>). For information about how to configure Bit Rate Throttling, see [Bit Rate Throttling Configuration Walkthrough](http://go.microsoft.com/fwlink/?LinkId=155153) (<http://go.microsoft.com/fwlink/?LinkId=155153>).

Maximum upload file size

This section describes the upload file size limitation, tells how to decide what the maximum upload file size limit should be, and how to configure it.

Maximum upload file size overview

The maximum upload file size is a setting that is used by the SharePoint Server 2010 Web application that specifies the maximum size of a file that a user can upload to the server. When a new Web

application is created, SharePoint Server 2010 sets the default maximum upload size to 50 MB. If a user tries to upload a file larger than the specified maximum upload size, the upload will fail.

Decide maximum upload file size

Every user that uploads a file to a library uses a connection to the server and increases the amount of data in the database. This impacts the load, response time and data capacity for a server. Depending on your scenario, this can negatively impact your server performance if the server is not configured to handle larger volumes of files. To determine what the upload file size limit should be for your server, consider the number of users for your site, and the size of the files they will upload. For example, if your users will primarily be uploading video files that are 500 MB, the upload file size limit should be large enough to easily accommodate the largest files users will upload. When planning to adjust the upload file size limit, keep in mind that this will also directly impact capacity planning for your server environment. For more information about planning for storage of large media files, see [Plan digital asset libraries \(SharePoint Server 2010\)](#).

Configure the maximum upload file size

In order to configure the upload file size in SharePoint Server 2010, a farm administrator must change the **Maximum Upload Size** value on the Web Application General Settings page in Central Administration.



Note:

If you increase the default maximum upload size for a Web application, and you also plan to use content deployment to move content from site collections within that Web application to another farm or site collection, you must also increase the default maximum upload size on the destination server, or the content deployment job will fail.

See Also

[Cache settings operations \(SharePoint Server 2010\)](#)

Plan for large Pages libraries (SharePoint Server 2010)

A Pages library is a document library that contains all the content pages for a publishing site. A site that has thousands or tens of thousands of pages stored in the Pages library must consider a unique set of issues that relate to managing these pages, and providing navigation between them in a site.

This article describes the use of large Pages libraries in Microsoft SharePoint Server 2010 publishing sites, and it provides information to help you determine whether to use large Pages libraries with your publishing solution, and how to plan for them. This article does not describe how to set up rules or page routing to use with large Pages libraries, and it does not discuss how to configure navigation for use with large Pages libraries. For information about how to plan sites, see [Plan sites and site collections \(SharePoint Server 2010\)](#).

In this article:

- [About large Pages libraries](#)
- [Determine whether to use a large Pages library](#)
- [Decide how to manage pages](#)
- [Plan for navigation](#)
 - [Planning the Global Navigation and the Current Navigation menus](#)
 - [Planning other Web parts for navigation](#)

About large Pages libraries

Pages libraries in SharePoint Server 2010 now support creating folders and storing pages within folders, so there could potentially be thousands to tens of thousands of pages that are stored in the Pages library for a single site. The Global Navigation and Current Navigation menus for a publishing site are directly tied to the Pages library. By default, new pages are put in the root of the Pages library as they are created. If the site has been configured to use auto-navigation, new pages will automatically be added to the Global Navigation and Current Navigation menus. However, pages that are put in a folder in the Pages library are not added to the navigation menus, and they must be added manually. Additionally, there is a limit to the number of links that can be displayed in either the Global Navigation or Current Navigation. If your solution will be using a single site that has lots of pages, you must plan for how to organize your content so that you can manage those pages and configure navigation within the site.

SharePoint Server 2010 provides several ways to manage the site content that is automatically stored in a large Pages library. One way is to enable the Content Organizer feature for a site and create rules that route pages to specific folders based on certain criteria, such as content type, title, scheduling dates, or target audience. Another way is to use the folder partitioning setting in the Content Organizer to automatically create folders after the target location contains a specified number of items. After the

target location has reached the maximum number of items, a new folder that has a specified folder name is automatically created, and all new items created will then be put into the new folder.

Although you can manage the organization of your site content manually, using a large Pages library together with the Content Organizer has the following benefits:

- **Automated page organization** The organization of pages can be managed automatically by using the Content Organizer, which allows for folder partitioning and page routing.
- **Lower site maintenance** Site owners spend less time managing pages in the site because the library can be managed automatically. Authors do not have to worry about putting pages in the correct location because the rules-based routing does it for them.
- **Improved query performance** The query load on the content database is reduced when pages are displayed to users because Content Query Web Parts query only a single library in which content is stored.

Determine whether to use a large Pages library

Before you plan to use a large Pages library, you must first determine whether a large Pages library is right for your solution. This will depend on how you intend to organize the content in your site. To decide whether using a large Pages library is right for your solution, answer the following questions:

- Will the v4.master page be the same for all content in the site?
- Will page layouts be the same for all content in the site?
- Will content types be the same for all pages in the site?
- Will permissions for users who have contributor, designer and approval access be the same for all content in the site?

If the answer to each of these questions is “Yes,” your solution might benefit from using a single site that has a large Pages library. If you answered “No” to any of these questions, you should use separate sites that have their own Pages libraries.

Decide how to manage pages

After you have decided to use a large Pages library, you must decide how to manage the pages that will be created. There are two ways in which you can manage pages for your site: manually or by using rules and page routing. We do not recommend managing pages manually because a large number of pages is involved. Instead, you should use the rules and page routing that are provided as part of the Content Organizer feature.

Before you can use rules and page routing for a site, you must start the Content Organizer feature by using the Manage site features page in Site Settings. After you start the Content Organizer feature, if you want to enable the automatic creation of folders in the Pages library, use the Content Organizer Settings page to turn on folder partitioning. Use the Content Organizer Rules page to create rules to route pages to the correct location in the Pages library.

Although rules can be set up for various criteria, you can use managed metadata to provide even more control over where pages are put in the library. For example, you can create term sets and route pages to certain folders based on the terms or managed keywords that authors assign to pages they create. For information about how to use managed metadata, see [Plan managed metadata \(SharePoint Server 2010\)](#).

When you plan to manage the content in the Pages library, think about the pages that authors will create. Will the content be similar enough that you can use automatic folder partitioning? Do you need to design a more structured library to contain the pages for your site? What folders do you need, and what criteria do you want to use to route pages to specific folders? Will you need to create a custom term store to provide authors a list of keywords to use with page routing?

Plan for navigation

The Global Navigation and Current Navigation menus do not display pages in folders, and the menus have limits on the maximum number of links that can be displayed, so you must plan for how users will navigate among the pages of your site. In general, planning navigation for a site that uses a large Pages library involves the following site elements:

Global Navigation and Current Navigation menus

Other Web parts for navigation

Planning the Global Navigation and the Current Navigation menus

Although pages that are added to the root of the Pages library are added to the Global Navigation and Current Navigation menus automatically, if your site will have lots of pages, you must decide which pages to display in the Global Navigation and Current Navigation menus. For example, you can create a series of pages that use the Welcome Page template to display a mix of authored content and Web Parts that link to other pages in the site and then only include the Welcome Pages in the Global Navigation and Current Navigation menus.

Use the Site Navigation Settings page in Site Settings to customize the Global Navigation and Current Navigation menus for your site. You can stop the navigation menus from automatically displaying links to sites below the top level site and pages. You can also specify only the links that you want to show to users and the order in which you want the links listed. This makes it possible for you to build a navigation system that is not dependent on the structure of the Pages library. If you do not want to manually update the navigation menus in the user interface, you can also use Microsoft Visual Studio 2010 to build a custom navigation menu for your site.

Planning other Web parts for navigation

SharePoint Server 2010 provides two navigation-specific Web Parts that can be added to Web Parts pages for publishing sites: the Table of Contents Web Part and the Summary Links Web Part.

The Table of Contents Web Part automatically displays site content for the first three levels of a site. However, the Table of Contents Web Part should not be used for publishing sites that have large Pages

libraries because it does not display pages in folders, and therefore will not accurately display the content hierarchy for the site. This Web Part is better suited for smaller publishing sites that have only a limited number of pages in their site.

The Summary Links Web Part makes it possible for page authors to create a list of links that can be grouped and styled on a Web Parts page. Although this provides an easy way for page authors to link to other pages, the limitation is that the list is static, and must manually be changed to add or remove items from the navigation. This Web Part is best used for targeting a short list of specific pages in a site, but scaling up to a longer list of Pages library links with many folders and pages could quickly become unmanageable.

You can also use a Lists and Libraries Web Part or a Content Query Web Part to create dynamic, custom navigation links on pages in your site. By using either Web Part, you can help reduce the cost of site maintenance and provide page authors the flexibility of providing dynamic content that makes it easy for users to locate new or popular content without having to manually update the navigation.

You can use a Lists and Libraries Web Part to display a view of any list or library in the site, such as the Pages library. You must first create a view that is configured to filter, sort, and group the content of the Pages library to return the items that you want to display. Then you must select that view in the Web Part on another page to display the library items. The result is a view into the Pages library that is dynamic and that will change when more pages are added to the library.

You can also use the Content Query Web Part to create a custom list of links to content from any list or library in the site, or from any other site in the site collection. By using the Content Query Web Part, you can specify the criteria that are used to display items in the Web Part, such as content type, title, scheduling dates, or target audience. For example, if your site uses page rating, you can create a Content Query Web Part that displays the top rated pages for your site. When the Content Query Web Part is used with a large Pages library, it provides more flexibility than the Summary Links Web Part, because the list is dynamic and reduces the amount of maintenance that is required to update static lists when pages are added or removed.

When you plan the navigation for your site, think about how users will navigate within the site. What are the key pages that must be displayed in the Global Navigation and Custom Navigation menus? What kinds of content does the site contain, and how should the content be grouped when it is displayed to users? Do you need lists of static or dynamic links to content, or a mix of both? When you plan the navigation for a site that uses a large Pages library, you must consider many of the same issues that you would consider when planning the navigation for any other site. For more information about how to plan site navigation, see [Plan site navigation \(SharePoint Server 2010\)](#).

See Also

[Sites and site collections overview \(SharePoint Server 2010\)](#)

[Plan sites and site collections \(SharePoint Server 2010\)](#)

[Plan managed metadata \(SharePoint Server 2010\)](#)

[Site navigation overview \(SharePoint Server 2010\)](#)

Content deployment overview (SharePoint Server 2010)

Content deployment is a feature of Microsoft SharePoint Server 2010 that you can use to deploy content from a source site collection to a destination site collection. This article summarizes the content deployment feature in SharePoint Server 2010. It describes the purpose and function of content deployment, explains content deployment paths and jobs, and explains the security options that are available when you deploy content. This article also explains how the content deployment process works, and it lists important factors and limitations of using content deployment. This article does not describe the steps that are involved in planning to use content deployment or how to set up and configure content deployment. For more information, see [Plan content deployment \(SharePoint Server 2010\)](#).

In this article:

- [What is content deployment?](#)
- [About deployment paths and jobs](#)
- [About content deployment security](#)
- [How content deployment works](#)
- [Important considerations in content deployment](#)

What is content deployment?

Content deployment deploys content from a source SharePoint Server 2010 site collection to a destination site collection. The complete source site collection can be deployed, or a subset of sites can be deployed. Content deployment, which is incremental by default, deploys only changed pages and related assets (such as images). A Quick Deploy feature supports the deployment of a single page by authors.



Note:

For the content deployment Quick Deploy feature to work, the source site collection must have been created by using the Publishing Portal template, or it must have the SharePoint Server Publishing Infrastructure feature enabled.

In most content deployment scenarios, the source site collection, from which content is being deployed, is in a server farm that is separate from the destination site collection. Typically, the destination server farm (the "production" farm) has tightened security to minimize the actions that can be done in the production environment. It is not expected that authoring will be done on the production server, because changes to content on the production server might be overwritten by a content deployment job. In most content deployment scenarios, the source server farm and the production server farm are

in independent Active Directory domains. For information about content deployment topologies, see [Design content deployment topology](#)

It is important to be aware that content deployment is a one-way process: content is deployed from a source site collection to a destination site collection. The content deployment feature does not support round-trip synchronization from source to destination and back again. Creating new content or changing existing content on the destination site collection can cause content deployment jobs to fail. Because of this, you should consider restricting permissions on the destination site collection so that users cannot make changes directly to content that is stored within that site collection.

In content deployment, the base URL of the source site collection can differ from the base URL of the destination site collection. The content deployment feature will fix links in the source content to work correctly in the destination location.

Content deployment deploys only content — Web pages, libraries, lists, and resources that are used by the deployed pages. It does not deploy programs, assemblies, features, or configuration information such as Web.config files. When a Web page is deployed, any items in the content database that the page depends on — such as images, style sheets, or layout pages — will also be deployed.

Content deployment deploys the most recent major and minor versions of a content item. For example, if version 2.7 of a Web page is being deployed, the most recent major version (2.0) of the page, and the most recent minor version (2.7), will be deployed to the destination site.

If an item has an associated publishing schedule, the scheduling information is deployed together with the item so that the schedule is followed in the destination site collection. For example, if an item that is scheduled to be published at 6:00 A.M. is deployed at 3:00 A.M., site users on the destination site cannot view the content until 6:00 A.M. For information about scheduling content, see [Plan content approval and scheduling](#).

A new feature of content deployment that was added for SharePoint Server 2010 is the option to use SQL Server database snapshots during export. If the database snapshots option is enabled, a snapshot of the source content database is created before the export phase of the content deployment job starts. The content deployment job then uses the database snapshot to perform the export, instead of exporting directly from the live content database. After the export has successfully completed, the snapshot is deleted. By using the database snapshot option, you eliminate any potential problems with users editing content in the content database while a content deployment job is running.



Note:

The SQL Server database snapshot option is only available if Microsoft SQL Server 2008 Enterprise edition is installed. If you are using Remote BLOB Storage (RBS), and the RBS provider that you are using does not support snapshots, you cannot use snapshots for content deployment or backup. For example, the SQL FILESTREAM provider does not support snapshots. For more information about RBS, see [Overview of Remote BLOB Storage \(SharePoint Server 2010\)](#).

About deployment paths and jobs

The following section describes content deployment paths and jobs.

Content deployment paths

A *content deployment path* defines a source site collection from which content deployment can originate and a destination site collection to which content is deployed. A path can be associated with only one site collection. A content deployment path specifies the following information:

- Authentication information that gives content deployment jobs permission to the destination site collection. To deploy content to the destination site collection, deployment jobs must have Central Administration credentials on the destination server. Jobs can connect by using Integrated Windows authentication or Basic authentication.
- Information about whether to deploy user names that are associated with the content, such as authors' names.
- Information about how to deploy permissions on the content. For more information, see [About content deployment security](#).

Content deployment jobs

A *content deployment job* deploys specified content on a specified schedule by using a specified path. After a path is defined, one or more content deployment jobs can be defined. A deployment job specifies:


- The path with which the job is associated.
- Whether the job uses SQL snapshots.
- The sites within the source site collection to deploy.
- The frequency at which to run the job and deploy the content.
- Whether to send e-mail when a job succeeds or fails and the e-mail addresses to use.

There are two kinds of standard content deployment jobs: full and incremental. These jobs are managed by a server farm administrator, and they enable you to specify whether to deploy all content, including any content that might have been deployed previously, or only content that was added, updated, or deleted since the last successful deployment. These jobs are run on a schedule that the server farm administrator specifies.

A third kind of content deployment job, Quick Deploy, is a special job that enables users to quickly publish content without waiting for the next standard content deployment job to run. This job runs automatically, at a specified interval.

The following table describes the kinds of content deployment jobs:

Job Type	Description
Incremental	<p>An incremental deployment job deploys all new, changed, or deleted content from the source to the destination. The first time that an incremental deployment job runs, it performs a full deployment. For each subsequent run of an incremental deployment job, new content is added to the destination, whereas updated content replaces content that has the same GUID but has older modification dates. Content that is deleted on the source is flagged so that it will also be deleted from the destination server. This is an important difference between full and incremental deployments.</p>
Full	<p>A full content deployment job deploys all content from the source to the destination, regardless of whether that content was previously deployed. Also, full deployment jobs do not check whether content that exists on the destination was deleted from the source. If you delete content on the source server and then perform a full deployment, that content will not be removed on the destination server. You should avoid using full deployment jobs except in specific cases where you know content has not been deleted on the source server.</p>
Quick Deploy	<p>A Quick Deploy job enables users, such as authors and editors, to quickly deploy a Web page. By default, a Quick Deploy job is created automatically when a new content deployment path is created, and it is set to run automatically every 15 minutes. When a user flags a page for inclusion in a Quick Deploy job, that page will be included in the next automatically scheduled Quick Deploy job. Only pages that are flagged by a user as Quick Deploy pages are included in the job. Alternatively, a farm administrator can manually run or cancel a Quick Deploy job at any time by using the Manage Content Deployment Paths and Jobs page. Any member of the Quick Deploy</p>

Job Type	Description
	<p>users group (which is created in sites that have the SharePoint Server Publishing Infrastructure feature enabled) can mark a Web page for deployment by using the Quick Deploy command.</p> <p> Note:</p> <p>It is possible to have a path that is defined in sites that do not have the Office SharePoint Server Publishing Infrastructure feature enabled. However, paths that are created in this manner will not have associated Quick Deploy jobs. If you want to add a Quick Deploy job to a path that was defined in a site that does not have the SharePoint Server Publishing Infrastructure feature enabled, first enable the SharePoint Server Publishing Infrastructure feature on the source site collection, and then edit and save the path again. The path will then have a Quick Deploy job associated with it.</p>

About content deployment security

Permissions to content on the destination server farm will usually differ from permissions to content on the source server farm. In many publishing solutions, the destination server farm authenticates users by using a different Active Directory domain than the one used in an authoring or staging environment, and there might not be a trust relationship between the two domains.

When you configure a content deployment path, you can select from the following security options:

- **All** Deploys all security-related information together with the content. This includes role definitions, access control lists (which map users and roles to the content they have permissions to view or edit), and users. This option is useful if the same set of users has the same permissions on the source and destination server farms. For example, when you deploy from an authoring server farm to a staging server farm, this option might be best because the same users need access to both sets of content. **All** is the default option.
- **Role Definitions Only** Deploys role definitions and access control lists that map the roles to the content but do not deploy users. In this option, the same roles apply in the source and destination server farms, but different users can be assigned to those roles in each server farm.

- **None** Deploys no security information. Security on the destination security farm must be managed by the administrators of that server farm by assigning users and roles to the farm's sites and content. For example, when you deploy from a staging server farm to a corporate Internet presence site, this option helps ensure that the security of the two server farms is managed separately.

For more information about security, see [Security planning for sites and content \(SharePoint Server 2010\)](#).

How content deployment works

Content deployment settings for both incoming and outgoing deployment jobs are configured on the Content Deployment Settings page, which is accessed from the General Application Settings page on the Central Administration Web site. You use the Content Deployment Settings page to accept or reject incoming content deployment jobs for a whole server farm. You can also set specific servers within your server farm to be used for receiving incoming content deployment jobs or for sending outgoing content deployment jobs. This enables you to spread the load for content deployment jobs across multiple servers in your server farm, based on the available server resources and the needs of your server farm.

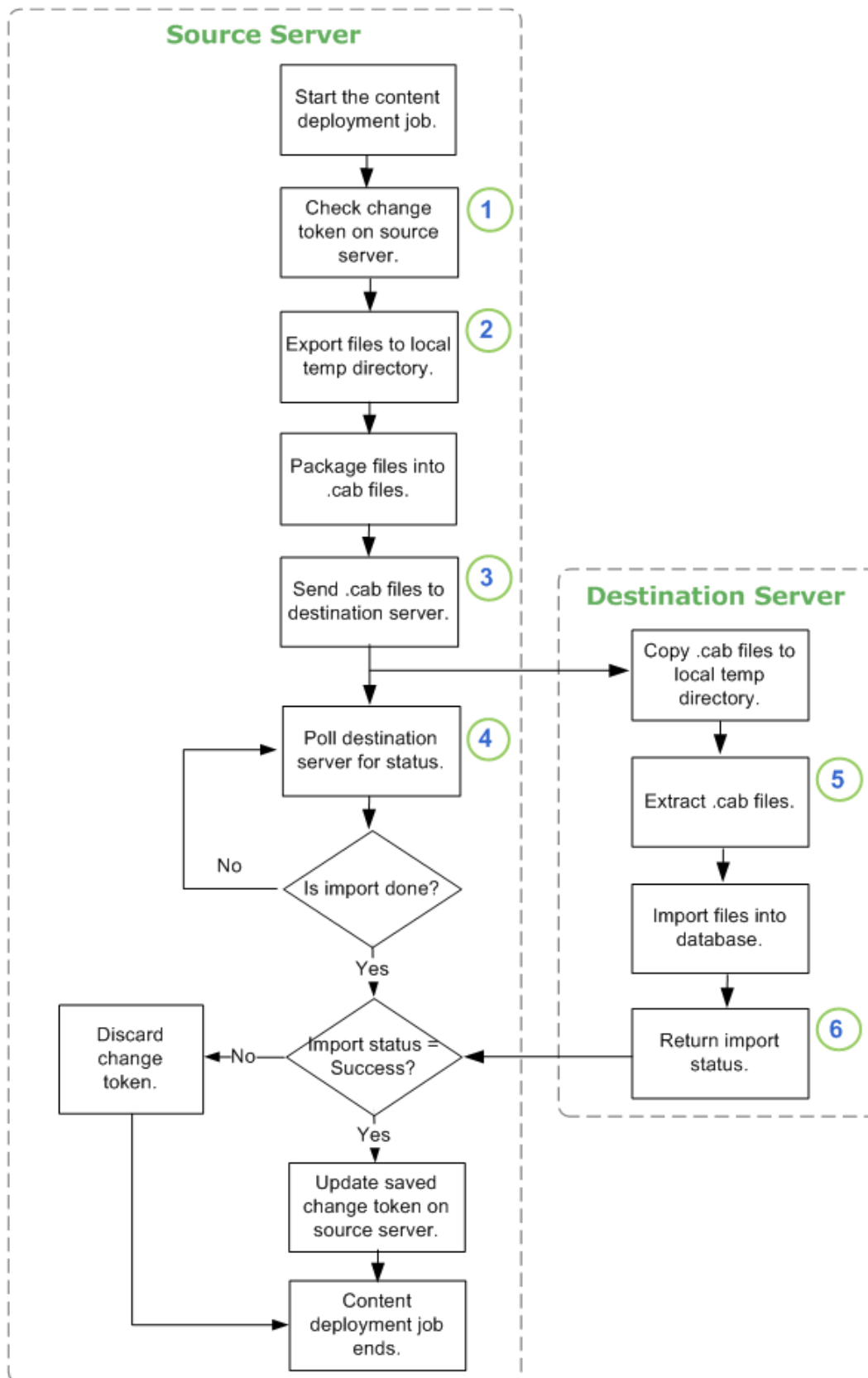



Note:


Depending on the kind of server farm you are using, you might not have to enable support for both incoming and outgoing deployment jobs. If your server farm is an authoring server farm, you do not have to configure incoming (import) settings. If your server farm is a production server farm, you do not have to configure outgoing (export) settings. However, if your server farm is a staging server farm, you must configure both incoming (import) and outgoing (export) settings.

The tasks that are involved in content deployment are controlled by the timer process on the server that hosts the Central Administration Web site, which is used to administer the content deployment jobs. This server could be the source server in the deployment server farm, or it could be a separate server in the farm. The content deployment job uses the service account information that is provided in the content deployment path settings to authenticate with a Web service on the destination server. This Web service acts as the pathway for all communication between the source and destination servers while the content deployment job runs.

The following illustration shows the process that the content deployment job undergoes from start to finish:



Callout	Description
1	<p>When a content deployment job starts, it checks the change token to determine when the last successful content deployment job was run. If the length of time between the last successful content deployment job and the current one is so long that the stored change token is no longer valid, it will run as a full content deployment job, not an incremental content deployment job.</p> <p>After the change token has been verified, the export process is started on the source server. If SQL snapshots are enabled for the content deployment job, a snapshot is taken before the export process starts.</p> <p> Note: In preparation for the export, settings such as the file location, base file name, and other values are specified for the deployment job.</p>
2	<p>Next, the content to be included is exported to a temporary directory on the source server, where it is packaged into .cab files for transport. If the deployment job has been configured to use SQL Server database snapshots, it will use a database snapshot as the source for the export; otherwise, it will export directly from the content database.</p> <p>Alternatively, you can use the Microsoft.SharePoint.Deployment.SPExport namespace from the SharePoint Server 2010 API to export content.</p> <p>After the source server has authenticated with the Web service on the destination server, it calls the Web service to prepare the import on the destination server.</p>
3	<p>After the files have been packaged into .cab files on the source server, the files are transported to a</p>

Callout	Description
	<p>local temporary directory on the destination server via HttpPost.</p> <p>The content deployment job then calls the Web service to start the import process on the destination server.</p> <p> Note:</p> <p>In preparation for the import, settings such as file location, base file name and other values are set by using the information that was stored in the content deployment job when the files were prepared on the source server.</p>
4	<p>While the import is in progress, the content deployment job calls the Web service to get the status of the import process. If the destination server does not respond with updated status within a certain amount of time, the content deployment job will contain a warning message that the job might have timed out. The content deployment job will continue to request updated status from the destination server, but might eventually fail and need to be re-run if the destination server repeatedly fails to respond.</p>
5	<p>During import, the .cab files are extracted to a temporary directory on the destination server, and then they are imported into the database. Any site collection features that are required by items that were included in the import are activated, and scheduling is then configured for the imported items.</p> <p>Alternatively, you can use the Microsoft.SharePoint.Deployment.SPImport namespace from the SharePoint Server 2010 API to import content.</p>
6	<p>After the import has finished, it returns either a Success or Failure status to the Central Administration server. If the import status is Success, the change token is saved. If the import</p>

Callout	Description
	status is Failure, the change token is discarded.

Important considerations in content deployment

The following list contains important considerations to be aware of when you use content deployment:

1. **Always deploy to an empty site collection for the initial content deployment job.** If the site collection already contains content, the initial content deployment job will fail. When you create the site collection on the destination server, use the **< Select template later >** option on the Custom tab of the Create Site Collection page in Central Administration to create an empty site collection. The first time that the content deployment job runs, the correct template and all associated configuration settings will be applied to the destination server.



Note:

Do not use the Blank Site template to create a destination site collection. The Blank Site template does not create an empty site collection and can cause the content deployment job to fail.

2. **The export and import servers must each host an instance of the Central Administration Web site.** When you configure content deployment settings for your server farm, you select the servers in your server farm to designate as export and import servers for content deployment. If you attempt to configure an export or import server that does not host the Central Administration Web site, no error message will be displayed. The content deployment export or import phase will not start. Be sure to deploy the Central Administration Web site on the export and import servers.
3. **Each server in your source and destination server farms must have identical updates.** Be sure that all SharePoint Server 2010 and Windows Server 2008 R2 and Windows Server 2008 with Service Pack 2 (SP2) updates have been applied and that any language packs, if they are needed, have been installed.
4. **The source and destination servers must have enough hard disk space for storing the files that are used during export and import.** During export, all files to be included in the content deployment job are stored in a temporary directory in the export server farm. Likewise, during import, files to be imported into the database are stored in a temporary directory on the destination server farm. Be sure that the location of the temporary directory for each server farm has sufficient disk space to accommodate the files that are included in the deployment job.
5. **If jobs will run infrequently, the time for keeping changes in the change log must be adjusted.** By default, the change log is configured to keep a record of any changes for 60 days. If the time between two incremental deployment jobs exceeds this time—for example, if it was 70 days since the last content deployment job was run—then the change log will not contain entries from before the last change token. If the time between jobs will be more than 60 days, you must

change the number of days specified for the Web application in the Central Administration Web site.

6. **Do not run content deployment jobs in parallel if the same path is used by both jobs.** Changes made by one job might conflict with changes made by another job that is running along the same path at the same time. If this happens, the content deployment job might fail.

See Also

[Plan content deployment \(SharePoint Server 2010\)](#)

[Design content deployment topology](#)

Plan content deployment (SharePoint Server 2010)

Content deployment is a feature of Microsoft SharePoint Server 2010 that you can use to copy content from a source site collection to a destination site collection. This article contains general guidance about how to plan to use content deployment with your SharePoint Server 2010 sites. It does not describe the purpose and function of content deployment, explain content deployment paths and jobs, or explain the security options when you deploy content. This article does not explain how the content deployment process works, nor does it explain how to set up and configure content deployment. For more information, see [Content deployment overview \(SharePoint Server 2010\)](#).

In this article:

- [About planning content deployment](#)
- [Determine whether to use content deployment](#)
- [Determine how many server farms you need](#)
- [Plan the export and import servers](#)
- [Plan content deployment paths](#)
- [Plan job scheduling](#)
- [Plan for large jobs](#)
- [Content deployment planning worksheet](#)

About planning content deployment

The planning process that is described in this article starts with helping you determine whether to use content deployment with your SharePoint Server 2010 solution. The remainder of the article describes the steps that are required to plan a content deployment solution: deciding how many server farms are necessary, planning the export and import servers, planning the content deployment paths and jobs, and special considerations for large jobs. You can record this information in the worksheet that is referenced in the [Content deployment planning worksheet](#) section.

Determine whether to use content deployment

Although content deployment can be useful for copying content from one site collection to another, it is not a requirement for every scenario. The following list contains reasons for why you might want to use content deployment for your solution:

- **The farm topologies are completely different.** A common scenario is one in which there are authors publishing content from an internal server farm to an external server farm. The topologies

of the server farms can be completely different. However, the content of the sites to be published is the same.

- **The servers require specific performance tuning to optimize performance.** If you have a server environment where both authors and readers are viewing content, you can separately configure the object and output caches on the different site collections based on the purpose of the site or user role.
- **There are security concerns about content that is deployed to the destination farm.** If you do not want users to have separate accounts on the production server, and you do not want to publish by using only approval policies, content deployment lets you restrict access to the production server.

Before you implement a content deployment solution, you should carefully consider whether content deployment is really necessary. The following list contains alternatives to using content deployment:

- **Author on production using an extended Web application** If you have a single-farm environment, you can choose to allow users to author content directly on the production farm and use the publishing process to make content available to readers. By using an extended Web application, you have a separate IIS Web site that uses a shared content database to expose the same content to different sets of users. This is typically used for extranet deployments in which different users access content by using different domains. For more information, see [Extend a Web application \(SharePoint Server 2010\)](#).
- **Create a custom solution** You can use the **Microsoft.SharePoint.Deployment.SPExport** and **Microsoft.SharePoint.Deployment.SPImport** namespaces from the SharePoint Server 2010 API to develop a custom solution to meet your needs. For more information, see [How to: Customize Content Deployment for Disconnected Scenarios](#).
- **Use backup and restore** You can use backup and restore to back up a site collection from one location and restore it to another location. For more information, see [Back up a site collection \(SharePoint Server 2010\)](#) and [Restore a site collection \(SharePoint Server 2010\)](#).

If you decide that using content deployment in SharePoint Server 2010 is right for your solution, continue reading this article.

Determine how many server farms you need

A typical content deployment scenario includes two separate server farms: a source server farm that is used for authoring, and a destination server farm that is used for production. You can also use content deployment to copy content between two separate site collections within the same server farm, or you can use a three-tier server farm that contains a server for authoring, one for staging and quality assurance, and one for production. If you will be using content deployment, you should also decide how many server farms are necessary for your solution. For more information about topologies for content deployment, see [Design content deployment topology](#)

Plan the export and import servers

After you have decided on a topology for your server farm, you must decide which servers will be the export and import servers. These are the servers in the server farm that are used to run the content deployment jobs. They do not have to be the same as the source or destination servers. However, the servers that are designated as export and import servers must have the Central Administration Web site installed. Decide which servers will be configured to either send or receive content deployment jobs and to record your decisions.

In the content deployment planning worksheet, record each server farm in your content deployment topology, and note its purpose. For each server farm, provide the URLs of the export server, the import server, or both. Also record the Active Directory domain that is used by the farm.

Plan content deployment paths

A content deployment path defines a source site collection from which content deployment can start and a destination site collection to which content is deployed. A path can only be associated with one site collection. To plan the content deployment paths that are needed for your solution, decide which site collections will be deployed and define the source and destination for each path. For more information about paths, see [Content deployment overview \(SharePoint Server 2010\)](#).

If you will be using a three-stage farm topology, you must also plan for how content will be deployed across the farms. In general, you should reduce the number of “hops” the content makes as it moves from authoring to staging and then to production. For example, if you want to test content on the staging farm before you push it to production, you can deploy content from the authoring farm to the staging farm first, and then deploy content from the authoring farm to the production farm after the content has been verified. This means that only the authoring farm is responsible for deploying content to all other farms in the environment. Although it is possible to deploy content from authoring to staging, and then from staging to production, it is not necessary to use this approach. When you design content deployment paths for a three-stage farm topology, you must also carefully plan the scheduling of the jobs that will deploy the content to the other farms in the environment. For more information about content deployment topologies, see [Design content deployment topology](#).

Record each path in the content deployment planning worksheet. For each path, enter the source and destination Web applications and site collections. Also record how much security information to deploy along the path: All, Roles only, or None.

Plan job scheduling

After you have defined the paths along which site content will be deployed, you must plan the specific jobs to deploy the content. A content deployment job lets you specify that a whole site collection or only specific sites in a site collection will be deployed for a specific path. Jobs also define the frequency with which they are run and whether to include all content, or only new, changed, or deleted content. You can associate multiple jobs with each path. For each path that you have defined, you must decide whether a job will deploy the whole site collection or will deploy specific sites.

As you plan the scope of your content deployment jobs, be sure to think about the order in which the jobs will run. You must deploy a parent site collection or site before you can deploy a site below it in the hierarchy. For example, if you have a site collection with two sites below it, Site A and Site B, and Site A also has two sites below it, Site C and Site D, you must create and run a job that will deploy the top-level site collection, before you can deploy Site A and Site B. You must also deploy Site A before you can deploy Site C and Site D. If you plan to use content deployment jobs that are scoped to specific sites, be sure to schedule the jobs appropriately so that sites higher in the hierarchy are deployed before sites lower in the hierarchy.

You must also decide when and how often to run each job. In general, you should schedule jobs to run during times when the source server has the least amount of activity. Content that is checked out for editing by a user when a content deployment job starts will be ignored by the content deployment job, and it will be copied with the next deployment job after it is checked in. You can configure a job to use a database snapshot of the content database in Microsoft SQL Server 2008 Enterprise Edition to minimize risk to the content deployment job.



Note:

If you are using Remote BLOB Storage (RBS), and the RBS provider that you are using does not support snapshots, you cannot use snapshots for content deployment or backup. For example, the SQL FILESTREAM provider does not support snapshots. For more information about RBS, see [Overview of Remote BLOB Storage \(SharePoint Server 2010\)](#).

If you will be using a three-stage farm topology, you must also plan for when content is deployed across the farms. For example, if you deploy content from the authoring farm to the staging farm to test and verify content, you should plan to schedule the job that deploys content to the production farm so that there is enough time to resolve any issues that are found on the staging farm.



Note:

Do not run content deployment jobs in parallel if the same path is used by both jobs.

For each path, record each associated job in the content deployment planning worksheet. If there is more than one job for a path, insert a row underneath the path for each job to be added. For each job, enter the scope and frequency with which the job will run.

Plan for large jobs

A content deployment job exports all content, as XML and binary files, to the file system on the source server and then packages these files into the default size of 10 MB .cab files. If a single file is larger than 10 MB, such as a 500 MB video file, it will be packaged into its own .cab file, which can be larger than 10 MB. The .cab files are then uploaded by HttpPost to the destination server where they are extracted and imported. If the site collection that will be deployed has a large amount of content, you must make sure that the temporary storage locations for these files on both the source server farms and the destination server farms have sufficient space to store the files. In many cases, you might not know the size or number of .cab files that will be included in the job until you start using content

deployment. But if you know that your site is large and will contain lots of content, make sure that you plan for sufficient storage capacity as part of your content deployment topology.



Note:

If your site will contain large files, such as video files, you might have to adjust the maximum file upload size for the Web application to accommodate the larger .cab file size. For more information, see [Plan for caching and performance \(SharePoint Server 2010\)](#).

Content deployment planning worksheet

Download an Excel version of the [Content deployment planning worksheet](#) (<http://go.microsoft.com/fwlink/?LinkID=167835&clcid=0x409>).

See Also

[Content deployment overview \(SharePoint Server 2010\)](#)

[Design content deployment topology](#)

Design content deployment topology

Content deployment is a feature of Microsoft SharePoint Server 2010 that you can use to deploy content from a source site collection to a destination site collection. This article describes elements of topologies designed for content deployment and illustrates typical content deployment topologies. For an overview of content deployment using SharePoint Server 2010, see [Content deployment overview \(SharePoint Server 2010\)](#). For information about planning to use content deployment with your solution, see [Plan content deployment \(SharePoint Server 2010\)](#).

In this article:

- [Elements of content deployment topologies](#)
- [Typical content deployment topologies](#)

Elements of content deployment topologies

Most content deployment topologies include two or more server farms, to separate the authoring environment from the production environment. A server farm used in content deployment can have one of the following purposes:

- **Authoring** The authoring farm contains the site collection that is used by the team that creates the content.
- **Production** The production farm contains the site collection that presents the content to the intended audience. This farm usually has tightened security.
- **Staging** The staging farm contains a site collection that is a copy of the production site collection, so the content can be reviewed and tested before it is published.

On any farm that exports content, you must specify a single server that hosts the Central Administration Web site as the *export server*. Similarly, on any farm that imports content, you must specify a single server that hosts the Central Administration Web site as the *import server*. These are the servers that host the timer jobs that run the export and import operations, and that pack, transport, and unpack the .cab files that contain the content that is exported and imported as part of content deployment. The export and import servers must have sufficient disk space to hold these .cab files in addition to the uncompressed copies of the files before and after compression. For more information about the content deployment process, including a list of important considerations to be aware of when you use content deployment, see [Content deployment overview \(SharePoint Server 2010\)](#).

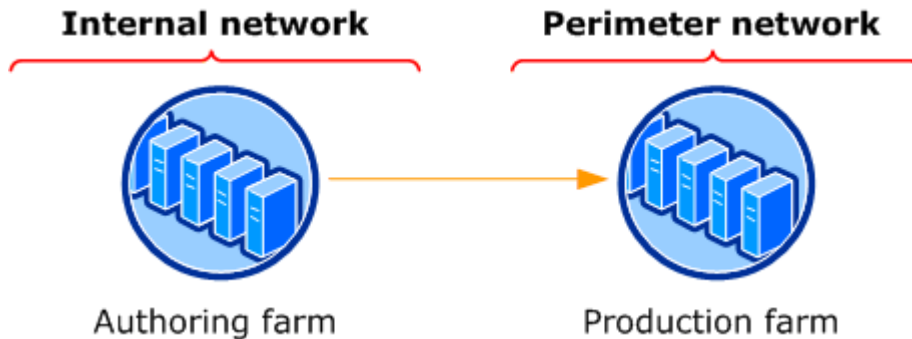
Typical content deployment topologies

This section illustrates common content deployment topologies.

Two-farm topology

The two-farm topology is a standard Internet site topology, and it is typical of topologies that are used to publish an Internet site, such as a corporation's Internet presence site or a news organization's online news site. It includes two server farms: one to host the authoring site collection along with other sites used by the authoring team, and the other to host the production site collection. For this topology, users of the production server farm belong to a separate Active Directory domain, and some production farm users might be anonymous. This topology is recommended for Internet-facing sites, and for extranet sites where users have read-only access to content.

The following figure shows a standard two-farm topology for content deployment:



In the two-farm topology, the authoring server farm contains the site collection that is used to author the site's content. A front-end Web server in the authoring farm must be configured to export content from the authoring site collection to the production farm. One server that hosts the Central Administration Web server in the production farm must be configured to import content from the authoring farm.

Often in the two-farm topology, the production farm is hosted in a perimeter network that is protected by outer and inner firewalls to increase security.

Variations on this topology include the following:

- **Single authoring farm publishing to multiple production farms** In this variation, multiple farms are deployed in the perimeter network. Each production farm can have the same content, or sites can vary from farm to farm. This topology can be configured in multiple ways:
 - The authoring farm can deploy to all the production farms.
 - The authoring farm can deploy to one production farm; by using content deployment, that production farm can then deploy to the other production farms.



Note:

Because a content deployment job is based on a path to a specific destination, deployments to multiple production farms are not synchronized. In this scenario, each production farm might have different content until all content deployment jobs have run.

- **Multiple authoring farms publishing to a single production farm** Different authoring teams, each working on their own authoring farm, can work on separate site collections that are published to separate site collections on a single production farm.

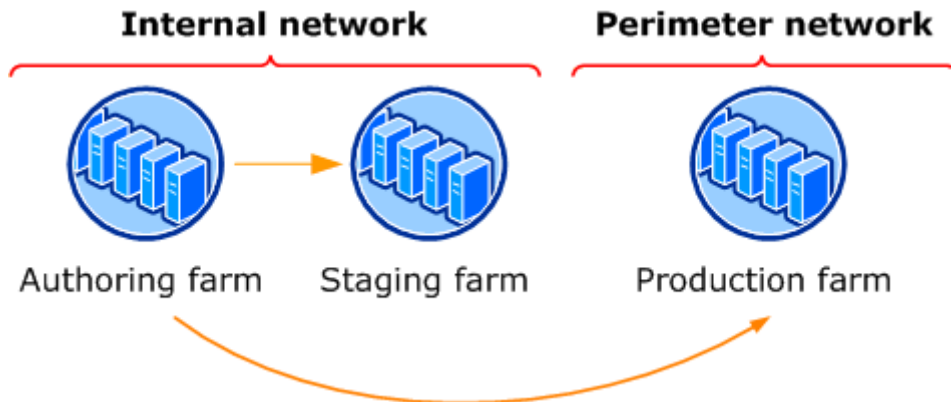
Three-stage topology

In some solutions, a three-stage topology is deployed and includes an authoring farm, a staging farm, and a production farm. The staging farm is used to test or review the content, in addition to custom Web Parts or code, before it is published to the production farm. Depending on the size of your SharePoint Server 2010 solution, the site collections for both authoring and staging can be located within the same farm, instead of two separate farms. This topology is recommended for the following situations:

- Environments where a multistage approval process is a business requirement.
- Validating content in an environment that more closely reflects the production environment before deploying it to production.
- Testing the content with custom Web Parts and code before moving it to the production farm.

In a typical three-stage content deployment topology, the authoring farm deploys to both the staging farm and the production farm. A front-end Web server in the authoring farm must be configured to export content. A front-end Web server in both the staging farm and the production farm must be configured to import content.

The following figure shows a standard three-stage topology for content deployment, where the authoring farm deploys content to both the staging farm and the production farm:



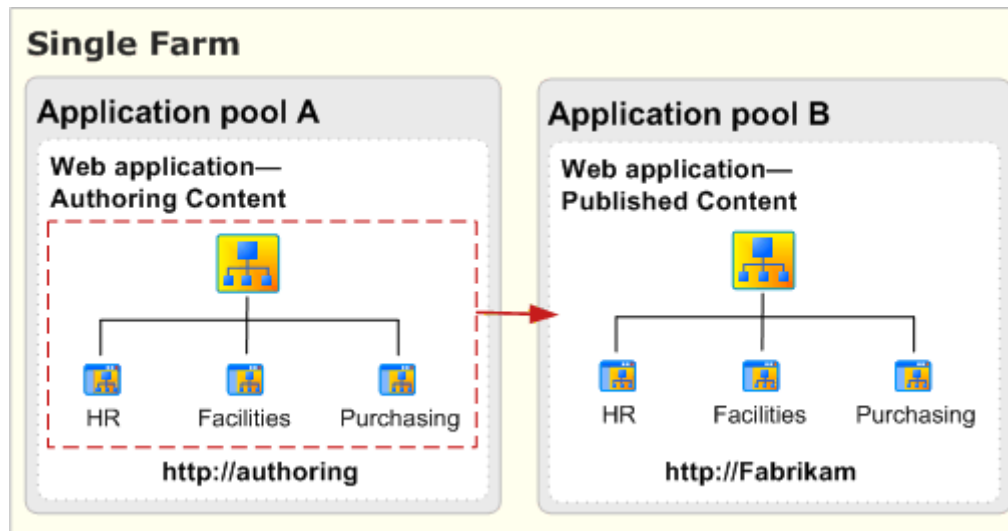
In a variation on the three-stage topology, the authoring farm deploys content to the staging farm, and the staging farm deploys content to the production farm. In this scenario, a server that hosts the Central Administration Web site in the staging farm must be configured to both import and export content.

Single-farm topology

Content deployment can be configured for use in a single server farm. In this topology, authors work in one site collection, and content is deployed to a duplicate publishing site collection on the same farm. The site collections used for authoring and production use separate content databases on the same database server. The site collections can be in the same Web application, or in separate Web applications. Security is managed by granting users permissions to the content rather than by using separate Active Directory domains. This topology is recommended for Intranet environments, external environments where verification of content or code in a staging environment is not a business

requirement, and for segregating security settings and authentication between two locations when only one farm is available or necessary.

The following figure shows a single-farm topology, where a site collection in one Web application is deployed to a site collection in another Web application in the same farm:



Note:

Using content deployment with a single-farm topology might not be the best approach for your SharePoint Server 2010 solution. One alternative to using content deployment is to extend the Web application. This option lets you have a separate IIS Web site that uses a shared content database to expose the same content to a different set of users. This is typically used for extranet deployments in which different users access content by using different domains. For information about extending a Web application, see [Extend a Web application \(SharePoint Server 2010\)](#). For a list of alternatives to using content deployment, see [Plan content deployment \(SharePoint Server 2010\)](#).

See Also

[Content deployment overview \(SharePoint Server 2010\)](#)

[Plan content deployment \(SharePoint Server 2010\)](#)

Variations overview

The *variations feature* in Microsoft SharePoint Server 2010 makes content available to specific audiences on different sites by copying content from a *source variation site* to each *target variation site*. When users visit the root site, they are redirected to the appropriate variation site, based on the language setting of their Web browser. If necessary, the content can be customized on the target variation site. For example, content on a target variation site can be translated into other languages before it is published. Variations can be used only on SharePoint Server 2010 sites that are created with one of the Publishing site templates, or on sites for which the SharePoint Server Publishing Infrastructure feature has been enabled.



Note:

Although variations can be used for multilingual solutions, the variations feature does not translate pages. To use variations for creating multilingual content, you can use workflows to route content for translation by another team or third-party vendor after the content is copied to target sites. For more information about workflows, see [Plan workflows \(SharePoint Server 2010\)](#).

This article provides an overview of the variations feature. It describes the elements of the variations feature, provides an overview of site and page creation for variation sites, lists some of the limitations of variations, and describes scenarios for using variations in SharePoint Server 2010. This article does not describe the tasks that are involved in planning a solution that uses variations. For information about planning to use variations in your solution, see [Plan variations](#). This article also does not describe how to create variation labels and hierarchies. For information about creating a variations site, see [Create a variations site](#).

In this article:

- [Use and benefits of variations](#)
- [Scenarios for using variations](#)
- [Elements of variations](#)
- [Understanding variations](#)
- [Understanding source variation and target variation site creation](#)
- [Understanding site and page creation](#)
- [Limitations of variations](#)

Use and benefits of variations

Many organizations have a global reach. However, even in domestic markets, organizations must reach a diverse customer base that might speak many different languages or that might need to have specific information that is based on regional differences, on various mobile devices, or on corporate branding.

These types of organizations need Web sites that deliver tailored content to suit different cultures, different markets, and different geographic regions. Producing and maintaining variations of a site can be difficult and time-consuming. By using variations as part of a SharePoint Server 2010 solution, site architects and site administrators can simplify the process of producing and maintaining these sites. The variations feature automates the creation of sites and pages, which eliminates having to manually create a site and all associated pages for each instance of a needed variation.

Scenarios for using variations

You can use variations to create different versions of similar content for users in many scenarios. The following table describes possible scenarios in which you might use variations.

Scenario	Description
Multiple languages	You can use variations to create sites and content for specific languages. In this scenario, the majority of the content is authored in the language of the source variation site and copied to some or all of the target variation sites for translation into different languages. For example, the content might be authored in English and be copied to target variations sites for translation into German, French and Spanish.
Multiple devices	You can customize the logic of the VariationRoot.aspx page to direct users to pages that are designed to work with different types of devices. For example, you might have target variation sites with pages designed for display on devices that have different screen sizes or screen resolutions.
Multiple locations or brands	You can use variations to create content for specific locations or brands. For example, a rental car company might have target variation sites for all the cities in which they have branch offices. Most of the company information is the same across branches, so variations are used for those pages, while other content, such as special offers or promotions, is created on the target variation sites for which it is needed.

Elements of variations

The variations feature consists of the following elements:

- **Variation root site** The variation root site provides the URL for all source and target variation sites and contains the landing page that redirects users to the correct variation site. This is not the same as the root site of a site collection, although you can specify the root site of a site collection to also be the root site of the variations hierarchy.
- **Variation labels** A variation label is an identifier that names a new variation site. Variations of a site are defined by creating variation labels, one for each planned variation.
- **Variation sites** The variation sites are the sites that are created based on the defined variation labels. There are two types of variation sites:
 - **Source variation site** The source variation site is the site where shared content is authored and published, and it is the site from which copies of the shared content are sent to the target variation sites. There can be only one source variation site in a single site collection. After a source variation site has been selected, it cannot be changed.
 - **Target variation sites** The target variation sites receive most of their content from the source variation site. Although new content can be created on a target variation site, that content is not shared with other sites and is unique to the site on which it was created.
- **Variations hierarchy** The variations hierarchy is the entire set of sites in all variation labels.
- **Variation pages** Variation pages are the publishing pages that are stored in the Pages library of the source variation site and the target variation sites. These pages and any dependent resources such as images and documents are the only content that is copied from the source variation site to the target variation sites.



Important:

We recommend that you do not add nonpublishing pages to the Pages library of a site that uses variations. If you do, the Variations Create Hierarchies Job Definition timer job might fail.

Understanding variations

The variations feature creates sites and copies content from a source variation site to one or more target variation sites. By default, the variations feature copies only publishing pages from the Pages library of the source variation site. The variations feature does not copy other site content, such as lists or other document libraries, unlike the content deployment feature, which copies all content, including lists and other document libraries, from one site to another. If the **Resources** option is configured to copy resources to target variation sites, then linked resources such as images and documents will be copied. Another important distinction between the variations and content deployment is that when the variations feature is used, copied content on target variation sites can be changed, unlike the content deployment feature, for which changing copied content is discouraged.

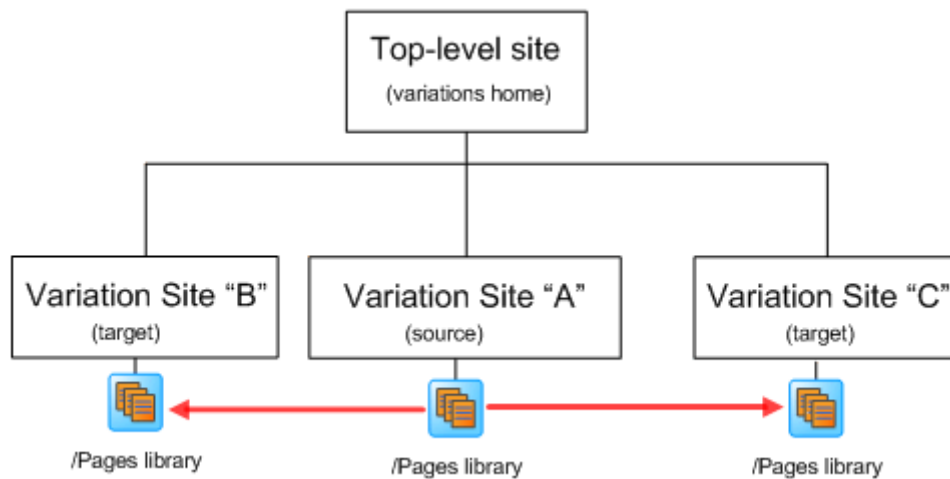
By default, when users visit the root site, they are redirected to the appropriate variation site, based on the language setting of their Web browser. For example, if a user's default browser language is French, SharePoint Server 2010 redirects that user to the French variation site. You can customize this behavior by replacing the default redirection page, VariationRoot.aspx, with a different page. This new page can implement logic that identifies the user's preferred language, the user's device, or another basis for varying sites. For information about customizing variation sites redirection, see [How to: Customize the Variation Root Landing Logic](#).

Variation labels

A variation label is an identifier that names a variation site. You select one variation label as the source, which represents the source variation site. The remaining variation labels are the target labels, representing the target variation sites to which content is copied. You create variation sites from variation labels by using the **Create Hierarchy** command on the **Variation Labels** page.

Only one set of variation labels, the variation hierarchy, can be defined for a site collection. The corresponding variation sites can be created anywhere within the site collection hierarchy. The source variation site and the target variation sites are always created as subsites of the variation root site. Users who visit the variation root site are redirected to the appropriate variation site.

The following illustration provides an example of a variation site hierarchy, and shows how publishing content from the Pages library is copied to target variation sites.



Three variation labels, "EN", "FR", and "DE" are created on the root site <http://contoso.com>. When the variations hierarchy is created, the corresponding variation sites, labeled "EN", "FR", and "DE", are created one level below the variation root site. Because site "<http://contoso.com/EN>" is specified as the source variation site, pages that are authored and published on site "<http://contoso.com/EN>" are copied to the target variation sites, "<http://contoso.com/FR>" and "<http://contoso.com/DE>".

When you create a variation label, you select a locale for it to use. The locale setting only assists with browser redirection, it does not affect the language of the user interface. If language packs have been installed on the front-end Web server, you can also select a language for the variation site. The language setting in SharePoint Server 2010 determines the language of the user interface on the variation site. If no language packs have been installed, the option to select a language is not available, and the variation site uses the default language of the SharePoint Server 2010 installation on the server, regardless of the locale that is selected for the variation label. For example, if SharePoint Server 2010 was installed by using the English version, and no language packs were installed, when a new variation label is created for the Japanese locale, the user interface for the new variation target site is in English, not Japanese. If you use variations for creating multilingual sites, and you want the user interface of a target variation site to be displayed using a specific language, you should install the language pack for each language before you create the variation sites. If a language pack is not available when a target variation site is created, the target variation site can still be created, and users can change the secondary language for a site by using the multilingual user interface. For information about the multilingual user interface, see [Multilingual user interface overview \(SharePoint Server 2010\)](#). For information about installing language packs, see [Deploy language packs \(SharePoint Server 2010\)](#).

Variation settings

Although you can specify any site within a site collection as the variation root site, variation settings are configured on the Site Collection Administration page of the top-level site within the site collection. The Variations Settings page is where you select the variation root site. After the variation root site has been selected and a variations hierarchy has been created, the root site cannot be changed.

In addition to specifying the root site, the Variations Settings page contains the following options:

- **Automatic Creation** Determines whether sites and pages on the source variation site are created automatically on the target variation sites. By default, this option is enabled. If you disable this option, sites and pages that are created on the source variation site must be manually created on the target variation sites.
- **Recreate Deleted Target Page** Determines whether a page should be re-created on a target variation site if the page was deleted from the target variation site, and the page on the source variation site has been republished. By default, this option is enabled. If you disable this option, deleted pages are not re-created on target variation sites.
- **Update Target Page Web Parts** Determines whether changes made to Web Parts on pages on a source variation site are also made on pages on target variation sites. By default, this option is enabled.
- **Notification** Sends e-mail to the contact of the welcome page of a target variation site when a new page or site is created or to the contact person of the specified page when a page is updated with revisions from the source variation site. By default, this option is enabled.
- **Resources** Specifies whether to use the same resources on the source variation site when pages are copied to target variation sites or to copy them to the target variation sites. Resources are

limited to files that are stored in a document library that can be referenced by a publishing page, such as images and documents. By default, this option is set to reference existing resources.

For information about specifying variations settings, see [Turn on variations settings so you can create variations of your site](#).

Variations timer jobs

The variations feature uses timer jobs to perform tasks such as creating and propagating sites and pages. A timer job runs inside OWSSTIMER, a Windows service for SharePoint Server 2010. Each timer job has its own default schedule for when the job runs. You can change the frequency with which each job runs on the Job Definitions page on the Central Administration Web site. The variations feature uses the following timer jobs:

- **Variations Create Hierarchies Job Definition** Creates a complete variations hierarchy by creating all variation sites and pages from the source variation site, based on the variation labels. By default, this timer job runs once a day
- **Variations Create Page Job Definition** Creates pages on the target variation sites when the **Automatic Creation** option has been disabled and a user manually creates a new page. By default, this timer job runs hourly.
- **Variations Create Site Job Definition** Creates variation sites when the **Automatic Creation** option has been disabled and a user manually creates a new variation site. By default, this timer job runs every 5 minutes.
- **Variations Propagate Page Job Definition** Creates and updates pages on target variation sites after a page on the source variation site has been approved or after it has been manually submitted by a user. By default, this timer job runs hourly.
- **Variations Propagate Site Job Definition** Creates variation sites when the **Automatic Creation** option is enabled. By default, this timer job runs every 5 minutes.

For information about timer jobs, see [View timer job status \(SharePoint Server 2010\)](#).

Understanding source variation and target variation site creation

Source variation and target variation sites are always created one level below the variation root site. Each variation site is created by using the same site template that is used to create the variation root site. However, each variation site can use a separate theme and a different master page. For information about themes, see [Themes overview \(SharePoint Server 2010\)](#).

When the variations hierarchy is first created, only sites that are based on the list of defined variation labels are created. If the variation root site has sites below it in a hierarchical site structure, and you want those sites to be included in the hierarchical site structure of each variation site, you must manually create the hierarchical structure of those sites below the source variation site after you have created the variation hierarchy. By default, the next time the Variations Create Hierarchies Job

Definition timer job runs, the sites are copied only to any new target variation sites that are created at that time. For information about how sites below the source variation site are created on existing target variation sites, see [Understanding site and page creation](#).

After the variations hierarchy is first created, whenever a new label is added to the variations hierarchy and the **Create Hierarchies** button is clicked, a new target variation site is created for each new label. By default, if the source variation site has content in the Pages library, or if it contains sites below it in the site hierarchy, those pages and sites are created on all new target variation sites only.

Understanding site and page creation

By default, any sites that are created below the source variation site and any pages that are published on the source variation site or on any sites below it in the site hierarchy are automatically copied to the target variation sites. If **Automatic Creation** has been disabled, sites and pages must be created manually on the chosen target variation sites.

This section describes the ways in which sites and pages are created on target variation sites.

Site creation

The first time the Variations Create Hierarchies Job Definition timer job runs and creates the variations hierarchy from the list of variation labels, only the source variation and target variation sites are created. After the source variation site has been created, you can create sites below it in the site hierarchy, and those sites are then created on the existing target variation sites the next time the Variations Propagate Site Job Definition timer job runs. If **Automatic Creation** is disabled, use the Site Content and Structure page on the source variation site to manually create, on a single target variation site, any site in the site hierarchy that is below the source variation site. The new site is created on the specified target variation site when the next Variations Create Site Job Definition timer job runs. You can do this any time that **Automatic Creation** is disabled.



Note:

When source variation and target variation sites are created, they are created using the default site definition provided by the template selected when the source label was created. No custom site configurations or settings are copied to the new sites. If you want the source variation and target variation sites to have custom site configurations or settings, such as navigation customizations, you must make those changes on each site after creating the variations hierarchy.

Page creation

If the **Publishing Site** template was selected when the source variation site was created, pages on the source variation site or on any site below it in the site hierarchy must be published before they are eligible to be copied to target variation sites. If the **Publishing Site with Workflow** template was selected, pages must be approved for publication by using the publishing workflow before they are eligible to be copied to target variation sites. By default, after a new page has been published or

approved for publication, if it uses workflows, it is copied to all target variation sites when the next Variations Propagate Page Job Definition timer job runs. If the page was published previously and has been modified and republished on the source variation site, it is copied to all target variation sites when the next Variations Propagate Page Job Definition timer job runs.

If the **Resources** setting is configured to enable copying resources, dependent resources, such as images, are copied to the target variation site. If versioning is enabled for the library on the target variation site where the resource is copied, and a copy of the resource already exists, the new version is appended to the existing resource, and the version history is updated. If versioning is not enabled for the library on the target variation site where the resource is copied, and a copy of the resource already exists, the new version overwrites the existing resource.

The first time a page on a source variation site is copied to a target variation site, the page title and content is copied to the target variation site. The next time the page is copied to the target variation site, only the page content is copied. The page title is not updated on target variation sites after the page is copied.



Note:

On target variation sites, a page that is copied from the source variation site is always assigned a minor version number. If the page is new to the target site, it is assigned version 0.1. If the page already exists on the target variation site, the copied page is assigned the next available minor version number. For example, if a target variation site has version 2.1 of a page and a new variation of that page is copied to the target site, the page becomes version 2.2. Pages and additional resources, such as images that are approved for publishing on the source variation site, are copied to the target variation site with their Approval status set to Draft, and they must be approved before they can be viewed by readers of the site.

If **Automatic Creation** has been disabled, a user must create the page for a specific variation label by using the **Create** command in the **Variations** group on the **Publish** tab of the page on the source variation site. The new page is copied to a single target variation site when the next Variations Create Page Job Definition timer job runs. If the page was published previously and has been modified and republished on the source variation site, it is copied only to the specified target variation site when the next Variations Propagate Page Job Definition timer job runs. For information about how to disable **Automatic Creation** for variation pages, see [Manage automatic propagation of variation pages](#). For more information about propagating variation pages, see [Variations: Propagate Pages on Your Terms](#).

By default, when a page that has been copied from the source variation site is deleted from a target variation site, that page is re-created on the target variation site the next time it is published on the source variation site and the next time the Variations Propagate Page Job Definition timer job runs. If **Recreate Deleted Target Page** has been disabled, deleted pages are not re-created on the target variations sites.

For information about creating variation source pages and working with content on variation target pages, see [Work with content updates on variations pages](#).

Limitations of variations

The following list contains information about the limitations of the variations feature in SharePoint Server 2010:

- **Variations feature is a single-tier hierarchy.**

The source and target variation sites exist at the same level within the site hierarchy, one level down from the variations root site. Although source and target variation sites can contain other sites, they cannot contain other target variation sites. For example, if you have a source variation site in English, and a target variation site in French, which has a French-Canadian site below it, the variations feature will only copy content from the English source variation site to the French target variation site. The variations feature cannot also copy content from the French target variations site to the French-Canadian site below it.

- **Variations feature does not copy everything from the source site.**

By default, the variations feature only copies publishing pages that are stored in the Pages library of the source variation site. Any additional resources, such as images that are used in those pages, are not copied but instead are referenced from the source variation site. You can choose to have those resources copied to the target variation sites by changing the **Resources** option on the Variation Settings page. Other site content, such as lists or libraries, are not copied at all and must be copied manually to any target variation site where those items are needed.

- **Content copying is unidirectional.**

The variations feature copies content from a source variation site to one or more target variation sites. You cannot use the variations feature to copy content from a target variation site back to a source variation site. Also, target variation sites cannot copy content to other target variation sites.

See Also

[Plan variations](#)

Plan variations

The *variations feature* in Microsoft SharePoint Server 2010 makes content available to specific audiences on different sites by copying content from a *source variation site* to each *target variation site*. If necessary, the content can then be customized in the target variation site. For example, content on a target variation site can be translated into other languages before it is published. Variations can be used only on SharePoint Server 2010 sites that are created with one of the Publishing site templates, or on a site for which the SharePoint Server Publishing Infrastructure feature has been enabled.



Note:

Although variations can be used for multilingual solutions, the variations feature does not translate pages. To use variations for creating multilingual content, you can use workflows to route content for translation by another team or third-party vendor after the content is copied to target sites. For more information about workflows, see [Plan workflows \(SharePoint Server 2010\)](#).

This article contains information about important items that you should consider when you are using variations in a publishing site collection, and it describes the tasks that are involved in planning a solution that uses variations in SharePoint Server 2010. This article does not provide an overview of variations, describe how to use variations, or explain how to create variation labels and hierarchies.

For more information about variations, see [Variations overview](#).

In this article:

- [About planning variations](#)
- [Important items to consider when planning to use variations](#)
- [Determine the types of variations needed](#)
- [Select the variation root site](#)
- [Specify the source variation site](#)
- [Plan target variation sites](#)
- [Decide how sites and pages will be created on target variation sites](#)
- [Plan variations timer job scheduling](#)
- [Variations planning worksheet](#)

About planning variations

The planning process that is described in this article starts with describing important items that you should consider when you plan to use variations with your SharePoint Server 2010 solution. The remainder of the article describes the steps that are required to plan for using variations with SharePoint Server 2010. These steps include the following:

- Determine the type of variations that are needed and select the variations root site.

- Specify the source variation site and plan the target variation sites.
- Decide how sites and pages will be created on the target source sites.
- Plan how variations timer jobs will be scheduled.

You can record this information in the [Variations planning worksheet](#).

Important items to consider when planning to use variations

Before you plan to use variations with your SharePoint Server 2010 solution, you should be aware of the interaction between variations and other SharePoint Server 2010 features. This section contains information about important items to consider when you are planning to use variations with your SharePoint Server 2010 solution.

Content approval

Content approval is the method by which site members with approver permissions control the publication of content. In content approval, content is considered published when a new major version is approved, because major versions are viewable by users who have read permissions. Content approval in variation sites requires major and minor versioning in the Pages libraries on the source and target variation sites. For more information about content approval, see [Plan content approval and scheduling](#).

Content approval has unique factors that you should consider when you are planning variations:

- On the source variation site, when a page is submitted for approval, an e-mail is generated and sent to the contact of the page by using the values stored in the following columns, in the following order:
 - Contact
 - Contact E-Mail Address
 - Modified By



Note:

Starting with the Contact field, each field is checked for contact information to which the notification can be sent. If the first field is empty, the next field is tried until contact information is found.

After the page is approved for publication, it is enabled for copying to the target sites. You can configure the variations settings so that content is either manually or automatically copied to the variation sites.

- On target variation sites, a page that is copied from the source variation site is always assigned a minor version number. If the page is new to the target site, it is assigned version 0.1. If the page already exists on the target variation site, the copied page is assigned the next available minor

version number. For example, if a target variation site has version 2.1 of a page and a new variation of that page is copied to the target site, the page becomes version 2.2.

- If a page is published on the source variation site, when the page is copied to target variation sites, the Approval status is set to Draft. If Content Approval and Document Versioning are enabled in the Pages library on target variation sites, the page must be approved on each target variation site before the page is available to readers.



Note:

Changes made to content that originates on the source variation site can supersede changes made to the content in the target variation site. For example, in language variations, if the source variation site is in one language and the target variation site is in another, the following situation might occur: An editor changes a localized page on the target variation site, assigning it a new minor version, 1.1. Then, a writer on the source variation site makes different changes to the same page, which is copied to the target variation site as version 1.2. This supersedes the version 1.1 changes to the page. In this example, an editor for the target variation site would have to restore the previous version of the topic from the Version History, accept the new version, or use the View Changes button to view differences between the current version and previous versions of the page, and manually merge the new and previous versions into a new version. For more information about versioning, see [Versioning, content approval, and check-out planning \(SharePoint Server 2010\)](#).

Site navigation

If you are using site variations, you must plan your site navigation experience carefully. In SharePoint Server 2010, site navigation links to the current site's peer sites are automatically generated and displayed in the Global Navigation and Current Navigation menus of a Web page. In variation sites, where the current site's peers are variations of the same site, you might not want to give site users the ability to browse to other sites in this manner. You can use the Navigation Settings page to change the settings of the Global Navigation and Current Navigation menus so that peer sites are not displayed.



Note:

Changes to site navigation on the source variation site are not copied to the target variation sites. If you want the site navigation on target variation sites to match the site navigation on the source variation site, you must manually make changes to the site navigation settings on the target variation sites.

For more information about site navigation planning, see [Plan site navigation \(SharePoint Server 2010\)](#).

Content deployment

Content deployment copies content from a source site collection to a destination site collection.

We recommend that you disable any content deployment jobs that include the source variations site while Variations Create Hierarchies Job Definition runs. If content deployment runs while the Variations

Create Hierarchies Job Definition timer job is also running, a target variation site that is only partially created may be copied to the target variation site on the destination server.

For more information about content deployment planning, see [Plan content deployment \(SharePoint Server 2010\)](#).

Web Parts

A Web Part is one of the building blocks of pages based on SharePoint Server 2010. Most Web Parts are designed to display a specific type of data, such as text, HTML, or images. SharePoint Server 2010 includes a set of Web Parts, and you can also develop or import custom Web Parts.

Web Parts are copied with pages of variation sites. If **Update Target Web Parts** is enabled, Web Parts are updated only on the target variations site if a page containing that Web Part has been copied to the target variation site. If the Web Part appears on multiple target pages on different target variation sites, all instances of the Web Part are updated when the page from the source variation site is copied to the target variation sites. For example, if a page contains a Media Web Part, and the Web Part is configured to point to video A that is stored on a site outside of the variations hierarchy, when the page is published on the source variation site, the page is copied to all target variation sites. The pages on the target variation sites display video A in the Media Web Part. If the page on the source variation site is updated and the Media Web Part is changed to point to Video B, the pages on the target variation sites continue to point to video A until the page on the source variation site is published and copied to the target variation sites. Also, if readers on the target variation sites do not have permission to view the video file that is stored outside the variations hierarchy, they cannot view the video. To prevent unintended Web Parts behavior such as this, you can configure the variations settings so that Web Part updates are not copied to target variation sites.

Multilingual sites

SharePoint Server 2010 has several features that enable you to support users in different regions or users who speak different languages. You can use these features to create Web sites in different languages and to enable users to view the user interface of a site in a language other than the one in which a site was created. If you plan to use variations with multilingual sites, there are additional steps that you must follow to enable multilingual support for the variation sites. For more information about how to plan multilingual sites, see [Plan for multilingual sites \(SharePoint Server 2010\)](#).

Determine the types of variations needed

Although variations are most commonly used to create multilingual sites, you can also use variations to create sites based on other criteria, such as regional differences, mobile devices, corporate or product branding, or company subsidiaries. Determine what types of variations are needed for your solution, and make a list of the sites that will belong to the variations hierarchy.



Note:

If you will be using variations for creating multilingual sites, and if you want the site administration pages of the target variation sites to be displayed in another language, you must install the language pack for each language that corresponds to a variations site. For information about how to install language packs, see [Deploy language packs \(SharePoint Server 2010\)](#).

Select the variation root site

The variation root site provides the URL for all variations sites and contains the landing page that redirects users to the correct variations site. The root site can be a site at any level in a site collection, including the top-level site. However, after you specify the root site, you cannot change it after you click **Create Hierarchies**, and you cannot use variations anywhere else in the site collection.



Note:

If the root site contains a site hierarchy or content in the Pages library that you want to be in the source variation site, you must manually copy them to the source variations site after it has been created.

Decide what site you want to use as the root site, and record the URL in the variations planning worksheet.

Specify the source variation site

The source variations site is the site where content to be shared by all sites is authored and published, and it is the site that is used to copy changes to the target variation sites. A source variation site is specified by selecting the **Source Variation** setting for a variation label on the Create Variation Label page in the Site Collection Administration Web site. There can be only one source variation site for a variations hierarchy. After a variation label has been specified for use as the source variations site, it cannot be changed.

Review the list of sites for your variations solution, and decide which site will be the source variation site. Record the source variations site for your solution in the top row of the **Sites data sheet** tab in the variations planning worksheet. Type the label as you want it to appear in the site URL, the label display name, and the name of the site owner who will be responsible for the site.

Plan target variation sites

Your planning goals for target variation sites will vary depending on the type of variation sites that you are implementing. For example, if the variation sites are based on languages, you might want the pages in the target variation sites to look the same as the pages in the source variation site. If the variations are based on devices, you will probably want the pages in the target variation sites to look different from the pages in the source variation site so that they fit the devices intended to display them.

Review the list of sites for your variations solution, and record each target variation site in the variations planning worksheet. For each site, enter the label as you want it to display in the site URL, the label display name, and the name of the site owner who will be responsible for the site.

Plan custom master pages, layout pages or style sheets

You can decide to use custom master pages, page layouts, or style sheets for variation sites. For example, in site variations based on devices, you could define simpler page layouts for those variations that are displayed on devices with limited screen sizes. Master pages, page layouts, and style sheets are managed only in the top-level site of a site collection, so you must make all changes to these resources in one of the following locations in the top-level site:

Resource	Location
Master Page	Master Page Gallery
Layout Page	Master Page Gallery
XSL styles and cascading style sheets	Style Library

You can also specify that each target variation uses its own custom page layout.

Plan custom content types

You must use the same content type (either the Page content type, or a content type based on the Page content type) for all pages that are stored in the Page library of the source variation site. The easiest way to do this is to use a content type that is defined in the Site Content Type Gallery in the top-level site of the site collection.

If you must have custom columns for one or more variation sites' pages, add them to the content type that you are using in your Page libraries. For example, if the variation sites are based on language differences, you can add a column that indicates whether the page has been localized. For more information about planning content types and columns, see [Content type and workflow planning \(SharePoint Server 2010\)](#).

In some site variations solutions, you do not have to edit or change the content on the target variation sites. For example, if your variation sites are device-dependent, you might want identical content displayed on all devices but with different layouts and master pages for each type of device. When this is the case, you can simplify the administration of your target variation sites by disabling required check-outs.

Decide how sites and pages will be created on target variation sites

If the source variation site will have other sites below it in a site hierarchy, you must plan for how those sites will be created on the target variation sites. There are two ways in which sites can be created on target variation sites: automatically and manually. By default, any new sites that are added below the source variation site will automatically be created on all target variation sites. However, if you disable automatic site and page creation for variations, for each new site that you create below the source variation site, you must manually specify the target variation sites on which the site should be created. Although enabling new sites to be created automatically on the target variation sites has the advantage of reducing tasks for the site administrator, the advantage to manual site creation is that you can specify which sites should be created on which target variation sites. This can be useful if you want a site to be included only on some, but not all target variation sites.

Publishing pages can also be created on target variation sites in two ways: automatically and manually. By default, any new pages that are published in the Pages library of the source variation site will automatically be created on all target variation sites. However, if you disable automatic site and page creation for variations, for each new page that you create on the source variation site, you must manually specify the target variation site on which the page should be created, and you can only specify a single target variation site. Disabling automatic site and page creation can be useful if you want certain pages to be included only on some, but not all target variation sites. However, requiring manual creation of pages on target variation sites can be a disadvantage if the source variation site will contain many pages. If you want to create variation pages on a subset of the target variation sites, we recommend that you use automatic creation to create the pages on all target variation sites, and then either delete the pages from the target variation sites where you do not want copies of the pages, or use a workflow to approve only the pages that you want to have copied to the target variation sites.

By default, automatic site and page creation is enabled. You should carefully consider the potential increase in administrative tasks if you decide to disable automatic site and page creation for your variations solution. Automatic site and page creation is configured on the Variations settings page in Site Collection Administration. For information about specifying variations settings, see [Turn on variations settings so you can create versions of your site](#).

Plan variations timer job scheduling

The variations feature uses timer jobs to perform tasks such as creating and propagating sites and pages. A timer job runs inside OWSTIMER, a Windows service for SharePoint Server 2010. Each timer job has its own default schedule for when the job runs. You can change the frequency with which each job runs on the Job Definitions page on the Central Administration Web site.

The following table lists the variations timer jobs and the default schedule for each job:

Job name	Default schedule
Variations Create Hierarchies Job Definition	Once a day
Variations Create Page Job Definition	Hourly
Variations Create Site Job Definition	Every 5 minutes
Variations Propagate Page Job Definition	Hourly.
Variations Propagate Site Job Definition	Every 5 minutes

You can specify when each job will run by setting a recurring schedule in minutes, hourly, daily, weekly, or monthly intervals. If you select daily, weekly, or monthly, you can specify a window of time for when the job should run, and the server will randomly select a time within the specified range in which to begin running the job. This option is most appropriate for high-load jobs that run on multiple servers in a farm. Be aware that running this type of job on all servers in the farm simultaneously might increase the server load and affect performance. To avoid this possibility, you can specify an exact starting time for a job.

You should carefully plan when the variations timer jobs should run on your servers, and set the recurring schedule accordingly. For example, to determine how often the **Variations Create Page Job Definition** or the **Variations Propagate Page Job Definition** should run and copy pages to target variation sites, consider how many pages will be created on the source variation site, and how often those pages will be updated.

Variations planning worksheet

Download an Excel version of the [Variations planning worksheet](#).

See Also

[Variations overview](#)

Plan information architecture for Web content management

A Web site's information architecture determines how the information in that site — its Web pages, documents, lists, and data — is organized and presented to the site's users. Information architecture is often recorded as a hierarchical list of site content, search keywords, data types, metadata, and other concepts.

Analyzing the information to be presented in an Internet or intranet Web site is an important early step in the site planning process, and this step provides the basis for planning the following:

- How the site will be structured and divided into a set of subsites.
- How data will be presented in the site.
- How site users will navigate through the site.
- How information will be targeted at specific audiences.
- How content will be tagged and how metadata will be managed.
- What the authoritative source is for terms.
- How search will be configured and optimized.

Although this article contains some guidance about how to analyze the information architecture requirements of your SharePoint Internet or intranet site, you will want to include an information architect or analyst on your site's planning and design team to ensure that your Web site plans fully consider the information architecture needs of your organization.

In this article:

- [General planning recommendations](#)
- [Plan the structure of your site](#)
- [Plan for social computing and collaboration](#)
- [Plan for managed metadata](#)
- [Plan for business intelligence and business data](#)
- [Plan for search](#)

General planning recommendations

It is useful to divide information analysis for Web site planning into the following stages:

- **Survey existing content and Web site structure** Your current Internet or intranet site reflects its current information architecture. Analyze and record how information and content are distributed across your current sites and subsites. Look at logs or other analysis tools to see what content is most frequently accessed and least frequently accessed.

- **Survey user requirements** Survey the current site users and intended site users, and record the kinds of information that they create or use. What information do they need in their daily work? Are they able to find that information easily? Does the current Web site structure help them understand the relationships among the different kinds of information that the site contains? Is there missing information? Note any problems the users have in finding or using information with the site's current architecture.
- **Survey business requirements** Survey the managers of the business unit or organization that the Web site is being designed for. What are the business needs of the site? Should the units or divisions of the business be reflected in the information architecture of the site? How will information be shared across business units, or will it be isolated within one unit? If the site is targeted at customers, what information should they first encounter? How will they explore information about products or services?

Use your analysis to create a detailed outline of your organization's content needs.

Plan the structure of your site

Your information architecture will determine the structure of your Internet or intranet site. By dividing the information architecture into business processes, projects, or large content groupings, and by using those divisions to sketch out a hierarchy of sites and content within each site, you can plan where information belongs within that hierarchy. For more information about how to plan the structure of sites based on Microsoft SharePoint Server 2010, see [Plan sites and site collections \(SharePoint Server 2010\)](#).

Plan for social computing and collaboration

The social computing and collaboration features in Microsoft SharePoint Server 2010 are built upon a database of properties that integrates information about people from many kinds of business applications and directory services. Your information architecture research should include information about users. This information can come from Microsoft products and technologies, such as Microsoft Exchange Server, the Active Directory directory service, and Microsoft SQL Server. It can also come from directories for tracking people in industry standard formats, such as Lightweight Directory Access Protocol (LDAP), or from line-of-business applications such as SAP.

By collecting user information, you can create unified and consistent user profiles across the organization. This will help you plan My Site Web sites, which are special SharePoint sites that provide a rich set of social networking and collaboration features that are personalized for each user. Self-service site creation enables users to create their own My Site Web sites. As you plan for collaboration features such as My Site Web sites, consider whether you should enable self-service site creation. Which users should have permission to create sites? Will there be a specific Web application where My Site Web sites are created? What kind of quotas are needed? For more information about My Site Web sites, see [Plan for My Site Web sites \(SharePoint Server 2010\)](#).

The people-related concepts that are recorded with your information architecture will also help you determine how to create a group of site users based on the business processes they participate in, the distribution lists and social networks they belong to, the content they are likely to create or view, or the organizational structure in which they work. For more information about how to plan for social computing and collaboration, see [Plan for social computing and collaboration \(SharePoint Server 2010\)](#).

Plan for managed metadata

Managed metadata is a hierarchical collection of centrally managed terms that you can define and then use as attributes for items in SharePoint Server 2010. As you plan your information architecture, consider the term sets that are needed to help categorize SharePoint items such as custom columns that are associated with documents and lists, or "Choice" or "Lookup" columns in an existing SharePoint Server site. For more information about term sets and managed metadata, see [Plan terms and term sets \(SharePoint Server 2010\)](#). You can use metadata navigation to improve list view navigation. Metadata navigation expands the capabilities of list views and combines it with a Key Filters control, which makes it easier for users to find content by filtering a view of documents to a subset based on one or more navigation filters. As you plan your information architecture, consider how users will likely use the site hierarchy to navigate document libraries, in addition to the filters that users will use to further filter data displayed in library list views. For more information about metadata navigation, see [Metadata navigation overview \(SharePoint Server 2010\)](#).

Plan for business intelligence and business data

Your information architecture includes the business intelligence and business data that your enterprise uses. By using business intelligence applications and tools you can organize your vision of organizational goals, processes, and performance requirements in a useful manner, and you can present that data as meaningful information. By using SharePoint Server 2010, you can present business data in the context of your site structure so that it is available to those audiences that need it. For example, on an enterprise's intranet site, employee payroll data — which must be available across the enterprise's divisional boundaries — could be presented on the enterprise's central site so employees could see their own data.

Data that is used by a limited audience could be presented in the sites that are used by that particular audience. For example, a customer support team could view and interact with customer support incidents in a site that is used only by that team, and a sales team could view customer data in a site that is dedicated to managing customer relationships. As another example, an enterprise's technical support team could use business intelligence features to view metrics on technical support tickets, such as how many tickets are assigned per support technician, how many are open or resolved, and what the satisfaction level of the customers is.

As you plan your information architecture, determine what business intelligence tools your enterprise needs and where the data in your existing business applications should be exposed in your enterprise's

Internet and intranet sites to be available to users who need it. For more information about business intelligence planning, see [Business intelligence planning](#). For more information about business data planning, see [Business data and processes planning \(SharePoint Server 2010\)](#).

Plan for search

As you plan your information architecture, in an information architecture survey, keep track of concepts, search terms, and properties that your site's users will use when they search for information on your site. You will want this data when you are creating the search schema for the site.

Remember that users might search for content by using broad conceptual terms to describe what they need. Your information architecture survey will be useful as a tool to help map users' terminology and concepts to the underlying information that they are looking for.

See Also

[Fundamental site planning \(SharePoint Server 2010\)](#)

[Plan Web content management \(SharePoint Server 2010\)](#)

[Plan site creation and maintenance \(SharePoint Server 2010\)](#)

Plan managed metadata (SharePoint Server 2010)

Managed metadata is a hierarchical collection of centrally managed terms that you can define and then use as attributes for items in Microsoft SharePoint Server 2010.

The first articles in this chapter explain key concepts about managed metadata in SharePoint Server 2010. Additional articles in the chapter provide guidance about how to identify managed metadata for your solution, and how to determine the services and connections that you will need to implement your solution. The articles include:

- [Managed metadata overview \(SharePoint Server 2010\)](#)
This article introduces the core concepts about managed metadata and then describes the way that you use managed metadata. The final section presents several benefits of incorporating managed metadata into your SharePoint Server 2010 solution.
- [Managed metadata service application overview \(SharePoint Server 2010\)](#)
This article describes the managed metadata service and connections and provides an example scenario for using them.
- [Managed metadata roles \(SharePoint Server 2010\)](#)
This article defines user roles and identifies the actions that members of each role can perform.
- [Plan terms and term sets \(SharePoint Server 2010\)](#)
This article contains guidance about how you can identify information that should become terms and term sets.
- [Plan to import managed metadata \(SharePoint Server 2010\)](#)
This article describes how to locate existing terms and term sets and prepare them to be imported into SharePoint Server 2010.
- [Plan to share terminology and content types \(SharePoint Server 2010\)](#)
This article contains procedures for identifying the managed metadata services and connections that your SharePoint Server 2010 solution requires.
- [Multilingual term sets \(SharePoint Server 2010\)](#)
This article describes the way that SharePoint Server 2010 supports defining and using managed metadata in more than one language.

Managed metadata overview (SharePoint Server 2010)

Managed metadata is a hierarchical collection of centrally managed terms that you can define, and then use as attributes for items in Microsoft SharePoint Server 2010.

This article introduces the core concepts about managed metadata, and then describes the way you use managed metadata. An example is used throughout this article to illustrate the concepts. The final section presents several benefits of incorporating managed metadata into your SharePoint Server 2010 solution.

In this article:

- [Understanding managed metadata](#)
- [Working with managed metadata](#)
- [Benefits of using managed metadata](#)

Understanding managed metadata

This section defines several key concepts related to managed metadata.

Terms and term sets

A *term* is a word or a phrase that can be associated with an item in SharePoint Server 2010. A *term set* is a collection of related terms. You can specify that a Microsoft SharePoint Server column must contain a term from a specific term set. *Managed metadata* is a way of referring to the fact that terms and term sets can be created and managed independently from the columns themselves.

Local term sets are created within the context of a site collection. For example, if you add a column to a list in a document library, and create a new term set to bind the column to, the new term set is local to the site collection that contains the document library.

Global term sets are created outside the context of a site collection. For example, the term store administrator could create a term set group called "Human Resources" and designate a person to manage the term set group. The group manager would create term sets that relate to Human Resources, such as job titles and pay grades in the Human Resources term set group.

Users can see only global term sets and term sets that are local to the user's site collection.

For example, imagine that you are organizing a conference. Each conference session is assigned a room, and the large sessions are also assigned an overflow room, in which attendees who do not fit into the primary room can watch the session broadcast on large screens. You decide to track all of the sessions and their associated details in a SharePoint Server list.

To keep track of the sessions and their conference rooms, you would:

1. Create a term set called "Conference rooms" to represent all of the available rooms.
2. Add a term to the term set for each conference room.
3. Create a content type called "Session."
4. Add two new columns to the content type: "Room" and "Overflow room."
5. Specify that the value of each of these columns must come from the "Conference rooms" term set.
6. Create a list of sessions.

In this example, whenever you add a session to the list, you provide values for the Room and Overflow room. If you subsequently reserve an additional room at the conference center, you would add a new term to the Conference rooms term set, and it would become a valid value for the two columns.

Managed terms, enterprise keywords, and the term store

Terms can be divided into two types:

- *Managed terms*, which are usually pre-defined, can only be created by users with the appropriate permissions, and are often organized into a hierarchy.
- *Enterprise keywords*, which are simply words or phrases that have been added to SharePoint Server 2010 items. All enterprise keywords are part of a single, non-hierarchical term set called the *keyword set*.

Managed terms and enterprise keywords are used differently. For more information, see the "Using terms" section later in this topic.



Note:

The word "managed" is often omitted when referring to managed terms when the meaning is clear from the context.

Both managed terms and enterprise keywords are stored in a database referred to as a *term store*.

Working with managed metadata

The following sections describe how terms are created and used. The conference room example is used throughout these sections to illustrate the concepts.

Creating terms

SharePoint Server 2010 includes the *Term Store Management Tool*, which you use to create and manage term sets. If you have the appropriate permissions you can use the Term Store Management Tool to:

- Create or delete a term set.
- Add, modify, or delete terms.
- Arrange managed terms within a term set into a hierarchy.
- Define synonyms.

- Import terms.
- Make enterprise keywords into managed terms by moving them into a term set.

Example:

To organize the conference sessions, you start by using the Term Store Management Tool to define a term set to represent all of the conference rooms. Then you use the same tool to add a term to the term set for each room. You start defining the following term set and terms:

- Conference rooms (term set)
 - La Nouvelle Ballroom
 - Room 256
 - Room 270
 - Mardi Gras Ballroom
 - Room 287
 - Room 391
 - Room 348b
 - Hall C
 - Hall D
 - Room 348a
 - Auditorium C
 - Auditorium D
 - Auditorium E
 - Room 354
 - Room 355
 - Room 293

The term set is becoming unwieldy; you will have to scroll through too long a list when you select a conference room. You decide to reorganize the term set based on which wing of the conference center the rooms are located in. The new term set and terms now have this structure:

- Conference rooms (term set)
 - Auditoriums
 - Auditorium C
 - Auditorium D
 - Auditorium E
 - Halls
 - Hall A
 - Hall B
 - Hall C
 - Hall D

- Ballrooms
 - La Nouvelle Ballroom
 - Mardi Gras Ballroom
- Second floor
 - Room 256
 - Room 270
 - Room 287
 - Room 293
- Third floor
 - Room 348a
 - Room 348b
 - Room 354
 - Room 355
 - Room 391

Because the terms "Auditoriums," "Halls," "Second floor," and "Third floor" are used only for grouping other terms and do not represent actual rooms in which sessions can be held, you do not make these terms available for users to select.

Using terms

A column is a location in a list in which to store information about a SharePoint Server item. When you define a column, you provide a name for the column, specify the column's type, and provide additional information that depends on the column type.

SharePoint Server 2010 introduces a new column type called *managed metadata*. When you create a managed metadata column, you specify the term set from which the column's value must come. When you want users to provide information for list items (including documents), and the valid values for the information are contained in a term set, use a managed metadata column. Create a new content type or modify an existing content type, and add the managed metadata column to the content type.

Example:

You create a new content type named "Session" to represent each session. Because conference sessions will have an associated room and overflow room, you add columns for each of these attributes. You add a column named "Room", choose **Managed Metadata** for the column's type, select the term set "Conference rooms" to provide values for the "Rooms" column, and indicate that the column can only contain a single value. You then create an additional column named "Overflow room," and map it to the same term set. Because not all sessions have an overflow room, you do not require that the column contain information. Because there may be multiple overflow rooms, you allow multiple values.

The definitions of the two columns are summarized in the following table:

Column name	Term set	Require a value?	Allow multiple values?
Room	Conference rooms	Yes	No
Overflow room	Conference rooms	No	Yes

Finally, you create a list of sessions.

Entering terms

When a user creates or uploads a new SharePoint Server item of a type that has columns that require a value, the user must provide a value. If the column is a managed metadata column, the *managed metadata control* is displayed, and the user interacts with this control to enter the value.

The managed metadata control allows the user either to type a value or to select a value by hierarchically navigating the term set that is associated with the column. If the user begins typing a value, the control displays all terms in the associated term set that begin with the characters the user has typed. The name of the term set and the term's position in the hierarchy are indicated along with the term itself.

If the column's definition allows multiple values, the user can select more than one term. If both the term set and the column's definition allow new terms to be added, the user can also create a new term and insert it at the appropriate place in the term set's hierarchy.

Example:

A conference administrator adds a new session. SharePoint Server displays a form that contains a field for each column that is associated with the Session content type. When the administrator creates the keynote session, in the **Room** field, the administrator displays the hierarchy of terms within the Conference rooms term set, and then selects **La Nouvelle Ballroom**.

The administrator cannot remember which auditorium is being used for the overflow from the keynote session, but knows that it is the middle one. The administrator types **aud**, and the following terms are displayed as options:

- Auditorium C [Conference rooms: Auditoriums]
- Auditorium D [Conference rooms: Auditoriums]
- Auditorium E [Conference rooms: Auditoriums]

The administrator selects **Auditorium D**.

Entering enterprise keywords

SharePoint Server 2010 includes a predefined column named *Enterprise Keywords*. You can add this column to content types. When a user adds a value to the Enterprise Keywords column, the *enterprise keyword control* is displayed, and the user interacts with the control to enter the value. The enterprise keyword control behaves in a similar fashion to the managed metadata control, except that the enterprise keyword control allows users to select enterprise keywords as well as managed terms.

When the user begins typing a value, the control displays the terms that begin with the characters the user has typed from both global term sets and term sets that are local to the user's site collection. The term set in which the term exists, as well as the term's position in the hierarchy are also displayed. There is usually an option for entering a new enterprise keyword as well. If the user enters a new enterprise keyword, the enterprise keyword is added to the keyword set.

The Enterprise Keywords column allows multiple values by default.

Example:

A conference administrator adds a new session to represent the welcome party for all attendees. The welcome party is being held at Mardi Gras World, which is an external venue. Because Mardi Gras World is not a member of the Conference rooms term set, the administrator cannot select it as the value of the **Room** field. However, the administrator does want to include the location, so that other administrators find this session when they search for the phrase "Mardi Gras World." The administrator notices that the Session content type includes the Enterprise Keywords column, and decides to add the venue as an enterprise keyword.

The administrator begins typing the name of the venue into the enterprise keyword field. When the administrator has typed **Mar**, the following options are displayed:

- Mardi Gras Ballroom [Conference rooms: Ballrooms]
- Create new

The administrator selects **Create New**, and creates the enterprise keyword **Mardi Gras World**, which is now added to both the Enterprise Keywords column of the list item and the keyword set.

Benefits of using managed metadata

This section describes several of the benefits of using managed metadata.

More consistent use of terminology

Managed metadata facilitates more consistent use of terms, as well as more consistent use of the enterprise keywords that are added to SharePoint Server items. You can pre-define terms, and allow only authorized users to add new terms. You can also prohibit users from adding their own enterprise keywords to items, and require them to use existing ones. Managed metadata also provides greater accuracy by presenting only a list of correct terms from which users can select values. Because enterprise keywords are also a type of managed metadata, even the enterprise keywords that users apply to items can be more consistent.

Because metadata is used more consistently, you can have a higher degree of confidence that it is correct. When you use metadata to automate business processes—for example, placing documents in different files in the record center based on the value of their department attribute—you can be confident that the metadata was created by authorized users, and that the value of the department attribute is always one of the valid values.

Better search results

A simple search can provide more relevant results if items have consistent attributes.

As users apply managed terms and enterprise keywords to items, they are guided to terms that have already been used. In some cases, users might not even be able to enter a new value. Because users are focused on a specific set of terms, those terms—and not synonyms—are more likely to be applied to items. Searching for a managed term or an enterprise keyword is therefore likely to retrieve more relevant results.

Dynamic

In previous versions of SharePoint Server, to restrict the value of an attribute to being one of a set of values, you would have created a column whose type is "choice", and then provided a list of valid values. When you needed to add a new value to set of choices, you would have to modify every column that used the same set of values.

By using managed metadata in SharePoint Server 2010, you can separate the set of valid values from the columns whose value must be one of the set of valid values. When you need to add a new value, you add a term to the term set, and all columns that map to that term set would use the updated set of choices.

Using terms can help you keep SharePoint Server items in sync with the business as the business changes. For example, assume your company's new product had a code name early in its development, and was given an official name shortly before the product launched. You included a term for the code name in the "product" term set, and users have been identifying all documents related to the product by using the term. When the product name changed, you could edit the term and change its name to the product's official name. The term is still applied to the same items, but its name is now updated.

Managed metadata service application overview (SharePoint Server 2010)

The managed metadata service application makes it possible to:

- Use managed metadata.
- Share content types across site collections and Web applications.

A managed metadata service publishes a term store and, optionally, content types; a managed metadata connection consumes these. This article describes the managed metadata service and connections, and provides an example scenario for using them. Before reading this article, you should understand the concepts described in the article [Managed metadata overview \(SharePoint Server 2010\)](#). To learn more about how to design your managed metadata service application topology, after reading this article, see [Plan to share terminology and content types \(SharePoint Server 2010\)](#). For instructions for creating a managed metadata service and connections, see [Managed metadata administration \(SharePoint Server 2010\)](#).

Managed metadata services

When you enable managed metadata in your SharePoint Server 2010 application, a managed metadata service and connection are created automatically. The service identifies the database to be used as the term store, and the connection provides access to the service. When you create new managed terms, or when users add enterprise keywords, these terms are stored in the database that is specified in the managed metadata service. When you publish a managed metadata service, a URL to the service is created. Before an administrator can create a connection to the service from another Web application, the administrator must know the URL of the service.

In addition to sharing managed metadata, you can also use the managed metadata service to share content types. By creating a new managed metadata service and specifying a site collection as the content type hub, you can share all content types in the site collection's content type gallery.

You can create multiple managed metadata services, and share multiple term stores and content types from multiple site collections. However, each managed metadata service must specify a different term store. When you specify a nonexistent database for the term store, a new database is created.

For more information about creating a managed metadata service, see [Create, update, publish, or delete a managed metadata service application \(SharePoint Server 2010\)](#).

Managed metadata connections

To use managed metadata, a Web application must have a connection to a managed metadata service. A Web application can have connections to multiple services, and the services can be local to the Web

application or remote. When you create a managed metadata service, a connection to the service is created automatically in the same Web application as the service.

After you create a connection to a managed metadata service, you can configure the following four options:

- **Default keyword location:** Whether to store new enterprise keywords in the keyword set in the term store associated with this managed metadata service.
- **Default term set location:** Whether to store the term sets that are created when you create new managed metadata site columns in this managed metadata service's term store.
- **Use content types:** Whether to make the content types that are associated with this managed metadata service (if any) available to users of sites in this Web application. This option is available only if the service has a hub defined to share content types.
- **Push-down Content Type Publishing updates from the Content Type Gallery to sub-sites and lists using the content type:** Whether to update existing instances of the changed content types in sub-sites and libraries.

If no connection is specified as the default keyword location, users cannot create new enterprise keywords. If no connection is specified as the default term set location, users can only specify an existing term set when they create a site column whose data type is managed metadata.



Important:

Within a Web application, do not make more than one connection the default keyword location.
Within a Web application, do not define more than one connection as the default term set location.

For more information about creating a connection to a managed metadata service, see [Create, update, or delete a managed metadata service connection \(SharePoint Server 2010\)](#).

Permissions for accessing a managed metadata service

Three things are required to create a connection to a managed metadata service:

- You must know the URL of the service.
- If this will be a cross-farm connection, the farm on which the service runs and the farm on which the connection runs must have a trust relationship.
- The service must have granted permission to the application pool account of the Web application in which the connection is created.

When you create a connection from a Web application to a service, the connection runs with the credentials of the Web application's application pool account. Users of sites in the Web application can perform different actions depending on the permission that the service grants to the application pool account. There are three levels of permission: read, restricted, and full.

The following table indicates which actions are enabled, depending on the permissions that the service grants.

Action	Read	Restricted	Full
View terms and term sets	Yes	Yes	Yes
Add existing terms and existing enterprise keywords to documents and list items	Yes	Yes	Yes
Bind columns to existing term sets	Yes	Yes	Yes
View and use content types from the content type hub (if the service provides a hub)	Yes	Yes	Yes
Add new terms to open term sets	No	Yes	Yes
Create new enterprise keywords (if the connection is configured to enable this)	No	Yes	Yes
Create local term sets (if the connection is configured to enable this)	No	Yes	Yes
Add and modify content types in the content type hub (if the service provides a hub)	No	No	Yes
Manage terms and term sets (if the user is authorized to do this)	No	No	Yes



Important:

By default, all application pool accounts that are local to the farm on which the service runs have full access to the managed metadata service. To grant an application pool account lesser

permission - that is, read-only or restricted access - you must first remove or reduce the permissions of the **Local farm** group.

For more information about granting permissions to access a managed metadata service application, see [Grant permission to access the managed metadata service \(SharePoint Server 2010\)](#).

Example scenario

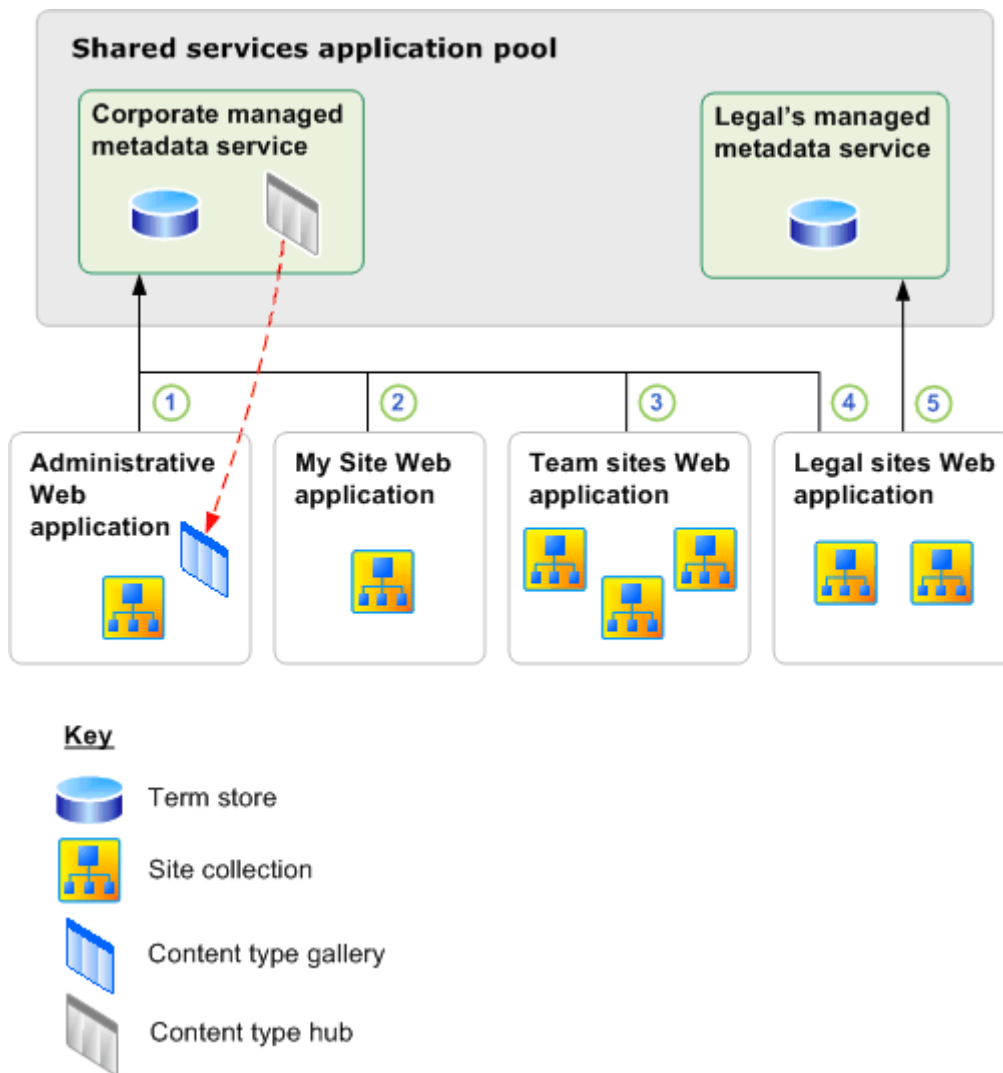
This example shows how a company configures its managed metadata services and connections. The company has already made the following decisions, which affect the design of the managed metadata services infrastructure:

- There is a corporate taxonomy that is strictly controlled and is used by everyone.
- All keywords will be stored centrally, to encourage people to reuse the same keywords.
- Every document that is created must include values for a core set of properties. The **document** content type has been updated to include columns for these additional properties.
- My Site Web sites will reside in one Web application, and site collections for various team sites will reside in another Web application.

The legal department also has a requirement that affects how the managed metadata services are configured. The legal department plans to represent confidential information by using term sets. These term sets must be available to users of all of the legal department's site collections, but users of other site collections may not view or use these term sets.

Design

The following figure shows how the company designs its managed metadata services infrastructure. The design is described in detail in the paragraphs that follow the figure.



The corporate managed metadata service is the primary managed metadata service for all SharePoint Server 2010 sites in the company. The corporate taxonomy is represented by global term sets in the term store that is associated with the corporate managed metadata service. The content type hub that is associated with the corporate managed metadata service makes shared content types available to users of all site collections.

Every Web application has a connection to the corporate managed metadata service. The connections from the My Site Web application, the team sites Web application, and the legal sites Web application, numbered 2, 3, and 4 in the figure, all have restricted access to the corporate managed metadata service. Restricted access lets users of the sites in these Web applications use the shared content types and global term sets, to add new enterprise keywords, and to create local term sets, but it prohibits them from modifying global term sets.

The administrative Web application hosts the site collection from which authorized users manage the corporate taxonomy and the shared content types. The site collection's content type gallery contains the shared content types, such as the updated **document** content type that reflects the additional required properties. This content type gallery is the content type hub of the corporate managed metadata service. The connection from the administrative Web application, numbered 1 in the figure, has full access to the corporate managed metadata service.

The term store that is associated with the legal department's managed metadata service contains term sets that represent confidential information that the legal department uses. Only the legal sites Web application has a connection to legal's managed metadata service, so that users of the site collections in the legal sites Web application can manage their term sets.

Permissions

The following table summarizes the permission that each managed metadata service grants to the accounts that the connections use to access the service. Note that **local farm** is explicitly given reduced permission. If you do not remove or reduce the permissions for local farm, other local accounts will connect to the services by using the permissions that are specified for local farm.

Account	Corporate managed metadata service	Legal's managed metadata service
Local farm	Read permission	No permission
Administrative Web application's application pool account	Full permission	No permission
My Site Web application's application pool account	Restricted permission	No permission
Team sites Web application's application pool account	Restricted permission	No permission
Legal sites Web application's application pool account	Restricted permission	Full permission

Connection parameters

All connections to the corporate managed metadata service specify that the corporate managed metadata service is the default location to store keywords. Because the connection from the legal sites Web application to the corporate managed metadata service is the default keyword location, its connection to legal's managed metadata service is not the default keyword location.

The connections from the administrative Web application, the My Site Web application, and the team sites Web application to the corporate managed metadata service specify that the corporate managed

metadata service is the place to store column-specific term sets. The connection from the legal Web application to the corporate managed metadata service specifies not to store column-specific term sets. The connection from the legal Web application to the legal managed metadata service specifies that it is the default location for column-specific term sets.



Note:

Either managed metadata service would be an acceptable location for column-specific term sets from the legal sites Web application. Because column-specific term sets are local to the site collection from which they are created, users of other site collections cannot see them.



Security Note:

Metadata publishing should not be enabled for any library that contains documents that might have metadata that other users should not see. Metadata publishing is disabled by default. For more information about metadata publishing, see [Configure Enterprise Metadata and Keyword Settings for a list or library](#).

All connections to the corporate managed metadata service specify that they will use content types and that they will push down content type changes. Because the legal's managed metadata service has no content type hub, the connection to legal's managed metadata service does not specify to use content types or to push-down content type changes.

The following table summarizes the connection settings for each connection to a managed metadata service. The connection numbers refer to the lines in the previous figure.

Connection	Managed metadata service	Web application	Default keyword location	Column-specific term set location	Use content types	Push-down content types
1	Corporate managed metadata service	Administrative Web application	Yes	Yes	Yes	Yes
2	Corporate managed metadata service	My Site Web application	Yes	Yes	Yes	Yes
3	Corporate managed metadata service	Team sites Web application	Yes	Yes	Yes	Yes
4	Corporate managed	Legal sites Web application	Yes	No	Yes	Yes

Connection	Managed metadata service	Web application	Default keyword location	Column-specific term set location	Use content types	Push-down content types
	metadata service					
5	Legal's managed metadata service	Legal sites Web application	No	Yes	N/A	N/A

See Also

[Managed metadata overview \(SharePoint Server 2010\)](#)

[Plan to share terminology and content types \(SharePoint Server 2010\)](#)

[Create, update, publish, or delete a managed metadata service application \(SharePoint Server 2010\)](#)

[Create, update, or delete a managed metadata service connection \(SharePoint Server 2010\)](#)

[Grant permission to access the managed metadata service \(SharePoint Server 2010\)](#)

Managed metadata roles (SharePoint Server 2010)

A user's role determines how the user can work with managed metadata. This article defines user roles and identifies the allowed actions associated with each role.

Roles and capabilities

Local term sets are created within the context of a site collection. For example, if you add a column to a list in a document library, and create a new term set to bind the column to, the new term set is local to the site collection that contains the document library. *Global* term sets are created outside the context of a site collection. Regardless of a user's role, the user can work with only global term sets and term sets that are local to the user's site collection.

The following table identifies user roles and the actions that members of each role can perform on managed metadata.

Role	Allowed actions
All users	Validate values for managed metadata columns. Create new enterprise keywords (if the managed metadata connection allows this).
Contributor	Create, rename, copy, reuse, move, and delete term sets. Modify a term set's description, owner, contact, stakeholders, submission policy, and whether the term set is available for tagging. Create, rename, copy, reuse, merge, deprecate, move, and delete terms. Modify a term's description, labels, default label, and whether the term is available for tagging.
Group manager	Perform all actions of the contributor role. Import a term set. Assign users to the contributor role or remove users from the contributor role.
Term store administrator	Perform all actions of the group manager role. Create and delete term groups.

Role	Allowed actions
	Assign users to the group manager role or remove users from the group manager role. Modify the working languages for the term store.
Farm administrator	Create a new term store. Connect to an existing term store. Assign users to the term store administrator role or remove users from the term store administrator role.

A user who is identified as a term set owner, term set contact, or term set stakeholder cannot perform any activities on managed metadata. The values term set owner, term set contact, and term set stakeholder exist only for documenting people or groups that are related to the term set in some way.



Note:

Identifying a user as the term set contact does not allow the user to perform any particular actions. However, if a contact is specified for a term set, site users will be able to make suggestions about the term set, and the suggestions will be sent to the contact.

End users of Microsoft SharePoint Server, such as site collection users, can perform the following activities that involve managed metadata:

Create search queries that are based on terms, and refine the results based on term sets.

Add managed metadata (managed terms and enterprise keywords) to documents and other SharePoint Server items, if the user has permission to modify the items.

Add enterprise keywords to non-SharePoint Server items, such as external Web pages and blog posts, if SharePoint is configured for social tagging.

Create new managed metadata fields and add the fields to content types.

When farm administrators, term store administrators, group managers, and contributors work with managed metadata, they use the Term Store Management Tool. End users perform their activities within the context of their SharePoint Server 2010 task.

See Also

[Managed metadata overview \(SharePoint Server 2010\)](#)

[Add and remove term store administrators \(SharePoint Server 2010\)](#)

Plan terms and term sets (SharePoint Server 2010)

Managed metadata is a hierarchical collection of centrally managed terms that you can define, and then use as attributes for items in Microsoft SharePoint Server 2010. Before reading this article, you should understand the concepts described in the article [Managed metadata overview \(SharePoint Server 2010\)](#).

Managed metadata and the way that you structure it can affect many parts of your SharePoint Server solution, such as the following:

- Valid values for columns, and the way users enter these values.
- The enterprise keywords that users can apply to SharePoint Server 2010 items.
- The way that search results can be refined.
- How documents are routed.
- The workflows that are applied to SharePoint Server items.
- The ways that users can sort and filter SharePoint Server items.
- If you are using social tagging, the tags that users can apply to items that are not SharePoint Server items.

In this article:

- [Plan: now or later](#)
- [About planning managed metadata](#)
- [Identify term sets](#)
- [Identify term set owners](#)
- [Determine term set groups](#)
- [Define term sets](#)
- [Managed metadata planning worksheets](#)

Plan: now or later

SharePoint Server 2010 enables you to use managed metadata even if you do not define all the terms and term sets first. You can start without any term sets and merely let users add enterprise keywords to items. As the set of enterprise keywords grows, you will probably want to organize it in some way to make finding and disambiguating keywords easier. At this point, you can create term sets and move some keywords into term sets. You might also want to clean up the terms in the new term set by deleting misspelled words, or indicating that some words are synonyms of other words.

Even if you decide to use the less formal approach of starting with enterprise keywords and creating term sets later, you should understand how to plan and organize managed metadata.

About planning managed metadata

Planning and organizing managed metadata involves the following steps:

1. Identify term sets.
2. Identify the owner of each term set.
3. Determine term set groups.
4. Define term sets.

The first three tasks are often performed by a group, a solution architect, or a taxonomist. The fourth task, defining term sets, can be distributed to the owners of the term sets.

The [Managed metadata planning worksheets](#) section contains links that you can use to download worksheets in which you can record your terms and term sets.



Note:

If you rearrange several columns in the detailed term set planning worksheet, you can also use the worksheet to import a term set into SharePoint Server 2010.

This article contains guidance about how to determine the managed metadata that your organization will use. This article does not cover how to divide the metadata among term stores or where you should deploy the services that host the term stores. For information about how to plan your managed metadata services, such as planning where you should store enterprise keywords, see [Plan to share terminology and content types \(SharePoint Server 2010\)](#).

Identify term sets

As you design your solution, you will discover attributes (metadata) that should be associated with certain items in SharePoint Server. You might also implement some of your organization's governance policies by using metadata. This section provides guidelines for identifying metadata, and explains which metadata should be *managed* metadata. Each item of managed metadata will be represented by a term set.

First, examine your solution for information that can be represented by metadata. The following table provides suggestions about where you are likely to locate metadata.

Potential metadata	Example
Custom columns associated with documents and lists	The confidentiality of a document: public, proprietary, confidential, trade secret, and so on
In cases in which the system or a user might take different actions based on a characteristic of an item, the characteristic itself is probably metadata.	If a request that was submitted by a specific individual is automatically approved, "submitted by" could be metadata.
Common things that users will want to sort or filter items on	If users are likely to filter items by the milestone that the item is associated with, "milestone" could

Potential metadata	Example
	be metadata.
Words or phrases that users are likely to “tag” items with	The subject of an image, such as “company logo” or “product XYZ icon”
“Choice” or “Lookup” columns in an existing SharePoint Server site	

Not all metadata is managed metadata, that is, a term set. The following table provides suggestions about when the metadata that you identified is likely to be managed metadata. If it is not clear whether to create a term set for an item of metadata, it is probably best not to create a term set. Instead, monitor the keywords that users add to items, and create a term set if several keywords have been created that would belong in the term set.

If the potential metadata...	... is it likely to be managed metadata?
Is something for which a user will select a valid value	Yes
Represents information that is likely to be used multiple times	Yes
Has valid values that can be organized hierarchically	Yes
Includes synonyms or abbreviations that should be aggregated	Yes
Can be applied in one language, but might be viewed in other languages	Yes
Is represented by a built-in column (such as “last modified date”)	No
Has an infinite number of valid values	No
Has different valid values in each location that it is used (such as different values for the “version” attribute depending on which product team’s site the item appears in)	No (or, if it is represented by managed metadata, it can be planned and managed at the site level.)
Has only “yes” and “no” as valid values	No

Record the term sets in the term sets planning worksheet, and add a brief description of each term set.

Identify term set owners

For each term set, you will have to determine an owner. The owner can then define the terms in the term set.

If the term set will be used across the organization, or if the terms in the term set will be strictly controlled, it is likely that there is already a group that manages the information. This group will probably manage the term set. If the information is not strictly controlled, you might have to nominate a person to own the term set and to periodically clean up the terms.

Record the owner of each term set in the term sets planning worksheet.

Determine term set groups

A term set group is a security boundary. Only a user who is designated a contributor to the group can manage the term sets in the group and create new term sets. However, all users can usually see all term sets. (For information about how to restrict certain users from seeing a term set, see [Plan to share terminology and content types \(SharePoint Server 2010\)](#).)

Organize the term sets into groups based on which users should be able to manage the term sets. If you identified one person as the owner of 25 term sets, perhaps those 25 term sets belong in a single group. If various people own a single term set each, but those people are all in the same department, this might also be a good reason to put the term sets in a single group. It is common to create groups based on the organization structure and let each organization designate a manager for the term set group. The term set group manager can then add people to the contributor role, and these people can manage the term sets that they own.

If there are term sets that some users should be unable to view, assign these term sets to separate groups. For example, if five of the term sets may only be viewed by members of the legal department, make a new term set group and include only these five term sets in the group. You will eventually create different managed metadata service instances for the public and private groups, as described in the article [Plan to share terminology and content types \(SharePoint Server 2010\)](#).

Define term sets

At this point, you have identified an owner for each term set. The owners can follow the rest of this process to define their term sets.

To define a term set, you must answer the following questions:

- What are the terms in the term set?
- How should the terms be organized?
- Who should be permitted to add terms to the term set?

Use the guidance in the following sections to define and organize the terms, and to determine whether the term set is open or closed. Record the term sets and descriptions in the detailed term set planning worksheet.

Identify the terms

Identify the terms in the term set. If you want to allow the multiple terms to represent the same concept (for example, “XYZ version 2” and “XYZ v2” might be alternate names for the same product), include both terms. Identify one term as the primary term, and indicate that the second term is a synonym of the first term.

Organize the terms

The primary organizing principle for a term set should be ease of navigation. End users will often interact with a term set by selecting one of the terms in the term set. Therefore, you should organize the term set to make it easy for users to find the correct term without too much scrolling. This suggests a hierarchical structure.



Note:

Because you can specify a custom sort order for a term set (for example, “Wednesday” should be presented before “Thursday,” even though that is not alphabetical order), you do not have to be concerned about the order in which you list the terms.

You can nest terms to a maximum of seven levels deep. Because you can declare a term to be “unavailable for tagging” (that is, a term that the user cannot select), you have flexibility in how you organize the hierarchy, as illustrated in the examples that are provided in the worksheets.

The following suggestions might help you decide how to structure a term set.

- In a well-understood domain, there is likely to be a standard organization. For example, the terms in a term set that represented sales offices might be organized using a geographic hierarchy. The first level terms could be thought of as continents. The second level terms would then be countries. The third level terms would be provinces within each country. The fourth level terms would be the cities in which the offices were located. Because the term set represents sales offices, only the fourth level of terms would be available for tagging.
- Are there already well known ways that information is organized in your organization? For example, does the folder structure on a file share represent the way that most people would organize information? Is the navigation structure of your intranet portal useful for finding information? You can refer to these examples of how information is currently structured as you organize your term set.



Note:

Think carefully before you use your organization’s internal structure to organize a term set. If your company reorganized next week, would you also have to reorganize the term set? Is the benefit worth the effort?

- If the terms in the term set are merely a list that has no clear method of organization, consider organizing them alphabetically. Make the letters of the alphabet the first-level terms, and make the terms themselves the second-level terms. If the term set is too large for that structure, you could create a second level of alphabetical organization (for example, “AA – AF”, “AG – AI”, “AJ – AR”,

and “AS – AZ”), and include the terms themselves at the third level. The terms in the levels that are merely for alphabetizing would be marked as unavailable for tagging.

Identify who can add terms

Anyone can add a new term to an open term set. Only people who have been identified as contributors to a term set group can add terms to a closed term set in the group. For example, a term set that represents cost centers would probably be closed. A term set that represents contributors to a charity might be open.

Record the details of each term set in a separate copy of the detailed term set planning worksheet.

Managed metadata planning worksheets

Download an Excel version of the [Term sets planning worksheet](http://go.microsoft.com/fwlink/?LinkId=163486&clcid=0x409) (http://go.microsoft.com/fwlink/?LinkId=163486&clcid=0x409).

Download an Excel version of the [Detailed term set planning worksheet](http://go.microsoft.com/fwlink/?LinkId=163487&clcid=0x409) (http://go.microsoft.com/fwlink/?LinkId=163487&clcid=0x409).

See Also

[Managed metadata overview \(SharePoint Server 2010\)](#)

[Plan to import managed metadata \(SharePoint Server 2010\)](#)

[Plan to share terminology and content types \(SharePoint Server 2010\)](#)

[Managed metadata roles \(SharePoint Server 2010\)](#)

Plan to import managed metadata (SharePoint Server 2010)

If your organization already has a corporate taxonomy or other hierarchical data that you want to use as managed metadata within Microsoft SharePoint Server 2010, consider importing the data instead of re-entering it. Examples of hierarchical data that you might want to import include a list of product names, a list of valid cost centers, and the hierarchy of divisions, departments, and groups within the organization.

Even if your organization does not have data to import, you could consider creating your taxonomy outside of the Term Store Management tool, and then importing the taxonomy. The Term Store Management Tool provides a convenient, simple way to create term sets and manage terms, but using it to create numerous term sets might take longer than importing the term sets. The Term Store Management Tool is convenient for day-to-day management of term sets once the term sets have been created.

In this article:

- [About planning to import managed metadata](#)
- [Locating existing data](#)
- [Organizing the data into managed metadata](#)
- [Cleaning up the data](#)
- [Formatting the data to be imported](#)
- [Importing the managed metadata](#)
- [Merging terms](#)

Before reading this article you should understand the concepts described in the [Managed metadata overview \(SharePoint Server 2010\)](#) article.

About planning to import managed metadata

Importing managed metadata involves the following tasks:

1. Locate the existing data.
2. Organize the data into SharePoint Server managed metadata.
3. Clean up the existing data.
4. Format the existing data into files to be imported.
5. Import the managed metadata.
6. Merge terms.

This article covers locating the existing data and cleaning up the data. The other tasks (organizing, formatting, importing, and merging) are not covered in this article because they are the same whether you are creating new managed metadata or importing existing data.

Locating existing data

Whether or not your organization has a formal taxonomy, there is probably data within the organization that could be used as managed metadata in SharePoint Server. Wherever you find a set of valid values for something, you are looking at potential managed metadata. Of course, the only data you should consider importing as managed metadata is data that is relevant to the business or to the scenarios that you are implementing.

Look for and gather information from the following sources:

- **Reference lists.** What lists do people turn to when they need information for their daily work? These might include a list of departments tacked to a bulletin board, a spreadsheet that contains the project numbers of all active projects, or an internal Web site that contains a company's product catalog.
- **Forms.** Examine the forms that are used in the organization. If the form contains a field for which only certain values are valid, the field might be information that you could represent with managed metadata. For example, an expense report form might have a field for a cost center. The cost center field is probably validated against a list of valid cost centers. "Cost center," therefore, might be a candidate for a SharePoint Server term set.
- **Taxonomies.** Many large companies have already created a corporate taxonomy. Certain public entities – for example, public schools – might be part of a larger system that has developed a taxonomy. Some industry groups have developed taxonomies for sharing information. Existing taxonomies are an excellent source for managed metadata.
- **Existing SharePoint Server choice fields.** If your organization has already implemented a SharePoint Server solution, examine the SharePoint Server solution for site columns whose type is Choice. When a user enters a value for a Choice site column, the user must choose from a list of valid values. That list of valid values could be a term set in SharePoint Server 2010.
- **Well-known hierarchies:** People in your organization are probably familiar with a number of common hierarchies, such as the folder structure on a file share or the navigation structure of your intranet portal. These hierarchies might guide how you organize the hierarchy of managed metadata.

Organizing the data into managed metadata

The process to organize managed metadata is the same whether you are starting with existing data or creating the data. For more information about organizing managed metadata, see [Plan terms and term sets \(SharePoint Server 2010\)](#).

Cleaning up the data

Once you have identified existing data and determined how to organize the managed metadata, you will probably need to review and refine the data that you have collected. First, however, delete any data that you have determined should not become managed metadata.

Perform the following actions for each term set in your managed metadata plan for which data already exists:

1. Merge lists. If you collected data from multiple sources, of which all will contribute terms for the term set, create a single list of the combined terms. For example, you might have collected a list of product names from the marketing department and a second list of product names from the manufacturing division. Combine these into one term set.
2. Remove duplicates. If a list of terms contains the same term multiple times, remove the duplicates. Ideally, the list of product names from the marketing department would be identical to the list from the manufacturing division, and so all terms would be duplicated.
3. Identify synonyms. If different organizations contributed terms to your list, it is very likely that they occasionally used different words for the same concept. In some cases, you might have to continue to allow different words (terms), even though they represent the same concept. When this occurs, keep all of the wordings of the term in the list. Identify the most common wording of the concept, and add a note indicating that is the primary term for that concept. Add a note to each of the other wordings indicating that they should be merged with the primary term.
4. Resolve misspellings and other errors. If the list contains multiple wordings for the same concept, and you do not have to preserve both wordings, choose one wording and remove the other wordings from the list. For example, one organization's list might hyphenate a multi-word product name while the other list separates the words with spaces. Or one organization's list might spell out the word "version" while another organization might abbreviate it to "ver". Choose one wording and remove the other wordings.

Formatting the data to be imported

For more information about the format of the import file, see [Managed metadata input file format \(SharePoint Server 2010\)](#). To find instructions for importing managed metadata, merging terms, and defining synonyms, see Office.com.

Importing the managed metadata

To find instructions for importing managed metadata, see Office.com. See [Victor Wilén's blog](#) (<http://go.microsoft.com/fwlink/?LinkId=190541&clcid=0x409>) for information about a tool to import term sets that was developed by a member of the SharePoint community.

Merging terms

When you import managed metadata, you only can add terms and term sets. You must use the Term Store Management Tool to merge terms. If your term plan contains notes indicating that some terms should be merged with other terms, merge the terms after you have imported them. To find instructions for importing managed metadata, merging terms, and defining synonyms, see Office.com.

See Also

[Managed metadata overview \(SharePoint Server 2010\)](#)

[Plan terms and term sets \(SharePoint Server 2010\)](#)

[Managed metadata input file format \(SharePoint Server 2010\)](#)

Plan to share terminology and content types (SharePoint Server 2010)

A managed metadata service publishes a term store and, optionally, content types; a managed metadata connection consumes them. To share term sets among site collections, you create a managed metadata service, and then create connections to the service from each Web application that contains a site collection. To share content types among site collections, you make one content type gallery the “hub” of a managed metadata service, create connections to the service from each Web application that contains a site collection, and specify that site collections should use the content types in the service.

This article contains procedures for identifying the managed metadata services and connections that your solution requires. The planning process that is described in this article is only a recommendation. There are other ways to structure your managed metadata services and connections.

Before reading this article, you should understand the concepts described in [Managed metadata service application overview \(SharePoint Server 2010\)](#) and [Services architecture planning \(SharePoint Server 2010\)](#).

In this article:

- [About planning managed metadata services and connections](#)
- [Identify managed metadata services](#)
- [Identify managed metadata connections](#)
- [Determine service account permissions](#)
- [Managed metadata services planning worksheet](#)

About planning managed metadata services and connections

Before you plan your managed metadata services and connections, you should already have identified the site collections in your solution and determined which site collections will be contained in which Web applications.

The planning process that is described in this article starts with the simplest configuration – a single managed metadata service – and then directs you to add more services in certain situations. Then you determine the connections to each managed metadata service, and determine the connection parameters. If you have to have more than one managed metadata service, keep track of the services that you identify, as well as a description of what each service’s term store will be used for and which site collection the service’s content type hub, if any, will come from. Also keep track of the connections that you identify, and the values of their connection parameters. You can record this information in the worksheets that are referenced at the end of this article.

Your organization's governance policies can affect how you design your managed metadata services and connections. For example, if there is a formal process for managing terms and term sets, this will affect how you set connection parameters. If every document that is created must have a certain set of attributes, you will probably want to have a content type hub in at least one service. Familiarize yourself with your organization's governance plan before you determine the managed metadata services and connections. For more information about how to create a governance plan, see [Governance overview \(SharePoint Server 2010\)](#).

Identify managed metadata services

If you intend to use terms and term sets, or if you intend to share content types across more than one site collection, your solution must use a managed metadata service. Assume for now that you will store all term sets and all enterprise keywords in a single term store. The first managed metadata service, called the "primary" managed metadata service in this article, contains this term store. If your solution includes shared content types, assume that a single content type gallery will contain all shared content types. This content type gallery, if it exists, is the hub of the primary managed metadata service. In many cases, the primary managed metadata service is the only managed metadata service that you will need.

A *local term set* is a term set that is created within the context of a site collection. For example, if you add a column to a list in a document library, and create a new term set to bind the column to, the new term set is local to the site collection that contains the document library.

A *global term set* is a term set that is created outside the context of a site collection. For example, the term store administrator could create a term set group called "Human Resources" and designate a person to manage the term set group. The group manager would create term sets that relate to Human Resources, such as job titles and pay grades in the Human Resources term set group. The keyword set, which contains enterprise keywords, is a global term set.

Users can see only global term sets and term sets that are local to the user's site collection.

A content type hub is associated with the content type gallery of a site collection. If you have to share content types from the content type galleries of multiple site collections, add a managed metadata service for each required content type hub.

Identify managed metadata connections

Add a connection from the primary managed metadata service to every Web application that contains a site collection. Use the following guidelines to determine the values of the connection's parameters.

- **Default keyword location:**
 - NO, if users of sites in the Web application might create an enterprise keyword that should be kept private.
 - Otherwise, YES.
- **Default column-specific term set location:**

- NO if administrators of sites in the Web application should be unable to make new term sets.
- Otherwise, YES.
- **Use content types:** YES, assuming that the primary managed metadata service has a content type hub and you have no reason not to use the content types.
- **Push down content types:** YES, assuming that the primary managed metadata service has a content type hub, and you have no reason not to push them down.

Any remaining managed metadata services will be used only for their content type hubs. Create a connection from every remaining managed metadata service to each Web application that contains a site collection that uses the content types. Use the following guidelines to determine the values of the connection's parameters:

- **Default keyword location:** NO.
- **Default column-specific term set location:** NO.
- **Use content types:** YES.
- **Push down content types:** YES, assuming you have no reason not to push them down.

For each Web application, review the values of the parameters of all connections to a managed metadata service. Check for the following conditions:

- If no connection is specified as the default keyword location, users of sites in the Web application will be unable to create new enterprise keywords. If this is not what you want, set the value of the default keyword location to YES in the connection to the managed metadata service whose term store should contain the new enterprise keywords.
- If no connection is specified as the default column-specific term set location, administrators of sites in the Web application will be unable to create new term sets when they create a managed metadata column. If this is not what you want, set the value of the default column-specific term set location to YES in the connection to the managed metadata service whose term store should contain the new term sets.
- Ensure that there is a maximum of one default keyword location per Web application. The managed metadata service does not support more than one default keyword location per Web application.
- Ensure that there is a maximum of one default column-specific term set location per Web application. The managed metadata service does not support more than one default column-specific term set location per Web application.

Determine service account permissions

When you create a connection from a Web application to a service, the connection runs with the credentials of Web application's application pool account. Users of sites in the Web application can perform different actions depending on the permission that the service grants to the application pool account. There are three levels of permission: read, restricted, and full.

The following table indicates which actions are enabled, depending on the permissions that the service grants.

Action	Read	Restricted	Full
View terms and term sets	Yes	Yes	Yes
Add existing terms and existing enterprise keywords to documents and list items	Yes	Yes	Yes
Bind columns to existing term sets	Yes	Yes	Yes
View and use content types from the content type hub (if the service provides a hub)View terms and term sets	Yes	Yes	Yes
Add new terms to open term sets		Yes	Yes
Create new enterprise keywords (if the connection is configured to enable this)		Yes	Yes
Create local term sets (if the connection is configured to enable this)		Yes	Yes
Add and modify content types in the content type hub (if the service provides a hub)			Yes
Manage terms and term sets (if the user is authorized to do this)			Yes

By default, all application pool accounts that are local to the farm on which the service runs have full access to the term store.

Use the following procedure to determine the permissions that each service will grant to each connection. Complete the planning steps once for each managed metadata service.

1. Determine the level of permission you want to grant automatically to all connections from within the local farm. Record the permission level in the **Local Farm permissions** column of the **services** page of the worksheet.
2. If any connections from within the local farm must have greater permissions than you granted to the Local Farm group, record the permission level in the **permissions** column of the **connections** tab of the worksheet.
3. If Web applications on remote farms will connect to the service, record the permission level of each remote connection in the **permissions** column of the **connections** tab of the worksheet.

Managed metadata services planning worksheet

Download an Excel version of the [Managed metadata services planning worksheet](http://go.microsoft.com/fwlink/?LinkId=164578) (<http://go.microsoft.com/fwlink/?LinkId=164578>).

See Also

[Managed metadata service application overview \(SharePoint Server 2010\)](#)

[Governance overview \(SharePoint Server 2010\)](#)

[Managed metadata overview \(SharePoint Server 2010\)](#)

[Services architecture planning \(SharePoint Server 2010\)](#)

[Create, update, publish, or delete a managed metadata service application \(SharePoint Server 2010\)](#)

[Create, update, or delete a managed metadata service connection \(SharePoint Server 2010\)](#)

[Grant permission to access the managed metadata service \(SharePoint Server 2010\)](#)

Multilingual term sets (SharePoint Server 2010)

Microsoft SharePoint Server 2010 lets you define and use managed metadata in multiple languages. This is done by distinguishing the term itself from the words that represent the term.

A *label* is a word or a phrase that represents a term. A term can have multiple labels. For example, a term set that represents the days of the week would contain seven terms. One term might have the labels “Sunday”, “dimanche”, and “domingo”. Regardless of which label a person uses when applying the term to an item, SharePoint Server recognizes it as the same term. You can configure SharePoint Server to use the label that is appropriate for the language in which a user views the site when it displays terms.

This article describes how multilingual managed metadata is created and used. Before you read this article, you should understand the concepts that are presented in the article [Managed metadata overview \(SharePoint Server 2010\)](#).

In this article:

- [Defining terms](#)
- [Using terms \(tagging\)](#)
- [How terms are displayed](#)
- [Recommendations](#)

Defining terms

Before you can define term sets in multiple languages, you must perform the following actions:

1. Install the SharePoint Server language pack for each language that you want to support.
2. From the Term Store Management Tool, add each language to the list of working languages for the term store.
3. Select one language to be the default language for the term store.

In each language, you can associate multiple labels with a term. One label in each language is the default label for the language. Use the other labels for synonyms or abbreviations.

The following table shows the labels in three languages for the terms in a term set that represents the days of the week.

English	French	Spanish
Monday (default) Mon	lundi (default)	lunes (default)
Tuesday (default)	mardi (default)	martes (default)

English	French	Spanish
Tue		
Wednesday (default) Wed	mercredi (default)	miércoles (default)
Thursday (default) Thu	jeudi (default)	jueves (default)
Friday (default) Fri	vendredi (default)	viernes (default)
Saturday (default) Sat	samedi (default)	sábado (default)
Sunday (default) Sun	dimanche (default)	domingo (default)

Within the Term Store Management Tool you can display the hierarchy of term sets and terms in any of the working languages. If a term does not have a label in the display language, the label in the default language of the term store is displayed. Term sets can also have a name in each language. If there is no name for the term set in the display language, the name in the default language is displayed.

By default, the terms in a term set are sorted according to the default sort order for the language. In many languages, this is alphabetical order. Using the default sort order often makes it easier for the user to find the correct term. However, in some cases there is a more appropriate order for the term set that is independent of language. A term set that represents clothing sizes is an example of a case in which using the default sort order for the language would not necessarily be natural. In these cases, you can specify a custom sort order for the term set. The custom sort order is used for all languages.

Using terms (tagging)

If SharePoint Server language packs are installed on the server, a farm administrator must specify a default language when creating a site collection. The site collection owner can then specify additional languages that the site collection will support. If a site collection supports multiple languages, users can change the display language when they browse to any page in the site collection. When a user changes the display language of a page, the new display language becomes the user's preferred language for the whole site collection.



Note:

Because the preferred language is stored in a cookie, the preferred language is associated with the user's computer. If a user views a site collection from multiple computers, the user may have a different preferred language on each computer.

When a user enters a term (for example, to provide a value for a managed metadata column that is bound to a term set), the user is presented with a list of options. These options are the labels *in the user's preferred language* for the terms in the term set. When a user types an enterprise keyword (for example, to add a social tag to a Web page), the suggestions that are displayed are the terms whose labels *in the user's preferred language* match what the user has typed. When a user enters a term or an enterprise keyword, if no label exists in the user's preferred language, the label in the default language for the term store is used.

When a user selects a term, instead of recording the label, SharePoint Server records an identifier that represents the term, and the language in which the term was selected. Because SharePoint Server does not simply record the label that a user selected, it can identify terms regardless of the language in which the terms were entered.

How terms are displayed

For each site collection, SharePoint Server maintains a cache of all of the terms that can be used within the site collection. Each entry in the cache contains:

- The identifier of the term.
- The text to display for the term in each language that is supported by the term store. For each language, the text is either:
 - The default label of the term in the language, if such a label exists, or
 - The default label of the term in the term store's default language, if a label does not exist in the language.

When displaying a term, SharePoint Server uses the term's identifier and the user's preferred language to obtain the text to display from the cache.



Note:

When a user enters a term, SharePoint Server uses the term store to display suggested terms and to validate the user's entry. SharePoint Server uses the cache only to display the terms that have been added to items.

Because SharePoint Server uses an identifier for a term, instead of a label, it can support the following multilingual scenarios:

- SharePoint Server displays terms in the user's preferred language, regardless of the language in which the term was entered.
- Tag clouds reflect the use of a term, regardless of the language in which the term was entered.

Changes made to the term store are propagated to each site collection's cache under the control of a timer job. There might be a delay between updating a term's label and the new label being displayed on the pages of a site collection.

Recommendations

As you plan how your SharePoint Server solution will use managed metadata in multiple languages, consider the following suggestions:

- If your SharePoint Server solution requires managed metadata in multiple languages, use the functionality described in this article. It is usually simpler to have a single term set that represents a concept, and then define labels for the terms in the term set in multiple languages, than to create a separate term set for each language.
- If you add a new working language to a term store, do so during a period when your SharePoint Server solution is not heavily used, such as at night. When you add a new working language, the next time that the timer job runs, SharePoint Server updates each site collection's cache and adds the text to display in the new language to every term in the cache. This may affect performance temporarily.
- Although you can import term sets, you can only import one label for each term. If you do not want to define labels in each language by using the Term Store Management Tool, consider using another program such as Microsoft Excel to create a file that contains the terms and term sets, and writing a custom program to import the data into the term store. For more information about how to write a custom program that manipulates managed metadata, see [the SharePoint 2010 SDK](http://go.microsoft.com/fwlink/?LinkId=190760&clcid=0x409) (<http://go.microsoft.com/fwlink/?LinkId=190760&clcid=0x409>).
- If your solution includes more than one term store, consider making each term store support the same set of working languages.

See Also

[Managed metadata overview \(SharePoint Server 2010\)](#)

[Plan for multilingual sites \(SharePoint Server 2010\)](#)

Business intelligence planning

In this section:

- [Business intelligence basics](#)

Business intelligence applications and tools enable you to organize your vision of organizational goals, processes, and performance requirements in a useful manner, and to present that data as meaningful information.

- [Excel Services overview \(SharePoint Server 2010\)](#)

Microsoft Excel 2010 is designed to help you analyze business data and increase business intelligence. Excel Services in Microsoft SharePoint Server 2010 is a Microsoft SharePoint Server shared service that you can use to publish Microsoft Excel 2010 workbooks on SharePoint Server.

- [Plan for PerformancePoint Services \(SharePoint Server 2010\)](#)

This section describes how to plan for PerformancePoint Services within a Microsoft SharePoint Server 2010 environment.

- [Plan for Visio Services \(SharePoint Server 2010\)](#)

This section discusses planning considerations for Visio Services in Microsoft SharePoint Server 2010.

- [Plan for Business Intelligence Indexing Connector \(SharePoint Server 2010\)](#)

This section describes planning considerations for Microsoft Business Intelligence Indexing Connector.

Business intelligence basics

Business intelligence applications and tools enable you to organize your vision of organizational goals, processes, and performance requirements in a useful manner, and to present that data as meaningful information.

Business intelligence planning is a key part of any deployment of Microsoft Office SharePoint Server. Start by identifying the business needs of your organization, and then determine the features of SharePoint Server that can help you collect, present, and act on data in your organization.

Plan your business intelligence needs: Consider the business processes and business data applications frequently used by your organization. What are your organizational goals? What processes and performance requirements do you want to measure and analyze? Plan the scenarios and determine the scope of the business intelligence analysis that your organization is considering.

Plan for business intelligence: Your data may exist in many forms and reside in different locations. Plan which data sources are needed for your core business intelligence and which analysis tools best meet your needs.

See Also

[Choosing a business intelligence tool in SharePoint Server](#)

[Architecture for business intelligence in SharePoint Server 2010](#)

[Secure Store for Business Intelligence service applications](#)

[Overview of documentation for SQL Server Reporting Services reports in SharePoint](#)

[Overview of PowerPivot documentation \(SharePoint Server 2010\)](#)

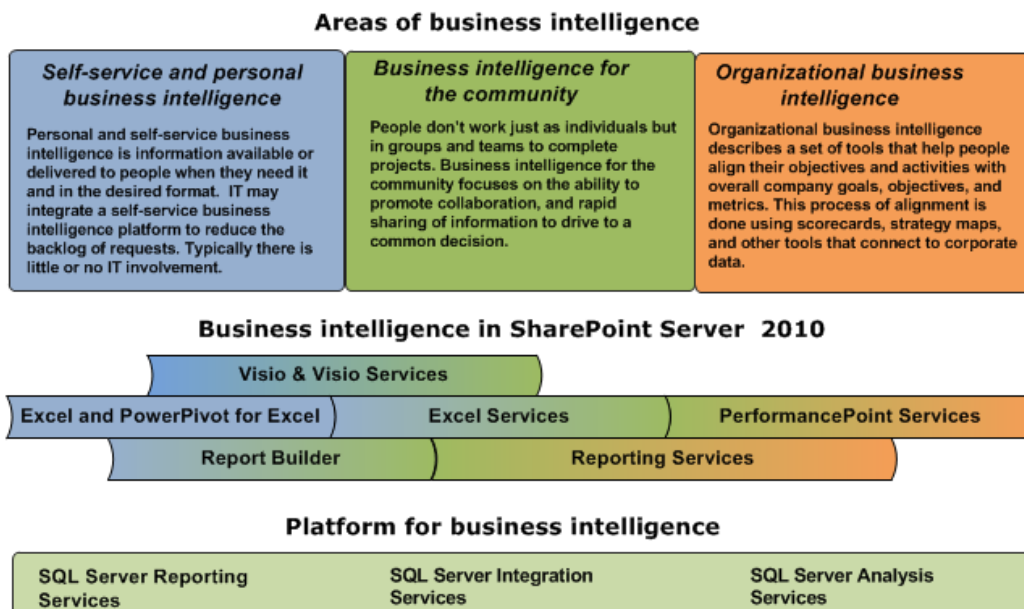
[Data warehousing, OLAP, and Analysis Services for SharePoint 2010](#)

Choosing a business intelligence tool in SharePoint Server

Microsoft has several business intelligence (BI) tools and applications that have BI features, each of which is important to understand as you decide what will work best for your situation. The BI tools that you should use depend on the specific problems that you are trying to solve.

Your daily business activities have associated information and insights that emerge in three main areas of business intelligence: personal, team, and organizational. There will be overlap across these areas. For example, a company's employees may use Microsoft Excel 2010 and Excel Services in Microsoft SharePoint Server 2010 to make relevant business decisions at the corporate level. PerformancePoint Services uses Excel, Visio Services, and Excel Services to complement its BI tools to deliver a corporate dashboard that may reflect elements of personal and team BI. By design, all Microsoft BI products interoperate so that teams and people inside an organization can move across the continuum of personal, team, and organizational and have all products work together.

The following diagram shows tools from both SQL Server and SharePoint Server to explain categories in which each tool is generally used.



Note:

This diagram shows products and tools from Microsoft Office, SharePoint 2010 Products, and SQL Server, for which separate licenses are necessary.

SQL Server provides a primary data infrastructure and business intelligence platform for trusted, scalable, and secure data. SharePoint Server can be used with SQL Server reporting and BI tools to show BI data in meaningful ways. To learn more about how SQL Server supports business intelligence in SharePoint 2010 Products, see [Overview of SQL Server in a SharePoint environment \(SharePoint Server 2010\)](#) and [Overview of SQL Server in a SharePoint environment \(SharePoint Foundation 2010\)](#).

The following sections explain when you might use a particular BI tool.

Services in SharePoint Server for business intelligence

Excel 2010

Excel 2010 is the end user's analysis tool of choice for viewing, manipulating, performing analysis on, generating intelligence from, and creating reports about an organization's data. For more information about Excel 2010, see [Microsoft Excel 2010](http://go.microsoft.com/fwlink/?LinkID=195375) (<http://go.microsoft.com/fwlink/?LinkID=195375>).

Excel Services

Excel Services is a SharePoint Server 2010 application service that provides server-side calculation and browser-based rendering of Excel workbooks. Excel Services can be used for the following:

- Real-time, interactive reporting to include parameterized what-if analysis
- Distribution of all or part of a workbook for analysis by multiple users
- A platform for building business applications

For more information about Excel Services, see [What's new for Excel Services \(SharePoint Server 2010\)](#).

Visio Services

The Visio Graphics Service is a service on the SharePoint Server 2010 platform that lets users share and view Microsoft Visio diagrams. The service also enables data-connected Microsoft Visio 2010 diagrams to be refreshed and updated from various data sources. For more information, see [Visio Services overview \(SharePoint Server 2010\)](#).

PerformancePoint Services

PerformancePoint Services in Microsoft SharePoint Server 2010 is a performance management service that uses tools to monitor and analyze business. It provides tools for building dashboards, scorecards, and key performance indicators (KPIs). PerformancePoint Services can help people across an organization make informed business decisions that align with company-wide objectives and strategy.

- You can bring together data from multiple data sources (including SQL Server Analysis Services, Microsoft SQL Server, SharePoint lists, and Excel Services) to track and monitor your data.
- The visualization Decomposition Tree is a new report type that lets you visually analyze higher-level data values from a multi-dimensional dataset.

For more information, see [What's new for PerformancePoint Services \(SharePoint Server 2010\)](#).

Matching a tool with a broad scenario

The following applications can be used in the scenarios described.

Tool	Scenario
Excel 2010	Giving users browser-based access to a server-calculated version of an Excel worksheet. Use Excel 2010 and Excel Services to view, refresh, and interact with analytic models connected to data sources. Also use them for analysis, filtering, and presentation of locally stored data.
Excel Services	Sharing content with multiple persons across an organization. Excel Services lets you take authored content in Excel 2010 and make it available in an Internet browser. Excel Services is also used with a model that can be widely distributed (for example, a mortgage calculator). In both scenarios, Excel Services enables the author to publish targeted content without making the underlying intellectual property available to consumers.
PerformancePoint Services	Creating dashboards, scorecards, and key performance indicators (KPIs) that deliver a summarized view of business a performance. PerformancePoint Services gives users integrated analytics for monitoring, analyzing, and reporting.
Visio Services	Building a visual representation of business structures that are bound to data. Examples include processes, systems, and resources. An engineer can use the visualization to create data-bound objects to represent a process.

SQL Server Reporting Services in SharePoint Server

SQL Server Reporting Services provides tools and services to help you create, deploy, and manage reports for your organization in your own Web site or in SharePoint Server. It also provides programming features that enable you to extend and customize reports. The report authoring tools work with an Office-type application and are fully integrated with SQL Server tools and components, and also the SharePoint Server environment. You can build reports on SharePoint lists, publish reports to SharePoint Server 2007 or 2010, incorporate reports inside your portal by using a Web Part for reports, and fully manage your reports published in SharePoint document libraries.

When you use SQL Server Reporting Services (SSRS) with SharePoint Server, there are two modes to select from. The standard mode is known as "Connected mode". It requires SharePoint Server, the SSRS add-in, and the SQL Server 2008 R2 Report Server. The new mode is "Local mode". It is a lightweight setup for Reporting Services to integrate with SharePoint Server. It only requires SharePoint Server and the SSRS add-in.

Use SQL Server Reporting Services when you want to deliver reports that publish at set intervals and on-demand. It's also suitable where report requirements are well established and customers are not always familiar with the underlying dataset. To view an overview with links to product documentation, see [Overview of documentation for SQL Server Reporting Services reports in SharePoint](#).

PowerPivot for Excel 2010

Microsoft SQL Server 2008 R2 PowerPivot for Microsoft Excel 2010 extends Excel to add support for large-scale data. It has an in-memory data store as an option for SQL Server Analysis Services. By using PowerPivot for Excel, you can merge multiple data sources to include corporate databases, worksheets, reports, and data feeds. There are client and server components for PowerPivot. The client is an extension to Excel workbooks that contain PowerPivot data that can be published to SharePoint Server 2010. Microsoft SQL Server 2008 R2 PowerPivot for Microsoft SharePoint 2010 is the server-side component that supports PowerPivot access in Microsoft SharePoint 2010 Products, much like Excel, which can be published to Excel Services.

Use PowerPivot for Excel when you want to combine native Excel functionality with the in-memory engine to let users interactively explore and perform calculations on large data sets and quickly manipulate millions of rows of data into a single Excel workbook for ad-hoc reports.

For more information about PowerPivot, see [Overview of PowerPivot documentation \(SharePoint Server 2010\)](#).

Architecture for business intelligence in SharePoint Server 2010

This article describes the physical architecture for application services that all enable you to implement business intelligence in Microsoft SharePoint Server.

This numbered list corresponds to the numbers in the following diagram. The diagram shows a farm deployment with four servers and the application services that you can use for business intelligence. To learn more about when you might use each service, see [Choosing a business intelligence tool in SharePoint Server](#).

1. The front-end Web servers run on Internet Information Services (IIS) and host the Web Parts for business intelligence services, Web services, and the proxy that are required for communication between the client and the service applications. A service application is a wrapper for the middle-tier business logic for an instance of the service. Your configuration of the application server and how many services are hosted will depend on the size of the farm and the number of users who need access. To learn more about services and topologies for SharePoint Server 2010, see the model called "Services in SharePoint 2010 Products" in [Technical diagrams \(SharePoint Server 2010\)](#).

The four services that are shown in the diagram are discussed in the following articles.

- [What's New \(Reporting Services in SharePoint Integrated Mode\)](http://go.microsoft.com/fwlink/?LinkId=185271) (<http://go.microsoft.com/fwlink/?LinkId=185271>)
- [Visio Services overview \(SharePoint Server 2010\)](#)
- [What's new for Excel Services \(SharePoint Server 2010\)](#)
- [What's new for PerformancePoint Services \(SharePoint Server 2010\)](#)



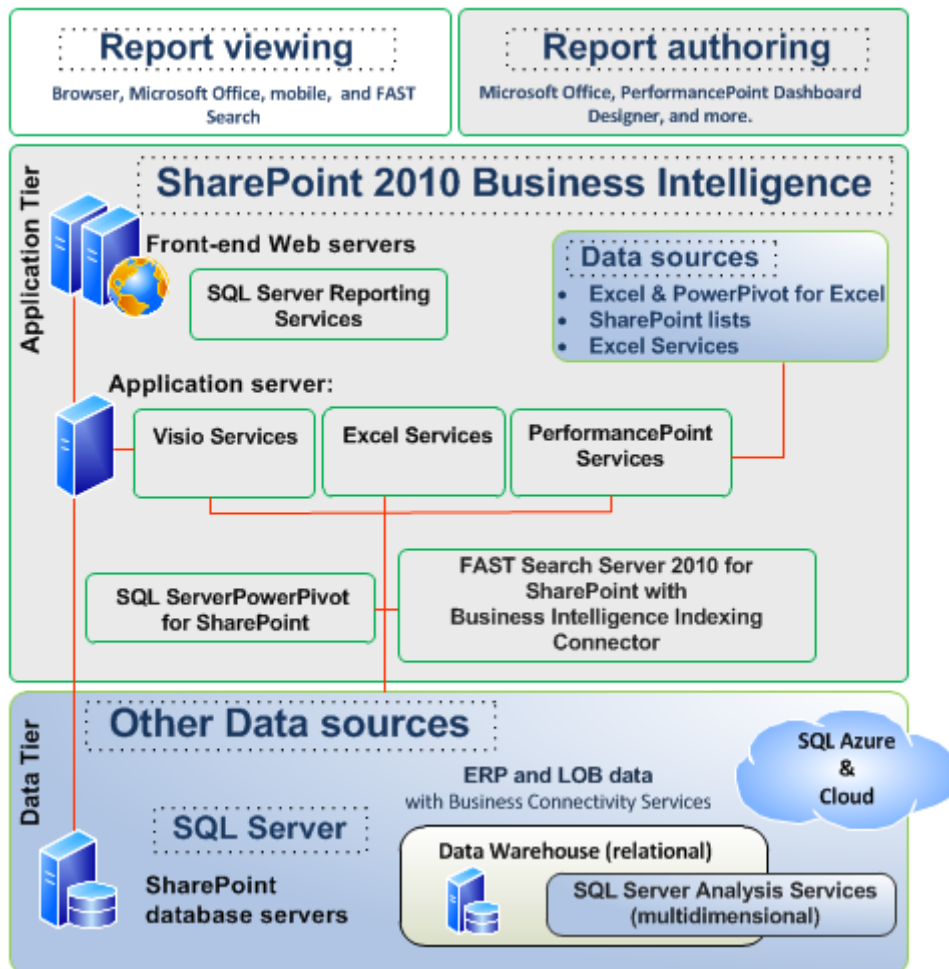
Note:

A running Secure Store Service application and Proxy are required to store the Unattended Service account password. To learn more about the unattended service account and how it uniquely applies to each service, see documentation for each service.

2. Each application service stores content in SharePoint Server by using a document library, site collection template (such as the Business Intelligence Center), or site lists in SharePoint Server for viewing reports. The Business Intelligence Center is created from an Enterprise site collection template. PerformancePoint Services is unique because it has a dashboard authoring tool that is accessed from the Business Intelligence Center or in an enabled Web site.

You can also use PerformancePoint Services to export dashboard objects such as charts, grids, scorecards, PivotTable reports to Excel 2010 or PowerPoint 2010. The resulting dashboard from authoring in PerformancePoint Dashboard Designer, can position report views from each service.

3. Report authoring for Visio and Excel ServicesExcel 2010 occurs in their respective Office client applications. SQL Server Report Builder 3.0 and PerformancePoint Dashboard Designer are client-side applications that are started from the Web.
4. Following the diagram is a partial list of possible data sources for business intelligence products or features and their respective business intelligence services. For information about additional data sources and provider types, see each product's individual documentation.



Data sources for business intelligence in SharePoint Server

Business intelligence tool	Supported SQL Server versions (32- & 64-bit)	Supported data providers	Other source data	For more information, see
Excel	SQL Server 2005 SQL Server 2008 SQL Server 2008 R2		Excel worksheet data	
PerformancePoint Services	SQL Server 2005 SQL Server 2008 SQL Server 2008 R2	For SQL Server, extends System.Data.SqlClient For SSAS, uses ADOMD.NET	SQL Server Reporting Services reports Visio Graphics Service reports SharePoint lists Can import SQL Server Analysis Services (SSAS) KPIs Excel worksheet data	Planning for PerformancePoint Data Sources
Excel Services	SQL Server 2005 SQL Server 2008 SQL Server 2008 R2	For SQL Server, extends System.Data.SqlClient For SSAS, uses MSOLAP (.4 by default) OLE DB ODBC	Excel worksheet data	
Visio Services	SQL Server 2005 SQL Server SQL Server 2008 R2	For SQL Server, extends System.Data.SqlClient For SSAS, uses ADOMD.NET		

Secure Store for Business Intelligence service applications

This article describes how Microsoft SharePoint Server 2010 business intelligence features use the Secure Store Service to provide access to external data sources (such as SQL Server) for SharePoint Server 2010 users. For the purposes of this article, the SharePoint Server 2010 *Business Intelligence service applications* are:

- Excel Services
- PerformancePoint Services
- Visio Services

The SharePoint Server 2010 Business Intelligence service applications offer two methods of data access for users:

- Integrated Windows authentication using Constrained Kerberos delegation
- Secure Store Service

This article covers the Secure Store Service and its relationship to the Business Intelligence service applications. For information about using Integrated Windows authentication with Constrained Kerberos delegation, see [Configure Kerberos authentication \(SharePoint Server 2010\)](#).

Secure Store Service

Secure Store is a feature in SharePoint Server 2010 that helps provide access to data outside SharePoint Server 2010 (for example, SQL Server data) by allowing a Business Intelligence service application to use a set of credentials with data access on behalf of a SharePoint Server 2010 user who is attempting to access that data. Such use of credentials by Business Intelligence service applications on behalf of users is called *impersonation*.

Secure Store provides this mapping between Business Intelligence services applications, users, and credentials through the use of a *Target Application*. A Secure Store Target Application is a collection of metadata that specifies which users shall be allowed access to a particular set of credentials that a Business Intelligence service application will use for impersonation when accessing external data. This metadata is stored in the Secure Store database along with the credentials themselves, which are encrypted.

Secure Store Target Applications can be used in many ways within SharePoint Server 2010, but for the purposes of SharePoint Server 2010 Business Intelligence scenarios, Target Applications consist of the following settings, configurable by the Farm Administrator:

- **Administrators** Target Application Administrators are users who have privileges to administer a given Secure Store Target Application. This can be the Farm Administrator or a specific user or users, depending on your needs. For Target Applications created by PerformancePoint Services,

the Administrator is configured automatically by PerformancePoint Services and the user configuring the Unattended Service Account is added as the Administrator.

- **Members** The Members of a Target Application are the users on behalf of whom the Business Intelligence Service Application will impersonate the Target Application Credentials when it accesses external data. This could be a single user, multiple users, or an Active Directory group. Members are also referred to as *Credential Owners*. For Target Applications created by PerformancePoint Services, the service account used by the PerformancePoint Services application pool is used as the Member.
- **Credentials** Target Application Credentials consist of an Active Directory account with direct access to data sources. (You must grant the required data access to this account directly — access to external data sources is not controlled by SharePoint Server 2010. This should be a low privileged account that only allows data access.) It is this account that is impersonated by Business Intelligence service applications to give users access to data.

The Administrators, Members, and Credentials are configurable by the Farm Administrator directly through Secure Store for Excel Services and Visio Services. For PerformancePoint Services, these values are configured through the **PerformancePoint Service Application Settings** and should not be modified through Secure Store.

Visio Services and Excel Services can use Secure Store using one of two methods:

- **Specified Target Application** A specific Target Application is specified by the Excel worksheet or the Visio Web drawing. When a user accesses the worksheet or Web drawing, Secure Store uses the credentials associated with that Target Application for data access. For Visio Services, this Target Application must be specified using an ODC file that is hosted on SharePoint Server 2010.
- **No specified Target Application (Unattended Service Account)** No Target Application is specified by the Excel worksheet or the Visio Web drawing. When a user accesses the worksheet or Web drawing connected to an external data source, Secure Store uses the Target Application specified in the Global Settings of Excel Services or Visio Services. When a Target Application is specified globally for a Business Intelligence service application, the Target Application Credentials are referred to as the *Unattended Service Account*.

PerformancePoint Services cannot specify a specific Secure Store Target Application — it can only use Secure Store with the Unattended Service Account.

The basic sequence of events that occurs is as follows:

1. A SharePoint Server 2010 user accesses a data-connected object such as an Excel Services worksheet, Visio Services Web drawing, or PerformancePoint Services dashboard.
2. If the object is configured to use Secure Store for data authentication, the Business Intelligence Service Application calls the Secure Store service to access the Target Application specified by the object.
3. If the user is a Member of that Target Application, the credentials stored in the Target Application are returned and the Business Intelligence Service Application impersonates the credentials while accessing the data.
4. The data is displayed to the user within the context of the worksheet, Web drawing, or dashboard.

Data connection files

All of the Business Intelligence service applications can use data connection files to specify authentication information. Excel Services and Visio Services use Office Data Connection (.ODC) files and PerformancePoint Services uses PerformancePoint Services Data Connection (.PPSDC) files. Use of such files allows multiple Excel Services worksheets, Visio Services Web drawings, or PerformancePoint Services dashboards to share a common set of data access parameters.

The SharePoint Server 2010 Business Intelligence service applications each use data connection files differently. For a description of how each uses data connection files, see the section for each service application, below.

The Unattended Service Account

Unattended Service Account refers to the credentials of a Secure Store Target Application that is specified in the global settings of a Business Intelligence service application. This Target Application is used to provide data access to users when another authentication method is not specified. For Visio Services, the Unattended Service Account is required any time that Integrated Windows authentication is not used, even if additional connection information is provided in the connection file (for example, a SQL Authentication string).

Data access from client and server

Microsoft Excel 2010 and Microsoft Visio 2010 are client applications that function independently from SharePoint Server 2010. Though they can publish documents to SharePoint Server 2010, they cannot use Secure Store directly for authentication to data sources. When you create or edit a data-connected worksheet or Web drawing, you must use Integrated Windows authentication or another applicable authentication method to connect directly to a data source from Excel 2010 or Visio 2010. (Other authentication methods you might use include SQL Authentication or an OLEDB connection string.) Once the worksheet or Web drawing is published to SharePoint Server 2010, Excel Services or Visio Services can use Secure Store to connect to the data source when displaying the content to a user.

PerformancePoint Services Dashboard Designer is directly integrated with SharePoint Server 2010. Dashboard Designer can use Secure Store directly to authenticate using the Unattended Service Account. As a result, users of Dashboard Designer do not need direct access to data sources through Integrated Windows authentication, provided the Unattended Service Account has the required access.

Excel Services and Visio Services

Excel Services and Visio Services use Secure Store similarly:

- Both can store a Secure Store Target Application that is specified in an ODC file.
- Both can use the Unattended Service Account.

However, there are some key difference between Excel Services and Visio Services, discussed in the sections that follow.

Excel Services

The data connections used by Excel Services must be configured in Excel 2010 prior to publication to a SharePoint Server 2010 site. An Excel 2010 worksheet can specify data connection information directly or it can include a pointer to an ODC file where connection information can be found.

The following authentication settings are available within a data-connected Excel 2010 workbook or ODC file:

- **Integrated Windows authentication** Specifies Integrated Windows authentication with Kerberos delegation to authenticate each individual user when viewing an Excel 2010 workbook through Excel Services.
- **SSS ID** Designates a specific Secure Store Service Target Application to be used for data source access.
- **None** Uses the credentials specified in the connection string, if any; otherwise it uses the Secure Store Unattended Service Account designated in the Excel Services global settings.

These settings can only be edited by opening the worksheet or ODC file in Excel 2010.

Visio Services

Visio Services supports two methods of data connection for Visio Web drawings:

- Embedded connection information
- External connection information that uses an ODC file

When you create a Visio diagram and connect it directly to a data source, Visio 2010 stores the data source information directly in the file when you publish the Web drawing to SharePoint Server 2010. When a user views the Web drawing, Visio Services connects to the data source using the Secure Store Unattended Service Account specified in the Visio Services global settings.

If, instead of connecting directly to a data source from Visio 2010, you connect to a data source using an existing ODC file stored on SharePoint Server 2010, Visio 2010 maintains the link to that ODC file when you publish the Web drawing. Visio Services then uses the connection information stored in the ODC file when it connects to the data source. This includes using a specific Secure Store Target Application if one is specified in the ODC file.

Visio 2010 cannot edit ODC files. We recommend that you do as follows to use an ODC file with a Visio Web drawing: Create the ODC file in Excel 2010, publish it to SharePoint Server 2010, and then connect to it as a data source from Visio 2010 when you create a new data-connected diagram. You must use Excel 2010 to edit ODC files if you want to change the data query, authentication information, specify a Target Application, or modify other settings.

Visio Services cannot parse complex SQL queries. If you attempt to use an ODC file containing a complex query, Visio Services may be unable to run the query and retrieve the data.

PerformancePoint Services

PerformancePoint Services only makes use of Secure Store through the Unattended Service Account. The choice between Integrated Windows authentication and the Unattended Service Account is made through Dashboard Designer when you create or edit a data source.

The Secure Store Target Application for the PerformancePoint Services Unattended Service Account is configured as part of the PerformancePoint Services service application settings by an administrator. While this Target Application appears on the Secure Store Target Applications list, it should not be modified directly through Secure Store.

Summary of differences

As described in this article, each of the Business Intelligence service applications makes use of Secure Store in a different way. The following table summarizes the Secure Store functionality and options for each Business Intelligence service application.



Note:

Each Business Intelligence service application supports Integrated Windows authentication. If Integrated Windows authentication is specified, the Secure Store options are not used.

Service application	Secure Store	Data connections
PerformancePoint Services	Unattended Service Account only.	Always made by using a PPSPDC file.
Excel Services	Secure Store Target Application can be specified in ODC file or embedded in XLSX file. When no Target Application is embedded or specified in an ODC file, the Unattended Service Account is used.	Embedded in spreadsheet or specified in an ODC file. ODC files must be edited in Excel 2010.
Visio Services	Secure Store Target Application can be specified in ODC file. When no ODC file is used or when ODC file does not specify a Target Application, the Unattended Service Account is used. Anytime non-Integrated Windows authentication is used, the unattended account is	Embedded in Web drawing or specified in an ODC file. Limited support for complex queries. ODC files must be edited in Excel 2010. (Visio 2010 cannot edit ODC files.)

Service application	Secure Store	Data connections
	required except if the ODC file specifies a different target application.	

See Also

[Configure the Secure Store Service \(SharePoint Server 2010\)](#)

[Excel Services overview \(SharePoint Server 2010\)](#)

[Configure Secure Store Service for Excel Services](#)

[PerformancePoint Services overview \(SharePoint Server 2010\)](#)

[Plan for PerformancePoint Services security \(SharePoint Server 2010\)](#)

[Plan for Visio Services \(SharePoint Server 2010\)](#)

Overview of documentation for SQL Server Reporting Services reports in SharePoint

Microsoft SQL Server 2008 Reporting Services (SSRS) is a server-based reporting platform that provides a full range of ready-to-use tools and services to help you create, deploy, and manage Reporting Services reports for your organization. This article gives an overview of SQL Server Reporting Services in SharePoint integrated mode and has links to articles in SQL Server Books Online to help you plan and configure. After you install the Reporting Services Add-in on Microsoft SharePoint Server and configure the two servers for integration, you can upload or publish report server content types to a SharePoint library and then view and manage those documents from a SharePoint site.

Reporting Services deploys in the following two modes.

- *Native mode* describes a report server that runs as an application server alone, without sharing the content database with SharePoint Server. Reports can be managed and viewed through the Report Manager Web application. SharePoint Web Parts can be enabled so you can select and view reports from a report server.
- *SharePoint integrated mode* means that a report server is integrated with a SharePoint Server farm and requires you to configure the report server in integrated mode and that you download and configure an add-in component on each of your SharePoint front-end Webs. Integrated mode allows the user to manage and view reports from SharePoint document libraries.

Overview of Reporting Services in SharePoint integrated mode

The following articles describe improvements to SQL Server 2008 that relate to business intelligence.

- [What's New \(Reporting Services in SharePoint Integrated Mode\)](#) Discusses new features in Reporting Services and improvements for the Reporting Services environment that is integrated with a SharePoint product. The new features include support for multiple SharePoint Zones, the SharePoint Universal Logging service, and Report Parts. There is also a new SharePoint List data extension that has query designer support.
- [Overview of Reporting Services and SharePoint Technology Integration](#) Explains that when you integrate a report server that uses an instance of one of the Microsoft SharePoint 2010 Products, items and properties are stored in the SharePoint content databases. The article provides a description of the databases that contain the content for the report server and SharePoint Server 2010.

Planning and architecture for Reporting Services in SharePoint integrated mode

The following articles describe SQL Server Reporting Services integration with SharePoint 2010 Products and how to plan an implementation.

- [Deployment Topologies for Reporting Services in SharePoint Integrated Mode](#) Describes different deployment scenarios and what is supported for integration between SharePoint 2010 Products and Reporting Services.
- [Requirements for Running Reporting Services in SharePoint Integrated Mode](#) Specifies edition and software requirements.
- [Storing and Synchronizing Report Server Content With SharePoint Databases](#) Shows how the SharePoint configuration and content databases synchronize with the report server database to store content and metadata.
- [How Do I Learn About SharePoint Integrated Mode](#) Provides a roadmap for SQL Server 2008 Reporting Services integration with SharePoint Server 2007. The roadmap includes multiple articles about planning and architecture, installation and configuration, Web Parts, programmability for SharePoint integrated mode, and more.

Configuration for Reporting Services in SharePoint integrated mode

The following articles describe steps to configure Reporting Services in SharePoint integrated mode.

- [Configuring Reporting Services for SharePoint 2010 Integration](#) Gives the step-by-step instructions for deployment and configuration scenarios for Reporting Services for SharePoint 2010 integration. Several "how-to" articles follow.
- [Default Configuration for SharePoint Integrated Mode \(Reporting Services\)](#) Refers to a Reporting Services installation where Setup installs and configures a report server instance and creates the report server database in a format that supports content storage.

Overview of PowerPivot documentation (SharePoint Server 2010)

This article maps to articles to show you how to plan, configure, and maintain Microsoft SQL Server 2008 R2 PowerPivot for Microsoft Excel 2010 and PowerPivot for SharePoint 2010. The name *PowerPivot* refers to a collection of applications and services that provide an end-to-end approach for creating data-driven, user-managed business intelligence solutions in Microsoft Excel workbooks. PowerPivot client and server applications support self-service business intelligence that puts powerful analytics within your reach, enabling you to find better information and insight into the numbers that drive decisions, objectives, and initiatives throughout the organization. The following is a brief description of PowerPivot components for Microsoft Office.

- PowerPivot for Excel 2010 is an add-in to Excel that provides tools for adding and integrating large amounts of data in Excel workbooks.
- PowerPivot for SharePoint 2010 extends SharePoint 2010 Products and Excel Services to add server-side processing, collaboration, and document management support for the PowerPivot workbooks that you publish to SharePoint 2010 Products.

Overview of PowerPivot for Excel and SharePoint

PowerPivot integrates with Excel and SharePoint 2010 Products. In an Excel environment, PowerPivot for Excel provides a familiar authoring and analytical experience on the workstation. In a SharePoint 2010 Products farm, PowerPivot for SharePoint is the set of server-side applications, services, and features that support team collaboration on business intelligence data.

The following articles provide more detail about PowerPivot features.

- [PowerPivot Overview](http://go.microsoft.com/fwlink/?LinkId=190359) (http://go.microsoft.com/fwlink/?LinkId=190359) Further explains what PowerPivot for Excel 2010 and PowerPivot for SharePoint 2010 are, shows a diagram that illustrates how a request for query processing moves through the farm, and explains who should use PowerPivot.
- [PowerPivot Features](http://go.microsoft.com/fwlink/?LinkId=190360) (http://go.microsoft.com/fwlink/?LinkId=190360) Gives an in-depth description of features for PowerPivot for Excel 2010 and PowerPivot for SharePoint 2010.
- [PowerPivot Concepts](http://go.microsoft.com/fwlink/?LinkId=190361) (http://go.microsoft.com/fwlink/?LinkId=190361) Explains the major concepts of PowerPivot for Excel 2010 and PowerPivot for SharePoint 2010.

Planning and architecture for PowerPivot in SharePoint and Excel Services

The following articles help you understand PowerPivot and how to plan implementation.

- [Plan a PowerPivot Deployment in a SharePoint Farm](http://go.microsoft.com/fwlink/?LinkId=190362)
(http://go.microsoft.com/fwlink/?LinkId=190362) Gives considerations and architecture to add PowerPivot to stand-alone server and to a server farm.
- [Plan for PowerPivot Integration with Excel Services](http://go.microsoft.com/fwlink/?LinkId=190363)
(http://go.microsoft.com/fwlink/?LinkId=190363) Explains how the server systems work together, how connections are established, and describes configuration settings that enable seamless interaction.
- [Plan for PowerPivot Integration with Reporting Services](http://go.microsoft.com/fwlink/?LinkId=190365)
(http://go.microsoft.com/fwlink/?LinkId=190365) Tells how you can build Reporting Services reports as a way to visualize PowerPivot data. Reporting Services offers rich data visualization through charts and maps, additional rendering formats, and comprehensive subscription and delivery options.

Deployment for PowerPivot in Excel Services and SharePoint 2010 Products

Deploying an instance of PowerPivot for SharePoint 2010 consists of installation and configuration tasks that result in an operational server that is immediately available to users in your organization. The following articles give detailed instructions for deploying PowerPivot for SharePoint.

- [Hardware and Software Requirements \(PowerPivot for SharePoint\)](http://go.microsoft.com/fwlink/?LinkId=190366)
(http://go.microsoft.com/fwlink/?LinkId=190366) Gives details for hardware and software for the client workstation, server installation, Excel Services, and SharePoint 2010 Products.
- [Installation \(PowerPivot for SharePoint\)](http://go.microsoft.com/fwlink/?LinkId=190367) (http://go.microsoft.com/fwlink/?LinkId=190367) Links to considerations for installation and "how-to" articles for installing PowerPivot in various ways. The articles include information about how to scale out PowerPivot for SharePoint 2010 Products and OLE DB and ADOMD.NET.
- [Default Configuration for PowerPivot for SharePoint](http://go.microsoft.com/fwlink/?LinkId=190368)
(http://go.microsoft.com/fwlink/?LinkId=190368) Explains installation procedures for the default configuration for PowerPivot for SharePoint 2010.
- [Configure PowerPivot for SharePoint](http://go.microsoft.com/fwlink/?LinkId=190369) (http://go.microsoft.com/fwlink/?LinkId=190369) Links to "how-to" articles for configuring PowerPivot for SharePoint 2010.

Data warehousing, OLAP, and Analysis Services for SharePoint 2010

This article describes data warehouses, OLAP, and Microsoft SQL Server Analysis Services (SSAS). It discusses how a data warehouse and SQL Server Analysis Services relate to the business intelligence application services in Microsoft SharePoint Server 2010. Additionally, the article discusses when you would use PowerPivot rather than SSAS.

Overview of data warehousing, OLAP, and PowerPivot and relation to SharePoint 2010

What is a data warehouse? A *data warehouse* is a database that functions as a repository for storing and analyzing numeric information. Core data in the data warehouse are typically numeric values that can be summarized or aggregated and are stored in a different structure than a typical transactional database structure. One reason a database warehouse structure differs from a transactional database structure is that pulling data can otherwise be very resource-expensive. Data warehouses enable you to store aggregated data instead of performing time and resource-sensitive ad-hoc queries to return summed values, as you would perform in a transactional database to create a report. This simplified definition is explained better in many books written for data warehouse professionals.

What is OLAP and how does it relate to a data warehouse? The term *online analytical processing* (OLAP) usually refers to specialized tools that make warehouse data easily available. An OLAP cube is a logical structure that defines the metadata. The term *cube* describes existing measure groups and dimension tables and should not be interpreted as having limited dimensions. A cube is a combination of all existing measure groups. A *measure group* is a group of measures that match the business logic of the data and is another logical structure that defines metadata so that client tools can access the data. Each measure group contains the detail values that are stored in the fact table (copied or dynamically retrieved values). OLAP cubes contain lots of metadata; metadata in its simplest definition is data about data. Multidimensional expressions, or MDX, is a metadata-based query language that helps you query OLAP cubes.

What is SQL Server Analysis Services (SSAS) and how does it relate to OLAP? Microsoft SQL Server Analysis Services (SSAS), formerly known as OLAP Services, provides server technologies that help speed up query and reporting processing. Analysis Services implements OLAP with technologies that simplify and quicken the process of designing, creating, maintaining, and querying aggregate tables while avoiding data explosion issues.

How do PerformancePoint Services and Excel Services relate to data warehouses, OLAP, or SSAS? Complex queries on OLAP cubes can produce business answers much faster than the same query on OLTP relational data. The data structures are different and used for different purposes. SSAS OLAP cubes are better for aggregating and reporting on data. SQL Server Analysis Services data

supplies business intelligence authoring tools such as Microsoft Excel, PerformancePoint Dashboard Designer, and Visio with an OLAP data source.

What is PowerPivot and how does it relate to SSAS? Microsoft SQL Server 2008 R2 PowerPivot for Microsoft Excel 2010 is an extension to Microsoft Excel that adds support for large-scale data. It has an in-memory data store as an option for SQL Server Analysis Services. Multiple data sources that can be merged include corporate databases, worksheets, reports, and data feeds. PowerPivot data that is inside an Excel workbook is detected, extracted, and processed separately on Analysis Services server instances within the farm. Then Excel Services in SharePoint gives the presentation layer in a browser window. For more information, see www.powerpivot.com.

When do I use PowerPivot versus SSAS? SSAS is an OLAP engine available for IT professionals to build sophisticated, high-performance solutions to deploy across the organization. Similarly to Excel, PowerPivot for Excel is for the information workers that build BI solutions for themselves instead of for the organization. The PowerPivot file can then be published to SharePoint Server or SharePoint Foundation for the team. To learn more about the differences, see the PowerPivot Team Blog post [Comparing Analysis Services and PowerPivot](http://go.microsoft.com/fwlink/?LinkId=192047) (<http://go.microsoft.com/fwlink/?LinkId=192047>).

Excel Services overview (SharePoint Server 2010)

Microsoft Excel 2010 is designed to help you analyze business data and increase business intelligence. Excel Services in Microsoft SharePoint Server 2010 is a Microsoft SharePoint Server shared service that you can use to publish Microsoft Excel 2010 workbooks on SharePoint Server. The published workbooks are available throughout your organization for knowledge workers to use. Any published workbook can be managed and secured according to your organizational needs and then shared throughout.

With business intelligence, you can store data that represents your organization's key business processes, to organize that data in a useful manner, and to present that data as meaningful information. Knowledge workers can act on that information to increase productivity and to provide feedback that improves underlying business processes.

What is Excel Services?

Excel Services supports sharing, securing, managing, and using Excel 2010 workbooks in a SharePoint Server Web site or document library. Excel Services consists of the Excel Calculation Services (ECS), Microsoft Excel Web Access (EWA), and Excel Web Services (EWS) components. These three components interact with SharePoint Server and Excel Services to contribute to organizational business intelligence processes. Essentially, Excel 2010 is an authoring tool and Excel Services is a reporting tool.

There are two primary interfaces in Excel Services; a Web-based UI that lets users view workbooks and spreadsheets in a browser, and a Web services interface for programmatic access.

Looking at a number of specific scenarios can help you understand how best to leverage Excel Services:

1. **Sharing spreadsheets through the browser** Users can save Excel 2010 spreadsheets to a SharePoint Server document library to give other users browser-based access to the server-calculated version of the spreadsheet. When the spreadsheet is accessed Excel Services loads the spreadsheet, refreshes the external data if needed, calculates it if necessary, and sends the resulting output view back through the browser. A user does not need to have Excel 2010 installed to view the spreadsheet. Users will always view the latest version of a spreadsheet, and they can interact with it in a browser; security permissions can be set to limit what access is provided to which user.
2. **Building business intelligence (BI) dashboards** A browser-based dashboard can be created using Excel and Excel Services without a single line of code.
3. **Reuse of logic encapsulated in Excel spreadsheets in custom applications** Besides a browser-based interface with the server, Excel Services provides a Web-service-based interface so

a published spreadsheet can be accessed programmatically by any application that uses Web services. The Web service applications can change values, calculate the spreadsheet, and retrieve some or all of the updated spreadsheet using that interface according to what security permissions have been set for the published spreadsheet.

4. **Report Building** One of the most useful features of Excel Services is report building. Reports can include business intelligence data or any type of data you are working with in Excel Services. Generating and publishing Excel Services reports is also one of the basic functions of a dashboard. Excel Services reports function very much like Excel Services workbooks and can be similar in appearance, whether you view an Excel Services workbook on your computer or in a SharePoint Server document library. Excel Services reports can include a variety of standard Excel Services features and functionality, such as conditional formatting, formulas, and charts. When you publish an Excel workbook to Excel Services, your workbook becomes the data source for an Excel Services report type in the Dashboard Designer. Creating any type of Excel Services report is a two step process. You begin by using the wizard to create the basic report structure. Then you select the data to display in the report.

See Also

[Plan Excel Services data providers \(SharePoint Server 2010\)](#)

[Plan Excel Services authentication \(SharePoint Server 2010\)](#)

[Overview of Excel Services architecture](#)

[Plan Excel Services data sources and external connections](#)

Overview of Excel Services architecture

This topic summarizes Excel Services Application architecture. Excel Services Application is built on the Microsoft SharePoint 2010 Products platform. Consequently, Excel Services Application uses Microsoft SharePoint 2010 Products authentication and authorization.

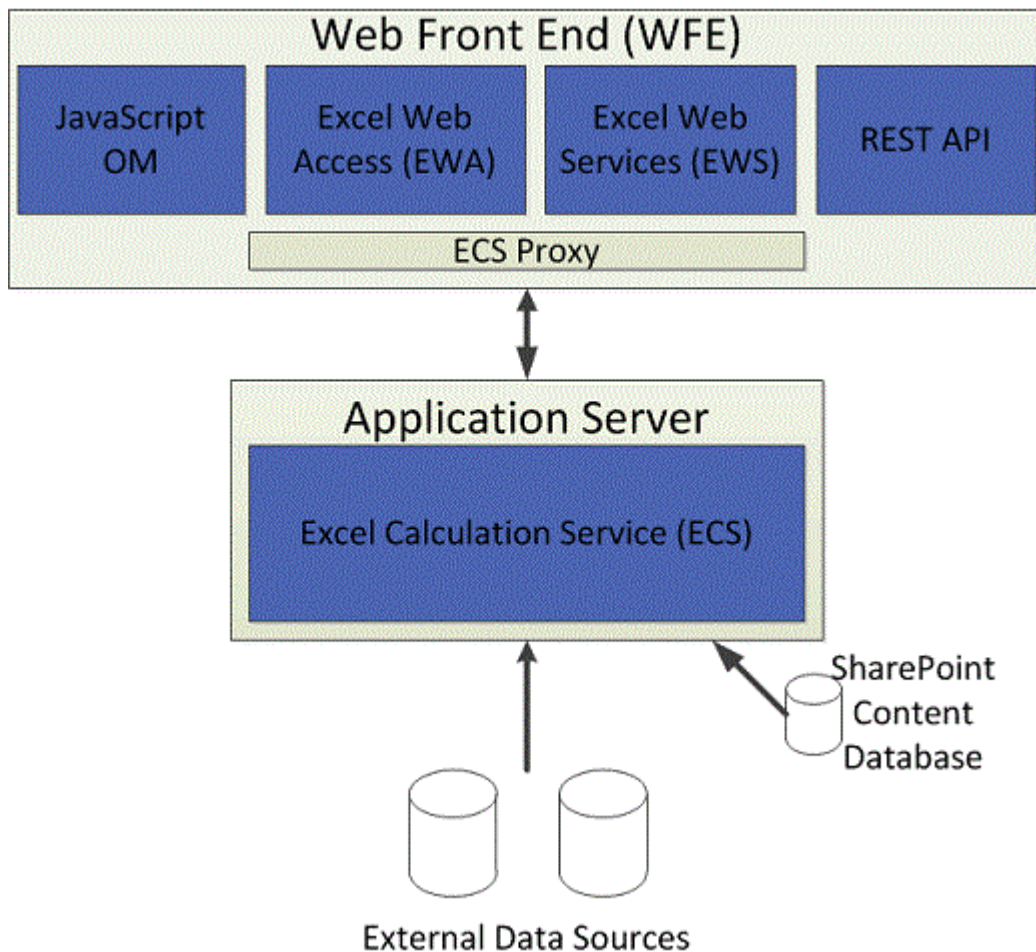
Excel Services Components

Excel Services in Microsoft SharePoint Server 2010 resides on a stand-alone SharePoint Server (for evaluation/test environments) or in a SharePoint farm, a typical production environment. Both the SharePoint Server 2010 and Excel Services architecture is designed to meet the deployment needs ranging from a department setup inside an organization to a global enterprise scenario. Excel Services includes three core components:

- **Excel Calculation Services** is the main Excel Services component that loads the spreadsheet and workbook, calculates the spreadsheets, updates external data, and maintains session state for interactivity.
- **Excel Web Access** is a Web front end component that delivers the Excel workbooks.
- **Excel Web Services** is a Web Service hosted in SharePoint that provides various methods for developers to create custom applications that are built on the Excel workbook.

These three Excel Services components reside either on the Web front end server or on the back end application server. A standard Excel Services production environment that has many users uses two or more Web front end and application servers. A test or development environment typically hosts all Excel Services components on a single server.

The following diagram shows the Excel Services basic 2010 architecture as related to the Microsoft SharePoint 2010 Products content database and the external data sources.



Performance and Scalability

Scalability for an Excel Services Application production environment is achieved by adding more memory or higher powered processors to the servers. Also, more Web front end servers or application servers can be added to the farm to address any instances that relate to server loads and performance. You can add more Web front end and application servers independently. For example, you can add many Web front end servers and use a single application server or add many applications servers and only deploy one or two Web front end servers.

If you have more large workbooks, or if your workbooks contain many calculations and use lots of external data that you may want to add more application servers to your farm. If you have basic or simple workbooks and many users view them or the workbooks contain many charts, or on the other hand, if you have dashboards with many workbooks, you might consider adding more Web front end servers to your farm.

The largest Excel Services performance factor depends on the type, size of the workbooks, and external data connections in the workbooks used with Excel Services. See [Plan Excel Services authentication \(SharePoint Server 2010\)](#) for more information about how to configure the Excel Services settings.

Plan Excel Services data sources and external connections

To configure Microsoft SharePoint 2010 Products to enable workbooks loaded on Excel Services Application to successfully refresh external data, you must understand the relationships and dependencies between SharePoint Server 2010 and Excel Services Application.

This article contains guidance to help you configure the following SharePoint Server 2010 application server components:

- Excel Services
- Secure Store Service

In this article:

- [Connections and Excel workbooks](#)
- [Data providers](#)
- [Authentication to external data](#)
- [Data connection libraries and managed connections](#)
- [Excel Services security and external data](#)

Connections and Excel workbooks

Every Excel workbook that uses external data contains a connection to a data source. Connections consist of everything that is required to establish communications with, and retrieve data from, an external data source. This includes the following:

- A connection string (a string that specifies which server to connect to and how to connect to it).
- A query (a string that specifies what data to retrieve).
- Any other specifics required to get the data.

Embedded and linked connections

Excel workbooks can contain embedded connections and linked connections. Embedded connections are stored internally as part of the workbook. Linked connections are stored externally as separate files that can be referenced by a workbook.

Embedded and linked connections function the same way. Both will correctly specify all the required parameters to connect to data successfully. Linked connection files can be centrally stored, secured, managed, and reused. They are a good choice when planning an overall approach to getting a large group of users connected to external data. For more information, see [Data connection libraries and managed connections](#).

For a single connection, a workbook can have both an embedded copy of the connection information and a link to an external connection file. The connection can be configured to always use an external connection file to refresh data from an external data source. In this example, if the external connection file cannot be retrieved, or if it does not establish a connection to the data source, the workbook cannot retrieve data. If the connection is not configured to use only an external connection file, Excel attempts to use the embedded copy of a connection. If that fails, Excel attempts to use the connection file to connect to the external data source. The ability to specify that only connection files can be used to establish a communications link to an external data source provides support for the managed connection scenarios described in [Data connection libraries and managed connections](#).

Excel Services can use connections coming from an external connection file and connections that are embedded in the workbooks. There are some restrictions for external connection files. For more information, see [Excel Services security and external data](#). If both kinds of connections are allowed on the server, the behavior is the same as the Excel behavior described earlier.

For security purposes, Excel Services Application can be configured to enable only connections from connection files. In this configuration, all embedded connections are ignored for workbooks loaded on the server, and connections are tried only when there is a link to a valid connection file that is trusted by the server administrator. For more information, see [Trusted data connection libraries](#).



Note:

There are many kinds of connection files, and Excel Services Application works only with Office data connection files (.odc).

Data providers

Data providers are drivers that client applications (such as Excel and Excel Services Application) use to connect to specific data sources. For example, a special MSOLAP data provider is used to connect to Microsoft SQL Server 2008 Analysis Services. The data provider will be specified as part of the connection in the connection string. It is not necessary to have extensive knowledge about data providers for the purposes of this article. But you do have to understand the following concepts:

- Data providers are typically well-tested, stable sets of libraries that can be used to connect to external data.
- Any data provider that is used by Excel Services must be explicitly trusted by the server administrator. For information about how to add a new data provider to the trusted providers list, see [Manage Excel Services connections](#).



Note:

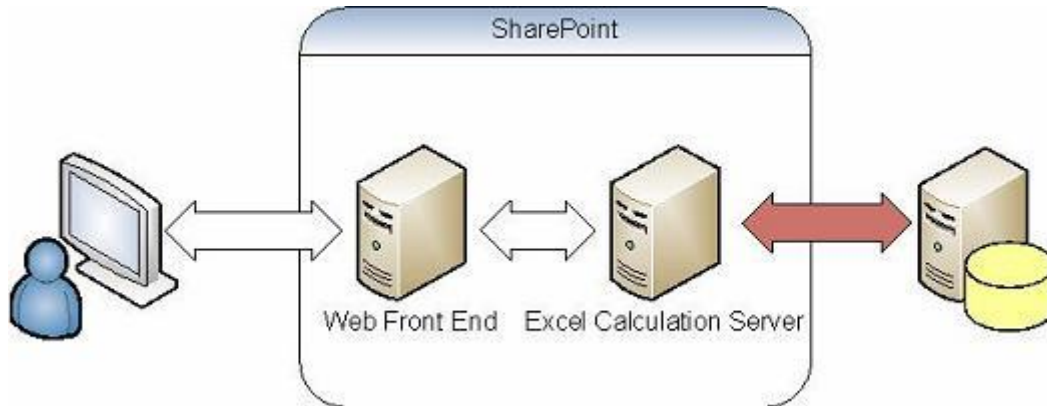
By default, Excel Services Application trusts many well-known and stable data providers. In most cases, you do not have to add a new data provider. Data providers are typically added for custom solutions.

- Data providers handle queries, parsing connection strings, and other connection-specific logic. This functionality is not part of Excel Services Application. Excel Services Application cannot control how data providers behave.

Authentication to external data

Data servers require that a user be authenticated, that is, identifying oneself to the server. The next step is authorization, communicating to the server the permitted actions associated with the user. Authentication is required for the data server to perform authorization, or to enforce security restrictions that prevent data from being exposed to anyone other than authorized users.

Excel Services Application has to communicate to the data source which user is requesting the data. In most scenarios, this is going to be the user viewing an Excel report in a browser. This section explains authentication between Excel Services Application and an external data source. Authentication at this level is shown in the following diagram. The arrow on the right side shows the authentication link from an application server that runs Excel Calculation Services to an external data source.



Note:

Excel Services Application accesses external data sources by using a delegated Windows identity. Consequently, external data sources must reside within the same domain as the SharePoint Server 2010 farm or Excel Services Application must be configured to use the Secure Store Service. If the Secure Store Service is not used and external data sources do not reside within the same domain, authentication to the external data sources will fail. For more information, see [Planning considerations for services that access external data sources](#) in “Services Architecture Planning.”

There are many ways to implement authentication. This article focuses on three methods that Excel Services Application supports:

- Integrated Windows authentication
- Secure Store Service
- None

Excel Services Application determines the kind of authentication based on a property of the connection. This property must be explicitly set, and can be set by using Microsoft Excel 2010 client. If the authentication type is missing, the default of Windows authentication is tried.

Integrated Windows authentication

SharePoint Server 2010 uses claims based authentication. Therefore, Excel Services Application also uses claims based authentication. Integrated Windows authentication is now used exclusively for IIS Authentication settings in SharePoint Server 2010. Also note that when Excel Services Application connects to a data source that is hosted on a different server than the server hosting Excel Services Application, Kerberos constrained delegation must be configured. In other words, when Excel Services Application connects to an external data source, you must configure and deploy Kerberos constrained delegation.

This method uses your Windows user identity to authenticate against a data source. For the purposes of this article, it is not important to know the specific mechanism that the operating system uses to do this (such as NTLM or constrained delegation). Windows authentication is typically the default method for external data access when you are using an Excel client to connect to data sources, such as SQL Server 2008 Analysis Services.

In most enterprise environments, Excel Services Application will be set up as part of a farm with the front-end web server, back-end Excel Services Application server, and data source, all running on different computers, as depicted in the diagram in [Authentication to external data](#). This means that delegation, or Kerberos protocol, (constrained delegation is recommended) will be required to enable data connections that use Windows authentication. This is because delegation is required to ensure that user identities can be communicated from computer to computer in a trusted and secure way. In a farm deployment, these kinds of connections will not work on Excel Services Application unless Kerberos protocol is configured correctly. See the Microsoft Download Center page, [Configuring Kerberos Authentication for Microsoft SharePoint 2010 Products](#) for more information about how to configure Kerberos constrained delegation for Excel Services Application.

Secure Store Service

SharePoint Server 2010 uses Secure Store Service authentication by including a Windows service and a secure credentials database. Excel Services Application supports the pluggable Secure Store Service functionality with which you can implement your own Secure Store Service provider. SharePoint Server 2010 provides a default Secure Store Service provider with Excel Services Application.

Secure Store Service is a centralized database that frequently is used to store credentials (a user ID and associated password pairing) that can be used by applications to authenticate to other applications. In this case, Excel Services Application relies on Secure Store Service to store and retrieve credentials for use in authenticating to external data sources.

Each Secure Store Service entry contains an application ID that serves as a lookup that is used to retrieve the appropriate set of credentials. Each application ID can have permissions set so that only specific users or groups can access the credentials that are stored for that application ID.

When provided with an application ID, Excel Services Application retrieves the credentials from the Secure Store database for the user who is accessing the workbook (either through the browser, or using Excel Web Services). Excel Services Application then uses those credentials to authenticate to the data source and retrieve data.

**Note:**

The application ID must be specified for the connection. For information about how to specify an application ID, see [Configure Secure Store Service for Excel Services](#).

None

This authentication method means that no credential retrieval should occur or that no special action is taken for authentication for the connection. For example, Excel Services Application does not try to delegate credentials, and does not try to retrieve credentials that are stored for the user from the Secure Store database. Instead, selecting None as the authentication connects to the Secure Store database as the process account and retrieves the credentials to use. In these cases, Excel Services Application hands the connection string to the data provider and lets the provider handle authentication.

In more practical terms, this means that typically a connection string will specify a user name and password to connect to the data source. However, sometimes the connection string specifies that the integrated security must be used. That is, the Windows identity of the user or computer that is issuing the request should be used to connect to the data source. In both cases, the unattended account is impersonated first and then the data source connection is made. The connection string and the provider determine the authorization method. Additionally, authorization can be based on either the credentials found in the connection string or the impersonated unattended account's Windows identity. For more information, see [Unattended account](#).

Data connection libraries and managed connections

A data connection library is a SharePoint Server 2010 list that is designed to store connection files, which can then be referenced by Office 2010 applications, such as Excel Services Application.

Data connection libraries give customers the ability to centrally manage, secure, store, and reuse data connections.

Reusing connections

Because the data connection library is in a well-known location in SharePoint Server 2010 and displays friendly business names and descriptions, users can reuse connections that were created by other users and configure them for their own purposes. A knowledgeable information worker or data expert can create connections, and other users can reuse them without having to understand the details about data providers, server names, or authentication. The location of the data connection library can even be published to Office clients so the data connections will be displayed in Excel Services Application or in any other client application that uses the data connection library. For more information, see [Manage Excel Services connections](#).

Managing connections

Because workbooks contain a link to the file in a data connection library, if something about the connection changes (such as a server name or a Secure Store application ID), only a single connection

file has to be updated instead of potentially hundreds of workbooks. The workbooks will obtain the connection changes automatically the next time that they use that connection file to refresh from Excel or on Excel Services Application.

Securing connections

The data connection library is a SharePoint list and supports all the permissions that SharePoint Server 2010 does, including per-folder and per-item permissions. The advantage that this provides on the server is that a data connection library can become a locked-down data connection store that is highly controlled. Many users may have read-only access to it. This enables them to use the data connections. But they can be prevented from adding new connections. By using access control lists (ACLs) with the data connection library, and letting only trusted authors to upload connections, the data connection library becomes a store of trusted connections. Trusted connections are connections that are known not to contain malicious queries.

Excel Services Application can be configured to load connection files only from data connection libraries that are explicitly trusted by the server administrator, and to block loading of any embedded connections. In this configuration, Excel Services Application uses the data connection library to apply another layer of security around data connections.

Data connection libraries can even be used together with the new Viewer role in SharePoint Server 2010 that enables those connections to be used to refresh workbooks loaded on Excel Services Application. If the Viewer role is applied, users cannot access the connection file contents from a client application, such as Excel. Therefore, the connection file contents are protected but still can be used for workbooks refreshed on the server.

Excel Services security and external data

Excel Services Application has many layers of security. The following subsections address only the concepts that are directly relevant to external data access.

Trusted file locations

Excel Services Application only loads workbooks from trusted file locations. A trusted file location is basically a directory (that might include all subdirectories) that the administrator has explicitly allowed workbooks to be loaded from. These directories are added to a list that is internal to Excel Services Application. This list is known as the trusted file locations list.

Trusted locations might specify a set of restrictions for workbooks loaded from them. All workbooks loaded from a trusted location adhere to the settings for that trusted location. Here is a short list of the trusted location settings that affect external data:

- How external data can be accessed. The options for this include the following:
 - No data access allowed (default).
 - Only connection files in a SharePoint Server 2010 data connection library allowed.

- Connections embedded in workbooks allowed in addition to connection files from a data connection library.
- Whether to show the query refresh warnings or not.
- Whether to fail the workbook load if external data does not refresh when the workbook opens. This is used in scenarios where the workbook has cached data results that will change depending on the identity of the user viewing the workbook. The objective is to hide these cached results, and ensure that any user who views the workbook can see only the data that is specific to that user. In this case, the workbook will attempt to refresh on open. You can set refresh on open for each connection. If the refresh fails, the workbook is not displayed to users who cannot open it in the Excel client.



Note:

This only works if the workbook is locked down by Viewer role permissions in SharePoint Server 2010, because a user who can open the workbook directly in Excel can always see the cached data results.

- External data cache expiration times. Data is shared among many users on the server to improve scale and performance, and these cache life times are adjustable. This accommodates scenarios in which query execution should be kept to a minimum because the query might take a long time to execute. In these scenarios, the data often changes only daily, weekly, or monthly instead of by the minute or every hour.

Trusted data connection libraries

As with workbook files, Excel Services Application only loads connection files from SharePoint Server 2010 trusted data connection libraries. A trusted data connection library is a library that the server administrator has explicitly added to an internal trusted list. For information about how data connection libraries let an administrator secure and manage connection files, see [Data connection libraries and managed connections](#). For information about how to trust a data connection library for use with Excel Services Application, see [Manage Excel Services connections](#).

Trusted data providers

Excel Services Application will only use external data providers that are added to an internal trusted providers list. This is a security mechanism that prevents the server from using providers that the administrator does not trust. For information about how to trust a data provider, see [Manage Excel Services connections](#).

Unattended account

The unattended account is a special account that Excel Services Application impersonates any time that it is trying a connection where the None authentication is selected, regardless of whether Integrated Windows authentication is used by the data source. Because Excel Services Application has no control over the data provider, and does not directly parse provider-specific connection strings, it has

to lessen security threats where the identity of Excel Services Application itself can be used to connect to a data source. The unattended account is used to lessen such threats.

Excel Services Application often will run with a highly privileged account. This level of permission is incorrect for users who are only viewing data. When external data authentication is set to either **None** or **Secure Store Service**, where the Secure Store Service application ID is not storing Windows credentials, the unattended account is impersonated before it tries to connect to data. Because the unattended account is not expected to have permissions to access the data source, this prevents accidental or malicious connections to data sources in the context of a privileged account.

If the unattended account has access to the data source (when authentication type is set to **None**), a connection is successfully established using the credentials of the unattended service account. Use caution when you design solutions that intentionally use this account to connect to data. This is a single account that potentially can be used by every workbook on the server. It is possible for any user loading a workbook on Excel Services Application and setting the authentication type to **None** to view that data by using the server. In some scenarios, this might be needed. However, Secure Store Service is the preferred solution for managing passwords on a per-user or per-group basis.

Plan Excel Services data providers (SharePoint Server 2010)

Excel Services in Microsoft SharePoint Server 2010 only attempts to process connections that use trusted data providers. For most common types of data connections (such as ODBC, OLEDB, Microsoft SQL Server and Analysis Services, and OLAP), it is not necessary to add a provider because by default Excel Services trusts many standard providers. Other providers are typically only added for custom solutions.

Every Microsoft Excel workbook that uses external data contains a connection to a data source. Connections consist of everything that is required to establish communications with, and retrieve data from, an external data source. This includes:

- A connection string (a string that specifies which server to connect to and how to connect to it).
- A query (a string that specifies what data to retrieve).
- Any other specifics required to get the data.

Data providers

Data providers are drivers that client applications (such as Excel and Excel Services) use to connect to specific data sources. For example, a special MSOLAP data provider is used to connect to Microsoft SQL Server 2005 and SQL Server 2008 Analysis Services. The data provider is specified as part of the connection in the connection string. The following list shows the important concepts about data providers that you need to understand.

- Data providers are typically well-tested, stable sets of libraries that can be used to connect to external data.
- Any data provider used by Excel Services must be explicitly trusted by the server administrator.



Note:

By default, Excel Services trusts many well-known and stable data providers. In most cases, it is not necessary to add a new data provider. Data providers are typically added for custom solutions.

- Data providers handle queries, parsing connection strings and other connection-specific logic. This functionality is not part of Excel Services. Excel Services cannot control how data providers behave.

Trusted data providers

Excel Services only uses external data providers that are added to an internal list of trusted providers. This is a security mechanism that prevents the server from using providers that the administrator does not want to allow.

Plan Excel Services authentication (SharePoint Server 2010)

In this article:

- [About Excel Services security](#)
- [Plan user authentication](#)
- [Plan communication among servers](#)
- [Plan external data authentication](#)

About Excel Services security

In addition to the security requirements for deploying Microsoft SharePoint 2010 Products, you must review security considerations for a deployment that includes Excel Services Application. Microsoft SharePoint Foundation 2010 provides the platform on which SharePoint Server 2010 is built.

Excel Services Application functionality, together with SharePoint Server 2010, is the primary way to control, secure, and manage access to Excel workbooks in the enterprise. Excel Services Application is an enterprise-class application server that is designed for performance, scalability, and security. An Excel Services Application deployment provides thin rendering of — and interactivity with — workbooks, and you can easily reuse workbook components, such as charts and PivotTable reports, that can be rendered in business intelligence dashboards.

Using Excel Services Application you can take advantage of server-side Excel spreadsheet calculations for custom applications, and users can lock workbooks and to help secure private data and intellectual property. This ensures that data within your workbooks is better protected while users interacting with workbooks on a server can take full advantage of the data refresh and recalculation functionality that is provided by Excel Services Application.

Security is an important component for enabling these data rendering scenarios. You must consider many factors when planning for an environment that helps ensure the security of workbooks that are rendered on a server. You must plan for managing the security of workbooks and managing the security of the server itself. Excel Services Application gives you a significant level of precise control for the processing and displaying of Excel workbooks. You can control how workbooks are opened on the server and the specific capabilities that are enabled for each workbook.

This article summarizes the security and authentication settings for Excel Services Application and related components that you must consider when planning a deployment. In addition, this article contains prescriptive guidance for using Excel Services Application to help secure and manage access to workbooks on the server.

The security model for Excel Services Application is based on the concept that to ensure data integrity and quality, an administrator must be able to centrally manage shared resources and user access to

corporate intellectual property contained in workbooks. To do this you can use Excel Services Application to specify:

- **Trusted file locations** These are SharePoint document libraries, UNC paths, or HTTP Web sites that have to be explicitly trusted before Excel Calculation Services can access them. Excel Calculation Services opens workbooks that are stored in trusted file locations only.
- **Trusted data providers** These are external databases that Excel Calculation Services is explicitly configured to trust when it is processing data connections in workbooks. Excel Calculation Services attempts to process a data connection only if the connection is to a trusted data provider.
- **Trusted data connection libraries** These are SharePoint document libraries that contain Office data connection (.odc) files. The .odc files are used to centrally manage connections to external data sources. Instead of allowing embedded connections to external data sources, Excel Calculation Services can be configured to require the use of .odc files for all data connections. The .odc files are stored in data connection libraries, and the data connection libraries have to be explicitly trusted before Excel Calculation Services will allow workbooks to access them.

By default, cross-domain workbook and data connection access is not allowed. To allow workbooks in trusted file locations (and data connections in trusted data connection libraries) to be accessed across domains by Web Parts, Web pages, or Web services, run the Windows PowerShell cmdlets, as shown in the examples in [Manage Excel Services with Windows PowerShell](#).

The requesting Web pages and the workbooks or data connections must reside in the same farm.



Note:

When you open a workbook in Excel Calculation Services, a temporary file is stored in the %TEMP% folder of the application server that is running Excel Calculation Services.

Plan user authentication

Excel workbooks that are opened by Excel Calculation Services should be stored in the SharePoint Server 2010 content database, because SharePoint Foundation 2010 maintains an access control list (ACL) for these files. Excel Calculation Services can also open workbooks from UNC paths and HTTP Web sites. However, we recommend that you use the SharePoint Server 2010 content database for workbook storage.

Authentication for user access to a SharePoint portal site is performed by SharePoint Foundation 2010. By default, SharePoint Foundation 2010 uses Integrated Windows authentication.

In addition to the listed authentication methods, Excel Services Application also supports generic forms-based authentication. However, configuring SharePoint Foundation 2010 to use generic forms-based authentication is not discussed here.

Plan communication among servers

Claims-based authentication is the default authentication mechanism in SharePoint Server 2010. It is a Microsoft and industry standard with broad support. Claims authentication helps improve security and

authentication when you deploy farms, Office Business Applications, and SharePoint services in different environments. Excel Services Application uses claims based authentication for all deployment scenarios, whether in a single server installation or in a farm environment. Additionally, the authentication and authorization of users to all content and resources within SharePoint Server 2010 is much more secure with claims-based authentication.

Plan external data authentication

Workbooks can contain embedded direct data connections and links to data connection files that are stored in data connection libraries. On refresh, depending on the configuration of Excel Services Application, the embedded direct data connection can be used to query the data source, or the data connection library link can be used to query the .odc file. The .odc file contains data connection information and must be stored in a data connection library.

To configure Excel Services Application to process connections to external data sources, select a setting in the **External Data** section of the Excel Services Add Trusted File Location page of the SharePoint Central Administration Web application.

To configure administrative settings for Excel Services Application, see [Manage Excel Services authentication](#) for more information.

Farm deployments that have integrated connections now use SharePoint Server 2010 claims based authentication. When Excel Calculation Services retrieves connection information, credentials are designated as Stored (to be retrieved from the Secure Store Service database), Integrated, or None. All data connections with integrated credentials, now use claims based authentication for deployments that are scaled out to multiple servers. A stand-alone deployment, also uses the default claims based authentication.

Imagine a data connection in a workbook opened in an Excel Calculation Services application server that uses Stored credentials. Excel Calculation Services has to retrieve valid credentials from a (Secure Store Service authentication database. Then, it uses the credentials to authenticate against a data source, before the data connection can be established.

Excel Services Application supports three data authentication methods: Integrated Windows authentication, Secure Store Service authentication, and None.

Integrated Windows authentication

Kerberos protocol is the recommended security configuration to use with Integrated Windows authentication. Because SharePoint Server 2010 uses claims based authentication, all Excel Services Application scenarios also use claims authentication. Integrated Windows authentication is now used exclusively for IIS Authentication Settings in SharePoint Server 2010. See the Microsoft Download Center page, [Configuring Kerberos Authentication for Microsoft SharePoint 2010 Products](#) for more information about how to configure Kerberos constrained delegation for Excel Services Application.

Secure Store Service authentication

By using Secure Store Service authentication, users can access multiple system resources without having to provide authentication credentials one or more times. SharePoint Server 2010 implements Secure Store Service authentication by including a Windows service and a secure credentials database. By using the pluggable Secure Store Service functionality supported by Excel Services Application, you can implement your own Secure Store Service provider. SharePoint Server 2010 includes a Secure Store Service provider that works with Excel Services Application.

Any Secure Store Service provider that you implement with Excel Services Application should send credentials together with a credential type, whether Windows credentials or other credentials. Excel Services Application uses the Secure Store Service database to retrieve credentials for connection authentication.

Secure Store Service authentication in SharePoint Server 2010 supports individual mappings and group mappings. Secure Store Service maintains a set of credentials for the application identities (App IDs) of resources that are stored in the SharePoint Server 2010 Secure Store Service database. For individual mappings, a security layer checks user credentials against multiple individual listings for an App ID that is stored in the Secure Store Service database. Individual mappings are useful if you must have logging information about individual user access to shared resources.

For group mappings, a security layer checks group credentials for multiple domain users against a single set of credentials for a resource that is identified by an App ID that is stored in the Secure Store Service database. This also works with forms-based authentication or any other claims providers you use. Group mappings are easier to maintain than individual mappings, and performance is better.

To enable Secure Store Service functionality for SharePoint Server 2010, create a new Secure Store Service in the SharePoint Central Administration Web site. See [Configure Secure Store Service for Excel Services](#) for more information.

None

When you specify **None** as the authentication method for your Excel Services Application deployment, Excel Services Application tries to use incoming connection strings to connect to the database specified in the string. Depending on the specific database provider, the database could use the connection string to authenticate the user.

Excel Services Application does not parse connection strings to determine an authentication method. The connection strings are passed to the database provider. Connection strings can specify that Integrated Windows authentication is required. Connection strings can also contain a specific user name and password. In either case, when you specify **None** as the authentication method, Excel Services Application requires the impersonation of an unattended service account.

If the database provider determines that the connection string specifies Integrated Windows authentication, and if the database authorizes access, the connection is established by using the security context of the unattended account. If the connection string contains a user name and password, and if the database authorizes access, the connection is established by using the security context of the authorized user account.

Unattended service account

The unattended service account is a privileged account that is encrypted and secured. The Secure Store Service (SSS) stores the unattended account credentials so that Excel Calculation Services can impersonate when establishing a data connection that uses SSS credentials from an environment that is not Windows-based, or **None**, as the authentication method. If an unattended service account is not configured, data connections will fail if SSS from an environment other than Windows, or **None**, is used as the authentication method.

Impersonating the unattended account protects SharePoint Server 2010 databases, and any other data sources that Excel Services Application can directly access, from unauthorized connections by client computers that are using Excel Calculation Services to open external data connections. When an unattended service account is impersonated, the credentials associated with an Excel Calculation Services application thread cannot be used to access any other databases. Also, when an unattended service account is impersonated, external data queries are run under the security context of a low-permissions account, instead of running under the security context of an Excel Calculation Services application thread that has higher permissions.

You can configure the unattended service account either as a domain account or as a local computer account. If the unattended service account is configured as a local computer account, ensure that the configuration is identical on every application server that runs Excel Calculation Services. The credentials for the unattended account are cached on the connection and on each workbook session. Each time that a workbook is loaded that has a data connection that uses the unattended account and if the credentials are not already cached for that connection, the unattended account is obtained from the Secure Store and used. In other words, the unattended account credentials are not cached globally but are instead taken from the Secure Store as needed for each session or data connection. Restrict the permissions of the unattended service account to enable only logging on to the network. Verify that the unattended service account does not have access to any data sources or SharePoint Server 2010 databases. See [Configure Secure Store Service for Excel Services](#) for more information.

Security settings

To configure administrative settings for Excel Services Application, including security settings, open the SharePoint Central Administration Web application and access the Excel Services Settings page. See [Manage Excel Services authentication](#) for more information.

The Excel Services Settings page provides configuration settings for the following:

- **Security** Excel Services Application authentication, communication and web service settings.
- **Load Balancing** Load balancing of Excel Services Application sessions across Excel Calculation Services processes.
- **Session Management** Behavior of Excel Calculation Services sessions.
- **Memory Utilization** Allocation of memory on Excel Calculation Services.
- **Workbook Cache** Settings related to caching workbook files on disk and in memory.
- **External Data** Handling external data connections in Excel Calculation Services.

You can also use the Excel Services Settings page to configure the options for file access method and connection encryption, which have a direct effect on secure deployment.

File access method

On the Excel Services Application Settings page, in the **Security** section, under **File Access Method**, select either **Impersonation** or **Process account**.

- **Impersonation** This enables a thread to run in a security context other than the context of the process that owns the thread. Select **Impersonation** to require Excel Calculation Services to authorize users when they try to access workbooks that are stored in UNC and HTTP locations. Selecting this has no affect on workbooks that are stored in SharePoint Server 2010 databases. In most server farm deployments in which front-end web servers and Excel Calculation Services application servers run on different computers, impersonation will require constrained Kerberos delegation.
- **Process account** If Excel Calculation Services application servers are opening workbooks from UNC shares or HTTP Web sites, the user account cannot be impersonated, and the process account must be used.

Connection encryption

You can use Internet Protocol Security (IPsec) or Secure Sockets Layer (SSL) to encrypt data transmission among Excel Calculation Services application servers, data sources, client computers, and front-end web servers. To require encrypted data transmission between client computers and front-end web servers, click the **Connection Encryption** setting **Required**. **Not required** is the default setting. If you change the **Connection Encryption** setting to **Required**, the Excel Calculation Services application server will only enable data transmission between client computers and front-end web servers over SSL connections.

If you decide to require encrypted data transmission, you will have to manually configure IPsec or SSL. You can require encrypted connections between client computers and front-end web servers while enabling connections that are not encrypted between front-end web servers and Excel Calculation Services application servers.

The Manage Excel Services page also lists the Trusted File Locations, Trusted Data Providers, Trusted Data Connection Libraries, and User Defined Function Assemblies pages for Excel Services Application.

Trusted file locations

Trusted file locations are SharePoint sites, UNC paths, or HTTP Web sites from which a server that runs Excel Calculation Services is permitted to access workbooks.

In the **Location** section of the Excel Services Add Trusted File Location page, you can configure the address, the location type, and whether child libraries of trusted file locations are also trusted. By selecting **Trust Children** you can improve manageability. However, you can also create a potential

security issue by enabling subsites and subdirectories of trusted locations to be automatically trusted as soon as they are created.

In the **Session Management** section, you can configure settings to help conserve resource availability and improve Excel Calculation Services performance and security. Performance can decrease when many users have multiple Excel Calculation Services sessions open at the same time. You can control resource consumption and limit the duration of open Excel Calculation Services sessions by configuring two time-out settings for open sessions.

The **Session Timeout** setting determines the time that an Excel Calculation Services session can remain open and inactive after each user interaction. The **Short Session Timeout** setting determines how long an Excel Calculation Services session can remain open and inactive after the initial session request. The **New Workbook Session Timeout** setting determines how long an Excel Calculation Services session for a new workbook can remain open and inactive before it is shut down. You can also control the number of seconds allowed for any single session request by configuring a **Maximum Request Duration** value. By limiting how long sessions remain open, you can help reduce the risk of denial-of-service attacks.

In the **Workbook Properties** section, you can configure a maximum size of any workbook, chart or image that is permitted to be opened in an Excel Calculation Services session. Performance and resource availability can be compromised when users open extremely large workbooks. Unless you control the allowable size of workbooks running in open Excel Calculation Services sessions, you risk users exceeding your resource capacity and causing the server to fail.



Note:

If an application server that runs Excel Calculation Services fails or is shut down, all open sessions on the server are lost. In a stand-alone installation, Excel Services Application will no longer be available. This means that workbooks cannot be loaded, recalculated, refreshed, or retrieved by Excel Calculation Services. In a server farm deployment that includes multiple application servers that run Excel Calculation Services, shutting down one server does not affect open sessions that are running on other servers. Users with sessions running on a server that is shut down are prompted to reopen their workbooks. When users start a new session, they are automatically routed to active application servers that are running Excel Calculation Services.

In the **External Data** section, you can determine whether workbooks stored in trusted file locations and opened in Excel Calculation Services sessions can access an external data source. You can designate whether **Allow External Data** is set to **None**, **Trusted data connection libraries only**, or **Trusted data connection libraries and embedded**. If you select either **Trusted data connection libraries only** or **Trusted data connection libraries and embedded**, the workbooks stored in the trusted file locations can access external data sources.

External data connections can be accessed only when they are embedded in or linked from a workbook. Excel Calculation Services checks the list of trusted file locations before it opens a workbook. If you select **None**, Excel Calculation Services will block any attempt to access an external data source. If you manage data connections for many workbook authors, consider specifying **Trusted**

data connection libraries only. This ensures that all data connections in all of the workbooks generated by authenticated workbook authors have to use a trusted data connection library to access any external data sources.

If you manage data connections for only a few workbook authors, consider specifying **Trusted data connection libraries and embedded**. This enables workbook authors to embed direct connections to external data sources in their workbooks, but still have access to trusted data connection libraries if the embedded links fail.

In the **Warn on Refresh** area of the **External Data** section, you can specify whether a warning is displayed before a workbook updates from an external data source. By selecting **Refresh warning enabled**, you ensure that external data is not automatically refreshed without user interaction.

In the **Display Granular External Data Errors** option, if you enable the Granular External Data Errors setting it provides descriptive error messages to display that provide helpful information for troubleshooting and fixing connection problems.

In the **Stop When Refresh on Open Fails** area, you can specify if Excel Calculation Services stops opening a workbook if the workbook contains a Refresh on Open data connection that fails. By selecting **Stopping open enabled**, you ensure that cached values are not displayed if an update operation fails when the workbook is opened by any user having View Only permissions to the workbook. When Refresh on Open is successful, cached values are purged. By clearing the **Stopping open enabled** check box, you risk displaying cached values if Refresh on Open fails.

In the **External Data Cache Lifetime** area of the **External Data** section, you can specify the maximum time that cached values can be used before they expire, and the maximum number of external data queries that can execute at the same time in a single session.

To ensure that only trusted users have access to workbooks stored in trusted locations, it is important to enforce ACLs on all trusted file locations.

There are three core scenarios to deploy Excel Services Application with SharePoint Server 2010: enterprise, small department, and custom.

In an enterprise deployment, consider the following guidelines:

- Do not configure support for user-defined functions.
- Do not enable workbooks to use embedded data connections to directly access external data sources.
- Limit the use of data connection libraries for external data source access from workbooks.
- Restrict the size of workbooks that can be opened in Excel Calculation Services.
- Selectively trust specific file locations and do not enable **Trust Children** for trusted sites and directories.

In a small department deployment, consider the following guidelines:

- Enable trust for all file locations that are used by department members to store workbooks.
- Enable **Trust Children** for all trusted sites and directories.
- Selectively restrict access to specific file locations if problems occur.

In a custom deployment, consider the following guidelines:

- Enable Excel Calculation Services to open large workbooks.
- Configure long session time-out settings.
- Configure large data caches.
- Create a single trusted location for this deployment.
- Do not enable **Trust Children** for this trusted location.

Trusted data providers

You can control access to external data by explicitly defining the data providers that are trusted and recording them in the list of trusted data providers. The list of trusted data providers designates specific external data providers to which workbooks opened in Excel Calculation Services are permitted to connect.

Before instantiating a data provider to enable a workbook to connect to an external data source, Excel Calculation Services checks the connection information to determine whether the provider appears on the list of trusted data providers. If the provider is listed, a connection is tried; otherwise, the connection request is ignored.

Trusted data connection libraries

A trusted data connection library is a document library from which you have determined that it is safe to access .odc files. Data connection libraries are used to help secure and manage data connections for workbooks that are accessed by a server that is running Excel Calculation Services. A list of trusted data connection libraries designates specific data connection libraries from which workbooks opened in Excel Calculation Services are permitted to access .odc files.

If a data connection is linked from a workbook that is accessed by a server that is running Excel Calculation Services, the server checks the connection information and the list of trusted data connection libraries. If the data connection library is listed, a connection is tried by using the .odc file from the data connection library; otherwise, the connection request is ignored.

View Only permissions

You can specify users who are only permitted to view workbooks by adding them to the SharePoint Server 2010 Viewers group or by creating a new group configured to use View Only permissions. By default, the Viewers group is configured to use View Only permissions. Users added to a group configured to use View Only permissions can view, open, interact with, refresh, and recalculate workbooks. But they are prevented from accessing the file source in any way, other than by using Excel Services Application. This helps you protect your proprietary information. The source data is never displayed to the designated users.

Workbooks and workbook data objects configured to use View Only permissions cannot be opened in Microsoft Excel 2010. However, a snapshot of the workbook, displaying only values and formatting of the server-viewable ranges, can be rendered in Excel 2010.

You can configure site settings in SharePoint Server 2010 to control access to workbook data by setting View Only permissions on centrally managed workbooks that are rendered in a web browser. You can also configure site settings in SharePoint Server 2010 to enable workbooks to refresh external data on the server, and to help secure and manage external data connections. See [Manage Excel Services authentication](#) for more information about how to save specified data objects as View Only items.

External data connections

The Excel Calculation Services component of Excel Services Application is used to connect to external data sources. Excel Calculation Services processes external data connection information that contains everything the server must have in order to connect to a data source. This includes how to authenticate, which connection string to use, which query string to use, and where and how to collect credentials to use for the connection. These connections can be defined in two locations: embedded within workbooks and in .odc files. The connection information is identical in both locations. The .odc files are small files that persist connection information in plain text and in a format that is reusable.

You can use the Excel 2010 client to author and edit .odc files and connections embedded in workbooks. In the Excel 2010 client, you can run the Data Connection Wizard or configure the settings in the **Connections** properties page. You can also export an .odc file based on these settings. The **Connections** properties page shows connection information, including Excel Services Application authentication properties.

.odc files

Workbooks can contain links to .odc files and embedded connection information. This enables workbooks to retrieve the .odc file, read the contents, and attempt to connect to an external data source if the embedded connection information fails. The .odc files must be managed and maintained to ensure that they contain accurate data connection information.

You can also configure Excel Calculation Services to use connection information from the .odc file exclusively instead of first trying to connect by using the embedded information. This approach enables administrators to deploy a small set of managed .odc files that provide updated connection information to many workbooks.

Workbook authors can specify, on a per-connection basis, which connection information the workbook can use. To do this, open Excel 2010 client and then click **Workbook Connections** on the **Data** tab. Add a connection to a workbook, open **Workbook Connections**, and then view the properties of the connection that you added. On the **Definition** tab, select **Always use connection file**. This setting enables the workbook to retrieve a connection file from a data connection library and use the connection information within the file to connect to an external data source. You can also configure this setting by selecting **Always use connection file** on the final page of the Data Connection Wizard.

Managing .odc files

Data connection libraries provide a repository for collections of .odc files. Administrators can manage data connections on the server by creating a data connection library and .odc files that require workbooks to always use a connection file. Workbooks that consume connections directly from a data connection library will always get updated connection information before they connect to a data source.

If data source information changes (for example, the server name), you only have to update one .odc file in the data connection library and all of the workbooks that consume the .odc file will be automatically updated the next time that they refresh. You can also use View Only permissions to restrict access to .odc files.

User-defined function assemblies

If your deployment scenarios include workbooks that contain user-defined functions to extend the capabilities of Excel Calculation Services, you must configure Excel Services Application to support user-defined functions.

To configure this support, you must enable user-defined functions on trusted file locations that contain workbooks that require access to user-defined functions. In addition, you must register user-defined function assemblies on the Excel Services Application user-defined function assembly list. See [Manage Excel Services authentication](#) for more information about how to enable user-defined functions.

Excel Services capacity planning

Many factors can affect the performance and availability of your Microsoft SharePoint Server 2010 deployment. These include network bandwidth and resource consumption. Excel Services in Microsoft SharePoint Server 2010 can affect system performance, depending on the volume of client connections and the number of concurrent Excel Calculation Services session requests. Calculation size and complexity can also affect Excel Calculation Services resource consumption.

The scope and complexity of Microsoft Excel 2010 calculations can affect the performance and availability of Excel Services Application in a SharePoint Server 2010 deployment. As calculation volume, complexity, and frequency increase, more system resources are consumed. Define baseline hardware requirements for system memory, CPU speed, and storage for each application server that runs Excel Calculation Services. See [SharePoint Server 2010 performance and capacity test results and recommendations](#) for the 2010 test results.

High Performance Computing Services for Excel 2010

You can now run Microsoft Excel 2010 workbooks and user-defined Functions on an Windows HPC cluster by using Windows HPC Server 2008 R2. For information about the Windows HPC Server 2008 R2 and Microsoft Excel 2010, collaboration, see [HPC Services for Excel](#)

Plan for PerformancePoint Services (SharePoint Server 2010)

This section describes how to plan for PerformancePoint Services within a Microsoft SharePoint Server 2010 environment.

In this section:

- [PerformancePoint Services overview \(SharePoint Server 2010\)](#)

PerformancePoint Services in Microsoft SharePoint Server 2010 is a performance management service that you can use to monitor and analyze your business. By providing flexible, easy-to-use tools for building dashboards, scorecards, and key performance indicators (KPIs), PerformancePoint Services can help individuals across an organization make informed business decisions that align with companywide objectives and strategy.

- [Overview of PerformancePoint Services architecture](#)

Microsoft SharePoint Server 2010 provides the flexibility to meet many different deployment solution goals.

- [Estimate performance and capacity requirements for PerformancePoint Services](#)

This article provides guidance on the effect that PerformancePoint Services has on topologies running Microsoft SharePoint Server 2010. It describes what was learned from having created a test farm to perform capacity testing for several different scenarios and various topologies.

- [Client hardware and software requirements for PerformancePoint Dashboard Designer](#)

This article describes the hardware and software requirements for running PerformancePoint Dashboard Designer.

- [Plan for importing PerformancePoint Server 2007 dashboard content to SharePoint Server 2010 \(SharePoint Server 2010\)](#)

To facilitate and simplify the process of importing dashboard content from Microsoft Office PerformancePoint Server 2007 into PerformancePoint Services in Microsoft SharePoint Server 2010, an import wizard is available.

- [Plan for PerformancePoint Services security \(SharePoint Server 2010\)](#)

In PerformancePoint Services in Microsoft SharePoint Server 2010, the objects stored in lists and document libraries are secured by the Microsoft SharePoint Server 2010 security model. On top of that model, PerformancePoint Services adds additional product features to the basic SharePoint Server 2010 framework to ensure that data sources and dashboard content are secure and protected from unwarranted access.

- [Authorization and permissions in PerformancePoint Services \(SharePoint Server 2010\)](#)

PerformancePoint Services uses the SharePoint Server security model to control user access to various functionality and tasks.

- [Planning for PerformancePoint data sources \(PerformancePoint Services\)](#)
This article is an overview of data sources that are available through PerformancePoint Services.
- [Best practices for SQL Server 2005 and 2008 OLAP cube design and MDX querying](#)
This article highlights best practices that help improve the performance of SSAS as a data source for Excel Services in Microsoft SharePoint Server 2010, PerformancePoint Services in Microsoft SharePoint Server 2010, and Visio Services in Microsoft SharePoint Server 2010.
- [Overview of PerformancePoint Services components](#)
PerformancePoint Services and Microsoft SharePoint Server 2010 front-end Web components work together to provide monitoring and reporting functionality.
- [Plan to customize PerformancePoint Services](#)
This article describes ways that you can customize the native functionality of PerformancePoint Services in Microsoft SharePoint Server 2010 by using the PerformancePoint Services SDK.
- [Plan a PerformancePoint dashboard to show organizational performance](#)
These articles cover PerformancePoint dashboards.
- [PerformancePoint Services and PowerPivot for Excel \(white paper\)](#)
This white paper discusses integration between PerformancePoint Services and PowerPivot for Excel.

See Also

[Best practices for SQL Server 2005 and 2008 OLAP cube design and MDX querying](#)

[PerformancePoint Services overview \(SharePoint Server 2010\)](#)

PerformancePoint Services overview (SharePoint Server 2010)

PerformancePoint Services in Microsoft SharePoint Server 2010 is a performance management service that you can use to monitor and analyze your business. By providing flexible, easy-to-use tools for building dashboards, scorecards, and key performance indicators (KPIs), PerformancePoint Services can help individuals across an organization make informed business decisions that align with companywide objectives and strategy. Dashboards, scorecards, KPIs, and reports help drive accountability. Integrated analytics help workers quickly move from monitoring information to analyzing it, and where appropriate, sharing it throughout the organization. Before PerformancePoint Services became part of Microsoft SharePoint Server 2010, Microsoft Office PerformancePoint Server 2007 was a standalone server. Now the functionality of Microsoft Office PerformancePoint Server 2007 is available as an integrated part of the Office SharePoint Server Enterprise license. PerformancePoint Services retains much of the same features and functionality as its predecessor while including additional benefits, enhancements, and new functionality.

**Note:**

An upgrade wizard is now available, that moves database objects from the previous version, PerformancePoint Server 2007, to PerformancePoint Services.

PerformancePoint Services

The following is an overview of features for PerformancePoint Services.

- Create interactive and context driven dashboards with scorecards, analytic reports and filters. Reports also integrate with SQL Server Reporting Services and Excel Services.
- Create scorecards that bring together data from multiple data sources (including Analysis Services, SQL Server, SharePoint lists and Excel Services) to track and monitor key drivers of your business.
- Use interactive analytic reports to identify driving forces and root causes, and apply filters to personalize your reports.
- Integrate your business intelligence applications and information with other powerful SharePoint Server features, such as collaboration and content management.

New features and enhancements

The following is an overview of new features and enhancements.

- PerformancePoint is a service in SharePoint Server. Dashboards and dashboard items are stored and secured within SharePoint Server lists and libraries, providing you with a single security and repository framework. The new architecture also takes advantage of SharePoint Server security framework, scalability, collaboration, backup and restore, and disaster recovery capabilities. You

can also include and link PerformancePoint Services Web Parts together with other SharePoint Server Web Parts on the same page. The new architecture also streamlines security models that simplify access to report data.

- The visualization Decomposition Tree is a new report type that you can use to quickly and visually break down higher-level data values from a multi-dimensional data set in order to understand the driving forces behind those values. The Decomposition Tree is available in scorecards and analytic reports rendered in your dashboards.
- The KPI Details report is a new report type that displays contextually relevant information about KPIs, metrics, rows, columns, and cells within a scorecard. The KPI Details report works as a Web Part that links to a scorecard or individual KPI and can be added to dashboards or any SharePoint Server page.
- Scorecards have been enhanced to make it easy for you to drill down and quickly access more detailed information. PerformancePoint scorecards also offer more flexible layout options, dynamic hierarchies, and calculated KPI features. Using this enhanced functionality, you can now create custom metrics that use multiple data sources. You also can sort, filter, and view variances between actual and target values to help you identify concerns or risks.
- With Better Time Intelligence filtering capabilities you can create and use dynamic time filters that are always up to date. Other improved filters enhance the ability for dashboard users to quickly focus in on information that is most relevant.
- SharePoint Web Parts can include and link to PerformancePoint Web Parts on the same page.
- It's easier to author and publish dashboard items using Dashboard Designer.
- There is increased support for accessibility compliance in individual reports and scorecards.
- Analytic reports have been enhanced to support value filtering, new chart types, and server-based conditional formatting.
- Support for SQL Server Analysis Services 2008 and SQL Server 2008.

Retired features

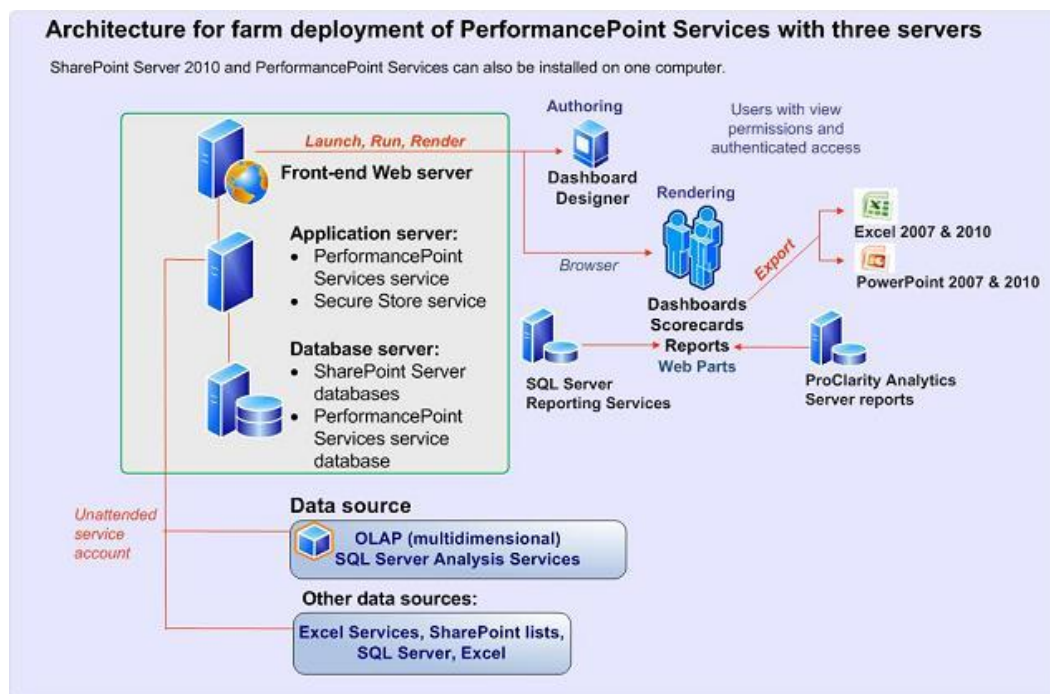
PerformancePoint Services no longer supports Trend Charts, PivotTable reports, PivotChart reports, Analysis Services 2000, and 32-bit server architecture.

Overview of PerformancePoint Services architecture

Microsoft SharePoint Server 2010 provides the flexibility to meet many different deployment solution goals. For more information about designing server farms and physical topologies, see the model called "Services in SharePoint 2010 Products" in [Technical diagrams \(SharePoint Server 2010\)](#).

PerformancePoint Services topology

The following is a farm topology that uses three servers for implementing PerformancePoint Services in Microsoft SharePoint Server 2010.



The front-end Web server runs on Internet Information Services (IIS) and hosts the PerformancePoint Services Web Parts, Web services, and proxy that are required for communication between the client and the PerformancePoint Services service application.

The service application is a wrapper for the middle-tier business logic for an instance of PerformancePoint Services.

PerformancePoint Services as a service application

In SharePoint Server 2010 services are no longer contained within a Shared Service Provider (SSP). Instead, the infrastructure for hosting services becomes part of SharePoint Server 2010, and the configuration of service offerings is more flexible. The service applications framework is a common service model that provides the following benefits.

- A consistent management experience and shared infrastructure for all services. As one of those services, PerformancePoint Services can provide the ability to perform bulk security operations for service applications in a farm from the SharePoint Central Administration Web site.
- Backup and recovery benefits also include restoring a site collection, site, or list content to a previous version or point-in-time.
- Services are installed by default, so there is no SSP setup or configuration.
- Sites can be configured to use only the services that are needed, rather than the entire list of services.
- Deploying services across sites and farms is more flexible. Similar to previous versions, a single set of services can be shared by all Web sites in a farm.

Estimate performance and capacity requirements for PerformancePoint Services

This article describes the effect that use of PerformancePoint Services has on topologies running Microsoft SharePoint Server 2010.



Note:

It is important to be aware that the specific capacity and performance figures presented in this article will differ from the figures in real-world environments. The figures presented are intended to provide a starting point for the design of an appropriately scaled environment. After you have completed your initial system design, test the configuration to determine whether the system will support the factors in your environment.

In this article:

- [Test farm characteristics](#)
- [Test results](#)
- [Recommendations](#)

For general information about how to plan and run your capacity planning for SharePoint Server 2010, see [Capacity management and sizing for SharePoint Server 2010](#).

Test farm characteristics

Dataset

The dataset consisted of a corporate portal built by using SharePoint Server 2010 and PerformancePoint Services that contained a single, medium-sized dashboard. The dashboard contained two filters linked to one scorecard, two charts, and a grid. The dashboard was based on a single Microsoft SQL Server 2008 Analysis Services (SSAS) data source that used the AdventureWorks sample databases for SQL Server 2008 Analysis Services cube.

The table that follows describes the type and size of each element on the dashboard.

Name	Description	Size
Filter One	Member selection filter	7 dimension members
Filter Two	Member selection filter	20 dimension members
Scorecard	Scorecard	15 dimension member rows by 4 columns (2 KPIs)
Chart One	Line chart	3 series by 12 columns

Name	Description	Size
Chart Two	Stacked bar chart	37 series by 3 columns
Grid	Analytic grid	5 rows by 3 columns

The medium dashboard used the Header and Two Columns template, and the dashboard item sizes were set to either auto-size or a specific percentage of the dashboard. Each item on the dashboard was rendered with a random height and width between 400 and 500 pixels to simulate the differences in Web browser window sizes. It is important to change the height and width of each dashboard item because charts are rendered based on Web browser window sizes.

Test scenarios and processes

This section defines the test scenarios and discusses the test process that was used for each scenario. Detailed information such as test results and specific parameters are given in the "Test results" sections later in this article.

Test name	Test description
Render a dashboard and randomly change one of the two filters five times with a 15 second pause between interactions.	<ol style="list-style-type: none"> 1. Render the dashboard. 2. Select one of the two filters and randomly select a filter value and wait until the dashboard is re-rendered. 3. Repeat four more times, randomly selecting one of the two filters and a random filter value.
Render a dashboard, select a chart, and expand and collapse it five times with a 15 second pause between interactions.	<ol style="list-style-type: none"> 1. Render the dashboard. 2. Select a random member on a chart and expand it. 3. Select another random member on the chart and collapse it. 4. Select another random member on the chart and expand it. 5. Select another random member on the chart and collapse it.
Render a dashboard, select a grid, and expand and collapse it five times with a 15 second pause between interactions.	<ol style="list-style-type: none"> 1. Render the dashboard. Select a random member on a grid and expand the member. 2. Select another random member on the grid

Test name	Test description
	<p>and expand it.</p> <p>3. Select another random member on the grid and collapse it.</p> <p>4. Select another random member on the grid and expand it.</p>

A single test mix was used that consisted of the following percentages of tests started.

Test name	Test mix
Render a dashboard and randomly change one of the two filters five times.	80%
Render a dashboard, select a chart, and expand and collapse it five times.	10%
Render a dashboard, select a grid, and expand and collapse it five times.	10%

Microsoft Visual Studio 2008 Load Testing tools were used to create a set of Web tests and load tests that simulated users randomly changing filters and navigating on grids and charts. The tests used in this article contained a normal distribution of 15-second pauses, also known as "think times," between interactions and a think time between test iterations of 15 seconds. Load was applied to produce a two-second average response time to render a scorecard or report. The average response time was measured over a period of 15 minutes after an initial 10 minute warm-up time.

Each new test iteration select a distinct user account from a pool of five thousand accounts and a random IP address (using Visual Studio IP Switching) from a pool of approximately 2,200 addresses.

The test mix was run two times against the same medium-sized dashboard. In the first run, the data source authentication was configured to use the Unattended Service Account, which uses a common account to request the data. The data results are identical for multiple users, and PerformancePoint Services can use caching to improve performance. In the second run, the data source authentication was configured to use per-user identity, and the SQL Server Analysis Services cube was configured to use dynamic security. In this configuration, PerformancePoint Services uses the identity of the user to request the data. Because the data results could be different, no caching can be shared across users. In certain cases, caching for per-user identity can be shared if Analysis Services dynamic security is not configured and the Analysis Services roles, to which Microsoft Windows users and groups are assigned, are identical.

Hardware setting and topology

Lab hardware

To provide a high level of test-result detail, several farm configurations were used for testing. Farm configurations ranged from one to three Web servers, one to four Application servers, and a single database server that was running Microsoft SQL Server 2008. A default enterprise installation of SharePoint Server 2010 was performed.

The following table lists the specific hardware that was used for testing.

	Web server	Application server	Computer that is running SQL Server	Computer that is running Analysis Services
Processor(s)	2px4c @ 2.66 GHz	2px4c @ 2.66 GHz	2px4c @ 2.66 GHz	4px6c @ 2.4 GHz
RAM	16 GB	32 GB	16 GB	64 GB
Operating system	Windows Server 2008 R2 Enterprise	Windows Server 2008 R2 Enterprise	Windows Server 2008 R2 Enterprise	Windows Server 2008 R2 Enterprise
NIC	1x1 gigabit	1x1 gigabit	1x1 gigabit	1x1 gigabit
Authentication	NTLM and Kerberos	NTLM and Kerberos	NTLM and Kerberos	NTLM and Kerberos

After the farm was scaled out to multiple Web servers, a hardware load balancer was used to balance the user load across multiple Web servers by using *source-address affinity*. Source-address affinity records the source IP address of incoming requests and the service host that they were load-balanced to, and it channels all future transactions to the same host.

Topology

The starting topology consisted of two physical servers, with one server acting as the Web and application server and the second server as the database server. This starting topology is considered a two-machine (2M) topology or a "1 by 0 by 1" topology where the number of dedicated Web servers is listed first, followed by dedicated application servers, and then database servers.

Web servers are also known as web front ends (WFE) later in this document. Load was applied until limiting factors were encountered. Typically the CPU on either the Web or application server was the limiting factor, and then resources were added to address that limit. The limiting factors and topologies

differed significantly based on the data source authentication configuration of either the Unattended Service Account or per-user Identity with dynamic cube security.

Test results

The test results contain three important measures to help define PerformancePoint Services capacity.

Measure	Description
User count	Total user count reported by Visual Studio.
Requests per second (RPS)	Total RPS reported by Visual Studio, which includes all requests and a static file requests such as images and style sheets.
Views per second (VPS)	<p>Total views that PerformancePoint Services can render. A view is any filter, scorecard, grid, or chart rendered by PerformancePoint Services or any Web request to the rendering service URL that contains RenderWebPartContent or CreateReportHtml. To learn more about CreateReportHtml and RenderWebPartContent, see the PerformancePoint Services RenderingService Protocol Specification (http://go.microsoft.com/fwlink/?LinkId=200609).</p> <p>IIS logs can be parsed for these requests to help plan the capacity of PerformancePoint Services. Also, using this measure provides a number that is much less dependent on dashboard composition. A dashboard with two views can be compared to a dashboard with 10 views.</p>



Tip:

When you are using a data source configured to use Unattended Service Account authentication, the rule for the ratio of dedicated servers is one Web server to every **two** application servers that are running PerformancePoint Services.



Tip:

When you are using a data source configured to use per-user authentication, the rule for the ratio of dedicated servers is one Web server to every **four** or more application servers that are running PerformancePoint Services.

At topologies larger than four application servers, it is likely that the bottleneck is the Analysis Services server. Consider monitoring the CPU and query time of your Analysis Services server to determine whether you should scale out Analysis Services to multiple servers. Any delay in query time on the Analysis Services server will significantly increase the average response time of PerformancePoint Services beyond the desired threshold of two seconds.

The tables that follow show a summary of the test results for both Unattended Service Account authentication and per-user authentication when scaling out from two to seven servers. Detailed results that include additional performance counters are included later in this document.

Unattended Service Account authentication summary

Topology (WFE x APP x SQL)	Users	Requests per second (RPS)	Views per sec (VPS)
2M (1x0x1)	360	83	50
3M (1x1x1)	540	127	75
4M (1x2x1)	840	196	117
5M (1x3x1)	950	215	129
6M (2x3x1)	1,250	292	175
7M (2x4x1)	1,500	346	205

Per-user authentication summary

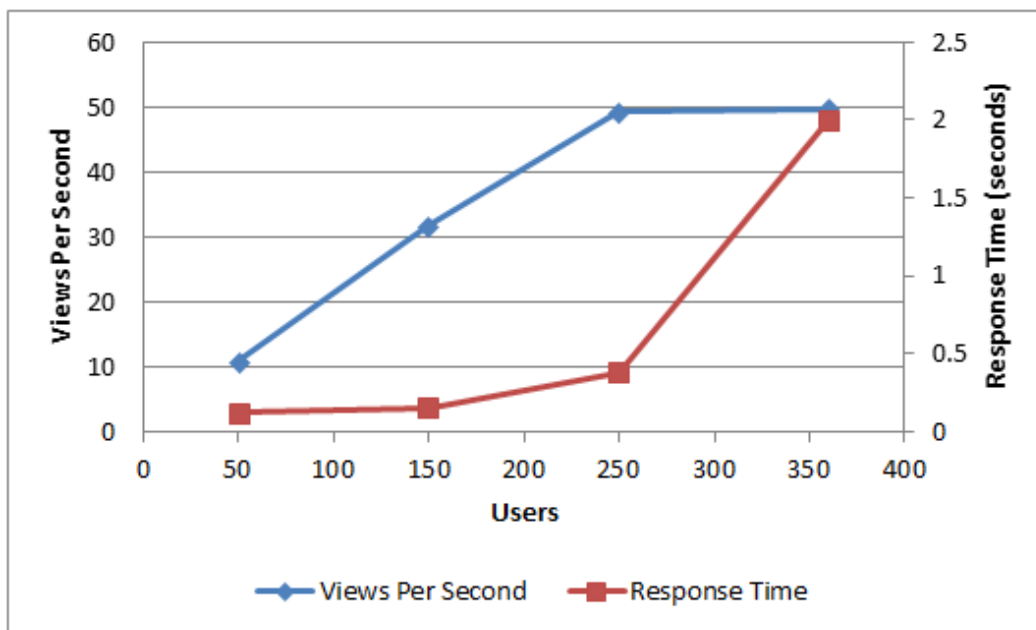
Topology (WFE x APP x SQL)	Users	Requests per second (RPS)	Views per sec (VPS)
2M (1x0x1)	200	47	27
3M (1x1x1)	240	56	33
4M (1x2x1)	300	67	40
5M (1x3x1)	325	74	44

2M and 3M topologies

To help explain the hardware cost per transaction and the response time curve, the load tests were run with four increasing user loads to the maximum user load for the 2M and 3M topologies.

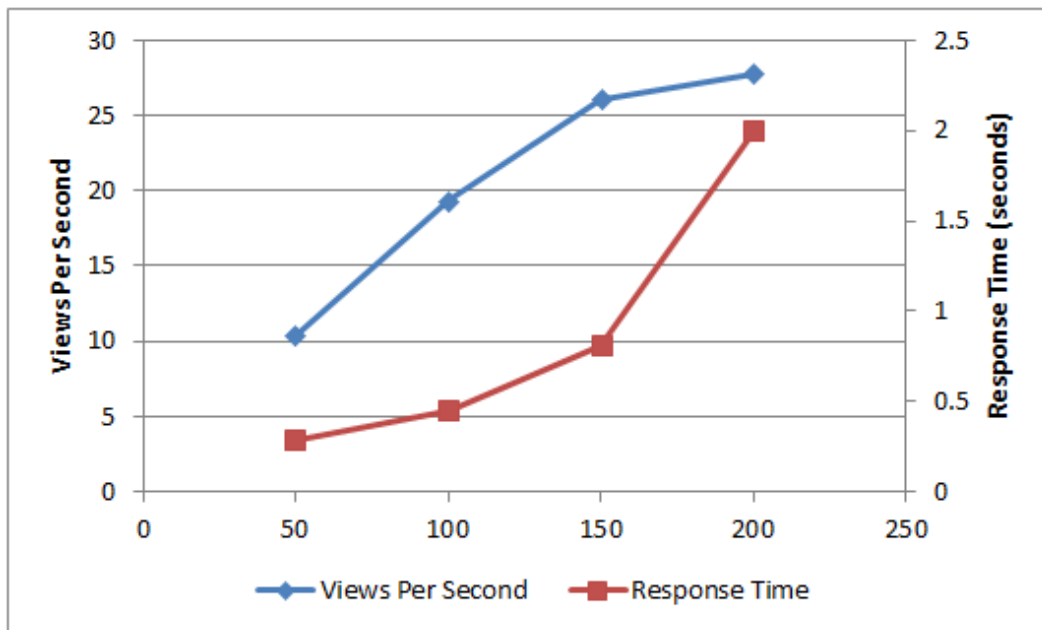
Unattended Service Account authentication

User count	50	150	250	360
Average WFE/APP CPU	19.20%	57.70%	94.00%	96.70%
RPS	18	53	83	83
Views per second	10.73	31.72	49.27	49.67
Average response time (sec)	0.12	0.15	0.38	2



Per-user authentication

User count	50	100	150	200
Average WFE/APP CPU	30.80%	61.30%	86.50%	93.30%
RPS	17	32	43	47
Views per second	10.3	19.32	26.04	27.75
Average response time (sec)	0.28	0.45	0.81	2

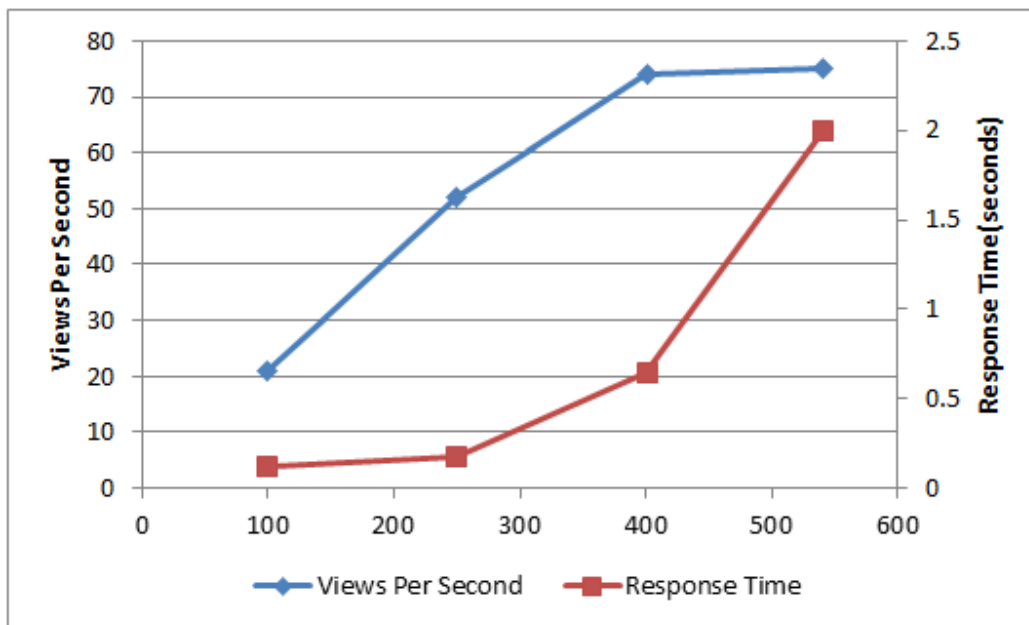


3M (1x1x1) farm results

Unattended Service Account authentication

User count	100	250	400	540
RPS	36	87	124	127
Views per second	21	52	74	75
Average response time (sec)	0.12	0.18	0.65	2

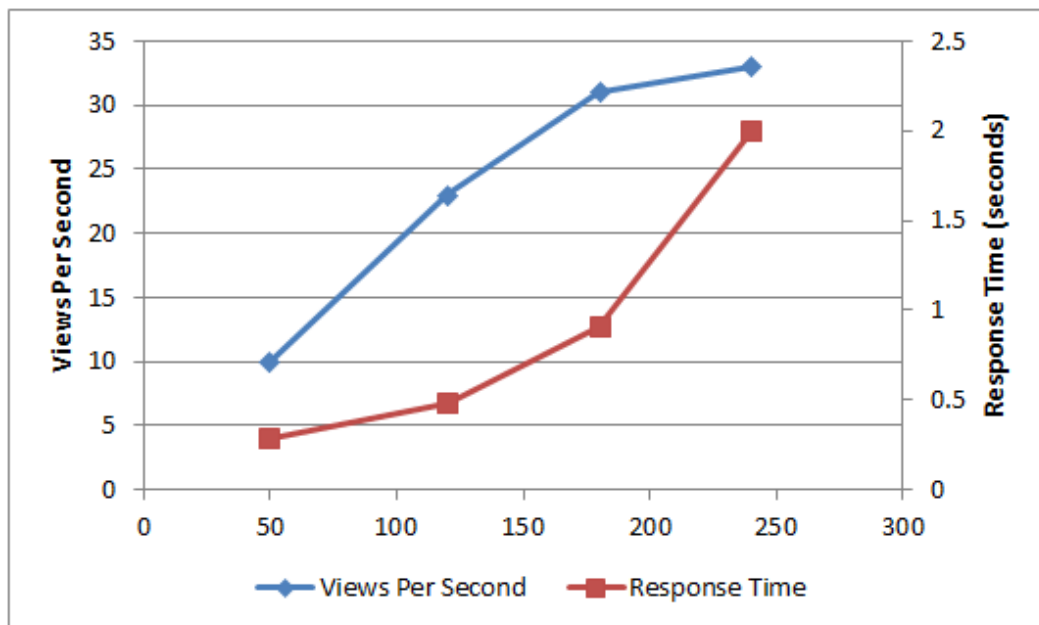
User count	100	250	400	540
Average WFE CPU	11%	28%	43%	46%
Max WFE private bytes of SharePoint Server Internet Information Services (IIS) worker process W3WP.	0.7 GB	1.4 GB	2.0 GB	2.4 GB
Average APP CPU	25%	62%	94%	95%
Max APP private bytes of PerformancePoint Services W3WP	5.9 GB10.8 GB	10.8 GB	14.1 GB	14.6 GB



Per-user authentication

User count	50	120	180	240
RPS	17	39	52	56
Views per second	10	23	31	33

User count	50	120	180	240
Average response time (sec)	0.28	0.48	0.91	2
Average WFE CPU	5%	12%	17%	19%
Max WFE private bytes of SharePoint Server W3WP	0.78 GB	1.3 GB	1.6 GB	1.9 GB
Average APP CPU	25%	57%	81%	81%
Max APP private bytes of PerformancePoint Services W3WP	19 GB	20.1 GB	20.5 GB	20.9 GB



4M+ results for Unattended Service Account authentication

Starting with a 4M topology, load was applied to produce a two-second average response time to render a scorecard or report. Next, an additional server was added to resolve the limiting factor (always CPU on the Web server or the application server) and then the test mix was re-run. This logic was repeated until a total of seven servers was reached.

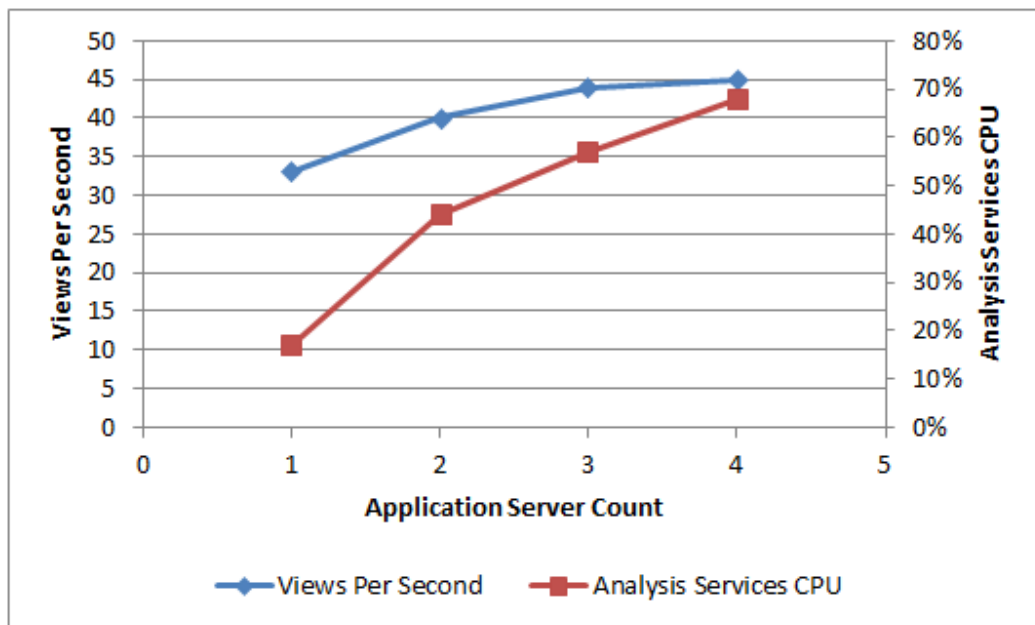
	4M (1x2x1)	5M (1x3x1)	6M (2x3x1)	7M (2x4x1)
User count	840	950	1,250	1,500
RPS	196	216	292	346
Views per second	117	131	175	206
Average. WFE CPU	77%	63%	54%	73%
Max WFE private bytes of SharePoint Server W3WP	2.1 GB	1.7 GB	2.1 GB	2.0 GB
Average APP CPU	83%	94%	88%	80%
Max APP private bytes of PerformancePoint Services W3WP	16 GB	12 GB	15 GB	15 GB

4M+ Results for per-user authentication

The same testing was repeated for a data source configured for per-user authentication. Note that adding an application server to create a four-application server topology did not increase the number of users or requests per second that could be supported by PerformancePoint Services because of the query delays that Analysis Services produced.

	3M (1x1x1)	4M (1x2x1)	5M (1x3x1)	6M (1x4x1)
User count	240	300	325	325
RPS	56	67	74	74
Views per second	33	40	44	45
Average. WFE CPU	19%	24%	26%	12%
Max WFE private bytes of SharePoint Server W3WP	2.1 GB	1.9 GB	1.9 GB	1.5 GB
Average APP CPU	89%	68%	53%	53%
Max APP private bytes of PerformancePoint	20 GB	20 GB	20 GB	20 GB

	3M (1x1x1)	4M (1x2x1)	5M (1x3x1)	6M (1x4x1)
Services W3WP				
Analysis Services CPU	17%	44%	57%	68%



Recommendations

Hardware recommendations

The memory and processor counters from the test tables should be used to determine the hardware requirements for an installation of PerformancePoint Services. For Web servers, PerformancePoint Services uses the recommended SharePoint Server 2010 hardware requirements. Application server hardware requirements may have to be changed when PerformancePoint Services consumes a large amount of memory. This happens when data sources are configured to per-user authentication or when the application server runs many dashboards with long data source timeouts.

The database server did not become a bottleneck in the tests and peaked at a maximum CPU usage of 31% under the 7M Unattended Service Account authenticated dashboard. The PerformancePoint Services content definitions such as reports, scorecards, and KPIs are stored in SharePoint lists and are cached in memory by PerformancePoint Services, reducing the load on the database server.

Memory consumption

PerformancePoint Services can consume large amounts of memory in certain configurations, and it is important to monitor memory usage of the PerformancePoint Services application pool.

PerformancePoint Services caches several items in memory, including Analysis Services and other data-source query results for the data source cache lifetime (a default of 10 minutes). When you are using a data source that is configured for Unattended Service Account authentication, these query results are only stored once and shared across multiple users. However, when you are using a data source that is configured for per-user authentication and Analysis Services dynamic cube security, the query results are stored once per user per view (that is, a "per filter" combination).

The underlying cache API that PerformancePoint Services uses is the ASP.NET Cache API. The significant advantage of using this API is that ASP.NET manages the cache and removes items (also known as a trim) based on memory limits to prevent out-of-memory errors. The default memory limit is 60 percent of physical memory. After reaching these limits, PerformancePoint Services still rendered views but response times increased significantly during the short period when ASP.NET removed cached entries.

The performance counter "ASP.NET Applications \ Cache API Trims" of the application pool hosting PerformancePoint Services can be used to monitor the ASP.NET cache trims that occur because of memory pressure. If this counter is greater than zero, then review the following table for possible solutions.

Problem	Solution
Application server processor usage is low and other services are running on the application server.	Add more physical memory or limit the memory of the ASP.NET cache.
Application server processor usage is low and only PerformancePoint Services is running on the application server.	If acceptable, configure the ASP.NET cache settings to have the cache use more memory, or add more memory.
Application server processor usage is high.	Add another application server.

A data source configured to use per-user authentication can share query results and cache entries if the Analysis Services role membership sets of the users are identical and if dynamic cube security is not configured. This is a new feature for PerformancePoint Services in Microsoft SharePoint Server 2010. For example, if user A is in role 1 and 2, and user B is in Role 1 and 2, and user C is in Role 1 and 2 and 3, only user A and user B share cache entries. If there is dynamic cube security, users A and B and also user C do not share cache entries.

Analysis Services

When PerformancePoint Services was being tested with per-user authentication, two Analysis Services properties were changed to improve multiple-user throughput performance. The following table shows the properties that were changed and the new value of each property.

Analysis Services property	Value
Memory \ HeapTypeForObjects	0
Memory \ MemoryHeapType	2

These two memory settings configure Analysis Services to use the Windows heap instead of the Analysis Services heap. Before changing these properties and while adding user load, response times increased significantly from 0.2 seconds to over 30 seconds while the CPU on the Web, application, and Analysis Services servers remained low. To troubleshoot, query time was collected by using Analysis Services dynamic management views (DMV), which showed an increase of individual query times from 10 milliseconds to 5000 milliseconds. These results led to modifying the above memory settings.

It is important to note that while this greatly improved throughput, according to the Analysis Services team, changing these settings has a small but measurable cost on single-user queries.

Before changing any Analysis Services properties, consult the [SQL Server 2008 White Paper: Analysis Services Performance Guide](http://go.microsoft.com/fwlink/?LinkID=165486) (<http://go.microsoft.com/fwlink/?LinkID=165486>) for best practices on improving multiple-user throughput performance.

Common bottlenecks and their causes

During performance testing, several common bottlenecks were revealed. A bottleneck is a condition in which the capacity of a particular constituent of a farm is reached. This causes a plateau or decrease in farm throughput. If high processor utilization was encountered as a bottleneck, additional servers were added to resolve the bottleneck. The following table lists some common bottlenecks and possible resolutions assuming processor utilization was low and not the bottleneck.

Possible bottleneck	Cause and what to monitor	Resolution
Analysis Services memory heap performance	By default, Analysis Services uses its own memory heap instead of the Windows heap, which provides poor multi-user	Change Analysis Services to use the Windows heap. See the "Analysis Services" section earlier in this article and the SQL Server 2008 White Paper: Analysis Services Performance Guide for instructions

Possible bottleneck	Cause and what to monitor	Resolution
	throughput performance. Review the Analysis Services query times using dynamic management views (DMV) to see if query times increase with user load and Analysis Services processor utilization is low.	(http://go.microsoft.com/fwlink/?LinkID=165486).
Analysis Services query and processing threads	By default, Analysis Services limits the number of query and processing threads for queries. Long running queries and high user loads could use all available threads. Monitor the idle threads and job queue performance counters under the MSAS 2008:Threads category.	Increase the number of threads available to query and process. See Analysis Services section and the SQL Server 2008 White Paper: Analysis Services Performance Guide for instructions (http://go.microsoft.com/fwlink/?LinkID=165486).
Application server memory	PerformancePoint Services caches the Analysis Services and other data source query results in memory for the data source cache lifetime. These items can consume a large amount of memory. Monitor the ASP.NET Applications \ Cache API Trims of the PerformancePoint Services application pool to determine whether cache removals or trims are being forced by ASP.NET because of low memory.	Add memory or increase the default ASP.NET cache memory limits. See Memory Consumption section earlier in this document for additional discussion. Also, see the ASP.NET cache element settings (http://go.microsoft.com/fwlink/?LinkId=200610) and Thomas Marquardt's blog post on Some history on the ASP.NET cache memory limits (http://go.microsoft.com/fwlink/?LinkId=200611).
WCF throttling settings	PerformancePoint Services is implemented as a WCF	If needed, change the Windows Communication Foundation (WCF) throttling behavior. See the

Possible bottleneck	Cause and what to monitor	Resolution
	<p>service. WCF limits the maximum number of concurrent calls as a service throttling behavior. Although long-running queries could hit this bottleneck, this is an uncommon bottleneck. Monitor the WCF / Service Model performance counter calls outstanding for PerformancePoint Services and compare to the current maximum number of concurrent calls.</p>	<p>WCF service throttling behaviors (http://go.microsoft.com/fwlink/?LinkId=200612) and Wenlong Dong's blog post on WCF Request Throttling and Server Scalability (http://go.microsoft.com/fwlink/?LinkId=200613).</p>

Performance monitoring

To help you determine when you have to scale up or scale out the system, use performance counters to monitor the health of the system. PerformancePoint Services is an ASP.NET WCF service and can be monitored by using the same performance counters used to monitor any other ASP.NET WCF service. In addition, use the information in the following tables to determine supplementary performance counters to monitor, and to which process the performance counters should be applied.

Performance counter	Counter Instance	Notes
ASP.NET Applications / Cache API Trims	PerformancePoint Services application pool	If the value is greater than zero, review the "Memory consumption".
MSAS 2008:Threads / Query pool idle threads	N/A	If the value is zero, review the "Analysis Services" section and SQL Server 2008 White Paper: Analysis Services Performance Guide (http://go.microsoft.com/fwlink/?LinkId=165486).
MSAS 2008:Threads / Query pool job queue length	N/A	If the value is greater than zero, review the "Analysis Services" section and SQL Server 2008 White Paper: Analysis Services Performance Guide (http://go.microsoft.com/fwlink/?LinkId=165486).

Performance counter	Counter Instance	Notes
MSAS 2008:Threads / Processing pool idle threads	N/A	If the value is greater than zero, review the "Analysis Services" section and SQL Server 2008 White Paper: Analysis Services Performance Guide (http://go.microsoft.com/fwlink/?LinkID=165486).
MSAS 2008:Threads / Processing pool job queue length	N/A	If the value is greater than zero, review the "Analysis Services" section and SQL Server 2008 White Paper: Analysis Services Performance Guide (http://go.microsoft.com/fwlink/?LinkID=165486).
WCF CountersServiceModelService 3.0.0.0(*)\Calls Outstanding	PerformancePoint Service Instance	If the value is greater than zero, see WCF Request Throttling and Server Scalability (http://go.microsoft.com/fwlink/?LinkID=200613).

See Also

[Plan for PerformancePoint Services \(SharePoint Server 2010\)](#)

Client hardware and software requirements for PerformancePoint Dashboard Designer

This article describes the hardware and software requirements for running PerformancePoint Dashboard Designer.

Hardware requirements

The following requirements apply to a single client computer running Dashboard Designer.

Component	Minimum requirement
Processor	1 - Dual Core 32-bit CPU (x86)
RAM	2 GB
Hard disk	2 GB

Software requirements

The following requirements apply to a single client computer running Dashboard Designer.

- A supported browser as described in [Plan browser support \(SharePoint Server 2010\)](#)
- [Microsoft Silverlight 2.0](#) (<http://go.microsoft.com/fwlink/?LinkId=166506>)

Plan for importing PerformancePoint Server 2007 dashboard content to SharePoint Server 2010 (SharePoint Server 2010)

To facilitate and simplify the process of importing dashboard content from Microsoft Office PerformancePoint Server 2007 into PerformancePoint Services in Microsoft SharePoint Server 2010, an import wizard is available. The data import wizard walks you through the process of importing your metadata: your dashboards, scorecards, data sources, KPIs, and so on. After the data import process is complete, you must "re-deploy" each individual dashboard.



Note:

We recommend that you set up a clean SharePoint Server 2010 environment before running the Import Wizard.

Reports types not supported in PerformancePoint Services

Not all report types that are supported in Microsoft Office PerformancePoint Server 2007 are supported in SharePoint Server 2010. The following types are not supported in SharePoint Server 2010 and therefore are not imported:

- Trend Analysis Charts
- Pivot Tables
- Pivot Charts
- Spreadsheets

Planning permissions and roles

PerformancePoint Services uses the SharePoint Server security model to control user access to various functionality and tasks. There are subtle yet significant changes in working with PerformancePoint Services in Microsoft SharePoint Server 2010 over Microsoft Office PerformancePoint Server 2007. In Microsoft Office PerformancePoint Server 2007, Monitoring Server has its own server and database that stored metadata and content. In Microsoft Office PerformancePoint Server 2007, security was applied globally at the server level and on each individual object.

In SharePoint Server 2010, the PerformancePoint metadata content is stored in SharePoint lists and document libraries. You therefore need to understand the differences between the assignment of permissions and roles between Microsoft Office PerformancePoint Server 2007 and SharePoint Server 2010.

**Note:**

In Microsoft Office PerformancePoint Server 2007, the administrator on the server computer is automatically made an administrator in PerformancePoint. In SharePoint Server 2010, that individual is not automatically made an administrator. If needed, this assignment may be done manually.

Roles and permissions

PerformancePoint Services uses SharePoint Server authorization groups and permissions. As you plan how your users will use the service, review the primary SharePoint Server roles.

- **Farm Administrator:** In order to edit Dashboard items, this role needs at least contributor permissions on content lists (or list items) and data source libraries (or library items).
- **Site collection Administrator** In order to edit Dashboard items, this role needs at least contributor permissions on data source libraries (or library items) only.
- **Site Administrator or List/Document Library contributor:** In order to edit Dashboard items, this role needs at least contributor permissions on content lists (or list items) and data source libraries (or library items).

**Important:**

If any person or role is tasked with re-deploying Dashboards after they have been imported from Microsoft Office PerformancePoint Server 2007, that person or role must have at least Designer permissions.

We recommend as a best practice that you create new SharePoint groups (or leverage existing ones) to help organize your roles within PerformancePoint Services. If you establish clear permission groups by work role you can keep better control over who has access to what.

The four server roles that are available in Microsoft Office PerformancePoint Server 2007 loosely map to predefined roles in SharePoint Server 2010. In PerformancePoint Services, they are Admin, Power Reader, Data Source Manager, and Create. In addition, two additional roles of Editor and Reader at the individual item level are set within Dashboard Designer. The table below maps out how roles in PerformancePoint Server 2007 map to PerformancePoint Services in Microsoft SharePoint Server 2010.

**Important:**

Being an administrator on the server does not automatically add you as an administrator in PerformancePoint Services in Microsoft SharePoint Server 2010.

PerformancePoint Server 2007 role	PerformancePoint Server 2007 Permissions	PerformancePoint Services in Microsoft SharePoint Server 2010 role	Comments
Admin	Edit any item and create	Contributor: Data	

PerformancePoint Server 2007 role	PerformancePoint Server 2007 Permissions	PerformancePoint Services in Microsoft SharePoint Server 2010 role	Comments
	new items	Content and Data Sources	
Power Reader	Read any items (used for SDK processes)	Read: Data Content and Data Sources	
Data Source Manager	Create new items (data sources only)	Contributor: Data Sources only	
Creator	Create new items (except for data sources)	Contributor: Data Content Only	
Item Permissions			
Editor	View, edit or delete the item	Contributor	
Reader	View the item	Read	

Another way of approaching access needs is to look at the permissions based on the tasks:

User task	PerformancePoint Services in Microsoft SharePoint Server 2010 permissions required
Launch Dashboard Designer	None, other than being an authenticated user in SharePoint Server 2010
Create PerformancePoint dashboard items and save them to a SharePoint list or document library	Contributor
Perform all Contributor tasks plus publish PerformancePoint Dashboards	Designer
View PerformancePoint dashboards and use interactive features	Read
Manage user permissions for Dashboard items	Full Control (Site) or Site Collection Administrator

Running the wizard

During the running of the import wizard, imported content such as dashboards and scorecards is placed in a single SharePoint list. Imported data sources are placed in a single document library for data

sources. To make the import process go smoothly, set up a clean SharePoint Server 2010 environment. Review the steps below and gather the account user name and passwords required.

Establish data source security: In PerformancePoint Server 2007, the security setting for data sources is stored in the Web.config file, located in C:\Program Files\Microsoft Office PerformancePoint Server\3.0\Monitoring\PPSMonitoring_1\WebService. In SharePoint Server 2010, the security is stored in each data source. Most PerformancePoint Server 2007 customers used "per-user identity" as their security method. If you are not certain how security was configured, examine your Web.config file (under the <appSettings> node. The three methods available in PerformancePoint Server 2007 are:

- A single shared user account is used to access all data sources.
- Per-user authentication (requires Kerberos delegation). In this case each user's account is used to access all data sources. The value of the setting bpm.serverconnectionperuser in the Web.config file is set to "True".
- Default authentication with user name in the connection string. This method uses a single shared user account to access all data sources, and the CustomerData connection string property is used to send the user name. If this method is used, the value of the setting bpm.UseASCustomerData is set to "True".

Log on to the PerformancePoint Server 2007 content server: Have your username and password ready, because the wizard prompts you for them.

Connect to the PerformancePoint Server 2007 content database: During the initial PerformancePoint 2007 setup, the name of the database defaults to PPSMonitoring. If the database was renamed, select the name of the database where the content resides.

Choose a list to store the dashboard items: KPIs, scorecards, reports, dashboards, filters, and indicators will be copied into the Dashboard Content list. Select a list that is empty. You must provide the name of the site collection, site, and list.

Choose a document library to store the data sources: Data sources are copied into a document library. The document library you select must support the PerformancePoint Data Source content type. A default install of PerformancePoint Services for SharePoint Server 2010 will create a data source-specific document library that you can use. We recommend that you select an empty document library. The site collection will have the same name as the site collection you provided for Dashboard items.

Review the settings: You will have an opportunity to review the settings you provided and edit them if necessary.

Review the import status: The wizard processes the data content import and provides a status bar on the progress of importing the content. It lists all elements and database sources that have been imported. After the importing process is completed, the final Import results appear, with each imported item listed by category (Data Sources, KPIs, Scorecards, and so on) and with information about any issues or errors that were encountered. The results also specify whether data source names were changed in the event of duplicates.

Post-migration tasks for the PerformancePoint dashboard author

Upgrading a PerformancePoint dashboard from Microsoft Office PerformancePoint Server 2007 to PerformancePoint Services in Microsoft SharePoint Server 2010 is fairly easy to do. The IT department typically handles the migration of PerformancePoint content, such as dashboard items and data source connections. Then, you complete the upgrade process by opening and testing dashboard items, making changes to reports and scorecards, and redeploying dashboards.

Post-migration tasks for the dashboard author

After the IT department has migrated dashboard content from PerformancePoint Server 2007 to PerformancePoint Services, you must perform several tasks to complete the upgrade:

- **Open and test migrated content.** Do this to make sure that dashboard items work correctly.
- **Remove report types that are no longer available and replace them with new reports.** Several report types that are available in PerformancePoint Server 2007 are not available in PerformancePoint Services. You must replace those reports by using other existing or new views.
- **Update reports and scorecards to use new functionality.** (This is optional.) PerformancePoint Services offers improved capabilities in reports and scorecards that you might want to use in your dashboards. For example, you can now display an analytic chart as a pie chart. Scorecards can now include more sophisticated key performance indicators (KPIs), such as KPIs that use calculated metrics.

Use the following resources to learn more about how to upgrade your dashboard content.

Task	Description
What's New in PerformancePoint Dashboards	This article describes the new features and functionality that are available in PerformancePoint dashboards.
Plan for importing PerformancePoint Server 2007 dashboard content to SharePoint Server 2010 (SharePoint Server 2010)	Certain kinds of reports that are available in PerformancePoint Server 2007 are not available in PerformancePoint Services. This article contains information about how to prepare to import PerformancePoint Server 2007 dashboard content. This includes which report types are not available in PerformancePoint Services.

Task	Description
Import Wizard: PerformancePoint Server 2007 content to SharePoint Server 2010	<p>This article describes the Import Wizard, a tool to help you import dashboard content from PerformancePoint Server 2007 to PerformancePoint Services.</p>
What's New in PerformancePoint Dashboard Designer	<p>This article describes new features and functionality in Dashboard Designer. These include tighter integration with SharePoint Server 2010, new and improved dashboard items that you can create, and an overview of the streamlined ribbon and workspace.</p>

See Also

[Create a plan for your PerformancePoint dashboard](#)

Plan for PerformancePoint Services security (SharePoint Server 2010)

In PerformancePoint Services in Microsoft SharePoint Server 2010, the objects stored in lists and document libraries are secured by the Microsoft SharePoint Server 2010 security model. On top of that model, PerformancePoint Services adds additional product features to the basic SharePoint Server 2010 framework to ensure that data sources and dashboard content are secure and protected from unwarranted access. Even though PerformancePoint Services has a dependency on the SharePoint Server 2010 security model, there are still special security considerations to consider and therefore plan and manage. All service-based security settings are managed within the SharePoint Server Central Administration Web site to facilitate the management of shared resources and user access.

This article covers areas to plan for in authentication, authorization, and data source authentication.

Authentication

In PerformancePoint Services, you can choose between three different methods for data source authentication.

- **Per User Identity:** Each user's own account is used to access all data sources. This method requires Kerberos delegation. A domain administrator must configure the Kerberos delegation between PerformancePoint Services and the data sources.



Note:

External data sources must reside within the same domain as the SharePoint Server 2010 farm. If external data sources do not reside within the same domain, authentication to the external data sources will fail. For more information, see [Planning considerations for services that access external data sources](#) in "Services Architecture Planning."

- **Unattended Service Account:** A single shared user account is used to access all data sources. This is a low privileged domain account stored in the Secure Store Service. In establishing your unattended service account, first determine if this account has the proper access to the data sources that will be required in the Dashboard.
- **Custom Data:** Provides the ability for SQL Server Analysis Services to include the currently authenticated user name as a parameter on the custom data field in an Analysis Services connection string. The Custom data option is only used for Analysis Services data sources and can be used against Analysis Services 2006 and 2008 servers.

Trusted Locations

In PerformancePoint Services, data source connections are contained in document libraries and data content (KPIs, filters, scorecards, etc.) is contained in document lists. In order to secure the content,

and prevent users from running queries against data sources if the objects in the query are not trusted, the lists and libraries must be established as “trusted” locations. The farm administrator has the option to have all locations in the farm set as “trusted” or opt to identify specific locations to trust. Because of the ability to easily define the location in the farm to be secured, the farm administrator is freed from having to secure the entire farm.

Trusted locations provide an additional layer of security that restricts query execution of data sources or of any object that is dependent on a data source that is not in a trusted location. The document library or any parent object up to the web application can be defined as trusted. In PerformancePoint Services, the configuration of trusted location settings is managed centrally through Central Administration. Configuration can also be managed by using Windows PowerShell 2.0 cmdlets. As you plan the security of PerformancePoint Services, consider if you want or need to secure your entire web application or to more tightly manage the location of secure data.

For example: Locations within a farm that are independently marked as “trusted” have the following SharePoint Server 2010 hierarchy for either data content or data sources:

1. Disable the use of Trusted Locations for either data sources and/or content for the entire farm.
2. Trust lists and/or document libraries in the web application.
3. Trust lists and/or document libraries in a site collection including any child sites.
4. Trust lists and/or document libraries in a site.
5. Trust an individual list and/or document library in the farm.

When verifying whether a location is trusted, the server will check if Trusted Locations is enabled. If that property is enabled, the server will check the list of trusted locations starting at the site collection and proceeding to each lower level of the hierarchy to verify that the content is trusted.

Items that do not use a data source do not have to be in a trusted location to be rendered. This includes Web pages, static KPIs, dashboards, and indicator icons.



Note:

Trusted data source locations cannot be defined on a list and Trusted content locations cannot be defined on a document library.

Trusted data content libraries

Trusted data content libraries are SharePoint Server 2010 document libraries that contain PerformancePoint Services data connection (.ppsd) files. The .ppsd files are used to centrally manage connections to data sources, including SQL Server databases, OLAP cubes, relational databases, and Excel Services spreadsheets.

The data sources are defined in Dashboard Designer and stored in a trusted data connection library on SharePoint Server 2010. A trusted data connection library is a document library that you have determined as safe. It restricts the use of the data source files, but still allows them to be read. A document library is created by default when provisioning PerformancePoint Services, however. Administrators can manage data connections on the server by creating more than one data connection

library. If a user updates data source connection in the document library, the information is shared and updated when a workspace file is opened in Dashboard Designer.

Trusted Lists for Dashboard Content

Reports, scorecards, KPIs, and filters are all required to be stored in a trusted SharePoint Server 2010 list. The list or any parent object up to the site collection can be defined as trusted during the initial configuration or later via Central Administration.

Data source security

In PerformancePoint Services the security setting for data sources is stored in each data source. The setting that determines if the server uses the currently authenticated user, unattended user account, or unattended user account using custom data is configured on each individual data source.

The Secure Store Service and Unattended Service Accounts

The SharePoint Server 2010 Secure Store Service provides the capability of securely storing data such as credentials and associating them to a specific identity or group of identities. The Secure Store Service is present on all SharePoint Server 2010 farms.

In PerformancePoint Services, each data source can be configured to use the currently authenticated user credentials or the “Unattended Service Account”. The unattended service account is a set of domain credentials that are impersonated when connecting to a data source. The server uses the unattended service account rather than the managed account for data source queries to prevent the PerformancePoint Services process from accessing the content database during query execution.

PerformancePoint Services stores and retrieves unattended service account credentials in the Secure Store Service. Because the server needs to keep both the user name and password in order to impersonate the user, the password for the unattended service account is stored in the Secure Store Service. The user name is stored in the PerformancePoint Services database so it is accessible and can be displayed in the settings page.

When creating your unattended service account, ensure that the account has the proper access to the data sources that will be required.

It is important to understand that unattended service account credentials are not cached globally. Rather, they are retrieved from the Secure Store Service only when they are needed. If you open a workspace file in Dashboard Designer with a data source that connects using the unattended service option and the credentials aren't already cached for that connection, the unattended service account password is retrieved from the Secure Store Service and uses the target data source.

Claims-based authentication

Claims-based authentication in SharePoint Server 2010 supports multiple authentication providers on a single web application and is used to pass the users identity between the front-end web servers and the application servers. PerformancePoint Services supports multiple authentication providers only when you use dashboard content through a web browser. Dashboard Designer is not supported when you directly access a URL for any web application using multiple authentication providers. In order to use the Dashboard Designer in this configuration, you must extend the web application to configure access to the new URL that is restricted to the Windows authentication provider.

See Also

[Configure the unattended service account for PerformancePoint Services](#)

[Configure claims authentication \(SharePoint Server 2010\)](#)

[Plan authentication methods \(SharePoint Server 2010\)](#)

[Plan for importing PerformancePoint Server 2007 dashboard content to SharePoint Server 2010 \(SharePoint Server 2010\)](#)

Authorization and permissions in PerformancePoint Services (SharePoint Server 2010)

Planning permissions and roles

PerformancePoint Services uses the SharePoint Server security model to control user access to various functionality and tasks. There are subtle yet significant changes in working with PerformancePoint Services in Microsoft SharePoint Server 2010 over Microsoft Office PerformancePoint Server 2007. In Microsoft Office PerformancePoint Server 2007, Monitoring Server has its own server and database that stores metadata and content. In Microsoft Office PerformancePoint Server 2007, security is applied globally at the server level and on each individual object.

In SharePoint Server 2010, the PerformancePoint metadata content is stored in SharePoint lists and document libraries. You therefore need to understand the differences between the assignment of permissions and roles between Microsoft Office PerformancePoint Server 2007 and SharePoint Server 2010. In Microsoft Office PerformancePoint Server 2007, the administrator on the server computer is automatically made an administrator. In SharePoint Server 2010, that individual is not automatically made an administrator. If needed, this assignment may be done manually.

Roles and permissions

PerformancePoint Services uses SharePoint Server authorization groups and permissions. As you plan how your users will use the service, review the primary SharePoint Server roles.

- **Farm Administrator:** In order to edit Dashboard items, this role needs at least contributor permissions on content lists (or list items) and data source libraries (or library items).
- **Site collection Administrator** In order to edit Dashboard items, this role needs at least contributor permissions on data source libraries (or library items).
- **Site Administrator or List/Document Library contributor:** In order to edit Dashboard items, this role needs at least contributor permissions on content lists (or list items) and data source libraries (or library items).



Important:

If any person or role is tasked with re-deploying Dashboards after they have been imported from Microsoft Office PerformancePoint Server 2007, that person or role must have at least Designer permissions.

We recommend as a best practice that you create new SharePoint groups (or leverage existing ones) to help organize your roles within PerformancePoint Services. If you establish clear permission groups by work role you can keep better control over who has access to what.

The four server roles that are available in Microsoft Office PerformancePoint Server 2007 loosely map to predefined roles in SharePoint Server 2010. In PerformancePoint Services, they are Admin, Power Reader, Data Source Manager, and Create. In addition, two additional roles of Editor and Reader at the individual item level are set within Dashboard Designer. The table below maps out how roles in PerformancePoint Server 2007 map to PerformancePoint Services in Microsoft SharePoint Server 2010.



Important:

Being an administrator on the server does not automatically add you as an administrator in PerformancePoint Services in Microsoft SharePoint Server 2010.

PerformancePoint Server 2007 role	PerformancePoint Server 2007 Permissions	PerformancePoint Services in Microsoft SharePoint Server 2010 role	Comments
Admin	Edit any item and create new items	Contributor: Data Content and Data Sources	
Power Reader	Read any items (used for SDK processes)	Read: Data Content and Data Sources	
Data Source Manager	Create new items (data sources only)	Contributor: Data Sources only	
Creator	Create new items (except for data sources)	Contributor: Data Content Only	
Item Permissions			
Editor	View, edit or delete the item	Contributor	
Reader	View the item	None	

Another way of approaching access needs is to look at the permissions based on the tasks:

User task	PerformancePoint Services in Microsoft SharePoint Server 2010 Permissions Required
Launch Dashboard Designer	None, other than being an authenticated user in SharePoint Server 2010

User task	PerformancePoint Services in Microsoft SharePoint Server 2010 Permissions Required
Create PerformancePoint Dashboard items and save them to a SharePoint list or document library.	Contributor
Perform all Contributor tasks plus publish PerformancePoint Dashboards	Designer
View PerformancePoint Dashboards and use interactive features	Read
Manage user permissions for Dashboard items	Full Control (Site) or Site Collection Administrator

See Also

[Plan for PerformancePoint Services security \(SharePoint Server 2010\)](#)

Planning for PerformancePoint data sources (PerformancePoint Services)

In PerformancePoint Services you must create a connection to the data source or sources you want to use in your dashboard. All data used in PerformancePoint Services is external data, living in data repositories outside of PerformancePoint. After you establish a data connection, you will be able to use the data in the various PerformancePoint feature areas.

PerformancePoint supports tabular data sources including SharePoint Lists, Excel Services, SQL Server tables and Excel workbooks; and multi-dimensional (Analysis Services) data sources



Important:

PerformancePoint Services does not support Office 2010 “Click-to-Run” (C2R) applications as data sources. You must install the client.

Tabular Data Sources

A user can create a data connection to SharePoint Lists, Excel Services, SQL Server tables or Excel workbooks. For these kinds of data sources, you can view a sample of the data from the Dashboard Designer tool and set specific properties for the data depending how you want the data to be interpreted within PerformancePoint. For example, you can indicate which datasets should be treated as a dimension. You can specify if a dataset is to be treated as a dimension or a fact; or if you do not want the data to be included, you can select **Ignore**. If you decide to set the value as a fact, you can indicate how those numbers should be aggregated in PerformancePoint Services. You can also use datasets that have time values within PerformancePoint Services and use the PerformancePoint Services time intelligence features to set time parameters and create dashboard filters.



Important:

Tabular data sources can only be used for PerformancePoint scorecards and filters.

SharePoint Lists

Data contained in a SharePoint List on a SharePoint Site can be used in PerformancePoint Services by creating a SharePoint List data source in Dashboard Designer. Data from SharePoint Lists can only be read. Modification to SharePoint List data must be done from SharePoint.

Users may connect to any kind of SharePoint List.

Excel Services

Data in Excel files published to Excel Services on a SharePoint Site can be used in PerformancePoint Services by creating an Excel Services data source. Supported published data can only be read in PerformancePoint Services. Published parameter values can be modified from the Dashboard Designer. If you use an Excel Services parameter in calculating a KPI, it is easy to make additional changes. PerformancePoint Services supports the following Excel Services components: Named Ranges, Tables and Parameters

noteDXDOC112778PADS Security Note

You cannot connect to an Excel Services as a data source when the site or library contain the workbook you are trying to connect to is set for Anonymous Access.

SQL Server tables

You can create a data source connection to a SQL Server database and use the data within PerformancePoint Services. SQL tables and SQL views are supported data sources within PerformancePoint Services.

noteDXDOC112778PADS Security Note

The data from SQL Server database is read-only. PerformancePoint Services does not support any modification to SQL Server database.



Note:

After creating and saving a SQL data source connection in Dashboard Designer, the data source is listed in the workspace browser with a pencil icon still viewable, as if the data source was not saved and still is being edited. This is a known issue.

Excel workbooks

You may use the content of an actual Excel file as a data source in PerformancePoint Services by creating an Excel Workbook data source connection and select only the data that is to be used. The original Excel file will be independent from the PerformancePoint copy. PerformancePoint Services 2010 supports Excel 2007 and Excel 2010 workbooks as data sources.

Multidimensional Data Sources

Analysis Services

Use data residing in a SQL Server Analysis Services cube in PerformancePoint Services by creating a data connection to the source. PerformancePoint Services lets you map the desired time dimension

and the required level of detail for its hierarchies to the internal PerformancePoint Services Time Intelligence.



Note:

Unlike tabular data sources, users cannot browse an Analysis Services cube within PerformancePoint's Dashboard Designer.

PowerPivot for Excel

In PerformancePoint Services you can use a PowerPivot model as a data source to build your PerformancePoint Services dashboards. To use PowerPivot as a data source within a PerformancePoint Services dashboard, you must have PerformancePoint Services activated on a SharePoint Server 2010 farm and have PowerPivot for SharePoint installed. Once a PowerPivot model has been created by using the PowerPivot add-in for Excel, this Excel file must be uploaded or published to a SharePoint site that has PowerPivot services enabled. Create the data source connection in Dashboard Designer using the Analysis Services data source template.

See Also

[Create a PowerPivot data connection \(PerformancePoint Services\)](#)

[Configure data sources to work with Time Intelligence by using Dashboard Designer](#)

[Configure Analysis Services data source time settings by using Dashboard Designer](#)

[Configure tabular data source time settings by using Dashboard Designer](#)

Best practices for SQL Server 2005 and 2008 OLAP cube design and MDX querying

When you are enabling Microsoft SQL Server 2005 or 2008 Analysis Services (SSAS) in a SharePoint Server 2010 business intelligence solution, proper cube design, efficient multidimensional expressions, and sufficient hardware resources are critical to optimal performance.

This article highlights best practices that help improve the performance of SSAS as a data source for Excel Services in Microsoft SharePoint Server 2010, PerformancePoint Services in Microsoft SharePoint Server 2010, and Visio Services in Microsoft SharePoint Server 2010. The first section lists articles that highlight Microsoft SQL Server 2008 enhancements that are specifically designed for business intelligence (BI) solutions, while the second section gives best practices that apply to SSAS 2005/2008 cube design, MDX query optimization, and more.

SQL Server 2008 enhancements for business intelligence

The following articles describe enhancements to SQL Server 2008 that relate to business intelligence.

- [Top 10 Performance and Productivity Reasons to Use SQL Server 2008 for Your Business Intelligence Solutions](#) Provides a list, with some details, of the top performance reasons to use SQL Server 2008 for your new business intelligence solutions and to upgrade to SQL Server 2008 for your existing business intelligence solutions.
- [SQL Server 2008 Upgrade Technical Reference Guide](#) Gives information about upgrading your existing Business Intelligence solutions to SQL Server 2008.
- [Scaling Up Reporting Services 2008 vs. Reporting Services 2005: Lessons Learned](#) Describes enhancements in performance and scalability to SQL Server Reporting Services 2008.
- [Performance Improvements for MDX in SQL Server 2008 Analysis Services](#) Explains where issues can occur in your existing MDX code that will prevent you from experiencing the performance improvements in SQL Server Analysis Services 2008.
- [SQL Server 2008 White Paper: Analysis Services Performance Guide](#) Describes how application developers can apply query and processing performance-tuning techniques to their SQL Server 2008 Analysis Services OLAP solutions.

Best practices for Analysis Services

- [Analysis Services Query Performance Top 10 Best Practices](#) Covers ten best practices for optimizing Analysis Services query performance.
- [SQL Server Best Practices Article: Identifying and Resolving MDX Query Performance Bottlenecks in SQL Server 2005 Analysis Services](#) Provides information about available MDX query

troubleshooting tools. The article also demonstrates how to use the most common of these tools to identify and resolve MDX query performance bottlenecks with individual MDX queries.

- [Analysis Services Processing Best Practices](#) Provides best practices for processing in SQL Server 2005 Analysis Services.
- [OLAP Design Best Practices for Analysis Services 2005](#) Outlines recommended best practices for designing OLAP databases in SQL Server 2005 Analysis Services to better meet the functional and performance needs of users.
- [Microsoft SQL Server 2005 Analysis Services Performance Guide](#) Describes how application developers can apply performance-tuning techniques to their Microsoft SQL Server 2005 Analysis Services Online Analytical Processing (OLAP) solutions.
- [Scale-Out Querying with Analysis Services](#) Describes how to set up a load-balanced scalable querying environment for Microsoft SQL Server 2005 Analysis Services so that you can handle a large number of concurrent queries to your Analysis Services servers. Load-balanced querying ensures that readers of OLAP cubes can consistently query for the latest aggregations throughout the day and distribute the load of all queries among the available servers. This scale-out querying architecture optimizes cube processing time, increases the frequency of cube update, and makes processing more robust because you can afford more frequent processing and transparent error recovery.

Overview of PerformancePoint Services components

PerformancePoint Services and Microsoft SharePoint Server 2010 Web Front End components work together to provide monitoring and reporting functionality. This section briefly describes the following PerformancePoint Services components.

- PerformancePoint Dashboard Designer
- PerformancePoint Web Parts and services
- Site collection features
- Site features

Dashboard Designer

Dashboard Designer is the application that dashboard authors can use to create and manage dashboards and their elements, including reports, such as strategy maps and analytic charts, scorecards, and filters. Dashboards help organizations measure, monitor, and manage business performance with live data from a variety of business data sources. A PerformancePoint dashboard contains a variety of reports and scorecards that retain their individual connections to their data sources and that you can view data from multiple data sources in a single dashboard page.

Web Parts

PerformancePoint Web Parts are built-in server-side controls that run inside the context of special pages (that is, Web Part Pages) within any SharePoint Server 2010 site. Dashboard Designer elements function as SharePoint Server 2010 Web Parts.

Reports Web Part: The Reports Web Part helps you create PerformancePoint Services content types except the scorecard, KPI details report and filters. This includes analytic charts and grids, SQL Server Reporting Services reports, Excel Services reports, the strategy map, and others.

KPI Details Web Part: The KPI Details report displays contextually relevant information about KPIs, metrics, rows, columns, and cells within a scorecard. This is surfaced as a Web Part that would be linked to a scorecard or individual KPI, and surface relevant metadata to the end user. This Web Part can be added to PerformancePoint Services dashboards or any SharePoint Server 2010 page.

Scorecard View Web Part: The Scorecard Web Part provides view functionality for the Scorecard. Without the scorecard Web Part, the user is not able to render the KPIs in the Dashboard. Scorecards

may be linked to other Web Parts, such as filters and reports, to create an interactive dashboard experience. Users can use the Scorecard Web Part to do the following.

- Locate and add a Scorecard View Web Part to a dashboard.
- Add KPIs, new and existing, to a scorecard.
- Enter an advanced scorecard design surface to define relationships between KPIs and objectives as well as edit KPI and objective properties.
- Build a new scorecard, launching the scorecard design surface.

Filter Web Part: Central to creating compelling interactive dashboards is the ability to synchronize Web Part content through connections.

A connection can be thought of as a link made between Web Parts to enable exchange of data and to provide interactivity between Web Parts. Connections are made between filter providers and filter consumers. Although you cannot connect all types of filters to all types of Web Parts, you do have the ability to link many types of filters to most types of Web Parts.

PerformancePoint Site collections

In order for the dashboard author to create content in a SharePoint Server farm, users must be able to supply information on the location of the content in the SharePoint Server hierarchy. Currently the Dashboard Designer bases queries on the location of the Web service. The Web service is located, or scoped, in a SharePoint Server 2010 site collection. SharePoint Server 2010 Web applications commonly have a default site collection so that the site collection feature is enabled.

PerformancePoint Sites

PerformancePoint Services site makes available to users the list and document library templates that use the following content types:

- Dashboard
- Scorecard
- Key Performance Indicator (KPI)
- Report
- Filter
- Icon set (for KPI indicators)

Plan to customize PerformancePoint Services

This article describes ways that you can customize the native functionality of PerformancePoint Services in Microsoft SharePoint Server 2010 by using the PerformancePoint Services SDK.

Development scenarios for PerformancePoint Services

PerformancePoint Services supports the following development scenarios.

Create Report, Filter, and tabular DataSource extensions You can create custom PerformancePoint Services objects by setting custom values for properties on native [ReportView](#), [Filter](#), and [DataSource](#) objects.

Perform authoring and rendering operations You can write code that dynamically performs authoring and rendering operations.

Create scorecard transforms You can create scorecard transforms that change the appearance, contents, or functionality of scorecards before querying the data source, after querying the data source, or before rendering the scorecard in the Web Part. For more information, see [Overview of Scorecard Transforms](#).

Create custom Web Parts You can create custom Web Parts that interact with PerformancePoint Web Parts. For more information, see [PerformancePoint Web Parts](#).

PerformancePoint Services and PowerPivot for Excel (white paper)

This white paper covers the basics of integration between PerformancePoint Services in Microsoft SharePoint Server 2010 and Microsoft SQL Server 2008 R2 PowerPivot for Microsoft Excel 2010. It describes the basics of utilizing a PowerPivot model in PerformancePoint Services and expands on issues and scope of using these applications together.

Download this guide as a Microsoft Word document (.docx).

[PerformancePoint 2010 and PowerPivot](http://go.microsoft.com/fwlink/?LinkId=192628) (http://go.microsoft.com/fwlink/?LinkId=192628) (27.9 MB)

Plan for Visio Services (SharePoint Server 2010)

This section discusses planning considerations for Visio Services in Microsoft SharePoint Server 2010.

In this section:

- [Visio Services overview \(SharePoint Server 2010\)](#)

Visio Services in Microsoft SharePoint Server 2010 is a service application that allows users to share and view Microsoft Visio web drawings. The service also enables data-connected Microsoft Visio 2010 diagrams to be refreshed and updated from a variety of data sources.

- [Plan Visio Services deployment](#)

Planning your Visio Services deployment before rollout can help give you the best system performance and user satisfaction, and also help you avoid unexpectedly overtaxing system resources in your SharePoint Server 2010 farm and related systems.

- [Plan Visio Services security \(SharePoint Server 2010\)](#)

Visio Services enables you to render published Visio Web Drawings. These drawings can be connected to external data, and drawing elements can be updated based on that data. Security is an important component for enabling these scenarios.

- [Data authentication for Visio Services](#)

This article covers connecting Visio Services Web drawings to data hosted on Microsoft SharePoint Server 2010 plus data hosted externally. Connection types and authentication methods are also covered.

- [Visio Services resources](#)

This article contains a list of Visio Services resources, including documentation, blog posts, and video demonstrations.

See Also

[Visio Graphics Service administration \(SharePoint Server 2010\)](#)

Visio Services overview (SharePoint Server 2010)

Visio Services in Microsoft SharePoint Server 2010 is a service application that lets users share and view Microsoft Visio Web drawings. The service also enables data-connected Microsoft Visio 2010 Web drawings to be refreshed and updated from various data sources.

Use and benefits of Visio Services

Visio Web drawings (*.vdw files) can be rendered by Visio Services and viewed in a Web browser. This lets users view Visio documents without having Visio or the Visio Viewer installed on their local computer. Basic exploration and navigation of these rendered Web drawings are supported within the Visio Web Access Web Part. Page designers can configure the user interface and functionality of the Web Part.

Visio Services can also refresh the data and recalculate the visuals of a Visio Web drawing hosted on a Microsoft SharePoint Server 2010 site. This enables published Web drawings to refresh connections to various data sources and to update affected data graphics and text fields.

Visio Web drawings can be published to SharePoint Server 2010 using Microsoft Visio Professional 2010 and Microsoft Visio Premium 2010.

Data sources supported by Visio Services

Connections to the data sources listed here may be refreshed by using Visio Services if they were created by using Microsoft Office Visio 2007 or Visio 2010 data-link technology and published using Visio 2010 publishing functionality. Refresh of data through any other mechanism into a Visio Web drawing will not be supported.

- SQL Server 7.0
- SQL Server 2000
- SQL Server 2005 (32- & 64-bit)
- SQL Server 2008 (32- & 64-bit)
- SQL Server 2008 R2 (32- & 64-bit)
- Sheet information that is stored in Excel workbooks (.xlsx files) published from Microsoft Office Excel 2007 or Microsoft Excel 2010 hosted on the same SharePoint Server 2010 farm
- SharePoint Server lists that are hosted on the same farm
- OLE DB or ODBC
- Custom Data Providers implemented as .NET Framework assemblies

**Note:**

For more information about creating custom data providers, see [Creating a Custom Data Provider with Visio Services](http://go.microsoft.com/fwlink/?LinkId=191029) (http://go.microsoft.com/fwlink/?LinkId=191029)

Published Visio drawings

Visio Web drawings (.vdw files) may be saved and published to SharePoint Server 2010 and shared with other users who do not have Visio 2010 or the Visio viewer installed on their computers, as Visio Services will render a .vdw format file natively in SharePoint Server 2010 as a Web page. .Vdw Web drawings can only be created by Microsoft Visio Professional 2010 or Microsoft Visio Premium 2010.

Standard Visio drawings (.vsd files) are not rendered by Visio Services and require Microsoft Visio to be viewed.

Plan Visio Services deployment

As an integrated part of Microsoft SharePoint Server 2010, Visio Services in Microsoft SharePoint Server 2010 is very easy to deploy. Planning your Visio Services deployment before rollout can help give you the best system performance and user satisfaction, and help you better manage system resources in your SharePoint Server 2010 farm and related systems.

Visio Services performance

Visio Services is implemented by using the Visio Graphics Service, which runs on one or more application servers in the farm. Like all such services, this service consumes resources such as processing capacity and memory on each server where the service is running.

System performance of application servers that are running the Visio Graphics Service may be affected by various factors such as the following:

- The size of the Web drawings being rendered
- The number of drawings connected to a data source
- The performance of the data sources to which Web drawings are connected
- The frequency of data refresh for data-connected Web drawings
- Peak loads of users who are accessing Web drawings
- Peak loads on external data sources accessed by Web drawings
- Complexity of Web drawings

The Web drawing size limit and refresh parameters can be adjusted by the administrator. Being able to adjust these parameters can help you adjust the performance of the server. If changing these parameters does not provide the desired performance, you may have to add processing capacity or memory.

When planning system resources for Visio Services, the most important factor is peak load. For example, if users will make heaviest use of the Visio Services functionality early Monday morning, plan your server capacity for that peak load. Peak load times can vary widely depending on how Visio Services is used within your organization. It is important to estimate peak loads as best as possible to avoid overtaxing system resources.

If your SharePoint Server 2010 farm has multiple application servers, each potentially running multiple services, you must decide which application server or servers that you want to run the Visio Services Service on. When determining this, look at the other SharePoint Server 2010 services that are running on each application server. When do the peak loads for each other service occur? Try to select an application server where the peak loads of other services occur at different times than those of Visio Services.

Besides peak load considerations, also consider the overall load of other services that are running on the same server. Some SharePoint Server services, such as Search, might benefit from their own

application servers that are not shared by other loads such as Visio Services. Others, such as Access Services, might be ideal companion loads for Visio Services depending on usage patterns.

In addition to SharePoint Server 2010 performance considerations, you should also examine the performance impact of Visio Services on your other systems. For example, if you have a data-connected Web drawing that is querying data from an Oracle database, what is the effect of your Visio Services peak load on that Oracle database? Large numbers of users querying any data source at the same time could put a strain on the resources of that data source.

The following best practices can be used to optimize the performance of Visio Services:

- Monitor the performance of the application servers in the farm and add CPU and memory if they are needed to handle peak loads.
- Start the Visio Graphics Service on more than one application server in the farm to gain additional performance.
- Run the Visio Graphics Service on an application server where other services experience peak load at a different time of day than the Visio Graphics Service.
- Limit the maximum Web drawing size.
- Increase the minimum cache age for Web drawings. This increases the interval in which a user sees cached data for a particular Web drawing.

Visio Graphics Service applications

SharePoint Server 2010 implements Visio Services through the Visio Graphics Service running on one or more application servers in the farm. It is within the Visio Graphics Service Application that you configure the various Visio Services settings, such as trusted data providers and drawing cache settings.

For many deployments, a single Visio Services Service Application is sufficient. However, SharePoint Server 2010 lets you create multiple service applications of each type if you want (for example, if you need to use different data sources that require different global settings or a different unattended service account within Visio Services).

Using a pilot deployment

To help determine capacity requirements for Visio Services, consider rolling Visio Services out to a limited pilot group that is representative of typical users. Giving a fairly small number of people access to Visio Services functionality lets you monitor server resource usage and effect on related systems, such as external data sources, without overtaxing system resources.

Once you have compiled performance data for the pilot group, you can extrapolate system requirements for Visio Services when you deploy it across your whole organization. The pilot data will also help you determine peak load requirements and times when peak loads are likely to occur.

By monitoring other affected systems — such as data sources used by data connected Web drawings — you can also determine the likely effect of Visio Services on other systems in your organization.

Monitoring

We highly recommend that you monitor system resources consumed by Visio Services — alongside the other services in your SharePoint Server 2010. It is typical for resource usage to increase over time as additional users are brought online and existing users make more use of Visio Services and other SharePoint Server technologies.

The SharePoint Server 2010 services architecture enables easy addition of application servers to the farm. As user demands increase, you can continue to add servers to the farm, and offload services — including Visio Services — to these new servers. Or, instead of offloading Visio Services to a new server, you can run the Visio Services service on multiple application servers at the same time to provide additional capacity and redundancy.

By monitoring resource usage, you can predict when additional capacity is likely to be needed and include the needed hardware in your organization's regular budgetary procedures. This also helps avoid system downtime or slow response caused by unexpectedly high server loads.

Backup and recovery of data

Visio Services settings and Visio documents stored in SharePoint Server 2010 libraries can be backed up by the farm administrator when doing a standard farm backup. However, be aware that if you publish Visio documents that are connected to data sources that are outside the farm, the data to which the Visio documents are connected is not backed up as part of a standard farm backup. In this case, the administrator of the system where the data resides should perform a sparse backup procedure.

Visio Professional 2010 and Visio Premium 2010 deployment

Visio Services lets you display Visio drawings using a Web Part without the need to have Microsoft Visio 2010 installed on the client computer. However, Visio Services does not allow for creating or editing Visio diagrams. As part of your deployment plan for Visio Services, you should also plan for the needs of diagram authors within your organization. Each diagram author who has to publish to Visio Services must have a copy of Microsoft Visio Professional 2010 or Microsoft Visio Premium 2010.



Important:

Visio Professional 2010 or Visio Premium 2010 is required for publishing to SharePoint Server 2010. Microsoft Visio Standard 2010 and previous versions of Visio cannot publish to the server.

Plan Visio Services security (SharePoint Server 2010)

In addition to the security requirements to deploy Microsoft SharePoint Server 2010, you should also review security considerations for a deployment that includes Visio Services in Microsoft SharePoint Server 2010. Visio Services enables you to render published Visio Web drawings. These drawings can be connected to external data, and drawing elements can be updated based on that data. Security is an important component for enabling these data-rendering scenarios. The Visio Graphics Service gives you a significant level of fine-grained control for the processing and displaying of Visio Web Drawings and what data sources they can connect to.

Web drawings that are not connected to data

Published Visio Drawings (.VDW files) must be stored in SharePoint document libraries to be opened by Visio Services. SharePoint Server 2010 maintains an access control list (ACL) for the files that are contained in the document library. By setting the library rules correctly you can limit access to a particular drawing.

Visio Web drawings that are connected to data

The Visio Graphics Service can connect to data sources. They include SharePoint lists, Excel workbooks hosted on the farm, databases such as Microsoft SQL Server, and custom data sources. You can control access to specific data sources by explicitly defining the data providers that are trusted and configuring them in the list of trusted data providers.



Note:

Visio Services accesses external data sources by using a delegated Windows identity. Consequently, external data sources must reside within the same domain as the SharePoint Server 2010 farm or Visio Services must be configured to use the Secure Store Service. If the Secure Store Service is not used and external data sources do not reside within the same domain, authentication to the external data sources will fail. For more information, see [Planning considerations for services that access external data sources](#) in “Services Architecture Planning.”

When Visio Services loads a data connected Web drawing, the service checks the connection information that is stored in the Web drawing to determine whether the specified data provider is a trusted data provider. If the provider is a member of the list, a connection is tried; otherwise, the connection request is ignored.

Once an administrator has configured Visio Services to enable connections to a particular data source, there are additional security configurations that must be made, depending on the kind of the data source. The following data sources are supported by Visio Services:

- Excel workbooks stored on SharePoint Server with Excel Services
- SharePoint lists
- Databases such as SQL Server databases
- Custom Data Providers

Visio Web drawings that are connected to SharePoint lists

Published Visio Drawings can be connected to SharePoint lists on the same farm that the drawing is hosted on. The user viewing the Web drawing must have access to both the drawing and the SharePoint list that the drawing is connected to. These permissions and credentials are managed by SharePoint Server 2010.

Visio Web drawings that are connected to Excel Services

Published Visio drawings can be connected to Excel workbooks hosted on the same farm as the Web drawing with Excel Services running and configured correctly. To view the Web drawing, the user must have access to both the drawing and the Excel workbook that the drawing is connected to. These permissions and credentials are managed by SharePoint Server 2010.



Note:

Excel workbooks can be connected to external data sources. For more information, see [Plan Excel Services data sources and external connections](#).

Visio Web drawings that are connected to SQL Server databases

When a published Visio Web drawing is connected to a SQL Server database, Visio Services uses additional security configuration options to establish a connection between the Visio Graphics Service and the database. Visio Web drawings can use connections stored in Office Data Connectivity (ODC) files. In order to author data-connected Web drawings that use the unattended account and the Secure Store Service, the users must first create Office Data Connectivity files by using Microsoft Excel.

The authentication methods supported by Visio Services are as follows:

- **Integrated Windows authentication** In this security model the Visio Graphics Service uses the drawing viewer's identity to authenticate with the database. Integrated Windows authentication with constrained Kerberos delegation is more helpful for increasing security than the other authentication methods shown in this list. This configuration requires constrained Kerberos delegation to be enabled between the application server that is running the Visio Graphics Service and the database server. The database itself might require additional configuration to enable Kerberos-based authentication, which is beyond the scope of this document.

- **Secure Store Service** In this security model the Visio Graphics Service uses the Secure Store Service to map the user's credentials to a different credential that has access to the database. The Secure Store Service supports individual and group mappings for both Integrated Windows authentication and other forms of authentication such as SQL Server Authentication. This gives administrators more flexibility in defining one-to-one, many-to-one, or many-to-many relationships. This authentication model can only be used by drawings that use an ODC file to specify the connection. The ODC file specifies the target application that will be used for credential mapping.
- **Unattended Service Account** For ease of configuration the Visio Graphics Service provides a special configuration where an administrator can create a unique mapping associating all users to a single account by using a Secure Store Target Application. This mapped account, known as the unattended service account, must be a low-privilege Windows domain account that is given access to databases. The Visio Graphics Service impersonates this account when it connects to the database if no other authentication method is specified. Note that this approach does not enable personalized queries against a database and does not provide auditing of database calls. This authentication method is the default authentication method that is used when you connect to SQL Server databases: if no ODC file is used in the Visio Web drawing that specifies a different authentication method, then Visio Services uses the credentials specified by the unattended account to connect to the SQL Server database.

In a larger server farm it is likely that Visio drawings will use a mix of the authentication methods described here. It is important to be aware of the following things:

- Visio Services supports usage of both the Secure Store Service and the unattended service account in the same farm. In Web drawings that are connected to SQL Server data but do not use ODC files, the unattended account is required and always used.
- If Integrated Windows authentication is selected, and authentication to the data source fails, Visio Services will not attempt to render the drawing using the unattended service account.
- Integrated Windows authentication can be used together with the Secure Store by configuring drawings to use an ODC file that specifies a Secure Store target application for those drawings that require specific credentials.

See Also

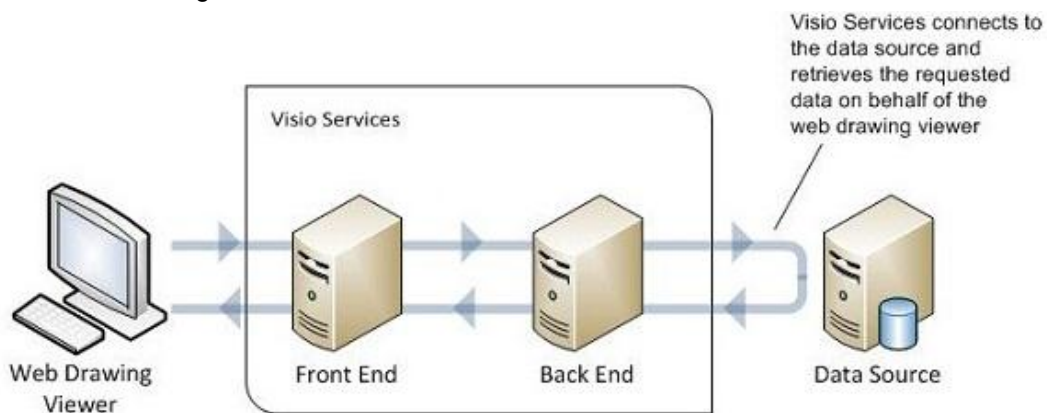
[Secure Store for Business Intelligence service applications](#)

Data authentication for Visio Services

Visio Services in Microsoft SharePoint Server 2010 supports data-connected Web drawings connected to various data sources, including the following:

- Data hosted within the SharePoint farm, such as a Microsoft Excel workbook or a SharePoint list.
- External data, such as Microsoft SQL Server data, or an OLE DB or ODBC data source.

Retrieving data from a data source requires that a user be authenticated by the data source and then authorized to access the data that is contained therein. In the case of a Web drawing, Visio Services will authenticate to the data source on behalf of the user who is viewing it in order to refresh the data to which the drawing is connected.



Which authentication method Visio Services can use to retrieve data depends on the type of the underlying data source, as outlined in the following table. For data sources that support more than one authentication method, data connections must specify which one to use.

Data source	Authentication method
SharePoint lists	SharePoint user permissions
Excel workbooks	SharePoint user permissions
SQL Server	One of: <ul style="list-style-type: none">• Windows authentication (integrated security)<ul style="list-style-type: none">• using Constrained Kerberos Delegation• using Secure Store• using the Unattended Service Account• SQL Server Authentication

Data source	Authentication method
OLE DB/ODBC	Varies per data source, typically a user-name and password pair stored in the connection string.

Custom data providers can also be used. For more information, see [Creating a Custom Data Provider with Visio Services](http://go.microsoft.com/fwlink/?LinkId=196860) (http://go.microsoft.com/fwlink/?LinkId=196860).

The following data sources are supported in Microsoft Visio but not in Visio Services:

- Access databases
- Excel workbooks not hosted on SharePoint Server
- OLAP

Connecting to data hosted on SharePoint Server

Visio Services supports data-connected Web drawings that are connected to data hosted within the SharePoint farm, including the following:

- Excel workbooks residing in a document library
- Data in SharePoint lists

Connecting to Excel workbooks

Visio Services uses the Web drawing viewer's SharePoint Server credentials to connect to an .xlsx Excel workbook. For the authentication operation to succeed, the following conditions must be met:

- Excel Services must be provisioned correctly and configured on the SharePoint farm.
- The workbook must be hosted on the same farm as the Web drawing.
- The Web drawing viewer must have at least "read" permissions to the Excel workbook.

No other configuration steps are required to enable this kind of data connection.



Note:

As part of connecting to an Excel workbook, Visio Services requests that Excel Services refresh the workbook if it contains connections to external data. In this case, the drawing viewer's identity is passed on to Excel Services so that Excel Services can authenticate to underlying data sources to refresh the workbook.

Connecting to SharePoint lists

Visio Services uses the Web drawing viewer's SharePoint Server credentials to connect to a SharePoint list. For the authentication operation to succeed the following conditions must be met:

- The SharePoint list must be hosted on the same farm as the Web drawing.
- The Web drawing viewer must have at least "read" permissions to the SharePoint list.

No other configuration steps are required to enable this kind of data connection.

Connecting to external data

Visio Services can connect to various external data sources, including SQL Server, OLE DB/ODBC, and custom data providers. To connect to the data source, Visio Services uses a specific data provider for each data source.

As a security measure, Visio Services must explicitly trust data providers before they can be used. For more information about trusted data providers, see [Configuring Visio Graphics Service trusted data providers \(SharePoint Server 2010\)](#).

Connecting to a Microsoft SQL Server data source can be done by using either:

- Windows authentication
- SQL Server Authentication

Other data sources use a connection string usually consisting of a user name and password.

Data connections

Visio Web drawings use one of two kinds of connections:

- Embedded connections
- Linked connections

Embedded connections are stored as part of the Visio Web drawing. Linked connections are stored externally to a Web drawing in Office Data Connection (ODC) files. To use a linked connection, a Web drawing must reference an .odc file that is also stored in the same farm as the Web drawing. Each data connection consists of:

- A connection string
- A query string
- An authentication method
- Optionally, some metadata required to retrieve external data

Each kind of connection has its advantages and drawbacks discussed here; choose the one that best suits your scenario.

Connection type	Embedded connections	ODC files
Data sources supported	<ul style="list-style-type: none">• SQL Server (only supports Kerberos delegation and the Unattended Service Account)• OLE DB/ODBC• Excel workbooks• SharePoint lists	<ul style="list-style-type: none">• SQL Server (supports all authentication methods)• OLE DB/ODBC

Connection type	Embedded connections	ODC files
	<ul style="list-style-type: none"> Custom Data Providers 	
Advantages	<ul style="list-style-type: none"> All connection information is stored in the Web drawing. Embedded connections require little administrative overhead to support. Embedded connections are easy to create. 	<ul style="list-style-type: none"> Linked connections can be centrally stored, managed, audited, shared and access to them controlled by using a data connection library. Drawing authors can use existing connections without having to create queries and connection string. If the data connection details for a data source change, an administrator only need update one ODC file. With that change, all Web drawings that refer to the ODC file will use the updated connection information when the next refresh occurs. (An example of this scenario is when the database server is moved or the database name is changed.)
Drawbacks	<ul style="list-style-type: none"> If the data connection details for a data source change, all Web drawings with embedded connections to that data source will have to be republished with updated connection information. Embedded data connections are more difficult to audit by SharePoint administrators. 	<ul style="list-style-type: none"> Creating a linked connection must be done by using Excel. Linked connections may require the help of a SharePoint administrator to share, manage and secure. Linked connections are saved in clear text and may contain database passwords. Extra care must be taken to help secure these files.

Choose a linked data connection, by using an ODC file, for scenarios in which you must have a data connection to an enterprise-scale relational data source such as SQL Server. Linked data connections

are most useful in scenarios in which they will be shared across many users and in which administrator control of the connection is important.



Note:

ODC files must first be created in Excel and exported to SharePoint Server before it can be used with Visio Services.

Choose an embedded connection for scenarios in which you have to have a quick data connection to a small or file-based data source that will only be used by some users.

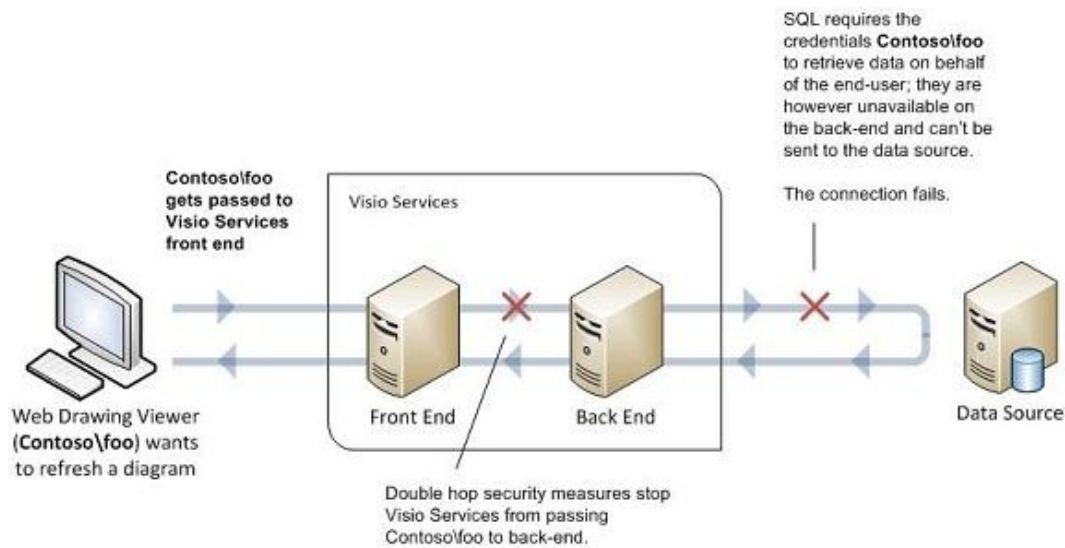
ODC files can be stored in a data connection library, a special kind of SharePoint document library. Centralizing data connections in such a document library has several advantages:

- Administrators can restrict write access to a data connection library to trusted data connection authors to ensure that only well tested and secure data connections are used by Web drawing authors.
- Administrators have a single location to manage data connections for a large group of users.
- Administrators can easily approve, audit, revert and manage data connection files by using document library versioning and workflow features.
- Data connection libraries can be reused across other Office applications such as Excel, Excel Services, Microsoft InfoPath 2010, InfoPath Forms Services, and Microsoft Word.
- End-users only have a single location to find drawing data, reducing confusion and user training.

For information about how to create data connection libraries, see [How to: Create and Use a Data Connection Library](http://go.microsoft.com/fwlink/?LinkID=188117) (<http://go.microsoft.com/fwlink/?LinkID=188117>). For information about how to create ODC files, see [Create, edit, and manage connections to external data](http://go.microsoft.com/fwlink/?LinkID=196894) (<http://go.microsoft.com/fwlink/?LinkID=196894>).

Windows authentication

Windows authentication requires that Visio Services present to SQL Server a set of Windows credentials. This kind of credential is common on Windows networks and is the same credential used to log on to computers on a Windows domain or to connect to a computer that is running Exchange Server. Windows credentials are considered the most secure and manageable means of controlling access to SQL Server databases. However, one obstacle to using Windows authentication with Visio Services is the Windows double hop security measure, wherein a user's credentials cannot be passed across more than one computer in a Windows network. Given that Visio Services is a multi-tiered system, special authentication methods are required for Visio Services to retrieve data on behalf of the end-user.



The authentication method to choose depends on various factors as outlined in the following table. Choose the one that best suits your scenario.

Authentication method	Kerberos delegation	Secure Store	Unattended Service Account
Description	Using constrained Kerberos delegation, the drawing viewer's Windows credentials are sent to the data source directly.	Using the Secure Store Service, the viewer's Windows credentials are mapped to another set of credentials specified in a Secure Store target application.	Using the Secure Store Service, all viewers are mapped to a unique set of credentials called the Unattended Service Account that is stored in a specific Secure Store target application specified in Visio Services Global Settings.
Data connection credentials	The Windows credentials of the Web drawing viewer.	The credentials specified in the Secure Store target application.	The credentials of the Unattended Service Account.

Authentication method	Kerberos delegation	Secure Store	Unattended Service Account
Advantages	<ul style="list-style-type: none"> • The Kerberos protocol is an industry standard in credentials management. • Kerberos ties into the existing Active Directory infrastructure. • Kerberos delegation permits auditing of individual accesses to a data source. • Given that the Web drawing viewer's identity is known, Web drawing creators can embed personalized database queries into Web drawings. 	<ul style="list-style-type: none"> • The Secure Store Service is part of SharePoint Server and is easier to configure than Kerberos. • Mappings are flexible: a user can be mapped either 1-to-1 or many-to-1. • Non-Windows credentials can be used to connect to data sources that do not accept Windows credentials. • Mappings created for Visio can be re-used by other business intelligence applications such as Excel Services. 	<ul style="list-style-type: none"> • The Unattended Service Account is the easiest authentication method to deploy and setup. • The Unattended Service Account does not require much administrative overhead.
Drawbacks	<ul style="list-style-type: none"> • Additional administrative effort required to configure for SharePoint Server and Visio Services. 	<ul style="list-style-type: none"> • Establishing and managing mapping tables requires some administrative overhead. • Secure Store permits limited auditing. In the many-to-1 scenario, individual incoming users are mapped into the same credentials through a target application, effectively blending them into one user. 	<ul style="list-style-type: none"> • Given that everyone is mapped to the same credentials, an administrator cannot distinguish who accessed a data source.

Authentication method	Kerberos delegation	Secure Store	Unattended Service Account
For the authentication operation to succeed ...	<ul style="list-style-type: none"> Kerberos delegation must be set up on a SharePoint farm. 	<ul style="list-style-type: none"> The Secure Store Service must be provisioned and configured on the Farm. It must also contain appropriate mapping information for a particular incoming user. Additionally the mapping information may need to be updated periodically to reflect password changes on the mapped account. 	<ul style="list-style-type: none"> The Secure Store Service must be provisioned and configured on the Farm. It must also contain appropriate the credentials for the Unattended Service Account. Additionally the mapping information may need to be updated periodically to reflect password changes on the mapped account. Visio Services must be configured to use Unattended Service Account.

Kerberos delegation

Choose Kerberos delegation for secure and fast authentication to enterprise-scale relational data sources that support Windows authentication. For information about configuring Kerberos delegation, see:

- [Configuring Kerberos Authentication for Microsoft SharePoint 2010 Products](http://go.microsoft.com/fwlink/?LinkId=196600)
(http://go.microsoft.com/fwlink/?LinkId=196600)
- [Configure Kerberos authentication \(SharePoint Server 2010\)](#)

Secure Store

Choose Secure Store for authentication to enterprise-scale relational data sources that may or may not support Windows Authentication. Secure Store is also useful in scenarios in which you want to control user credential mappings.

For information about using Secure Store with Visio Services, see [Secure Store for Business Intelligence service applications](#).



Video demonstration

This demonstration shows the steps for configuring Visio Services with Secure Store.

[Watch the video](http://go.microsoft.com/fwlink/?LinkId=196864) (<http://go.microsoft.com/fwlink/?LinkId=196864>). To download the video file, right-click the link, and then click **Save Target As**.

Unattended Service Account

For ease of configuration the Visio Graphics Service provides a special configuration where an administrator can create a unique mapping where all users are mapped into to a single set of credentials.

This account, known as the Unattended Service Account, must be a low-privilege Windows domain account. Visio Service impersonates this account when it connects to a data source on behalf of a Web drawing viewer.

It is a best practice to give this account as few network permissions as possible, typically only access to log into the network and access the data source you want to have users connect to. For best security, be sure that the Unattended Service Account does not have access to the SharePoint Configuration and Content databases.

The Unattended Service Account is used by Visio Services:

- When an ODC file specifies the use of the Unattended Service Account for either Windows or SQL Server Authentication
- When no ODC is used, and Kerberos authentication fails



Note:

The unattended account can be a local computer account of type Windows. If the unattended service account is configured as a local computer account, ensure that the configuration is identical on every application server running Visio Services. For manageability reasons, it is best practice to use a domain account

Choose the Unattended Service Account when connecting to small ad-hoc deployments in which security is less important or for which speed of deployment is essential.

For information about using the Unattended Service Account with Visio Services, see [Secure Store for Business Intelligence service applications](#).

Video demonstration

This demonstration shows the steps for configuring Visio Services with the Unattended Service Account.

[Watch the video](http://go.microsoft.com/fwlink/?LinkId=196865) (http://go.microsoft.com/fwlink/?LinkId=196865). To download the video file, right-click the link, and then click **Save Target As**.

SQL Server Authentication

SQL Server Authentication requires that Visio Services present a SQL Server user name and password to a SQL Server data source to authenticate. Visio Services extracts this username and password from the data connection's connection string and passes it to the data source.

To reduce security risks, Visio Services impersonates the Unattended Service Account when connecting to such a data source.

Authentication against OLEDB/ODBC data sources

Authentication to third party data sources typically requires that Visio Services present a user name and password to a data source. Like SQL Server Authentication, Visio Services extracts this user name and password from the data connection's connection string and passes them to the data source.

To reduce security risks, Visio Services impersonates the Unattended Service Account when connecting to such a data source.

Data refresh

Visio Services supports refreshing drawings connected to one or more of the following data sources:

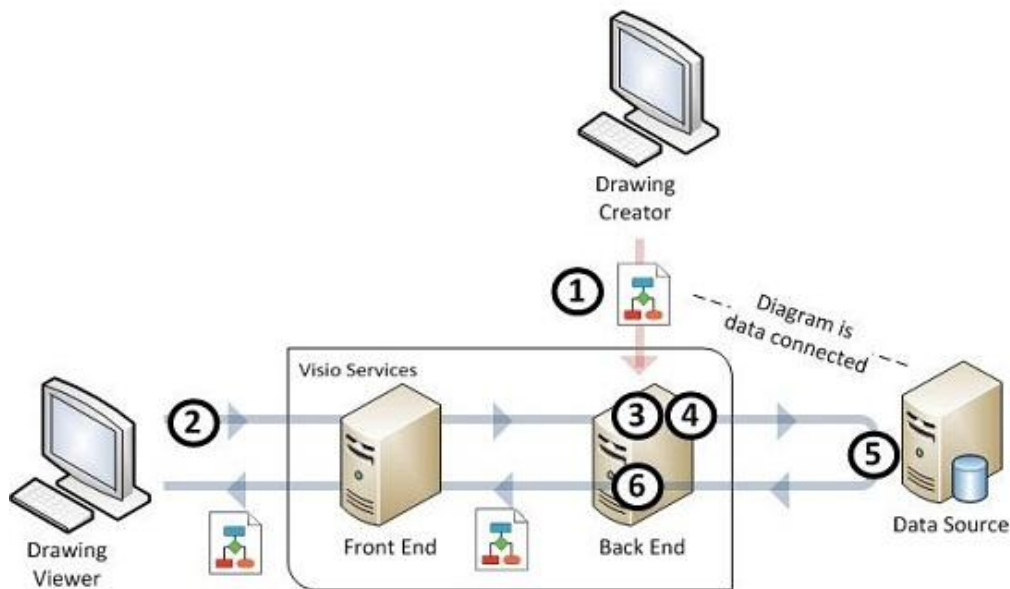
- SQL Server
- SharePoint lists
- Excel workbooks hosted in SharePoint Server
- Oracle 9i, 9iR2, 10g, 10gR2, 11g, 11gR2, and DB2 9.2



Note:

If the data source you plan to connect to is not in the list above, you can add support for it by creating a Visio Custom Data Provider. This technology enables you to wrap your existing data sources into one that Visio Services can consume. For more information, see [Creating a Custom Data Provider with Visio Services](http://go.microsoft.com/fwlink/?LinkId=191029) (http://go.microsoft.com/fwlink/?LinkId=191029) in the MSDN Library Online.

External data refresh is the result of the following set of steps through Visio Services.



1. **Creating a drawing:** A drawing author uploads a data-connected Web drawing to SharePoint Server 2010.
2. **Triggering Refresh:** The drawing viewer triggers refresh on a data-connected Web drawing.
3. **Data Connections:** Visio Services retrieves data connection information for each external data source in the drawing.
4. **Trusted Data Providers:** Visio Services checks to see if there is a trusted data provider it can use to retrieve data.
5. **Authentication:** Visio Services authenticates into the data source and retrieves the requested data on behalf of the drawing viewer.
6. **Drawing Refresh:** Visio Services updates the Web drawing based on the data source data and returns it to the viewer.

Refresh can be triggered in one of following ways from within the browser:

- The end-user opens the Web drawing.
- The end-user clicks on the refresh button on an already open Web drawing.
- The end-user loads a page that contains the Visio Web Access Web part which was configured to refresh automatically by a site designer.



Note:

A SharePoint site designer must place the Visio Web Access Web part on a page and configure it to refresh periodically.

Refresh can also be triggered in third party solutions by calling through JavaScript the **vwaControl.Refresh()** method of the Visio Web Access Web Part's Mash-up API. For more

information, see [Customizing Visio Web Drawings in the Visio Web Access Web Part](http://go.microsoft.com/fwlink/?LinkID=196503) (<http://go.microsoft.com/fwlink/?LinkID=196503>) in the MSDN Library Online.

If there are no previously cached versions of this Web drawing, any of these actions will trigger a refresh and update the Web drawing. For information about configuring cache settings for Visio Services, see [Configure Visio Graphics Service global settings \(SharePoint Server 2010\)](#).

Visio Services resources

The following is a list of available resources for Visio Services in Microsoft SharePoint Server 2010.

Documentation, references, and white papers

[Plan for Visio Services \(SharePoint Server 2010\)](#)

[Visio Graphics Service administration \(SharePoint Server 2010\)](#)

[Plan Visio Services security \(SharePoint Server 2010\)](#)

[Visio Services 2010 performance and capacity planning white paper](#)

(<http://go.microsoft.com/fwlink/?LinkID=191156>)

[Creating a Custom Data Provider with Visio Services](#) (<http://go.microsoft.com/fwlink/?LinkID=191029>)

[Visio Services Class Library](#) (<http://go.microsoft.com/fwlink/?LinkID=196500>)

[Visio 2010 Software Development Kit](#) (<http://go.microsoft.com/fwlink/?LinkID=196501>)

[Customizing Visio Web Drawings in the Visio Web Access Web Part](#)

(<http://go.microsoft.com/fwlink/?LinkID=196503>)

Blog posts

[Introducing Visio Services](#) (<http://go.microsoft.com/fwlink/?LinkID=196504>)

[Publishing Diagrams to Visio Services](#) (<http://go.microsoft.com/fwlink/?LinkID=196505>)

[Embedding a Web Drawing in a SharePoint Page](#) (<http://go.microsoft.com/fwlink/?LinkID=196506>)

[Installing and Configuring Visio Services](#) (<http://go.microsoft.com/fwlink/?LinkID=196507>)

[SharePoint Workflow Authoring in Visio Premium 2010 \(Part 1\)](#)

(<http://go.microsoft.com/fwlink/?LinkID=196508>)

[SharePoint Workflow Authoring in Visio Premium 2010 \(Part 2\)](#)

(<http://go.microsoft.com/fwlink/?LinkID=196509>)

[No Code Mashups with Visio Services and Web Part Connections](#)

(<http://go.microsoft.com/fwlink/?LinkID=196510>)

[The Visio Services JavaScript Mashup API](#) (<http://go.microsoft.com/fwlink/?LinkID=196511>)

Video demonstrations

[Set up Visio Services](#) (<http://go.microsoft.com/fwlink/?LinkID=196857>)

[Configure Visio Services with the Unattended Service Account](#)

(<http://go.microsoft.com/fwlink/?LinkID=196865>)

[Configure Visio Services with Secure Store](#) (<http://go.microsoft.com/fwlink/?LinkID=196864>)

Plan for Business Intelligence Indexing Connector (SharePoint Server 2010)

This section describes how to plan for using Microsoft Business Intelligence Indexing Connector.

- [Introduction to Business Intelligence Indexing Connector](#)

This article describes how you can deploy Microsoft Business Intelligence Indexing Connector to improve search capabilities for Microsoft Excel and Microsoft SQL Server Reporting Services (SSRS) reports and the underlying data sources that are part of the report.

- [Determine software requirements for Business Intelligence Indexing Connector](#)

This article describes the software prerequisites for the Business Intelligence Indexing Connector front end (on Microsoft SharePoint Server 2010) and back end (on Microsoft FAST Search Server 2010 for SharePoint) installations.

- [Overview of Business Intelligence Indexing Connector architecture](#)

This article describes the topology for a multi-server deployment of Microsoft Business Intelligence Indexing Connector added to Microsoft FAST Search Server 2010 for SharePoint (the back end) and SharePoint Server (the front end).

- [Overview of Business Intelligence Indexing Connector search tab interface](#)

This article describes the elements of the Reports search page.

Introduction to Business Intelligence Indexing Connector

Administrators can deploy Microsoft Business Intelligence Indexing Connector to improve search capabilities for Microsoft Excel and Microsoft SQL Server Reporting Services (SSRS) reports and the underlying data sources that are part of the report. When Microsoft Business Intelligence Indexing Connector is installed, an additional search tab, **Reports**, appears on the search page. By using the **Reports** search tab, users see improved results, descriptions, thumbnails, previews, and they are able to refine search results.

Searchable business intelligence assets include .xlsx and .xlsm documents, which are natively created by Microsoft Office Excel 2007 and Excel 2010. PowerPivot and SQL Server Reporting Services files are also searchable files.



Note:

XLSX and XLSM documents can be created by using older versions of Excel if the compatibility pack is installed.

By deploying Microsoft Business Intelligence Indexing Connector you can help the end user search and locate business intelligence assets that include the following characteristics:

- Additional data that is not explicitly included in the document text of the report, but that is still searchable.
- Additional business intelligence characteristics like dimensions and measures.

Features of Business Intelligence Indexing Connector

Business Intelligence Indexing Connector lets you increase relevancy of search queries in the following ways:

1. After you search in the **Reports** tab, refinement categories enable you to limit the results of your search by selecting available filters underneath the categories. Three additional categories are included that are not part of FAST Search Server 2010 for SharePoint. They are **Report Format**, **Data Sources**, and **Data Category**.

As an example, if you select **Table** as a format underneath the **Report Format** category, the results of your search are limited to files that contain tables. Likewise, if you select **Chart**, the results of your search are limited to files that have charts in them.

2. When you perform a search on the **Reports** tab, the resulting file description for each file provides description headers that help you locate the file that you want. Additionally, thumbnail images of files in PowerPivot Gallery and a preview option are available. The following list contains descriptions of the headers.

**Note:**

If there are no keywords, database names, or Excel worksheet names, the description header does not appear.

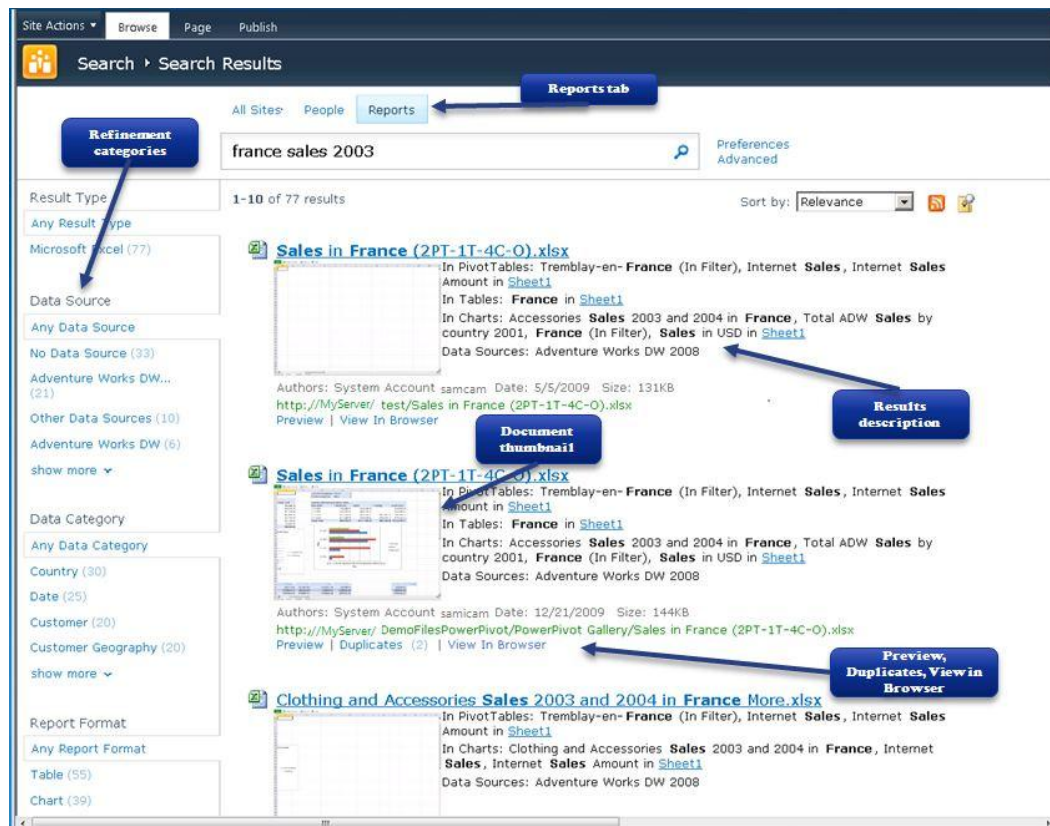
- **In PivotTable:** Displays the keywords from your search that are located in a PivotTable.
- **In Tables:** Displays the keywords from your search that are located in a table.
- **Charts:** Displays the keywords from your search that are located in a chart.
- **Data Sources:** Displays the name of the data source, if one exists.
- **Sheets:** Displays the name of the Excel 2010 worksheet, if an Excel file exists.
- **Other Key words:** Displays other keywords from your search.

Refinement categories are located to the left of the search page to help you filter the results.

Overview of Business Intelligence Indexing Connector search tab interface

Microsoft Business Intelligence Indexing Connector improves search capabilities for Microsoft Excel and for Microsoft SQL Server Reporting Services (SSRS) reports and the underlying data sources that are part of the report. Users see improved results, descriptions, thumbnails, and previews, and they are able to refine search results.

This article describes the elements of the Reports search page.



Reports tab

When the Business Intelligence Indexing Connector feature is installed on SharePoint Server 2010, a **Reports** tab is added to the **All Sites** and **People** tabs, and possibly other customized search tabs. The **Reports** tab helps you narrow search results to Excel and SQL Server Reporting Services files, and it takes advantage of the additional search features described in this document.

Results description

Business Intelligence Indexing Connector offers rich result descriptions. These descriptions are broken down into categories to give users a better understanding of the search results. If the categories **PivotTables**, **Tables**, **Charts**, or **Gauges** are found in a search, the keywords are highlighted and included within the context of where they are located. In the previous illustration, **Sheet1** is a link to the worksheets that contain the keywords "France" and "Sales".



Note:

A link appears for each worksheet in which keywords are found.

Additionally, the **Data Sources** description category lists the supported data sources found in the document. The **Other** description category displays other relevant information found in the document that is not covered by the previous categories.

The following are characteristic of keywords that are identified in Excel files and show how business intelligence artifacts can be crawled and identified by Business Intelligence Indexing Connector.

- Excel files include Office Excel 2007 and Excel 2010, (xlsx and xlsx files).
- Keywords located in Excel hidden rows and columns are identified and show the results description.
- Keywords are also identified from the Excel PivotTable filters even though they may not be visible. In the previous illustration, **(In Filter)** shows you the keyword **France** even though **France** may currently not be the selected filter value.
- **(In Row/Column)** appears after a highlighted keyword to show that the keyword is located in a PivotTable row or column. Likewise, **(In Categories/Series)** indicates that the keyword was found in a category or series in a Chart.

Document thumbnail

The document thumbnail is part of the results description of each result and provides a small visual representation of the actual document.



Note:

Document thumbnails only appear if the Excel and SQL Server Reporting Services documents are located in the Microsoft SQL Server PowerPivot for Microsoft SharePoint Reports Gallery.

Preview, duplicates, View in Browser

The **Preview** displays images for each sheet or page of the document. If the same document is located, **Duplicates** shows in parentheses how many of the same documents exist. When the **View in Browser** option is available, you can open the worksheet, within the Excel file, in the browser.



Note:

Microsoft Silverlight 3 security restrictions may prevent thumbnail images and preview from rendering in the results description. By default all requests, other than images and media, Silverlight 3 only allows for site-of-origin communication. For more information, see [Making a Service Available Across Domain Boundaries](http://go.microsoft.com/fwlink/?LinkId=190990) (<http://go.microsoft.com/fwlink/?LinkId=190990>) in the MSDN Library online.

Refinement categories

Refinement categories let you refine the results of your search when you select available filters. The following three additional refinement categories are included when Business Intelligence Indexing Connector is installed.

1. The **Data Sources** category lets you limit the results to reports that have data sources that the documents use.
2. The **Data Category** category limits the results to reports that have keywords that refer to metadata, such as dimension names and measure names in charts and PivotTables.
3. The **Report Format** category limits results to reports that have tables and charts.

For example, if you select **Table** as a format under the **Report Format** category, the results of your search are limited to show only files that contain tables. Likewise, if you select **Chart**, the search results show only those files that have charts in them.

Determine software requirements for Business Intelligence Indexing Connector

This article describes the software prerequisites for the Business Intelligence Indexing Connector front end (on Microsoft SharePoint Server 2010) and back end (on Microsoft FAST Search Server 2010 for SharePoint) installations. Before you can install the Business Intelligence Indexing Connector you must install SharePoint Server 2010 and FAST Search Server 2010 for SharePoint. To learn more, see [Overview of Business Intelligence Indexing Connector architecture](#).

This article does not describe software prerequisites for Microsoft SharePoint Server 2010 or FAST Search Server 2010 for SharePoint. You must review the documentation for each technology on which Business Intelligence Indexing Connector is installed. For more information, see:

- [Hardware and Software Requirements for Installing SQL Server 2008 R2](#) (<http://go.microsoft.com/fwlink/?LinkId=191076>)
- [Hardware and software requirements \(SharePoint Server 2010\)](#)
- [System requirements \(FAST Search Server 2010 for SharePoint\)](#)

Software requirements for Business Intelligence Indexing Connector: back end

You must install the following prerequisites before you install Business Intelligence Indexing Connector.

- Microsoft .NET Framework 3.5 Service Pack 1
- Microsoft SQL Server 2008 R2 Feature Pack:
 - SQL Server 2008 R2 ADOMD.NET
 - SQL Server 2008 R2 Analysis Management Objects
 - SQL Server 2008 R2 System CLR Types
- Open XML SDK 2.0 for Microsoft Office (OpenXMLSDKv2.msi)
- To crawl PowerPivot for Excel 2010 files, you must install the following components on FAST Search Server 2010 for SharePoint servers that are document processors:
 - Microsoft Excel 2010
 - PowerPivot for Excel 2010

Installing software prerequisites

To install the prerequisites for Business Intelligence Indexing Connector, you can go to the Web sites listed in this section.

- [Microsoft .NET Framework 3.5 Service Pack 1](#) (<http://go.microsoft.com/fwlink/?LinkId=131037>)

- [Microsoft SQL Server 2008 R2 Feature Pack](#)
- [Open XML SDK 2.0 for Microsoft Office](http://go.microsoft.com/fwlink/?LinkId=191094) (http://go.microsoft.com/fwlink/?LinkId=191094)
- [Microsoft SQL Server 2008 R2 PowerPivot for Microsoft Excel 2010](http://www.powerpivot.com/download.aspx) (http://www.powerpivot.com/download.aspx)

Software requirements for Business Intelligence Indexing Connector: front end

The following are software prerequisites for the Business Intelligence Indexing Connector front end. For related information, see [Hardware and software requirements \(SharePoint Server 2010\)](#).

- Microsoft SharePoint Server 2010 Enterprise
- Internet browser



Note:

To open documents in a file share by using Firefox, see [Links to local or network pages do not work](http://go.microsoft.com/fwlink/?LinkId=191096) (http://go.microsoft.com/fwlink/?LinkId=191096) for a workaround.

Install Microsoft Business Intelligence Indexing Connector — front end and back end

After you have installed the prerequisites, download Business Intelligence Indexing Connector for the front end and back end at the following sites.

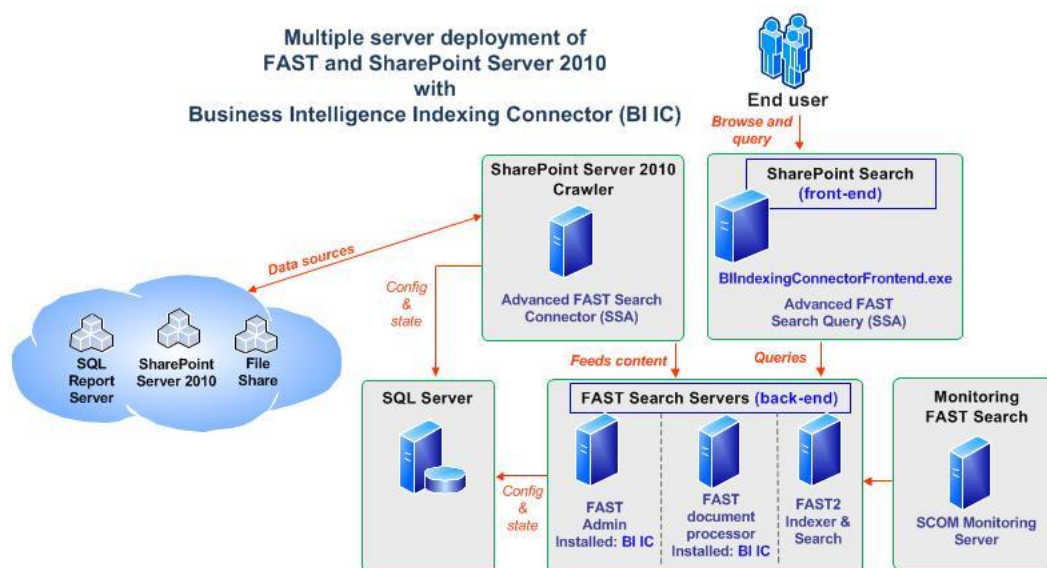
- [Microsoft Business Intelligence Indexing Connector – Back end](http://go.microsoft.com/fwlink/?LinkID=195026) (http://go.microsoft.com/fwlink/?LinkID=195026)
- [Microsoft Business Intelligence Indexing Connector – Front end](http://go.microsoft.com/fwlink/?LinkID=195027) (http://go.microsoft.com/fwlink/?LinkID=195027)

See Also

[Overview of Business Intelligence Indexing Connector architecture](#)

Overview of Business Intelligence Indexing Connector architecture

This article describes the topology for a multi-server deployment of Microsoft Business Intelligence Indexing Connector added to Microsoft FAST Search Server 2010 for SharePoint (the back end) and SharePoint Server (the front end). One server is defined as the admin server, where administrative services are installed and run. The deployment can be scaled out to create a multiple-server deployment by adding one or more non-admin servers. A non-admin server does not have the administrative services set up, but connects to a running admin server in a multiple-server deployment. The non-admin server runs services such as search, indexing, and document processing. For more information, see [Plan the search solution \(FAST Search Server 2010 for SharePoint\)](#).



Setup for FAST Search Server 2010 for SharePoint (back end)

The FAST Search Server 2010 for SharePoint admin and document-processor servers must have Business Intelligence Indexing Connector installed. In Business Intelligence Indexing Connector configuration wizard, you have the option to enable crawling functionality for external data sources. By default, the feature is disabled. Business Intelligence Indexing Connector extends the existing IFilter interface for Office on the FAST Search Server 2010 for SharePoint crawl servers. The following kinds of files are crawled and processed by the Business Intelligence Indexing Connector.

- Microsoft Office Excel 2007 and Microsoft Excel 2010 workbooks (.xlsx)

- Microsoft Macro-Enabled workbooks (.xlsm)
- SQL Server Reporting Services and Report Builder 3.0 reports (.rdl)

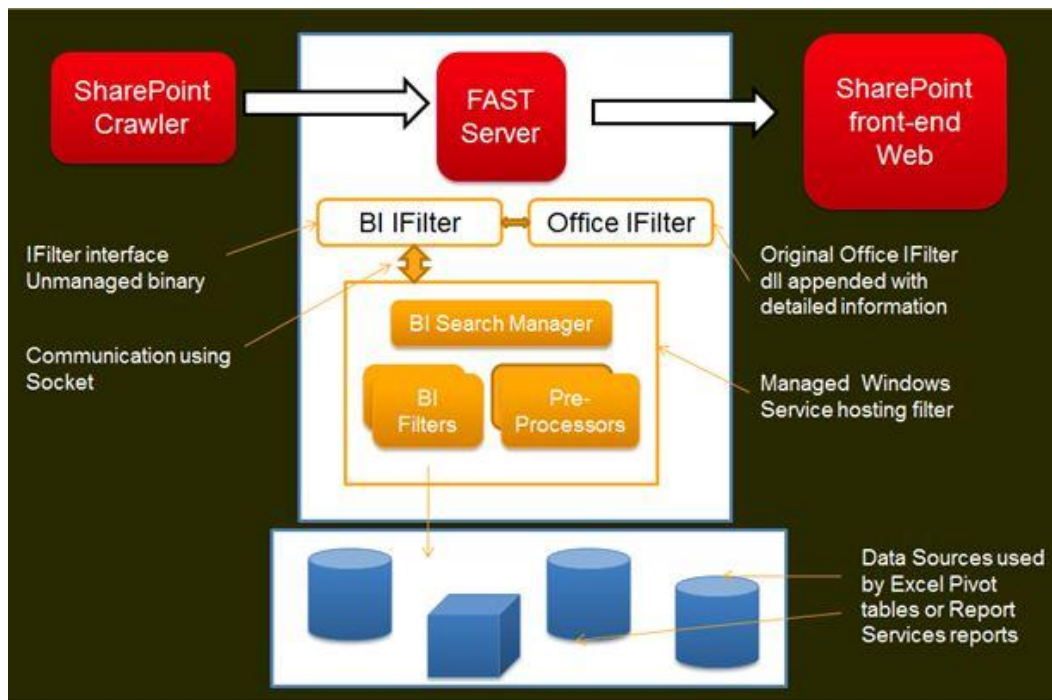
For more information about installation, see [Install and configure Business Intelligence Indexing Connector \(back end\)](#).

Setup for SharePoint Server 2010 (front end)

Business Intelligence Indexing Connector client setup is a feature that can be enabled on the SharePoint Server farm. It customizes the specified FAST search site. The setup occurs when you install the SharePoint Solution Pack. For more information, see [Install or uninstall SharePoint Solution Pack for Business Intelligence Indexing Connector \(front end\)](#).

Logical architecture

The following is a closer look at the architecture for Business Intelligence Indexing Connector.



Understanding planning solutions and scenarios (white paper)

This white paper provides direction and samples for developing a Business Intelligence solution with planning, budgeting, and forecasting capabilities for financial, human resources, and other departments inside an organization. The solution's architecture is designed to enable planning functions that include write-back capabilities to SQL Server Analysis Services cubes, SharePoint lists, and relational databases. This white paper also shows how the Microsoft platform can be used in developing financial calculations and business rules to link forms and reports data through Microsoft Excel. Workflow and security are also included as functions to enable collaboration in Microsoft SharePoint 2010 environments. Finally, building and maintaining a centralized data model is discussed.

[Download file](#)

Business data and processes planning (SharePoint Server 2010)

The topics in this section will help you plan and build solutions that integrate business processes and forms with your enterprise's data.

In this section:

- [Plan for Business Connectivity Services \(SharePoint Server 2010\)](#)
- [Plan InfoPath Forms Services \(SharePoint Server 2010\)](#)
- [Plan workflows \(SharePoint Server 2010\)](#)

Plan for Business Connectivity Services (SharePoint Server 2010)

Microsoft Business Connectivity Services are a set of services and features that provide a way to connect Microsoft SharePoint Server 2010 solutions to sources of external data and to define external content types that are based on that external data. Planning for Microsoft Business Connectivity Services should include thinking about the kinds of solutions your organization requires, and planning for security and administration of those solutions.

In this section:

- [Business Connectivity Services overview \(SharePoint Server 2010\)](#)
Defines the concept of Microsoft Business Connectivity Services and explains how solutions that are based on Microsoft Business Connectivity Services can take advantage of the integration of client applications, servers, services, and tools in the Microsoft Office 2010 suites.
- [Business Connectivity Services security overview \(SharePoint Server 2010\)](#)
Describes the security architecture of the Microsoft Business Connectivity Services server and client, the supported security environments, the authentication modes available to connect external content types to external systems, the authorization options available on stored objects, and the general techniques for configuring Microsoft Business Connectivity Services security
- [Business Data Connectivity service administration overview \(SharePoint Server 2010\)](#)
Provides an overview of administration of the Microsoft Business Connectivity Services and emphasizes on how to use the Business Data Connectivity service.
- [Plan Business Connectivity Services client integration \(SharePoint Server 2010\)](#)
Discusses how users can take external lists offline to Microsoft Outlook 2010 and Microsoft SharePoint Workspace 2010. This article also provides guidances on how users can interact with external systems from their Microsoft Office 2010 client applications more securely.
- [Diagnostic logging in Business Connectivity Services overview \(SharePoint Server 2010\)](#)
Describes how to use diagnostic logs on servers that are running Microsoft SharePoint Server 2010 and on Microsoft Office 2010 client applications to troubleshoot issues related to Microsoft Business Connectivity Services. It also describes how to use Activity IDs to trace issues to external systems.
- [Plan to upgrade to Business Connectivity Services \(SharePoint Server 2010\)](#)
Describes how upgrading from the Microsoft Office SharePoint Server 2007 Business Data Catalog to the Microsoft Business Connectivity Services in SharePoint Server 2010 works and provides guidance about how to upgrade various types of solutions that use the Business Data Catalog.

See Also

[Microsoft Business Connectivity Services](#)

Business Connectivity Services overview (SharePoint Server 2010)

Microsoft SharePoint Server 2010 and the Microsoft Office 2010 suites include Microsoft Business Connectivity Services, which are a set of services and features that provide a way to connect SharePoint solutions to sources of external data and to define external content types that are based on that external data. External content types resemble content types and allow the presentation of and interaction with external data in SharePoint lists (known as external lists), Web Parts, Microsoft Outlook 2010, Microsoft SharePoint Workspace 2010, and Microsoft Word 2010 clients. External systems that Microsoft Business Connectivity Services can connect to include SQL Server databases, SAP applications, Web services (including Windows Communication Foundation Web services), custom applications, and Web sites based on SharePoint. By using Microsoft Business Connectivity Services, you can design and build solutions that extend SharePoint collaboration capabilities and the Office user experience to include external business data and the processes that are associated with that data.

Microsoft Business Connectivity Services solutions use a set of standardized interfaces to provide access to business data. As a result, developers of solutions do not have to learn programming practices that apply to a specific system or adapter for each external data source. Microsoft Business Connectivity Services also provide the run-time environment in which solutions that include external data are loaded, integrated, and executed in supported Office client applications and on the Web server.

Typical solutions based on Business Connectivity Services

Solutions that are based on Microsoft Business Connectivity Services can take advantage of the integration of client applications, servers, services, and tools in the Microsoft Office 2010 suites. Information workers typically perform much of their work outside the formal processes of a business system. For example, they collaborate by telephone or e-mail messages, use documents and spreadsheets from multiple sources, and switch between being online and offline. Solutions that are based on Microsoft Business Connectivity Services can be designed to fit within these informal processes that information workers use:

- They can be built by combining multiple services and features from external data systems and from the Office 2010 suites to deliver solutions that are targeted to specific roles.
- They support informal interactions and target activities and processes that occur mostly outside formal enterprise systems. Because they are built by using SharePoint 2010 Products, solutions that are based on Microsoft Business Connectivity Services promote collaboration.
- They help users perform tasks within the familiar user interface of Office applications and SharePoint 2010 products.

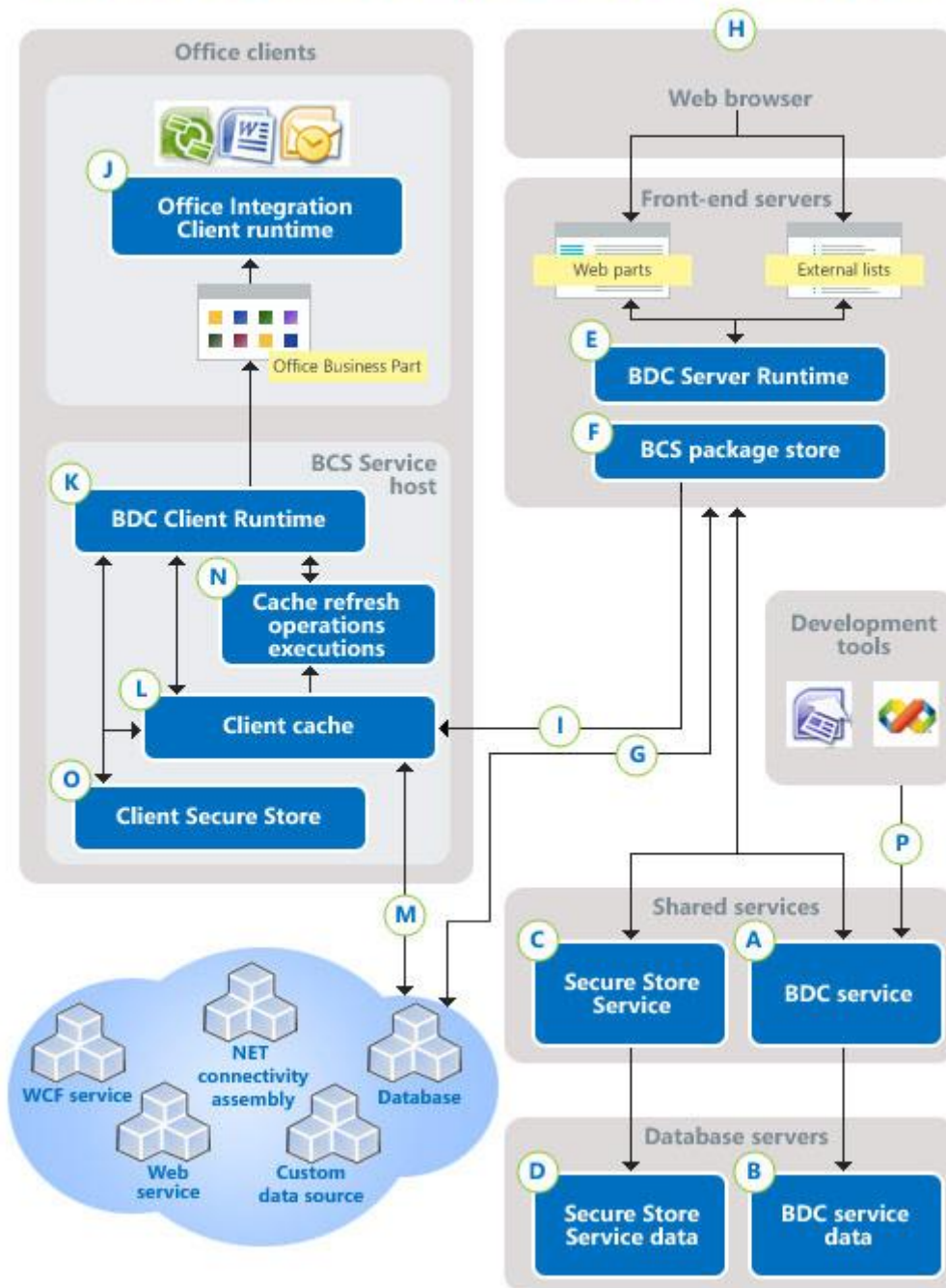
Here are some examples of solutions that are based on Microsoft Business Connectivity Services:

- **Help desk** An enterprise implements its help desk, which provides internal technical support, as a solution that is based on Microsoft Business Connectivity Services. Support requests and the technical support knowledge base are stored in external databases and are integrated into the solution by using the Business Data Connectivity service. The solution displays both support requests and the knowledge base in the Web browser. Information workers can view their current requests either in a Web browser or in Microsoft Outlook. Tech support specialists view the requests assigned to them in a browser, by using Microsoft Outlook, and, when offline, by using Microsoft SharePoint Workspace. Workflows take support issues through each of their stages. Managers on the technical support team can view dashboards that display help desk reports. Typical reports indicate the number of support issues assigned to each support specialist, the most critical issues currently, and the number of support incidents that are handled by each support specialist during a given time period.
- **Artist tracker** A talent agency integrates its database of artists into its internal Web site. The complete list of artists, their contact information, and schedules can be taken offline in Microsoft SharePoint Workspace or in Microsoft Outlook. Recording contracts can be generated and filled from the Web site, Microsoft SharePoint Workspace, or Microsoft Outlook, and a workflow guides each contract through its various stages. New artists can be added from the Web site or from Microsoft Outlook. By using this solution, agents always have the information that they need nearby and they can perform many key tasks by using familiar Office interfaces.
- **Sales Dashboard** A sales dashboard application helps sales associates in an organization quickly find the information that they need and enter new data. Sales orders and customer information are managed in an external database and integrated into the solution by using Microsoft Business Connectivity Services. Depending on their roles, team members can view sales analytics information, individual team members' sales performance data, sales leads, and a customer's contact information and orders. Sales professionals can view their daily calendars, view tasks assigned to them by their managers, collaborate with team members, and read industry news, either from a Web browser, from Microsoft Outlook, or offline in Microsoft SharePoint Workspace. By using Microsoft Word 2010, managers can author monthly status reports that include data from the external systems.

Business Connectivity Services architecture

The following illustration shows the architecture of Microsoft Business Connectivity Services.

Business Connectivity Services architecture



A: The Business Data Connectivity service (BDC service) provides a means for storing and securing external content types and related objects. An external content type defines the following:

- A named set of fields of data, such as "customer."
- The operations for interacting with the data's external system, such as read, write, and create.
- The connectivity information that lets solutions that use the external content type connect to the external system.

B: External content types are stored in a dedicated Business Data Connectivity service database.

C: The Secure Store Service securely stores credential sets for external systems and associates those credential sets with identities of individuals or with group identities. A common scenario for the Secure Store Service is a solution that authenticates against an external system in which the current user has a different account for authentication on that external system. When used with Business Data Connectivity service, the Secure Store Service provides a way to authenticate users and groups on external data sources.

D: Credential sets that the Secure Store Service uses are stored in a dedicated, secure database.

E: The Business Data Connectivity Server Runtime on front-end SharePoint servers uses the Business Data Connectivity data to connect to external systems and execute operations on the external systems for access by thin clients such as Web browsers.

F: The BCS Package Store holds the deployable Microsoft Business Connectivity Services packages that contain information that Microsoft Business Connectivity Services and Office applications need to interact with the external systems from rich clients.

G: The Business Data Connectivity service supports connecting to relational databases, Web services, Windows Communication Foundation (WCF) services, .NET connectivity assemblies, and custom data connectors that comply with the requirements of Microsoft Business Connectivity Services.

H: SharePoint Web sites use Business Data Web parts and SharePoint external lists to interact with external data that Microsoft Business Connectivity Services expose.

I: When users connect to an external list, a BCS package is downloaded to the client computers.

J: On supported Office clients (Microsoft SharePoint Workspace, Microsoft Outlook, and Microsoft Word), the Office Integration Client Runtime acts as a connector between Microsoft Business Connectivity Services running on the client and Office applications.

K: The Business Data Connectivity Client Runtime, on client computers, uses the Business Data Connectivity service data and Secure Store Service data to connect to and execute operations on external systems for access by supported rich clients.

L: Business Data Connectivity service data and Secure Store Service data is cached on client computers.

M: The Business Data Connectivity Runtime, on client computers, supports connecting to SQL Server and other relational databases, Web services, and custom data connectors that comply with the requirements of the Microsoft Business Connectivity Services.

N: The client cache refresh synchronizes the cache with the Microsoft Business Connectivity Services data and Secure Store Service data.

O: The Client Secure Store enables end users to configure their client mappings in the credential database.

P: Solution developers can use Microsoft SharePoint Designer 2010 and Visual Studio 2010 to create external content types and BDC models.

noteDXDOC112778PADS Security Note

We recommend that you use Secure Sockets Layer (SSL) on all channels between client computers and front end servers. Also we recommend using Secure Sockets Layer or Internet Protocol Security (IPSec) between servers running Microsoft SharePoint Server 2010 and external systems.

Business Connectivity Services security overview (SharePoint Server 2010)

This article describes the security architecture of the Microsoft Business Connectivity Services server and client, the supported security environments, the authentication modes available to connect external content types to external systems, the authorization options available on stored objects, and the general techniques for configuring Microsoft Business Connectivity Services security.

In this article:

- [About this article](#)
- [Business Connectivity Services security architecture](#)
- [Business Connectivity Services authentication overview](#)
- [Business Connectivity Service permissions overview](#)
- [Securing Business Connectivity Services](#)

About this article

Microsoft Business Connectivity Services include security features for authenticating users to access external systems and for configuring permissions on data from external systems. Microsoft Business Connectivity Services are highly flexible and can accommodate a range of security methods from within supported Microsoft Office 2010 applications and from the Web browser.

Business Connectivity Services security architecture

This section describes the Microsoft Business Connectivity Services security architecture when authenticating from a Web browser and when authenticating from a supported Office 2010 client application, such as Microsoft Outlook 2010.

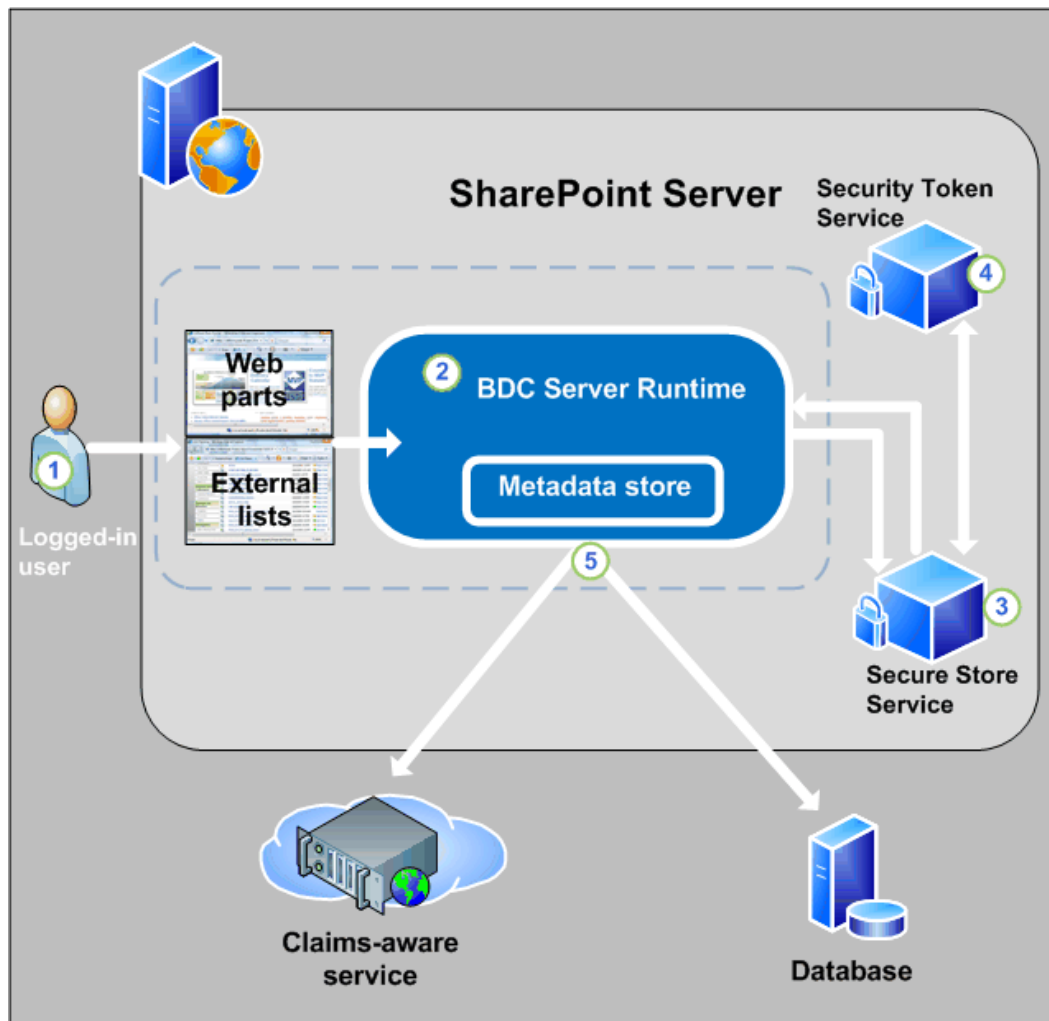


Security note:

We recommend that you use Secure Sockets Layer (SSL) on all channels between client computers and front end servers. Also we recommend using Secure Sockets Layer or Internet Protocol Security (IPSec) between servers running Microsoft SharePoint Server 2010 and external systems. An exception is that you cannot use SSL when transmitting messages to external systems using the SOAP 1.1 protocol or when connecting to a SQL server database. However, in those cases you can use IPSec to protect the data exchange.

Accessing external data from a Web browser

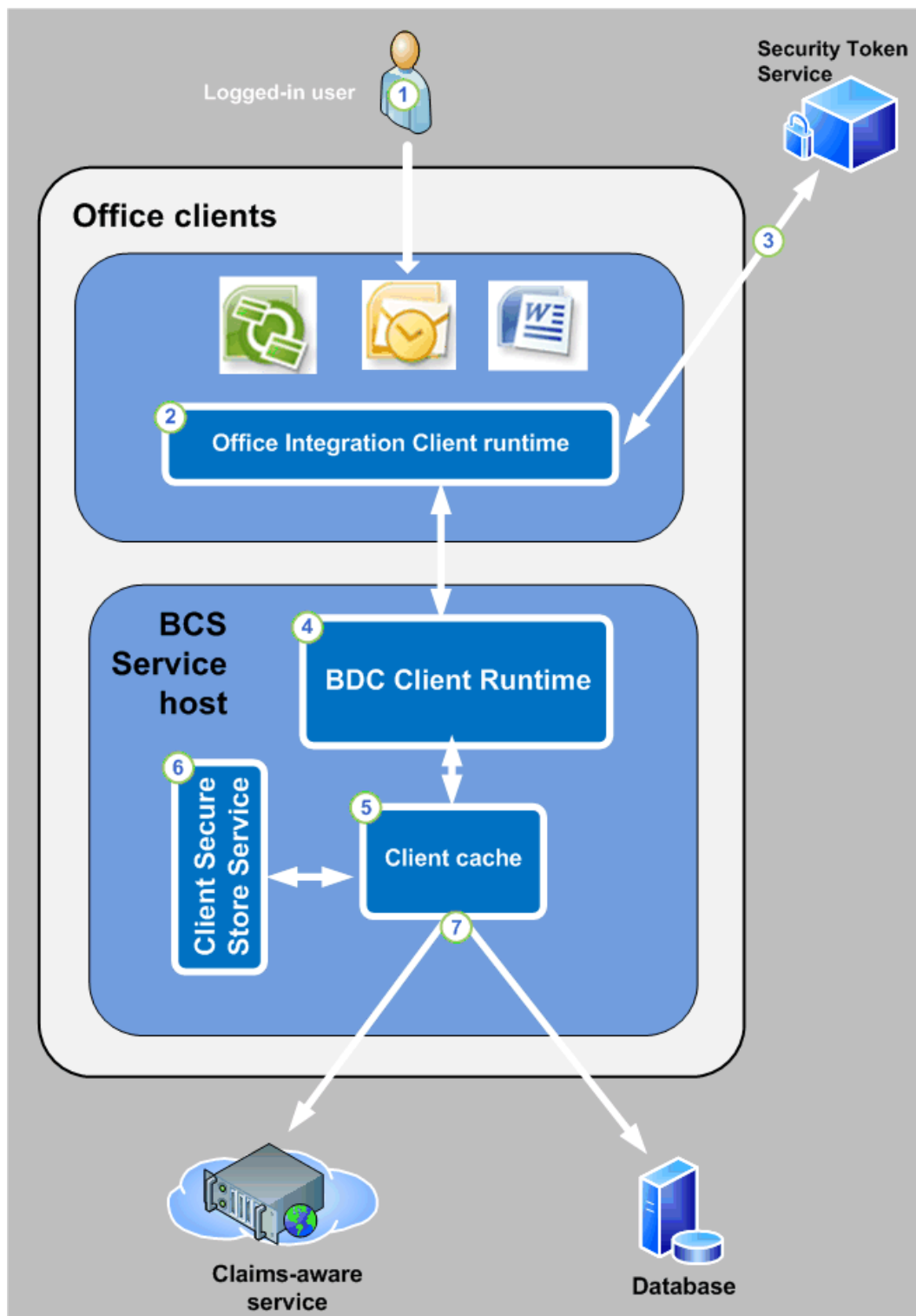
When a user accesses external data from a Web browser, three systems are involved: the logged on user's client computer, the Web server farm, and the external system.



1. From Web browsers, users typically interact with external data in external lists or by using Web Parts.
2. The BDC Server Runtime on front-end servers uses data from the Business Data Connectivity service to connect to and execute operations on external systems.
3. The Secure Store Service securely stores credential sets for external systems and associates those credential sets to individual or group identities.
4. The Security Token Service is a Web service that responds to authentication requests by issuing security tokens made up of identity claims that are based on user account information.
5. Microsoft Business Connectivity Services can pass credentials to databases and Web services that are configured to use claims-based authentication. For an overview of the Secure Store Service, see [Plan the Secure Store Service \(SharePoint Server 2010\)](#). For an overview of claims-based authentication, see [Plan authentication methods \(SharePoint Server 2010\)](#).

Accessing external data from an Office client application

When accessing external data from a supported Office client application, two systems are involved: the client computer of the logged on user and the external system. This model is supported when a user interacts with external data by using Outlook 2010, Microsoft SharePoint Workspace, or Microsoft Word 2010.



1. Outlook 2010 users typically use external data in Outlook items such as Contacts or Tasks. SharePoint Workspace 2010 users can take external lists offline and interact with them. Word 2010 users can insert external data into Word documents.
2. The Office Integration Client Runtime acts as a connector between Microsoft Business Connectivity Services running on the client and the supported Office applications.
3. If the external data is configured to use claims-based authentication, the client interacts with the Security Token Service on the SharePoint farm to get a claims token. (See [Configure claims authentication \(SharePoint Server 2010\)](#) for more information.)
4. The BDC Client Runtime on client computers uses the data from the Business Data Connectivity service to connect to and execute operations on external systems for rich client access.
5. The Client Cache caches information from the Business Data Connectivity service and Secure Store Service that is needed to securely connect to external data. The cache is refreshed from the SharePoint farm to incorporate updated information.
6. The client Secure Store Service enables end users to configure their security credentials.
7. Microsoft Business Connectivity Services can pass credentials to databases and claims aware services.

Business Connectivity Services authentication overview

Microsoft Business Connectivity Services can be configured to pass authentication requests to external systems by using the following types of methods:

- **Credentials** These are typically in the form of name/password. Some external systems may also require additional credentials such as a personal identification number (PIN) value.
- **Claims** Security Assertion Markup Language (SAML) tickets can be passed to claims-aware services that supply external data.

Configuring Business Connectivity Services for credentials authentication

Microsoft Business Connectivity Services can use credentials that a user supplies to authenticate requests for external data. The following methods by which users can supply credentials for accessing external data are supported:

- Windows authentication:
 - Windows Challenge/Response (NTLM)
 - Microsoft Negotiate
- Authentication other than Windows
 - Forms-based
 - Digest


- Basic


When configuring Microsoft Business Connectivity Services to pass credentials, the solution designer adds authentication-mode information to external content types. The authentication mode gives Microsoft Business Connectivity Services information about how to process an incoming authentication request from a user and map that request to a set of credentials that can be passed to the external content system. For example, an authentication mode could specify that the user's credentials be passed directly through to the external data system. Alternatively, it could specify that the user's credentials should be mapped to an account that is stored in a Secure Store Service which should then be passed to the external system.


You associate an authentication mode with an external content type in the following ways:

- When you create an external content type in Microsoft SharePoint Designer or Microsoft Visual Studio 2010.
- If the external system is a Web service, you can use the Microsoft Business Connectivity Services administration pages to specify the authentication mode.
- You can specify the authentication mode by directly editing the .XML file that defines the external content type.

The following table describes the authentication modes of the Microsoft Business Connectivity Services:

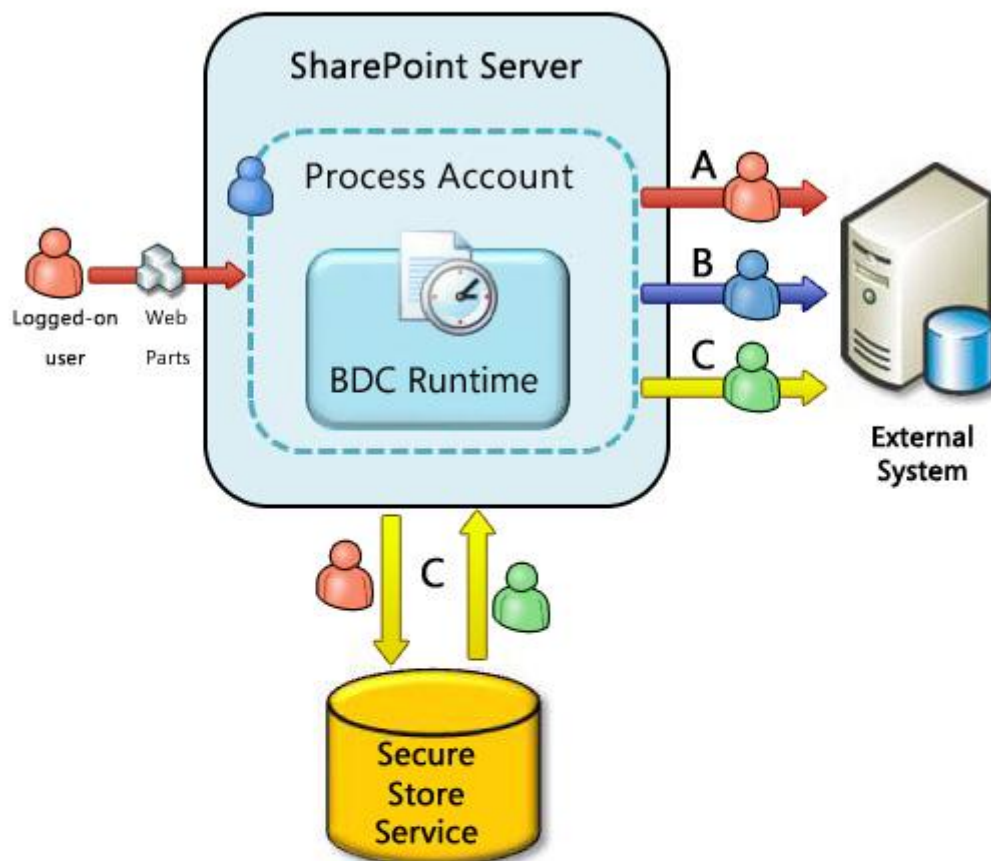
Authentication mode	Description
PassThrough	<p>Passes the credentials of the logged-on user to the external system. This requires that the user's credentials are known to the external system.</p> <p> Note: If the Web application is not configured to authenticate with Windows credentials, the NT Authority/Anonymous Logon account is passed to the external system rather than the user's credentials.</p> <p>This mode is called User's Identity in the Microsoft Business Connectivity Services administration pages and in SharePoint Designer 2010.</p>

Authentication mode	Description
RevertToSelf	<p>When the user is accessing external data from a Web browser, this mode ignores the user's credentials and sends the application pool identity account under which the BCS runtime is running on the Web server to the external system. When the user is accessing external data from an Office client application, this mode is equivalent to PassThrough mode, because Microsoft Business Connectivity Services running on the client will be running under the user's credentials.</p> <p>This mode is called BDC Identity in the Microsoft Business Connectivity Services administration pages and in SharePoint Designer 2010.</p> <p> Note:</p> <p>By default, RevertToSelf mode is not enabled. You must use Windows PowerShell to enable RevertToSelf mode before you can create or import models that use RevertToSelf. For more information, see RevertToSelf authentication mode. RevertToSelf mode is not supported in hosted environments.</p>
WindowsCredentials	<p>For external Web services or databases, this mode uses a Secure Store Service to map the user's credentials to a set of Windows credentials on the external system.</p> <p>This mode is called Impersonate Windows Identity in the Microsoft Business Connectivity Services administration pages and in SharePoint Designer 2010.</p>

Authentication mode	Description
Credentials	<p>For an external Web service, this mode uses a Secure Store Service to map the user's credentials to a set of credentials that are supplied by a source other than Windows and that are used to access external data. The Web service should use basic or digest authentication when this mode is used.</p> <p> Important: To help preserve security in this mode, we recommend that the connection between the Microsoft Business Connectivity Services and the external system should be secured by using Secure Sockets Layer (SSL) or Internet Protocol Security (IPSec).</p> <p>This mode is called Impersonate Custom Identity in the Microsoft Business Connectivity Services administration pages and in Office SharePoint Designer.</p>
RDBCredentials	<p>For an external database, this mode uses a Secure Store Service to map the user's credentials to a set of credentials that are supplied by a source other than Windows. To help preserve security in this mode, we recommend that the connection between the Microsoft Business Connectivity Services and the external system should be secured by using Secure Sockets Layer (SSL) or IPSec.</p> <p>This mode is called Impersonate Custom Identity in the Microsoft Business Connectivity Services administration pages and in Office SharePoint Designer.</p>

Authentication mode	Description
DigestCredentials	<p>For a WCF Web service, this mode uses a Secure Store Service to map the user's credentials to a set of credentials using Digest authentication.</p> <p>This mode is called Impersonate Custom Identity – Digest in the Microsoft Business Connectivity Services administration pages and in SharePoint Designer 2010.</p>

The following illustration shows the Microsoft Business Connectivity Services authentication modes when it uses credentials.



SSS authentication modes:

- Windows Credentials
- RdbCredentials
- Credentials

SSS account mapping

- Group Credentials



- Individual Credentials




Microsoft Confidential


A : PassThrough mode

B : RevertToSelf mode

C : SSS authentication

 : Process account

 : Logged-on user

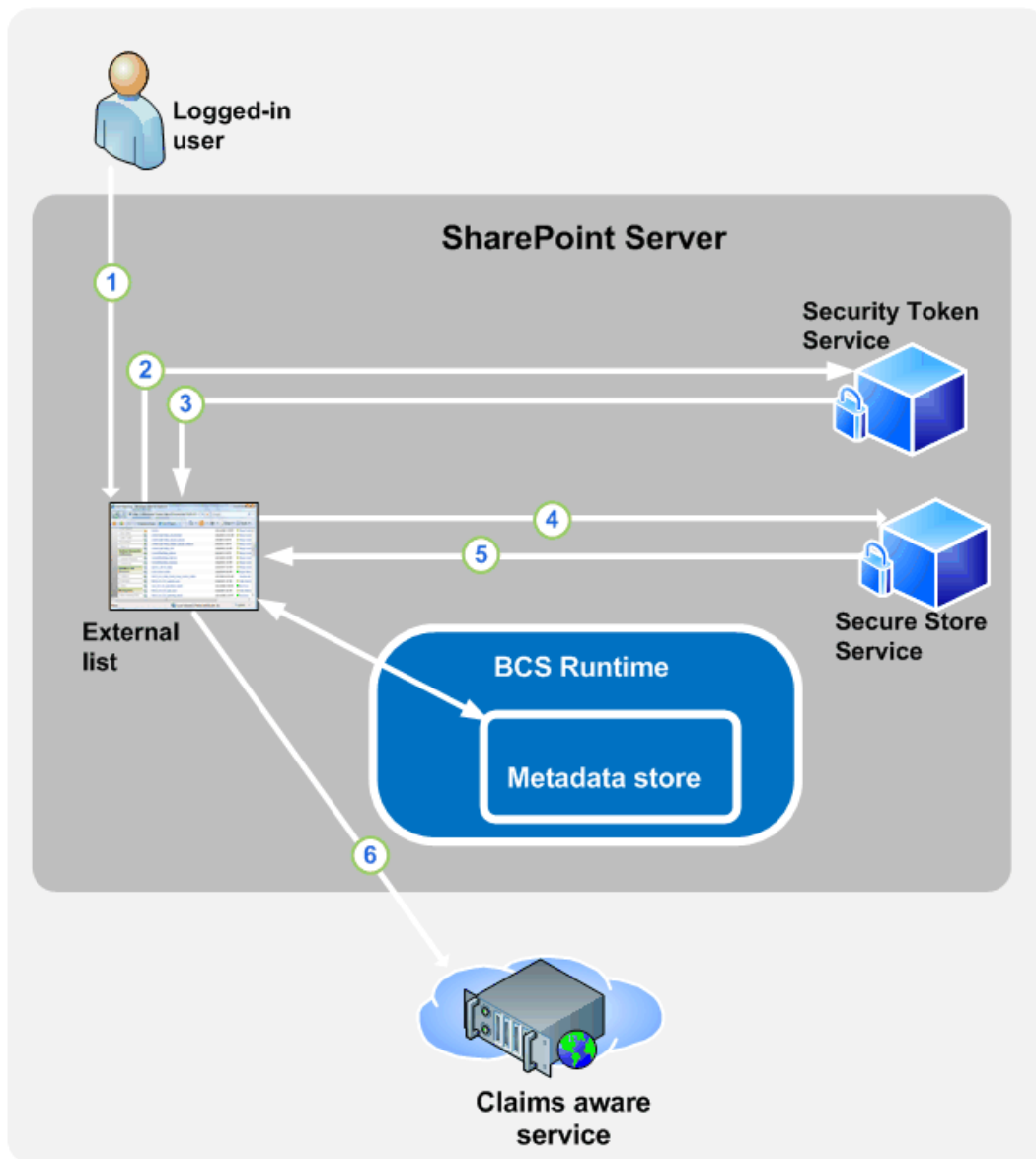
 : External system credentials

- In PassThrough (User's Identity) mode (A) the logged-on user's credentials are passed directly to the external system.
- In RevertToSelf (BDC Identity) mode (B) the user's logon credentials are replaced with the credentials of the process account under which Microsoft Business Connectivity Services is running, and those credentials are passed to the external system.
- Three modes use the Secure Store Service: WindowsCredentials (Impersonate Windows ID,) RdbCredentials (Impersonate Custom ID,) and Credentials. In those modes, the user's credentials are mapped to a set of credentials for the external system and Microsoft Business Connectivity Services passes those credentials to the external system. Solution administrators can either map each user's credentials to a unique account on the external system or they can map a set of authenticated users to a single group account.

Configuring Business Connectivity Services for claims-based authentication

Microsoft Business Connectivity Services can provide access to external data based on an incoming security tokens and it can pass security tokens to external systems. A security token is made up of a set of identity claims about a user, and the use of security tokens for authentication is called "claims-based authentication." SharePoint Server 2010 includes a Security Token Service that issues security tokens.

The following illustration shows how the Security Token Service and the Secure Store Service work together in claims-based authentication:



1. A user tries an operation on an external list that is configured for claims authentication.
2. The client application requests a security token from the Secure Token Service.
3. Based on the requesting user's identity, the Secure Token Service issues a security token that contains a set of claims and a target application identifier. The Secure Token Service returns the security token to the client application.
4. The client passes the security token to the Secure Store Service.
5. The Secure Store Service evaluates the security token and uses the target application identifier to return a set of credentials that apply to the external system.

6. The client receives the credentials and passes them to the external system so that an operation (such as retrieving or updating external data) can be performed.

Business Connectivity Service permissions overview

Permissions in Microsoft Business Connectivity Services associate an individual account, group account, or claim with one or more permission levels on an object in a metadata store. By correctly setting permissions on objects in Microsoft Business Connectivity Services, you help enable solutions to securely incorporate external data. When planning a permissions strategy, we recommend that you give specific permissions to each user or group that needs it, in such a way that the credentials provide the least privilege needed to perform the needed tasks.



Caution:

Properly setting permissions in Microsoft Business Connectivity Services is one element in an overall security strategy. Equally important is securing the data in external systems. How you do this depends on the security model and features of the external system and is beyond the scope of this article.

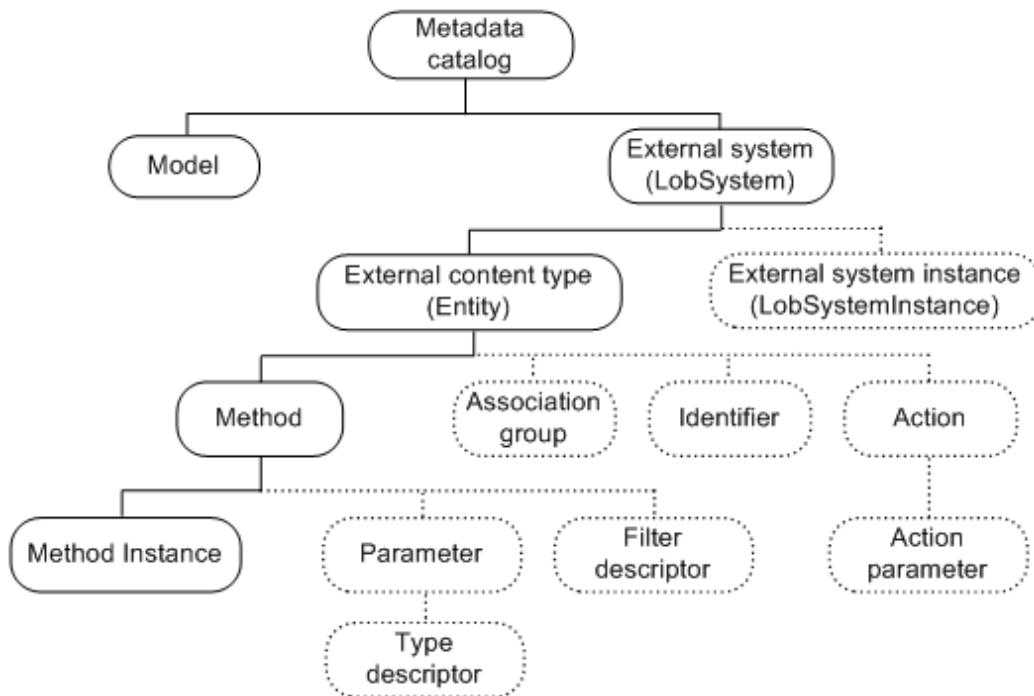


Note:

Business Connectivity Services uses the permissions on the metadata objects and the permissions on the external system to determine authorization rules. For example, a security trimmer can keep external data from appearing in users' search results. However, if users somehow discover the URL to the trimmed external data, they can access the external data if they have the necessary permissions to the metadata object and the external system. The correct way to prevent users from accessing external data is to set the appropriate permissions both in Business Connectivity Services and in the external system.

What can permissions be set on?

Each instance of the Business Data Connectivity service (or, in the hosting case, each partition) contains a metadata store that includes all the models, external systems, external content types, methods, and method instances that have been defined for that store's purpose. These objects exist in a hierarchy as depicted in the following illustration:



Note:

In the previous hierarchy graphic, labels in parentheses are the names of objects as they are defined in the Microsoft Business Connectivity Services metadata schema. The labels that are not in parentheses are the names of each object as it appears in the user interface of the Business Data Connectivity service. For a full discussion of the Microsoft Business Connectivity Services metadata schema, along with walkthroughs of many development tasks, see the [Microsoft SharePoint 2010 Software Development Kit](http://go.microsoft.com/fwlink/?LinkId=166117&clcid=0x409) (<http://go.microsoft.com/fwlink/?LinkId=166117&clcid=0x409>).

The hierarchy of objects in a metadata store determines which objects can propagate their permissions to other objects. In the illustration, each object on which permissions can be set, and optionally propagated, is shown with a solid line; each object that takes its permissions from its parent object is shown with a dotted line. For example, the illustration shows that an External System (LobSystem) can be secured by assigning permissions to it, but an Action cannot be assigned permissions directly. Objects that cannot be assigned permissions take the permissions of their parent object. For example, an Action takes the permissions of its parent External Content Type (Entity).



Security Note:

When the permissions on an object in a metadata store are propagated, permission settings to all children of that item are replaced by the permissions of the propagating object. For example,

if permissions are propagated from an External Content Type, all Methods and Method Instances of that External Content Type receive the new permissions.

Four permission levels can be set on the metadata store and the objects it contains:

- Edit



Security Note:

The Edit permission should be considered highly privileged. With the Edit permission, a malicious user can steal credentials or corrupt a server farm. We recommend that, in a production system, you give Edit permission only to users whom you trust to have administrator-level permissions.

- Execute
- Selectable in clients
- Set permissions

The following table defines the meaning of these permissions on the various objects for which they can be set.

Object	Definition	Edit permissions	Execute permissions	Selectable in clients permissions	Set permissions permissions
Metadata store	The collection of XML files, stored in the Business Data Connectivity service, that each contain definitions of models, external content types, and external systems.	The user can create new external systems.	Although there is no “Execute” permission on the metadata store itself, this setting can be used to propagate Execute permissions to child objects in the metadata store.	Although there is no “Selectable in clients” permission on the metadata store itself, this setting can be used to propagate these permissions to child objects in the metadata store.	The user can set permissions on any object in the metadata store by propagating them from the metadata store.

Object	Definition	Edit permissions	Execute permissions	Selectable in clients permissions	Set permissions permissions
Model	An XML file that contains sets of descriptions of one or more external content types, their related external systems, and information that is specific to the environment, such as authentication properties.	The user can edit the model file.	The “Execute” permission is not applicable to models.	The “Selectable in clients” permission is not applicable to models.	The user can set permissions on the model.
External system	The metadata definition of a supported source of data that can be modeled, such as a database, Web service, or .NET connectivity assembly.	The user can edit the external system. Setting this permission also makes the external system and any external system instances that it contains visible in SharePoint Designer.	Although there is no “Execute” permission on an external system itself, this setting can be used to propagate Execute permissions to child objects in the metadata store.	Although there is no “Selectable in clients” permission on an external system itself, this setting can be used to propagate these permissions to child objects in the metadata store.	The user can set permissions on the external system.

Object	Definition	Edit permissions	Execute permissions	Selectable in clients permissions	Set permissions permissions
External content type	A reusable collection of metadata that defines a set of data from one or more external systems, the operations available on that data, and connectivity information related to that data.	Although there is no “Edit” permission on an external content type itself, this setting can be used to propagate these permissions to child objects in the metadata store.	The user can execute operations on the external content type.	The user can create external lists of the external content type.	The user can set permissions on the external content type.
Method	An operation related to an external content type.	The user can edit the method.	Although there is no “Execute” permission on a method itself, this setting can be used to propagate Execute permissions to child objects in the metadata store.	There is no “Selectable in clients” permission on a method.	The user can set permissions on the method.
Method instance	For a particular method, describes how to use a method by using a specific set of default values.	The user can edit the method instance.	The user can execute the method instance.	There is no “Selectable in clients” permission on a method instance.	The user can set permissions on the method instance.

Special permissions on the Business Data Connectivity service

Along with the general capabilities of setting permissions described earlier, there is a set of special permissions for the Business Data Connectivity service:

- **Farm administrators** have full permissions to the Business Data Connectivity service. This is necessary, for example, to be able to maintain or repair an instance of the service. However, be aware that the farm administrator does not have execute permissions on any object in the metadata store and this right must be given explicitly by an administrator of an instance of the Business Data Connectivity service if it is required.
- **Windows PowerShell** users are farm administrators and can run commands on the Business Data Connectivity service.
- **Application pool accounts on front end servers** have the same permissions to the Business Data Connectivity service as farm administrators. This permission is necessary to generate deployment packages based on Microsoft Business Connectivity Services.
- **SharePoint Designer users** should, in most cases, be given the following permissions on the whole metadata store: Edit, Execute, and Selectable in clients. SharePoint Designer users should not be given Set permissions permissions. If necessary, you can limit the permissions of the SharePoint Designer user to a subset of the metadata store.



Caution:

To help ensure a secure solution, SharePoint Designer should be used to create external content types in a test environment in which Edit permissions can be assigned freely. When deploying the tested solution to a production environment, remove the edit permissions to help protect the integrity of the external data.

Common tasks and their related permissions

This section describes common tasks in the Business Data Connectivity service and the required permissions to perform them.

Task	Permissions
Create a new object in the metadata store	To create a new metadata object, a user must have edit permissions on the parent metadata object. For example, to create a new method in an external content type, a user must have permissions on the external content type. See the illustration earlier in this article for child/parent relationships among objects in the metadata store.
Delete an object from the metadata store	To delete a metadata object, a user must have edit permissions on that object. To delete an object and all its child objects (such as deleting an external content type and all its methods) the edit permission is also required on all the child objects.

Task	Permissions
Adding an external content type to a model	To add an external content type to a model, a user must have edit permissions on the model.
Importing models	To import a model to the metadata store, a user must have edit permissions on the metadata store. If explicit permissions are not assigned on the model, the user who imported it will be given edit permissions on the model.
Exporting models	To export a model from the metadata store, a user must have edit permissions on the model and on all external systems contained in the model.
Generating a deployment package	Deployment packages are generated by the application pool account that is used by the front-end server. This account has full permissions to the metadata store so that it can perform this task.
Setting initial permissions on the metadata store.	When an instance of the Business Data Connectivity service is first created, its metadata store is empty. The farm administrator has full permissions to the store and can set initial permissions.
Generate a deployment package from a consuming farm.	The Business Data Connectivity service application can be shared across server farms. In order to generate a deployment package from a consuming farm (the farm that connects to a remote location to use the Business Data Connectivity service application), the application pool account that is used by the front-end server of the consuming farm must have permissions to the metadata store on the publishing farm. For more information about how to assign permissions to the application pool account, see Set permissions to enable a consuming farm to generate a deployment package .

Securing Business Connectivity Services

This section discusses additional measures that can be used to help secure Business Connectivity Services

Service account

For security isolation, the Business Data Connectivity service application and the front-end server should not use the same service account.

Server to server communication

Securing the communication between the Business Data Connectivity service application and external systems helps ensure that sensitive data is not compromised. You need to use an encrypted communication channel to protect data that is sent between servers running SharePoint Server 2010 and external systems. Internet Protocol security (IPsec) is one method that can be used to help protect communication. The choice of which method to use depends on the specific communication channels you are securing and the benefits and tradeoffs that are most appropriate for your organization.

Applications that use FileBackedMetadataCatalog

For security reasons, RevertToSelf authentication mode is disabled on SharePoint Server 2010 by default. However, this does not prevent applications that use the FileBackedMetadataCatalog class from importing models and executing calls that use RevertToSelf authentication. This can result in elevating privileges for users by granting privileges to the application pool account. You should review all applications to ensure that they do not use FileBackedMetadataCatalog class and RevertToSelf authentication before installing them on a production system.

Business Data Connectivity service administration overview (SharePoint Server 2010)

Microsoft Business Connectivity Services are administered primarily by using the Business Data Connectivity service. This article provides an overview of Microsoft Business Connectivity Services administration with emphasis on how to use the Business Data Connectivity service.

Some configuration of Microsoft Business Connectivity Services, such as creating new target applications and providing credentials for those applications, is also done in the Secure Store Service. For an overview of the Secure Store Service, see [Plan the Secure Store Service \(SharePoint Server 2010\)](#). For descriptions of Secure Store Service operations that support Microsoft Business Connectivity Services, see [Configure the Secure Store Service \(SharePoint Server 2010\)](#).

The Business Data Connectivity service

The Business Data Connectivity service is a shared service and takes advantage of the SharePoint 2010 Products shared services architecture. In Microsoft SharePoint Server 2010, services are not contained within a Shared Services Provider (SSP) as they were in Microsoft Office SharePoint Server 2007. Instead, the infrastructure for hosting services has been moved into SharePoint Foundation 2010 and is included with SharePoint Server 2010. The configuration of services is more flexible. Individual services can be configured independently by different sets of administrators. Multiple instances of the same service, such as the Business Data Connectivity service, can run on the same farm, each with a unique set of administrators.

An instance of the Business Data Connectivity service can be shared across server farms. For example, a Business Data Connectivity service can be run in a central farm and accessed from regional locations so that the same solution is available across these locales.

Within a server farm, you deploy service applications such as the Business Data Connectivity service, by one of the following methods:

- Selecting services while running the Initial Configuration Wizard and choosing the Business Data Connectivity service.
- Adding services one by one on the Manage Service Applications page in the Central Administration Web site.
- Using Windows PowerShell.

Shared services such as the Business Data Connectivity service can each be administered in isolation. The administrators of a particular instance of a shared service may only have permissions to administer that service instance and are not necessarily able to administer other services or other features in the Central Administration Web site. This feature, called *delegated administration*, allows administration to

be managed by administrators who have expertise in the particular service being administered but who are not members of the central IT organization. Thus, for example, an administrator of a Business Data Connectivity service application in an enterprise might be familiar with the following information:

- The particular external content types being managed by that Business Data Connectivity service application
- The solutions supported by it
- The security implemented on the external data sources that provide the data

The administrator would have permissions to administer those objects but would not have permissions to administer other elements of the SharePoint deployment.

For more information about shared services, see the two model posters: [c](http://go.microsoft.com/fwlink/?LinkId=177411)

(<http://go.microsoft.com/fwlink/?LinkId=177411>) and [Cross-farm Services in SharePoint 2010 Products](http://go.microsoft.com/fwlink/?LinkId=177412) (<http://go.microsoft.com/fwlink/?LinkId=177412>).

What can be administered in the Business Data Connectivity service?

Using the Business Data Connectivity service, administrators can manage the following types of objects:

- **External content types:** An external content type is a named set of fields, such as “Customer,” “Order,” or “Contact,” that define an object in a business application along with the methods to create, read, update, or delete that object in its external data source.

Typical tasks that administer an external content type include setting an external content type’s permissions, adding actions to an external content type to provide users with new functionality, and associating profile pages with an external content type to customize its appearance when viewed.

For information about managing external content types, see [Manage external content types \(SharePoint Server 2010\)](#).

- **External systems and external system instances:** An external system is a supported source of data, such as a Web service, SQL Server database, and other relational databases, that can be modeled by the Microsoft Business Connectivity Services. An instance of an external system includes connection and authentication information for a specific instance of an external data source.

Typical tasks that administer an external system include setting permissions on the external system, viewing the external content types that are associated with it, and viewing instances of an external system. Typical tasks that administer an instance of an external system include setting the authentication mode and the type of the external system instance.

For information about managing external systems and external system instances, see [Manage external systems \(SharePoint Server 2010\)](#).

- **BDC models and resource files:** The Business Data Connectivity service supports two types of XML application definition files: application models and resource files. An application model contains the XML descriptions of one or more external content types.

A resource file enables you to import or export only the localized names, properties, and permissions for one or more external content types. The types of information that can be contained in a resource file include the following:

- Localized names for the external content types in a particular locale. Localized names that you import are merged with the existing localized names in the Business Data Connectivity service database. If a particular localized name for a locale already exists, it is overwritten with the new information.
- Properties for external content types. Properties that you import are merged with the existing property description in the Business Data Connectivity service database. If a property already exists, its value is overwritten with the information from the imported file.
- Permissions, which are access control lists (ACLs) for external content types. Permissions that you import are stored along with the existing permissions information in the Business Data Connectivity service database.

Typical tasks that administer a BDC model include importing and exporting models or resource files, setting permissions on them, and viewing the external content types associated with a model.

For information about managing BDC models, see [Manage BDC models \(SharePoint Server 2010\)](#).

Plan Business Connectivity Services client integration (SharePoint Server 2010)

Microsoft Business Connectivity Services offers users many ways of interacting with external systems from their Microsoft Office 2010 client applications. This article discusses how users can take external data offline to Microsoft Outlook 2010 and Microsoft SharePoint Workspace 2010.

When a user clicks the **Connect to Outlook** or **Sync to SharePoint Workspace** button on an external list, a ClickOnce application deployment package is created and installed on the client computer. This enables users to work with external data as native Outlook Item types (for example, Contacts, Tasks, and Appointments) in Outlook and as lists in SharePoint Workspace. Depending on their permissions, users can perform read and write operations on the external data, even when they are working offline or if the external system connectivity is slow, intermittent, or unavailable. The external data is synchronized when connection to the server becomes available.

The ability to take external lists offline takes advantage of the native capabilities of Business Connectivity Services, Microsoft SharePoint Server 2010, and Office 2010 applications. You can build more advanced Business Connectivity Services solutions that use customizing features or code.

For more information about advanced Business Connectivity Services solutions, see [Building Solutions with Business Connectivity Services](http://go.microsoft.com/fwlink/?LinkID=202359) (http://go.microsoft.com/fwlink/?LinkID=202359).

In this article:

- [Prerequisites](#)
- [Installing deployment packages](#)
- [Security considerations](#)

Prerequisites

The server must have Microsoft SharePoint Server 2010 with an Enterprise client access license (CAL) installed. The client computer must have Microsoft Office Professional Plus 2010 installed.

The following list describes additional client computer requirements:

- **Internet Explorer** The deployment mechanism uses ActiveX controls. Because Internet Explorer is the only browser that supports ActiveX controls, taking external lists offline is supported only in Internet Explorer. If you use another browser, such as Firefox, the **Connect to Outlook** and **Sync to SharePoint Workspace** buttons are disabled.
- **Microsoft .NET Framework 3.5** The Microsoft .NET Framework 3.5 or a later version must be installed on the client computer.
- **Business Connectivity Services** By default, the Business Connectivity Services feature is installed when Office Professional Plus 2010 is installed. If the .NET Framework 3.5 is not installed when a user installs Office, Business Connectivity Services will not be installed. After the .NET

Framework 3.5 is installed on the client computer, the Business Connectivity Services feature is installed when the user first takes an external list offline, and then the deployment package is installed. If the Business Connectivity Services feature was disabled by the user, the user must update his or her Office installation and enable the Business Connectivity Services feature. The Business Connectivity Services feature is available in the **Office Shared** group.

Installing deployment packages

The following sections discuss settings that can affect deployment package installations.

ClickOnce applications and trust-prompt behavior

The deployment packages are ClickOnce applications. All the rules, regulations, and limitations that govern general ClickOnce applications apply to the deployment packages also. The ClickOnce security model relies on trusted publishers and user prompting to determine whether a ClickOnce application will be installed on the client computer. ClickOnce applications are signed with a certificate that identifies the publisher. The certificates provide the following basis for making trust decisions:

- If the ClickOnce application is signed by a trusted publisher, the application will automatically be installed. The user is not prompted.
- If the ClickOnce application is not signed by a trusted publisher, ClickOnce does not automatically trust the application. The user is prompted to confirm that he or she wants to install the application.



Note:

By default, Business Connectivity Services uses a self-signed certificate to sign its deployment packages. Because the certificate is self-signed, it is not from a trusted certification authority (CA).

However, trust prompting can be affected by other settings, such as the Internet Explorer security zone that the ClickOnce application is being installed from. The following table lists example paths and URLs, their corresponding security zones, and the default trust-prompt behavior.

ClickOnce application URL or path	Security zone	Default trust-prompt behavior
C:\Contoso\Clientsolution\Customer.vsto	My Computer	Allow user prompting.
http://contoso/clientsolution/customer.vsto	Local intranet	Allow user prompting.
\\contoso\clientsolution\customer.vsto	Local intranet	Allow user prompting.
http://fabrikam.contoso/clientsolution/customer.vsto	Internet	No user prompting allowed unless the application is signed by a certificate that is

ClickOnce application URL or path	Security zone	Default trust-prompt behavior
		issued by a trusted CA.
http://www.contoso.com/clientsolution/customer.vsto	Internet	No user prompting allowed unless the application is signed by a certificate that is issued by a trusted CA.
\\172.16.4.1\clientsolution\customer.vsto	Internet	No user prompting allowed unless the application is signed by a certificate that is issued by a trusted CA.

The following list describes some things that can be done to stop deployment failures that are caused by default trust prompts.

- Sign the deployment packages with a trusted certificate** By default, Business Connectivity Services uses a self-signed certificate to sign its deployment packages. As a result, users will either be prompted to confirm that they want to install the application, or the deployment package will fail to install with no user prompts (if the external list resides in the Internet security zone). To resolve these issues, you can provide a certificate issued by a trusted CA that can be used to sign the deployment packages. For more information about how to provide a trusted certificate, see [How to: Get Rid of the Publisher Cannot Be Verified Alert When Taking External Lists Offline](http://go.microsoft.com/fwlink/?LinkID=202362) (<http://go.microsoft.com/fwlink/?LinkID=202362>).
- Users can add the SharePoint site to their list of trusted sites in Internet Explorer** Adding a site to the Internet Explorer list of trusted sites changes the deployment package security zone to the Trusted zone. The Trusted zone allows user prompting. If you have a deployment package that is not signed with a trusted certificate and resides in the Internet security zone, adding the site to the list of trusted sites gives the user the opportunity to decide whether to install the deployment package.



Note:

This action should be taken only for sites that the user can trust.

- Internet Explorer Enhanced Security Configuration** Internet Explorer Enhanced Security Configuration restricts the ability of users to browse Internet and intranet Web sites. This can cause deployment packages to fail to install without any errors displayed. As a workaround, you can do any of the following:
 - Sign the deployment packages with a trusted certificate.
 - Users can add the SharePoint site to their list of trusted sites in Internet Explorer.

- Turn off Internet Explorer Enhanced Security Configuration for users.

For more information about ClickOnce applications, see [ClickOnce Security and Deployment](http://go.microsoft.com/fwlink/?LinkId=195784) (<http://go.microsoft.com/fwlink/?LinkId=195784>).

Secure Store Service group mappings

Secure Store Service application IDs are used to map users to credential sets. Mappings are available for groups or individuals. In a group mapping, every user who is a member of a specific domain group is mapped to the same set of credentials. In an individual mapping, each individual user is mapped to a unique set of credentials.

If the external content type that is associated with an external list uses a group mapping, when users attempt to take the external list offline they are prompted for the group credentials. In most cases, users will not know the group credentials and will be unable to take the external list offline.

You can do one of the following:

- Modify the external content type to use an individual mapping.
- Modify the external content type to prevent users from trying to take the external list offline. Open the external content type in SharePoint Designer and set the **Offline Sync for external list** field to **Disabled**. This disables the **Connect to Outlook** and **Sync to SharePoint Workspace** buttons in the external list ribbon.

For more information about the Secure Store Service, see [Configure the Secure Store Service \(SharePoint Server 2010\)](#).

Sign in as Different User

When you are using Windows authentication, the **Sign in as Different User** feature is not supported for installing deployment packages. You cannot take an external list offline if you are logged on to a client computer by using one account and then log on to the SharePoint site by using a different user account. To take an external list offline, you must use the same user account to log on to both the client computer and the SharePoint site.

Security considerations

The following sections discuss additional measures that can be used to help secure Business Connectivity Services when you are working with rich client applications.

Secure communications

We recommend that you use Secure Sockets Layer (SSL) on all channels between client computers and front-end Web servers. This helps ensure that sensitive data is not compromised.

External list permissions

Each external list is associated with an external content type. The permissions on the external content type specify who can perform specific actions on the external content type. The Execute permission is required to execute operations (such as read or update) on an external content type and also to generate a deployment package for the external list. However, after a deployment package is created for an external list, any user who can access that external list can download and install the deployment package. In other words, a user who does not have Execute permission on the external content type, but has Read permission level on the external list, cannot see the items in the external list, but may still be able to take the external list offline. To help ensure that sensitive data is not disclosed, we recommend that you ensure that the permissions on an external list are equal to the permissions of the associated external content type.

Outlook Web Access Web Parts

Outlook Web Access Web Parts enable users to display selected content from folders in their Office Outlook e-mail account in a SharePoint site. If users have taken external data offline to Outlook, using the Outlook Web Access Web Parts can result in sharing of sensitive data. We recommend that administrators educate users to share their Outlook folders only with people whom they can trust.

Client throttle limits

Setting throttle limits on the client computer can help limit denial of service threats that are caused by a user who submits queries that return a large amount of data or take lots of processing time. You can use registry-based policy keys to set throttle limits on client computers. The supported way to manage registry-based policy keys is to use Group Policy to apply the registry policy settings.

The Business Connectivity Services policy settings are included in the Office14.adm file, which can be downloaded from [Office 2010 Administrative Template files \(ADM, ADMX, ADML\) and Office Customization Tool](http://go.microsoft.com/fwlink/?LinkId=189316) (<http://go.microsoft.com/fwlink/?LinkId=189316>).

The following table describes the Business Connectivity Services registry-based policy keys that can be used to set throttle limits. The keys are located under

HKEY_CURRENT_USER\Software\Policies\Microsoft\office\14.0\Common\Business Data.



Note:

The following table lists only the main policy settings that can be used to set client throttle limits. To see the complete list of available Business Connectivity Services policy settings, refer to the Office2010GroupPolicyAndOCTSettings_Reference.xls file that is included on the following download page: [Office 2010 Administrative Template files \(ADM, ADMX, ADML\) and Office Customization Tool](http://go.microsoft.com/fwlink/?LinkId=189316) (<http://go.microsoft.com/fwlink/?LinkId=189316>).

Key	Type	Value	Description
Synchronization\Query	REG_ DWORD	1-32,767	Specifies the

Key	Type	Value	Description
Instances Limit			<p>maximum number of items that can be added to the cache by Business Connectivity Services as the result of executing a query.</p> <p>Some queries can return many items to be added to the cache. This increases the size of the client cache (potentially exceeding the 4-GB limit that is imposed by the Microsoft SQL Server Compact Edition database), increases the work that is required to keep the client cache synchronized, and increases the load on the external system. When the limit is reached, the processing stops. The query is marked as failed and will be retried later.</p> <p>The default is 2,000 items.</p>
Synchronization\Query Timeout	REG_ DWORD	1-360 (minutes)	<p>Specifies the number of minutes Business Connectivity Services will spend processing a single query.</p> <p>Some queries can take significant time</p>

Key	Type	Value	Description
			<p>before all results are retrieved and processed. During this time, no other operation can be processed. When the time-out is exceeded, the processing stops. The query is marked as failed and will be retried later.</p> <p>Typical values range from 3 to 10 minutes. The default is 5 minutes.</p>
Limits\Database\Items\Max	REG_DWORD	1-2,000,000	<p>Specifies the maximum number of items the database connector can return per request.</p> <p>Typical values range from 1,000 to 3,000 items. The default is no data limit.</p>
Limits\Database\Timeout\Max	REG_DWORD	1-75,000,000 (milliseconds)	<p>Specifies the number of milliseconds to wait until an open database connection is terminated.</p> <p>Typical values range from 5,000 to 180,000 milliseconds (5 seconds to 3 minutes). The default is no time-out.</p>
Limits\Wcf\Size\Max	REG_DWORD	1-1,000,000,000 (KB)	<p>Specifies the maximum amount of data a Web service</p>

Key	Type	Value	Description
			connector can return per request. Typical values range from 512 KB to 524,288 KB (512 MB). The default is no data limit.
Limits\Wcf\Timeout\Max	REG_DWORD	1-75,000,000 (milliseconds)	Specifies the number of milliseconds to wait until an open Web service connection is terminated. Typical values range from 5,000 to 180,000 milliseconds (5 seconds to 3 minutes). The default is no time-out.

Diagnostic logging in Business Connectivity Services overview (SharePoint Server 2010)

You can troubleshoot issues related to Microsoft Business Connectivity Services on servers that are running Microsoft SharePoint Server 2010 and on Microsoft Office 2010 client applications by using event logs and trace logs on either client or server. Also, each entry to the event log or trace log has an associated Activity ID that can be used to track a problem from client or server to the external data source.

**Note:**

In addition to the logging methods discussed in this topic, you can use Microsoft System Center Operations Manager Management Pack to monitor a solution that is based on Microsoft Business Connectivity Services. For more information about how to configure System Center Operations Manager Management Pack, see the guide including in the management pack download at [Microsoft SharePoint 2010 Products Management Pack](http://go.microsoft.com/fwlink/?LinkId=184971) (<http://go.microsoft.com/fwlink/?LinkId=184971>).

In this article:

- [Diagnostic logging in Business Connectivity Services](#)
- [About Activity IDs](#)
- [Diagnostic logging on servers](#)
- [Diagnostic logging on Office 2010 clients](#)
- [Example: using diagnostic logging](#)

Diagnostic logging in Business Connectivity Services

For solutions that are based on Microsoft Business Connectivity Services, diagnostic logging occurs both on servers that are running SharePoint Server 2010 and on Office 2010 clients. There are two logs: the event log and the trace log. They both record diagnostic information that Microsoft Business Connectivity Services generate. Event logs record error messages. Trace logs contain more in-depth information, such as stack traces and informational messages. In general, trace logs provide more details than event logs.

Each logged item of information includes an Activity ID, which is a unique GUID value. Activity ID values can also be sent to external systems when a Create, Update, or Delete operation occurs on an item. By using Activity IDs, an action can be traced from the server or client to the external data source. For more information about Activity IDs, see [About Activity IDs](#).

You can set the level of diagnostic logging for the event log and for the trace log. This will limit the types and amount of information that will be written to each log. The following tables define the levels of logging available for the event log and trace log:

Event log levels

Level	Definition
None	No logging occurs.
Critical	This message type indicates a serious error that has caused a major failure in the solution.
Error	This message type indicates an urgent condition. All error events should be investigated.
Warning	This message type indicates a potential problem or issue that might require attention. Warning messages should be reviewed and tracked for patterns over time.
Information	Information messages do not require any action, but they can provide valuable data for monitoring the state of your solution.
Verbose	This event log level corresponds to lengthy events or messages.

Trace log levels

Level	Definition
None	No trace logs are written.
Unexpected	This level is used to log messages about events that cause solutions to stop processing. When set to log at this level, the log will only include events at this level.
Monitorable	This level is used to log messages about any unrecoverable events that limit the solution's functionality but do not stop the application. When set to log at this level, the log will also include critical errors (Unexpected level).
High	This level is used to log any events that are unexpected but which do not stall the processing

Level	Definition
	of a solution. When set to log at this level, the log will include warnings, errors (Monitorable level) and critical errors (Unexpected level).
Medium	When set to this level, the trace log includes everything except Verbose messages. This level is used to log all high-level information about operations that were performed. At this level, there is enough detail logged to construct the data flow and sequence of operations. This level of logging could be used by administrators or support professionals to troubleshoot issues.
Verbose	When set to log at this level, the log includes messages at all other levels. Almost all actions that are performed are logged when you use this level. Verbose tracing produces many log messages. This level is typically used only for debugging in a development environment.

Diagnostic logs are useful both in development and production environments, but requirements for the level of logging will probably differ depending on the kind of environment. When planning for diagnostic logging in Microsoft Business Connectivity Services, consider the business needs and the lifecycle stage of the environment before you set the logging level.

For example, during solution design, you might, for debugging purposes, set both logging levels to Verbose to capture all the messages that are generated about the state of the system. Conversely, in a production environment, you might want to capture only messages in the categories High, Monitorable, and Unexpected for trace logs and the categories Critical and Error for event logs. Doing this will save logging disk space and limit any negative performance effects of logging.

About Activity IDs

A unique GUID value called an Activity ID is generated on the server and Office client for each Create, Update, or Delete operation on external data in a solution based on Microsoft Business Connectivity Services. Anything related to the operation that is logged in the trace log or event log includes its Activity ID value.



Important:

In the event logs and trace log files on the server, Activity ID values are labeled as “CorrelationId” values.

The Activity ID value generated for a Create, Update, or Delete operation is sent to the external system along with other information related to that operation. If the external system has a logging mechanism, this value can be captured and logged on that system. Therefore, if an operation causes entries to the SharePoint server or Office client logs, the same operation can be traced to the external system by using its Activity ID value. This facilitates end-to-end troubleshooting of issues.

Often, an operation such as Create will cause multiple events to be written to the logs. When this happens, the same Activity ID value is used for all events that are logged for the operation. This is useful in troubleshooting issues because the recurring value of the Activity ID facilitates finding all events for a particular operation. Conversely, when the same type of operation occurs repeatedly, a unique Activity ID value is generated for each operation instance. For example, if an item of an external content type is updated twice, each update operation will be associated with a unique Activity ID value.



Tip:

In some circumstances, the Business Data Connectivity service will retry an operation if it failed to go through to the external system. In those cases, the same Activity ID will be used for the retried operation.

Diagnostic logging on servers

By default, Microsoft Business Connectivity Services logging is enabled on SharePoint Server 2010 servers. The default logging levels are:

- For the event log: Critical and Error
- For the trace log: Medium

Should diagnostic logging of Microsoft Business Connectivity Services become disabled, enable it by selecting **Business Connectivity Services** on the Diagnostic Logging page in SharePoint Server 2010 Central Administration. You can also use Windows PowerShell to configure event logs and trace logs on the server. For example, you can change the drive that logging writes to, and you can set the level of verbosity of logging.

For more information about logging in SharePoint Server 2010, such as how to set the location of the log files, see [Configure diagnostic logging \(SharePoint Server 2010\)](#).

You can use Windows PowerShell to view the event logs on the server and you can export the logs, for example to a spreadsheet program. For more information, see [View diagnostic logs \(SharePoint Server 2010\)](#).

Microsoft Business Connectivity Services output two categories to the trace log on SharePoint Server 2010 front end Web servers: **BDC_Shared_Services** and **SS_Shared_Service**. You can use the Event Viewer to open the trace log, and you can filter on the relevant log entries by searching on “SPS_BusinessData” (for Microsoft Business Connectivity Services outputs) and “SPS_SecureStoreService”.

Diagnostic logging on Office 2010 clients

Event logs and trace logs for Microsoft Business Connectivity Services solutions are available on Microsoft Office 2010 suites clients that use the Microsoft Business Connectivity Services infrastructure. By default Event logging for Microsoft Business Connectivity Services is enabled on clients. However, to protect performance, only errors and critical errors are logged and this setting cannot be changed. Windows client computers include an Event Viewer that you can use to view event logs. For information about how to view event logs for a specific version of Windows, consult the product documentation.

Trace logging is disabled by default on client computers to help enhance performance. You should only enable trace logging on client computers if you are encountering problems that you want to diagnose. For example, if an event log entry indicates an error might be caused by a an activity that is related to Microsoft Business Connectivity Services, then enable trace logging to gather additional data the next time that the event occurs.

The method for enabling trace logging and reading the logs varies depending on the version of Windows on the computer. On client computers that are running Windows 7 or Windows Vista, you can use the **Performance Monitor** utility to enable tracing to capture Microsoft Business Connectivity Services events. For the steps for enabling Microsoft Business Connectivity Services tracing on computers that are running Windows 7 or Windows Vista, see [Use tracing on the client \(SharePoint Server 2010\)](#). On computers running Windows XP, you enable tracing by running a script that uses the `logman` command.

The following sample script uses the `logman` command to enable trace logging:

```
rem This script will enable logging, directing log messages to a file specified by the
"%FILE_NAME%" given by the user.

@setlocal

@echo off

pushd %~dp0

set PATH_NAME=%TEMP%\BCS

set FILE_NAME=%PATH_NAME%\ETWTraceLog

set TRACE_COLLECTION=BCS

::tracelog -start BCS -guid #b8622a02-c377-46b1-b861-38a787a8e44a -b 128 -flags 0xFFFF -level 5
-f "%FILE_NAME%.etl"

md "%PATH_NAME%" 1>nul 2>nul

logman create trace %TRACE_COLLECTION% -p "{b8622a02-c377-46b1-b861-38a787a8e44a}" 0xFFFF 5 -o
"%FILE_NAME%.etl" -ets

echo.

echo Business Connectivity Services tracing has been started. To end press any key.

echo.
```

pause

As on the server, a unique Activity ID value is generated for each Create, Update, or Delete operation on an item in the client. These values are recorded in the logs and sent to external systems along with other information about operations. Also, a solution can be configured so that Activity Id values are displayed in error messages. This facilitates troubleshooting problems encountered by solution users.



Important:

Because the required version of the Event Tracing for Windows programming interface on which Activity ID generation is dependent is not available on the Windows XP operating system, Activity ID generation is not supported on clients running Windows XP.

Example: using diagnostic logging

This short, simplified scenario illustrates the use of diagnostic logging in a production environment. An enterprise has deployed a new time card submission solution based on Microsoft Business Connectivity Services. This solution uses an external system to store timecard information for employees, such as vacation time and sick leave, and to interact with employees and the payroll system when employees report absence from work. Employees use a Web Part to interact with the system.

On the server farm, the logging levels are set to the default values for Microsoft Business Connectivity Services:

- For the event log: Critical and Error
- For the trace log: Medium

In this scenario, an employee submits a value for the number of sick leave hours but neither the employee nor his manager receives a confirming email message reporting that the sick leave time was successfully submitted. The employee calls the internal technical support service and reports the issue.

The support technician recognizes that the time card application is based on Microsoft Business Connectivity Services. She checks the event log but finds no error associated with the identity of the user at the time the user submitted the time card request. She then checks the trace log, where she finds the evidence of the activity: an Update operation associated with the user at the appropriate time. The Update operation in the trace log includes an Activity ID value which the support technician notes.

The support technician knows that logging is also supported on the external system. Using the Activity ID, she locates the item logged on the external system and finds evidence of an error written to the log at the end of the Update operation: the update failed because the employee had used up all of his allotted sick leave time. She also notes that there is no log entry confirming that an email message was generated on the external system immediately at the end of the Update operation. The support technician concludes that there is an error in the logic of the time card application. Although the application properly did not allocate sick time pay when the employee exceeded his allotted amount of hours, it failed to generate an email message informing the employee of the issue. She reports the problem to the development team that created the application and the development team updates the application.

See Also

[Use tracing on the client \(SharePoint Server 2010\)](#)

[Monitoring overview \(SharePoint Server 2010\)](#)

[Configure diagnostic logging \(SharePoint Server 2010\)](#)

[Business Connectivity Services overview \(SharePoint Server 2010\)](#)

Plan to upgrade to Business Connectivity Services (SharePoint Server 2010)

This article describes how upgrading from the Microsoft Office SharePoint Server 2007 Business Data Catalog to the Microsoft Business Connectivity Services in Microsoft SharePoint Server 2010 works and provides guidance about how to upgrade various types of solutions that use the Business Data Catalog.

- [The Business Data Catalog, Application Registry, and Business Data Connectivity service](#)
- [How Business Connectivity Services upgrade works](#)
- [Upgrading by using database attach](#)
- [Solution-specific upgrade considerations](#)

The Business Data Catalog, Application Registry, and Business Data Connectivity service

This section introduces the three services involved in an upgrade from the Business Data Catalog to SharePoint Server 2010.

In Office SharePoint Server 2007, the Business Data Catalog integrates external data into solutions. The Business Data Catalog was implemented as a shared service in the context of the Shared Services Provider framework. The Business Data Catalog stored application definitions, which are similar, but not identical, to BDC models in SharePoint Server 2010.

In SharePoint Server 2010, a new service, the Business Data Connectivity service, integrates external data into solutions. This service is implemented in the new shared services framework of SharePoint Server 2010. For an overview of the new shared services framework, see [Services architecture planning \(SharePoint Server 2010\)](#). The new Business Data Connectivity service is part of the Microsoft Business Connectivity Services, which is a set of client and server capabilities that includes a full set of read-write capabilities with external systems, integration with Microsoft Office 2010 clients, and a rich set of development tools. The object model of the Business Data Connectivity service is not backward compatible with the object model that the Business Data Catalog uses, although the object models are similar.

Another service in SharePoint Server 2010, the Application Registry Service, directly replaces the Business Data Catalog from Office SharePoint Server 2007 and supports the backward compatibility of solutions that are based on the Business Data Catalog. Because the Application Registry Service supports the object model of the Business Data Catalog, the Application Registry Service can support solutions that depend on the Business Data Catalog object model. The Application Registry service can support applications originally created in Office Microsoft Office SharePoint Server 2007 that include one or more of the following elements:

- Custom Web Parts
- Custom code
- Searching

All of these depend on the object model or metadata schema of the Business Data Catalog and will continue to work in SharePoint Server 2010 by using the Application Registry Service.

The Web Parts that were provided by Office SharePoint Server 2007, such as the Business Data List Web Part and the Business Data Association Web Part, are upgraded in SharePoint Server 2010 to run against the Business Data Connectivity service and do not run against the Application Registry Service. For more information, see [Web Parts](#).



Important:

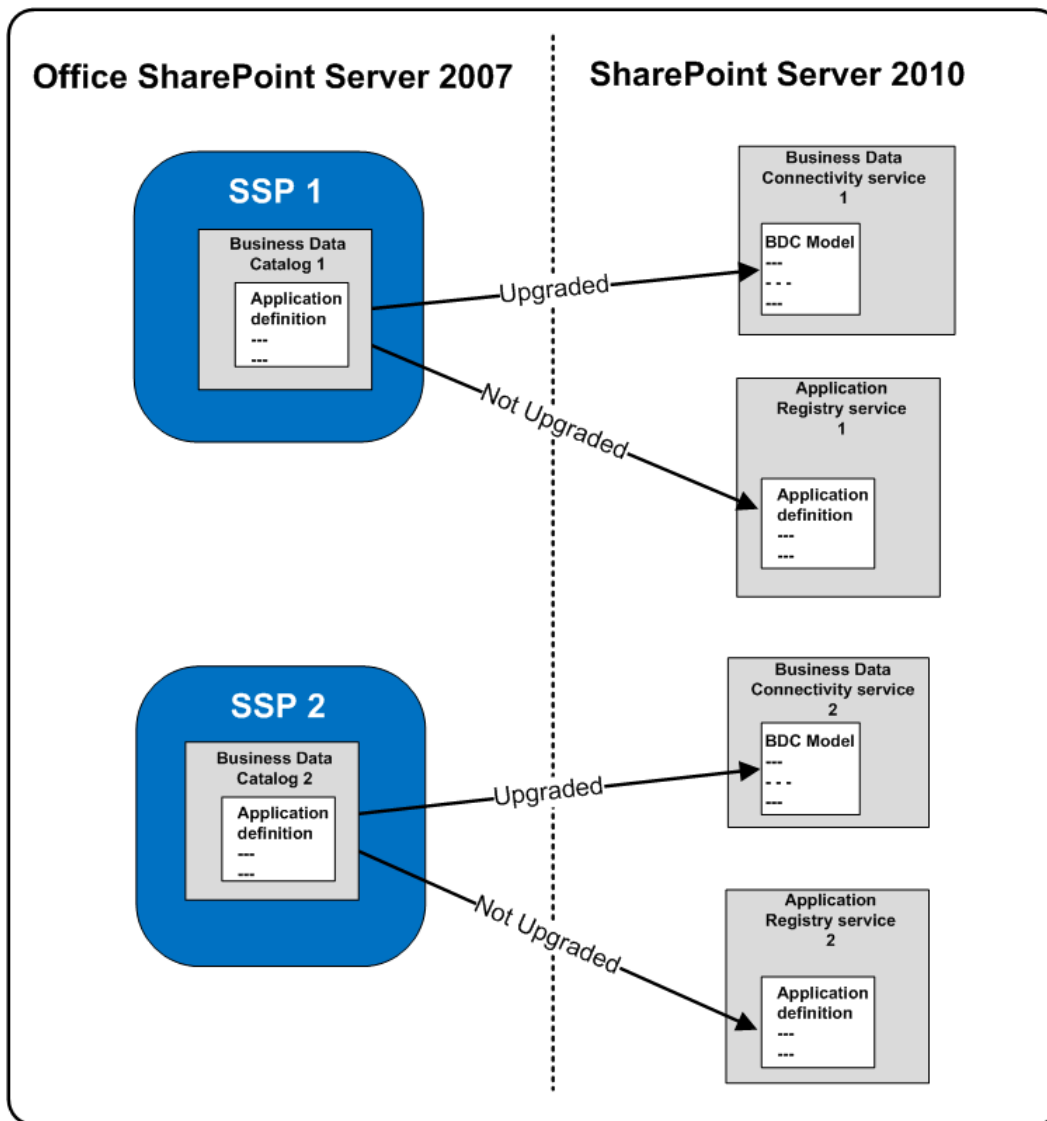
Note that the Application Registry Service is deprecated and exists only to support the backward compatibility of solutions that are based on the Office SharePoint Server 2007 Business Data Catalog. Do not build new solutions that depend on the deprecated Application Registry Service. Build new solutions on the Business Data Connectivity service.

How Business Connectivity Services upgrade works

During upgrade from Office SharePoint Server 2007 to SharePoint Server 2010 , the following occurs:

1. For each Business Data Catalog that is running in the context of a Shared Service Provider (SSP) in Office SharePoint Server 2007, two services are created in SharePoint Server 2010:
 - An Application Registry Service to store application definitions that are compatible with the Office SharePoint Server 2007 Business Data Catalog object model.
 - A Business Data Connectivity service to store BDC models, which are compatible with the SharePoint Server 2010 Business Data Connectivity Services object model. The new object model supports the new features in the Business Data Connectivity service.
2. A copy of each application definition in the Business Data Catalog is placed in the Application Registry service and, an upgraded version that is based on the new object model is placed in the Business Data Connectivity service.

The following illustration shows how this upgrade works. In the graphic, two shared service providers are shown in Office SharePoint Server 2007, and the resulting four services (two instances each of the Application Registry Service and the Business Data Connectivity service) are shown in SharePoint Server 2010 after upgrading:



Warning:

Only in-place upgrading is supported for upgrading solutions that are based on the Office SharePoint Server 2007 Business Data Catalog. For general information about in-place upgrading, see [Perform an in-place upgrade \(SharePoint Server 2010\)](#).

Upgrading by using database attach

There are two approaches to upgrading from Office SharePoint Server 2007 to SharePoint Server 2010: in-place upgrade and database attach upgrade. For more information, see [Upgrade process overview \(SharePoint Server 2010\)](#). If you use the database attach method to upgrade and you want to

continue to use the external data that is accessed through one or more application definitions, you must complete the following tasks:

1. Export application definitions that the solution requires from the Office SharePoint Server 2007 Business Data Catalog. For more information, see [Export application definition](http://go.microsoft.com/fwlink/?LinkId=189010) (http://go.microsoft.com/fwlink/?LinkId=189010).
2. After upgrading, update the solution to use the object model and features of the Microsoft Business Connectivity Services. This includes updating the application definitions to become BDC models, which are compatible with Microsoft Business Connectivity Services. For information about the Business Data Connectivity service object model, see [Microsoft SharePoint 2010 Software Development Kit](http://go.microsoft.com/fwlink/?LinkId=166117) (http://go.microsoft.com/fwlink/?LinkId=166117).
3. Import the updated BDC models into the Business Data Connectivity service. For more information, see [Manage BDC models \(SharePoint Server 2010\)](#).

For more information about database attach upgrading, see [Prepare the new SharePoint Server 2010 environment for a database attach upgrade](#) and [Perform post-upgrade steps for a database attach upgrade \(SharePoint Server 2010\)](#).

Solution-specific upgrade considerations

This section contains recommendations about how to manage specific aspects of solutions that are upgraded to run in SharePoint Server 2010.

- [Models](#)
- [Web Parts](#)
- [Search](#)
- [Single sign-on](#)
- [Single sign-on](#)
- [Maintaining parent and child farm relationships](#)

Models

As described in a previous section, when you upgrade from the Office SharePoint Server 2007 Business Data Catalog to Microsoft Business Connectivity Services, two versions of each application definition in the Business Data Catalog are created in SharePoint Server 2010. One identical copy of the application definition is stored in the Application Registry service; another version of the application definition file is upgraded to the Microsoft Business Connectivity Services metadata schema and, now called a “BDC model,” is stored in the new Business Data Connectivity service.



Important:

When application definitions are upgraded to BDC models and stored in the Business Data Connectivity service, all errors or warnings that are related to the conversion process are written to the upgrade log. For information about the upgrade log, see [Verify upgrade and review upgraded sites \(SharePoint Server 2010\)](#).

We recommend that you use the upgraded BDC model in the Business Data Connectivity service for future development of the solution. However, if you do additionally develop the original application definition in the Application Registry service, you should make similar changes on the updated BDC model that is stored in the Business Data Connectivity service. This is so that features that rely on the new object model, such as the Web Parts that are included in SharePoint Server 2010, will provide a consistent user experience. Also, because the Application Registry Service is deprecated, you ensure that a solution will be available for future upgrades to subsequent product versions if you develop on the BDC model.

Web Parts

Office SharePoint Server 2007 included the following Web Parts for building solutions that use the Business Data Catalog:

- Business Data List Web Part
- Business Data Association Web Part
- Business Data Actions Web Part
- Business Data Details Web Part
- Business Data Catalog Filter Web Part

The Web Parts that were provided by Office SharePoint Server 2007 are upgraded in SharePoint Server 2010 to run against the Business Data Connectivity service. Such Web Parts that you used in a Office SharePoint Server 2007 solution will continue to work after upgrade by using the upgraded Web Parts and the Business Data Connectivity service. Custom Web Parts in a Office SharePoint Server 2007 solution, however, will run against the Application Registry Service after upgrade. For data consistency, we recommend that you obtain new versions of custom Web Parts that can run against the Business Data Connectivity service.

These Web Parts have been upgraded in SharePoint Server 2010 to work on BDC models in the new Business Data Connectivity service. If a solution in Office SharePoint Server 2007 depends on one or more of these Web Parts, you should upgrade the solution to the Microsoft Business Connectivity Services object model and use the new versions of the Web Parts to interact with it. They are not available to run directly on application definitions in the Application Registry service.

Search

To search external data in Office SharePoint Server 2007, you create a content source that points to an application definition in the Business Data Catalog. If you search content in this manner in Office SharePoint Server 2007, you can continue to search the same external data after upgrading to SharePoint Server 2010 BY using the Application Registry Service. No modification to the content source is required on upgrade for search to work. It is recommended that you also create a new content source definition that points to the Business Data Connectivity service.

Single sign-on

In Office SharePoint Server 2007, some authentication modes for solutions that use the Business Data Catalog also use the Single Sign-On service, which provides storage and mapping of credentials such as account names and passwords. Application definitions in the Business Data Catalog that depended on the Single Sign-On service included information that references the service.

The Secure Store Service replaces the Single Sign-On service in SharePoint Server 2010. For more information, see [Plan the Secure Store Service \(SharePoint Server 2010\)](#). After you upgrade to SharePoint Server 2010, application definitions that were upgraded to the Application Registry service and to the Business Data Connectivity service will be updated to refer to the Secure Store Service. For information about how to migrate data from the Single-Sign-on service to the Secure Store Service, see [Perform post-upgrade steps for an in-place upgrade \(SharePoint Server 2010\)](#)



Important:

If you used a custom Single Sign-On provider in Office SharePoint Server 2007, then you will need to upgrade the provider in SharePoint Server 2010 and update the BDC models and application definitions manually in the Application Registry service and the Business Data Connectivity service.

Maintaining service databases on separate servers

In Office SharePoint Server 2007, the database that the Shared Services Provider uses can optionally be on a separate server from the SharePoint content database. After you upgrade to SharePoint Server 2010, the databases for both the Business Data Connectivity service and the Application Registry service will always be on the same server as the content databases. If you want to maintain separate databases for these services, you will have to do this as a post upgrade step by backing up each service's database and then restoring the databases to a separate server.

Maintaining parent and child farm relationships

In Office SharePoint Server 2007, the Shared Services Provider that includes the Business Data Catalog could be in a farm that services one or more child farms that use the shared services. If you want this service to be available to child farms after you upgrade to the Application Registry service, you will have to use the `CreateDelegatedApplicationProxy` Windows PowerShell command to create a new proxy that connects to the Application Registry Service's database on the parent farm. The following sample script illustrates the use of this command for this purpose:

```
$proxy=[Microsoft.Office.Server.ApplicationRegistry.SharedService.ApplicationRegistryServiceApplicationProxy]::CreateDelegatedApplicationProxy("<NAME>", "Data Source=<SQLserverName>;Initial Catalog=<ApplicationRegistry DB NAME>;Integrated Security=true")
```

For the Business Data Connectivity service, this operation can be done using SharePoint Central Administration.

See Also

[Plan for Business Connectivity Services \(SharePoint Server 2010\)](#)

Plan InfoPath Forms Services (SharePoint Server 2010)

InfoPath Forms Services is the service that supports the deployment and integration of InfoPath browser forms in Microsoft SharePoint Server 2010. Employees, customers, and business partners of an organization use forms to standardize, customize, and validate data collection. Forms are often deployed as one element in a business solution that uses a broad functionality of SharePoint Server 2010 features.

The articles in this chapter can help you determine the aspects of your SharePoint Server 2010 deployment to govern and include discussions of governance techniques. The articles include:

- [About forms in SharePoint Server 2010](#)
Defines InfoPath Forms Services, explains the types of forms used in SharePoint sites and solutions, and describes the common administrative tasks for server farm administrators and other IT pros.
- [Plan a forms-driven application](#)
Describes the considerations to use when planning for InfoPath forms in a SharePoint application.
- [Plan for user form templates \(SharePoint Server 2010\)](#)
Describes the considerations to use when planning for user form templates.
- [Plan to upgrade form templates during an upgrade to SharePoint Server 2010](#)
Describes the process that is used to successfully upgrade form templates during an upgrade from Office SharePoint Server 2007 to SharePoint Server 2010 and the additional steps that are required.
- [InfoPath 2010 Enhanced Integration with SharePoint Server 2010 and Its Implications When Designing Forms for Applications \(white paper\)](#)
Describes how to use InfoPath List forms, InfoPath document-based forms, and InfoPath Web Parts to create enterprise-level applications in SharePoint Server 2010.

See Also

[InfoPath forms administration \(SharePoint Server 2010\)](#)

About forms in SharePoint Server 2010

In this article, you will learn about the benefits of using InfoPath forms to gather data in Microsoft SharePoint Server 2010 solutions. It describes the role of forms in broader business solutions, and the types of forms you can create by using Microsoft InfoPath 2010.

In this article:

- [InfoPath forms overview](#)
- [Role of forms in SharePoint solutions](#)
- [Types of InfoPath forms](#)
- [Deploying forms](#)
- [Filling out forms](#)

InfoPath forms overview

All business processes involve gathering data of some kind, and the success of any business process relies on the quality and integrity of that data. By using InfoPath forms in your SharePoint solutions, you can ensure that only valid, good quality data is gathered. InfoPath 2010 achieves this by letting you standardize, customize, and optimize the process of gathering data without writing any code.

The following are some key features that InfoPath provides to ensure data integrity:

- **Custom data validation** ensures that users cannot submit forms that contain invalid data.
- **Customized layout.** By using multiple views, you can simplify the form-filling experience by splitting forms into multiple pages or creating separate views that are optimized for specific users or tasks. Conditional formatting lets you optimize the form-filling experience. For example, you can show or hide fields in the form based on values entered by users.
- **Data connections** to other sources such as SharePoint lists or Web services let you pull supporting, contextual data into your forms.



Note:

InfoPath Forms Services accesses external data sources by using a delegated Windows identity. Consequently, external data sources must reside within the same domain as the SharePoint Server 2010 farm, or InfoPath form templates must be configured to use the Secure Store Service. If the Secure Store Service is not used and external data sources do not reside within the same domain, authentication to the external data sources will fail. For more information, see [Planning considerations for service applications that access external data sources](#) in “Services Architecture Planning.”

Code is not required to implement the above features. However, you can write managed code by using Visual Studio Tools for Applications (VSTA) to add more advanced functionality to your forms.

Role of forms in SharePoint solutions

Forms play a role in all business process management solutions. The following are just a few examples of the types of business solutions and applications that use InfoPath forms:

- **Form-driven applications** start with a form that is used to gather data and add in other SharePoint functionality to process and report on the data including workflows, reporting, custom Web Pages, and external lists. Examples include a help desk Web site, a procurement center, or an employee hiring center for a human resources department.
- **Document workflow applications** automate document-based business processes. Adding InfoPath forms to workflows lets you gather information from the users at predefined times in the life of the workflow and let users interact with the tasks for that workflow. Examples include contract approval, technical specification review, or processing for legal documents.
- **Business data applications** start with data in an external system that is made available in SharePoint Server 2010 through Business Connectivity Services. Data in the external systems is accessible in SharePoint Server 2010 through external lists and can be added, viewed, updated, and deleted by using InfoPath forms. Examples include customer relationship management and enterprise resource planning.
- **Ad hoc business processes** can be managed by using SharePoint lists. The forms used for creating, reading, and updating items in these lists can be customized by using InfoPath 2010. Examples include simple issue tracking or tasks lists.
- **Document information panels** use InfoPath forms to display document metadata within the Microsoft Office system client application.

Types of InfoPath forms

Depending upon an organization and its business resources and needs, form designers can be professionals dedicated to forms-based design and development, or other users such as information workers, overall solution designers, site designers, site collection administrators, Web developers, or farm administrators who have other primary responsibilities.

The type of form they design will depend on the requirements of the business solution.

InfoPath components

InfoPath consists of the following two main components:

- **InfoPath 2010** is the Office system client application that has the following two modes:
 - **Microsoft InfoPath Designer 2010** is used to create form templates. Form templates define the data structure, appearance, and behavior of a form. After the design process has been completed, form designers publish the form template to SharePoint Server 2010 so that users can start filling out forms based on that template.
 - **Microsoft InfoPath Filler 2010** is used to fill out forms that are based on form templates created in InfoPath Designer.

- **InfoPath Forms Services in Microsoft SharePoint Server 2010** enables forms to be rendered in the Web browser. It is available as an Enterprise feature of SharePoint Server 2010. Farm administrators configure settings for InfoPath Forms Services on the SharePoint Central Administration Web site.

Web browser vs. Filler-only forms

The first design decision that form designers have to make is whether to create a Web browser form or a Filler-only form. Browser forms can be filled out in a Web browser, whereas Filler-only forms can only be filled out in InfoPath Filler 2010.

The major benefit of creating browser forms is that users who fill out the forms do not have to have InfoPath 2010 installed on their computers. These forms can be filled out in the browser. InfoPath Forms Services is required to publish browser forms to SharePoint Server 2010 and to render them in the browser. However, it is important to note that certain functionality and controls are not supported in Web browser forms. This is the most common reason for creating a Filler-only form.

Web browser forms

Browser forms include the following types:

- List forms
- External list forms
- Form library forms
- Workflow forms

SharePoint list forms

New in SharePoint Server 2010, you can now extend and enhance the forms used for creating, editing and viewing items in SharePoint lists by using InfoPath 2010. InfoPath form templates that are published to SharePoint lists replace the default SharePoint list form.

Each content type for the list has its own SharePoint list form that can be customized in InfoPath Designer.

By customizing list forms in InfoPath Designer, you can enhance your SharePoint lists by adding the following functionality that is not supported in default SharePoint list forms:

- **Add rules to validate data that users add to forms.** For example, if you are using forms to track orders for an organization, you can enforce a spending limit for individual orders.
- **Add rules to format data based on particular conditions.** Form designers can show or hide fields, enable or disable controls, switch views, or set values for data in a field. For example, in an inventory-tracking form template, you can show different information to a user who requests new inventory than to the organizational administrator who purchases new inventory.
- **Query data from other sources.** For example, you can check the availability of inventory in an external system and let users know immediately if a particular item is unavailable.

- **Change the layout and appearance of forms.** For example, you could move the required fields to a prominent position in the form to make them easier for users to find and fill out, or add views that are optimized for different tasks or user permissions.

These customization options and the built-in views and filtering options available for SharePoint lists enable list forms to be used in many kinds of business solutions, from ad hoc processes to more complex, structured business processes.

SharePoint list form templates cannot contain managed code or repeating or nested data. If the overall solution requires managed code or complex data structures in forms, you must use a form library.

External list forms

Business Connectivity Services lets you work with data from external systems such as databases or Web services by using SharePoint external lists. External lists are created from external content types. As for regular SharePoint lists, InfoPath 2010 can be used to customize the forms used to create, read, update and delete items in external lists.

Form library forms

A form library is a SharePoint document library that uses an InfoPath form as its default content type. New forms are created by clicking the **New** button in the form library and individual forms that are filled out by users are stored as XML files in the library.

Form library forms should be used if your solution requires any of the following features:

- Repeating or nested data
- Digital signatures
- Managed code
- Form data stored as XML

Workflow forms

Form designers can use InfoPath Designer to customize the forms that users fill out to interact with tasks in SharePoint workflows.

For more information about how to design form templates, see [InfoPath Help and How-to](http://go.microsoft.com/fwlink/?LinkId=185225) (http://go.microsoft.com/fwlink/?LinkId=185225).

For more information about how to design form templates that have custom code, see [InfoPath Forms Services](http://go.microsoft.com/fwlink/?LinkId=185226) (http://go.microsoft.com/fwlink/?LinkId=185226) in the SharePoint Developer Center.

Deploying forms

Publishing options for form library forms depend upon whether the form template contains managed code, the permissions of the form designer, and InfoPath Forms Services settings.

Publishing browser forms without code

Web browser forms that do not contain managed code can be published directly to SharePoint Server 2010 by any form designer with the Design permission level. Form designers publish form templates by using InfoPath Designer.

Farm administrators can restrict the ability to publish browser forms to SharePoint Server 2010 by disabling the publishing of browser-enabled form templates on the InfoPath Forms Services configuration options page in Central Administration. This option applies to form library forms only, and enables organizations that want more centralized control of form templates to require administrator approval before publishing browser forms to SharePoint Server 2010. This option does not prevent users from publishing Filler-only forms to SharePoint Server 2010. These forms can only be filled out by using InfoPath Filler.

Publishing browser forms with code

Form designers can add managed code to their forms by using Visual Studio Tools for Applications (VSTA). VSTA is an optional installation component available in Setup for InfoPath 2010. Form templates that have managed code can be published in the following two ways:

- **Sandboxed solutions** Site collection administrators can publish form templates that have code to form libraries as sandboxed solutions.
- **Administrator-approved forms** Farm administrators can upload form templates to a library in Central Administration, and then activate the form templates on site collections as site content types that users with the Design permission level can select.

Form templates as sandboxed solutions

Site collection administrators can publish form templates that have code to form libraries as sandboxed solutions, if the Microsoft SharePoint Foundation Sandboxed Code Service is running on the farm. This lets a form designer who is a site collection administrator publish forms without requiring farm administrator approval. The sandboxed solutions will run in an environment that has access to a subset of the server object model. The farm administrator can set usage limits to protect the server from malicious code.

Form templates that contain code that requires full trust to execute cannot be published as sandboxed solutions. They must be approved by a farm administrator. For more information about sandboxed solutions, see [Sandboxed solutions overview \(SharePoint Server 2010\)](#).

Administrator-approved form templates

Farm administrators can upload form templates that have code, including code that requires full trust to execute, to a form template library in Central Administration. This can be a preferred option in organizations that are willing to sacrifice ease of deployment for more centralized control of forms and managed code. Form templates that are uploaded to the form template library in Central Administration are called administrator-approved form templates.

After saving the form template, the form designer contacts the farm administrator. The farm administrator reviews the form template to ensure that it meets organizational performance and security standards. If the form template meets those standards, the farm administrator uploads it to the form template library in Central Administration for the farm.

A farm administrator has to activate a form template for a specific site collection before it can be used on the site collection. Activating a form template puts the form template in a library of form templates for the site. Users can then add the form template as a content type to a form library. Anyone with the Design permission level can add a form template content type.

For more information, see [Manage administrator-approved form templates \(SharePoint Server 2010\)](#).

Filling out forms

After a form template has been published, users can start filling out forms that are based on that form template.

Browser vs. Filler forms

There are two main form-filling environments, the Web browser and InfoPath Filler. Any form can be filled out in InfoPath Filler. Only browser-enabled forms can be filled out in a Web browser.

Farm administrators can disable rendering of forms in the Web browser by configuring settings on the InfoPath Forms Services configuration page in Central Administration. This applies only to user form templates that have been published to form libraries, and not to administrator-approved form templates or form templates published to lists. If rendering of Web browser forms is disabled, forms will open in InfoPath Filler if InfoPath is installed on the user's computer.

Offline form-filling

SharePoint list forms can be filled out offline by using SharePoint Workspace. Form library forms can be filled out offline by using InfoPath Filler.

InfoPath Form Web Part

InfoPath browser forms can also be hosted on Web pages by using the InfoPath Form Web Part. For more information about the InfoPath Form Web Part, see [About the InfoPath Form Web Part](#) (<http://go.microsoft.com/fwlink/?LinkId=187903&clcid=0x409>).

See Also

[Plan InfoPath Forms Services \(SharePoint Server 2010\)](#)

[Sandboxed solutions overview \(SharePoint Server 2010\)](#)

Plan a forms-driven application

Many SharePoint Server applications contain InfoPath forms. A subset of these applications are actually driven by a form. These form-driven applications usually share the following characteristics:

- They automate a business process, such as placing an order or completing employee performance evaluations.
- There is a key piece of structured information, instances of which flow through different activities to complete the business process.

Although each form-driven application is unique, the structure found in form-driven applications often is of a common design. If your application fits this common design, you might be able to use the design presented in this article and modify it to fit your specific case.

This article describes a design for a particular kind of Microsoft SharePoint Server 2010 application that uses forms. It does not cover how to design other kinds of SharePoint Server applications or how to design the forms themselves. For more information about how to design Microsoft InfoPath 2010 forms, see [Office.com](http://go.microsoft.com/fwlink/?LinkId=187550&clcid=0x409) (<http://go.microsoft.com/fwlink/?LinkId=187550&clcid=0x409>).

In this article:

- [Structure of a form-driven application](#)
- [About planning a common form-driven application](#)
- [Identifying the key piece of information](#)
- [Using a list or a form library](#)
- [Workflow](#)
- [Additional data sources](#)
- [Portals](#)
- [Summary](#)

Structure of a form-driven application

A complex form-driven SharePoint Server application can contain the following components:

- A SharePoint site in which to host the application.
- A form template that captures the core piece of information. The form template might have different views for different groups of users or for different stages of the information's life cycle.
- A SharePoint list or library in which to store instances of the completed form template (known as *forms*).
- A workflow that routes an item through a business process. The workflow starts when a new form is created.

- SharePoint lists that contain auxiliary information that is used to populate fields in the form template. Forms and workflows might be associated with these lists to manage the information in the list.
- External databases or line-of-business (LOB) applications that provide data for the form template or the workflow.
- Business logic represented as validation rules in the form template or as part of the workflow.
- A Web page that serves as a portal and enables users to create a new instance of the form template and to view other information about the forms. There might be multiple portals for different audiences.

Your application does not have to match this structure exactly. Some form-driven SharePoint Server applications do not contain all these components, and other applications add small variations, such as having more than one workflow.

About planning a common form-driven application

To design a common kind of form-driven application, you first determine the key piece of information that drives the business process. Then you decide whether to store the information in a SharePoint list or a library, and define the workflow that is used to process the information. Next you determine any additional data sources that will be needed. Finally, you design the portals through which users will access the application.

Identifying the key piece of information

The first step in planning a form-driven application is to determine the key piece of information around which the application revolves. In many situations, the key piece of information is obvious. In a helpdesk application, for example, the key piece of information is probably a service request. In an employee performance review process, the key piece of information is probably a performance review form. In a purchasing system, the key piece of information is probably an order.

Identify the key piece of information that drives the process. If the key piece of information is not obvious, consider the following suggestions:

- If the application will automate an existing process, is there a document or a file that is handed off from one person to another as the process progresses? That document or file is likely to be the key piece of information.
- Does a process start when an item is created or when an item appears in a certain location? That item could be the key piece of information.
- The key piece of information probably has some structure, and might grow or change as it is processed. For example, an order contains the customer's name and address, a list of items complete with quantity and price, and other details. More information such as a tracking number is added to the order as it is processed.

- The key piece of information probably has a status associated with it, and the status changes over time.

If you cannot determine the key piece of information that drives the process, the design presented in this article probably does not fit your application.

When you implement the application, you will create a form template for this key piece of information. This form template will be called the “core form” throughout this article.

Using a list or a form library

Determine whether you will store instances of the core form in a SharePoint list or in a SharePoint Server form library.

If possible, use a list. A list-based solution is simpler and more efficient. However, there are certain situations in which a list will not work. If any of the following conditions is true, use a form library:

- You have to keep a history of changes to form instances.
- The core form contains repeating sections, such as an arbitrary number of accomplishments in an employee’s review form.
- The core form has nested data, such as an order form that contains an item, where an item can contain a product code, a quantity, a size, and a price.
- The core form will contain code.

The following are some situations in which a form might contain code:

- The form includes buttons that perform custom actions.
- The value of one field in the form is based on a complex combination of other values in the form.
- Instances of the core form will be digitally signed.
- You have to store data about each instance of the core form in XML.

If you store instances of the core form in a list, each field in the core form will become a column in the list, and each instance of the core form will become a list item. If you store instances of the core form in a form library, each instance will be turned into an XML document, and the documents will be stored in the library.

Workflow

The business process starts when something happens to an instance of the core form. Often, creating a new instance of the core form itself starts the business process, but other events such as an instance of the core form being modified or being assigned to a person could also start a process.

The business process routes the instance of the core form through the people and systems that have to perform actions. If the core form were a service request, for example, creating a new service request might start a process that assigns the service request to a service representative to interact with the person who originated the request. The service representative might take various actions depending on

the outcome of the discussion with the originator: for example, escalate the request to a senior representative, mark the request as resolved, forward the request to the orders department if the originator should be sent a replacement, and so on.

Identify the steps and decision points that are involved in processing an instance of the core form. This sequence of steps will be represented in SharePoint Server as a workflow. For more information about workflows, see [Plan workflows \(SharePoint Server 2010\)](#).

Additional data sources

A form template can retrieve data from external sources such as a database, a Web service, or a SharePoint list. One common use for external data is to populate a list of valid values for a field in the form template, such as a list of cost centers. You could also use a rule to calculate the value of one field based on a combination of external data and the values of other fields. For example, the value of an “approver” field might be obtained by using an external data source to look up the manager of the employee whose name was entered in the “submitted by” field.

Identify the external data that the core form will access. For each source of external data, indicate where the data comes from. For example, does the data come from a SharePoint list, an SQL database, a LOB system such as SAP, or some other source?



Note:

You can access some LOB data directly from SharePoint Server lists by creating an external content type. For more information about how to create external content types, see [Business Connectivity Services overview \(SharePoint Server 2010\)](#).

For any SharePoint lists that provide data to the core form, consider how you will manage the data in the list. Will you create a form to enter new data in the list? Are workflows needed to manage items in the list? For example, if the core form uses a list of cost centers, you might add an Approval workflow to the list.

Portals

Who will use the application? Are there different user roles, with members of one role performing different actions or viewing different information than users of other roles? If users in different roles will do different things with the application, consider creating a portal for each role. Tailor the actions and information that are available in each portal to the role of users who are using the portal.

For example, in an employee performance review application, there are probably at least three roles:

- Employees, who complete performance review forms.
- Managers, who add information to performance review forms and approve performance reviews.
- Human resources professionals, who create reports and aggregate information from performance reviews.

Employees would access the performance review application through an employee portal that could enable them to create a new performance review form and to track whether their own performance

review was approved by their manager. Managers would access the application through a manager portal that might show a list of their employees with an indication of whether the employee had already submitted a performance review form, and a link to open an employee's performance review form. Human resource professionals would access the application through an HR portal that might show summary statistics of how many performance review forms were approved, submitted but not yet approved, or not yet submitted.

The simplest kind of portal to create is merely a view over the SharePoint list or library in which instances of the core form are stored. You can use a filter or apply conditional formatting to customize the view to the specific user.

You can also design a custom Web page for each user role, and give each user the URL appropriate to their role to access the application. On the portal Web pages, you might include some of the following elements:

- A **New** button to create a new instance of the core form.
- A summary/detail view of instances of the core form. The summary would be a list of form instances, filtered by some criteria, and the detail would be the details of whichever form instance was selected in the summary. You can create the summary view by using a List View Web Part or a Content Query Web Part. You can create the detail view by hosting the core form in the InfoPath form Web Part. For more information about the List View Web Part, see [the Microsoft SharePoint team blog](http://go.microsoft.com/fwlink/?LinkId=187653&clcid=0x409) (http://go.microsoft.com/fwlink/?LinkId=187653&clcid=0x409). For more information about the Content Query Web Part, see [How to: Customize the Content By Query Web Part by Using Custom Properties](http://go.microsoft.com/fwlink/?LinkId=187635&clcid=0x409) (http://go.microsoft.com/fwlink/?LinkId=187635&clcid=0x409). For more information about the InfoPath form Web Part, see [How to: Host an InfoPath Form in the InfoPath Form Web Part](http://go.microsoft.com/fwlink/?LinkId=187637&clcid=0x409) (http://go.microsoft.com/fwlink/?LinkId=187637&clcid=0x409). You can use a different filter on the portals for different roles. For example, on the portal for users who create new form instances, you might filter by instances that were created by the same user. On a portal for approvers, you might filter on instances awaiting the user's review and approval.
- Statistics about the process, such as the number of form instances processed per day, or measurements related to the subject area of the application.

Summary

If you have been able to determine characteristics of your application that correspond to most of the previous sections, it is likely that you can implement the application by following the paradigm of a form-driven application. Create a SharePoint site to host the application. Create a form template for the core form; create a list or a library to store instances of the core form; and associate the form template with the list or library. Add a workflow that is triggered when a new form is added to the list or library. Create and populate any additional lists that are needed to provide data for the form template. Create one or more portals through which users will interact with the application.

See Also

[About forms in SharePoint Server 2010](#)

[InfoPath forms administration \(SharePoint Server 2010\)](#)

Plan for user form templates (SharePoint Server 2010)

This article describes the settings that are specific to user form templates and help you plan which InfoPath Forms Services features to enable for user form templates.

In this article:

- [About user form templates](#)
- [Browser-enabled user form templates](#)
- [Plan external data access](#)
- [Data connection library](#)

About user form templates

Form templates can be divided into three categories depending on how they are deployed: user form templates, administrator-approved templates, and form templates as sandboxed solutions. User form templates are form templates that are deployed by a non-administrator, for example, a form designer. They can be deployed by any user who has the Design permission-level on the site. User form templates can be published to a list or a form library.

Administrator-approved form templates are form templates that must be deployed by a farm administrator. Administrator-approved form templates may contain code or business logic that requires additional permission levels. In order for an administrator-approved form template to be available widely, it must be uploaded to the Central Administration site by a farm administrator and then activated by a site collection administrator.

Form templates as sandboxed solutions are form templates that have code and that are published to form libraries as sandboxed solutions. This lets a form designer who is a site collection administrator publish forms without requiring farm administrator approval. The sandboxed solutions run in an environment that has access to a subset of the server object model. A farm administrator can set usage limits to protect the server from malicious code. For more information about sandboxed solutions, see [Plan for sandboxed solutions \(SharePoint Server 2010\)](#) and [Enable sandboxed solutions on the farm \(SharePoint Server 2010\)](#).

For more information about forms and form templates, see [About forms in SharePoint Server 2010](#).

Browser-enabled user form templates

By default, when form designers create a form template, the form template is browser-enabled so that users can fill it out in a Web browser. This enables users to fill out the forms when they do not have Microsoft InfoPath Filler 2010 installed on their computers. You must decide the following:

**Note:**

These settings apply only to user form templates that are published to form libraries. Workflow form templates and list form templates are not affected.

- **Do you want to allow users to publish browser-enabled user form templates?** You can decide whether to allow form designers to publish browser-enabled form templates to the SharePoint Server farm. If form designers cannot publish browser-enabled form templates, they must ask farm administrators to publish any browser-enabled form templates for them. Restricting the ability to publish Web browser forms to administrators gives you tighter control over the availability of Web browser forms.
- **Do you want to render browser-enabled user form templates?** You can decide whether users can use a Web browser to fill out browser-enabled user forms. If you decide not to render browser-enabled user form templates, the forms will open in InfoPath Filler 2010 if InfoPath is installed on the user's computer.

By default, browser-enabled user form templates are published and rendered. For more information about how to change these settings, see the [Configure browser-enabled user form templates](#) section in "Configure InfoPath Forms Services".

Plan external data access

A user form template can access data from external sources such as a database, a Web service, or a SharePoint list. This section summarizes the settings that affect external data access that you must consider when you plan to deploy user form templates.

- Cross-domain access
- InfoPath Forms Services Web service proxy
- Authentication information in data connection files

Cross-domain access

By default, user form templates cannot make cross-domain connections. That is, any external data connections must reside within the same domain as the SharePoint Server farm where the user form template is published. If you have user form templates that must access data that is in a different domain, you can enable cross-domain access for user form templates.

For more information about how to change the cross-domain access setting, see the [Configure authentication and data connection settings for InfoPath Forms Services](#) section in "Configure InfoPath Forms Services".

InfoPath Forms Services Web service proxy

Form templates can use the InfoPath Forms Services Web service proxy to connect to a Web service. The InfoPath Forms Services Web service proxy forwards SOAP requests to a Web service to enable authorization. The InfoPath Forms Services Web service proxy uses a Secure Store Service application

ID that is contained in a data connection file for authentication. Using the InfoPath Forms Services Web service proxy makes possible the following two scenarios that are not otherwise achievable with forms that are running in a Web browser:

- Authentication to a Web service on a third tier when Windows NTLM authentication would typically prevent re-use of user credentials.
- Symmetrical authentication between a form in InfoPath Filler and the same form that is running in a Web browser.

By default, user form templates cannot use the InfoPath Forms Services Web service proxy. For more information about how to enable user form templates to use the InfoPath Forms Services Web service proxy, see [Configure Web service proxy for InfoPath Forms Services \(SharePoint Server 2010\)](#).

Authentication information in data connection files

Form templates can use data connection (.udcx) files to specify data connection settings for forms. A data connection is a dynamic link between a form template and a data source that stores or provides data for the form template. Data connection files can include alternate authentication information such as a Secure Store Service application ID. By default, user form templates cannot use authentication information in a data connection file. You must enable user form templates to use authentication information in data connection files if you have the following scenarios:

- You want user form templates to be able to use the InfoPath Forms Services Web service proxy for authentication.
- You have Web browser user forms that require multi-tier authentication. For example, if you have a Web browser user form that has to access a database server, you may encounter multi-tier authentication problems. In this scenario, a user's Windows credentials would have to be passed from the Web browser to the server that InfoPath Forms Services is running on and then to the database server. This kind of delegation is not allowed in the NTLM authentication system that is used by Microsoft Windows. Multi-tier authentication problems occur more frequently with Web browser forms because forms that are opened in InfoPath Filler can authenticate directly with the data source.

By default, user form templates cannot use authentication information in a data connection file. For more information about how to change this setting, see the [Configure authentication and data connection settings for InfoPath Forms Services](#) section in "Configure InfoPath Forms Services".

Data connection library

SharePoint Server provides a centrally managed data connection library that enables farm administrators to manage the data connection files that are used by administrator-approved form templates. You should consider creating a data connection library that can be used by user form templates. A data connection library is a specific kind of library that is used to store data connection files.

A data connection library provides the following benefits:

- Connection settings can be shared across multiple forms and multiple forms designers.
- If a data connection setting has to be changed, you can update the data connection file without modifying the form templates that use that data connection file.
- By default, approval is required before a data connection file can be used by everyone. This enables the site collection administrator (and any designated approvers) to control the kinds of connections to allow.

To effectively plan for a data connection library, consider the following:

- **How many data connection libraries do you need?** The files in the data connection library can be used by forms and form templates that are in the same site collection as the data connection library.
- **Who should be able to upload data connection files?** Users with the Contributor permission-level can upload data connection files to the library.
- **Who should be able to approve the data connection files?** When a data connection file is uploaded, the owner of the file can use the file. However, approval is required before the data connection file can be used by anyone else. Users with the Design permission-level can approve data connection files.

For more information about data connection libraries, see [How to: Create and Use a Data Connection Library](http://go.microsoft.com/fwlink/?LinkId=188117) (<http://go.microsoft.com/fwlink/?LinkId=188117>).

See Also

[About forms in SharePoint Server 2010](#)

[Configure the Secure Store Service \(SharePoint Server 2010\)](#)

Plan to upgrade form templates during an upgrade to SharePoint Server 2010

This article describes the additional steps that are required to upgrade form templates when you upgrade InfoPath Forms Services from the version in Microsoft Office SharePoint Server 2007 and Microsoft Office Forms Server 2007 to the version in Microsoft SharePoint Server 2010.

In this article:

- [About upgrading forms during an upgrade to SharePoint Server 2010](#)
- [Upgrade form templates during a database attach upgrade to SharePoint Server 2010](#)
- [Upgrade form templates during an in-place upgrade to SharePoint Server 2010](#)

About upgrading forms during an upgrade to SharePoint Server 2010

When you upgrade from Office SharePoint Server 2007 or Microsoft Office Forms Server 2007 to SharePoint Server 2010, InfoPath Forms Services is also upgraded.

The following upgrade and migration scenarios are relevant for InfoPath Forms Services in SharePoint Server 2010:

- Database attach upgrade

Upgrades forms and form templates from InfoPath Forms Services in Microsoft Office SharePoint Server 2007 or from Microsoft Office Forms Server 2007 to another server that is running SharePoint Server 2010. Existing administrator-approved form templates and data connections must be migrated to the new deployment and upgraded to work with InfoPath Forms Services in Microsoft SharePoint Server 2010.



Note:

A database attach upgrade is the only way to upgrade Microsoft Office Forms Server 2007 to SharePoint Server 2010.

- In-place upgrade

Upgrades forms and form templates from InfoPath Forms Services for Microsoft Office SharePoint Server 2007 to SharePoint Server 2010 on the same server. All forms and form templates are upgraded automatically during an in-place upgrade.

For more information about how to upgrade from Office SharePoint Server 2007 to SharePoint Server 2010, see [Upgrading to SharePoint Server 2010](#).

Upgrade form templates during a database attach upgrade to SharePoint Server 2010

A database attach upgrade enables you to move your content to a new farm or new hardware. During a database attach upgrade, you detach all the content databases from an existing farm and then attach the databases to a new server farm installation. When you attach the databases to the new server farm, the upgrade process runs and upgrades the data in place. A database attach upgrade is the only way to upgrade Microsoft Office Forms Server 2007 to SharePoint Server 2010.

The content database contains only some files that are used by InfoPath forms. These files include the following:

- User form templates (.xsn), which are form templates that are deployed by non-administrators. They do not require full trust and do not use data connections that are managed by an administrator. User form templates (.xsn) can be published to lists and form libraries by individual contributors.
- Data connection files (.udcx) for user form templates.
- Content from filled-out forms (.xml) in lists and form libraries.

These files are upgraded automatically when you perform a database attach upgrade.

On the other hand, administrator-approved form templates and data connection files in the Central Administration libraries are stored in the configuration database and must be migrated separately. These files include the following:

- Administrator-approved form templates (.xsn files) that are uploaded to the Manage Form Templates library on the Central Administration site.
- Data connection files (.udcx files) that are uploaded to the Manage Data Connection Files library on the Central Administration site.

If you copy and attach the content database before these files have been imported, the sites that contain content that depend on administrator-approved forms will produce errors during import.

To avoid these errors, we recommend that you follow these steps in the order listed:

1. Perform the prerequisite steps for upgrading to SharePoint Server 2010.
2. Install and configure SharePoint Server 2010 on the new server.
3. Export administrator-approved form templates and data connection files from the configuration database on the original server that is running Office SharePoint Server 2007 or Microsoft Office Forms Server 2007. For more information about how to export and import administrator-approved forms, form templates, and data connection files between configuration databases, see the [Export and import administrator-approved form template files between configuration databases](#) section later in this article.
4. Import the files to the configuration database on the new server that is running SharePoint Server 2010.
5. Follow the steps in [Attach databases and upgrade to SharePoint Server 2010](#) to attach and upgrade the databases from the original server to the new SharePoint Server 2010 server.

6. If the URL of the new server differs from the URL of the previous server, run a Windows PowerShell 2.0 cmdlet to update links that are used in the upgraded forms and form templates. For more information about how to update the links that are used in upgraded forms and form templates, see the [Update form template links to the server](#) section later in this article.

Export and import administrator-approved form template files between configuration databases

The forms data in the content database includes only the files that are used by user forms, form templates, and data connection files that are uploaded to lists and document libraries. It does not include administrator-approved form templates and data connection files that are uploaded to the Central Administration site.

To avoid problems when you upgrade form templates, export the content in the configuration database of the original server that is running Office SharePoint Server 2007 or Microsoft Office Forms Server 2007 and import it to the new server that is running SharePoint Server 2010.

To export administrator-approved form templates and data connection files that are stored in the Office SharePoint Server 2007 or Microsoft Office Forms Server 2007 configuration database, use the **-o exporttipsadminobjects** command in the Stsadm command-line tool.

To import administrator-approved form templates and data connection files to the configuration database on the new server, use the Windows PowerShell 2.0 [Import-SPInfoPathAdministrationFiles](#) cmdlet.

Data-connection time-out settings and other settings on the Configure InfoPath Forms Services administration page are not migrated and must be configured on the new server.

For more information about how to use the Stsadm command-line tool, see [Stsadm command-line tool \(Office SharePoint Server\)](#) (<http://go.microsoft.com/fwlink/?LinkId=186993>).

Update form template links to the server

During upgrades or content migrations, it is common to change the URL of the server that contains forms and form templates. If the URL of the new server differs from the URL of the previous server, links that forms and form templates use might have to be updated to the URL of the new server. These include links in the form data to the form template file location, or links in the form template to data connection files or other resource files.

Both user form templates and administrator-approved templates can be affected.

To update links that are used in user form templates and data connection files, run the [Update-SPInfoPathUserFileUri](#) cmdlet.

To update links that are used in administrator-approved form templates and data connection files, run the [Update-SPInfoPathAdminFileUri](#) cmdlet.

These cmdlets update the name of the server in URLs that form templates and data connection files use. They do not update links in forms that are moved between libraries by users, or in other forms

content that is stored in the content database. Users can fix those links by using the **Relink All** command that is available in the form library settings. Forms and form templates that are upgraded or migrated by using the database attach process are relinked automatically.

Upgrade form templates during an in-place upgrade to SharePoint Server 2010

Form templates for InfoPath Forms Services are upgraded automatically as part of any in-place upgrade of SharePoint Server.

For more information about how to upgrade in place to SharePoint Server 2010, see [Upgrade in place to SharePoint Server 2010](#).

InfoPath 2010 Enhanced Integration with SharePoint Server 2010 and Its Implications When Designing Forms for Applications (white paper)

The enhanced features of Microsoft InfoPath 2010 and its tighter integration with Microsoft SharePoint Server 2010 make InfoPath 2010 an enterprise-class development tool. With the ability to access any number of data sources dynamically and apply sophisticated rule sets to manipulate information and form behavior, InfoPath forms and Web Parts can address the most demanding enterprise application requirements.

This white paper describes how to use InfoPath List forms, InfoPath document-based forms, and InfoPath Web Parts to create enterprise-level applications in SharePoint Server 2010. To download the white paper in the format that you want, click the following links:

- [InfoPath 2010 Enhanced Integration with SharePoint Server 2010 and Its Implications When Designing Forms for Applications \(Microsoft Office Word\)](http://go.microsoft.com/fwlink/?LinkId=190499&clcid=0x409)
(http://go.microsoft.com/fwlink/?LinkId=190499&clcid=0x409)
- [InfoPath 2010 Enhanced Integration with SharePoint Server 2010 and Its Implications When Designing Forms for Applications \(PDF\)](http://go.microsoft.com/fwlink/?LinkId=190500&clcid=0x409)
(http://go.microsoft.com/fwlink/?LinkId=190500&clcid=0x409)
- [InfoPath 2010 Enhanced Integration with SharePoint Server 2010 and Its Implications When Designing Forms for Applications \(XPS\)](http://go.microsoft.com/fwlink/?LinkId=190501&clcid=0x409)
(http://go.microsoft.com/fwlink/?LinkId=190501&clcid=0x409)

Plan workflows (SharePoint Server 2010)

A *workflow* is a feature of Microsoft SharePoint Server 2010 that moves documents or list items through a specific sequence of actions or tasks related to a business process. Workflows can be used to manage common business processes such as document review or approval.

In this section:

- [Workflows overview \(SharePoint Server 2010\)](#)
This article introduces the types of business processes that workflows can facilitate and describes the workflows included in SharePoint Server 2010.
- [Choose a workflow authoring tool \(SharePoint Server 2010\)](#)
This article describes the different Microsoft supported workflow authoring tools, how they can be used together for rapid workflow authoring.
- [Plan for approval and review processes in workflows \(SharePoint Server 2010\)](#)
Because workflows help an enterprise move documents through the review and approval processes, this article explains how the Approval workflow works and how to plan to use it effectively.
- [Plan for workflow security and user management \(SharePoint Server 2010\)](#)
This article highlights some aspects of workflow behavior that relate to security and raises other issues for administrators and workflow developers to consider when they plan to configure and develop workflows.
- [Approval Workflow: A Scenario \(SharePoint Server 2010\)](#)
This article shows how an approval-type workflow that is created in Microsoft SharePoint Designer 2010 or Workflow Designer in Visual Studio 2010, and that is then hosted by using SharePoint Server 2010 might look.

See Also

[Workflow administration \(SharePoint Server 2010\)](#)

Workflows overview (SharePoint Server 2010)

The *workflow* feature in Microsoft SharePoint Server 2010 enables solution architects, designers, and administrators to improve business processes. Fundamentally, a workflow consists of two things: the forms that a workflow uses to interact with its users and the logic that defines the workflow's behavior. Understanding how workflows are created requires knowledge about both.

In this article:

- [Workflow overview](#)
- [Benefits of using workflows](#)
- [Predefined workflows](#)
- [Sample workflow scenario](#)
- [Workflow types: Declarative and compiled](#)
- [Workflow templates](#)
- [Workflow associations](#)
- [Office client interoperability](#)

Workflow overview

Workflows in SharePoint Server 2010 enable enterprises to reduce the amount of unnecessary interactions between people as they perform business processes. For example, to reach a decision, groups typically follow a series of steps. The steps can be a formal, standard operating procedure, or an informal implicitly understood way to operate. Collectively, the steps represent a business process. The number of human interactions that occur in business processes can inhibit speed and the quality of decisions. Software that simplifies and manages this "human workflow" enables the automation of interactions among groups who participate in the process. This automation results in more speed, overall effectiveness of the interactions, and often a reduction in errors.

You can model business processes by using flow charts, such as those created using Microsoft Visio 2010 and can represent business processes by using workflow terminology. You can automate business processes, such as document approval, by associating a workflow with data in SharePoint Server 2010. For example, you can create a workflow to route a document for review, track an issue through its various stages of resolution, or guide a contract through an approval process.

One problem that many IT departments face when implementing business processes that require participation of information workers is that those processes do not integrate with the way people actually work. For a business process to be effective, it must be integrated with the familiar, everyday tools and applications used in the workplace so that it becomes part of the daily routine of information workers. In the electronic workplace, this includes integration with e-mail, calendars, task lists, and collaboration Web sites.

Benefits of using workflows

The primary benefits of using workflows are to facilitate business processes and improve collaboration.

Business processes that enterprises use depend on the flow of information or documents. These business processes require the active participation of information workers to complete tasks that contribute to their workgroup's decisions or deliverables. In SharePoint Server 2010, these types of business processes are implemented and managed by using workflows.

Examples of business processes that could be facilitated by workflows include:

- **Contract approval** Guiding a proposed contract among members of an organization who must approve or reject it.
- **Expense reporting** Managing the submission of an expense report and associated receipts, reviewing the report, approving it, and reimbursing the submitter.
- **Technical support** Guiding the progress of a technical support incident as it is opened by a customer, investigated by a support engineer, routed to technical experts, resolved, and added to a knowledge base.
- **Interviewing** Managing the process of interviewing a job candidate. This includes scheduling and tracking interview appointments, collecting interview feedback as it accumulates, making that feedback available to subsequent interviewers, and facilitating the hire/no-hire decision.
- **Content publishing** Managing the approval of the publication of content on the Internet presence site of an enterprise.

Automating business processes

Businesses depend on business processes. Although those processes often involve software, the most important processes in many organizations depend on people. Workflows can automate interactions among the people who participate in a process to improve how that process functions, increase its efficiency, and lower its error rate.

Many processes can benefit from automated support for human interactions. Examples include the following:

- **Approval** A common aspect of human-oriented business processes is the requirement to get approval from multiple participants. What is being approved can vary widely, ranging from a Microsoft Word document that contains next year's marketing plan to an expense report from a trip to a conference. In every case, some number of people must review the information, perhaps appending comments, and then indicate approval or rejection.
- **Coordinating group efforts** Whether it is preparing a response to a request for proposal (RFP), managing the translation of a document into one or more languages, or something else, many processes require people to work together in an organized way. By defining the steps of the process through an automated workflow, the group's work can be made more efficient and the process itself more predictable.

- **Issue tracking** Many business processes generate a list of outstanding issues. An automated workflow can be used to maintain that list, assign issues to the people who can resolve them, and track the status of that resolution.

To support these kinds of automated business processes, SharePoint Server 2010 can run workflow applications. Based on Windows Workflow Foundation 3.5, these applications interact with people through a Web browser. For more information about Windows Workflow Foundation 3.5, see [Windows Workflow Foundation](http://go.microsoft.com/fwlink/?LinkId=127778) (http://go.microsoft.com/fwlink/?LinkId=127778).

Workflows improve collaboration

Workflows help people collaborate on documents and manage project tasks by implementing business processes on documents and items on a SharePoint site or site collection. Workflows help organizations follow consistent business process practices. Workflows increase organizational efficiency and productivity through management of the tasks and steps involved in those business processes. Workflows speed up decision making by helping to ensure that the appropriate information is made available to the appropriate people at the time that they need it. Workflows also help ensure that individual workflow tasks are completed by the appropriate people and in the appropriate sequence. This enables the people who perform these tasks to concentrate on performing the work instead of on the work processes.

For example, on a SharePoint Server 2010 site, you can create a workflow to be used with a document library to route a document to a group of people for approval. When the author starts this workflow, the workflow creates document approval tasks, assigns these tasks to the workflow participants, and then sends e-mail alerts to the participants.

When the workflow is in progress, the workflow owner or the workflow participants can check progress on the Workflow Status page. When the workflow participants complete their workflow tasks, the workflow ends, and the workflow owner is automatically notified that the workflow has finished.

Predefined workflows

Microsoft SharePoint Server 2010 includes the following predefined workflow templates that address common business scenarios:

- **Collect Feedback** Routes a document or item to a group of people for feedback. Reviewers can provide feedback, which is then compiled and sent to the person who initiated the workflow. By default, the Collect Feedback workflow is associated with the Document content type, and therefore it is automatically available in document libraries.
- **Approval** Routes a document or item to a group of people for approval. By default, the Approval workflow is associated with the Document content type, and therefore it is automatically available in document libraries. A version of the Approval workflow is also associated by default with the Pages library on a publishing site, and can be used to manage the approval process for the publication of Web pages.

The Approval workflow is a staged approval model (that is, the first set of approvers can undergo the review and approval process, then the next set of approvers, and so on). Each stage or approval set can also have its own behavior. For example, members of the first group of approvers can do their review in serial approval order (one after the other), members of the second group can do their review in parallel (reviewers can provide feedback in any order), and so on.

- **Disposition Approval** Manages document expiration and retention by letting participants to decide whether to keep or delete expired documents. The Disposition Approval workflow supports record management processes and is intended for use primarily in a Records Center site.
- **Collect Signatures** Routes a document that was created in a Microsoft application to a group of people to collect their digital signatures. This workflow must be started in applications in the 2007 Microsoft Office system and the Microsoft Office 2010 suites such as Microsoft Word. Participants must complete their signature tasks by adding their digital signatures to the documents in the relevant client program. By default, the Collect Signatures workflow is associated with the Document content type, and therefore is automatically available in document libraries. However, the Collect Signatures workflow appears for a document in the document library only if that document contains one or more Microsoft Office Signature Lines.

For more information on Microsoft Office Signature Lines, see [Add or remove a digital signature in Office documents](http://go.microsoft.com/fwlink/?LinkId=157408) (<http://go.microsoft.com/fwlink/?LinkId=157408>).

- **Three-state** Designed to track the status of a list item through three states (phases). It can be used to manage business processes that require organizations to track a high volume of issues or items, such as customer support issues, sales leads, or project tasks.

The Three-state workflow is so named because it tracks the status of an issue or item through three different states, and through two transitions between the states. For example, when a workflow is initiated on an issue in an Issues list, SharePoint Server 2010 creates a task for the assigned user. When the user completes the task, the workflow changes from its initial state (Active) to its middle state (Resolved) and creates a task for the assigned user. When the user completes the task, the workflow changes from its middle state (Resolved) to its final state (Closed), and creates another task for the user to whom the workflow is assigned at that time. Note that this workflow is only supported on lists, not libraries.

- **Translation Management** Manages manual document translation by creating copies of the document to be translated and by assigning translation tasks to translators. This workflow is available only for Translation Management libraries.
- **Issue Tracking** Routes an issue to team members for resolution. It presents a Web page to the user who makes possible the entry of new issues; for example, customer complaints. As an issue progresses through different workflow states, the Web page of the user changes to reflect appropriate events; for example, a Web page that was closed when an issue is resolved.

You can customize each of these workflows for an organization in several ways. For example, when you add a workflow to a list, library, content type, or site to make it available for use on documents or items, you can customize the tasks lists and history lists where information about the workflow is stored.

When a site user starts a workflow on a document, item, or site, the user might have the option to further customize the workflow by specifying the list of participants, a due date, and task instructions. You can also make a copy of a predefined workflow to use as a starting point when creating a custom workflow.

Sample workflow scenario

Imagine that you work for Adventure Works, a sports store franchise that sells bicycles worldwide. This company has sales representatives that visit different countries to help new franchisees open new sports stores.

The scenario described in this section is one where an expense report is submitted for approval. If the expense report is for less than \$5,000.00, a manager is required to approve, disapprove, or forward it. If the expense report is equal to, or more than, \$5,000.00, a manager must review the expense report, comment on it, and then if the manager recommends approval, it is forwarded to a vice-president, who must approve or disapprove it.

In this scenario, the expense report form is an InfoPath Forms Services form. The workflow is a sequential Approval type of workflow project created in Microsoft SharePoint Designer 2010, and is composed of both automated tasks and tasks that require human action. The workflow is running on SharePoint Server 2010.

1. The sales representative — the first workflow participant — browses to an intranet self-service portal and selects the Expense Report form. A data entry page opens. The sales representative first fills out a simple expense report form that contains entries for the person's name, the expense purpose, the expense total, and the name and e-mail address of the person's direct manager. The sales representative then clicks **Submit** to submit the form.

Upon submission of the form, the data is saved centrally, the workflow is initiated, and the review task is assigned to the approver (in this case, the sales representative's manager).

2. The workflow notifies the sales representative's manager. The notification is an e-mail message that contains instructions for completing the task and provides a link to a Web site that displays the Expense Report form.
3. The manager, the second workflow participant, goes to the Web site and reviews the expense report. The workflow task item provides three actions that the manager can perform: Approve, Disapprove or Forward.
 - If the expense report is less than \$5,000.00, the manager sees options to **Approve** or **Disapprove** the expense report.
 - If the expense report is more than \$5,000.00, the manager sees options to **Forward** the expense report to a company vice president, or to **Disapprove** the expense report at the manager's level.

4. The manager takes action to approve, disapprove, or forward the expense report, and the workflow continues:
 - If the expenses are approved by the manager, the task completion sends a message to the workflow indicating that the task is completed, the workflow notifies the sales representative through an e-mail message, and then the workflow adds the expense data to the line-of-business (LOB) accounting system.
 - If the expenses are not approved by the manager, he types an explanation for his decision. The task completion sends a message to the workflow indicating that the task is completed, and then the workflow notifies the sales representative through an e-mail message.
 - If the manager selects the option to forward the expense report to a company vice president, the manager makes relevant comments in the form and then clicks **Forward**. The workflow then notifies the vice president through an e-mail message that contains instructions for completing the task and provides a link to a Web site that displays the Expense Report form.
5. The vice president — the third workflow participant — is given the option to **Approve** or **Disapprove** the expense report. When the vice president acts to approve or disapprove the expense report, the workflow continues.
 - If the vice president approves the expenses, the expense data is added to the accounting system, the workflow notifies the sales representative and manager through e-mail, and then the workflow notifies SharePoint that the task is completed.
 - If the vice president does not approve the expenses, the vice president types an explanation for the decision into the form. The workflow notifies the sales representative and manager through e-mail, and then the workflow notifies SharePoint that the task is completed.

As you can imagine, there are many ways to expand the functionality of this workflow within the context of this scenario. For example, you can configure the workflow so that if the vice president disapproves the expense report, the report is returned to the sales representative's manager. The manager can further justify the expense and resubmit it for approval to the vice president, can pass along the disapproval to the sales representative, or take some other action.

In this sample expense report scenario, the business rules are always the same. This workflow solution defines the manager and vice president approvers, defines the business logic for the routing of the workflow, and predefines the content of the notifications. However, many real-world applications have complex business rules. Routing for approval can depend on many business variables. Notifications can also change, depending on other variables.

For example, imagine that in the same expense reporting solution, you have to route the expense report to as many as ten managers, depending on the expense purpose, the expense total, and the date of submission. Additionally, depending on the expense purpose, the content of the notifications sent by the workflow contain some small differences. This means that there can be multiple workflow solutions with different routing levels and notifications.

Microsoft SharePoint Server 2010 enables you to create and implement workflow solutions to meet the business needs of your organization. It does this by leveraging the workflow design and customization

features of SharePoint Designer 2010 and Microsoft Visual Studio 2010 Premium Edition, the prototyping functionality of Microsoft Visio 2010 used with SharePoint Designer 2010, and the capabilities of the Microsoft Office system.



Note:

You can use the prototyping functionality of Visio 2010 through SharePoint Designer 2010 only if you are running the Premium edition of Visio 2010.

Workflow types: Declarative and compiled

An important distinction to understand about workflows is whether they are a declarative workflow, such as those created using Microsoft SharePoint Designer 2010 or a compiled workflow, such as those created using Visual Studio 2010. A declarative workflow is a workflow that is built from conditions and actions that are assembled into rules and steps, and that sets the parameters for the workflow without writing any code.

A compiled workflow, like declarative workflows, can also be built from conditions and actions without the workflow author actually writing code but also enable the workflow author to add custom code to the workflow. Regardless of whether a workflow author adds custom code to a code-centric workflow, the most important distinction to understand is the difference in the way that declarative and compiled workflows are run on the server. A compiled workflow is stored on a server running SharePoint Server 2010 as a precompiled dll file whereas a declarative workflow is deployed on a server running SharePoint Server 2010 as an Extensible Object Markup Language (XOML) file and compiled in the content database each time an instance of the workflow is started. For more information about the Microsoft supported tools for authoring workflows, see [Choose a workflow authoring tool \(SharePoint Server 2010\)](#).

Workflow templates

When creating a custom workflow using SharePoint Designer 2010, you can choose to create a workflow that will only be used with a specific list, library, content type, or site. Alternatively, you can choose to create a reusable workflow template, which can be associated with multiple lists, libraries, content types, or sites.



Note:

SharePoint Designer 2010 does not support creating reusable workflows for sites. Instead, you can use Visual Studio 2010 to create them.

When authoring a workflow, you can also choose to make it global, which means that once it is activated on a site it will be active for all the sub-sites below that site as well. However, you cannot use SharePoint Designer 2010 to create a global workflow and then save the workflow as a WSP file.

Workflow associations

SharePoint Server 2010 takes advantage of the Workflow Foundation runtime. One or more workflow templates, each containing the code that defines a particular workflow, can be installed on a server. Once this is done, an association can be created between a specific template and a document library, list, content type, or site. This template can then be loaded and executed by the SharePoint Server 2010-hosted Workflow Foundation runtime, creating a workflow instance.

Like all Workflow Foundation workflows, those based on SharePoint Server 2010 rely on Workflow Foundation runtime services. The Workflow Foundation standard persistence service allows the state of a persisted workflow to be linked with the document or item, and allows for long-running business processes that can span days, months, or years.

SharePoint workflows can be associated with lists, libraries, and content types. Reusable workflows created using Visual Studio 2010 can also be associated with sites. The following table describes the minimum permissions required to associate a workflow.

Associate workflow with	Minimum permissions required
List or library	Full Control permission level on the list or library
List or library content type	Member of the Site Owners group on the SharePoint site
Site content type	Member of the Site Owners group on the SharePoint site
Site	Member of the Site Owners group on the SharePoint site

For more information about workflow associations, see [Add a workflow association \(SharePoint Server 2010\)](#).

Office client interoperability

Implementation of business processes that require the participation of information workers is often hampered by the fact that those processes are rarely integrated with the way people work. For a business process to be effective, it must be integrated with the familiar, everyday tools and applications used in the workplace so that the process becomes part of the daily routine of information workers. In the electronic workplace, this includes coordination with e-mail, calendars, task lists, collaboration Web sites, and client applications such as Microsoft Outlook and Microsoft Word. This integrated approach is implemented in workflows based on SharePoint Server 2010.

SharePoint Server 2010 delivers workflow applications through a technology called Windows Workflow Foundation. People interact with these workflows in the Web browser and in the 2007 Microsoft Office

system and the Microsoft Office 2010 suites through programs such as Microsoft Word. SharePoint Server 2010 uses Windows Workflow Foundation functionality to deliver several workflows that automate common business processes. Additionally, custom workflows can be created by using various tools, and the workflows are then associated with SharePoint libraries, lists, content types, and sites.

Choose a workflow authoring tool (SharePoint Server 2010)

What is a workflow? Fundamentally, it consists of two things: the forms a workflow uses to interact with its users and the logic that defines the workflow's behavior. Understanding how workflows are created requires knowing something about both.

Because it communicates with users through a Web browser, a workflow relies on ASP.NET to display its forms. Accordingly, those forms are defined as .aspx pages. A workflow can potentially display its own forms at four points in its lifecycle:

- **Association:** When an administrator associates a workflow template with a particular document library or list, he might be able to set options that will apply to every workflow instance created from this association. If a workflow author chooses to allow this, she must provide a form that lets the administrator specify this information.
- **Initiation:** The initiator of a workflow might be allowed to specify options when he starts a running instance. In the approval scenario just described, for instance, the options included specifying the list of workflow participants and defining how long each one had to complete his or her task. If a workflow allows this, its author must provide a form to allow the initiator to set these options.
- **Task Completion:** The running workflow instance must display a form to the participants in the workflow to let them complete their task. This form is what allowed the approvers in the earlier scenario to make comments on the document and indicate their approval or rejection.
- **Modification:** The creator of a workflow can allow it to be modified while it's running. For example, a workflow might allow adding new participants after it has begun executing or extending the due date for completing tasks. If this option is used, the workflow must display a form at this point to let a participant specify what changes should be made.

Workflows built by using Microsoft SharePoint Server 2010 can use forms created with InfoPath. A workflow's logic is always defined as a group of activities, just as with any workflow based on the Windows Workflow Foundation (WF). To specify the logic and forms for a workflow, Microsoft provides two different tools, each targeting a different audience. Software developers can use the Workflow Designer feature of Windows Workflow Foundation. This tool runs inside Visual Studio 2010 Professional Edition and provides a graphical environment for organizing activities into workflows. Information workers, a less technical group, can use Microsoft SharePoint Designer 2010 to create workflows without writing code. The next two sections examine how workflows can be created by using each of these tools.

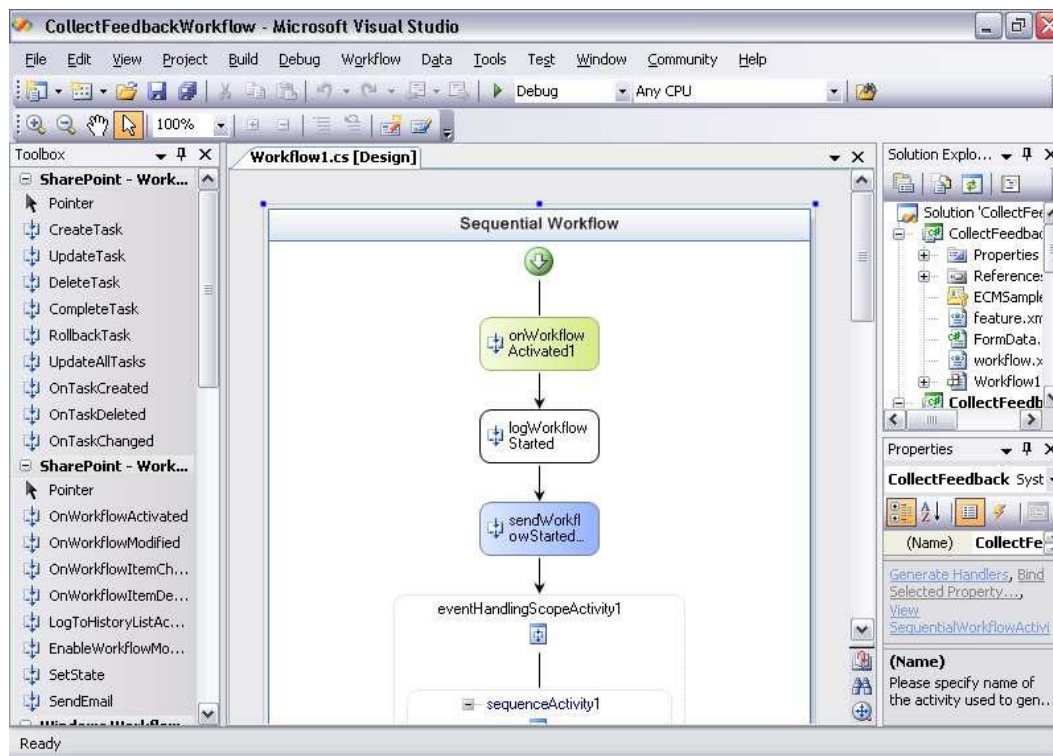
Authoring workflows with Visual Studio 2010 and WF Workflow Designer

Workflows using SharePoint Server can use Microsoft InfoPath 2010 workflow forms rather than just .aspx forms. To create these forms, a workflow author uses Microsoft InfoPath. This tool provides a graphical editor that lets an author define the form's content. Developers who prefer to work entirely within the Visual Studio environment can use the Professional Edition of Visual Studio 2010.

Once they're created, InfoPath workflow forms are attached to a workflow by using a workflow.xml file, just as with ASP.NET forms. Unlike ASP.NET forms, however, developers don't need to write custom code to move information between InfoPath workflow forms and a workflow. Instead, SharePoint Server and InfoPath provide this link, making life simpler for the people who create workflows.

In many ways, a workflow is like a flowchart. Given this, it makes sense to provide a graphical tool that lets developers specify a workflow's actions. This tool is SharePoint Workflow tools in Visual Studio 2010 Professional, which is a project type that uses the Windows Workflow Foundation (WF) Workflow Designer, and adds deployment and forms support for Workflows. Developers can use WF Workflow Designer to define graphically a workflow's activities and the order in which those activities will be executed. The screen below shows a simple example of how this looks in Microsoft Visual Studio.

Collect Feedback Workflow



The activities available for use appear in the Toolbox on the left side of the screen. A developer can drag these activities onto the design surface to define the steps in a workflow. The properties of each activity can then be set in the Properties window that appears in the lower right corner.

The Base Activity Library Windows Workflow Foundation provides a group of fundamental activities, as described earlier. SharePoint Server also provides a set of activities designed expressly for creating workflows. Among the most important of these are the following:

- **OnWorkflowActivated:** provides a standard starting point for a workflow. Among other things, this activity can accept information supplied by a SharePoint administrator by using the Association form when the workflow is associated with a document library, list, content type, or site. It can also accept information supplied by the Initiation form when the workflow is started. Every workflow must begin with this activity.
- **CreateTask:** creates a task assigned to a particular user in a task list. For example, the approval workflow in the scenario described earlier used this activity to add a task to the task list used by each of the participants. This activity also has a **SendEmailNotification** property that, when set to true, automatically sends an e-mail message to the person for whom this task was created.
- **OnTaskChanged:** accepts information from the Task Completion form. The approval workflow in the earlier scenario used this activity to accept the input of each participant when the document was approved.
- **CompleteTask:** marks a task as completed.
- **DeleteTask:** removes a task from a task list.
- **OnWorkflowModified:** accepts information from the Modification form, which can then be used to change how this instance of the workflow behaves. If the workflow's creator chooses not to include any instances of this activity in the workflow, that workflow cannot be modified while it's running.
- **SendEmail:** sends email to a specified person or group of people.
- **LogToHistoryList:** writes information about the workflow's execution to a history list. The information in this list is used to let users see where a workflow is in its execution, look at the workflow's history after it's completed, and more. To allow this kind of monitoring, the workflow's author must write information to a History list at appropriate points in the workflow's execution. Because it provides its own mechanism for tracking workflows, SharePoint Server doesn't support WF's standard tracking service.

A typical pattern for a simple workflow begins with an **OnWorkflowActivated** activity, and then uses a **CreateTask** activity to assign a task to a participant in the workflow. The BAL's standard **While** activity might then be used to wait until the user completes the task. To learn when this has happened (perhaps the user makes multiple changes to the task, then checks a box on the Task Completion form when she's done), an **OnTaskChanged** activity executes within the **While**, extracting whatever information the user has entered on that form. When the user has completed the task, a **CompleteTask** activity might execute, followed by a **DeleteTask**. The workflow can then go on to the next participant, using **CreateTask** to assign a task to him, and so on. And of course, other things can occur, such as sending e-mail, logging information to the history list, or even including the BAL's **Code** activity, which allows running arbitrary code.

All of the activities provided by SharePoint Server are concerned with letting workflows operate within the SharePoint environment. The business logic a workflow implements is entirely up to the creator of that workflow. In fact, a developer authoring a workflow is free to create and use her own custom activities—she's not required to use only those provided by SharePoint Server and WF.

As described earlier, Windows Workflow Foundation supports sequential, parallel, and state machine workflows. A workflow created with the WF Workflow Designer can also use any of these options. To allow this, SharePoint Server adds project types to Visual Studio, one for each of these workflow styles.

Whatever style is chosen, the developer must define more than just the workflow's logic; he must also specify the .aspx or InfoPath forms it should use. To do this, the developer relies on a file named `element.xml`. This file provides a template that the developer fills in to specify what form, if any, should be displayed at each of the four points at which a workflow is allowed to do this.

A developer must do some work to pass information between a workflow and the .aspx forms it uses. The **Microsoft.Windows.SharePoint.Workflow** namespace exposes an object model for developers. Using the types in this namespace, the creator of a workflow can pass information from an .aspx form to the workflow and vice-versa.

Once a workflow and its forms have been created, the developer must package them into what is referred to as a Feature. A SharePoint administrator must then install this Feature, which includes installing the workflow's assemblies to the target system's global assembly cache. The new workflow will now be visible to the administrator as a workflow template that can be associated with a document library, list, content type, or site.

For a software developer, creating a workflow by using Visual Studio and the WF Workflow Designer isn't especially hard. The developer needs to understand the specifics of working in this environment, but much of what he's doing will be familiar. Yet software developers aren't the only people who'd like to author workflows. As described next, people who aren't professional developers can also create workflows using Microsoft SharePoint Designer 2010.

Authoring workflows with Microsoft SharePoint Designer 2010

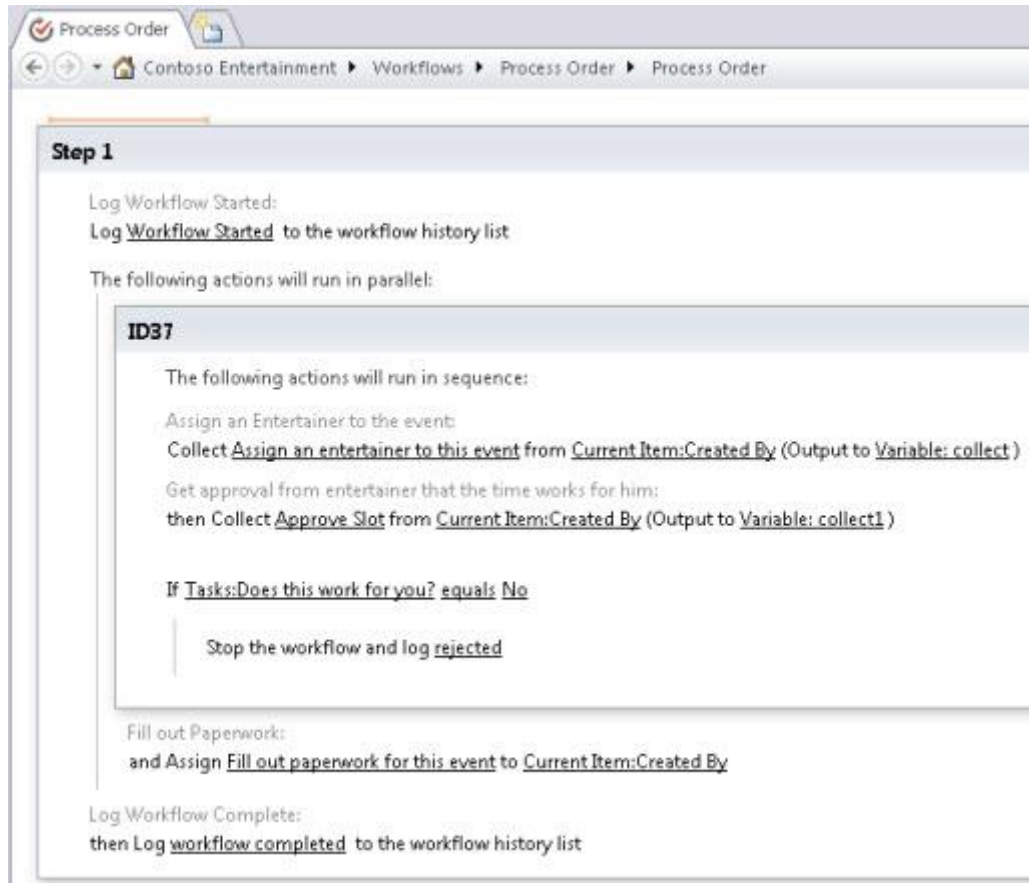
Microsoft SharePoint Designer 2010 is a separate application that is available as a free download. Microsoft SharePoint Designer enables information workers and others to add application logic (implemented as a workflow) to SharePoint sites. This is certainly a useful goal, but Microsoft SharePoint Designer also addresses another important problem. If a developer creates a workflow by using Visual Studio, that workflow must be deployed on a server that's running SharePoint Server like any other feature. Yet many SharePoint administrators won't allow arbitrary code to be deployed on their servers, believing that the risk of destabilizing the system is too great. Being able to create straightforward business logic tied to documents and list items is very useful, however, and it's something that many SharePoint users need. Along with allowing less technical people to create workflows, Microsoft SharePoint Designer also addresses this problem by providing a safer way to define and deploy business logic on servers running SharePoint Server.

The workflow scenarios that Microsoft SharePoint Designer is intended to address are different in some ways from those addressed by Visual Studio and WF Workflow Designer. While it's certainly possible to create complex applications, the intent of Microsoft SharePoint Designer is to let users add business logic to SharePoint sites. For example, suppose that a site contains a list that allows its users to submit change requests. Microsoft SharePoint Designer could be used to create a workflow that automatically informs the submitter when her change request is accepted or rejected. Similarly, a custom workflow might inform a particular group of users whenever a new document is added to a particular document library. Performing this kind of custom notification isn't complicated—creating the workflows is easy—but it's challenging with earlier versions of SharePoint Server because of administrators' reluctance to install user-written code.

There's an obvious question here: why should logic created with Microsoft SharePoint Designer be treated any differently? What makes SharePoint administrators willing to allow workflows built with this tool to be deployed on the systems for which they're responsible? The answer is that workflows built with Microsoft SharePoint Designer can only use activities from an administrator-controlled list. In addition to the activities provided by SharePoint Server, a site administrator can choose whether to include custom activities created by a developer on this list. By defining exactly what workflows are allowed to do, a SharePoint administrator can have more confidence that deploying logic created by using Microsoft SharePoint Designer won't destabilize his system.

Both because it's intended for information workers rather than developers and because it emphasizes simpler scenarios, Microsoft SharePoint Designer uses a different model for creating workflows than the Visual Studio-hosted WF Workflow Designer. Instead of a graphical approach, Microsoft SharePoint Designer uses a rule-based approach. It's somewhat similar to the Rules Wizard in Microsoft Outlook, a tool that's familiar to many people. The screen below illustrates how a user of Microsoft SharePoint Designer defines a step in a workflow. Notice that this workflow runs some actions in parallel; some actions run serially. Earlier versions of SharePoint Server supported running actions only serially; actions only ran consecutively.

Process Order Workflow



Each step can have a condition and an action. The condition determines whether this step's action should be executed, as in the **If** statement shown above. The choices for actions include things such as assigning an entertainer to an event, collecting approval, and many more. Each of these actions is actually carried out by some SharePoint Server activity, and the activities used here are the same as with Visual Studio and WF Workflow Designer. The list of actions can also include any other activities allowed by the SharePoint administrator for this site, including custom activities created by developers. In SharePoint Server, there is also a special set of activities available that allows users to customize the common approval or collect feedback paradigm of "create a set of tasks and wait for them to complete" in a special designer in Microsoft SharePoint Designer.

Even though its user interface looks quite different from the graphical approach used with Visual Studio and WF Workflow Designer, Microsoft SharePoint Designer creates a standard WF workflow. What's actually produced is a workflow that is sequential, parallel, or combination of both with conditions expressed by using the WF rules engine. Workflows created with this tool do have some limitations, however. For example, they can't be modified while they're running, unlike those built by using Visual Studio and WF Workflow Designer, and only sequential and parallel workflows can be created—state machines aren't supported. Also, workflows built with this tool can be authored against a specific

document library, list, or site when they're designed. Workflow authors can also create a general workflow template that can be later associated with any library, list, or content type. While this does place limits on how a workflow can be used, it also makes deploying the workflow much simpler. In fact, when a user finishes authoring a workflow with Microsoft SharePoint Designer, the tool provides a one-click deployment of the workflow to the target site, which includes activating the workflow. This is significantly less complicated than the multi-step deployment process required for workflows created using Visual Studio and WF Workflow Designer.

Workflows created using Microsoft SharePoint Designer can also display customized forms. Rather than require workflow authors to create .aspx pages directly, however, the tool instead generates those pages. The author specifies details about how the generated pages should look, such as what fields they should contain, and Microsoft SharePoint Designer takes care of the rest. Of the four points in a workflow's lifecycle where forms can be used, however, only two are used with workflows created by using Microsoft SharePoint Designer: Initiation and Task Completion. Because every workflow created with this tool must be associated with a particular document library, list, content type, or site there's no need for an association step and hence no Association form. And since these workflows can't be modified while they're running, there's no need for a Modification form.

Microsoft SharePoint Designer also provides the ability to import workflows that were created using Microsoft Visio 2010. This enables business managers or workflow authors to create the workflow logic using a well known graphical environment. A workflow author can then import the workflow logic into Microsoft SharePoint Designer, modify it if necessary, and then publish it to a SharePoint site.

SharePoint Server provides a great deal of functionality for creating document-oriented workflows. Yet ultimately, it's a platform for development and execution. On its own, it provides no workflow functionality that's directly usable by end users. Workflows running on SharePoint Server also have other restrictions, such as the inability to interact with participants by using Office client applications.

Authoring tool comparison

The following table shows the important differences between the tools that Microsoft supports for creating workflows in SharePoint Server using both SharePoint Designer and WF Workflow Designer in Visual Studio 2010 Professional Edition.

Capability/Requirement	SharePoint Designer	WF Workflow Designer in Visual Studio
Workflows can be created using only actions that are approved by site administrators?	Yes	No
Workflows are accessible in client applications (other than the browser)?	Yes	Yes

Capability/Requirement	SharePoint Designer	WF Workflow Designer in Visual Studio
Can use Microsoft Visio Professional to create workflow logic?	Yes	No
Need to write code?	No	Yes
Additional activities (other than those provided by SharePoint Server) are provided?	No	Yes
Can create custom activities?	No	Yes
InfoPath forms can be used in the workflow?	Yes	Yes
Workflow can be modified while it is running?	No	Yes
One-click publishing of workflows?	Yes	Yes
Workflows can be deployed remotely?	Yes	No
Can be made available across the farm?	No	Yes
Can be scoped to a site collection?	Yes	Yes

Plan for approval and review processes in workflows (SharePoint Server 2010)

As part of planning workflows for your Microsoft SharePoint Server 2010 deployment, you will also want to plan how to use workflows for approval and review processes.

In this article:

- [Workflow approval overview](#)
- [How the Approval workflow works](#)
 - [Example — Manage the document approval process by using a workflow](#)
- [Hybrid review model](#)

Workflow approval overview

The workflow approval process in Microsoft SharePoint Designer 2010 enables Microsoft SharePoint Server 2010 users to customize individual tasks, inside a workflow, that require approval.

For example, occasionally you might have a project, such as the completion of a specific document, where the usual turnaround time must be reduced. You can set the workflow approval process so that if the task is not completed by a stated time, the document is automatically rejected.

The approval process routes the item through one or more people, who are each assigned a task to review the item and decide about what to do with it. For example, if a document is uploaded to a document library, approvers can approve or reject the document, reassign the approval task to another person, or request changes to the document.

How the Approval workflow works

The Approval workflow routes a document or item that is saved to a list or library to a group of people for approval. By default, the Approval workflow is associated with the Document content type and therefore is automatically available in document libraries.

The Approval workflow supports business processes that involve sending a document or item to colleagues or managers for approval. The Approval workflow makes an approval business process more efficient by managing and tracking all of the human tasks in the process and by providing a record of the process after it is completed.

You can start an Approval workflow directly from a document or item in a list or library. To start a workflow, you select the workflow that you want to use, and then you fill out a workflow initiation form that specifies the workflow participants (approvers), a due date, and any relevant task instructions. After a workflow starts, the server assigns tasks to all participants. If e-mail alerts are enabled for the server, the server also sends e-mail alerts to all participants. Participants can click a link in the e-mail task alert to open the document or item to be approved. Participants can approve, reject, or reassign their

approval tasks. They can also request a change to the document or item that is to be approved. Participants have the option of completing their workflow tasks from either directly from certain programs that are part of the 2007 Microsoft Office system and the Microsoft Office 2010 suites, or from the SharePoint Server 2010 Web site. When the workflow is in progress, the workflow owner or the workflow participants can view the Workflow Status page to see which participants have completed their workflow tasks. When the workflow participants complete their workflow tasks, the workflow ends and the workflow owner is automatically notified that the workflow is finished.



Note:

If a user runs a SharePoint Server 2010 workflow task from a 2007 Microsoft Office system program, the workflow does not open in the 2007 Microsoft Office system program. Instead the user is redirected to an InfoPath Forms Services in Microsoft SharePoint Server 2010 browser form on the SharePoint Server 2010 Web site.

When the workflow is in progress, the workflow owner or the workflow participants can view the Workflow Status page to see which participants have completed their workflow tasks. When the workflow participants complete their workflow tasks, the workflow ends and the workflow owner is automatically notified that the workflow is finished.

The default Approval workflow for document libraries is a serial workflow, in which tasks are assigned to participants one at a time. By default, a version of the Approval workflow is also associated with Pages libraries on a publishing site, and the workflow can be used to manage the approval process for the publication of Web pages. Depending on what you want to customize, you can use either Office SharePoint Designer or to SharePoint Server 2010 itself to customize these pre-associated versions of the Approval workflow to meet the needs of your organization, or you can add a new version of the Approval workflow to a list, library, or content type.

Example — Manage the document approval process by using a workflow

If you have ever had to obtain approval from several people for a project plan, a proposal, or some other important document, you probably know that it can be a frustrating experience. First, how do you manage the logistics of getting the document to the appropriate people? Do you send the document to people by e-mail and request responses in e-mail? Do you schedule a meeting? Second, how do you keep track of which people have approved the document, and how do you formally record their approval? Do you save multiple e-mail messages? Do you save meeting notes?

Consider the problems that you could face weeks or months from now, if you could not remember who approved something, or who sent a particular piece of feedback. Unless your organization has developed formal approval processes, you might be in a situation where you could not locate this kind of information when you needed it.

You can avoid issues such as these by using the Approval workflow available on a SharePoint Server 2010 site to route your documents to colleagues for approval. Workflows reduce the time that is required to coordinate common business processes, such as document approval, by managing and tracking the human tasks involved in these processes. Because the Approval workflow assigns tasks,

sends reminders, tracks participation, and creates a record of the whole process, you can concentrate on performing your work instead of tracking down people and dealing with the logistics of the document approval process. If you ever have to document that you received approval for something, you can provide stakeholders with a link to the workflow history for the document, which shows who approved or rejected a document, or who did not finish a workflow task. In this manner, the Approval workflow makes the approval process more transparent.



Important:

The ability to start an Approval workflow from Microsoft Office Word 2007, Microsoft Word 2010, Microsoft Office Excel 2007, Microsoft Excel 2010, Microsoft Office PowerPoint 2007 or Microsoft PowerPoint 2010 is available only in the 2007 Microsoft Office system and Microsoft Office 2010 suites, and in the stand-alone versions of Office Word 2007, Word 2010, Office Excel 2007, Excel 2010, Office PowerPoint 2007 and PowerPoint 2010.

Hybrid review model

The hybrid Approval workflow model is a *staged* approval model. Approvals can be created in sets or stages; that is, the first set of approvers can complete the review and approval process, and then the next set of approvers, and so on. But each stage or approval set can also have its own order of events; that is, members of the first group of approvers can perform their review in serial approval order, members of the second group can perform their review in parallel, and so on.

Plan for workflow security and user management (SharePoint Server 2010)

Before deploying workflows in Microsoft SharePoint Server 2010 to users, administrators might have concerns about security issues, such as information disclosure or elevation of privilege. This article highlights some aspects of workflow behavior that relate to security and raises other issues for administrators and workflow developers to consider when they plan to configure and develop workflows.

In this article:

- [List manager, administrator, and developer roles and responsibilities](#)
- [Running workflows as an administrator](#)
- [Workflow configuration settings](#)
- [Information disclosure in task and workflow history lists](#)
- [Spoofing and tampering attacks in the task and workflow history lists](#)
- [User-Impersonation Step type for declarative workflows](#)

List manager, administrator, and developer roles and responsibilities

The following are some common workflow actions and the related responsibilities, which explain the role of administrators and developers in running workflows.

Workflow developers

Develop workflow schedule and template Workflow developers are responsible for coding the assembly that contains the business logic that will run on a SharePoint item. This assembly is called a workflow schedule. They are also responsible for packaging the workflow forms and assembly into a workflow feature, or into a workflow template.

Site administrators

Manage Central Administration workflow settings Site administrators can control general workflow settings, such as task alert results and external participant settings on the SharePoint Central Administration Web site.

Deploy Workflow features Site administrators can install workflow features on a site collection to make them available for association.

List administrators (anyone with Manage List or Web Designer permissions)

Add workflows List administrators must associate (add) a workflow template to a list or content type, according to the business needs of the list or content type. This association makes the workflow template available to end users, who can then select default values and settings.

Remove workflows List administrators can remove workflow associations from a list or content type, or prevent new instances from running.

Terminate a workflow If a workflow instance fails, list administrators can stop a running workflow instance, such as when a workflow instance produces an error or does not start, by using the

Terminate this workflow link on the Workflow Status page. This action is reserved for administrators.

Running workflows as an administrator

The most important security concept to be aware of is that workflows run as part of the system account in SharePoint Server 2010, through the identity application pool settings on the server computer and domain. This means that within SharePoint Server 2010, workflows have administrator permissions. On the server, workflows have the same permissions as the application pool, which frequently has administrator permissions. These permissions enable workflows to perform actions that ordinary users cannot perform, such as routing a document to a specific location or records center, or adding a user account to the system.

This setting, that workflows have administrator permissions, cannot be changed. It is up to the workflow schedule (that is, the workflow code) to detect user actions and, based on those actions, continue or roll back changes, or impersonate a user in order to mimic that user's permissions.

When they deploy workflows, administrators must understand the actions that the workflow will perform so that they can assess possible risks associated with elevation of permission in a workflow and help the workflow developer reduce any security concerns.

Workflow configuration settings

SharePoint Server 2010 has some configuration settings that administrators have to set according to their security needs.

Required permissions to start a workflow

In addition to preventing the elevation of permissions in the code, list administrators can restrict the permission level that is required to start a workflow during the association process. Administrators can select either of two permission levels to start a specific workflow association: Edit Item or Manage List.

The default setting for associating a workflow is to allow users with Edit Item permissions to manually start a workflow. This means that any authenticated SharePoint Server 2010 user on the list who has Edit Item permissions can start an instance of this workflow association. If during workflow creation the

administrator selects the option to require that the user have Manage Lists permissions in order to start the workflow, only list administrators can start an instance of this association.

Because workflows are designed to be used by standard contributors, most workflows do not require the restriction to Manage Lists permissions. However, administrators can use this setting for workflows such as a document disposal workflow, where the administrator wants only certain people to execute the disposal actions.

Central Administration settings

The following settings can be found on the Central Administration page by clicking **Application Management** and, in the **Web Applications** section, clicking **Manage web applications**. On the Web applications page, select the Web application that you want to configure, and in the **Manage** group of the ribbon click **General Settings**, and then select **Workflow**. The Workflow Settings page opens, and the following settings are displayed:

- User-Defined Workflows
- Workflow Task Notifications

Enable user-defined workflows

By default, user-defined workflows are enabled for all sites on the Web application, as shown in the **User-Defined Workflows** section of the Workflow Settings page. When this option is selected, users can define workflows in a workflow editor such as the SharePoint Designer 2010 workflow editor. Users who define these workflows must have Manage List permissions on the site to which they are deploying the workflow.

Task notification for users without site access

On the Workflow Settings page, in the **Workflow Task Notifications** section, you can set options for sending notifications about pending workflow tasks to users who do not have access to the site.

Internal users

In SharePoint Server 2010 it is possible to resolve the names of internal users in the directory service who are not members of the site or who do not have access to that task. In this case, an administrator can select the **Alert internal users who do not have site access when they are assigned a workflow task** option in the **Workflow Task Notifications** section to set whether such users receive a task notification by e-mail. This option means that users are alerted when they are assigned a workflow task. This option is enabled by default, and the e-mail message that users receive contains a link that they can click to request access to the site (administrators must still grant access). This e-mail message might also contain information about the document. This information can include the title of the document and instructions from the workflow owner. If there are information disclosure concerns associated with internal users who are not members of the site, administrators might want to disable the **Alert internal users who do not have site access when they are assigned a workflow task** setting.

External users

External users who are not in the directory service but who are assigned a well-formed SMTP e-mail address can still be assigned workflow tasks. Because external users will find it difficult to access the document, SharePoint Foundation 2010 and SharePoint Server 2010 includes a setting, **Allow external users to participate in workflow by sending them a copy of the document**, which makes it possible to send external users a task notification by e-mail with the document attached. When this option is enabled, the task is assigned to the workflow owner, and the external user can complete the task by sending e-mail to the owner.

By default, the option **Allow external users to participate in workflow by sending them a copy of the document** is disabled. But this setting can be useful in situations that require external participation, such as approval of business documents that involve external customers.

Administrators who enable this setting (select **Yes**) must verify that the workflow schedule supports the external participant setting. For example, when a task is created for an external user, the custom workflow must specify the external e-mail address in the **OnBehalfEmail** property in the **SPWorkflowTaskProperties** object that was used to initialize the task). Several built-in workflows in SharePoint Server 2010 support this setting.

Custom workflow developers who want to enable this functionality must work with administrators to determine whether there are information disclosure risks in attaching the actual document to an external e-mail message. Administrators must evaluate the benefits and risks when enabling this setting.

Information disclosure in task and workflow history lists

Because tasks and history list items can contain data about users and the actions they perform on documents, the items might disclose confidential information. For example, a promotion Approval workflow might collect feedback on its tasks that an organization wants only the workflow owner and each participant in the task to see.

Task and history lists are typical lists in a site. By default, therefore, all readers can view tasks and history items. Administrators and developers must determine the information that cannot be disclosed and decide whether to help secure task and history items that are created by the workflow.

Securing these items can be done in several ways. For example, administrators can set list-level permissions. If disclosure is to be private — that is, not publicly available but available to a specific group of people — administrators can create a new task or history list and set permissions for the list that are targeted to that group. If administrators do not want anyone to see history events on a workflow status page, they can remove view permissions to the workflow history list from which a status page pulls its information. Users who do not have permissions to view the history list itself, or any item on the list, will receive an Access Denied error when they open any status page that pulls data from that history list.

If finer restrictions are required, workflow developers can set per-item permissions when they create tasks or history items. The CreateTask activity has a **SpecialPermissions** property that gives only specified permissions to access the newly created task. The LogToHistoryList activity does not have such a property, so to set per-item permissions on history list items, administrators must use the object model (OM) in SharePoint Server 2010. Per-item permissions can affect performance negatively and should not be used unless they are necessary.

Tasks and history items do not have to be handled in the same manner. Administrators can mix and match list permissions and item-level permissions.

Spoofing and tampering attacks in the task and workflow history lists

Any contributor can modify tasks or history items if there are no restrictions on those lists. This means that malicious users can modify task descriptions to give participants incorrect instructions or to order participants to click malicious links. To change the perceived results of a process, malicious users also can add false or inaccurate history events or can modify history events to make them false or inaccurate.

As detailed earlier, task and history lists are normal lists in a site. By default, there are no permission restrictions on either task lists or history lists. To avoid spoofing and tampering attacks, administrators must determine the vulnerabilities that exist and either restrict access to columns in a list (for example, make vulnerable columns such as task descriptions read-only so that only the workflow can set them on item creation), set special permissions on the list, or set item-level permissions on the items in a list.

Security issues in the workflow history list

A key benefit of workflows is the ability to track process information to provide visibility into a process. The workflow history list is a repository for this information, where a workflow status page can search for data related to a workflow instance and can make this information available to users. Users can see all items to which they have access in the history list.

However, because the workflow history list tracks information, users might assume that it can be used as an audit trail for events. This is not the case: Workflow history is not a security feature. History lists are standard SharePoint lists that are used for storing events that are visible to any user and that have no special permissions associated with them. By default, users can modify and add events if they have edit and add permissions on the site. To audit events, use the SharePoint's Audit Log feature. Only administrators can access this log and the log does not require additional work to protect it from tampering attacks.

To better protect the history list, administrators can restrict edit and add permissions to the list, so that only system account administrators (for example, workflow administrators) and list administrators can add items. List administrators must have add permissions to log "Terminate this workflow" events. If edit and add permissions are restricted on the history list, users still must be granted view permissions in order to see status information.

User-Impersonation Step type for declarative workflows

The User-Impersonation Step type can be used to run sections of declarative workflows by the person who authored the workflow rather than by the workflow's initiator. Declarative means a model that you use to create the workflow and set the parameters for the workflow without writing any code.

In SharePoint Server 2010, declarative workflows always run in the user context of the workflow initiator unless an impersonation step is encountered. If an impersonation step is encountered, the declarative workflow is run in the context of the workflow associator. The default workflow tasks respect SharePoint permissions by impersonating the user who started a workflow when the workflow is run. This arrangement keeps things fairly safe in SharePoint Server 2010, but blocks many scenarios in which a workflow designer with high permission levels wants to author a powerful workflow that can be completed successfully by users who have lower permission levels.

Through a safe and scoped form of privilege elevation, site actions can be automated through workflow. This reduces the burden on a SharePoint site administrator. Automation of a high-security process is useful in publishing and approval scenarios in which existing actions are enabled to impersonate someone other than the workflow's initiator.

The following are sample scenarios that demonstrate the User-Impersonation Step type:

- **Publish to a secure list**

Jackie has locked down the Pages document library for the public face of her SharePoint site. She has set up an Approval workflow that, by using Microsoft SharePoint Designer 2010, submits content from site contributors for approval. Jackie puts her workflow actions in an impersonation step so that the workflow actions will always impersonate her, a site administrator, as the author of the workflow.

When Connie (a contributor) posts a content draft to the Pages library of the site, and tries to publish her article, that action causes Jackie's Approval workflow to start so that the post can be reviewed and approved. Tasks are sent out to the approvers in the workflow on behalf of Connie. Upon review and approval by the approvers, the system sets the moderation status of the post to "Approved", even though Connie does not have permission to approve pages.

- **Granting permissions to users**

Joanne has set up a workflow in SharePoint Designer 2010 that uses a user-impersonation action "Add User to Group" to grant Design permissions to her site. Because the workflow uses an impersonation scope, the action of adding a user to the group will always be performed on Joanne's behalf.

The rest of the workflow lets contributors visit the site and complete a form to log their access request to a list.

For example, a separate user, Olivier, receives a task when Connie, a user, logs a request, and when he approves the task, Connie is added to the Designer group for the site even though neither Olivier nor Connie has Manage Lists permissions on Joanne's site.

- **Templates and taking ownership**

William created several workflows in SharePoint Designer 2010 and saved them as templates for reuse across the company, but he soon leaves the company. His account is removed, his administrator status is revoked, and now the SharePoint Designer 2010 workflows that William created fail to complete due to the loss of William's permissions.

A parent SharePoint site administrator, John, can intervene for each workflow, without having to re-create the workflows in SharePoint Designer 2010. John takes ownership of the administrative symptoms in each broken template. After doing this, secure publishing and access granting now occur under John's name instead of William's — and nothing else has changed.

The following are workflow actions that can be impersonated:

- Set Content Approval Status (as Owner)
- Create List Item (as Owner)
- Update List Item (as Owner)
- Delete List Item (as Owner)
- Add/Remove/Set/Inherit List Item Permissions (as Owner)

As a SharePoint administrator, you must consider the possible security effects of incorporating impersonation into workflows on the SharePoint site. This applies to new actions but also to existing actions such as updating list items.

For example, consider a model in which user-impersonation actions in the workflow could still run as the initiator. If a user has administrator permissions only over a site in the site collection, that user could maliciously create a workflow to gain rights to the parent Web of the site. All that the malicious user would have to do is to persuade the administrator to upload a file to a document library on the malicious user's site to begin the workflow's attack and compromise the whole parent Web of the site.

This risk prompted development of the restriction "user-impersonation actions always impersonate their associator" in SharePoint Designer 2010. The *associator* is the person who associates a workflow to a particular list or Web. In SharePoint Server 2010 declarative workflows, the associator is the same person as the workflow author; that is, the user who builds the workflow in SharePoint Designer 2010. However, the associator can also be anyone who associates a declarative workflow template. The concern now is that the author/associator is forced to accept responsibility for anything that occurs because of a User-Impersonation Step type, because the author/associator's credentials are being used in the elevation. This requires that the authors/associators understand the workflows they design or associate. Therefore, during workflow creation, SharePoint Designer 2010 provides a cautionary message on the workflow creation page to the author/associator about the User-Impersonation Step type.

Approval Workflow: A Scenario (SharePoint Server 2010)

The most common example of human workflow in most organizations is some variation of approval: A group of people must approve or reject some document, and perhaps add comments to explain their decisions. This article shows how an approval-type workflow that is created in SharePoint Designer 2010 or Workflow Designer in Visual Studio 2010, and that is then hosted by using SharePoint Server 2010 might look. Before reading this example, it is useful to define the roles that different people play.

- **Workflow author** The developer or information worker who creates a workflow template.
- **SharePoint Server 2010 administrator** The person who installs a workflow template and associates it with a document library or list.
- **Workflow initiator** The person who starts a workflow, causing a workflow instance to be created from a particular workflow association.
- **Workflow participants** The people who interact with a workflow instance to complete the business process that it supports.

As described in the following section, people in each of these roles play their own parts in creating, installing, starting, and using a workflow.

Authoring a workflow

Microsoft provides two options for creating workflows in SharePoint Server 2010. Developers use Visual Studio 2010 and Workflow Designer, whereas information workers use the rules-based approach that SharePoint Designer 2010 provides. In both cases, the result is a workflow template that must be deployed to a server that is running SharePoint Server 2010. This scenario assumes that a workflow template has already been created.

Associating a workflow

Before you can use a workflow, you must install a workflow template on a server that is running SharePoint Server 2010, and then you must associate the workflow with a particular document library, list, content type, or (in the case of a site workflow) site. You can then start the workflow from any document or item in that library or list. Although workflows cannot be explicitly started from content types, a workflow that is associated with a content type can be started from a document or list item to which that content type is attached. Because workflows operate in the same manner on items and documents, a workflow template can typically be attached to a list, library, or content type. You can also create a template that can be associated only with a particular list or library.

Both installation and association are performed automatically for workflows that are deployed by using SharePoint Designer 2010. However, when you use Visual Studio to deploy workflows, a server

administrator must explicitly install the workflow template. In addition, a user must associate the template with a library, list, content type, or site. Whoever creates this association also assigns the association a unique name, which enables users to reference it. Optionally, the workflow author can let the person who creates the association set options for the workflow behavior, such as a default list of people who can always participate in the process. The same template can be associated with multiple libraries, lists, or content types, and each association can be customized as required. After the association is created and any available options are set, a workflow initiator can create a workflow instance from this association, as described in the following section.

Associating a workflow with a site

Site workflows are associated with the site itself. An item does not have to be started for the workflow to run.

You can use site workflows for processes that do not have a list item context. For example, you could create a workflow to request permissions for the site, a workflow to request and provision a new site, or a workflow that uses context that is stored outside the SharePoint site, without having to create a corresponding SharePoint list item from which to start the workflow.

Site workflows can be associated with a site through the site's settings and can be started on the site itself. SharePoint Designer 2010 can also deploy site workflows directly to a site.

Site workflows work in the same way as list items, as described earlier in this article, except that site workflows cannot be started from a document or item in a library or list.

For more information, see [Add a workflow association \(SharePoint Server 2010\)](#)

Starting a workflow

SharePoint Server 2010 provide three options to start an instance of a workflow. All three options run the workflow from the beginning every time. (If an instance of a workflow that is created from a particular association is already running on a particular document or list item, it is not possible to start another instance of the workflow on the same document or item.) The following are the options for starting a workflow:

- A SharePoint Server 2010 user can manually start a workflow.
- You can configure a workflow to run automatically when a user creates a document or item.
- You can configure a workflow to run automatically when a user changes a document or item.

For example, a Microsoft Word user can upload a new document to a site's document library. This causes an instance of a workflow that is associated with that library to start.

This scenario uses the first of these three options: manually starting an Approval workflow for a document. To start a workflow instance from a document in a document library, a SharePoint Server 2010 user does the following:

1. Points to the document and selects **Workflows** from the drop-down menu or ribbon.
2. Selects the workflow to start.

For example, for a document in a document library, only two choices are ordinarily available — **Approval** and **Collect Feedback**. If an administrator has associated other workflow templates with this document library, the names of those workflow templates also appear.



Note:

The predefined Approval and Collect Feedback workflows are available only in SharePoint Server 2010.

When a workflow is started (that is, when an instance of a workflow is created), it can also display a screen that enables a user to specify relevant information. For an **Approval** workflow, this information includes the name of each person who must approve the document, an indication of when each approval is due, and a list of people to be notified. After this information is supplied, the user clicks **Start**. The workflow begins to execute and requests each participant to review the document in the order in which names were entered on this screen.

When a workflow is started, it can also optionally send an e-mail message to the person who started it. Similarly, a workflow can inform its creator by e-mail when it has finished. You can also configure the workflow to notify the participants in the workflow — in this example, the people who are approving the document—by e-mail that the workflow has something for them to do.

Interacting with a workflow

The concept of tasks models the interaction between a person and a running workflow. A task is a unit of work that is assigned to an individual. For example, each person on this workflow's approval list will be assigned a task that requests approval of the document. SharePoint Server 2010 can have a task list for every site, and a running workflow can add tasks to this list that specify the person or persons who are assigned to each task. Users of that site can see the work that is awaiting them by accessing their task list through a Web browser. Optionally, you can have a custom task list for just your workflow tasks.



Note:

If you are running SharePoint Server 2010, users can synchronize the site's task list to their Microsoft Outlook task list.

To a SharePoint Server 2010 user, the list of waiting tasks is merely another list. In this example, the user browses to the team SharePoint site and selects the option to view the list of **Tasks** that are assigned to him. To work on a task, the user in this example clicks the task name.

Because the way that a workflow interacts with participants can vary, the workflow itself defines the screen that is displayed to the user. In this example, the workflow provides options to approve or reject the document and a text box in which participants can type comments.

Other available options let users reassign the task to another person or to request a change. Here, the user might type a comment, and then click **Approve**. The workflow then creates a task in the task list of the next person in its list of approvers. When every participant has responded, the workflow ends.

SharePoint Server 2010 workflows also provide other options, including the following:

- The initiator of a workflow can check the status of the workflow.

For example, in the scenario described here, the initiator might check the progress of the approval process.

- A workflow can be modified while it is executing.

The workflow's author determines the allowed modifications, if any. An Approval workflow, for example, could allow the addition of a new approver while the workflow is in progress. The ability to modify in-progress workflows is important because it reflects how people actually work. Because spontaneous change to business processes is a part of life within any business, SharePoint Server 2010 workflows were designed to let users handle this.

Summarizing the process

When a workflow template is installed on a site and associated with a document library, list, site, or content type, a site user can start an instance of a workflow.

1. The process starts when the workflow initiator selects a document and starts an instance of a workflow.
2. The initiator creates a workflow instance from this association.
3. The user customizes this new instance and starts it.
4. The running instance of the workflow adds a task to the task list of a participant.

The approval workflow that is used in this scenario assigns these tasks sequentially. However, you can assign tasks to many participants at the same time, which allows tasks to be performed in parallel.

5. Participants in the workflow can learn about tasks that the workflow has assigned to them by checking their task lists.
6. Each participant interacts with the running instance of the workflow to complete assigned tasks.

In the example described here, this interaction required approving a document, but the interaction could be anything that the workflow author wants.

It is worth noting that the document on which a workflow runs is not itself sent from person to person. Instead, the document remains on the site, and each workflow participant is given a link to it. In fact, there is no requirement that the workflow use the document or item with which it is associated. Another point worth emphasizing is that SharePoint Server 2010 itself defines what is displayed to the initiator of the workflow and the participants in the workflow in steps 1, 2, and 5. However, the workflow author defines the forms that are used in step 6. This allows the author to control how users customize and interact with the workflow.

Together with a platform for creating human workflow applications, SharePoint Server 2010 provides a predefined Issue Tracking workflow. This Three-state workflow allows for assigning active issues to participants and tracking those issues. Once created, an issue can be moved from an Active state to a Resolved state, which indicates that the responsible workflow participant has handled it, and then to a

Closed state, which indicates that the workflow initiator has accepted the resolution and closed the issue.

The following section describes a scenario using the predefined Approval workflow in SharePoint Server 2010.

Approval workflow scenario

The workflow described in this section is the predefined Approval workflow provided with SharePoint Server 2010. All user interaction with the workflow happens by using InfoPath Forms Services workflow forms that are displayed in applications in the Microsoft Office system.

1. The process starts when the workflow is associated with a document library or list. The workflow initiator creates a running workflow instance. This can be done from SharePoint Server 2010 or can be done directly from a Microsoft Word document.
2. When a user clicks the **Start** link for the **Approval** workflow, the workflow begins and opens a InfoPath Forms Services workflow form. The predefined Approval workflow enables its initiator to customize its behavior by specifying a list of approvers, setting how long each one has to perform his task, and so on.
3. The people who are listed as approvers in this workflow are each sent an e-mail message in the order in which their names were entered.
4. The approver can examine the document by clicking the document name link in the body of the e-mail message.

In this scenario, the content of the workflow's Task Completion form is defined as an InfoPath Forms Services workflow form and is displayed directly in Microsoft Outlook. The approver can add comments, and then approve or reject the document.

Access Services planning

In this section:

- [Introduction to Access Services \(SharePoint Server 2010\)](#)

Access Services is a service application available in Microsoft SharePoint Server 2010 that allows users to edit, update, and create linked Microsoft Access 2010 databases that can be viewed and manipulated by using an internet browser, the Access client, or a linked HTML page.

- [Improving the reach and manageability of Access 2010 database applications with Access Services \(white paper\)](#)

With Access Services in Microsoft SharePoint Server 2010, the IT department can centrally manage and govern Access Web databases. Together, Microsoft Access 2010, Access Services, and SharePoint 2010 Products empower users to easily track, report, and share custom database applications and for themselves, their team, and their organizations.

Introduction to Access Services (SharePoint Server 2010)

Access Services is a service application available in Microsoft SharePoint Server 2010 that allows users to edit, update, and create linked Microsoft Access 2010 databases that can be viewed and manipulated by using an internet browser, the Access client, or a linked HTML page.

Who should use Access Services?

IT professionals and end users should use Access Services to allow the use of Access databases inside a Web browser, to publish and share information across teams, and to create and modify databases where no Access client is available.

Features of Access Services

Access Services allows you to create, edit, and save Access databases in the following ways:

- By allowing instantiation of a Microsoft SharePoint Server database on any computer that can connect to and has permission to use Access Services on a networked computer running SharePoint Server.
- By allowing the creation, publishing, and sharing of a SharePoint Server Web database from any computer that can connect to and has permission to publish to a computer that is running SharePoint Server and that has Access 2010 installed.
- By allowing the download, modification, and republishing of modified data in an Access Services database from any computer that has Access 2010 installed and can connect to a computer running SharePoint Server.

Improving the reach and manageability of Access 2010 database applications with Access Services (white paper)

Organizations have to provide users the tools that they need to work effectively and efficiently. For many years, Microsoft Access has helped organizations and users manage their business information. It enables them to easily create and deploy database applications that address a specific business need, task, or process. With the addition of Access Services in Microsoft SharePoint Server 2010, IT can centrally manage and govern Access Web databases. Together, Microsoft Access 2010, Access Services, and SharePoint 2010 Products empower users to easily track, report, and share custom database applications and for themselves, their team, and their organizations. All this is achieved without straining limited IT resources, and while still allowing IT to centrally manage and govern the Access Web databases.

Download this white paper as a Microsoft Word document (.docx).

[Improving the Reach and Manageability of Access 2010 Database Applications with Access Services](#)
(1.92 MB)

Plan site creation and maintenance (SharePoint Server 2010)

If you plan on having more than a few site collections in your Microsoft SharePoint Server 2010 environment, you need to be sure that you have a plan for site creation and maintenance. Without such a plan, it is difficult to control or track when SharePoint sites are created, whether sites are still active, and when you can safely remove inactive sites. Before you deploy and make sites available to users, you need to answer questions such as:

- Do you want to tightly control site creation or to allow many users to create sites?
- At which level in the site hierarchy should additional sites be created?
- How do you find and remove unused sites in your environment?

Articles and worksheets help you design and record a plan for site creation and maintenance. This will help you prepare to manage growth in your environment.

In this section:

- [Plan process for creating sites \(SharePoint Server 2010\)](#)
Discusses how to determine which type of site creation process will fit your organization, and which method to use to implement that process.
- [Plan site maintenance and management \(SharePoint Server 2010\)](#)
Discusses how to plan for maintaining your SharePoint sites from the beginning to make sure that your sites stay current, useful, and usable.
- [Plan quota management \(SharePoint Server 2010\)](#)
Contains guidance about how to determine settings for quota templates and recycle bins, and how to decide whether or when to delete unused Web sites.
- [Reporting and usage analysis overview](#)
This article explains how the Web Analytics service and its reporting features can be used to manage sites and site collections.

Plan process for creating sites (SharePoint Server 2010)

Some organizations need to maintain tight control over who can create sites, or when sites are created. Other organizations can allow users more access and freedom to create sites when needed. This article helps you determine which type of site creation process will fit your organization, and which method to use to implement that process.

In this article:

- [Determine who can create sites and a method for site creation](#)
- [Plan for Self-Service Site Management](#)
- [Plan for custom site creation processes](#)
- [Worksheet](#)

Determine who can create sites and a method for site creation

By default, new site collections (and therefore new top-level Web sites) can only be created by using Central Administration, which means that they can only be created by members of the Farm Administrators group. This behavior might suit your organization if you want your environment to be tightly controlled and managed, with only a few people allowed to add top-level sites. However, the default top-level site creation method might not suit your organization if you have any of the following requirements:

- You want users to be able to easily create informal, perhaps even disposable, top-level sites, such as for short-term projects.
- You want to create an informal space for team, group, or community interaction.
- You are hosting top-level sites (either internally or externally) and want the process for requesting and receiving a top-level site to be as quick and low cost as possible.

There are several ways to allow users to create their own sites, while still maintaining some control over your environment. Consider which of the following methods will work best for your organization.

- **Self-Service Site Management** In Central Administration, you can turn on Self-Service Site Management to allow users to create site collections under the /sites path (or other path you specify) within a particular Web application. This method is best used when you want to allow groups or communities to create sites. This method also works well if you are hosting sites and want to allow users to create sites without waiting for a complicated process. The sign-up page for Self-Service Site Management can be customized or replaced with a page that includes all of the information you might need to integrate with a billing system or to track custom metadata about the site at creation time. This method does not work well when large numbers of users need access to

multiple sites. Because Self-Service Site Management creates site collections, which have separate permissions, users need to be added uniquely to different site collections. If you use subsites instead, the users can be inherited from the parent site in the site collection.

- **Subsites of existing sites** Limit users to creating subsites of existing sites, rather than new site collections and top-level sites. Any user who has the Full Control or Manage Hierarchy permission level on an existing site can create a subsite. This method is the most limited, because you still control how many site collections there are. Because the sites are always subsites of other sites, they can either be easy to organize (if there are just a few) or very difficult to organize and browse (for example, if everyone in your organization wants a subsite and they create them at different levels in the site collection's hierarchy, the site collection can soon become very difficult to navigate).



Note:

If you do not want users to have this capability, you can remove the Create Subsites right from the Full Control and Manage Hierarchy permission levels, either at the site collection or Web application level.

- **Personal sites** Allow users to create personal sites (also known as My Site Web sites). Personal sites are site collections stored under the /personal path of the Web application. Personal sites are created for individual users, so they are not the appropriate method to use if you are trying to create sites for groups or communities. For more information about personal sites, see [Plan for My Site Web sites \(SharePoint Server 2010\)](#).



Note:

Keep in mind that none of these methods can control how much space each site takes up in your content databases. To control site sizes, you should use quotas and set a size limit for site collections. You cannot set individual size limits for subsites. For more information, see [Plan site maintenance and management \(SharePoint Server 2010\)](#).

Plan for Self-Service Site Management

Self-Service Site Management allows users to create and manage their own top-level Web sites automatically. When you turn on Self-Service Site Management for a Web application, users can create their own top-level Web sites under a specific path (by default, the /sites path). When turned on, this capability advertises itself with an announcement added to the top-level site at the root path of the Web application, so any users who have permission to view that announcement can follow the link.



Note:

If you want to use a path other than /sites for Self-Service Site Management, you must add the path as a wildcard inclusion. For more information, see [Collaboration site planning \(SharePoint Server 2010\)](#).

This capability can obviously affect the security for your Web server. Self-Service Site Management is disabled by default — you must turn on the feature to use it. You enable Self-Service Site Management

for a single Web application at a time. If you want to use it on all Web applications in your server farm, you must enable it for every Web application individually.

If you enable Self-Service Site Management, you should consider the following:

- Generally, you should require a secondary site collection administrator. Administrative alerts, such as those for when quotas are exceeded, or checking for unused Web sites, go to the primary and secondary administrators. Having more than one contact reduces administrator involvement with these sites because the secondary contact can perform required tasks even if the primary contact is not available.
- Define a storage quota and set it as the default quota for the Web application.
- Review the number of sites allowed per content database. Combined with quotas, this will help you limit the size of the databases in your system.
- Enable unused Web site notifications, so that sites that are forgotten or no longer of value can be identified.

Because Self-Service Site Management creates new top-level Web sites on an existing Web application, any new sites automatically conform to the Web application's default quota settings, unused Web site notification settings, and other administrative policies.

Plan for custom site creation processes

You can, of course, create your own process for site creation by using a custom form to request a site that integrates with a back-end billing system to charge a customer's credit card or a corporate cost center. If you have a complicated system or process that you want to include as part of site creation, you should create a custom application to call the site creation interface and perform any other tasks you require. However, if you simply want to add a few custom fields to the site creation page (for example, to track which department in your company is requesting a particular site), you should consider using Self-Service Site Management and customize the sign-up page to include the information that you need. You can customize the `scsignup.aspx` page in the site definition to include the metadata that you need without having to develop an entire application.

For more information about building custom applications or editing pages in a site definition, see the [SharePoint 2010 developer portal on MSDN](http://go.microsoft.com/fwlink/?LinkId=178818) (<http://go.microsoft.com/fwlink/?LinkId=178818>).

Worksheet

Use the following worksheet to plan the process for creating sites:

- [Site Creation and Maintenance Worksheet](http://go.microsoft.com/fwlink/?LinkId=193521) (<http://go.microsoft.com/fwlink/?LinkId=193521>)

Plan site maintenance and management (SharePoint Server 2010)

All Web sites, particularly sites that have more than one author, get cluttered. Periodic review and cleanup can help keep your site functioning well, whether your site is large or small. If you build a plan for maintaining your site or sites from the beginning, you can ensure that they stay current, useful, and usable.

In this article:

- [Plan for site maintenance](#)
- [Plan for managing site collections](#)
- [Worksheet](#)

Plan for site maintenance

Your site maintenance plan will be different from that for any other environment, and it will contain different elements. Site maintenance is different for sites managed by an IT department than it is for user-created sites and managed sites. However, some best practices for a site maintenance plan include:

- Ask users what they want in IT-managed sites. Perform periodic surveys to determine what your users need from the site.
- Use usage logs and reports to find out which areas of the site are being used, and then correlate that with user surveys to find out what can be improved.
- Archive obsolete content or sites. However, if you are going to archive or delete obsolete content or sites, be sure that users understand that plan and that you perform these actions only at predictable times. For example, publish a schedule of when you are going to archive content or delete unused sites.
- Periodically review site permissions. For example, review the permissions quarterly to remove permissions for any users who have left the group or project.
- Select a reasonable time interval for your maintenance activities. For example, if you plan to conduct periodic user surveys, do not conduct them more than twice a year (and preferably, no more than once a year).
- Create a plan for regular backups of site content. Determine or discover how often backups will be made, and the process for restoring content when necessary. For more information about planning for backup and restore, see [Plan for backup and recovery \(SharePoint Server 2010\)](#).

Start now, during your planning process, to create a plan for site maintenance. Record your plan, including how often to tune up the site and archive content. Get your plan reviewed by members of your team and representatives of your user base. This way, you can identify any concerns that users might

have now, determine how best to address these concerns, and have a plan for site maintenance in place by the time your site goes live.

You can record this information in the [Site Creation and Maintenance Worksheet](http://go.microsoft.com/fwlink/?LinkId=193521) (<http://go.microsoft.com/fwlink/?LinkId=193521>).

Plan for managing site collections

One part of your site maintenance plan should be a plan for how to manage the size and number of site collections in your environment. This is most important if you are allowing Self-Service Site Management. Most organizations want to be able to predict and control how much growth they can expect from sites because of the impact that they can have on database resources. For example, if a particular content database contains 100 sites, and one of those sites is taking up more than 50 percent of the space, then that site might need to be in its own content database. This will ensure that you preserve some room for additional growth, while maintaining the ability to back up and restore the databases.

Two methods for managing site collections are:

- **Site collection quotas** Use this method to control how large site collections can become.
- **Site use confirmation and deletion** Use this method to monitor and remove unused site collections.

Plan site collection quotas

Use quotas to track and limit site storage. You can send a warning e-mail message to site collection administrators when site storage reaches a particular size (in megabytes), and then lock the site to further content when site storage reaches a maximum size. When you perform your database and server capacity planning, determine what size limits (if any) you want to enforce. The following list describes how to take the best advantage of quotas:

- Create different quota templates for different site types. For example, you might want different quotas for different divisions, or for different customer types, or for different paths (perhaps sites under the /sites path only get 100 MB per site collection, whereas sites under the /vip path can take up to 300 MB per site collection). Whenever you create a site collection from Central Administration, you can specify on which quota template it is based. Note that sites created by using Self-Service Site Management use the default quota for the Web application. For more information, see [Create, edit, and delete quota templates \(SharePoint Server 2010\)](#).
- Give enough room for reasonable growth in sites. Depending on what each site is used for, storage space needs can vary dramatically. Sites are designed to grow over time as they are used. A quota limit of 50 MB is unlikely to be enough storage space to start with for most sites, and is unlikely to be anywhere near enough for a site that has a long life.
- Allow for reasonable notice between the warning e-mail message and locking the site for exceeding its quota. For example, do not set the warning limit to 80 MB and the site storage limit to 85 MB. If

users are in the middle of uploading several large files, they will not be happy if blocked from completing that task with very little notice.

Plan site use confirmation and deletion

You need to plan how to handle sites that become inactive after a project has ended, or sites that users created just to test out some ideas, and then abandoned. Site use confirmation and deletion can help you keep your environment cleaner, by helping you identify when sites are no longer needed. This feature works by automatically sending an e-mail message to site owners to see if they consider their site active. If the owner does not respond to the e-mail message (after a specified number of messages over a specified length of time), the site can be deleted.

To plan for site use confirmation and deletion, decide the following:

- How long you want to wait before checking to see if a site is inactive. The default length of time for team or project sites is 90 days after site creation, but you should probably give owners longer than that. For a test or personal site, 90 days is probably too long. Usually a site that was created, was actively used, and is now ready to be deleted or archived, took at least six months and probably a few years to complete that life cycle. Reminders every six months are valuable for those situations.
- How often you want to send an e-mail message to site owners to see if their sites are inactive. After the first e-mail message, if the site administrator does not respond, you can continue with additional notices at daily, weekly, or monthly intervals.
- Whether you want to automatically delete unused sites. If the site administrator does not respond to multiple e-mail messages, do you want to go ahead and delete the site automatically? We recommend that you make a backup first. You can do so by making sure that regular backups are performed. You can use the [SharePoint 2010 developer portal on MSDN](http://go.microsoft.com/fwlink/?LinkId=178818) (<http://go.microsoft.com/fwlink/?LinkId=178818>) to customize this functionality so that it automatically makes a backup of the site before deletion, but this is not default behavior.
- If you are going to automatically delete unused sites, how many e-mail messages will you send to site owners before you do so? By default, four weekly notices are sent before site deletion, but you can increase or decrease this number to suit your needs.

For more information, see [Manage unused Web sites \(SharePoint Server 2010\)](#).

Worksheet

Use the following worksheet to plan for site maintenance and management:

- [Site Creation and Maintenance Worksheet](http://go.microsoft.com/fwlink/?LinkId=193521) (<http://go.microsoft.com/fwlink/?LinkId=193521>)

Plan quota management (SharePoint Server 2010)

A *quota* specifies storage limit values for the maximum amount of data that can be stored in a site collection. Quotas also specify the storage size that, when reached, triggers an e-mail alert to the site collection administrator. *Quota templates* apply these settings to any site collection in a SharePoint farm.

By default, a quota contains 300 points. A *point* is a relative measurement of resource usage, for example, CPU cycles, memory, or page faults. Points enable comparisons between measurements of resource usage that could not be compared otherwise. For example, it takes millions of CPU cycles to make up one point, but each time a sandboxed solution stops working is counted as one point. For more information about sandboxed solutions, see [Sandboxed solutions overview \(SharePoint Server 2010\)](#).

Quotas are particularly useful when you are using Microsoft SharePoint Server 2010 in enterprise environments, such as a company-wide intranet or an Internet Service Provider (ISP). You should use quotas in these environments to ensure that one site collection cannot use so many resources that other site collections can no longer function. You can assign a quota template to a site collection when you create the site collection, or you can assign a quota template at a later time. You can also reverse a decision to use quotas at any place in the site collection hierarchy.

You should also consider quotas when you plan your taxonomy and information architecture. For information about how to plan managed metadata services, such as planning where to store enterprise keywords, see [Plan to share terminology and content types \(SharePoint Server 2010\)](#).

In this article:

- [About planning quota management](#)
- [Determine quota template settings](#)
- [Determine recycle bin settings](#)
- [Delete unused Web sites](#)

About planning quota management

The basic steps to plan quota management are the following:

1. Determine quota template settings
2. Determine recycle bin settings
3. Delete unused Web sites

This article contains guidance about how to determine the quota settings for site collections in an enterprise. This article does not include prerequisite information such as how to configure outgoing e-mail, start the Disk Quota Warning timer job, or plan performance and capacity.

Determine quota template settings

There is no default quota template for site collections in a SharePoint Server 2010 environment. For example, a quota for a site collection might use the following settings as a starting point:

1. Automated e-mail is sent to a site collection administrator when the size of the site reaches 450 megabytes (MB).
2. Users are prevented from uploading additional documents when the size of a site collection reaches 500 MB.

You must evaluate the size and number of items that you expect users to store in their sites. You must also adjust these settings appropriately to ensure that the sites are used in accordance with an organization's best practices. For example, if a specific team or group in an organization has a business need to store a greater volume of content on its team site, you can adjust the quota limits for that site collection.

The size of the data reported by quotas does not necessarily match the size of the storage in the database. This is because the quota feature estimates storage figures for empty sites (that is, sites that contain no user content) and includes those figures in the quota, in addition to the actual storage from the database. The estimated size of an empty site includes the real size of the template pages for SharePoint Server 2010, for example, the forms pages and the pages in the `_layouts` directory.

If you change the values for a quota template, those changes apply only to new site collections to which you apply the template. SharePoint Server 2010 does not apply the changed quota values to existing sites collections unless you use the object model to update the quota values in the database.

Determine recycle bin settings

The recycle bin can help to prevent the permanent deletion of content. The recycle bin enables site owners to retrieve items that users have deleted, without requiring administrator intervention such as restoring files from backup tapes. Key planning considerations include whether to use the second-stage recycle bin and how much space to allocate.

The recycle bin is turned on and off at the Web application level. By default, the recycle bin is turned on in all the site collections in a Web application.

The recycle bin has two stages. When a user deletes an item, the item is automatically sent to the first-stage recycle bin. By default, when an item is deleted from the first-stage recycle bin, the item is sent to the second-stage recycle bin. The second-stage recycle bin stores items that users have deleted from their recycle bins. Only site collection administrators can restore items from the second-stage recycle bin. The size that is specified for the second-stage recycle bin increases the total size of the site. You must plan data capacity accordingly.

Consider allocating at least a small amount of space, for example, 10 percent, to the second-stage recycle bin to accommodate cases in which a user mistakenly deletes an important document, a folder in a document library, or a column in a list.

Items in both the first-stage and the second-stage recycle bins are automatically deleted when the time period specified for the deleted items expires (by default, 30 days). However, when the size limit of the second-stage recycle bin is reached, items are automatically deleted starting with the oldest items. Site collection administrators can also empty the second-stage recycle bin manually. For more information, see [Configure Recycle Bin settings \(SharePoint Server 2010\)](#).

Delete unused Web sites

You can delete a quota template if you change your quota structures. However deleting a quota template does not delete quota values from site collections to which a quota template has been applied. If you want to remove quotas from all site collections that use a specific quota template, you must use the object model or perform a SQL Server query.

Automatic deletion of unused Web sites can help you lessen the risk of deleting data that is critical to business operations. You should include the following tasks in your planning process:

- Require a secondary contact for all sites. If the site owner is not available or leaves the organization, the secondary contact can confirm the usage of the site. If you do not have a secondary contact and you shorten the number of days or number of notices that are given before you delete an unused site, you might accidentally delete a site that is still required.
- Archive sites before they are deleted automatically. You will be able to restore the sites that contain business-critical information or plan to store the content databases for a longer duration, so that a deleted site can be restored.

For more information, see [Manage unused Web sites \(SharePoint Server 2010\)](#).

Reporting and usage analysis overview

This article describes the Web Analytics service application in Microsoft SharePoint Server 2010 and its features.

This article does not discuss capacity for the Web Analytics service application. For information about capacity, see "Capacity Requirements for Web Analytics Shared Service in SharePoint Server 2010", which can be accessed from [Performance and capacity test results and recommendations \(SharePoint Server 2010\)](#).

In this article:

- [Overview](#)
- [Reporting](#)
- [Web Analytics workflow](#)
- [Web Analytics Web Part](#)

Overview

In Microsoft SharePoint Server 2010, the Web Analytics service application enables you to collect, report, and analyze the usage and effectiveness of your SharePoint Server 2010 sites.

The Web Analytics features are reporting, Web Analytics workflow, and Web Analytics Web Part.

Before any of the Web Analytics features can be used, you must configure the services that support the Web Analytics service application. Next, you must create the View Web Analytics Data permission group and assign this permission to users. For more information, see [Configure Web Analytics service application](#).

Reporting

The reporting feature lets an administrator create reports that can be used to understand what users are doing and how they are using a site.

There are three areas of Web Analytics reports available: Traffic, Search, and Inventory.

A list of different reports is aggregated at a site, site collection, Web application, and/or Search service application level for each farm.

Typically, the metrics for each report are displayed one of the following ways:

- A *trend* report shows how a particular metric is doing over a period of time.
- A *rank* report shows the top 2000 results for a particular metric.
- A *list* report shows search-related results for the Best Bet reports.

By default, all reports show data for a period of 30 days. For additional information about how to view and customize a report, see [View Web Analytics reports \(SharePoint Server 2010\)](#).

Traffic reports

The traffic reports capture user behavior information related to total clicks, frequent users, popular pages, and information about navigation to and from the current SharePoint component, for example, a site collection or Web application.

The following table describes the available traffic reports.

Name	Type	Description	Available at Web application level	Available at site collection level	Available at site level	Available at Search service application level
Number of Page Views	Trend	Displays the number of page views per day during the specified date range. Every time a user visit generates a server page request, a page view is recorded.	X	X	X	
Number of Daily Unique Visitors	Trend	Displays the number of unique visitors during the specified date range. Use this report to determine the number of unique visitors on a given day, understand visitor traffic patterns, or identify periods of high or low traffic.	X	X	X	

Name	Type	Description	Available at Web application level	Available at site collection level	Available at site level	Available at Search service application level
Number of Referrers	Trend	Displays the amount of traffic from Web pages that refer traffic to your site. Use this report to identify traffic patterns related to referrers.	X	X	X	
Top Pages	Rank	Displays the most popular pages based on page views during the specified date range. Use this information to target individual pages for improvement or to measure the effectiveness of specific content.	X	X	X	
Top Visitors	Rank	Displays the top unique visitors to your Web site during the specified date range. Use this report to understand who uses your Web site the	X	X	X	

Name	Type	Description	Available at Web application level	Available at site collection level	Available at site level	Available at Search service application level
		most.				
Top Referrers	Rank	Displays the top Web pages that refer traffic to your site. Use this report to understand which pages are directing traffic to your site and how visitors arrive at your site.	X	X	X	
Top Destinations	Rank	Displays the external Web pages your visitors most frequently go to when they leave pages on your site. Use this report to understand your visitors' traffic patterns or to analyze the effectiveness of your referrals to other pages.	X	X	X	
Top Browsers	Rank	Displays the Internet browser types and versions most commonly	X	X	X	

Name	Type	Description	Available at Web application level	Available at site collection level	Available at site level	Available at Search service application level
		used by visitors. Use this report to understand the technologies that your visitors use and to configure your site for optimal viewing.				

Search reports

The search reports capture user behavior information related to the queries on a site that is using a configured search service application.

The Best Bets Suggestion report helps site collection administrators determine what the most relevant search result is for a given keyword. In Microsoft SharePoint Server 2010, suggestions are periodically sent out for new Best Bets using all the search metrics that have been collected. Site collection administrators can look through the Best Bets suggestions and easily accept or reject them.

The Best Bets Suggestion, Best Bet Usage, Best Bet Action History reports are available on the Site Collection Web Analytics Reports page.

Typically, Best Bet reports are displayed in list format.

If you want to retrieve basic information about search queries within your farm, use either the trend or rank search reports. If you want to get a more in-depth view of how the site is being used, use the Best Bet reports.

The following table describes the available search reports.

Name	Type	Description	Available at Web application level	Available at site collection level	Available at site level	Available at Search service application level
Number of Queries	Trend	Displays the number of search queries performed during the specified date range. Use this report to identify search query volume trends and to determine times of high and low search activity.		X	X	X
Top Queries	Rank	Displays the most popular search queries performed during the specified date range. Use this report to understand what types of information visitors are seeking.		X		X
Failed Queries	Rank	Displays search queries that returned no search results or results that received zero or low click-through rates. Use this		X		

Name	Type	Description	Available at Web application level	Available at site collection level	Available at site level	Available at Search service application level
		report to identify search queries that might create user dissatisfaction and to improve the discoverability of content.				
No Result Queries	Rank	Displays search queries that returned no search results. Use this report to identify search queries that might create user dissatisfaction and to improve the discoverability of content.				X
Best Bet Suggestion	List	Recommends URLs that are the most likely results for particular search queries, based on analysis of usage patterns. The site administrators can accept or reject these		X		

Name	Type	Description	Available at Web application level	Available at site collection level	Available at site level	Available at Search service application level
		suggestions. If they accept, the corresponding query-URL pair is added to the search keywords list.				
Best Bet Usage	List	Displays how Best Bet suggestions are doing over time. For every Best Bet query issued, the report shows the percentage of clicks on the Best Bet URL compared to other URLs.		X		
Best Bet Action History	List	Displays the actions that were performed by the site administrator on the Best Bet Suggestion Report.		X		

Inventory reports

The inventory reports are targeted to help site administrators manage a site by keeping track of the site structure and storage and version issues.

The following table describes the available inventory reports.

Name	Type	Description	Available at Web application level	Available at site collection level	Available at site level	Available at Search service application level
Number of Site Collections	Trend	Displays a daily snapshot of the number of site collections in a Web application during the specified date range.	X			
Storage Usage	Trend	Displays a daily snapshot of storage usage for a site collection during the specified date range. Use this report to understand the storage growth pattern for a site collection.		X		
Number of Sites	Trend	Displays a daily snapshot of the number of sites in a site collection during the specified date range.		X	X	
Top Site Product	Rank	Displays the top site		X	X	

Name	Type	Description	Available at Web application level	Available at site collection level	Available at site level	Available at Search service application level
Versions		product versions based on the product version property of each site in a site collection. Use this report to understand the site product version distribution in your site collection.				
Top Site Languages	Rank	Displays the top site languages based on the number of sites created using each language. Use this report to understand the site language distribution in your site collection.		X	X	

Web Analytics workflow

The Web Analytics workflow enables you to have reports sent out on a schedule or when certain criterion is met. For example, you can have a workflow that sends you e-mail every time the total number of pages views drop by 80 percent week over week.



Note:

To create a Web Analytics workflow, you must be a member of the Farm Administrators group.

Web Analytics Web Part

Site managers can add the Web Analytics Web Part to any page on a site. For example, the Most Viewed Content report can be configured on the site to display data. The data in the Web Part is continuously refreshed as new content or new search queries become more popular. For information about how to configure the Web Analytics Web Part, see [Configure Web Analytics service application](#)

Plan e-mail integration (SharePoint Server 2010)

Enabling communication is a critical component for creating Web applications in which group members can interact with each other and keep up with changes to information through the use of alerts. The site collection features that are dependent on communications being properly set up include:

- Alerts that notify group members when things have changed.
- Administrative messages related to requests for site access and other site administration issues.
- Discussion groups.

To make the most effective use of the communications features, planning should include understanding the software requirements and maintenance considerations.

Plan communication by using the following articles:

- [Plan incoming e-mail \(SharePoint Server 2010\)](#), which provides information on how to set up e-mail for discussion groups.
- [Plan outgoing e-mail \(SharePoint Server 2010\)](#), which provides information on how to use alerts and administrative messages.

Plan incoming e-mail (SharePoint Server 2010)

The incoming e-mail feature of Microsoft SharePoint Server 2010 enables SharePoint sites to receive and store e-mail messages and attachments in lists and libraries. This article helps server and farm administrators understand the choices they need to make before they deploy the incoming e-mail feature for their organization.

In this article:

- [About incoming e-mail](#)
- [Key decisions for planning incoming e-mail](#)
- [Configuration options and settings modes](#)

About incoming e-mail

The incoming e-mail feature enables teams to store the e-mail that they send to other team members without opening the SharePoint site and uploading the content that was sent in e-mail. This is possible because most types of lists and libraries can be assigned a unique e-mail address.

Before configuring incoming e-mail, you must perform the following tasks:

- If you are using the basic scenario, each SharePoint front-end Web server must be running the Simple Mail Transfer Protocol (SMTP) service and the Microsoft SharePoint Foundation Web Application service.
- If you are using the advanced scenario, you can use one or more servers in the server farm to run the SMTP service and to have a valid SMTP server address. Alternatively, you must know the name of a server outside the farm that is running the SMTP service and the location of the e-mail drop folder.

For more information about installing the SMTP service, see [Configure incoming e-mail \(SharePoint Server 2010\)](#).

Key decisions for planning incoming e-mail

As you plan to implement incoming e-mail, you must decide whether to use a basic or and advanced scenario, as described below.

Using a basic scenario

You can enable a basic incoming e-mail scenario by installing the Simple Mail Transfer Protocol (SMTP) service on the server running SharePoint Server 2010 and enabling incoming e-mail by using the automatic settings mode with all default settings. In this scenario, e-mail is delivered directly to your SMTP server and SharePoint Server 2010 periodically checks for e-mail in the default e-mail drop folder that is automatically configured by the SMTP service.

Selecting the automatic settings mode and accepting all the default settings is the easiest way to enable incoming e-mail because all configuration settings are made for you and, therefore, little expertise is required. For most organizations, this configuration is all that is needed.

You enable a basic incoming e-mail scenario in the following steps:

1. The server administrator uses the Add Features Wizard to install the SMTP Server feature on the server from which you want to receive incoming e-mail. This installs and starts the SMTP service on that server.
2. The farm administrator enables incoming e-mail by using the automatic settings mode and accepting all the default values.
3. The site collection administrator enables the incoming e-mail feature on the libraries and lists in which they want to store incoming e-mail and assigns each library and list a unique e-mail address in the form *address@SMTPserveraddress*, for example, *sharedfiles@SMTPserver.contoso.com*.

When users send e-mail to the address of a list or library, SharePoint Server 2010 detects that new e-mail has been delivered and sends it to the appropriate list or library based on the e-mail address.



Note:

You can also use the automatic settings option in an advanced scenario and select whether to use the Microsoft SharePoint Directory Management service, a safe e-mail server, and an incoming e-mail server display address. These options are all discussed in the "Using the advanced scenario" later in this article.

If this basic scenario meets your needs, you can skip the remainder of this article. For more information, see [Configure incoming e-mail \(SharePoint Server 2010\)](#).

Using an advanced scenario

For more advanced administrators, additional choices are available, some of which require more expertise to deploy than choosing the basic scenario with all default options. This section describes the following configuration options:

- SharePoint Directory Management service
- Incoming e-mail server display address
- Safe e-mail server
- E-mail drop folder

If you use the advanced scenario to configure incoming e-mail, you will need to perform additional procedures. For more information, see [Configure incoming e-mail \(SharePoint Server 2010\)](#).

SharePoint Directory Management service

The SharePoint Directory Management service connects SharePoint sites to your organization's user directory to provide enhanced e-mail features. The benefit of using this service is that it enables users to create and manage e-mail distribution groups from SharePoint sites. This service also creates contacts in your organization's user directory so people can find e-mail-enabled SharePoint lists in their

address books. However, using SharePoint Directory Management service requires more management because it is communicating with Active Directory Domain Services (AD DS).



Note:

It is recommended that you use Microsoft Exchange Server together with SharePoint Directory Management service. If you do not, you must customize your own directory management service.

You can configure the SharePoint Directory Management service by using either the automatic or the advanced settings mode. You can choose to enable the SharePoint Directory Management service in your SharePoint server farm, or you can use the SharePoint Directory Management service of another farm. One advantage of using the service running on another farm is that Active Directory permissions are managed in a centralized place (that is, on the other farm).

To enable this service on a server or server farm running SharePoint Server 2010, the SharePoint Central Administration application pool account used by SharePoint Server 2010 must have write access to the container that you specify in Active Directory. This requires an Active Directory administrator to set up the organizational unit (OU) and the permissions on the OU. The advantage of using the SharePoint Directory Management service on a remote farm is that you do not need the help of an Active Directory administrator to create and configure the OU if the OU already exists.



Note:

There are a number of procedures that you need to perform if you plan to use SharePoint Directory Management service. For more information, see [Configure incoming e-mail \(SharePoint Server 2010\)](#).

A typical directory management scenario proceeds in the following steps:

1. A site collection administrator creates a new SharePoint group.
2. The administrator chooses to create a distribution list to associate with that SharePoint group and assigns an e-mail address to that distribution list.
3. Over time, the administrator adds users to and removes users from this SharePoint group. As users are added to and removed from the group, the SharePoint Directory Management service automatically adds and removes them from the distribution list, which is stored in the Active Directory directory service. Because distribution lists are associated with a particular SharePoint group, this distribution list is available to all members of that SharePoint group.
4. By default, e-mail addresses are automatically generated for discussion boards and calendars on team sites and then added to the team distribution list. The e-mail addresses for these two lists will be in the following form, by default: *GroupAddress.discussions* and *GroupAddress.calendar*.
5. By including e-mail addresses for discussion boards and calendars in the distribution list, all e-mail and meeting invitations sent to this distribution list will be archived in the team site.

For more information about SharePoint Directory Management Service, see [Inside SharePoint: SharePoint Directory Integration](#) (<http://go.microsoft.com/fwlink/?LinkId=151766>).

SharePoint Directory Management service configuration options

When you configure the SharePoint Directory Management service to create distribution groups and contacts in Active Directory, you must provide the following information:

- Name of the Active Directory container in which new distribution groups and contacts will be created. This must be provided in the following format:

OU=ContainerName, DC=DomainName, DC=TopLevelDomainName

Example

OU=SharePointContacts, DC=Contoso, DC=com

- Name of the SMTP server to use for incoming e-mail (or accept the default SMTP server if one exists). This must be provided in the following format:

Server.subdomain.domain.top-level_domain

For example, SharePointServer.support.contoso.com

- Whether to accept messages from only authenticated users.
- Whether to allow users to create distribution groups from SharePoint sites. If you choose yes for this option, you can also choose whether users can do any combination of the following actions:
 - Create a new distribution group.
 - Change a distribution group's e-mail address.
 - Change a distribution group's title and description.
 - Delete a distribution group.

When configuring the SharePoint Directory Management service to create distribution groups and contacts using a remote SharePoint Directory Management service, you must provide the following information:

- The URL of the remote directory management service, for example, http://server:adminport/_vti_bin/SharePointEmailWS.asmx.
- The name of the SMTP server to use for incoming e-mail.
- Whether to accept messages from only authenticated users.
- Whether to allow users to create distribution groups from SharePoint sites.

Incoming e-mail server display address

Administrators can specify the e-mail server address that will be displayed in Web pages when users create an incoming e-mail address for a site, list, or group. This setting is often used in conjunction with the SharePoint Directory Management service to provide a more friendly e-mail server address for users to type, for example, mylist@example.com.

Safe e-mail server

You can configure SharePoint Server 2010 to accept e-mail from any e-mail server or only e-mail that has been routed through a safe-e-mail server application.

You can derive the following benefits by routing e-mail through a safe e-mail server:

- **User authentication:** The SMTP service cannot authenticate users who send e-mail to your site, but Exchange Server can. The server administrator can use the SharePoint Central Administration Web site to specify that the system accept e-mail from authenticated users only if the e-mail is sent through Exchange Server.
- **Spam filtering:** Exchange Server provides spam filtering to eliminate unsolicited commercial e-mail before it is forwarded to its destination — in this case, the server running SharePoint Server 2010. Another technique that can reduce spam is to allow members of the team site to archive e-mail only in lists on which you have granted write permissions to members.
- **Virus protection:** Exchange Server provides virus protection for e-mail routed through it.



Note:

Because this option is only available in automatic mode, you cannot specify one or more safe e-mail servers and also specify an e-mail drop folder.

E-mail drop folder

If the SMTP service is running on another server than on the SharePoint server, you must specify the location from which SharePoint Server 2010 retrieves incoming e-mail. You specify the e-mail drop folder so that SharePoint Server 2010 knows from where to retrieve incoming e-mail. However, if you specify a specific e-mail drop folder, SharePoint Server 2010 cannot detect configuration changes on the remote e-mail server that is delivering the e-mail to your drop folder. This means that if an administrator configures the e-mail server to no longer deliver e-mail to this folder, SharePoint Server 2010 cannot detect that the configuration has changed, and therefore will not be able to retrieve the files from the new location.



Note:

When incoming e-mail is set to advanced mode, you must ensure that you have the proper permissions on the e-mail drop folder. For more information, see [Configure incoming e-mail \(SharePoint Server 2010\)](#).



Note:

Because this option is only available in advanced mode, you cannot specify an e-mail drop folder and also specify one or more safe e-mail servers.

Configuration options and settings modes

As a farm administrator, you have two settings modes from which to choose when enabling incoming e-mail: automatic and advanced. As described in the "Using a basic scenario" section, you can choose the automatic settings mode with default settings. However, the automatic settings mode has additional options that you can choose.

The following table describes the configuration options and whether they are configured on the Configure Incoming E-Mail Settings page in Central Administration by using the automatic settings mode or the advanced settings mode.

Configuration option	Automatic settings mode	Advanced settings mode
Safe e-mail servers	Yes	No
E-mail drop folder	No	Yes
SharePoint Directory management service	Yes	Yes
Incoming e-mail server display address	Yes	Yes

The advanced and automatic settings modes are similar in that they both enable farm administrators to configure the SharePoint Directory Management service and the e-mail server address to display in Web pages. These settings modes differ in that the automatic settings mode replaces the ability to choose what e-mail servers to accept e-mail from with the ability to specify the folder to which e-mail is dropped. SharePoint Server 2010 uses this e-mail drop folder to detect new e-mail messages.



Note:

The e-mail drop folder setting is not available in automatic mode, because that mode automatically sets the e-mail drop folder to the folder that is specified by the SMTP service.

Plan incoming e-mail worksheet

Download a Word version of the [Plan incoming e-mail worksheet](http://go.microsoft.com/fwlink/?LinkId=200542) (http://go.microsoft.com/fwlink/?LinkId=200542). Use this worksheet to plan incoming e-mail in order to enable SharePoint sites to receive and store e-mail messages and attachments in lists and libraries.

See Also

[Configure incoming e-mail \(SharePoint Server 2010\)](#)

[Plan outgoing e-mail \(SharePoint Server 2010\)](#)

[Configure outgoing e-mail \(SharePoint Server 2010\)](#)

Plan outgoing e-mail (SharePoint Server 2010)

Outgoing e-mail is the foundation on which site administrators can implement several e-mail notification features. These features help end users track changes and updates to individual site collections and allow site administrators to deliver status messages.

This article helps site administrators understand both the uses for integrating outgoing e-mail and the requirements for integrating it into their site collections.

In this article:

- [About outgoing e-mail](#)
- [Key planning phases of outgoing e-mail](#)

About outgoing e-mail

Properly configuring outgoing e-mail is a requirement for implementing e-mail alerts and notifications. The outgoing e-mail feature uses an outbound Simple Mail Transfer Protocol (SMTP) service to relay e-mail alerts and notifications. These e-mail features include the following:

- **Alerts**

In a large and growing site collection, users need an efficient way to keep up with updates to lists, libraries, and discussions. Setting up alerts provides an effective means to stay on top of changes. For example, if many users work on the same document, the owner of the document can set up alerts to be notified whenever there are changes to this document. Users can specify which areas of the site collection or which documents they want to track and decide how often they want to receive alerts.



Note:

Users must have at least View permissions to set up alerts.

- **Administrative messages**

Site administrators might want to receive notices when users request access to a site or when site owners have exceeded their specified storage space. Setting up outgoing e-mail enables site administrators to receive automatic notifications for site administration issues.

Outgoing e-mail support can be enabled at both the server farm level (available in the **System Settings** section of the Central Administration Web site) and at the Web application level (available in the **Application Management** section of the Central Administration Web site). Therefore, you can specify different settings for a specific Web application. Outgoing e-mail settings at the Web application level override those set up at the server farm level.

Key planning phases of outgoing e-mail

Before you configure outgoing e-mail, you must have an SMTP service to relay e-mail alerts and notifications.

The outgoing e-mail settings include several components that must be considered when planning for this feature:

- An SMTP service to relay e-mail alerts and notifications. You will need the DNS name or IP address of the SMTP mail server to use.
- An address to use in the header of an alert message that identifies the sender of the message.
- A Reply-to address that is displayed in the To field of a message when a user replies to an alert or notification.
- A character set to use in the body of alert messages.

Outbound SMTP server

The SMTP service is a component of Internet Information Services (IIS); however, it is not enabled by default with IIS. It can be enabled by using Add or Remove Programs in Control Panel.

After determining which SMTP server to use, the SMTP server must be configured to allow anonymous access and to allow e-mail messages to be relayed. Additionally, the SMTP server must have Internet access if you want the ability to send messages to external e-mail addresses.

For more information about installing, configuring, and managing the SMTP service, see [Help for Internet Information Services \(IIS\) Manager](http://go.microsoft.com/fwlink/?LinkId=72343) (<http://go.microsoft.com/fwlink/?LinkId=72343>).



Note:

Only a member of the Farm Administrators group can configure an SMTP server. The user must also be a member of the local Administrators group on the server.

From and Reply-to addresses

When configuring outgoing e-mail, you can configure the following two addresses:

- **From address**

Alerts and notifications are sent from an administrative account on the server farm. This account is probably not the one you want to be displayed in the From field of an e-mail message. The address that you use does not need to correspond to an actual e-mail account; it can be a simple friendly address that is recognizable to an end user. For example, "Site administrator" might be an appropriate From address.

- **Reply-to address**

This is the address that will be displayed in the To field of a message if a user replies to an alert or notification. The Reply-to address should also be a monitored account to ensure that end users receive prompt feedback for issues they might have. For example, a help desk alias might be an appropriate Reply-to address.

Character set

When you configure outgoing e-mail, you will need to specify the character set to use in the body of e-mail messages. A character set is a mapping of characters to their identifying code values. The default character set for outgoing e-mail is Unicode UTF-8, which allows most combination of characters (including bidirectional text) to co-exist in a single document. In most cases, the default setting of UTF-8 works well, although East Asian languages are best rendered with their own character set.

Be aware that if you select a specific language code, the text is less likely to appear correctly in mail readers configured for other languages.

See Also

[Configure outgoing e-mail \(SharePoint Server 2010\)](#)

Enterprise search planning (SharePoint Server 2010)

The articles in this section describe the process for planning an enterprise search solution, including analyzing the current environment, determining the search team, and planning the search topology, crawling and federation, people search, and the end-user search experience. The articles also describe how to plan for training for administrators and end-users.

In this section:

[Gather information about the current search environment \(SharePoint Server 2010\)](#). The first step in the planning process is to gather information that helps you plan the search solution. This includes information about your organization, topology, current settings for search, and performance and usage reports.

[Determine the enterprise search team and stakeholders \(SharePoint Server 2010\)](#). This article discusses the roles and stakeholders that you should include when you plan a search solution.

[Plan for crawling and federation \(SharePoint Server 2010\)](#). This article discusses how to plan whether to crawl or federate content, and how to plan the appropriate settings for crawling and federation.

[Plan the topology for enterprise search \(SharePoint Server 2010\)](#). This article describes resources that can help you plan a topology for the search solution, including performance and capacity planning.

Gather information about the current search environment (SharePoint Server 2010)

An important step in planning the enterprise search solution is to gather information about the current environment, including the following types of information and reports:

- Information about the organization
- Information about the topology
- Current settings for search
- Performance and usage reports

You will need this information for planning the search topology, crawling and federation, people search, and the end-user search experience.

Organization information

Gather the following information about the organization:

- User, business, and functional requirements for the enterprise search solution, along with any service level agreements (SLAs). This information will help you to design and build the search solution and verify whether the solution meets the requirements during testing.
- Contact information for existing farm administrators, search administrators, site collection administrators, site owners, and any other stakeholders for the enterprise search solution. This information will help you to plan the enterprise search team, and it also provides a contact list for any communications that occur during planning, deployment, and operations.

Topology information

Gather the following information about the topology:

- Current topology diagrams. You will refer to these while planning the topology and planning for people search.
- Locations of content repositories that should be included in search results, including SharePoint sites, Web sites, file shares, Exchange public folders, business data sources, user profile stores, Lotus Notes, and external sites.
- Locations of users.

Current search settings

If you are starting from a previous version of SharePoint products and technologies, gather the following information about current settings for search:

- Default content access account
- Content source settings, including the following settings for each content source:
 - Content source name
 - Content source type
 - Start addresses
 - Crawl settings
 - Full crawl schedule
 - Incremental crawl schedule
- Crawler impact rules, including the following settings for each crawler impact rule:
 - Site (URL)
 - Request frequency
- Crawl rules, including the following settings for each crawl rule:
 - Path
 - Crawl configuration (excluded or included items)
 - Content access account
- Third-party or custom connectors (called protocol handlers in prior versions)
- File types included in the file-type inclusions list, and whether they required an additional IFilter
- File types removed from the file-type inclusions list
- Languages for which word breakers and stemmers are installed
- Farm-level search settings, including the following information:
 - Contact e-mail address
 - Proxy server settings (address, port, whether to bypass for local addresses, and addresses for which you do not want to use a proxy server)
 - Crawler time-out settings (connection time and request acknowledgement time)
 - SSL certificate warning configuration
- Scope settings
- Crawl settings
- The following additional settings:
 - Federated locations
 - Server name mappings
 - Indexer performance settings
 - Crawled properties
 - Managed properties
 - Search result removal
 - Alerts

- Keywords
- Best Bets
- Authoritative pages

Performance and usage reports

Gather the following performance and usage data:

- Performance metrics from search administration reports, if available. You will use this information when you plan the topology. For more information, see [Use search administration reports \(SharePoint Server 2010\)](#).
- Usage metrics from Web analytics reports. You will use this information when you design the end-user experience for search.

See Also

[Determine the enterprise search team and stakeholders \(SharePoint Server 2010\)](#)

[Plan for crawling and federation \(SharePoint Server 2010\)](#)

Determine the enterprise search team and stakeholders (SharePoint Server 2010)

Your search team should include the following administrative roles:

- Farm administrators
- Search service application administrators
- Site collection administrators (or a representative if there are many)
- Site owners (or a representative if there are many)

For more information about each of these roles in a Microsoft SharePoint Server 2010 environment, see [Choose administrators and owners for the administration hierarchy \(SharePoint Server 2010\)](#).

In addition, you should include the following roles as either team members or project stakeholders:

- Project manager, who defines and drives the planning and implementation of the enterprise search solution
- IT administrators, who manage the information technology organization
- Solutions architects, who design information technology architecture solutions that are often cross-domain and cross-functional
- Developers, who are responsible for customizing search solutions and creating search-enabled applications
- IT operations managers, who plan operations for one or more server farms in the organization
- Other SharePoint service application administrators
- Directory services administrators
- Database administrators who own content that should be included in the search solution
- Any site collection administrators not included in the search team
- Any site owners not included in the search team

See Also

[Gather information about the current search environment \(SharePoint Server 2010\)](#)

[Plan for crawling and federation \(SharePoint Server 2010\)](#)

Plan for crawling and federation (SharePoint Server 2010)

Before end-users can use the enterprise search functionality in Microsoft SharePoint Server 2010, you must crawl or federate the content that you want to make available for users to search. Planning to crawl or federate includes the following tasks:

- [Plan content sources](#)
- [Plan file-type inclusions and IFilters](#)
- [Plan for authentication](#)
- [Plan connectors](#)
- [Plan to manage the impact of crawling](#)
- [Plan crawl rules](#)
- [Plan search settings that are managed at the farm level](#)
- [Plan for federation](#)

Plan content sources

A *content source* is a set of options that you can use to specify what type of content is crawled, what URLs to crawl, and how deep and when to crawl. The default content source is **Local SharePoint sites**. You can use this content source to specify how to crawl all content in all Web applications that are associated with a particular Search service application. By default, for each Web application that uses a particular Search service application, SharePoint Server 2010 adds the start address of the top-level site of each site collection to the default content source..


Some organizations can use the default content source to satisfy their search requirements. However, many organizations have to have additional content sources. Plan additional content sources when you have to do the following:


- Crawl different types of content — for example, SharePoint sites, file shares, and business data.
- Crawl some content on different schedules than other content.
- Limit or increase the quantity of content that is crawled.
- Set different priorities for crawling different sites.

You can create up to 500 content sources in each Search service application, and each content source can contain as many as 500 start addresses. To keep administration as simple as possible, we recommend that you limit the number of content sources that you create.

Plan to crawl different kinds of content

You can only crawl one kind of content per content source. That is, you can create a content source that contains start addresses for SharePoint sites and another content source that contains start addresses for file shares. However, you cannot create a single content source that contains start addresses to both SharePoint sites and file shares. The following table lists the kinds of content sources that you can configure.

Use this kind of content source	For this content
SharePoint sites	<p>SharePoint sites from the same farm or different Microsoft SharePoint Server 2010, Microsoft SharePoint Foundation 2010, or Microsoft Search Server 2010 farms</p> <p>SharePoint sites from the same farm or different Microsoft Office SharePoint Server 2007, Windows SharePoint Services 3.0, or Microsoft Search Server 2008 farms</p> <p>SharePoint sites from Microsoft Office SharePoint Portal Server 2003 or Windows SharePoint Services 2.0 farms</p> <p> Note: Unlike crawling SharePoint sites on SharePoint Server 2010, SharePoint Foundation 2010, or Search Server 2010, the crawler cannot automatically crawl all subsites in a site collection from previous versions of SharePoint Products and Technologies. Therefore, when crawling SharePoint sites from previous versions, you must specify the start address of each top-level site and the URL of each subsite that you want to crawl.</p>
Web sites	<p>Other Web content in your organization that is not located in SharePoint sites</p> <p>Content on Web sites on the Internet</p>
File shares	Content on file shares in your organization
Exchange public folders	Microsoft Exchange Server content
Lotus Notes	E-mail messages stored in Lotus Notes databases

Use this kind of content source	For this content
	 Note: Unlike all other kinds of content sources, the Lotus Notes content source option does not appear in the user interface until you have installed and configured the appropriate prerequisite software. For more information, see Configure and use the Lotus Notes connector (SharePoint Server 2010) .
Business data	Business data that is stored in line-of-business applications

Plan content sources for business data

Business data content sources require that the applications hosting the data are specified in an Application Model in a Business Data Connectivity service application. You can create one content source to crawl all applications that are registered in the Business Data Connectivity service, or you can create separate content sources to crawl individual applications.

Often, the people who plan for integration of business data into site collections are not the same people involved in the overall content planning process. Therefore, include business application administrators in content planning teams so that they can advise you how to integrate the business application data into content and effectively present it in the site collections.

Crawl content on different schedules

You must decide whether some content is crawled more frequently than other content. The larger the volume of content that you crawl, the more likely it is that you are crawling content from different content repositories. The content might not be of the same type and might be located on servers of varying capacities. These factors make it more likely that you have to add content sources to crawl the different content repositories on different schedules.

Primary reasons that content is crawled on different schedules are as follows:

- To accommodate down times and periods of peak usage.
- To more frequently crawl content that is more frequently updated.
- To crawl content that is located on slower servers separately from content that is located on faster servers.

In many cases, not all of this information can be known until after SharePoint Server 2010 is deployed and has run for some time. In these cases, you must specify crawl schedules after the farm is in

production. Nonetheless, it is a good idea to consider these factors during planning so that you can plan crawl schedules based on the information that you have.

The following two sections provide more information about crawling content on different schedules.

Considerations for planning crawl schedules

You can configure crawl schedules independently for each content source. For each content source, you can specify a time to do full crawls and a separate time to do incremental crawls. Note that you must run a full crawl for a particular content source before you can run an incremental crawl. Even if you specify an incremental crawl for content that has not yet been crawled, the system performs a full crawl.



Note:

Because a full crawl crawls all content that the crawler encounters and has at least read access to, regardless of whether that content was previously crawled, full crawls can take significantly more time to complete than incremental crawls.

We recommend that you plan crawl schedules based on the availability, performance, and bandwidth considerations of the crawl and query servers.

When you plan crawl schedules, consider the following best practices:

- Group start addresses in content sources based on similar availability and with acceptable overall resource usage for the servers that host the content.
- Schedule incremental crawls for each content source during times when the servers that host the content are available and when there is low demand on the resources of the server.
- Stagger crawl schedules so that the load on the servers in the farm is distributed over time.
- Schedule full crawls only when you have to for the reasons listed in the next section. We recommend that you run full crawls less frequently than incremental crawls.
- Schedule administration changes that require a full crawl to occur shortly before the planned schedule for full crawls. For example, we recommend that you schedule creating the crawl rule before the next scheduled full crawl so that an additional full crawl is not necessary.
- Base concurrent crawls on the capacity available. For best performance, we recommend that you stagger the crawling schedules of content sources. You can optimize crawl schedules over time as you become familiar with the typical crawl durations for each content source.

Reasons to do a full crawl

Reasons for a Search service application administrator to do a full crawl include the following:

- A software update or service pack was installed on servers in the farm. See the instructions for the software update or service pack for more information.
- A Microsoft Office SharePoint Server 2007 shared services administrator or SharePoint Server 2010 Search service application administrator added a new managed property. A full crawl is

required for the new managed property to take effect immediately. If you do not want the new managed property to take effect immediately, a full crawl is not required.

- You want to re-index ASPX pages on Windows SharePoint Services 3.0 or Microsoft Office SharePoint Server 2007 sites.



Note:

The crawler cannot discover when ASPX pages on Windows SharePoint Services 3.0 or Office SharePoint Server 2007 sites have changed. Because of this, incremental crawls do not re-index views or home pages when individual list items are deleted. We recommend that you periodically do full crawls of sites that contain ASPX files to ensure that these pages are re-indexed.

- You want to detect security changes that were made on a file share after the last full crawl of the file share.
- You want to resolve consecutive incremental crawl failures. If an incremental crawl fails one hundred consecutive times at any level in a repository, the system removes the affected content from the index.
- Crawl rules have been added, deleted, or modified.
- You want to repair a corrupted index.
- The Search service application administrator has created one or more server name mappings.
- The credentials for the user account that is assigned to the default content access account or a crawl rule have changed.

The system does a full crawl even when an incremental crawl is requested under the following circumstances:

- A search administrator stopped the previous crawl.
- A content database was restored, or a farm administrator has detached and reattached a content database.



Note:

If you are running Office SharePoint Server 2007 with the Infrastructure Update for Microsoft Office Servers or SharePoint Server 2010, you can use the restore operation of the Stsadm command-line tool to change whether a content database restore causes a full crawl.


- A full crawl of the site has never been done from this Search service application.
- The change log does not contain entries for the addresses that are being crawled. Without entries in the change log for the items being crawled, incremental crawls cannot occur.

You can adjust schedules after the initial deployment based on the performance and capacity of servers in the farm and the servers hosting content.

Limit or increase the quantity of content that is crawled

For each content source, you can specify how extensively to crawl the start addresses. You also specify the behavior of the crawl by changing the crawl settings. The options that are available for a particular content source vary based on the content source type that you select. However, most crawl options specify how many levels deep in the hierarchy from each start address to crawl. Note that this behavior is applied to all start addresses in a particular content source. If you have to crawl some sites at deeper levels, you can create additional content sources that include those sites.

You can use crawl setting options to limit or increase the quantity of content that is crawled. The options available in the properties for each content source vary depending on the content source type that is selected. The following table describes best practices when you configure crawl setting options.

For this kind of content source	If this pertains	Use this crawl setting option
SharePoint sites	You want to include the content that is on the site itself and you do not want to include the content that is on subsites, or you want to crawl the content that is on subsites on a different schedule.	Crawl only the SharePoint site of each start address
SharePoint sites	You want to include the content on the site itself. -or- You want to crawl all content under the start address on the same schedule.	Crawl everything under the host name of each start address
Web sites	Content available on linked sites is unlikely to be relevant.	Crawl only within the server of each start address
Web sites	Relevant content is located on only the first page.	Crawl only the first page of each start address
Web sites	You want to limit how deep to crawl the links on the start addresses.	Custom — Specify the number of pages deep and number of server hops to crawl  Note: For a highly connected site, we recommend that you start with a small

For this kind of content source	If this pertains	Use this crawl setting option
		number, because specifying more than three pages deep or more than three server hops can crawl all the Internet.
File shares Exchange public folders	Content available in the subfolders is unlikely to be relevant.	Crawl only the folder of each start address
File shares Exchange public folders	Content in the subfolders is likely to be relevant.	Crawl the folder and subfolders of each start address
Business data	All applications that are registered in the BDC metadata store contain relevant content.	Crawl the whole BDC metadata store
Business data	Not all applications that are registered in the BDC metadata store contain relevant content. -or- You want to crawl some applications on a different schedule.	Crawl selected applications

Other considerations when planning content sources

You cannot crawl the same start addresses by using multiple content sources in the same Search service application. For example, if you use a particular content source to crawl a site collection and all its subsites, you cannot use a different content source to crawl one of those subsites separately on a different schedule.

In addition to considering crawl schedules, your decision about whether to group start addresses in a single content source or create additional content sources depends largely upon administration considerations. Administrators often make changes that update a particular content source. Changing a content source requires a full crawl of the content repository that is specified in that content source. To make administration easier, organize content sources in such a way that updating content sources, crawl rules, and crawl schedules is convenient for administrators.

Plan file-type inclusions and IFilters

Content is only crawled if the relevant file name extension is included in the file-type inclusions list and an IFilter is installed on the crawl server that supports those file types. Several file types and IFilters are included automatically during initial installation. When you plan for content sources in your initial deployment, determine whether content that you want to crawl uses file types that are not included. If file types are not included, you must add those file types on the Manage File Types page during deployment and ensure that an IFilter is installed and registered to support that file type.

If you want to exclude certain file types from being crawled, you can delete the file name extension for that file type from the file type inclusions list. Doing so excludes file names that have that extension from being crawled. For a list of file types and IFilters that are installed by default, see [File types and IFilters reference \(SharePoint Server 2010\)](#).

Plan for authentication

When the crawler accesses the start addresses that are listed in content sources, the crawler must be authenticated by, and granted access to, the servers that host that content. This means that the domain account that is used by the crawler must have at least read permissions on the content.

By default, the system uses the default content access account. Alternatively, you can use crawl rules to specify a different content access account to use when crawling particular content. Whether you use the default content access account or a different content access account specified by a crawl rule, the content access account that you use must have read permissions on all content that is crawled. If the content access account does not have read permissions, the content is not crawled, is not indexed, and therefore is not available to queries.

We recommend that the account that you specify as the default content access account has access to most of your crawled content. Only use other content access accounts when security considerations require separate content access accounts.

For each content source that you plan, determine the start addresses that cannot be accessed by the default content access account, and then plan to add crawl rules for those start addresses.



Important:

Ensure that the domain account that is used for the default content access account or any other content access account is not the same domain account that is used by an application pool associated with any Web application that you crawl. Doing so can cause unpublished content in SharePoint sites and minor versions of files (that is, history) in SharePoint sites to be crawled and indexed.

Another important consideration is that the crawler must use the same authentication protocol as the host server. By default, the crawler authenticates by using NTLM. You can configure the crawler to use a different authentication protocol, if it is necessary.

If you are using claims-based authentication, ensure that Windows authentication is enabled on any Web applications to be crawled.

Plan connectors

All content that is crawled requires that you use a connector (known as a protocol handler in previous versions) to gain access to that content. SharePoint Server 2010 provides connectors for all common Internet protocols. However, if you want to crawl content that requires a connector that is not installed with SharePoint Server 2010, you must install the third-party or custom connector before you can crawl that content. For a list of connectors that are installed by default, see [Default connectors \(SharePoint Server 2010\)](#). For information about how to install connectors, see [Install connectors \(SharePoint Server 2010\)](#).

Plan to manage the impact of crawling

Crawling content can significantly decrease the performance of the servers that host the content. The impact that this has on a particular server varies depending on the load that the host server is experiencing and whether the server has sufficient resources (especially CPU and RAM) to maintain service-level agreements under ordinary or peak usage.

Search administrators can use crawler impact rules to manage the impact the crawler has on the servers that are being crawled. For each crawler impact rule, you can specify a single URL or use wildcard characters in the URL path to include a block of URLs to which the rule applies. You can then specify how many concurrent requests for pages are made to the specified URL or decide to request only one document at a time and wait some seconds that you choose between requests.

Crawler impact rules specify the rate at which the crawler requests content from a particular start address or range of start addresses (also known as a site name). A crawler impact rule applies to all content sources in the Search service application and request frequencies apply per crawl component. The following table shows the wildcard characters that you can use in the site name when you are adding or editing a crawler impact rule.

This wildcard character	Has this result
* as the site name	Applies the rule to all sites.
. as the site name	Applies the rule to sites that have dots in the name.
*. <i>site_name</i> .com as the site name	Applies the rule to all sites in the <i>site_name</i> .com domain (for example, *.adventure-works.com).
*. <i>top-level_domain_name</i> as the site name	Applies the rule to all sites that end with a specific top-level domain name, for example, *.com or *.net.
?	Replaces a single character in a rule. For example, *.adventure-works?.com applies to all sites in the domains adventure-works1.com,

This wildcard character	Has this result
	adventure-works2.com, and so on.

You can create a crawler impact rule that applies to all sites in a particular top-level domain. For example, *.com applies to all Internet sites that have addresses that end in .com. For example, an administrator of a portal site might add a content source for samples.microsoft.com. The rule for *.com applies to this site unless you add a crawler impact rule specifically for samples.microsoft.com.

You can coordinate with the administrators of search systems that are crawling content in your organization to set crawler impact rules based on the performance and capacity of the servers. For most external sites, this coordination is not possible. Requesting too much content on external servers or making requests too frequently can cause administrators of those sites to limit access if crawls are using too many resources. During initial deployment, set the crawler impact rules to make as small an impact on other servers as possible while still crawling enough content frequently enough to ensure that the freshness of the index meets your service-level agreement. After the farm is in production, you can adjust crawler impact rules based on data from crawl logs.

Plan crawl rules

Crawl rules apply to all content sources in the search service application. You can apply crawl rules to a particular URL or set of URLs to do the following things:

- Avoid crawling irrelevant content by excluding one or more URLs. This also helps reduce the use of server resources and network traffic, and to increase the relevance of search results.
- Crawl links on the URL without crawling the URL itself. This option is useful for sites that have links of relevant content when the page that contains the links does not contain relevant information.
- Enable complex URLs to be crawled. This option directs the system to crawl URLs that contain a query parameter specified with a question mark. Depending upon the site, these URLs might not include relevant content. Because complex URLs can often redirect to irrelevant sites, it is a good idea to enable this option on only sites where you know that the content available from complex URLs is relevant.
- Enable content on SharePoint sites to be crawled as HTTP pages. This option enables the system to crawl SharePoint sites that are behind a firewall or in scenarios in which the site being crawled restricts access to the Web service that is used by the crawler.
- Specify whether to use the default content access account, a different content access account, or a client certificate for crawling the specified URL.

Because crawling content consumes resources and bandwidth, it is better to include a smaller amount of content that you know is relevant than a larger amount of content that might be irrelevant. After the initial deployment, you can review the query and crawl logs and adjust content sources and crawl rules to be more relevant and include more content.

Plan search settings that are managed at the farm level

Several settings that are managed at the farm level affect how content is crawled. Consider the following farm-level search settings while planning for crawling:

- **Contact e-mail address:** Crawling content affects the resources of the servers that are being crawled. Before you can crawl content, you must provide in the configuration settings the e-mail address of the person in your organization whom administrators can contact if the crawl adversely affects their servers. This e-mail address appears in logs for administrators of the servers being crawled so that those administrators can contact someone if the impact of crawling on performance and bandwidth is too great, or if other issues occur.

The contact e-mail address should belong to a person who has the necessary expertise and availability to respond quickly to requests. Alternatively, you can use a closely monitored distribution-list alias as the contact e-mail address. Regardless of whether the content that is being crawled is stored internally to the organization or not, quick response is important.

- **Proxy server settings:** You can choose whether to use a proxy server when crawling content. The proxy server to use depends on the topology of your SharePoint Server 2010 deployment and the architecture of other servers in your organization. You will likely have to use a proxy server when crawling Internet content. For more information about how to configure proxy server settings for search, see [Configure farm-level proxy server settings \(SharePoint Server 2010\)](#) and [Configure proxy server settings for search \(SharePoint Server 2010\)](#).
- **Time-out settings:** Time-out settings are used to limit the time that the search system waits while connecting to other services.
- **SSL setting:** The Secure Sockets Layer (SSL) setting determines whether the SSL certificate must exactly match to crawl content.

Plan for federation

Federated search is the concurrent querying of multiple Web resources or databases to generate a single search results page for end-users. When you add a federated location, end-users can search for and retrieve content that has not been crawled by servers in the local system. Federated locations enable queries to be sent to remote search engines and feeds. Accordingly, the system renders the results to end-users as if the federated content were part of the crawled content.

SharePoint Server 2010 supports the following types of federated locations:

- **Search index on this server.** You can use any local or remote site in your organization that has a server that is running SharePoint Server 2010 as a federated location. For example, imagine that a SharePoint site on a Human Resources server in your company is the only available source of employee contact information. Even if the site is not part of your crawl scope, you can configure a federated location for it so that users who initiate a search from your Search Center site can retrieve employee contact information results that they are authorized to see. The following conditions apply:

- a. The location is set to **Search Index on this Server**.
- b. No query template is required. SharePoint Server 2010 uses the object model to query a location.
- c. Default server authentication is used.
- d. Advanced search queries are not supported.
- **OpenSearch 1.0 or 1.1.** You can use any public Web site that supports the OpenSearch standard as a federated location. An example of such a location is an Internet search engine such as Bing, or a search results page that supports RSS or Atom protocols. For example, imagine that you want users who search your internal sites for proprietary technical research to also see related research information from public Web sites. By configuring a federated location for a Bing search query, Web search results will be automatically included for users. The following conditions apply:
 - a. Queries can be sent to a search engine as a URL, such as <http://www.example.com/search.aspx?q=TEST>.
 - b. Search results are returned in RSS, Atom, or another structured XML format.
 - c. Location capabilities, query templates, and response elements are part of an OpenSearch description (.osdx) file that is associated with the location.
 - d. Extensions to OpenSearch that are specific to SharePoint Server 2010 support the ability to include triggers and the ability to associate XSL code with search results.
 - e. The choice of metadata to display in the search results is determined by the OpenSearch location.

For more information about OpenSearch, visit <http://www.opensearch.org>.

When a search query is sent to a federated location, it is sent as URL parameters in a format called a query template. The system then formats and renders the results as XML for users of the Search Center site. The XML is displayed in a Web Part on the search results page as readable text. You can add and configure Web Parts on the search results page as a Federated Search Results Web Part, Top Federated Results Web Part, or Core Results Web Part. By default, the search results page contains three Federated Search Results Web Parts.

Consider the following questions when you are determining whether you want to display federated search results to users:

1. **Do you want to display custom results for particular searches?** To help ensure that the federated location returns results that match specific queries, you can use trigger rules. When you create a trigger rule for a federated location, the Web Part that is associated with that location displays results only for user queries that match the pattern or prefix that you specify.
2. **Can you use a URL to specify which results to retrieve for a query?** To create a federated location, you must specify a query template, which is the combination of the URL and the parameters that are required to send a search query and return the results as XML. When you add this information to the **Query template** field on the Add Federated Location page, you must format the string correctly (as shown in the example on the Add Federated Location page) or the search results provider will not return any results.

3. **Can users access the links that are provided by the federated location?** If your organization grants only limited access to Internet resources, using an Internet search engine as a federated location might frustrate users because they will not be able to view some search results.
4. **Is authentication required?** If the federated location requires authentication, you must provide the correct credentials. Many federated locations, such as Internet search engines, do not require credentials.

Plan authentication types for federation

Several kinds of user authentication, per-user and common credentials, are available for federated search. However, realize that collecting credentials requires a Web Part extension for non-Kerberos authentication types in per-user authentication. In the authentication and credentials information section of the location definition, you specify the authentication type for the federated location. The authentication type can be one of the following:

- **Anonymous**
No credentials are required to connect to the federated location.
- **Common**
Each connection uses the same set of credentials to connect to the federated location.
- **Per-user**
The credentials of the user who submitted the search query are used to connect to the federated location.

For the common and per-user authentication types, you must also specify one of the following authentication protocols:

- **Basic**
Basic authentication is part of the HTTP specification and is supported by most browsers.

noteDXDOC112778PADS Security Note

Web browsers that use Basic authentication transmit passwords that are not encrypted. By monitoring communications on the network, a malicious user can use publicly available tools to intercept and decode these passwords. Therefore, we do not recommend Basic authentication unless you are confident that the connection is secure, such as with a dedicated line or a Secure Sockets Layer (SSL) connection.

- **Digest**
Digest authentication relies on the HTTP 1.1 protocol as defined in the RFC 2617 specification at the World Wide Web Consortium (W3C) Web site. Because Digest authentication requires HTTP 1.1 compliance, some browsers do not support it. If a browser that is not compliant with HTTP 1.1 requests a file when Digest authentication is enabled, the request is rejected because Digest authentication is not supported by the client. Digest authentication can be used only in Windows domains. Digest authentication works with Windows Server 2008, Windows Server 2003, and

Microsoft Windows 2000 Server domain accounts only, and may require the accounts to store passwords as encrypted plaintext.

- NTLM

User records are stored in the security accounts manager (SAM) database or in the Active Directory database. Each user account is associated with two passwords: the LAN Manager-compatible password and the Windows password. Each password is encrypted and stored in the SAM database or in the Active Directory database.

- Kerberos (per-user authentication type only)

By using the Kerberos protocol, a party at either end of a network connection can verify that the party on the other end is the entity it claims to be. Although NTLM enables servers to verify the identities of their clients, NTLM does not enable clients to verify a server's identity, nor does NTLM enable one server to verify the identity of another. NTLM authentication is designed for a network environment in which servers are assumed to be trusted.

- Forms-based

A forms-based authentication cookie is nothing but the container for an authentication ticket. Each request passes the ticket as the value of the cookie and the ticket is used on the server to identify an authenticated user. However, cookieless forms-based authentication passes the ticket in the URL in an encrypted format. Cookieless forms-based authentication is used because client browsers might block cookies. This feature is introduced in the Microsoft .NET Framework 2.0.

If you are using claims-based authentication in your environment, ensure that Windows authentication is also enabled on any content sources to be crawled. For more information about authentication methods in SharePoint Server 2010, see [Plan authentication methods \(SharePoint Server 2010\)](#).

See Also

[Gather information about the current search environment \(SharePoint Server 2010\)](#)

[Determine the enterprise search team and stakeholders \(SharePoint Server 2010\)](#)

Plan the topology for enterprise search (SharePoint Server 2010)

This article lists resources that you can use to help you plan a search topology for Microsoft SharePoint Server 2010. These resources include information about how to plan server farms and topologies, search architectures, governance, and capacity.

The following resources can help you plan a search topology:

- [Performance and capacity test results and recommendations \(SharePoint Server 2010\)](#) This white paper helps you plan for performance and capacity for SharePoint Server 2010.
- [White paper: SharePoint 2010 for hosters \(SharePoint Server 2010\)](#) This white paper helps you plan the multi-tenancy requirements in a hosted environment.
- [Search Environment Planning for Microsoft SharePoint Server 2010](#) (<http://go.microsoft.com/fwlink/?LinkID=167736>) This technical diagram describes primary architecture design decisions for search environments.
- [Search Architectures for Microsoft SharePoint Server 2010](#) (<http://go.microsoft.com/fwlink/?LinkID=167739>) This technical diagram details the physical and logical architecture components that make up a search system and shows common search architectures.
- [Design Search Architectures for Microsoft SharePoint Server 2010](#) (<http://go.microsoft.com/fwlink/?LinkID=167742>) This technical diagram describes initial design steps to determine a basic design for a SharePoint Server 2010 search architecture.

See Also

[Gather information about the current search environment \(SharePoint Server 2010\)](#)

[Determine the enterprise search team and stakeholders \(SharePoint Server 2010\)](#)

[Plan for crawling and federation \(SharePoint Server 2010\)](#)

Planning worksheets for SharePoint Server 2010

In this article:

- [Planning worksheets by task](#)
- [Planning worksheets by title](#)

This article provides links to worksheets that you can use to record information that you gather and decisions that you make as you plan your deployment of Microsoft SharePoint Server 2010. Use these worksheets in conjunction with — not as a substitute for — [Planning and architecture for SharePoint Server 2010](#).

Planning worksheets by task

For this task	Use this worksheet	To do this
Plan sites and site collections (SharePoint Server 2010)	Site planning data worksheet (http://go.microsoft.com/fwlink/?LinkID=167837)	Plan top level site collections and sites, and record decisions about site themes and navigation.
Plan site navigation (SharePoint Server 2010)	Site planning data worksheet (http://go.microsoft.com/fwlink/?LinkID=167837)	Plan top level site collections and sites, and record decisions about site themes and navigation.
Plan for using themes (SharePoint Server 2010)	Site planning data worksheet (http://go.microsoft.com/fwlink/?LinkID=167837)	Plan top level site collections and sites, and record decisions about site themes and navigation.
Plan incoming e-mail (SharePoint Server 2010)	Plan incoming e-mail worksheet (http://go.microsoft.com/fwlink/?LinkId=200542)	Plan incoming e-mail in order to enable SharePoint

For this task	Use this worksheet	To do this
		sites to receive and store e-mail messages and attachments in lists and libraries.
Plan content deployment (SharePoint Server 2010)	Content deployment data worksheet (http://go.microsoft.com/fwlink/?LinkID=167835)	Plan the export and import servers in the farms in your content deployment topology, and to plan the content deployment paths and jobs.
Plan managed metadata (SharePoint Server 2010)	Term sets planning worksheet (http://go.microsoft.com/fwlink/?LinkID=163486)	Determine basic taxonomy, including term, usage, owner, and group.
Plan managed metadata (SharePoint Server 2010)	Detailed term set planning worksheet (http://go.microsoft.com/fwlink/?LinkID=163487)	Determine taxonomy including detailed identifying characteristics such as measurements.
Plan managed metadata (SharePoint Server 2010)	Managed metadata services planning worksheet (http://go.microsoft.com/fwlink/?LinkID=164578)	Plan to share metadata information using managed metadata services and connections.
Document management planning (SharePoint Server 2010)	Document management participants worksheet (http://go.microsoft.com/fwlink/?LinkID=165871)	Identify document management planning stakeholders and record document management practices.
Document management planning	Analyze document usage worksheet (http://go.microsoft.com/fwlink/?LinkID=165873)	Record information gathered when analyzing document

For this task	Use this worksheet	To do this
(SharePoint Server 2010)		usage.
Document management planning (SharePoint Server 2010)	Policy worksheet (http://go.microsoft.com/fwlink/?LinkID=165883)	Plan information management policies for content types.
Records management planning (SharePoint Server 2010)	In-place records planning worksheet (http://go.microsoft.com/fwlink/?LinkID=185011)	Identify record types and content types to be stored in normal document libraries.
Plan for backup and recovery (SharePoint Server 2010)	Backup and recovery planning workbook (http://go.microsoft.com/fwlink/?LinkID=184385)	Help you plan strategies for backup and recovery for SharePoint Server 2010 environment.
Document management planning (SharePoint Server 2010)	Document management planning (SharePoint Server 2010)	Plan a content type.
Plan and prepare for upgrade (SharePoint Server 2010)	Upgrade worksheet (http://go.microsoft.com/fwlink/?LinkID=179928)	Record information about your environment while you prepare for upgrade.
Metadata-based routing and storage planning (SharePoint Server 2010)	Content Organizer settings worksheet (http://go.microsoft.com/fwlink/?LinkID=189018&clcid=0x409)	Determine and record how the content organizer settings in your site can be an effective part of your metadata-based content routing and storage solution.

For this task	Use this worksheet	To do this
Metadata-based routing and storage planning (SharePoint Server 2010)	Content Organizer rule worksheet (http://go.microsoft.com/fwlink/?LinkId=189019&clcid=0x409)	Plan rules that will be an effective part of your metadata-based routing and storage solution.

Planning worksheets by title

Use this worksheet	For this task	To do this
Analyze document usage worksheet (http://go.microsoft.com/fwlink/?LinkId=165873)	Document management planning (SharePoint Server 2010)	Record information gathered when analyzing document usage.
Backup and recovery planning workbook (http://go.microsoft.com/fwlink/?LinkId=184385)	Plan for backup and recovery (SharePoint Server 2010)	Help you plan strategies for backup and recovery for SharePoint Server 2010 environment.
Content deployment data worksheet (http://go.microsoft.com/fwlink/?LinkId=167835)	Plan content deployment (SharePoint Server 2010)	Plan the export and import servers in the farms in your content deployment topology, and to plan the content deployment paths and jobs.
Content Organizer rules worksheet (http://go.microsoft.com/fwlink/?LinkId=189019&clcid=0x409)	Metadata-based	Plan rules that will be an

Use this worksheet	For this task	To do this
	routing and storage planning (SharePoint Server 2010)	effective part of your metadata-based routing and storage solution.
Content Organizer settings worksheet (http://go.microsoft.com/fwlink/?LinkID=167835)	Metadata-based routing and storage planning (SharePoint Server 2010)	Determine and record how the content organizer settings in your site can be an effective part of your metadata-based content routing and storage solution.
Content type worksheet (http://go.microsoft.com/fwlink/?LinkID=165878)	Document management planning (SharePoint Server 2010)	Plan a content type.
Detailed term set planning worksheet (http://go.microsoft.com/fwlink/?LinkId=163487&clcid=0x409)	Plan managed metadata (SharePoint Server 2010)	Determine taxonomy including detailed identifying characteristics such as measurements .
Document libraries worksheet (http://go.microsoft.com/fwlink/?LinkID=165874)	Document management planning (SharePoint Server 2010)	Plan libraries based on sites and on document types.
Document management participants worksheet (http://go.microsoft.com/fwlink/?LinkID=165871)	Document management	Identify document

Use this worksheet	For this task	To do this
	t planning (SharePoint Server 2010)	management planning stakeholders and record document management practices.
In-place records planning worksheet (http://go.microsoft.com/fwlink/?LinkId=185011&clcid=0x409)	Records management planning (SharePoint Server 2010)	Identify record types and content types to be stored in normal document libraries.
Managed metadata services planning worksheet (http://go.microsoft.com/fwlink/?LinkId=164578)	Plan managed metadata (SharePoint Server 2010)	Plan to share metadata information using managed metadata services and connections.
Plan incoming e-mail worksheet (http://go.microsoft.com/fwlink/?LinkId=200542)	Plan incoming e-mail (SharePoint Server 2010)	Plan incoming e-mail in order to enable SharePoint sites to receive and store e-mail messages and attachments in lists and libraries.
Policy worksheet (http://go.microsoft.com/fwlink/?LinkId=165883)	Document management planning (SharePoint Server 2010)	Plan information management policies for content types.
Site planning data worksheet	Plan sites	Plan top level

Use this worksheet	For this task	To do this
(http://go.microsoft.com/fwlink/?LinkID=167837)	and site collections (SharePoint Server 2010) Plan site navigation (SharePoint Server 2010) Plan for using themes (SharePoint Server 2010)	site collections and sites, and record decisions about site themes and navigation.
Term sets planning worksheet (http://go.microsoft.com/fwlink/?LinkId=163486)	Plan managed metadata (SharePoint Server 2010)	Determine basic taxonomy, including term, usage, owner, and group.
Upgrade worksheet (http://go.microsoft.com/fwlink/?LinkId=179928)	Plan and prepare for upgrade (SharePoint Server 2010)	Record information about your environment while you prepare for upgrade.