

Microsoft SharePoint 2013

Designing and Architecting Solutions

Shannon Bray
Miguel Wood
Patrick Curran

Microsoft SharePoint 2013

Designing and Architecting Solutions

Get the information you need to make good SharePoint implementation decisions

Determine the best design for your SharePoint implementation by gaining a deeper understanding of how the platform works. Written by a team of SharePoint experts, this practical guide introduces the Microsoft SharePoint 2013 architecture, and walks you through design considerations for planning and building a custom SharePoint solution. It's ideal for IT professionals, whether or not you have experience with previous versions of SharePoint.

Discover how to:

- Dive deeper into SharePoint 2013 architecture components
- Gather requirements for a solution that fits your needs
- Upgrade from SharePoint 2010 to SharePoint 2013
- Design service applications for performance and redundancy
- Provide the right storage plan for a SharePoint farm
- Map authentication and authorization requirements to your solution
- Take steps necessary to design a security enhanced implementation
- Plan your business continuity management strategy
- Validate your SharePoint architecture to ensure success

About the Authors

Shannon Bray, MCT, MCPD, MCITP, MCSE, MCSA, is the Chief Architect at a large organization. He is a Microsoft Certified Solutions Master (MCSM) for SharePoint and a Microsoft Certified Master (MCM) for SharePoint 2010.

Miguel Wood, MCPD, MCITP, MCSE, MCSA, MCT, helps organizations improve business processes through technology. He is a Microsoft Certified Solutions Master (MCSM) for SharePoint and a Microsoft Certified Master (MCM) for SharePoint 2010.

Patrick Curran, MCT, MCTS, MCP, MCITP, MCPD, MCSA, is a Senior SharePoint Architect. Patrick helps clients throughout the world integrate SharePoint into their organizations.

ISBN: 978-0-7356-7168-3



U.S.A. \$29.99
Canada \$31.99
[Recommended]

Microsoft Office/Microsoft SharePoint



Microsoft SharePoint 2013: Designing and Architecting Solutions

Shannon Bray
Miguel Wood
Patrick Curran

Copyright © 2013 by Shannon Bray, Miguel Wood, and Patrick Curran

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

ISBN: 978-0-7356-7168-3

1 2 3 4 5 6 7 8 9 LSI 8 7 6 5 4 3

Printed and bound in the United States of America.

Microsoft Press books are available through booksellers and distributors worldwide. If you need support related to this book, email Microsoft Press Book Support at mspinput@microsoft.com. Please tell us what you think of this book at <http://www.microsoft.com/learning/booksurvey>.

Microsoft and the trademarks listed at <http://www.microsoft.com/about/legal/en/us/IntellectualProperty/Trademarks/EN-US.aspx> are trademarks of the Microsoft group of companies. All other marks are property of their respective owners.

The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted herein are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

This book expresses the author's views and opinions. The information contained in this book is provided without any express, statutory, or implied warranties. Neither the authors, Microsoft Corporation, nor its resellers, or distributors will be held liable for any damages caused or alleged to be caused either directly or indirectly by this book.

Acquisitions and Developmental Editor: Kenyon Brown

Production Editor: Christopher Hearse

Editorial Production: S4Carlisle Publishing Services

Technical Reviewer: Chris Givens

Indexer: Bob Pfahler, Potomac Indexing, LLC

Cover Composition: Ellie Volckhausen

Illustrator: S4Carlisle Publishing Services

Contents at a glance

Introduction

xv

PART I PLANNING FOR MICROSOFT SHAREPOINT 2013

CHAPTER 1	Understanding the Microsoft SharePoint 2013 architecture	3
CHAPTER 2	Introducing Windows PowerShell and SharePoint 2013 cmdlets	37
CHAPTER 3	Gathering requirements	61

PART II DESIGN CONSIDERATIONS FOR MICROSOFT SHAREPOINT 2013

CHAPTER 4	Understanding the service application model	91
CHAPTER 5	Designing for SharePoint storage requirements	125
CHAPTER 6	Mapping authentication and authorization to requirements	171
CHAPTER 7	Designing for platform security	213
CHAPTER 8	Upgrading your SharePoint 2010 environment	245

PART III CONCEPTS FOR INFRASTRUCTURE RELIABILITY

CHAPTER 9	Maintaining and monitoring Microsoft SharePoint	297
CHAPTER 10	Planning your business continuity strategy	339
CHAPTER 11	Validating your architecture	375

Index

431

Contents

<i>Introduction</i>	xv
---------------------------	----

PART I PLANNING FOR MICROSOFT SHAREPOINT 2013

Chapter 1	Understanding the Microsoft SharePoint 2013 architecture	3
------------------	---	----------

Exploring the SharePoint farm components	3
The SharePoint farm hierarchy	4
Examining the SharePoint file system	18
IIS files	19
SharePoint Root	22
Feature Fallback behavior	25
Deployed solutions	26
Examining the SharePoint databases	28
SQL Server aliases	29
SharePoint system databases	29
SharePoint service application databases	30
Putting it all together	32

Chapter 2	Introducing Windows PowerShell and SharePoint 2013 cmdlets	37
------------------	---	-----------

Exploring the role of Windows PowerShell	38
Brief history of Windows PowerShell	38
Basic Windows PowerShell syntax	39
Understanding the benefits of Windows PowerShell	39
Windows PowerShell 3.0 enhancements	42

Chapter 3 Gathering requirements 61

PART II DESIGN CONSIDERATIONS FOR MICROSOFT SHAREPOINT 2013

Chapter 4 Understanding the service application model 91vi Contents

Service application proxies	94
Service application implementation	95
Database(s) and service applications	96
Examining the changes in OWA	96
How does it work?	97
Examining the changes in workflows	98
How does it work?	99
Discovering the new web service applications available in	
SharePoint 2013	100
Access Services	100
The App Management service	101
The Machine Translation service	102
The WMS application	103
Discovering the updated web service applications available in	
SharePoint 2013	104
Access Services 2010	105
The BDC service	105
Excel Services	106
The MMS application	106
The PPS application	107
The Search service application	108
SSS	111
UPA	111
VGS	112
WAS	113
Discovering the service applications that SharePoint	
creates automatically	114
The Application Discovery and Load Balancer service	115
STS	115
Exploring service federation	117
Putting it all together	119
Creating the farm trust	120
Configuring the topology service	121
Publishing your service application	122

What do you think of this book? We want to hear from you!

Microsoft is interested in hearing your feedback so we can continually improve our books and learning resources for you. To participate in a brief online survey, please visit:

microsoft.com/learning/booksurvey

Connecting to your service application	123
Setting the service application permissions	123

Chapter 5 Designing for SharePoint storage requirements 125

Database engine requirements for SharePoint 2013	126
Overview of HA options	127
Failover clustering	127
Database mirroring	127
Log shipping	127
AlwaysOn Failover Cluster Instances.	128
AlwaysOn Availability Groups	128
Initial optimization and configuration of SQL Server for SharePoint 2013.	128
SQL Server collation.	128
SQL Server MDOP	130
Additional SQL Server considerations	132
Overview of SharePoint Server 2013 databases and their roles.	138
SharePoint Foundation 2013 databases.	138
SharePoint Server 2013 databases	142
SharePoint 2013 Integrated Reporting Services databases	144
SQL Server 2008 R2 (SP1) and SQL Server 2012 system databases.	145
SharePoint 2013 database capacity planning	145
SQL Server storage and IOPS.	145
Enabling SQL Server BI within SharePoint 2013	147
Shredded Storage in SharePoint 2013.	160
Putting it all together.	161
Introducing SQLIO	162
Testing your SQL Server internal drives	162
Testing your iSCSI network drives	166
Testing your SQL Server drives from SharePoint	167
Stress testing your SQL Server drives	168
Disk subsystem performance information.	169

What do you think of this book? We want to hear from you!

Microsoft is interested in hearing your feedback so we can continually improve our books and learning resources for you. To participate in a brief online survey, please visit:

microsoft.com/learning/booksurvey

Chapter 6	Mapping authentication and authorization to requirements	171
	Analyzing AuthN options	171
	Windows authentication.	172
	Anonymous authentication	182
	Claims-based authentication	184
	Examining SharePoint authentication components and methodologies.	197
	Authentication zones.	197
	AAMs	198
	Self-Service Site Creation	199
	Site collection host headers	200
	Investigating authentication (AuthN) services.	200
	Claims to Windows Token Service (c2WTS).	200
	Secure Store Service (SSS).	202
	Business Connectivity Services (BCS)	202
	Exploring authorization (AuthZ) features.	203
	Application policies	203
	The People Picker.	206
	Sharing.	208
	Putting it all together.	209
Chapter 7	Designing for platform security	213
	Preventing, tracking, and reporting on SharePoint installations	214
	Preventing SharePoint installations	214
	Tracking SharePoint installations.	215
	Reporting SharePoint installations	217
	Encrypting communications	218
	Certification Authorities (CAs)	219
	Communication between the client and server	219
	Server-to-server communication	223
	SharePoint server communications.	223
	SSL and SQL Server	225
	IPsec IKEv2.	229

Planning for and configuring SQL Server Transparent Data Encryption	231
Installing SharePoint using least privileges	233
Application pools	234
User accounts	235
SharePoint Managed Accounts	238
Understanding roles and group permissions	239
Roles	239
Group permissions	240
Putting it all together	240

Chapter 8 Upgrading your SharePoint 2010 environment 245

Introducing upgrades	246
Authentication upgrade	247
Database attach upgrade	247
In-place upgrade	247
Supported databases	247
Unsupported databases	248
Upgrade implementation improvements	248
Overview of upgrade process	253
Preparing your upgrade	256
Cleaning up your content databases	256
Cleaning up your environment	257
Documenting your environment	258
Documenting current settings	258
Documenting environment settings	260
Documenting service settings	260
Managing customizations	261
Documenting other services	263
Classic-mode authentication	266
Language packs	267
Limiting downtime	267
Testing your upgrade	269
Upgrading your test farm	269
Validating your test upgrade	271

Learning from your deployment	271
Implementing your upgrade	271
Minimizing downtime	271
Upgrading your production farm	272
Monitoring progress	273
Validating your upgrade	274
Verifying your upgrade	274
Troubleshooting your upgrade	274
Putting it all together	275
Scenario	275
Current farm	275
End goal	277
Test environment	278
Test migration strategy	279
Building the test farm	279
Upgrading the Search Service application	280
Upgrading the Search Center	281
Federating the Search service	282
Upgrading the remaining service applications	283
Trust but verify	286
Service application creation	286
Customizations	287
Upgrading the content databases	287
Upgrading My Site content	288
Upgrading sites	290
Back to Search	292
Finally!	293

PART III CONCEPTS FOR INFRASTRUCTURE RELIABILITY

Chapter 9 Maintaining and monitoring Microsoft SharePoint 297

Monitoring a SharePoint environment	298
Understanding SharePoint monitoring requirements	299
Monitoring the health of SharePoint	303
Configuring usage and health provider settings	304

Monitoring performance counters	308
Monitoring page performance	311
Monitoring SharePoint storage	312
Tuning and optimizing a SharePoint environment	313
Resource throttling	314
SQL optimization	315
Defining and executing a database maintenance plan	317
Fine-tuning network performance	320
Planning and configuring caching	322
Introducing Request Management	329
Troubleshooting a SharePoint environment	332
Understanding the issues through logs	332
Putting it all together	334

Chapter 10 Planning your business continuity strategy 339

Planning for your business continuity needs	340
Understanding what to protect	340
Applying BCM objectives	340
Exploring the features native to SharePoint	342
Native backup and restore	343
Site-collection backup	345
Exporting a site or list	346
Recovering data from an unattached content database	348
Restoring data using the Recycle Bin	348
Read-only improvements	350
Avoiding service disruptions	351
Implementing various business continuity techniques	353
Failover clustering	354
Database mirroring	357
Log shipping	361
AlwaysOn Availability	363
Implementing network load balancing	368

Putting it all together.	370
Contoso requirements.	370
Key factors.	371
Solution	372

Chapter 11 Validating your architecture 375

Verify farm contentment	376
Microsoft ULS Viewer	376
Developer Dashboard	379
Verifying port allocation and web requests.	380
Fiddler	381
Internet Explorer Developer Tools	381
Verifying Kerberos with Klist.	382
Inspecting your network packets.	382
Microsoft Network Monitor.	383
Microsoft Message Analyzer	383
Testing your environments	383
Inspecting your IIS logs.	384
Performance-testing your environment	388
Load-testing your environment.	394
Stress-testing your environment	414
Other load test options.	417
Putting it all together.	418
Scenario	418
Verifying sites	420
Verifying Kerberos	421
Setting up Visual Studio Ultimate 2010	423
Create the load test	424
Documentation.	428
Finally!	430

<i>Index</i>	431
--------------	-----

Introduction

Welcome to *Microsoft SharePoint 2013: Designing and Architecting Solutions*. The purpose of this book is to help Microsoft SharePoint professionals understand the Microsoft SharePoint 2013 architecture and give them the tools they need to be successful in planning, designing, building, and validating infrastructure architectures. As with previous versions of SharePoint, SharePoint 2013 contains many features with which you will be familiar. Some features have not changed at all; others will have changed, but at a high level, and will provide similar functionality that will not be new to you. However, there are new components that you will need time to understand fully before you can decide how they will benefit you and your organization.

With this version of SharePoint, Microsoft focuses on various implementations whether they exist on premises, in the cloud, or a combination of the two. With the focus now on social collaboration and the ability to share content in a variety of ways, how you build your implementation will define how users can share, organize, discover, build, and manage ideas and content in a SharePoint environment.

The following descriptions define these concepts further:

- **Share** You can share your content and information, spreading it socially, spreading it online, and spreading it easily across multiple places and devices where you might need to interconnect, whether it is on premises, mobile, tablet, in a cloud, or at a client site.
- **Organize** This is how you structure and categorize the information, whether it is project, team, or information held in documents using Microsoft Office 2013 applications, such as Microsoft Outlook or Microsoft Project, and syncing your content in SharePoint to your desktop with Microsoft SkyDrive Pro.
- **Discover** This concept includes connecting people across your organization, the discovery of insights and answers through the use of business intelligence, and finding what you're looking for by using enterprise Search. In this version of SharePoint, Microsoft has invested a great deal of effort into the integration of enterprise Search.
- **Build** SharePoint 2013 has undergone major changes to the application model for how to build applications that are hosted on systems that are maintained by organizations, on premises, or when the systems are maintained outside the control of an organization, in the cloud; how to publish these applications internally through a corporate catalog; and how to publish them outside an

organization, as well as sharing them across on-premises farms and cloud-based farms through a public store. The new application mode also makes it possible for applications to be shared within Microsoft Office applications using the new Windows 8 interface computers, laptops, Ultrabooks, tablets, and Windows Phone. These are now introduced to the Office 2013 applications.

- **Manage** SharePoint 2013 provides better support for managing SharePoint as a platform. It can be run in the cloud with Microsoft Office 365. It contains new archiving, eDiscovery, and case management capabilities that include SharePoint 2013, Microsoft Exchange Server 2013, and Microsoft Lync 2013.

You can find more details about Office 365 at office365.microsoft.com.

Microsoft still aims for SharePoint to be a self-service product; that is, a product that provides users with the ability to complete their tasks with no-code solutions by using the browser and Office applications.

SharePoint 2013 consists of two products: Microsoft SharePoint Foundation 2013 and Microsoft SharePoint Server 2013. The exposure of two sets of functionality still exists and is implemented using standard and enterprise client access licenses with a new licensing model. There is no longer a separate Microsoft FAST Search Server for SharePoint. You will find much of the functionality that was included in that product now incorporated as part of SharePoint 2013. Another change is that Office Web Apps (OWA) is a separate product and should be installed on servers on which SharePoint is not installed. Also, if your organization is a heavy user of SharePoint to automate business processes, there are changes that allow you to distribute the workflow business logic onto servers where SharePoint is not installed. How you design your solutions will weigh heavily on the architecture and design strategies that are applied during the planning phases. Because planning is an important aspect, this book also shares with you how SharePoint has changed, what questions you need to ask to be successful in designing the architecture, how service applications work, and much more.

Who this book is for

Although this book offers insight into many of the new features of SharePoint 2013, it is not designed for the typical user or business user. This book is best suited for SharePoint professionals who plan on designing, planning, or implementing architectures that support organizations ranging from the small to large enterprise farms. This book will help you understand how SharePoint works at its core and will

provide everything from how SharePoint is structured to how to take your SharePoint 2010 environment and upgrade it to SharePoint 2013. Finally, one of the topics that has rarely been addressed is the validation of your infrastructures. This book will walk you through the concepts you need to not only build successful solutions, but to test them against a wide variety of workloads.

This book does not provide step-by-step instructions on how to install or complete tasks by using SharePoint 2013 or provide an in-depth coverage or analysis of the new functions. You can find those details in the following books:

- *Microsoft SharePoint 2013 Plain & Simple* by Johnathan Lightfoot, Michelle Lopez, and Scott Metker, which is aimed at users who are new to SharePoint.
- *Microsoft SharePoint 2013 Step by Step* by Olga Londer and Penelope Coventry, which is aimed at new and intermediate users.
- *Microsoft SharePoint 2013 Inside Out* by Darvish Shadravan, Penelope Coventry, Tom Resing, and Christine Wheeler, which is aimed at intermediate and advanced power users (who are also referred to as *citizen* or *consumer developers*). This book is also aimed at project managers, business analysts, and small business technicians.
- *Microsoft SharePoint 2013 App Development* by Scot Hillier and Ted Pattison, which is aimed at professional developers.

Regardless of your role, we hope that this book helps you to understand how SharePoint works at its core and that you take away the importance of how planning and design can provide success in your new SharePoint 2013 environment.

Assumptions about you

This book is designed for readers who have experience with installing and administering SharePoint; it is assumed that the reader isn't brand new to the topic, and many of the topics in this book are considered either essential building blocks for a successful implementation or advanced topics. A book of this size cannot cover every feature; therefore, it is assumed that you have advanced familiarity with SharePoint already. The focus is on planning, designing, building, upgrading, and testing your SharePoint 2013 infrastructures.

Organization of this book

This book provides a high-level preview of the various new or changed features you might want to use in SharePoint 2013. This book is structured so that you, as an IT professional, understand the architectural changes before detailing features that the business might need you to install. This book is broken down into three distinct parts: “Planning for Microsoft SharePoint 2013,” “Design considerations for Microsoft SharePoint 2013,” and “Concepts for infrastructure reliability.”

Chapter 1, “Understanding the Microsoft SharePoint 2013 architecture,” discusses the core architecture components of SharePoint, including how the file system works and coexists with both v14 and v15 architectures and feature fallback behaviors. The chapter breaks down the core components of the SharePoint farm and the databases.

Chapter 2, “Introducing Windows PowerShell and SharePoint 2013 cmdlets,” introduces the reader to what Windows PowerShell has to offer and explains how to use self-discovery to learn what you need to know to be successful.

Chapter 3, “Gathering requirements,” discusses the importance of gathering requirements prior to implementation and how the requirements will affect design choices. The chapter highlights logical and physical architecture design so that you will have these critical concepts in mind while learning the specifics of SharePoint design concepts.

Chapter 4, “Understanding the service application model,” goes into depth on the differences between services and service applications in SharePoint 2013, while highlighting design considerations for each. It will ensure that you understand how to properly plan and implement service applications that have supporting databases or support federated scenarios.

Chapter 5, “Designing for SharePoint storage requirements,” identifies what storage options are available for SharePoint 2013 and why understanding storage requirements is important to the process. The chapter then discusses planning for Microsoft SQL Server 2008 R2, SQL Server 2012, and SQL Server Reporting Services.

Chapter 6, “Mapping authentication and authorization to requirements,” goes into detail about the changes between SharePoint 2010 and SharePoint 2013 authentication models and then discusses options for each. The chapter provides a step-by-step process for configuring SharePoint to work with Windows, Forms-Based, and Federated authentication models. The chapter closes by going into detail about the changes in SharePoint authorization and highlights features that should be considered prior to the rollout.

Chapter 7, “Designing for platform security,” highlights topics such as privileges, service and managed accounts, shell admin accounts, preventing and tracking installations, antivirus options, server communications, and other platform security topics to ensure that you understand how your decisions affect how SharePoint works.

Chapter 8, “Upgrading your SharePoint 2010 environment,” discusses the new features of the SharePoint 2013 upgrade process and makes sure that you have a clear understanding of the options that are available. The chapter then goes into detail on preparing for and performing all aspects of the upgrade process. It closes by discussing troubleshooting techniques and common pitfalls and concerns.

Chapter 9, “Maintaining and monitoring Microsoft SharePoint,” discusses monitoring and maintaining the environment to ensure that it keeps up with the changes in the organization. This chapter highlights timer jobs, SharePoint reports, and the ULS Viewer. It continues by giving you an understanding of the health analyzer and demonstrates how to implement custom rules. The chapter closes by discussing patching strategies.

Chapter 10, “Planning your business continuity strategy,” discusses the importance of planning reliability. Prior to building the SharePoint farm, an organization should have a plan that focuses on how their decisions will affect their responses to critical issues. This chapter highlights planning an organization’s SLAs, RPO, and RTO and discusses how to build a BCM plan that will satisfy those requirements.

Chapter 11, “Validating your architecture,” discusses how to validate the design choices that have been implemented in the farm. It demonstrates how to create load tests that will test and validate your architecture to ensure that it holds up to the peak demands of your organization.

We hope that this book serves you well as you move forward with the new SharePoint 2013 platform.

Acknowledgments

One of the most challenging things one can do is choose to write a book and then complete it. Writing a book on a brand-new technology ups the bar a bit because the process often starts while the product is still in the early beta cycles. Screens and features change, and core pieces of functionality are either introduced late in the cycle or removed just before the release of the product. Knowing this, the authors of this book have tried to give you their best attempt at sharing the knowledge that they believe is critical when designing architectures. There are a number of folks on our team that helped bring this project to light, namely Kenyon Brown, Microsoft Press

senior editor, who presented us with the opportunity to write this book. We would also like to thank our project editor, Kathryn Duggan, for her tireless efforts to make sure that we presented our thoughts in a clear and concise manner. Her job had to be difficult, and her efforts are truly appreciated. The final stages of the book process offer their own set of challenges. We would like to thank our production editor, Christopher Hearse, for putting the finishing touches on our chapters. Finally, we would like to thank our technical editor, Chris Givens, for his efforts to make sure that the information presented here is both clear and factual. Chris did a fantastic job and really helped the authoring team to present the book we all tried so desperately to produce.

Shannon Bray

Since this was my second book, I had envisioned that the process would have magically become easier, but as mentioned above, writing a book about SharePoint while it is still in beta is difficult, to say the least. I knew that I would need help bringing the best book possible to the market, and I chose my team very carefully. I would truly like to thank the efforts of Patrick Curran and Miguel Wood for helping me complete this book. We certainly found enough challenges to last a lifetime. I am also super pleased that my longtime friend, Chris Givens, agreed to join our project. I have known Chris for many years and always respected him as both author and trainer.

When you are writing a book in the beta stages, it is impossible to personally understand every aspect of the product. With the release of SharePoint 2013, there were many hidden components that I needed to dive into. Often, I found myself referring to the hard work of members of the Microsoft Certified Solution Master (MCSM) community and while it is impossible to name them all, I want to thank Spence Harbar and Wictor Wilén for their contributions to both the community and for allowing me to reference some of their work in this book. I would also like to thank the members and instructors of the MSCM beta rotation (U3); I was lucky to have spent two fantastic weeks with the best SharePoint folks on the planet.

Finally, I would be remiss if I didn't thank my beautiful family. I spent many more hours at the keyboard than I had expected, and it took time away from them. Without the support of my wife, Anna, and her ability to keep our two daughters (Eden and Kenna) wrangled in, I would not have been able to complete this project.

Patrick Curran

I am not usually one to mince words and spend a lot of time talking, so when Shannon Bray asked me to help write this book, I was a bit nervous about having to come up with content. Luckily, I was working with Shannon and Miguel, so when I did have questions, I was able to get excellent guidance for content. It was also fantastic

working with Chris Givens because I really felt like I was writing the book for him. Having such a perfectionist as Chris for our technical editor, I did not want to disappoint and bore him with trivial SharePoint information. It was great to come up with things in this book that were both new to Shannon and Miguel (the MCSMs) and Chris, so I think that there is content in this book suitable for everyone.

I cannot express enough how grateful I am to Shannon Bray, who since day one at Planet Technologies has challenged me to continually learn more about SharePoint and become more involved with the SharePoint community. It has been a pleasure to work with him on this project, and also get to know him personally over the years. Thank you, Shannon, for everything.

I was lucky enough to meet Miguel Wood in Denver before he joined the Planet Technologies team, and I don't think I have ever met anyone who could usurp knowledge as well as Miguel. Every conversation I have with this man, I learn something new and simultaneously see that there is another level of knowledge that I have yet to obtain. I am lucky to have access to Miguel for guidance on a professional level, and a personal level, and really appreciate the guidance that he has given me over the years. Thank you, Miguel.

Having locked myself in my office for many nights to help get this book into your hands, I would like to thank my beautiful wife, Sandra, for having to manage our three lovely children in the evenings and a grumpy, sleep-deprived adult in the mornings. I must thank my father, Arthur, who has done nothing but encourage and support me through whatever wackiness I decide to pursue. And I want to mention my beautiful wee-ones, Liam, Mamie, and Michelle, who have been more understanding and patient than any children should have to be at the nightly disappearance of their father.

Miguel Wood

When I agreed to assist with writing this book, I had no idea of the challenge I was undertaking. I must first thank my coauthors for keeping me to task and for their much needed assistance in this endeavor. It is an honor to work with both Shannon and Patrick, not only on this book, but professionally as well. Also, I cannot fail to echo the previous acknowledgment of Chris Givens, our technical editor, and the entire Microsoft Press team in their undeserving patience with me and ensuring a quality product.

The knowledge and ability required to write a book like this requires the acknowledgment of many people, starting with my parents, for allowing me to indulge my technical addiction at such a young age. There is no doubt that I must also acknowledge, especially for this book, my fellow MCM R6 (#WUB) and MCSM Beta instructors and candidates for filling in my many gaps of the technical knowledge

required to participate in authoring this book. Of course, I would also like to thank the numerous clients and friends who entrusted me to assist them in successfully meeting their visions and goals with the SharePoint platform.

During the entire book authoring process, personal life is near nonexistent, especially while maintaining a demanding work schedule. Therefore, I must thank my loving, supportive wife, Cara, and my sons for understanding and forgiving the extended time spent away from them, during not only this project but also my demanding work and travel schedule. Honestly, without their undying support, I could not have completed this book or be the person I am today.

Finally, I need to thank my closest friend and brother, Allen Capps, for always believing in me, even when I didn't; my in-laws, for their faith and personal investment in my profession; and, finally, my close friends in Time Machine for maintaining my minimal sanity and "enabling" Mike Honcho via live music, crazy antics, and the blue dolly.

Errata & Book Support

We've made every effort to ensure the accuracy of this book and its companion content. Any errors that have been reported since this book was published are listed on our Microsoft Press site:

<http://aka.ms/SP2013DAS/errata>

If you find an error that is not already listed, you can report it to us through the same page.

If you need additional support, email Microsoft Press Book Support at *mspinput@microsoft.com*.

Please note that product support for Microsoft software is not offered through the addresses above.

We want to hear from you

At Microsoft Press, your satisfaction is our top priority, and your feedback our most valuable asset. Please tell us what you think of this book at:

<http://www.microsoft.com/learning/booksurvey>

The survey is short, and we read every one of your comments and ideas. Thanks in advance for your input!

Stay in touch

Let's keep the conversation going! We're on Twitter: *<http://twitter.com/MicrosoftPress>*

Introducing Windows PowerShell and SharePoint 2013 cmdlets

In this chapter, you will learn about:

- Exploring the role of Windows PowerShell with Microsoft SharePoint.
- Understanding the benefits of Windows PowerShell.
- Configuring user permissions for Windows PowerShell and SharePoint 2013.
- Examining the Windows PowerShell Management Shells.
- Working with Windows PowerShell cmdlets.

Since the release of Microsoft SharePoint 2010, administrators have needed some form of knowledge about Windows PowerShell. With SharePoint 2013, knowledge of Windows PowerShell has become more of a requirement since Windows PowerShell is replacing the deprecated, but still supported, *Stsadm.exe* command-line utility. As such, it is critical for SharePoint administrators to have strong Windows PowerShell skills. It is well documented that the SharePoint Central Administration web user interface does not allow for full configuration, administration, and management of the SharePoint environment.

It is important to set the proper expectations for this chapter. Although this chapter will introduce essential Windows PowerShell knowledge for SharePoint administrators, it is not intended to provide complete knowledge of Windows PowerShell, the SharePoint 2013 PowerShell cmdlets, and all the possible combination of scripts or uses; to do so would require an entire book, if not more. Instead, this chapter concentrates on giving you the ability to discover and learn the best ways to create powerful management scripting tools, as well as evaluate the numerous online Windows PowerShell snippets available publicly. To refer to the cliché, the purpose of this chapter is to teach you to fish, or, rather be self-sufficient, rather than simply handing you the fish or reviewing every single cmdlet available in Windows PowerShell and the SharePoint 2013 PowerShell library.

It is also important to note that as of January 2010, all Microsoft server products comply with the Microsoft Common Engineering Criteria (CEC), which requires Windows PowerShell support. Therefore, all of the core technologies and the majority of systems that will integrate with SharePoint will also support, if not rely on, Windows PowerShell scripting to configure, manage, and administer their systems as well.

Windows PowerShell is critically important to SharePoint administration, but knowledge of Windows PowerShell fundamentals is a critical skillset for any Microsoft IT professional or any individual using Microsoft products and technologies.

Exploring the role of Windows PowerShell

Windows PowerShell is a task automation tool created by Microsoft to allow users, typically server administrators, to automate and perform tasks on their on-premises SharePoint environment server(s) or while working with SharePoint Online environments. In this section, you will be introduced to a quick history of the tool and then explore the basic syntax.

Brief history of Windows PowerShell

Windows PowerShell 1.0 was first released in conjunction with Windows Server 2008. At the time, it was described as a new task-based command-line shell and scripting language designed especially for system administration. Windows PowerShell is built on the Microsoft .NET Framework and is meant to assist IT professionals to control and automate the administration of the Windows operating system and applications that run on Windows. Windows PowerShell has built-in commands, called *cmdlets* (pronounced “command-lets”), which let Windows PowerShell users manage the computers from the command line.

Windows PowerShell 2.0 was released in conjunction with Windows Server 2008 R2 and Windows 7. However, a downloadable install, known as the *Windows Management Framework*, was also made available for previous versions of the Windows server and client operating systems. The Windows Management Framework not only included Windows PowerShell 2.0, but also the Windows Remote Management (WinRM) 2.0 and Background Intelligent Transfer Server (BITS) 4.0. Arguably, the single most important improvement in Windows PowerShell 2.0 was the introduction of the feature to allow remoting—the ability to run commands on one or more remote computers from a single computer running Windows PowerShell. Windows PowerShell remoting allows for multiple ways of connecting, including interactive (1:1), fan-out (1:many), and fan-in [many:1 by using an Internet Information System (IIS) hosting model]. Additionally, Windows PowerShell 2.0 introduced the Windows PowerShell Integrated Scripting Environment (ISE). The Windows PowerShell ISE is a host application for Windows PowerShell, allowing users to run commands and write, test, and debug scripts in a single Windows-based graphical user interface (GUI) with multiline editing, tab completion, syntax coloring, selective execution, context-sensitive help, and support for right-to-left languages.

Finally, Windows PowerShell 3.0 was released in conjunction and as part of the Windows Server 2012 and Windows 8 operating systems. Again, however, a downloadable version of the Windows Management Framework 3.0 includes Windows PowerShell 3.0. Windows Management Framework 3.0 and is available only for Windows Server 2008 R2 with Service Pack 1 (SP1) and/or Windows 7 with Service Pack 1 (SP1). Additionally, the .NET Framework 4.0+ is a prerequisite requirement install on both Windows Server 2008 R2 SP1 and Windows 7 SP1.

When SharePoint Server 2013 is installed, applicable Windows PowerShell cmdlets are available via the SharePoint 2013 Management Shell. Basically, the SharePoint 2013 Management Shell is simply a Windows PowerShell shell that registers the SharePoint 2013 PowerShell extensions (Microsoft.SharePoint.PowerShell.dll) for use within the shell. Most, if not all, aspects of SharePoint 2013 can be managed via Windows PowerShell and the SharePoint extensions.

Basic Windows PowerShell syntax

To reduce the complexity of Windows PowerShell scripting and syntax, all Windows PowerShell scripts revolve around cmdlets. A cmdlet is a single-feature command consisting of a verb and noun separated by a dash (-) that manipulates objects in Windows PowerShell. Examples include *Get-Help*, *Get-Process*, and *Start-Service* (Verb-Noun). This is very different than other command-line interfaces (CLIs) or shells where commands are comprised of executable programs ranging from basic to complex.

In Windows PowerShell, most cmdlets are simple and designed to be used in combination with other cmdlets. For example, the *get* cmdlets only retrieve data, the *cmdlets* only establish or change data, the *format* cmdlets only format data, and the *out* cmdlets only direct the output to a specified destination.

Additionally, PowerShell cmdlets have a help file that you can access by typing:

```
PS C:\> Get-Help <cmdlet-name> -Detailed
```

The detailed view of the cmdlet help file includes a description of the cmdlet, the command syntax, descriptions of the parameters, and example(s) that demonstrate use of the cmdlet. However, specific sections of the cmdlet help file can be accessed via specific parameters such as *-Examples*.

In essence, the most important fact to remember in Windows PowerShell syntax is the verb-noun format with required and/or optional parameters.

Understanding the benefits of Windows PowerShell

GUIs, including the SharePoint Central Administration web application and site, typically use common, basic concepts understood by most computer users. A CLI such as Windows PowerShell uses a different approach to expose information. As it is completely text based from the user's perspective, it is important to know command names before using them. However, it is possible to compose complex commands that are equivalent to the features in a GUI environment. It is important to become familiar with commonly used commands and command parameters. Unfortunately, most CLIs do not have patterns that can help the user to learn the interface. Because CLIs were the first operating system shells, many command names and parameter names were not selected using common language or syntax.

Windows PowerShell is designed to improve the command-line and scripting environments by eliminating longstanding challenges and adding new, modern features, including:

- **Discoverability** Windows PowerShell makes it easy to discover commands and syntax, as well as features.

For example, to find a list of cmdlets that view and change Windows services, simply type:

```
Get-Command *-service
```

After discovering the cmdlet that accomplishes a task, it is possible to learn more about the specific cmdlet by using the *Get-Help* cmdlet. For example, to display help about the *Get-Service* cmdlet, type:

```
Get-Help Get-Service
```

To fully understand the output of the specific cmdlet, pass (also known as “pipe,” based on the character being used) the output object to the *Get-Member* cmdlet to see all the members of the returned *Get-Service* output “object.” For example, the following command displays information about the members of the object output by the *Get-Service* cmdlet:

```
Get-Service | Get-Member
```

- **Consistency** The consistency of Windows PowerShell is one of its primary assets. For example, knowing how to use the *Sort-Object* cmdlet for one scenario enables the user to know how to use it for other scenarios. It is not necessary to learn different sorting routines for different cmdlets or systems. Additionally, new cmdlet developers can rely on existing cmdlets rather than creating new cmdlets for the same functionality. Windows PowerShell encourages developers to use the framework that provides basic features, as well as be consistent about the usage of the interface.
- **Interactive and scripting environments** Windows PowerShell is a combined interactive and scripting environment that grants access to command-line tools and COM objects, and also enables access to the power of the .NET Framework Class Library (FCL). This environment improves upon the Windows command prompt (*Cmd.exe*), which provides an interactive environment with multiple command-line tools. It also improves upon Windows Script Host (WSH) scripts, which let you use multiple command-line tools and COM automation objects but do not provide an interactive environment.
- **Object orientation** Although Windows PowerShell is text-based from an interaction perspective, Windows PowerShell is based on objects, not text. The output of a command is an object. It is then possible to send (or “pipe”) the output object of one command to another command as its input. Therefore, Windows PowerShell provides a familiar interface for people with other shell experience, while introducing new object-oriented command-line functionality. In other words, it extends the concept of sending data between commands by enabling a user to send objects, rather than simply text as used in STSADM. This also increases the speed in which cmdlets will execute when compared to its STSADM counterparts.

- **Easy transition to scripting** Windows PowerShell makes it easy to transition from typing commands interactively to creating and running scripts. It is possible to enter commands at the Windows PowerShell command prompt to discover the commands that perform a task. Once those commands are known and/or tested, save those commands in a transcript or a history before copying them to a file for use as a script. This offers several advantages; the first is that it documents a repeatable process. The second is the speed at which the process can be duplicated.

Although it may be tempting to use the SharePoint 2013 Products Configuration Wizard and/or the SharePoint Farm Configuration Wizard, most experienced SharePoint professionals, as well as Microsoft, will recommend against the use of wizards in production SharePoint 2013 environments. Additionally, many SharePoint 2013 administrators prefer to utilize the SharePoint 2013 Central Administration web application for making configuration changes to the farm and/or its servers and services.

Utilizing Windows PowerShell to manage and administer the configuration of a SharePoint 2013 environment has many advantages, including:

- **Change control** It is surprising how many production environments do not maintain a change control log. When configuration changes are done via SharePoint 2013 Central Administration or another graphical interface, creating change control log entries can be extremely difficult and usually incomplete. Short of numerous screenshot captures and/or video files, it can be unclear as to what changes have been made to the environment.
- **Disaster recovery** One of the most beneficial items of utilizing Windows PowerShell for installation and configuration is maintaining the scripts in the event of a catastrophic system disaster. The saved scripts can be used to quickly re-create the environment in a disaster recovery situation.
- **Documentation** Most organizations, although directed, fail to maintain proper documentation on their SharePoint 2013 environments. By using Windows PowerShell, it is easy to add the executed scripts from installation, configuration, and change to maintain the documentation of the environment.
- **Fine-tuning configuration** It should be obvious, but not everything can be done via SharePoint 2013 Central Administration or other GUIs. In fact, only via Windows PowerShell can you potentially access all the possible configurations to ensure maximum efficiency and performance of the SharePoint 2013 environment.

It needs to be pointed out that there are certain cases where Windows PowerShell should and should not be used (the latter very rarely), especially when installing and configuring a SharePoint 2013 environment. The most prominent of these scenarios is when configuring the SharePoint 2013 Search service application. You will see an example of this in the final section of this chapter.

Windows PowerShell 3.0 enhancements

Since you will be working with Windows PowerShell 3.0 in SharePoint 2013, it is important to know about several of the significant features that improve its usability and allow you to better manage environments. With Windows PowerShell 3.0, new core cmdlets have been added to support much of the new functionality of Windows PowerShell 3.0. Since Windows PowerShell 3.0 is built upon .NET Framework 4.0, you can use new classes in Windows PowerShell, including Parallel Computing, Windows Communication Foundation (WCF), and Windows Workflow Foundation (WF).

Module autoloading is a significant enhancement to Windows PowerShell 3.0. When using the *Get-Command* cmdlet, it searches and retrieves all cmdlets and functions from all modules that are installed on the computer system, even if the module is not imported into the current session. Automatic importing of modules is triggered by using the cmdlet in a command, running *Get-Command* for a cmdlet without wildcards, or running *Get-Help* for a cmdlet without wildcards. The automatic loading of modules can be controlled by using the *\$PSModuleAutoLoadingPreference* preference variable. The Windows PowerShell 3.0 console improves the tab completion functionality by completing the names of cmdlets, parameters, parameter values, enumerations, .NET Framework types, COM objects, hidden directories, and more.

Arguably, one of most important new features of Windows PowerShell 3.0 is the ability to schedule Windows PowerShell background jobs and manage them both in Windows PowerShell and Task Scheduler. Essentially, scheduled jobs are a combination of Windows PowerShell background jobs and Task Scheduler tasks. Scheduled jobs run asynchronously in the background, like Windows PowerShell background jobs. But, like Task Scheduler tasks, you can run scheduled jobs on a one-time or recurrent schedule or in response to an action or event. You can also view and manage scheduled jobs in Task Scheduler, enable and disable them as desired, use them as templates, run them, and set conditions for which they start automatically. A perfect example would be performing backups of SharePoint on a scheduled basis.

With Windows PowerShell 3.0, support for an updatable, enhanced online help system has been introduced. You can check for and download updated help files for the cmdlets in your modules. The *Update-Help* cmdlet automatically checks and identifies the newest help files, downloads them, validates them, and installs them in the correct language-specific directory for the module. The *Get-Help* cmdlet has been enhanced with a new parameter, *-Online*, which will open the online version of the help topic in your default web browser.

New to Windows PowerShell 3.0 is the ability to utilize persistent, user-managed disconnected sessions, using a PSSession object. The PSSession object is created by using the *New-PSSession* cmdlet and are saved on the remote computer. Unlike Windows PowerShell 2.0, the PSSession is no longer dependent on the session in which it is created. It is now possible to disconnect from a session without disrupting the commands running in the session. Therefore, you can close the session and even shut down the computer and later reconnect to the remote session from the same or different computer system. Additional cmdlets and parameters have been introduced to support the use of remote, disconnected sessions. However, both the originating (client) and terminating (server) ends of the connection must be running Windows PowerShell 3.0.

These are just a few of the Windows PowerShell 3.0 enhancements that can significantly increase your ability to manage your SharePoint and Windows systems.

Configuring permissions for Windows PowerShell and SharePoint 2013

Depending on whether your SharePoint 2013 environment is hosted on-premises or through SharePoint Online, the administrator who is executing the various SharePoint cmdlets will need a number of permissions. By default, the account that is used to install SharePoint will have the required rights and will be able to provide other accounts with these permissions if needed. The *Add-SPShellAdmin* cmdlet can be used to grant permissions for users to run the 2013 cmdlets, but to be able to execute this cmdlet yourself, you must have membership in both the *securityadmin* fixed server role on the SQL Server instance and the *db_owner* fixed database role on the databases that are to be updated. Additionally, you will also need to be a member of the Administrators group on the server on which you are running the Windows PowerShell cmdlets. The *Add-SPShellAdmin* cmdlet will add a user to the *SharePoint_Shell_Access* role for a specified database. If the role does not exist, it will be created at the time of execution. If no database is provided in the *-database* parameter, then the default behavior is to apply the settings to the Central Administration configuration database. If you use the *-database* parameter, the user is added to the role on the farm configuration database, the Central Administration content database, and the specified database. Using the *-database* parameter is the preferred method because most of the administrative operations require access to the Central Administration content database. In addition to adding the specified user to the *SharePoint_Shell_Access* role for the specified database, the user is also added to the *WSS_Admin_WPG* group on all web servers. In SharePoint 2010, the use of *Add-SPShellAdmin* would also add the user specified by the *-UserName* parameter to the *dbo_owner* role of the database; this is no longer the case in SharePoint 2013. Executing the cmdlet on a particular database will add the user to the *SharePoint_Shell_Access* and *SPDataAccess* roles. The following example will grant a backup administrator account (spAdmin2) the appropriate database permissions to execute SharePoint cmdlets against the database. It is worth noting that you will need to execute the *Add-SPShellAdmin* cmdlet for all database to which you want to grant access:

```
$db = Get-SPContentDatabase Contoso_Content_Main  
Add-SPShellAdmin -UserName spAdmin2 -database $db
```

Additionally, the execution policy will be restricted for Windows PowerShell execution. To enable Windows PowerShell execution, you will need to change the execution policy (see Figure 2-1). Remembering your basic format of Windows PowerShell cmdlets, you will use the verb-noun combination of *Set-ExecutionPolicy* with the desired parameters. Setting the execution policy can be scoped out to the individual user (as shown in Figure 2-1), or to the server level (as shown in Figure 2-2).

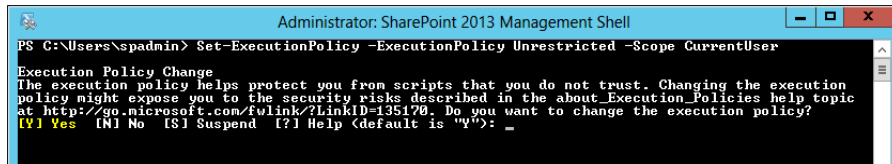


FIGURE 2-1 Users have the ability to set permissions for the execution policy for the current user only.

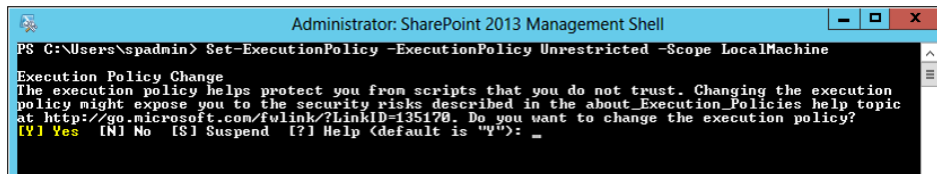


FIGURE 2-2 Users have the ability to set execution policy at the server level.

To run scripts, the minimum required execution policy for SharePoint 2013 is RemoteSigned, although the default policy for Windows PowerShell is Restricted. If the policy is left as Restricted, the SharePoint 2013 Management Shell will change the policy for Windows PowerShell to RemoteSigned. This means that you must select Run As Administrator to start the SharePoint 2013 Management Shell with elevated administrative permissions. This change will apply to all Windows PowerShell sessions.

In regards to SharePoint Online, you must be assigned the global administrator role on the SharePoint Online site on which you are running the Windows PowerShell cmdlets. You will learn more about the SharePoint Online management shell in the next section.

Examining the Windows PowerShell management shells

As a SharePoint professional, you will have a number of tools available to allow you to interact with a particular SharePoint environment. While much of the information in this chapter is directed at executing Windows PowerShell cmdlets in on-premises deployments, SharePoint Online has its own management shell that you should be aware of. The SharePoint Online Management Shell allows the SharePoint professional to manage users, sites, and organizations instead of using the SharePoint Online Administration Center. To set up the SharePoint Online Management Shell environment, you will need to install the Windows Management Framework 3.0 (<http://go.microsoft.com/fwlink/?LinkID=244693>) and the SharePoint Online Management Shell (<http://go.microsoft.com/fwlink/?LinkID=255251>). Once the shell is installed, you will need to execute the `Connect-SPOService` cmdlet similar to the following example. This will need to be done prior to being able to manage users and site collections:

```
Connect-SPOService -Url https://contoso-admin.sharepoint.com
                  -credential admin@contoso.com
```



Note If you forget to execute the *Connect-SPOService* cmdlet prior to executing other SharePoint Online cmdlets, you will receive an error message stating that no connection is available.

With the install of SharePoint Server 2013 on premises, you will have direct access to the SharePoint 2013 Management Shell. The SharePoint Management Shell is simply a Windows PowerShell shell with the additional execution of a script to add the necessary Windows PowerShell cmdlet library and shell environmental variables to easily access SharePoint cmdlets and commands. The actual startup script's default location is:

```
C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\
  15\CONFIG\PowerShell\Registration\SharePoint.ps1
```

However, most administrators will prefer to use the more fully functional Windows PowerShell Integrated Scripting Environment (ISE) included with the Windows Server operating system.

Windows PowerShell can be utilized to run many aspects of Windows Server 2012, and out of the box, there are few ways to run your Windows PowerShell cmdlets on the server. As shown in Figure 2-3, users have the ability to use the default command-line-driven Windows PowerShell environment and add the appropriate snap-ins for what they are trying to accomplish. For example, if a user wanted to use Windows PowerShell to execute SharePoint cmdlets, they would need to add the SharePoint snap-in as follows:

```
PS C:\> Add-PSSnapin Microsoft.SharePoint.PowerShell
```

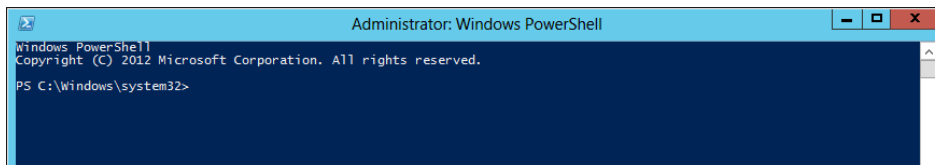


FIGURE 2-3 You can execute cmdlets with out-of-the-box Windows PowerShell.

Unfortunately, by default, the Windows PowerShell ISE will not include the SharePoint cmdlets. If you attempt to load the snap-in after it has been loaded, it will throw an error. While the error will not halt the execution, you probably don't want to see red text flying by every time you execute your scripts. To eliminate the error, it is common that you will see *-EA 0* at the end of the module name. This is telling Windows PowerShell to handle the error silently. You can use either of the following shorthand examples to load the SharePoint snap-in:

```
PS C:\> asnp Microsoft.SharePoint.PowerShell -EA 0
```

```
PS C:\> asnp *sharepoint* -EA 0
```

However, as seen in Figure 2-1 and Figure 2-2, users can simply open the SharePoint 2013 Management Shell to execute SharePoint-specific cmdlets. The downside to using the shell is that it too is a command-line-driven tool. While there are hundreds of cmdlets, once the snap-in has been loaded, you will be able to select cmdlets by typing the verb and then tabbing (that is, using the Tab key and/or Shift-Tab key combination) through a list of available cmdlets. One improvement over the previous version of Windows PowerShell and SharePoint is that the parameters are now viewable in IntelliSense, making it much easier to be productive with the tool.

With Windows Server 2008 R2, Microsoft introduced the Windows PowerShell ISE. ISE is an environment built for users to have more robust control of their scripts and how the scripts are run. In Windows Server 2008 R2, ISE is a Windows feature that needs to be enabled either manually or by running the following cmdlet:

```
PS C:\> Add-WindowsFeature PowerShell-ISE
```

With Windows Server 2012, ISE is already up and running by default. Figure 2-4 shows the ISE for Windows PowerShell 3.0 in Windows Server 2012.

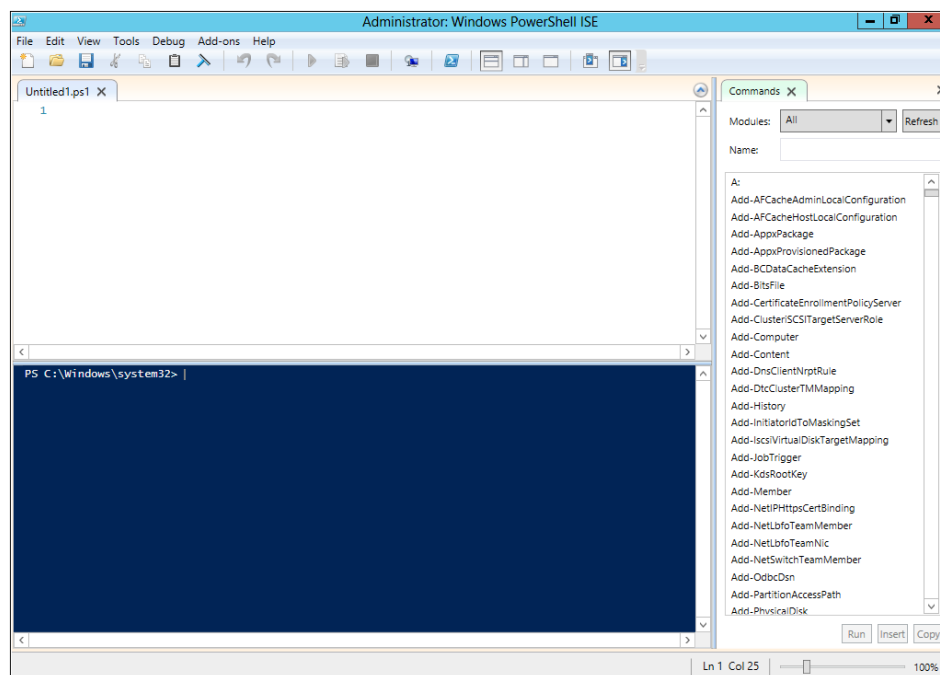


FIGURE 2-4 Windows PowerShell ISE is available by default in Windows Server 2012.

Since the SharePoint 2013 cmdlets and environment variables are not available by default when using the Windows PowerShell ISE, you can either add the SharePoint snap-in or change the profile for either a specific user or for all users. By default, no profiles are present on a fresh install of a server, but they can be modified to preload the SharePoint snap-in.

Selecting a profile to use in Windows PowerShell ISE

Windows PowerShell ISE supports profiles for the current user and all users of Windows PowerShell ISE. It also supports the Windows PowerShell profiles that apply to all hosts. The profile that you use is determined by how you use Windows PowerShell and Windows PowerShell ISE.

If you use only Windows PowerShell ISE to run Windows PowerShell, then save all your items in one of the ISE-specific profiles, such as the `CurrentUserCurrentHost` profile for Windows PowerShell ISE or the `AllUsersCurrentHost` profile for Windows PowerShell ISE.

If you use multiple host programs to run Windows PowerShell, save your functions, aliases, variables, and commands in a profile that affects all host programs, such as the `CurrentUserAllHosts` or the `AllUsersAllHosts` profile, and save ISE-specific features, like color and font customization, in the `CurrentUserCurrentHost` profile for Windows PowerShell ISE profile or the `AllUsersCurrentHost` profile for Windows PowerShell ISE.

Table 2-1 lists profiles that can be created and used in Windows PowerShell ISE. Each profile is saved to its own specific path.

TABLE 2-1 Profile paths for different profile types for Windows PowerShell

Profile Type	Profile Path
"Current user, Windows PowerShell ISE"	<i>\$profile.CurrentUserCurrentHost</i> , or <i>\$profile</i>
"All users, Windows PowerShell ISE"	<i>\$profile.AllUsersCurrentHost</i>
"Current user, All hosts"	<i>\$profile.CurrentUserAllHosts</i>
"All users, All hosts"	<i>\$profile.AllUsersAllHosts</i>

For example, assuming that all users use all hosts for Windows PowerShell for the SharePoint server, you would execute the following Windows PowerShell command in the Windows PowerShell ISE. Be aware that the following example applies to the "All users, All hosts" case, as noted in the *\$profile.AllUsersAllHosts* property of the *New-Item* cmdlet and *powerShell_ise.exe* command. You would change the property based upon the desired profile type:

```
if (!(Test-Path $profile.AllUsersAllHosts)) {  
    New-Item -Type file -Path $profile.AllUsersAllHosts -Force  
}  
powershell_ise $profile.AllUsersAllHosts
```

As shown in Figure 2-5, this will open another tab in ISE called *profile.ps1*, into which you would enter the following code and save the profile file:

```
$ver = $host | select version  
if ($ver.Version.Major -gt 1) {  
    $host.Runspace.ThreadOptions = "ReuseThread"  
}  
if ((Get-PSSnapin "Microsoft.SharePoint.PowerShell" -ErrorAction SilentlyContinue) -eq $null) {  
    Add-PSSnapin "Microsoft.SharePoint.PowerShell"  
}
```

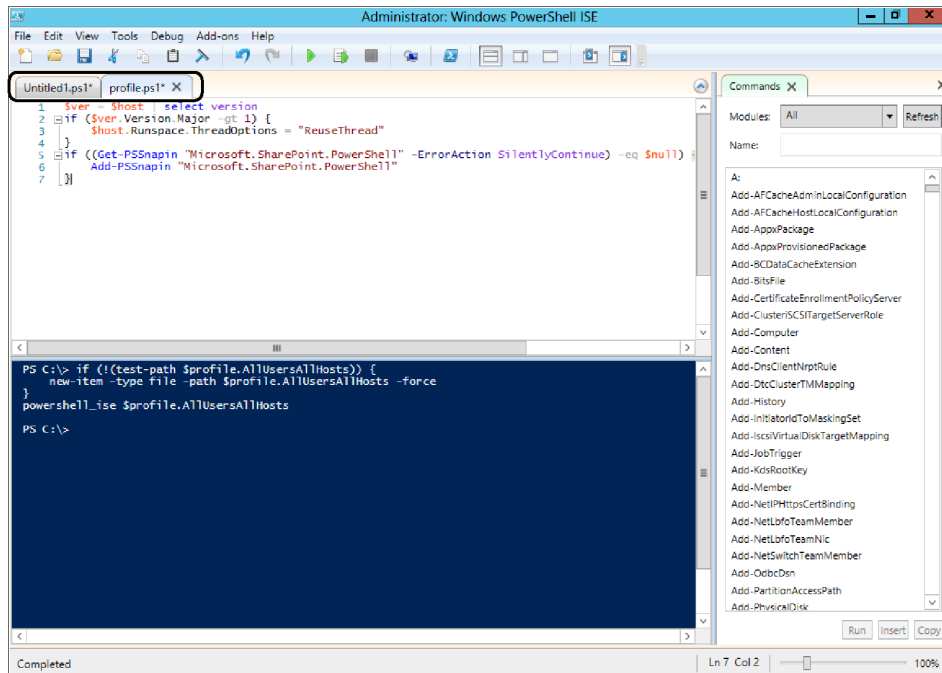


FIGURE 2-5 The process for setting up a default profile in ISE.



Note The SharePoint 2013 Management Shell and the Windows PowerShell console differ in the use of the *ReuseThread* option, which defines how the threading model is used. The use of the SharePoint 2013 Management Shell is defined by the line `{Host.Runspace.ThreadOptions = "ReuseThread"}`, which is in the *SharePoint.ps1* file.

Working with Windows PowerShell cmdlets

As noted previously, all Windows PowerShell cmdlets are verb-noun combinations. And, as one of the specifications for Windows PowerShell is for cmdlets to be discoverable, the most essential cmdlets are those that allow you to discover which cmdlets to use and how to use them.

First and foremost would be to find the appropriate cmdlet. The *Get-Command* cmdlet helps users accomplish this. With the additional optional parameters of *-Noun*, *-Module*, *-Syntax*, or others and the ability to use wildcards, the *Get-Command* can be extremely powerful in helping to find the right cmdlet. Additionally, if you are using the Windows PowerShell ISE, you can also use the Commands Add-On (see Figure 2-6), which by default is found on the right side of the Windows PowerShell ISE GUI.

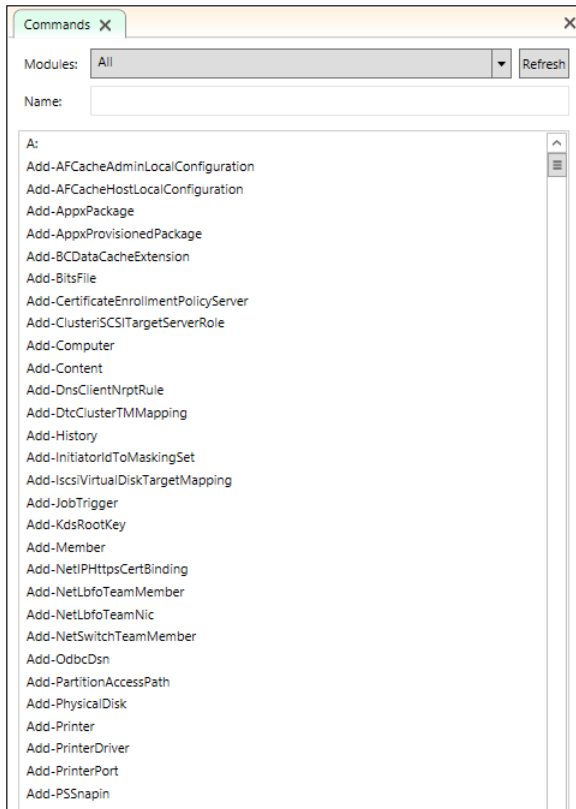


FIGURE 2-6 The default Commands Add-On for Windows PowerShell ISE.

Once the desired cmdlet is determined, use the *Get-Help* cmdlet and/or the *Show-Command* cmdlet. The *Get-Help* cmdlet will display the textual help for the cmdlet specified. The *Show-Command* cmdlet will show a graphical (GUI) window with both specific and common parameters for the command. Figure 2-7 is an example of the *Show-Command* window for the cmdlet.

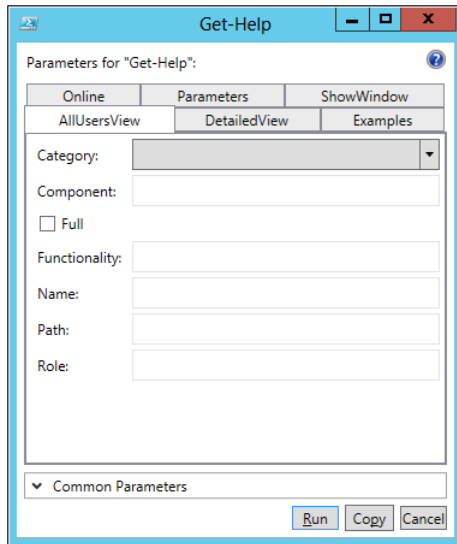


FIGURE 2-7 The GUI to set the parameter for the *Get-Help* cmdlet.

When using Windows PowerShell, you can keep the cmdlet help references up to date online if you wish. You will be prompted to do this when first using Windows PowerShell, but can always manually force an update using the *Update-Help* cmdlet in any Windows PowerShell shell.

As shown in Figure 2-8, once you have found the cmdlet that you desire, use the *Get-Command -Syntax* cmdlet/parameter combination to get just the syntax of the cmdlets.

```
PS C:\Windows\system32> Get-Command -Syntax Get-Help

Get-Help [[-Name] <string>] [-Path <string>] [-Category <string[]>] [-Component <string[]>]
[-Functionality <string[]>] [-Role <string[]>] [-Full] [<CommonParameters>]

Get-Help [[-Name] <string>] -Detailed [-Path <string>] [-Category <string[]>] [-Component
<string[]>] [-Functionality <string[]>] [-Role <string[]>] [<CommonParameters>]

Get-Help [[-Name] <string>] -Examples [-Path <string>] [-Category <string[]>] [-Component
<string[]>] [-Functionality <string[]>] [-Role <string[]>] [<CommonParameters>]

Get-Help [[-Name] <string>] -Parameter <string> [-Path <string>] [-Category <string[]>]
[-Component <string[]>] [-Functionality <string[]>] [-Role <string[]>] [<CommonParameters>]

Get-Help [[-Name] <string>] -Online [-Path <string>] [-Category <string[]>] [-Component
<string[]>] [-Functionality <string[]>] [-Role <string[]>] [<CommonParameters>]

Get-Help [[-Name] <string>] -ShowWindow [-Path <string>] [-Category <string[]>] [-Component
<string[]>] [-Functionality <string[]>] [-Role <string[]>] [<CommonParameters>]

PS C:\Windows\system32> |
```

FIGURE 2-8 An example of using the *Get-Command* together with the *-Syntax* parameter for *Get-Help*.

For more technical information about the cmdlet, use the *Get-Help -Detailed* parameter. For examples of how to use the cmdlet, use the *Get-Help -Examples* parameter, as shown in Figure 2-9. And finally, for the full help file, including all the information on syntax, technical information, and examples, use the *Get-Help -Full* cmdlet/parameter combination.

```

PS C:\> Get-Help Get-Help -Examples

NAME
    Get-Help

SYNOPSIS
    Displays information about Windows PowerShell commands and concepts.

----- EXAMPLE 1 -----
PS C:\> Get-Help

This command displays help about the Windows PowerShell help system.
----- EXAMPLE 2 -----
PS C:\> Get-Help *

This command displays a list of the available help topics.
----- EXAMPLE 3 -----
PS C:\> Get-Help Get-Alias
PS C:\> Help Get-Alias
PS C:\> Get-Alias -?

These commands display basic information about the Get-Alias cmdlet. The "Get-Help" and "-?" commands display the information on
a single page. The "Help" command displays the information one page at a time.
----- EXAMPLE 4 -----
PS C:\> Get-Help about.*

This command displays a list of the conceptual topics included in Windows PowerShell help. All of these topics begin with the
characters "about-". To display a particular help file, type "get-help <topic-name>", for example, "Get-Help about_Signing".

This command displays the conceptual topics only when the help files for those topics are installed on the computer. For
information about downloading and installing help files in Windows PowerShell 3.0, see Update-Help.
----- EXAMPLE 5 -----
The first command uses the Get-Help cmdlet to get help for the Get-Command cmdlet. Without help files, Get-Help display the
cmdlet name, syntax and alias of Get-Command, and prompts you to use the Update-Help cmdlet to get the newest help files.
PS C:\> Get-Help Get-Command

```

FIGURE 2-9 The output of the *Get-Help* examples after help has been updated.

Many times, you will see references in the help provided about the *PipeBind* object parameter (see Figure 2-10). As discussed earlier in the chapter, keep in mind that all Windows PowerShell cmdlets return objects. You can assign that object optionally to a Windows PowerShell variable composed of unique alphanumeric combinations preceded by a \$ symbol (for example, *\$webApp*). Additionally, you can “pipe” the output object of one cmdlet to the next cmdlet in an execution line using the pipe (|) symbol.

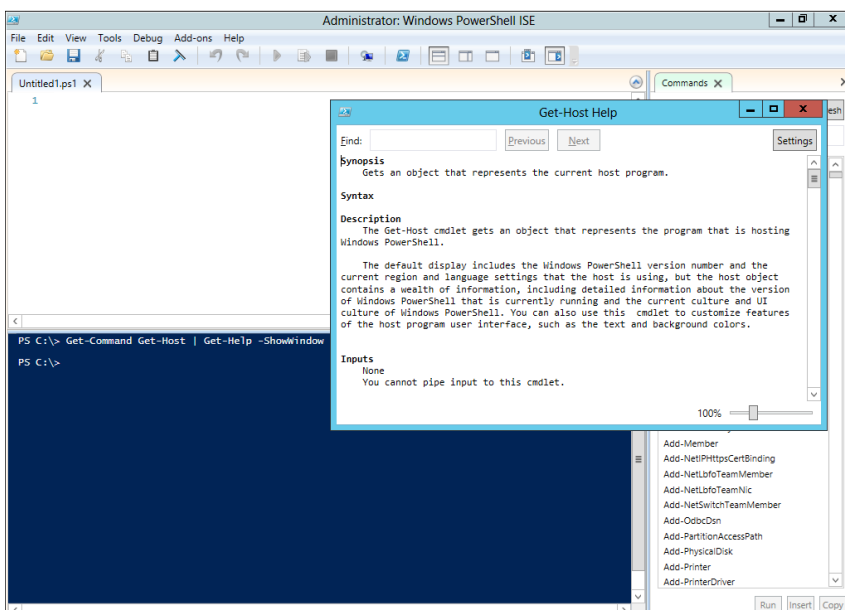


FIGURE 2-10 The *Get-Host* Help window within Windows PowerShell ISE.

Pipe binding allows you to pass output objects without creating variables when references to objects are necessary to return the desired object. If you believe that you may need a reference to an object later, you can assign it to a variable and then pipe in the variable. It is possible to pass the output object from the first cmdlet to the second, and then the output object from the second to the third, and so on, keeping in mind the object type for each cmdlet may and will probably be different. The key is to ensure the output object type of the cmdlet is the required input object type for the PipeBind object of the next cmdlet.

Discovering SharePoint-specific Windows PowerShell cmdlets

As discussed previously in the chapter, the *Microsoft.SharePoint.PowerShell.dll* file provides the library of almost all SharePoint-specific Windows PowerShell cmdlets. When the *Add-PSSnapin Microsoft.SharePoint.PowerShell* cmdlet is executed—either manually, by opening the SharePoint 2013 Management Shell, or by a *profile.ps1* file for the Windows PowerShell ISE, the references for the SharePoint cmdlets defined in the library are added to the shell environment. Using the *Get-Command* cmdlet with the *-Module* parameter, you can easily get the entire list of SharePoint-specific Windows PowerShell cmdlets, as shown in Figure 2-11. Additionally, you can use the Windows PowerShell ISE Command Add-On and limit the cmdlets in the window to the Microsoft.SharePoint.PowerShell module, as shown in Figure 2-12.

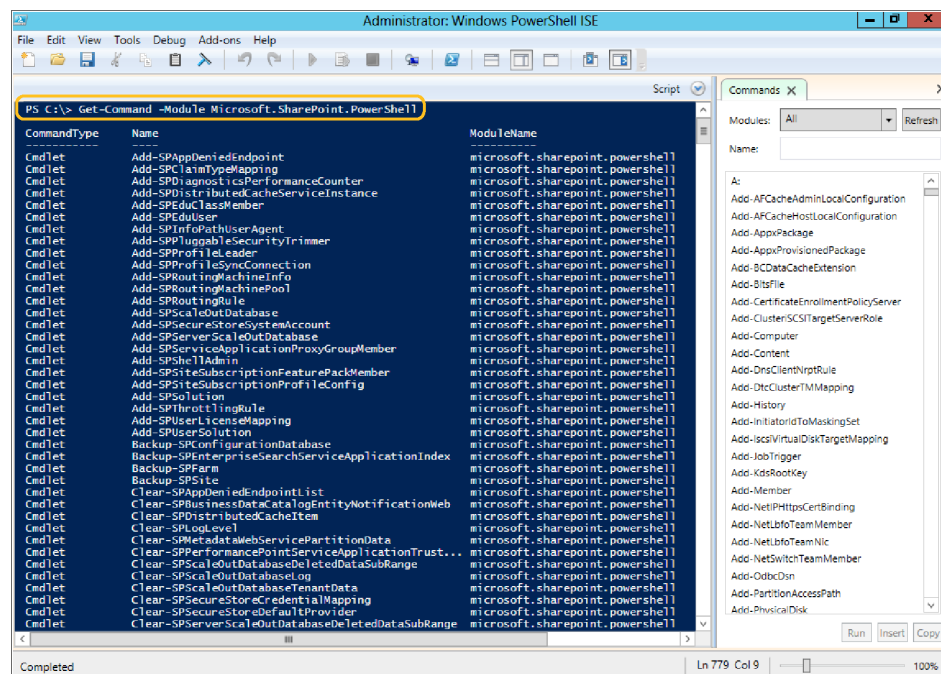


FIGURE 2-11 An example of the output of *Get-Command -Module* for Microsoft.SharePoint.PowerShell.

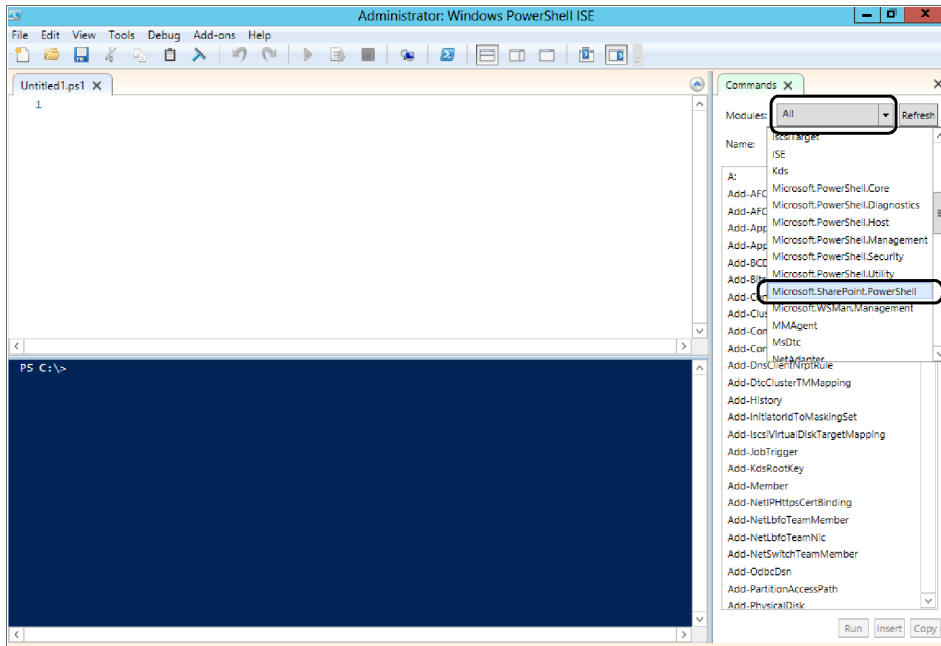


FIGURE 2-12 An example of how to find all of the SharePoint Windows PowerShell cmdlets through the ISE interface.

One of the most interesting tools now available to help with the building of Windows PowerShell scripts is Microsoft TechNet's Windows PowerShell for SharePoint Command Builder (<http://www.microsoft.com/resources/TechNet/en-us/Office/media/WindowsPowerShell/WindowsPowerShellCommandBuilder.html>). This interactive web-based design (see Figure 2-13) allows you to explore Windows PowerShell cmdlets for different SharePoint platforms, including Microsoft SharePoint Foundation 2010, SharePoint Server 2010, SharePoint Online, SharePoint Foundation 2013, SharePoint Server 2013, and Microsoft Office 365. Additionally, there are a few quick steps available for common functions. The design surface allows for the input of values that will provide the parameters for the cmdlets and, finally, copying to the Clipboard to allow quick pasting in the Windows PowerShell shell.

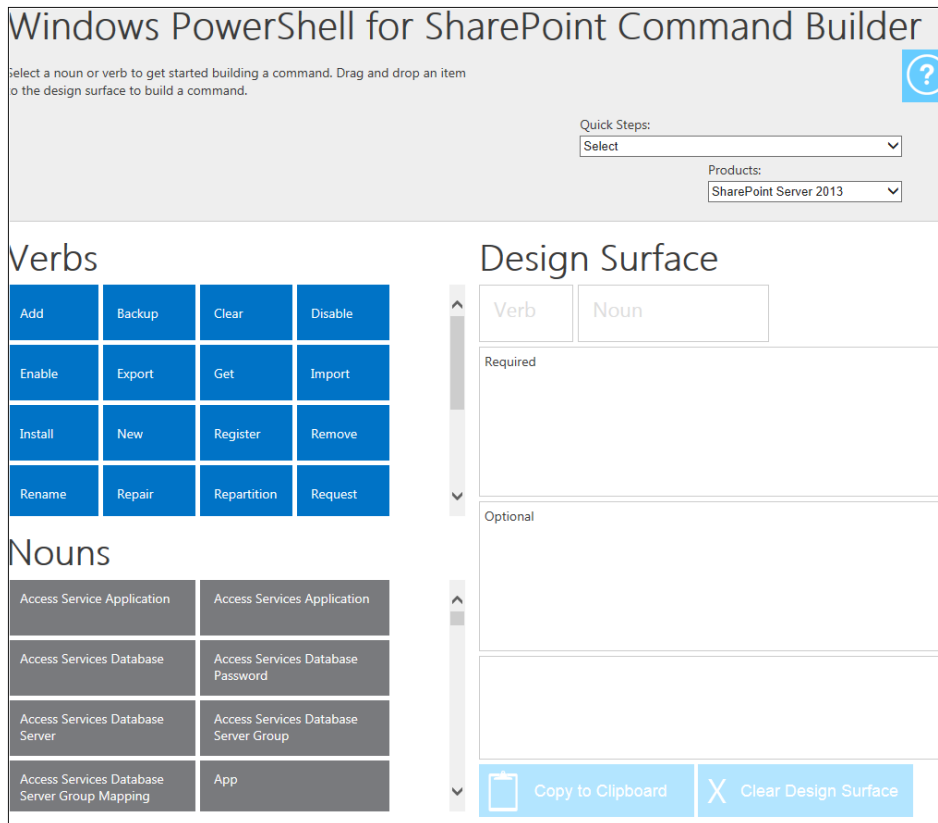


FIGURE 2-13 An image of the Windows PowerShell for SharePoint Command Builder on TechNet.

Another interesting website contains the TechNet documentation providing the index of Windows PowerShell cmdlets for SharePoint 2013 (<http://technet.microsoft.com/en-us/library/ff678226.aspx>). The online index provides all of the SharePoint 2013 cmdlets in both verb and noun order, as well as denoting the new cmdlets for SharePoint 2013 and links to the online documentation for each cmdlet, as shown in Figure 2-14. It is important to note that the cmdlets for on-premises and online are different and the SharePoint Online cmdlets are limited in their functionality, as opposed to the ones provided for on-premises.


The screenshot shows the TechNet website interface for the 'Index of Windows PowerShell cmdlets for SharePoint 2013'. The page includes a sidebar with navigation links, a main title, a summary, and a table of cmdlets.

Index of Windows PowerShell cmdlets for SharePoint 2013

SharePoint 2013 | Other Versions | This topic has not yet been rated - [Rate this topic](#)

Published: July 16, 2012

Summary: Lists Windows PowerShell cmdlets that are included with SharePoint 2013 in verb order and noun order.

Cmdlets that have the  icon next to them are new cmdlets in SharePoint 2013.

Applies to: [SharePoint Foundation 2013](#) | [SharePoint Server 2013](#)

- [SharePoint 2013 cmdlets in verb order](#)
- [SharePoint 2013 cmdlets in noun order](#)

SharePoint 2013 cmdlets in verb order








Cmdlet name	Description
Add-SPAppDeniedEndpoint 	Adds an endpoint to the Apps denied endpoint list.
Add-SPClaimTypeMapping 	Adds a claim mapping to a trusted security token service (STS) identity provider.
Add-SPDiagnosticsPerformanceCounter	Adds an instance of a performance counter.
Add-SPDistributedCacheServiceInstance 	Adds an instance of the distributed cache service to a local server.
Add-SPInfoPathUserAgent	Adds a user agent to a farm.
Add-SPProfileLeader	Adds a company leader.
Add-SPRoutingMachineInfo 	Adds a new routing target to the farm.
Add-SPRoutingMachinePool 	Adds a new machine pool.
Add-SPRoutingRule 	Adds a routing rule.
Add-SPScaleOutDatabase 	Adds an existing scale-out database to the specified service application.

FIGURE 2-14 The webpage from TechNet that has the index of SharePoint 2013 Windows PowerShell cmdlets.

The last tool that you should be aware of is the Microsoft TechNet Script Center (<http://gallery.technet.microsoft.com/scriptcenter>) which is not limited to SharePoint Windows PowerShell scripts; rather, it is a community-driven gallery that provides many Windows PowerShell scripts and *can* be a good place to learn additional scripting techniques. Keep in mind, though, that because anyone can upload scripts to the Script Center, the scripts available should still be examined and tested thoroughly before using.

Working with SharePoint disposable objects

It is important to remember, similar to programming with certain SharePoint objects, the need to dispose—that is, to explicitly call the *IDisposable* interface *Dispose()* method on those objects to release the memory allocation back to the system. Failing to do this in programming, or within Windows PowerShell scripts, can lead to memory leaks, which in turn can lead ultimately to system performance issues or even failure. When using Windows PowerShell to work with certain SharePoint objects, most commonly *SPSite* and *SPWeb* objects, you should ensure proper disposal of the objects.

This need for disposal arises from the use of any object that contains or references an *SPRequest* object. The *SPRequest* object contains a reference to an unmanaged COM object that handles communications with the database server. Any SharePoint or third-party application or Windows PowerShell script that calls upon any object with reference(s) to the *SPRequest* object should be examined to ensure proper disposal.

Windows PowerShell will automatically dispose of objects used in commands that appear on a single line or as part of a piped command. However, if you create a variable that stores a reference to an *SPSite* or *SPWeb* object, you will need to explicitly or manually dispose of the object when you are done working with the variable. Luckily, there are certain tricks, including a couple of important SharePoint cmdlets, to assist you in ensuring proper SharePoint disposable object disposal.

An example of a single line and piping that would not need the object's *Dispose()* method called is:

```
Get-SPWebApplication | Get-SPSite -limit all | ForEach-Object { Write-Host $_.Url }
```

However, as mentioned previously, when using an assigned variable to represent the SharePoint disposable object, you will need to explicitly call the *Dispose()* method like so:

```
$web = Get-SPWeb http://portal.tekfocus.com/it  
# Additional PowerShell script working with $web variable  
$web.Dispose()
```

Finally, arguably the easiest method is to simply wrap all SharePoint disposable objects using Windows PowerShell within the combination of a *Start-SPAssignment -Global* and a *Stop-SPAssignment -Global* cmdlet pair, as shown here:

```
Start-SPAssignment -Global  
# PowerShell script using SharePoint disposable objects  
Stop-SPAssignment -Global
```

Any objects defined between the *Start-* and *Stop-SPAssignment -Global* cmdlets will automatically have their *Dispose()* methods called.

Putting it all together

So far in this chapter, you have gained an understanding of the importance of Windows PowerShell in a SharePoint environment and how to ensure that you will be able to both use it from a permission standpoint and explore the hundreds of cmdlets that are available. While it only takes a little tinkering to get an intermediate level of understanding of the Windows PowerShell cmdlets for SharePoint, there are additional advanced tasks in SharePoint that benefit from the use of Windows PowerShell. Prior to diving into some advanced scenarios, you will first explore how to build a simple site. From there, you will get an understanding of how to configure one of the more complex service applications: SharePoint Search.

Creating a sample site (site collection)

In this first example, the key takeaway is the use of variables that are prefixed with the dollar sign (\$). While you don't always need to use variables, they do offer some readability to your scripts. Examine the following syntax. The use of SharePoint cmdlets doesn't happen until the final line, where you see the use of the *New-SPSite* cmdlet and a number of parameters that follow. The parameters are prefixed with the hyphen (-).

```
Add-PSSnapin Microsoft.SharePoint.PowerShell -EA 0
$siteURL = "http://portal"
$owner = "Contoso\Administrator"
$template = "BLANKINTERNET#0"
$name = "Portal"
$description = "This is a sample site that was built using PowerShell."
New-SPSite $siteURL -OwnerAlias $owner -name $name -Template $template -Description $description
```

SharePoint 2013 Search service application topology management

SharePoint 2013 uses the web-based interface within the SharePoint 2013 Central Administration web application to show the current status of the search topology. The primary way to change the topology is by using Windows PowerShell. This is a significant change from the previous version, as SharePoint Server 2010 also included a web-based option for changing the topology. The reason for this is that the core search architecture of SharePoint 2013 has a more complex and flexible topology that can be changed more efficiently by using Windows PowerShell. Also, keep in mind, the search capabilities of SharePoint Foundation 2013 have changed and are now based on the same search implementation as SharePoint Server 2013. This provides many improvements, but it also means that the search configuration is very different from the SharePoint 2010 Foundation but similar to SharePoint Server 2013 Search.

For specific information on changing the SharePoint Server 2013 Search service application topology, refer to <http://technet.microsoft.com/en-us/library/jj862356.aspx>, which contains references to Windows PowerShell to modify the SharePoint 2013 Search service application topology.

In this next example, you will see the use of comments. These are notes created by the author and are ignored by the system. These lines begin with the pound sign (#). The script may look a little more daunting than the last, but because of commenting, everything here should make sense to you.

```
Add-PSSnapin Microsoft.SharePoint.PowerShell -EA 0
# Define App Pool Name
$saAppPoolName = "SharePoint Web Services Default"
# Search Specifics, we are single server farm
# Define Search Server Name
$searchServerName = "Contoso-SP3"
# Define Search Application Name
$serviceAppName = "Contoso Search Service Application"
# Define Root for Search Database Names
$searchDBName = "Contoso_Search"
# Assign the Application Pool for Service Application Endpoint
$saAppPool = Get-SPServiceApplicationPool $saAppPoolName
# Start Search Service Instances
Write-Host "Starting Search Service Instances..."
Start-SPEnterpriseSearchServiceInstance $searchServerName
Start-SPEnterpriseSearchQueryAndSiteSettingsServiceInstance $searchServerName
# Create the Search Service Application
Write-Host "Creating Search Service Application and Proxy..."
$searchServiceApp = New-SPEnterpriseSearchServiceApplication -Name $serviceAppName
                    -ApplicationPool $saAppPoolName -DatabaseName $searchDBName
# Create the Search Service Proxy

$searchProxy = New-SPEnterpriseSearchServiceApplicationProxy -Name "$serviceAppName Proxy"
                    -SearchApplication $searchServiceApp
# Clone the default Topology (which is empty) and create a new one and then activate it
Write-Host "Configuring Search Component Topology..."
$clone = $searchServiceApp.ActiveTopology.Clone()
# Get the Search Instance Information
$searchServiceInstance = Get-SPEnterpriseSearchServiceInstance | where
    {$_ .Server.Address -eq $searchServerName}
# Create the Search Service Administration Component
New-SPEnterpriseSearchAdminComponent -SearchTopology $clone
    -SearchServiceInstance $searchServiceInstance
# Create the Search Service Content Processing Component
New-SPEnterpriseSearchContentProcessingComponent -SearchTopology $clone
    -SearchServiceInstance $searchServiceInstance
# Create the Search Service Analytics Processing Component
New-SPEnterpriseSearchAnalyticsProcessingComponent -SearchTopology $clone
    -SearchServiceInstance $searchServiceInstance
# Create the Search Service Crawler Component
New-SPEnterpriseSearchCrawlComponent -SearchTopology $clone
    -SearchServiceInstance $searchServiceInstance
# Create the Search Service Indexing Component
New-SPEnterpriseSearchIndexComponent -SearchTopology $clone
    -SearchServiceInstance $searchServiceInstance
# Create the Search Service Query Component
New-SPEnterpriseSearchQueryProcessingComponent -SearchTopology $clone
    -SearchServiceInstance $searchServiceInstance
# Activate Search
$clone.Activate()
Write-Host "Search Done!"
```

You can read more about the Search service and its components in the "Search service application" section of Chapter 4, "Understanding the service application model."

As you work with Windows PowerShell, you will undoubtedly need a deeper understanding of its capabilities. While this chapter has given you a good foundation, as well as enough insight into Windows PowerShell to be successful, there are still many elements that have been left out. There are many online references and books that can help you gain a deeper understanding of this material.

Gathering requirements

In this chapter, you will learn about:

- The importance of gathering requirements.
- Defining information architecture requirements.
- Mapping information architecture requirements to logical components.
- Designing the physical architecture.
- Mapping requirements to SharePoint Online and hybrid implementations.

Microsoft SharePoint has been and continues to be many things to many people. Even today, if someone asks, “What is SharePoint?” each person has a unique idea of what the product can and cannot do. SharePoint is and always will be a platform to be implemented to meet the needs of the organization that it is meant to serve. As such, arguably the most important task of a successful implementation, is to gather all of the requirements that reflect the business goals of the organization to ensure the maximum value is attained with the implementation of the SharePoint 2013 platform. Ultimately, the reason that everyone seems to have a different interpretation of what SharePoint is, is that a successful implementation should be molded to the needs of each different organization. Therefore, it is important for you, as its designer, to ask the appropriate questions to ensure that you are building a solution that satisfies organizational goals and visions while providing quantifiable value to the people using SharePoint.

In this chapter, you will gain an understanding of the importance of gathering requirements and then take a high-level view of the core components of the product. You will then take those requirements and map them to the logical and physical architectures to ensure a successful solution. Finally, you will examine what options you have with the new Microsoft SharePoint Online offering and discuss possible hybrid solutions. The final section, “Putting it all together,” includes some example questions that you can add to your arsenal to help ensure the success of your implementation.

Importance of gathering requirements

At the very basic level of explanations, requirement gathering should be no different than what you typically do in organizing your personal and professional lives. In your life, you may set goals and determine the plan for accomplishing those goals. Typically, the more detailed the goals and plans for accomplishing those goals are, the better chance you have at being successful at meeting those goals. The approach to implementing SharePoint 2013 is no different.

SharePoint 2013 provides many capabilities to organizations and users. Unfortunately, many organizations either attempt to provide too many, if not all, of the capabilities of the platform, or too few, approaching the implementation with little or no clearly defined expectations. The majority of this book covers the technical elements of an implementation and configuration of SharePoint. The skills necessary for a successful gathering of requirements encompasses not only a deep understanding of the technical capabilities of SharePoint 2013, but also a thorough insight into the organization, its functions, and its goals, both tactical and strategic.

Why gather requirements?

All stakeholders of the SharePoint 2013 solution should understand and be able to communicate the importance of the requirement gathering process to both other stakeholders and, ultimately, the users of the proposed solution. Although seemingly based on common sense, you may run across the situation where the organization has decided to proceed with an implementation without a clear understanding of what the implementation is meant to provide to the organization. This leads to frustration with the solution, and possibly to an abandonment of the solution by the very users that the platform is meant to assist. Organizations that have implemented previous SharePoint environments and other enterprise systems without a thorough understanding of the capabilities and value that the platform provides to the users have found many different challenges throughout the lifecycle of the solution, including:

- Failure of implementing relevant enterprise metadata and content types
- SharePoint site “sprawl” or unruly, difficult-to-manage site structure
- Inability to implement future solutions due to initial configuration decisions and actions

An implementation project that fails to properly gather and understand the requirements may cause a “rip-and-replace” reimplementation of the SharePoint platform. The effort, time, and cost of the initial nonrequirement-focused deployment would be, at the very least, wasted.

When to gather requirements

The primary deployment stage within which to gather the requirements is planning. When you finish the planning stage, you should have documented the following:

- An infrastructure design to support your solution
- A detailed description of how you will implement the solution

- A plan for testing and validating the solution
- A site and solution architecture
- An understanding of the monitoring and sustained engineering requirements to support the solution
- A record of how the solution will be governed
- An understanding of how the solution will be messaged to the organization's users to drive adoption of the solution

Be aware, however, as you move through the future deployment stages of development, proof of concept, pilot, user acceptance testing, and, finally, production, you will likely update your requirements and plans. Like most planning, the important thing to remember is to do the *right amount* of planning. Too little can create significant additional work, consume unbudgeted resources, and/or ultimately detract from the overall success of the implementation. At the same time, too much planning can obviously take away time and resources up front, but also prevent corrections during a full deployment cycle.

Planning for success

Initially, the key question to be answered is, "Why are we implementing SharePoint 2013?" A clear, concise answer is the basis for a successful implementation. To that end, it is a question that can rarely be answered solely by the organization's IT professionals. The most valid answer will come from the organizational leaders and the interpretation of their understanding of the organization's goals and objectives. As such, it is critical to identify the individuals who are responsible for the business objectives, especially those to be addressed by a SharePoint deployment. These individuals will typically be identified by project management as key stakeholders. Additionally, more detailed business objectives to be addressed may come from key individuals within specific business or organizational units. The information gathered from these stakeholders is not only the basis for the implementation, but can also provide the initial information to be documented in the governance and information architecture deliverables. Finally, some, if not all, of these stakeholders should also be identified as potential members of the ongoing governance team or board to ensure that the solution continues to meet the needs of the organization throughout its lifecycle.

Metrics for success

If the process for gathering requirements is simply setting goals for the platform, how will you measure the business or organizational success of the SharePoint 2013 implementation? This question must be addressed and clearly answered to ensure that all stakeholders know when the goals of the project have been met. Even if the technology is close to perfect, the platform is not successful unless it has a positive impact on the organization's goals.

Although ideally it is best to use quantifiable metrics, an information management, collaboration, or content management system is difficult to quantify for users. Therefore, look for examples from users and/or other stakeholders to provide the measurable value statements. A specific example

with a simple quantitative statement of organizational value at the end can have significant impact and help to measure the success of the project. For instance, a user could say that for a specific task, instead of the previous two days of effort, the task was completed in four hours. Quantitatively, this translates to a reduction of 75 percent effort to accomplish the task. If the task is repetitive and/or completed by multiple users, this can be easily translated to a cost benefit. If 10 users did the same task monthly, with a savings of 12 hours per task execution, the organization could claim the savings of 120 hours each month *for that task alone!* The features offered by SharePoint 2013 are vast, and no one organization will probably use all of them. Implementing SharePoint in an organization may cause people to examine their processes and find better ways of implementing everyday tasks. With the abilities of SharePoint Search, using metadata for documents instead of folders, or even presenting people with electronic workflows, can save time and money.



Important One of the most critical questions to ask is, “What will determine the success of the project?”

Identifying stakeholders

Once the decision has been made to implement SharePoint, the stakeholders have a number of expectations. Whether formally or informally, there is a vision as to what the outcome of deploying SharePoint 2013 will provide to the organization. However, since SharePoint 2013 is meant to empower all users with access to information, and due to the fact that SharePoint 2013 integrates or connects to so many other systems, there are many people and/or groups of people who will have an impact on the success of the implementation. Identifying the individuals or stakeholders having an impact on the successful implementation is the first step in the process of gathering requirements.

In most organizations, the IT department will lead the technical implementation. However, IT is not necessarily fully aware of the organization’s goals. In a worst-case scenario of a deployment of an IT product or technology, IT will simply deploy it without guidance from the organization. You may have seen examples where IT has approached a product or technology with a “build it and they will come” mentality. With SharePoint 2013, this approach is extremely dangerous to the overall success of the deployment. As with most business- and/or user-centric solutions, user adoption is paramount to the success of the project. You should always keep in mind that the role of IT is to support the organization and its goals. You could deploy the perfect technical solution, but if the users do not adopt and use it, the project is a failure. However, a successful SharePoint 2013 implementation is still very much dependent on IT.

Who are the other stakeholders? As with most things in a SharePoint 2013 implementation, it depends. It depends on the purpose of the platform. For example, in deploying a self-service portal for company and personnel information, typically the key executive stakeholder will be the corporate communications director and/or the human resources (HR) director. On the other hand, a customer-focused extranet portal may have the chief marketing officer or similar person as its key stakeholder. Each stakeholder may have his or her own agenda and have a number of requirements that need to be met. Typically, SharePoint may not be successful when those needs are captured. A

critical error in deploying SharePoint 2013 is attempting to enable all the functionality and features that come with the product in the first iteration of deployment and expecting that it will satisfy the needs of the stakeholders.

Finally, as mentioned earlier, expect the platform to increase functionality over time, preferably in an iterative or phased approach. Therefore, remember that for each phase or iteration, the group of stakeholders may change. Review Table 3-1 for the most common stakeholders that participate in a project.

TABLE 3-1 Potential SharePoint implementation project stakeholders

Stakeholder	Requirements Input
Executive sponsor(s)	Provide the goal/vision of the platform
Content providers	Provide type(s) of information to be made available on the platform
Users	Provide input to ensure that the solution addresses the needs of users
IT department	Provide guidance in complying with IT policies and guidelines, as well as support for connected and/or integrated systems
SharePoint project team	Provide expertise on the platform capabilities, limits, and ease of solution implementation

The more stakeholders there are, the more information you can gather to provide detailed requirements, as well as ensure comprehensive coverage of important objectives. Additionally, you will gather more support for the implementation. Keep in mind that stakeholder and user support is arguably the most important aspect of any SharePoint 2013 deployment.

The inclusion of employees who may not have official titles but are influencers within the organization can help evangelize the solution. These *champions* can be influential in the organization due to how much they're respected. They can be essential in making an implementation successful.

Organizational objectives

The organizational objectives are the goals of the project put in the perspective of the organizational goals. The organizational or business objectives of the SharePoint 2013 solution should be clearly defined, documented, and measurable. Every requirement should have traceability to the objectives of the system. In fact, if a requirement does not have traceability to the objectives, it should be postponed for a later iteration or phase of the solution or removed altogether. Similarly, the implementation's goal should connect to the organizational mission or goal. This simply defines the SharePoint 2013 project's value to the organization.

In addition to organizational objectives, by deploying SharePoint 2013 within an organization, there can be additional value to the organization. These, too, should have traceability to organizational goals. Some of the additional organizational benefits can include:

- Ensuring less time to find and access information that organizational members need to accomplish their work
- Ensuring less time to find and utilize others' skills and expertise

- Consolidation of organizational classification of information
- Minimizing onboarding time for new members until they are productive
- Improving customer service and/or partner collaboration by providing access to internal organizational information

Prioritizing goals

In implementing a platform based on SharePoint 2013, scope can rapidly become an issue. As such, the first step for successful implementations is to prioritize the objectives to be addressed by the system. Ideally, it is best to identify three to five primary capabilities of the platform that will provide the most organizational impact and ensure that they are implemented extremely well. Ensure that the expectations of the platform are clear and communicated to all stakeholders and users early in the project lifecycle. Also, try to keep the complexity of the solution to a minimum by relying on SharePoint 2013 native functionality. By staying within the native functionality boundaries of the platform, especially during the initial iteration cycle, the project schedule and cost will be minimized, furthering the probability of success. During interviews with stakeholders, use the time to explain the out-of-the-box features and receive feedback about whether the native functionality will be acceptable to the user. Finally, ensure that the vision of the solution, not just the initial iteration, is communicated. This is even more critical in the initial iteration of implementation of the SharePoint 2013 platform.

Mapping objectives to functionality

Once the proposed system's objectives (goals) have been defined, the next step is to map the identified, prioritized objectives to SharePoint 2013 functionalities. This is one of the most important responsibilities of the SharePoint 2013 project team and administrators, as a broad and deep knowledge of SharePoint 2013 is critical to accomplishing this successfully. For example, an objective that ensures less time to find and utilize others' skills and expertise could depend, and therefore be mapped to, the native SharePoint 2013 features of user profile search, My Sites, newsfeeds, community sites, and blogs. Another example would be the objective of improving customer service and/or partner collaboration by providing access to internal organizational information. This requirement would map to SharePoint 2013 functionalities of extranet support, web content management, Business Connectivity Services (BCS), and/or mobile access.



Note Implementing a tool should not dictate a process, and the tools selected to enable a process should not dictate the requirements. It is important to remember the requirements first and the technology second.

Deliverables

The requirements-gathering activities will contribute significant key artifacts for the implementation project. Different groups of stakeholders and individuals will have an impact on different deliverables. Optimally, the project team will gain this information from the stakeholder groups. The best method

is by interview, capturing the respective information from each group. However, it may be difficult and too cost and time prohibitive to engage every stakeholder group in personal or group interviews. At the same time, communication solely via email can be less informative and limiting to both the stakeholders and the project team. Be flexible in the communication and information-gathering process.

At a minimum, you should deliver usage scenarios, functional requirements, nonfunctional requirements, taxonomy and metadata, content types that will be implemented, and site structure, and then work with them in defining the governance plan and documentation.

Usage scenarios

You should compile a list of usage scenarios that will capture the most common-use cases of the system. Additionally, the list should be prioritized, based on the commonality and criticality of the scenario. The usage scenarios will also provide additional information and basis for governance documentation, user documentation, and training. There is no dependency on other deliverables.

Key stakeholders All users of the system

Example A user will access a file from the archive (file share), make edits, and upload to SharePoint, providing metadata for the document.

Functional requirements

Functional requirements are derived from the usage scenarios and therefore are dependent on the completion of the prioritized list of usage scenarios. The list of functional requirements should also be prioritized based on criticality to business impact, commonality, and ease of implementation.

Key stakeholders Executive sponsorship, business analysts, and SharePoint project team

Secondary stakeholders System and application administrators of integrated or connected systems

Example HR information requires a unique retention policy.

Nonfunctional requirements

Nonfunctional requirements are those that do not affect the functionality of the system, but are typically affected by policies and procedures of the organization. These can include organization- or IT-specific guidelines for IT systems such as security, high availability, capacity considerations, etc. Since nonfunctional requirements are not strongly tied to usage scenarios, the gathering of these requirements can be done in parallel with other requirement deliverables.

Key stakeholders IT management and SharePoint project team

Secondary stakeholders System and application administrators of integrated or connected systems

Example All data must be secured via encryption—at rest and during transmission.

Taxonomy and metadata

In SharePoint 2013, the classification of the information stored within the system is more critical than prior versions. Additionally, the classification or taxonomy of the information will typically occur at multiple levels. In a simple organization, this could be implemented via a two-tier system of enterprise- or organization-wide classification and department classification. In more complex and larger organizations, it could have several tiers of classification. While gathering requirements and usage scenarios from users, the primary question to be asked in determining taxonomies will be, “How do you organize your data?” You will find individuals with the same role and/or responsibilities organizing the same type of data differently. With a well-designed taxonomy and metadata, SharePoint 2013 provides the ability for users to find the information as they desire, rather than by a fixed structure. At the same time, this ability can initially be challenging to users who have only dealt with file shares and folder structures. Most users coming from a simple folder structure have used the folder names and/or file names to provide the classification with which they feel most comfortable. At a minimum, the taxonomy and metadata will provide input to content types and site structure and, optionally, the configuration of the Managed Metadata Service (MMS) application of the platform.

Key stakeholders Users and business analysts

Secondary stakeholders SharePoint project team

Example All documents in the organization must be identified with office location, department, and sensitivity classification.

Content types

Once information types and classification are determined from both usage scenarios and taxonomy and metadata, the content types and hierarchical level of the types can be defined and planned. Similar to taxonomy and metadata and, in fact, so closely related, the content type structure and level at which they are implemented can be at the organization, department, and or specific need level(s). Additionally, this may affect configuration of one or more MMS applications and their corresponding enterprise content type hubs.

Key stakeholders Business analysts, SharePoint project team, and administrators

Example Since all documents need to include the office location, department, and sensitivity classification, an enterprise content type will be created as the base document content type for the organization.

Site structure

As mentioned previously, SharePoint 2013 is more reliant on classification of the information than the storage or site structure of the platform. However, performance and system constraints will still affect the overall site structure for a successful deployment. The taxonomy and content type hierarchies will typically have the most impact on the site structure. The best way to describe the importance and relationship of classification and structure is to remember users will only find and use information in one of two ways if they do not already have the location of the information—either via Search or

navigable site structure. Therefore, even if the SharePoint 2013 implementation is more reliant than ever on Search via metadata navigation and/or content, site structure is still just as critical.

One important thing to consider is how governance will dictate what users can and cannot do when creating new sites. You probably want to avoid situations where users are allowed to create sites that change the overall structure of the site.

Key stakeholders SharePoint project team, administrators, and business analysts

Example While there are a number of ways to organize sites within a specific organization, review Figure 3-1 as an example of how sites may be broken up functionally.

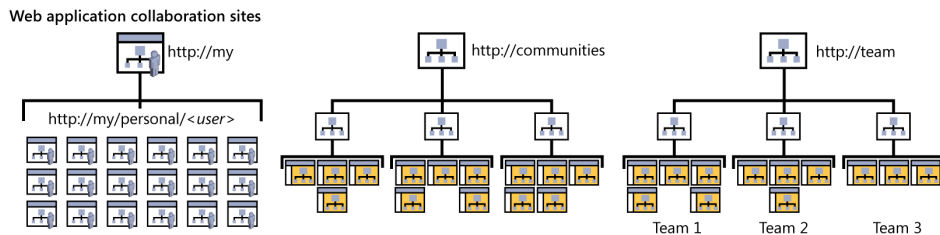


FIGURE 3-1 Site structures should be broken up in functional ways.

Governance plan and documentation

During the discovery of the organizational objectives that the solution is addressing, the initial governance plan should also be considered. Governance will be “living,” in that it will and should be constantly changing as the solution matures. It is also essential in keeping the boundaries of use by keeping the platform manageable. It can also minimize the amount of configuration by SharePoint administrators to maintain the governance of the system. This is similar to having a policy within the organization not to browse inappropriate Internet sites. IT could create, administer, and maintain a block and/or accept a list of appropriate Internet sites, but the additional overhead could be significant. Rather, by communicating to all users via an Internet usage policy (governance), the organization can minimize the risk and cost of enforcement. This should be the same approach for SharePoint 2013 and good communication mechanisms of the proper use and purpose of the platform. In the final section of this chapter, you will be presented with a number of sample questions that will help determine the governance needs of an organization.

Key stakeholders Business analysts, SharePoint project team and administrators, and the IT department

Example All organizational files to be shared with others will be placed in the SharePoint portal.

Information architecture

Information architecture (IA) has several definitions. It is the structural design of shared information environments. For the intranet and websites, it can be considered the combination of organization, labeling, search, and navigation systems. It is both an art and a science of organizing information within software to support findability and usability. Successful IA addresses all three of the following information access scenarios:

- The user knows where the information is located.
- The user doesn't know where the information is located.
- The user doesn't know if the information exists.

Ignoring the first scenario, which is obvious, the platform provides access to relevant information in a combination of two ways; search and navigable structure.

The enterprise search capabilities native to SharePoint products and technologies has been in the platform for the last several versions. With the new consolidated features of the SharePoint 2013 Search engine, which could be easily be described as a combination of SharePoint Server 2010 Search and FAST Search for SharePoint 2010, the native search capabilities have become more powerful.

The site structure will define the primary navigable structure. However, again, there are improvements within SharePoint 2013 for navigation, allowing for dynamic navigation based on taxonomy. Implementing metadata or taxonomic navigation can be a very powerful tool for the findability and usability of the SharePoint 2013 environment. However, the dependency for successful implementation is a well-defined and planned taxonomy. Additionally, expect difficulty in user adoption of this type of navigation. Most users will have difficulty in moving from a single dimension folder-based structure of files and/or sites to a dynamic presentation of navigation. Additionally, if the information is not "tagged" with the proper taxonomic properties, the navigation structure can fail the organization.

All of the deliverables of the requirement-gathering phase are necessary to complete the IA. However, since you are primarily concentrated on search and navigability, the most important deliverables will be site structure, taxonomy, content types, and usage scenarios. However, functional and nonfunctional requirements, as well as governance, can have an impact on the final IA. Thus, your primary task will be to concentrate on site architecture and taxonomy or metadata architecture.

Benefits of IA

The stakeholders who have made the investment in the SharePoint 2013 environment will expect an intuitive platform for users to find the information being exposed by the solution. Additionally, good IA can have additional benefits, including:

- Reducing IT workload and costs by identifying and potentially removing duplicate or redundant information within the organization.

- Improving user experience and, therefore, adoption by providing intuitive, navigable structure(s) that reduce the amount of time to locate information that users need. As well, users will feel more positive about the system as a whole, driving even more user adoption.
- Minimizing risk of compliance of both internal and external regulations, policies, and procedures.

Site architecture

The site architecture will provide the relationship of the information in SharePoint and, optionally, outside the platform. It will show the relationship between web applications, site collections, sites and, optionally, libraries within the sites. The initial site architecture will typically start with the IA deliverable. Most organizations will be tempted to begin with organizational structure–based site architecture. Although this may benefit smaller, departmental organizations, consider the site architecture options based on usage or flow of information. The primary factors that influence site architecture are:

- **Content responsibility** Who is creating the content? Who is managing the content? For example, the HR department is responsible for all human resources–related information on the platform.
- **Content security** If there are specific security requirements for the information, this may cause a separate site hierarchy for the content.
- **Content size** The amount of information and/or content within a site hierarchy can have an impact on the system as a whole and influence the site architecture.

You should also minimize the depth of the site architecture hierarchy. If a user needs to navigate seven levels to access information, the number of links or clicks can be frustrating to the user. Depending on the use of SharePoint 2013 social features and Office 2013 SkyDrive Pro, this can be minimized for common information. However, users will need training and/or communication surrounding this new functionality.

Some of the other topics that should be examined may include the use of publishing sites versus team sites. While governance may dictate who will modify or review content, the features will need to be determined during the planning phases.

Metadata architecture

Metadata architecture is as critical, if not more critical, than site architecture. It allows for the classification and organization of information in many different ways. Additionally, many of the enhanced SharePoint 2013 IA features rely on good metadata or information descriptors. It can affect relevant search, findability, and organizational compliance, to name a few.

Basically, metadata architecture is the combination of content types, local metadata (columns), and organizational metadata (managed metadata).

Content types are, in fact, primarily defined by the metadata used to define the type of information. The metadata for content types will either be defined at the site (collection) level or at the enterprise (metadata managed) level. With the MMS application, first introduced in SharePoint 2010, you have the ability to manage not only enterprise metadata, but also enterprise content types (that is, content types used across site collections, web applications, and even SharePoint farms). Content types can therefore be defined at the site (collection) level or across the enterprise.

Local metadata consists of two different types of columns: site columns and list columns. Site columns are defined at the site collection level and can be used to help define local (site) content types or be used throughout lists and libraries within the site hierarchy. List columns are defined and only available within the list or library where they are defined and configured.

Managed metadata is metadata that is managed centrally in a SharePoint 2013 MMS application, and optionally available both inside and outside the SharePoint 2013 farm.

Finally, there are very strong advantages of well-designed content type and metadata hierarchies. Since the metadata architecture is hierarchical, changes at a higher level can be optionally passed to lower levels of the hierarchy. This inheritance provides a significant benefit of ease to management and administration for changes in the taxonomy.

User challenges

Arguably, the most difficult part of applying the metadata architecture will be communicating and/or training the users on the benefits and proper use of metadata. Folders, the classic IA cornerstone in legacy information management systems, will still be strongly engrained as the method for organizing information. The classic folder structure was, in essence, the classification of information by location only. However, this structure presents the following problems:

- Folders do not allow for dynamic views of your content.
- Folder structures are fairly immutable or difficult to change.
- Folders assume that everyone accessing the information organizes information in the same manner.
- Folder structures lead to duplicate content, as users may put content in two folders since the content applies to both folders.

Since users will need to change the way they work with information by adopting the process of applying metadata information, it is also extremely critical to communicate the benefits of using metadata to organize information.

Logical architecture

After the completion of the IA, the process moves to creating the logical architecture to support that IA. The logical architecture simply ensures that all the services, service applications, and external systems needed for the IA are available and documented for the implementation of the environment. In a conceptual view, one could imagine the logical architecture as the listing and detailing of all

of the building blocks required for the implementation, both internal and external, to the entire SharePoint deployment.

Typically, it is easiest to start by listing the service applications needed, and then understanding and mapping the communication pipelines between the components and/or services listed. Once this is complete, the architect will easily move into the physical architecture and capacity planning of the physical servers to host the services (components) of the complete system. Figure 3-2 illustrates an example of a logical architecture with a dedicated enterprise search farm.

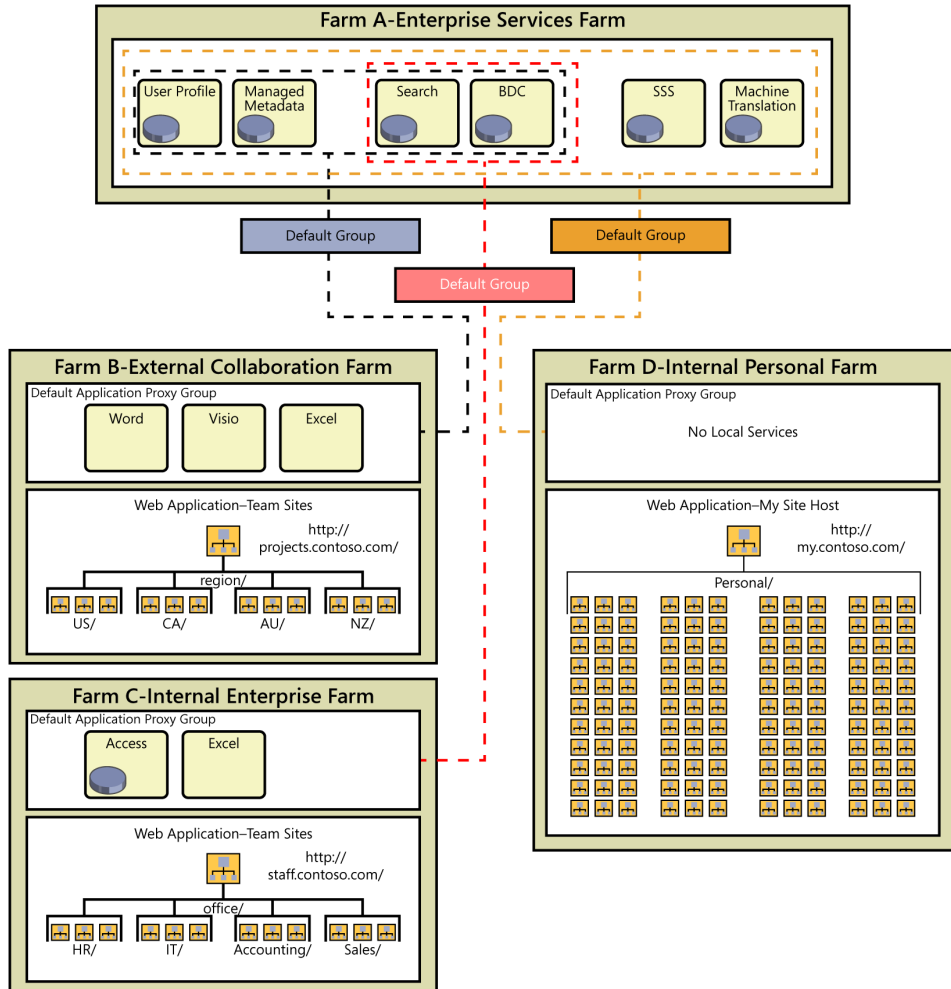


FIGURE 3-2 The diagram illustrates a logical architecture.

Physical architecture

With the logical architecture complete, the next step is to map the logical components (services) to physical servers. Even if the servers are virtualized using Hyper-V or VMWare, the virtual machines are considered, for this phase of design, physical servers. However, the virtual physical hosts need to be considered, if not by the SharePoint technical team, by the administrators managing the virtual environments. It is strongly recommended that virtual guests—physical servers of the SharePoint deployment—utilize dedicated resources such as virtual cores and RAM. Additionally, other factors, such as if and when to utilize hyper-threading (allowing for more CPU cores at a cost of overall CPU capability) or ensuring non-uniform memory access (NUMA) boundaries for RAM, should be considered by the technical architects and/or administrators of the virtual environment.

The physical architecture, as shown in Figure 3-3, will typically show the physical servers, with the services allocated to each server and communication pipelines between the servers.

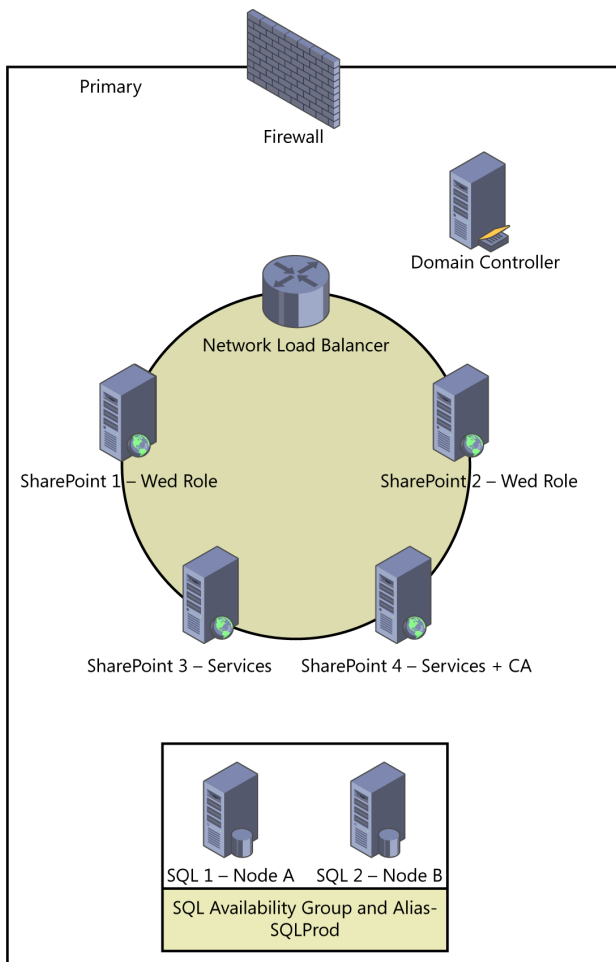


FIGURE 3-3 This diagram illustrates an example of the physical architecture.

Understanding system requirements

SharePoint 2013 supports several installation scenarios. At release, these include the following:

- Single server with a built-in database or single server that uses Microsoft SQL Server
This scenario should only be used for development and/or evaluation installations of Microsoft SharePoint Foundation 2013.
- Single-server farm installation with built-in database or single server that uses SQL Server
This scenario is optimal for development and/or evaluation installations of SharePoint Server 2013.
- Multiple server farm installation
This will be the most common scenario for pilot, user acceptance testing, and/or production deployments of SharePoint Server 2013.

Minimum hardware and software requirements

As the SharePoint platform and its capabilities have increased from previous versions, the minimum hardware requirements have also increased from previous versions. In planning, the technical architect will need to keep these in mind when developing the implementation plan. However, as always, these system requirements should not drive the implementation. The organizational requirements, and therefore the solution requirements, should always have priority over technical requirements.

For the latest guidance on the technical requirements for SharePoint, review <http://technet.microsoft.com/en-us/library/cc262485.aspx>.

For all of the scenarios listed on the provided resource, the hard disk space for the system drive only provides for the base installation. You must plan for and provide sufficient space for diagnostics including logging, debugging, and other requirements. In production, you must have additional space for other operations. You must maintain, at a *minimum*, two times as much free space as you have RAM. So additional requirements may include:

- Drive space of five times RAM.
- Distributed cache RAM requirements.
- Workflow server farm.
- Office Web Apps farm.
- Search in SharePoint 2013 requires more resources than in the previous version.
- A single-server farm with a number of service applications will require 24 GB of memory.

All servers that belong to a server farm, including database servers, must physically reside in the same datacenter. Redundancy and failover between closely located datacenters that are configured

as a single farm (“stretched farm”) are not supported in SharePoint 2013, which is a significant change from the previous version.

Most production environments will consist of an implementation based on the multiple server three-tier farm. In this scenario, separate, dedicated database server(s) will be required.

It is important to keep in mind the SharePoint 2013 software boundaries and limits. Expect them to change during the lifecycle of the SharePoint 2013 products and technologies. For more information, visit <http://technet.microsoft.com/en-us/library/cc262971.aspx>.

Keep in mind that the software and hardware listed above are the *minimum* requirements. Capacity planning, especially from the database perspective, is covered in Chapter 5, “Designing for SharePoint’s storage requirements.”

SharePoint Online and hybrid architecture considerations

With Wave 15, the emphasis on SharePoint Online is clear. With Microsoft offering a “to the cloud” message and the public release and availability of the new version of Office 365 in February 2013, many organizations are now examining the SharePoint Online platform as an option to provide SharePoint 2013 capabilities to their organizations. Of course, the first and main thing to consider when examining SharePoint Online as an option for deployment is to examine requirements. As stated previously in this chapter, requirements are sometimes very difficult to keep as the goal of all deployments.

When to consider SharePoint Online or hybrid architecture

With either a sole SharePoint Online deployment or a hybrid architecture approach to SharePoint, there are many benefits to considering the cloud-hosted model. Additionally, although SharePoint Online is available as a stand-alone, purchasable Software as a Service (SaaS) model, most organizations will move directly to Office 365, which will include one of the two SharePoint Online plans: Plan 1, sometimes referred to as SharePoint Online Standard, or Plan 2, sometimes referred to as SharePoint Online Enterprise. To map to Office 365, SharePoint Online Plan 2 is currently available in Office 365 Plans E1, E3, and E4. However, one of the benefits of SharePoint Online is that the feature set can grow fairly rapidly, especially considering Microsoft communications concerning “rolling releases” of software updates and enhancements to the Microsoft Cloud offerings.

Again, the first things to consider are the features available to organizations from all possible deployment scenarios, especially between on-premises (on-prem) and SharePoint Online. Review Figure 3-4 for an overview of the features and the offerings they map to.

Functional Area	Feature	SharePoint Server Standard	SharePoint Server Enterprise	SharePoint Online (Plan 1)	SharePoint Online (Plan 2)
Apps	App Catalog and Marketplace	New	New	New	New
	Team Sites	•	•	•	•
Collaboration	Work Management	New	New	New	New
	Social	•	•	•	•
	External Sharing			New	New
	Basic Search	•	•	•	•
Search	Standard Search	•	•	•	•
	Enterprise Search		•		•
	Content Management	•	•	•	•
Content Management	Records Management	•	•	•	•
	E-discovery, ACM, Compliance		New		New
Business Intelligence	Microsoft Excel Services, PowerPivot, PowerView		New		New
	Scorecards & Dashboards		•		
	Microsoft Access Services		•	•	•
	Microsoft Visio Services		•		•
Business Solutions	Form-Based Application		•		•
	SharePoint 2013 Workflow		New	New	New
	Business Connectivity Services	•	•		•

FIGURE 3-4 A comparison of SharePoint on-premises and SharePoint Online features.

Arguably, the most attractive feature of moving to a SharePoint Online environment is the lowering of associated overhead of managing the infrastructure, both from a supporting hardware as well as from a manpower perspective. From a financial perspective, the cost associated with providing the functionality for the users shifts from the classic capital expenditure (CAPEX) classification to an operating expense (OPEX) classification. In essence, especially for rapidly changing organizations, the cost for user business productivity software (email, collaboration, instant messaging, VoIP audio and video conferencing, when bundled with Office 365) can be clearly planned on a per-user basis, even for thousands of users. High availability and disaster recovery planning and implementation is already included, which with on-prem (or dedicated hosting) deployments can multiply costs exponentially, if not planned accordingly.

The biggest requirement driving necessity for an on-premises or hybrid deployment is typically business intelligence capabilities. Following right behind the business intelligence requirement seems to be the exposure of line-of-business (LOB) data via BCS. However, exposing LOB data via BCS is quite possible under certain conditions in the hybrid architecture.

If the organizational requirements extend beyond a purely SharePoint Online deployment, there are still more benefits to a hybrid architecture than solely relying on an on-prem deployment. Assuming that the organization subscribes or plans to subscribe to Office 365, SharePoint Online is included in the subscription. At that point, it makes financial as well as common sense to offload as much of the configuration, management, and administration of SharePoint capabilities and functions to reduce the overhead for on-prem deployments.

Hybrid architectures can provide the following:

- **Federated search** Users in the cloud and the on-premises domain environment will be able to obtain search results that include content from both locations.
- **BCS** Makes LOB data available to applications for SharePoint and external lists in SharePoint Online.
- **Single sign-on (SSO)** Users who are connected to either the corporate network or Office 365 have to authenticate only once in a given session to access resources in both the on-premises SharePoint farm and SharePoint Online.
- **Directory synchronization** User accounts in the on-premises Active Directory Domain Services (AD DS) domain automatically synchronize to Office 365.
- **One-way or two-way server-to-server trust** A trust relationship between the on-premises SharePoint farm and SharePoint Online that enables secure connections and data flow.

There are three primary architectures when considering hybrid deployments concentrated on providing search and BCS capabilities. From the most basic to the most complex, they are:

- **One-way hybrid search**
 - **SSO** Users who are connected to the corporate network have to authenticate only once in a given session to access resources in both the on-premises SharePoint farm and SharePoint Online.
 - **Directory synchronization** User accounts in the on-premises AD DS domain use Active Directory Federation Services (AD FS) to automatically synchronize to Office 365.
 - **One-way server-to-server trust** A one-way trust relationship is established between SharePoint Online and the on-premises SharePoint farm.
 - **Federated search** Users in your on-premises domain environment will be able to get search results that encompass content from both locations.

Figure 3-5 gives an example of a one-way hybrid search architecture, showing users who are connected to the corporate network and have access to SharePoint Online.

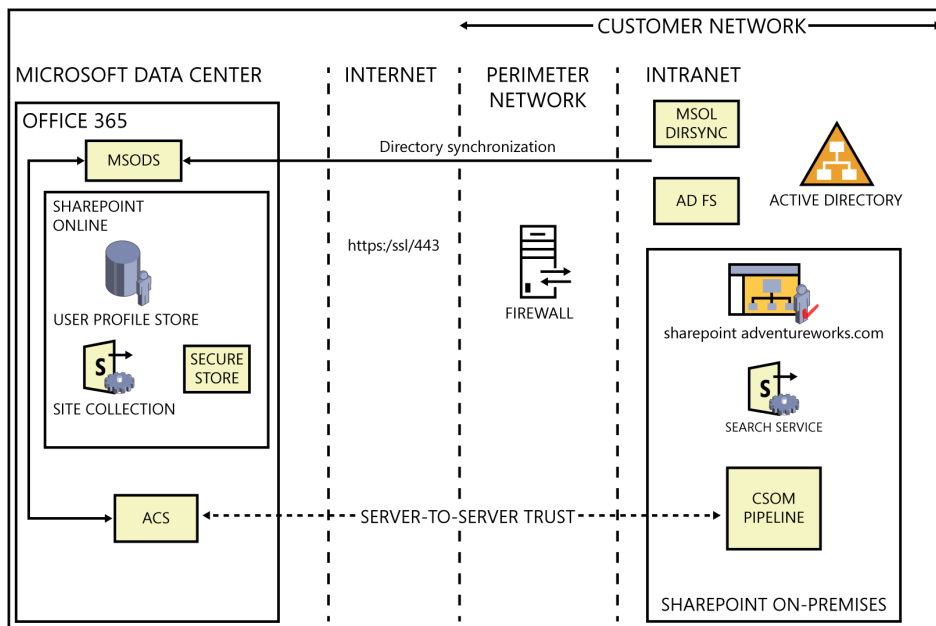


FIGURE 3-5 An example of a one-way hybrid search architecture.

■ Two-way hybrid search

- **SSO** Users who are connected to either the corporate network or Office 365 have to authenticate only once in a given session to access resources in both the on-premises SharePoint farm and SharePoint Online.
- **Directory synchronization** User accounts in the on-premises AD DS domain automatically synchronize to Office 365.
- **Two-way server-to-server trust** A certificate-based two-way trust relationship is established between the on-premises SharePoint farm and SharePoint Online.
- **Two-way federated search** Users in Office 365 and in your on-premises domain environment will be able to get SharePoint Search results that encompass content from both locations.

Figure 3-6 gives an example of a two-way hybrid search architecture showing users who are connected to the corporate network and have access to SharePoint Online.

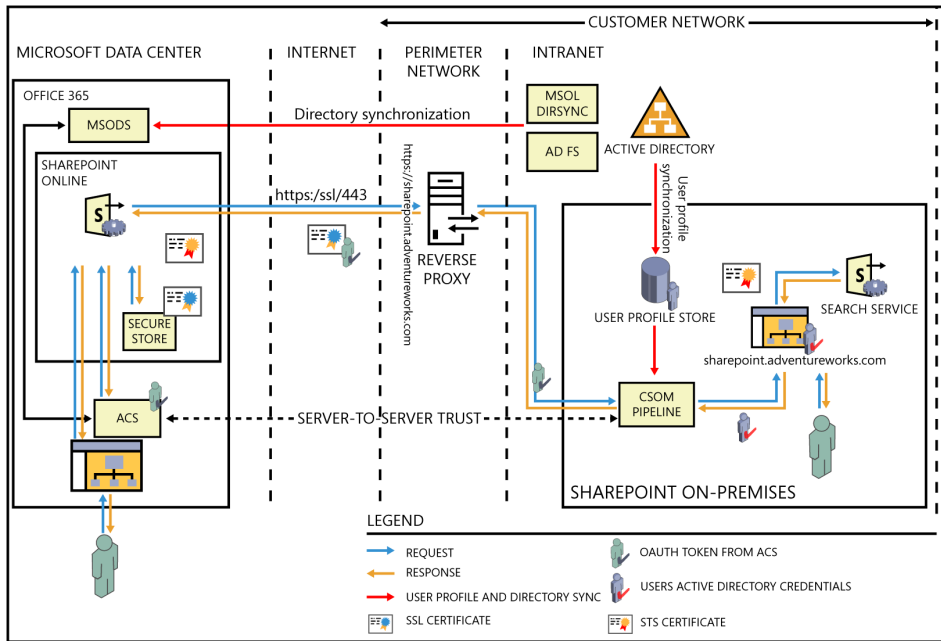


FIGURE 3-6 An example of a two-way hybrid search architecture.

■ Hybrid BCS architecture

A SharePoint 2013 BCS hybrid solution provides a bridge for companies that want to take advantage of cloud-based SharePoint Online to access on-premises LOB data while keeping that proprietary data safely maintained on their corporate intranet. The SharePoint BCS hybrid solution does not require opening holes in the firewall to allow traffic through and it does not require you to move your LOB data out into the perimeter network. The SharePoint BCS hybrid solution uses the on-premises BCS services to connect to the LOB data and then, through a reverse proxy, securely publish the endpoint out to the BCS services in SharePoint Online 2013, as shown in Figure 3-7.

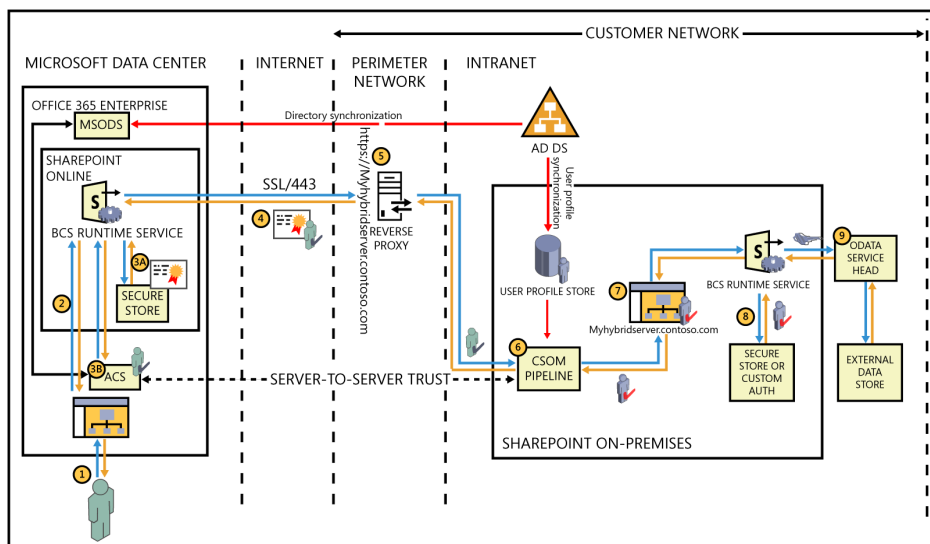


FIGURE 3-7 An example of the hybrid BCS architecture.

Hybrid requirements

Although the SharePoint 2013 technical project team is usually considered to be completely responsible for the deployment of the supporting infrastructure of the system, most organizations will have to include other IT professionals (such as Network Infrastructure, Directory Services, and Certificate Services teams) within their organization to provide the necessary infrastructure requirements to implement a SharePoint 2013 hybrid deployment architecture. Keep in mind, however, that organizations that have already invested in an Office 365 deployment may already have these requirements in place. For example, if an organization has already deployed Exchange Online, either completely online or hybrid, the majority of the prerequisites from an infrastructure perspective should already be in place. It is beyond the scope of this chapter to detail the “how” to deploy the prerequisites, but the following requirements will be necessary:

- AD DS within a functional-level forest of Windows Server 2008, Windows Server 2008 R2, or Windows Server 2012
- On-premises deployment of AD FS 2.0 exposed to the Internet
- On-premises deployment of the Microsoft Online Services Directory Synchronization (DirSync) tool
- An Office 365 Enterprise or SharePoint Online subscription with 15.0.0.4420 as the minimum build number (SharePoint Online Plan 2, Office 365 Enterprise Plan E1, E3, or E4)
- On-premises SharePoint Server 2013 farm that has each of the following configured:
 - Enterprise Search site collection configured with a public external URL (for example, *https://sharepoint.contoso.com*) by using an alternate access mapping
 - An Secure Sockets Layer (SSL) certificate issued by a public root authority

- An App Management Service Proxy installed and published in the SharePoint farm
 - A Subscription Settings service application enabled and configured
 - A Search service application, configured as appropriate
 - A User Profile service application—User profiles contain detailed information about people in an organization. A user profile organizes and displays all of the properties related to each user, together with social tags, documents, and other items related to that user. In the BCS hybrid scenario, it is used to map the users, ACS OAuth credentials to the users' domain credentials.
 - A Client-Side Object Model (CSOM) pipeline—The CSOM receives the incoming request from the reverse proxy and maps the OAuth user tokens from ACS to the users' domain credentials.
 - A Site/Site collection—A site collection created expressly for the purpose of facilitating all hybrid request communication. The web application that this site collection is in has an alternate access mapping configured.
 - BCS Runtime Service SharePoint for on-premises—The BCS Runtime service is the SharePoint service application that manages all BCS functionality, such as administration, security, and communications.
 - Secure Store service SharePoint for on-premises—This is the credential-mapping SharePoint service application. In the SharePoint BCS hybrid solution, SharePoint on-premises stores the mapping of the users' domain credentials to the credentials that are used to access the external data source.
- OData service head—The SharePoint BCS hybrid solution only supports the OData protocol. If your external data is not natively accessible via an OData source, you must use Microsoft Visual Studio to build and deploy an OData service head for it.
 - A reverse proxy device with an Internet connection that permits unsolicited inbound traffic [for example, Microsoft Unified Application Gateway (UAG) 2010 SP3].
 - An Internet domain (such as *tekfocustlab.com*) and access to Domain Name System (DNS) records for the domain

SharePoint Online will need to have the following configured as well:

- BCS Runtime Service Online & Office 365 (O365) Microsoft Online Directory Services (MSODS)—Provides directory services in O365 that you can synchronize with your on-premises AD DS. The synchronization is done through user profile synchronization and allows users to use the same account for both on-premises and cloud authentication.
- SharePoint Online Secure Store service—This is the credential-mapping SharePoint service application. In the SharePoint BCS hybrid solution, SharePoint Online stores an SSL Server certificate that authenticates the SharePoint Online request to the reverse proxy.

- Windows Azure Access Control service (ACS)—This is the Azure security token service that performs authentication and issues security tokens when a user logs on to a SharePoint Online site. It looks up credentials in the MSODS, which has been synchronized with the on-premises AD accounts. This allows the user to use the same set of credentials for both the on-premises and online environments.

Finally, it will be necessary to replace the default token signing certificate for the SharePoint Secure Token service application (one of the default service applications created upon the creation of a new SharePoint Server 2013 farm). It can be replaced with a public certificate, which is recommended, or with a new self-signed certificate that you can create in Internet Information Services (IIS) Manager. A domain-issued certificate *is not supported*.

A hybrid deployment can initially seem to be complex. However, careful planning, approach, and implementation is typically done once, with little maintenance needed.

Putting it all together

As you already hopefully know, gathering, defining, and prioritizing requirements for a system is arguably the most important step in a successful implementation. The balance between planning and actually implementing the system is critical. It is a well-documented fact that there is rarely too much planning. Also, planning will lower the time of implementation if done correctly. Always keep the system requirements as the primary decision-making factors for everything within the implementation. A test implementation will ensure a smooth production implementation. Finally, it is extremely rare to find too much documentation, so ensure that it is created, maintained, and referred to often. This section highlights a number of questions that may enable you, as the Microsoft SharePoint Architect, to gather the requirements that you need to build a successful implementation; while each set of questions is organized into various buckets or individual categories, they may span several.

Planning a successful SharePoint solution strategy

Prior to gathering the requirements for how SharePoint should work, it is important to understand the following:

- Who are the key players involved with the project?
- What are the reporting relationships between the stakeholders?
- Why is each stakeholder involved?
- What are the overall business objectives and vision statements for each?
- How do the business objectives relate to the organization's strategic initiatives and mission?
- Are there any differing goals, conflicts, and so on?
- What will determine the success of the project?
- What processes are in place to maximize and measure user adoption?
- What are the plans for continuing the maintenance and monitoring of the environment?

These questions are more than just “fill in the blanks.” These are generally conversation starters to ensure that enough thought has been put into these areas. Oftentimes, these items are overlooked and it is tough to understand if the implementation was successful or not.

It is important to map the overall business objectives to SharePoint functionality early in the process to help decide if SharePoint is a good solution—these cannot be forced.

Planning a governance strategy

Once you have been able to identify who the stakeholders are and how they will be measuring success, you will be able to continue by gaining an understanding of the governance strategy.

One of the first steps in defining a governance strategy is to build a governance team that will include members of IT, Corporate Training, HR, Corporate Communications, Cyber Security, Site Collection administrators, etc. Some example questions that you will find useful when gathering governance requirements include:

- Who will make up the governance board?
- What is the vision statement of the project?
- What are the defined roles and responsibilities?
- What are the policies and standards with regard to:
 - SharePoint content
 - Information design
 - Security
 - Features
 - Navigation
 - Custom code
 - Composite applications
 - Branding

Besides knowing what rules will need to be followed, it is equally important to determine how governance will be enforced.

Planning the IA

The IA of SharePoint describes how content will be organized and accessed. This can be a time-consuming process and should be well planned prior to adding content into SharePoint. Some of the key topics around this area include:

- What type of sites will be included and how will they be accessed?
- What type of content will SharePoint contain?

- Are there any security restrictions on particular content?
- How will users find this information?
- What is the expected user experience?
- How will documents be stored?
- What properties are important for each type of document?
- What are the commonalities of these properties?
- How are the documents being stored prior to moving them to SharePoint?
- Are there any specific policies around document management?
- What managed terms and keywords are expected?

Identifying business processes that will use SharePoint 2013

SharePoint 2013 offers much more than document management. It is important to identify what business processes need to be implemented using the solution as well. It is also important to help the stakeholders understand which processes can be simplified in SharePoint and which ones cannot. The following questions will help identify possible business processes:

- What business processes are tied to content stored in SharePoint?
- Are there any policies around information stored in SharePoint?
- What type of browsers will be supported in the organization?
- Who is responsible for defining the processes?
- Who is responsible for maintaining or creating workflows with SharePoint?
- What composite applications will be available?
- Do you expect to integrate LOB data within SharePoint?

Understanding the security requirements

Security within SharePoint is a critical piece, and understanding the requirements prior to building out the farm is critical to project success. Changes in security requirements can affect how the overall SharePoint farm is constructed and the communication protocols used for each. It is important to gather the following information:

- Who will have access to the system?
- Who will maintain those users?
- How will they access the system?
- Will a user be able to access the system externally? If so, will their rights differ?

- What form of authentication will be used?
- Will the farm contain Personally Identifiable Information (PII) or information that has specific security around it?
- Are there any specific requirements on how the data should be stored? (Transparent Data Encryption [TDE], and so on)
- Are there any specific requirements on how the information should be transported?
- Will certificates be used within the system?
- Will ports be blocked?
- Will antivirus software be configured?
- What server-hardening techniques are expected?
- Are there plans for any gateway or proxy devices?
- What type of load balancers will be used and what protocols do they support?
- Are there any predefined network topologies that we need to be aware of?
- Will Rights Management be required?
- Will compliance and auditing be required?
- What is the planned response to security threats? (governance)
- What are the current password policies?
- What are the current Group Policy Object (GPO) settings?
- Will security groups be maintained in AD and/or SharePoint?

Understanding the business intelligence requirements

Business intelligence requires additional planning over and above what is needed on a typical SharePoint implementation and can be critical in not only dictating how the farm is constructed, but what SQL Server product is used for support. The following items should be considered:

- Do you plan to incorporate any of the following?
 - Reports
 - Charts
 - Dashboards
 - Scorecards
 - Key Performance Indicators (KPIs)
 - Excel Services

- PerformancePoint
- Microsoft Visio
- Each of these points will need to be planned not only for its existence, but also for the type of data, freshness of data, amounts expected, and so on.

Understanding the role of the Office client

Office has tight integration with SharePoint 2013. The following questions will help you determine the possible Office client solutions:

- What version of Office is available to the users?
- Will all users have a copy of Office locally?
- Do you expect that users will work offline?
- Do you expect a large number of users to be accessing the same documents?
- Do you expect coauthoring?
- Do you expect to support mobile applications?

Understanding the performance and reliability requirements

Once the content of the SharePoint site has been identified, it will be important to gain an understanding of the performance and reliability requirements. The following list of questions will help you determine the appropriate requirements that support performance and reliability:

- How many concurrent users do you expect?
- What is your expected growth or adoption rate over the next three years?
- What is the expected performance metrics?
- What services do you expect to be available at all times?
- Do you expect to span geographical areas? Distance?
- What is the distance from AD to SharePoint to SQL Server?
- When are your peak hours?
- Who is responsible for monitoring and maintaining SharePoint? (governance)
- What is the expected load on the servers?
- How much RAM can we expect to have on each server?
- What is the network speed?
- Are there any specific encryption requirements?

- Will all of the servers be virtual? Vendor? Blade distribution? NUMA boundaries? RAM allocation?
- Describe the load balancers that will be used. Who is responsible for their configuration?
- Is there a budget for Staging and Development servers?
- What is the current software development lifecycle process?
- Are there any concerns about being able to acquire the requested hardware?
- How much time is allocated in the deployment plan for performance and reliability testing?

Every SharePoint 2013 implementation is different, and it is vital to understand the requirements that will help you build a system that will promote success. This section is meant to help trigger requirements that may be critical to the architecture component; you may think of additional questions that help you drive forward to a successful project.

Index

Symbols

- \$ (dollar sign) symbol, assigning objects to PowerShell variable using, 51
- _app_bin folder, 21
- | (pipe) symbol, piping output object to cmdlet, 51–52
- _vti_pvt folder, 21

A

- AAM (Alternate Access Mappings)
 - path-based site collections utilizing, 10
 - zones and, 198
- Access
 - adding databases to SharePoint, 100–101
 - backward compatibility of imported databases, 105
 - supported databases for, 248
- Access Services web service application, 100–101, 105, 144
- ACS (Azure Access Control Service)
 - authorization server for cloud-hosted apps, 191–193
 - for SharePoint Online, 83
 - SSS and, 202
- Active Directory Federation Services (ADFS)
 - 2.0, configuring SAML token-based authentication environments using, 186
- Add-PSSnapin Microsoft.Share-Point.PowerShell cmdlet, 52
- AD DS (Active Directory Domain Services)
 - Claims mode option allowing for implementation of, 198
 - password management and, 238
 - Windows authentication and, 184–185
- Add-SPDiagnosticsPerformanceCounter cmdlet, 310
- Add-SPServiceApplicationProxyGroupMember cmdlet, 94
- Add-SPShellAdmin cmdlet, 43
- ADFS (Active Directory Federation Services)
 - 2.0, configuring SAML token-based authentication environments using, 186
- ADMIN folder, 23
- administrator
 - farm
 - configuring anonymous policies, 206
 - configuring user policies across zones, 204
 - picking objects for web application policies, 206–207
 - site collection
 - granting user rights, 13–14
 - HTML field security, 15
 - picking objects for web application policies, 206
 - recycle bin, 349
 - search features for, 13
- admin.sitemap file, 21
- ADMISAPI folder, 22
- aliases, SQL Server, 133–137
- “All users, All hosts” profile type, creating Windows PowerShell ISE, 47
- “All users, Windows PowerShell ISE” profile type, creating Windows PowerShell ISE, 47
- Alternate Access Mappings (AAM)
 - path-based site collections utilizing, 10
 - zones and, 198
- AlwaysOn Availability
 - protecting databases using, 363–367
 - using SQL Server aliases in configurations, 367
- AlwaysOn Availability Groups
 - about, 128
 - SQL Server support for, 126
- AlwaysOn Failover Cluster Instances
 - about, 128
 - SQL Server support for, 126
- analytics processing component, of Search service application, 109

Analytics Reporting database

- Analytics Reporting database, 30, 110, 141
- analyzing performance of webpages using, 300–302
- Anonymous Access option, 183
- anonymous authentication, 182–183
- anonymous policies, farm administrator
 - configuring, 206
- API, SharePoint, using in Central Administration, 10–11
- App authentication. *See* App model
- App_Browsers folder, 21
- AppFabric component, 322
- App for SharePoint service application, 31
- Application Discovery and Load Balancer
 - service, 115. *See also* Topology service
- Application Management service, 101–102
- application policies, as AuthZ feature, 203–205
- application pool, 8–9
- application pool recycles, ASP.NET, 310
- application pools
 - maximum limit of, 234
 - recommended, 234
 - SSS recommendation for, 202
 - trusted subsystem model and, 171
- App Management database, 140
- App Management service application, 30
- App model
 - about, 17
 - authentication, 190–193
 - joining farm and sandbox solutions, 26–27
- architecture
 - about, 376
 - about SharePoint pipeline, 32–35
 - deployed solutions
 - about, 26–27
 - App model, 27
 - farm solutions, 26–27
 - sandbox solutions, 26
 - farm components, 3–20
 - about, 3–4
 - databases, 10. *See also* SharePoint databases
 - folders, 18
 - libraries, 16–18
 - lists, 16–18
 - site collections, 10–14
 - sites, 15
 - web applications, 4–9
 - Feature Fallback infrastructure in, 25–26
 - file system
 - about, 18
 - IIS files, 19–22
 - service application
 - about, 91
 - databases, 30–32
 - key concepts of, 92–93
 - SQL Server aliases, 29, 133–137
 - validating
 - inspecting network packets, 382–383
 - testing environments. *See* testing environments
 - verifying Kerberos with Klist, 382, 421–422
 - verifying port allocation and web requests, 380–381
 - verifying working of farm, 376–380
- arrows, associated with IIS files, 19
- .asmx pages, location of, 22
- ASP.NET
 - application pool recycles, 310
 - claims-based identity management system based on, 185–186
 - enabling FBA in web application using, 8
 - influencing performance of SharePoint, 34
 - requests queued, 310
 - requests rejected, 310
 - requests wait time, 310
- .aspx pages
 - location of, 22
 - serving in decompile mode, 34
- attach upgrade, 247–248
- authentication
 - anonymous, 182–183
 - App model, 190–193
 - Claims-based
 - about, 184–185
 - default, 172
 - NETBIOS user names and, 194–196
 - standards used in, 184
 - STS and, 115–116
 - Claims mode, 6, 8, 172, 180–182, 198
 - Classic mode, 6, 172, 180–182, 266–267
 - Forms-Based Authentication, 185–186
 - impersonation/delegation, 171
 - mapping
 - about, 197
 - alternate access mappings, 198
 - authentication zones and, 197–198
 - NTLM credentials, 173–174, 320–321

- OAuth, 180, 190
- options when configuring web application, 8
- Server-to-server authentication, 193
- trusted subsystem, 171
- upgrading to SharePoint 2013, 247
- Windows
 - Basic protocol in, 172
 - configuring Kerberos, 176–180
 - Digest protocol in, 173
 - Kerberos and Negotiate protocols, 175–176
 - Kerberos protocol, 175–176
 - NTLM credentials in, 173–174
- zones, 197–198
- AuthN (authentication)
 - about, 171
 - OAuth authentication protocol in, 190
 - options, 171–172
 - services
 - about, 200
 - BCS, 202–203
 - c2WTS, 200–201
 - SSS, 202
- AuthZ (authorization)
 - features
 - application policies, 203–205
 - People Picker control, 206–207
 - sharing, 208
- autocreate statistics, 132
- Autogrowth factor, 317
- Availability Group Listener, 366
- Availability Group vs. database mirroring, 365
- available mbytes, of memory, 309
- Available Tests section, in New Load Test Wizard, 407
- avg. disk queue length, 309
- Avg. Page Time, locating in New Load Test Wizard, 412–413
- Avg. Response Time, locating in New Load Test Wizard, 412–413
- Azure Access Control Service (ACS)
 - about, 27
 - authorization server for cloud-hosted apps, 191–193
 - for SharePoint Online, 83
 - SSS and, 202
- Azure Provisioned App deployment model
 - about, 27

B

- Backup and Restore Job Status link, 344
- backup compression, using, 138
- backup features
 - about, 343–345
 - native, 343–345
 - Site-collection, 345–346
- Backup-SPFarm cmdlet, 344–345
- Backup-SPSite cmdlet, 346
- backward compatibility
 - maintaining, 100
 - of Access web databases, 105
- backward-compatible event handlers, resource throttling, 314
- baseline information, for farm, 161
- Basic authentication, 172
- BCM (Business Continuity Management), applying objectives, 340–341
- BCS (Business Connectivity Service)
 - authorization components of, 202–203
 - hybrid solution, 78, 80–81
 - SSS and, 202
- BCS Runtime service
 - as hybrid architecture requirement in implementation project, 82
 - for SharePoint Online, 82
- BDC (Business Data Connectivity) service, 31, 105, 122
- BI (Business Intelligence)
 - dashboards, 107
 - requirements in implementation project, 86
 - SQL Server
 - enabling, 147–165
 - support for, 127
- Binary Large Object (BLOB)
 - Cache, 327–329
 - in DocStreams table, 161
- Bin folder, 21–22
- Bing.com, 15
- BLOB (Binary Large Object)
 - Cache, 327–329
 - in DocStreams table, 161
- blocking SharePoint installations, 214–215
- browser. *See also* Internet Explorer (IE)
 - publishing within Excel documents within, 106
 - refresting Visio diagrams in web browser, 112–113

- using Get-Help cmdlet to open topics in, 42
- Browser Mix, setting in New Load Test Wizard, 409
- Buildversion.cnf files, 21
- Business Connectivity Service (BCS)
 - authorization components of, 202–203
 - hybrid solution, 78, 80–81
 - SSS and, 202
- Business Connectivity Service (BCS) Runtime service
 - as hybrid architecture requirement in implementation project, 82
 - for SharePoint Online, 82
- Business Continuity Management (BCM), applying objectives, 340–341
- business continuity strategy, planning
 - about, 339–340
 - applying BCM objectives, 340–341
 - avoiding service disruptions, 351–353
 - backup and restore strategies
 - exporting site or list, 346–347
 - native backup and restore, 343–345
 - Read-only improvements, 350–351
 - recovering data from unattached content database, 348
 - Recycle Bin feature, 348–349
 - Site-collection backup, 345–346
 - implementing network load balancing, 368–369
 - protecting databases
 - about, 353–354
 - AlwaysOn Availability for, 363–367
 - database mirroring for, 357–361
 - failover clustering for, 354–356
 - log shipping for, 361–362
 - understanding what to protect, 340
- Business Data Connectivity (BDC) service, 31, 105, 117, 122
- Business Data Connectivity database, 140
- Business Intelligence (BI)
 - dashboards, 107
 - requirements in implementation project, 86
 - SQL Server
 - enabling, 147–165
 - support for, 127
- business processes, identifying in implementation project, 85

C

- c2WTS (Claims to Windows Token Service), 200–201
- .cab files, location of, 22
- cache
 - faults/sec of memory, 309
 - monitoring, 308
- Cache Host, repairing, 323
- Caching Cluster, repairing, 324
- caching, planning and configuring
 - BLOB Cache, 327–329
 - Distributed Cache service, 322–324
 - Object Cache, 326–327
 - Output Cache, 325–326
- capacity planning, database, 145–147
- CAS (Code Access Security), 18
- CEC (Common Engineering Criteria), Microsoft, 37
- Central Administration
 - Configure Usage And Health Data Collection, 306–307
 - content database, 139, 146
 - enabling c2WTS, 201
 - encrypting communications of, 241–243
 - extend web application feature, 197
 - files supporting, 21
 - methods of, 10–11
 - monitoring upgrade process using, 273
 - over SSL, 238, 240–241, 243
 - registering or deleting SharePoint Managed Accounts through, 238
 - setting permissions for Topology service, 121–122
 - Slowest Pages report, 311
 - starting Request Management service in, 330
 - turning on Use Self-Service Site Creation right, 199–200
 - ULS logs in, 22
 - View Analytics reports option, 307
 - web application
 - policies in, 203
 - showing current status of search topology, 57
- Central Administration GUI
 - deploying service applications by, 92, 94
 - implementing service applications, 95
- Certification Authorities (CA), obtaining, 219–220
- change control, 41
- change log, resource throttling, 314
- Claims-based authentication
 - about, 184–185
 - default, 172
 - NETBIOS user names and, 194–196
 - standards used in, 184
 - STS and, 115–116
- claims-based identity management system

- Forms-Based Authentication as, 185–186
- claims encoding, 194–196
- Claims mode authentication, 6, 8, 180–182, 198
- Claims to Windows Token Service (c2WTS), 200–201
- Classic mode authentication, 6, 172, 180–182, 266–267
- cliconfg.exe (SQL Client Configuration Utility),
 - configure SQL Server aliases using, 133–134
- client and server, encrypting communications
 - between client and server, 219–222
- Client folder, 22
- Client-Side Object Model (CSOM) pipeline, as
 - hybrid architecture requirement in implementation project, 82
- CLIs vs. GUIs, 39
- cloud-based services, ACS as trusted broker for, 193
- cloud-hosted apps, utilizing in App
 - authentication, 191–193
- cluster mode, supporting unicast and multicast
 - methods, 369
- cmd.exe (Windows Command Prompt), 40
- cmdlets
 - about, 39
 - discovering SharePoint-specific PowerShell, 52–55
 - index of, 54–55
 - learning about specific, 40
 - running on server, 45
 - working with PowerShell, 48–52
- Code Access Security (CAS), 18
- collation, SQL Server, 128–130
- Column default value settings feature, 18
- Command Add-Ons for Windows PowerShell
 - ISE, 48–49, 52
- comments, using, 58
- Common Engineering Criteria (CEC), Microsoft, 37
- components, service application, 92–93
- concurrent users
 - about, 299–303
 - setting load test standards for peak user percentage of, 396
- CONFIG folder, 22
- configuration content databases, 10
- configuration database, 10, 139, 146, 248
- configuration file hierarchy, 33
- Configure Service Accounts, 201
- Configure Usage And Health Data Collection, 306–307

- Connect drop-down menu, 123
- Connect-SPOService cmdlet, 44
- content database, 139–140, 146, 256–257, 280, 287–288
- content deployment source feature, 15
- content processing component, of Search service
 - application, 109
- Content Query Web Part, 327
- content types
 - as implementation project deliverable, 68
 - in information architecture, 70, 71
 - in metadata architecture, 72
- content type syndication hub, 15
- Contoso-SQL1, server properties, 316
- CONTROLTEMPLATES folder, 23
- Convert-SPWebApplication cmdlet, 182, 267
- Counter Sets, setting up in New Load Test Wizard, 409
- crawl component, of Search service application, 109
- Crawl database, 30, 110, 141
- Create New Web Application form, 221
- Create, Read, Update, Or Delete (CRUD) operations,
 - BDC service allowing, 105
- cross-farm site permissions, 15
- cross-list queries, 327
- cross-site collection publishing, 15
- CRUD (Create, Read, Update, Or Delete) operations,
 - BDC service allowing, 105
- CSOM (Client-Side Object Model) pipeline, as
 - hybrid architecture requirement in implementation project, 82
- “Current user, All hosts” profile type, creating
 - Windows PowerShell ISE, 47
- “Current user, Windows PowerShell ISE” profile type,
 - creating Windows PowerShell ISE, 47
- custom development code files, compiling of, 21
- custom web services, SSS and, 202
- Custom zone, in web applications, 8, 197

D

- daily time window for large queries, resource
 - throttling, 314
- DAS (Direct Attached Storage), 147
- Dashboard, Developer, 379–380
- dashboards, business intelligence, 107
- Data Alerts for Reporting Services reports, 145
- database capacity planning, 145–147
- Database Encryption Key (DEK), 231

database layer, SharePoint

- database layer, SharePoint, 125
- database maintenance plan, defining and executing, 317–320
- database mirroring
 - about, 127
 - protecting databases using, 357–361
 - SQL Server support for, 126–128
 - vs. Availability Group, 365
- databases
 - about, 138
 - adding to SharePoint Access, 100–101
 - engine requirements, 126–128
 - for integrating SharePoint with SSRS, 144–145
 - for Search service application, 110
 - managing growth of data, 137–138
 - protecting
 - about, 353–354
 - AlwaysOn Availability for, 363–367
 - database mirroring for, 357–361
 - failover clustering for, 354–356
 - log shipping for, 361–362
 - service applications and, 96
 - shrinking, 319
- databases, SharePoint
 - about, 10, 138
 - Analytics Reporting, 31, 110, 141
 - App Management, 140
 - attach upgrade of, 247–248
 - Business Data Connectivity, 140
 - Central Administration content, 139, 146
 - configuration, 139, 146
 - content, 139–140, 146, 256–257, 280, 287–288
 - Crawl, 31, 110, 141
 - Link, 31, 110, 141
 - Machine Translation Services, 143
 - Managed Metadata, 143
 - PerformancePoint services, 144
 - PowerPivot, 143–144
 - Profile, 30, 112, 142, 264–266, 285
 - Search Administration, 30, 110, 140
 - Search service application, 140
 - Secure Store, 141
 - Social, 143, 264–266, 285
 - SQL Server aliases, 29, 133–137
 - State Service, 143
 - Subscription Settings, 142
 - supported, 247–248
 - Synch, 30, 112, 142, 264–266, 285
 - system databases, 29
 - unsupported, 248
 - Usage, 142, 307, 308
 - user credentials stored in LDAP, 185
- data files, shrinking, 319
- DBCC CheckDB statement, 318
- db_owner, 239
- Debug Cache Information option, 326
- debugging, webpages using IE Dev Tools, 381
- decompile mode, serving .aspx pages in, 34
- Default zone, in web applications, 8, 197
- deferred site collection upgrade, 249
- delegation, configuring server for, 176–180
- deliverables, implementation project
 - about, 66–67
 - content types, 68
 - functional requirements, 67
 - governance plan (Internet usage policy) and documentation, 69
 - Nonfunctional requirements, 67
 - site structure, 68–69
 - taxonomy and metadata, 68
 - usage scenarios, 67
- Deny All permissions, user permission policy and, 204–205
- Deny Write permissions, user permission policy and, 204–205
- deployed solutions
 - about, 26–27
 - App model, 27
 - farm solutions, 26–27
 - sandbox solutions, 26
- DesingHttpModule, 33–34
- Developer Dashboard, SharePoint
 - about, 379–380
 - analyzing performance of webpages using, 300
 - enabling/disabling, 312
 - monitoring page performance using, 311–312
- Digest authentication, 173
- Direct Attached Storage (DAS), 147
- directory synchronization, in hybrid architecture, 78, 79
- Disaster Recovery (DR) plan
 - about, 41, 339–340
 - applying BCM objectives, 340
 - avoiding service disruptions, 351–353
 - backup and restore strategies
 - exporting site or list, 346–347
 - native backup and restore, 343–345
 - Read-only improvements, 350–351

- recovering data from unattached content
 - database, 348
 - Recycle Bin feature, 348–349
 - Site-collection backup, 345–346
- implementing network load balancing, 368–369
- protecting databases
 - about, 353–354
 - AlwaysOn Availability for, 363–367
 - database mirroring for, 357–361
 - failover clustering for, 354–356
 - log shipping for, 361–362
- requirements for supported options using SQL
 - Server, 126–128
- understanding what to protect, 340

disks

- avg. disk queue length, 309
- monitoring, 308
- pass-through, 322
- % free space, 309
- % idle time, 309

disk subsystem performance analysis, 169

disposable objects, working with, 56

Dispose() method, 56

Distributed Cache service, 322–324

DLL (Dynamic-Link Library) files, Microsoft.Office, 21

.DLL files

- custom development code files as, 21
- location of, 22

DNS (Domain Name System)

- URLs configured in, 7

DocumentTemplates folder, 23

dollar sign (\$) symbol, assigning objects to

- PowerShell variable using, 51

Domain CA

- encrypting communications between client and
 - server using, 219–220
- using, 219, 241

Domain Name System (DNS)

- URLs configured in, 7

domains

- default external, 15

DR (Disaster Recovery) plan

- about, 339–340
- applying BCM objectives, 340–341
- avoiding service disruptions, 351–353
- backup and restore strategies
 - exporting site or list, 346–347
 - native backup and restore, 343–345
 - Read-only improvements, 350–351

- recovering data from unattached content
 - database, 348
 - Recycle Bin feature, 348–349
 - Site-collection backup, 345–346
- implementing network load balancing, 368–369
- protecting databases
 - about, 353–354
 - AlwaysOn Availability for, 363–367
 - database mirroring for, 357–361
 - failover clustering for, 354–356
 - log shipping for, 361–362
- requirements for supported options using SQL
 - Server, 126–128
- understanding what to protect, 340

Dynamic-Link Library (DLL) files, Microsoft.

- SharePoint, 21

E

ECMAScript, within Excel Services application, 106

email notifications, during upgrade to SharePoint
 2013, 253

encrypting communications

- about, 218
- between client and server, 219–222
- Central Administration communications, 241–243
- enabling SSL encryption between applications
 - and SQL Server, 225–229
- installing SharePoint using least privileges, 233–239
- obtaining Certification Authorities, 219–220
- planning for configuring SQL Server TDE, 231–233
- server to server, 223
- using IPsec and IKEv2, 229–231

environment, SharePoint

- maintenance cycle in, 297–298
- monitoring
 - about, 298
 - health of environment, 303–307
 - page performance, 311–312
 - performance counters, 308–311
 - SharePoint storage, 138, 312–313
- setting up Online Management Shell
 - environment, 44
- troubleshooting, 332–333
- turning and optimizing
 - fine-tuning network performance, 320–322

errors during stress testing

- planning and configuring caching, 322–329
- Request Management service, 329–331
- resource throttling, 314–315
- SQL optimization, 315–317
- using database maintenance plan, 317–320
- errors, during stress testing, 415
- evaluation site collections upgrade, 250–251
- event viewer, browsing and managing event logs, 300
- Excel, publishing within browser output from, 106
- Excel Services application
 - about, 106
 - c2WTS and, 200, 201
 - SSS and, 202
 - upgrading, 263
- Execution Groups, Request Management
 - Configuration element, 330–331
- .exe files, location of, 22
- exporting site or list, in BCM plan, 346–347
- export search configuration, search features for site collection administrators, 14
- Export-SPWeb cmdlet, 347
- external domains, allowed by default, 15
- Extranet zone, in web applications, 8, 197

F

- failover clustering
 - about, 127
 - protecting databases using, 354–356
 - requirement in HADRON, 363
 - requirements for supported options using SQL Server, 126–128
- farm
 - about, 4
 - baseline information for, 161
 - building services, 119–124
 - configuring, 93
 - creating trust, 118, 120–121, 282–283
 - determining capacity of, 397–398
 - implementing services, 117–119
- farm administrator
 - configuring anonymous policies, 206
 - configuring user policies across zones, 204–205
 - picking objects for web application policies, 206–207
- farm components, 3–20
 - about, 3–4
 - databases, 10. *See also* SharePoint databases
 - folders, 18
 - libraries, 16–18
 - lists, 16–18
 - site collections, 10–14
 - sites, 15
 - web applications, 4
- Farm Configuration database, 248
- farm configuration databases, 10
- Farm Configuration Wizard, SharePoint
 - deploying service applications using, 96
 - for configuring usage and health provider settings, 304
 - using, 41
- farm solutions, joining sandbox solutions and, 26
- FAST Service, integration into Search service application, 108–110
- FBA (Forms-Based Authentication), 8, 185–186
- feature fallback behavior, 249
- Feature Fallback infrastructure, 25–26
- FEATURES folder, 23
- features vs. apps, 17
- federated searches, in hybrid architecture, 78, 79
- federated Search farm, incorporating after SharePoint 2013, 292
- federating services, 117–119, 282–283
- Fiddler, 381
- FileReadChunkSize property, 161
- files, reducing size of, 138
- file system
 - about, 18
 - IIS files, 19–22
- FileWriteChunkSize property, 161
- fill factor setting, 319
- FIM service, 112
- fine-tuning
 - configuration, 41
 - page load performance, 393
- folders
 - about, 18
 - as method for organizing information in implementation project, 72
- forms-based authentication claims
 - c2WTS and, 201
- Forms-Based Authentication (FBA), 8, 185–186
- Foundation Subscription Settings service
 - application, 31
- FQDNs (Fully Qualified Domain Names), configuring SPNs for, 178
- Full Control permissions configuration, user permission policy and, 204
- Full Control right, 191

- Full Read configuration, user permission policy and, 204
- Full recovery mode, database mirroring in, 358
- Fully Qualified Domain Names (FQDNs), configuring SPNs for, 178
- functional requirements, as implementation project deliverable, 67

G

- GAC (Global Assembly Cache)
 - custom development code files deployed in, 21
 - wpresources folder located in, 21
- gathering implementation project requirements
 - about, 62–64
 - business intelligence requirements, 86
 - deliverables
 - about, 66–67
 - content types, 68
 - functional requirements, 67
 - governance plan (Internet usage policy)and documentation, 69, 84
 - Nonfunctional requirements, 67
 - site structure, 68–69
 - taxonomy and metadata, 68
 - usage scenarios, 67
 - hardware and software requirements, 75–76
 - hybrid architectures
 - considerations, 77–80
 - requirements, 81–82
 - identifying business processes, 85
 - identifying stakeholders, 64–65
 - information architecture, 70–72, 84–85
 - logical architecture, 72–73
 - mapping objectives to functionality, 66
 - Office client role, 87
 - organizational objectives, 65–66
 - performance and reliability requirements, 87–88
 - physical architecture, 74
 - planning strategy
 - about, 63–64
 - for SharePoint solution, 83–84
 - prioritizing goals, 66
 - questions to ask in, 64, 83–88
 - security requirements, 85–86
 - SharePoint Online
 - considerations, 76–77
 - requirements, 82–83
 - tools and process, 66

- understanding system requirements, 75
- Get-Command cmdlet, 42, 48, 52
- Get-Help cmdlet, 40, 42, 49–50
- Get-SPAlternateUrl cmdlet, 258
- Get-SPAuthenticationProvider cmdlet, 258
- Get-SPClaimTypeEncoding cmdlet, 196
- Get-SPContentDatabase cmdlet, 260
- Get-SPDatabase cmdlet, 260
- Get-SPDiagnosticPerformanceCounter cmdlet, 308
- Get-SPLogEvent cmdlet, 333
- Get-SPManagedPath cmdlet, 260
- Get-SPShellAdmin cmdlet, 255
- Get-SPSite cmdlet, 291–292
- Get-SPUsageApplication cmdlet, 305
- Get-SPUsageService cmdlet, 305
- Get-SPUser cmdlet, 260
- Get-SPWeb cmdlet, 260
- global.asax file, 21
- Global Assembly Cache (GAC)
 - custom development code files deployed in, 21
 - wpresources folder located in, 21
- goals of implementation project
 - mapping to functionality, 66
 - prioritizing, 66
- governance plan (Internet usage policy) and documentation
 - as implementation project deliverable, 69
 - planning strategy, 84
- Gradual Site Delete timer job, 350
- green zone
 - in setting load test standards, 396
 - setting threshold values, 411
- group permissions, understanding, 240
- guidelines, for configuring SPNs, 177–178
- GUIs vs. CLIs, 39

H

- HADRON (High Availability Disaster Recovery - AlwaysOn), 363–367
- HA (High Availability)
 - option overview, 127–128
 - requirements for supported options using SQL Server, 126–128
- hardware and software requirements, in implementation project, 75–76
- HCCab folder, 22
- Health Analyzer, SharePoint
 - about, 299

Health Logging Schedule

- correcting index fragmentation, 319
- rules, 303–304, 318
- troubleshooting using reports from, 336, 338
- Health Logging Schedule, 307
- health reports, viewing environment, 299
- Help folder, 22
- High Availability Disaster Recovery-AlwaysOn (HADRON), 363–367
- High Availability (HA)
 - option overview, 127–128
 - requirements for supported options using SQL Server, 126–128
- host headers, 7, 200
- host header sites, 7
- host-named site, 10–11
- host root partition, in virtual servers, 322
- Hot standby redundancy method, 358
- “How to: Configure Network Emulation Using Test Settings in Visual Studio”, 408
- HTML fields, security allowing inserting iFrames in, 15
- HTTP modules
 - web applications modifying, 33–34
- http request monitoring and throttling, resource throttling, 314
- hybrid architectures, in implementation project
 - considerations for, 77–80
 - requirements for, 81–82
- hybrid BCS architecture, 80–81

I

- IA (Information Architecture), in implementation project, 70–72, 84–85
- Identity claim component, of SAML token-based authentication, 186
- Identity provider Security Token Service (IP-STS), 186
- IDisposable interface Dispose() method, 56
- IE Dev Tools (Microsoft Internet Explorer Developer Tools)
 - debugging webpages using, 381
 - enabling, 391
 - result codes in performance testing using, 392
- IE (Internet Explorer)
 - enabling developer tools, 420–421
 - recording web sessions using, 417
- iFrames, security allowing inserting in HTML fields, 15
- IIS (Internet Information Services)
 - files, 19–22
 - folders, 22–23
 - hosting WCF framework, 94
 - testing environment using logs from, 384–385
 - trusted subsystem model and application pools in, 171
 - web application
 - making changes to web.config file, 198
 - website
 - creating web application and, 7
 - web applications hosted in, 6
- IKEv2 (Internet Key Exchange version 2), IPsec and, 229–231
- IMAGES folder, 23
- impersonation/delegation authentication, 171
- implementation project requirements, gathering
 - about, 62–64
 - business intelligence requirements, 86
 - deliverables
 - about, 66–67
 - content types, 68
 - functional requirements, 67
 - governance plan (Internet usage policy) and documentation, 69, 84
 - Nonfunctional requirements, 67
 - site structure, 68–69
 - taxonomy and metadata, 68
 - usage scenarios, 67
 - hardware and software requirements, 75–76
 - hybrid architectures
 - considerations, 77–80
 - requirements, 81–82
 - identifying business processes, 85
 - identifying stakeholders, 64–65
 - information architecture, 70–72, 84–85
 - logical architecture, 72–73
 - mapping objectives to functionality, 66
 - Office client role, 87
 - organizational objectives, 65–66
 - performance and reliability requirements, 87–88
 - physical architecture, 74
 - planning strategy
 - about, 63–64
 - for SharePoint solution, 83–84
 - prioritizing goals, 66
 - questions to ask in, 64, 83–88

- security requirements, 85–86
 - SharePoint Online
 - considerations, 76–77
 - requirements, 82–83
 - tools and process, 66
 - understanding system requirements, 75
 - implementation project SQL Server installation, 128–132
 - import search configuration, search features for site
 - collection administrators, 14
 - Import-SPWeb cmdlet, for import of sites and lists, 347
 - index component, of Search service application, 110
 - index fragmentation, measuring and reducing, 319
 - index of cmdlets, 54–55
 - index performance, fine-tuning, 319
 - InfoPath Forms services
 - c2WTS and, 200, 201
 - upgrading, 263
 - Information Architecture (IA), in implementation project, 70–72, 84–85
 - Information Technology (IT) department, role in implementation, 64
 - in-memory caching, 322
 - in-place upgrades, 247
 - Input/Output (I/O) planning, 145–147
 - Input/Output operations per second (IOPS)
 - correlation between throughput and, 166
 - SQL Server storage and, 145–147
 - installations SharePoint
 - preventing, 214–215
 - reporting, 217
 - tracking, 215–217
 - Integrated Scripting Environment (ISE), Windows PowerShell
 - about, 38, 45
 - Command Add-Ons for, 48–49, 52
 - integration with Microsoft products
 - OWA (Office Web Applications) server, 96–98
 - Internet Explorer Developer Tools (IE Dev Tools)
 - debugging webpages using, 381
 - Internet Explorer (IE)
 - enabling developer tools, 420
 - recording web sessions using, 417
 - Internet Information Services (IIS)
 - files, 19–22
 - folders, 22–23
 - hosting WCF framework, 94
 - testing environment using logs from, 384–385
 - trusted subsystem model and application pools
 - in, 171
 - web application
 - making changes to web.config file, 198
 - website
 - creating web application and, 6, 7
 - Internet Key Exchange version 2 (IKEv2), 229–231
 - Internet usage policy (governance plan) and documentation
 - as implementation project deliverable, 69
 - planning strategy, 84
 - Internet zone, in web applications, 8, 197
 - Intranet zone, in web applications, 8, 197
 - I/O (Input/Output) planning, 145–147
 - IOPS (Input/Output operations per second)
 - correlation between throughput and, 166
 - SQL Server storage and, 145–147
 - IP load balancer, with Windows NT Server 4.0, 368–369
 - IPsec
 - about, 229
 - IKEv2 and, 229–231
 - IP-STS (Identity provider Security Token Service), 186
 - ISAPI folder, 22
 - iSCSI attached storage, size recommendation of NIC
 - for, 321
 - iSCSI network drives, testing, 166–167
 - ISE (Integrated Scripting Environment), Windows PowerShell
 - about, 38, 45
 - Command Add-Ons for, 48–49, 52
 - IT (Information Technology) department, role in implementation, 64
- ## J
- JavaScript Object Notation (JSON), using to establish ACS trust relationship, 193
- ## K
- KDC (Key Distribution Center), Kerberos, 321
 - Kerberos
 - configuring, 176–180
 - enabling for SQL Server BI, 160
 - Key Distribution Center, 321
 - Negotiate and
 - about, 175–176

- as recommended security configuration to
 - use with Windows authentication, 8
- pass-through authentication, 202
- verifying using Klist, 382, 421–422
- vs. NTLM, 320–321
- Kerbtray, verifying Kerberos using, 382
- Key Distribution Center (KDC), Kerberos, 321
- key stakeholders, 68, 69
- key stakeholders, in implementation project, 67
- Klist
 - about, 382
 - verifying Kerberos using, 421–422
- Knowledge Base (KB), Microsoft articles
 - on supportability regarding SQL collation, 128

L

- landing page, creating for business continuity plan, 373
- language packs, upgrading to SharePoint 2013 and, 267
- languages, translating, 102
- large lists, 17
- Large Object Binary (LOB) data, Business Data Connectivity service and, 105
- LAYOUTS folder, 23
- layouts.sitemap file, 21
- LCIDSts folder, 22
- LDAP (Lightweight Directory Access Protocols)
 - database, storing user credentials in, 185
- least privileges, installing SharePoint using, 233–239
- libraries, 16–18
- Lightweight Directory Access Protocols (LDAP)
 - storing user credentials in, 185
- Link database, 30, 110, 141
- list items
 - about, 18
- lists
 - about, 16–18
 - export of, 347
 - import of, 347
- list unique permissions threshold, resource throttling, 314
- list view lookup threshold, resource throttling, 314
- list view threshold for auditors and administrators, resource throttling, 314
- list view threshold, resource throttling, 314
- Load Pattern, determining in New Load Test Wizard, 405
- load testing environments
 - about, 394
 - building demo, 417–418
 - building load test, 403–413, 424–427
 - creating Performance Web Test, 401–403
 - determining capacity of farm, 397–398
 - notification when breaching thresholds, 411
 - options for, 394–395
 - setting standards for, 396
 - setting up Visual Studio Ultimate 2012 in, 398–400
 - Visual Studio Ultimate in, 417
- LOB (Large Object Binary) data, Business Data Connectivity service and, 80, 105
- Log Collection, 307
- log files, managing growth of data, 137–138
- logical architecture, in implementation project, 72–73
- Log Parser, 333
- Log Parser, combing through IIS logs using, 385–388
- Log Parser Studio (LPS), setting up, 386–388
- LOGS folder, 22
- log shipping
 - about, 127
 - SQL Server support for, 126–128
- log shipping, for protecting databases, 361–362
- logs, troubleshooting using, 332–333
- LPS (Log Parser Studio), setting up, 386–388
- LUN activity, 315

M

- machine.config file, 33
- Machine Pool, Request Management Configuration element, 330–331
- Machine Translation service application, 31, 102–103, 117, 143
- Machine Translation Services database, 143
- maintenance cy4cle, turning and optimizing, 317
- maintenance cycle
 - about, 297–298
 - monitoring environment
 - about, 298
 - determining health of environment, 303–307
 - page performance in, 311–312
 - performance counters, 308–311
 - SharePoint storage, 312–313
- troubleshooting, 332–333
- turning and optimizing
 - fine-tuning network performance, 320–322

- planning and configuring caching, 322–329
 - Request Management service, 329–331
 - resource throttling, 314–315
 - SQL optimization, 315–317
- Maintenance Plan Wizard, 319–320
- Managed Accounts, SharePoint
 - for application pool, 9
 - platform security and, 238–240
- Managed Metadata database, 143, 248
- Managed Metadata service (MMS) application, 31, 106–107, 117
- managed paths, site collections added using, 10
- Management Framework 3.0, Windows, 44
- management shell, PowerShell, 44–46
- Manage right, 191
- mapping authentication
 - about, 197
 - alternate access mappings, 198
 - authentication zones and, 197–198
- mapping objectives to functionality, in gathering
 - implementation project requirements, 66
- Max Degree of Parallelism (MAXDOP), 315
- Max Degree of Parallelism (MDOP), SQL Server, 130–132
- MDOP (Max Degree of Parallelism), SQL Server, 130–132
- memory
 - available mbytes, 309
 - cache faults / sec, 309
 - monitoring, 308
 - pages / sec, 309
- metadata
 - and taxonomy as implementation project
 - deliverable, 68
 - architecture in completing information
 - architecture, 71
- Microsoft Access
 - adding databases to SharePoint, 100–101
 - backward compatibility of imported
 - databases, 105
- Microsoft Common Engineering Criteria (CEC), 37
- Microsoft Excel. *See also* Excel Services application
 - publishing within browser output from, 106
- Microsoft Internet Explorer Developer Tools (IE Dev Tools)
 - debugging webpages using, 381
 - enabling, 391
- Microsoft Knowledge Base (KB) article, on
 - supportability regarding SQL
 - collation, 128
- Microsoft Log Parser, combing through IIS logs
 - using, 385–388
- Microsoft Management Console (MMC), configuring
 - SQL Server instance to use SSL, 225
- Microsoft Message Analyzer (MMA), capturing
 - network packets using, 383
- Microsoft Network Monitor (Netmon)
 - capturing network packets using, 383
 - Visual Round Trip Analyzer and, 389
- Microsoft Office 365, SSL protocol on sites hosted
 - by, 7
- Microsoft.Office.dynamic-link.library file, 21
- Microsoft Office SharePoint Server (MOSS) 2007, use
 - of zones in, 197
- Microsoft Office Web Apps
 - downloading, 96
 - integration with Microsoft products, 96–98
 - performance counters, 310
- Microsoft PowerShell. *See also* PowerShell
 - creating initial farm using, 29–30
 - launching SQL Server Client Network Utility, 29
- Microsoft SharePoint
 - as trusted subsystem, 171
 - pipeline, 32–35
 - SQL Servers versions supported by new, 353
- Microsoft.SharePoint.dynamic-link.library file, 21
- Microsoft SharePoint Foundation Subscription
 - Settings service application, 31
- Microsoft SharePoint Foundation Web Application
 - (SPFWA) service, 329
- Microsoft.SharePoint.WorkflowServices.Activities.dll
 - file, 23
- Microsoft SQL Server. *See* SQL Server
 - AlwaysOn Availability feature of, 363–367
 - versions supported by new SharePoint, 353
- Microsoft SQL Server Reporting Services
 - (SSRS), 144–145, 300
- Microsoft TechNet documentation, index of
 - cmdlets, 54–55
- Microsoft TechNet Script Center, 55
- Microsoft Translator, 102
- Microsoft ULS Viewer, setting up, 377–379
- Microsoft Visio. *See also* Visio Graphics service
 - (VGS)
 - refreshing diagrams in web browser, 112–113

- Microsoft Visual Studio Team System (VSTS) 2008 Team Suite, using for demo test load, 417–418
 - Microsoft Visual Studio (VS) Ultimate 2012
 - in load testing environments, 417
 - setting up in testing environment, 398–400, 423
 - Microsoft Word, conversion of documents in, 113–114
 - MigrateUsers() method, 181–182
 - MIIS key
 - exporting, 265
 - importing, 266
 - MMA (Microsoft Message Analyzer), capturing network packets using, 383
 - MMC (Microsoft Management Console), configuring SQL Server instance to use SSL, 225
 - MMS (Managed Metadata service) application, 32, 106–107, 117
 - module autoloading, 42
 - monitoring environment
 - about, 297–298
 - determining health of environment, 303–307
 - page performance in, 311–312
 - performance counters, 308–311
 - SharePoint storage, 138, 312–313
 - monthly processes, in monitoring environment, 303
 - MOSS (Microsoft Office SharePoint Server) 2007, use of zones in, 197
 - MSSQLSERVER properties, viewing protocols for, 227–228
 - Multicast method, 369
 - multi-tenancy, Self-Service Site Creation and, 13
 - My Site host, upgrading, 254, 287, 288–290
- N**
- NAS (Network Attached Storage), 147
 - native backup and restore, 343–345
 - NDS (Novell Directory Services), 185
 - Negotiate, Kerberos and, 8, 175–176
 - NetBIOS names, configuring SPNs for, 178
 - NETBIOS user names, 194–196
 - .NET Framework
 - about, 42
 - Claims-based authentication built on classes in, 184
 - .NET Framework Class Library (FCL), 40
 - Netmon (Network Monitor)
 - capturing network packets using, 383
 - Visual Round Trip Analyzer and, 389
 - network
 - fine-tuning performance of, 320–322
 - troubleshooting, 382–383
 - Network Attached Storage (NAS), 147
 - Network Emulation, setting up, 408
 - Network Interface Card (NIC)
 - monitoring, 308
 - size recommendation of, 321
 - Network Load Balancer (NLB)
 - implementing, 368–369
 - total requests per hour logs and, 387
 - Network Load Balancing Manager, 369
 - Network Mix, selecting network in New Load Test Wizard, 408
 - Network Monitor (Netmon)
 - capturing network packets using, 383
 - Visual Round Trip Analyzer and, 389
 - network packets, inspecting, 382–383
 - New Load Test Wizard
 - in load testing
 - Available Tests section in, 407
 - determining Load Pattern, 405
 - locating Avg. Page Time in, 412–413
 - locating Avg. Response Time in, 412–413
 - Run Settings window in, 410
 - selecting Network Mix, 408
 - selecting Test Mix Model, 405
 - setting Browser Mix, 409
 - setting up Counter Sets, 409
 - Test Mix window in, 407
 - in stress testing
 - creating Test Mix Model, 416
 - Step Load test settings, 416
 - selecting Think Time Profile, 404
 - New-SPAuthenticationProvider cmdlet, 267
 - New-SPClaimTypeEncoding cmdlet, 196
 - New-SPMetadataServiceApplication cmdlet, 255
 - New-SPMetadataServiceApplicationProxy cmdlet, 255, 285
 - New-SPSite cmdlet, 10–11, 57
 - New-SPTrustedSecurityTokenIssuer cmdlet, 193
 - New-SPUsageApplication cmdlet, 305
 - New-SPWebApplication cmdlet, 267
 - New-SPWebApplicationExtension cmdlet, 197
 - NIC (Network Interface Card)
 - monitoring, 308
 - size recommendation of, 321
 - NLB (Network Load Balancer)
 - implementing, 368–369

- total requests per hour logs and, 387
- Nonfunctional requirements, as implementation
 - project deliverable, 67
- Novell Directory Services (NDS), 185
- Novell eDirectory, 185
- NTLM (Windows NT LAN Manager) credentials
 - about, 8
 - credentials in Windows, 173–174
 - pass-through authentication, 202
 - search crawl the web application with AD DS, 198
 - vs. Kerberos, 320–321
- NUMA nodes, 321

O

- OAuth authentication
 - about, 190
 - Claims mode and, 180
 - cloud-hosted apps and, 192–193
 - S2S authentication using, 193
- Object Cache, 326–327
- objectives of implementation project
 - mapping to functionality, 66
 - prioritizing, 66
- object model override, resource throttling, 314
- objects, working with disposable, 56
- ODATA (Open Data Protocol)
 - service head, 82
 - within Excel Services application, 106
- Office 365, SSL protocol on sites hosted by, 7
- Office and SharePoint Products Network Monitor
 - Parsers, 383
- Office.microsoft.com, 15
- Office, role in implementation project of, 87
- Office Web Applications (OWA)
 - downloading, 96
 - integration with Microsoft products, 96–98
 - performance counters, 310
- one-way hybrid search architecture, 78
- one-way server-to-server trust, in hybrid
 - architecture, 78
- Online Management Shell environment, SharePoint,
 - setting up, 44–46
- Online Secure Store service, SharePoint
 - for SharePoint Online, 82
- Open Data Protocol (ODATA)
 - service head, 82
 - within Excel Services application, 106

- Operations Manager, with System Center
 - Management Pack, 300
- optimizing, SQL Server
 - about, 161
 - initial optimization and configuration of, 128–130
 - about, 128
 - autocreating statistics, 132
 - dedicated server, 132
 - managing growth of data and log files, 137–138
 - MDOP, 130–132
 - monitoring storage and performance, 138
 - using aliases, 133–137
- optimizing, turning and, environment
 - fine-tuning network performance, 320–322
 - planning and configuring caching, 322–329
 - Request Management service, 329–331
 - resource throttling, 314–315
 - SQL optimization, 315–317. *See also* optimizing, SQL Server
 - using database maintenance plan, 317–320
- organizational objectives, in gathering
 - implementation project requirements, 65–66
- Output Cache, 325–326
- OWA (Office Web Applications)
 - downloading, 96
 - integration with Microsoft products, 96–98
 - performance counters, 310

P

- packet tracing, 382–383
- page load performance, fine-tuning, 393
- Page Output Cache, 325
- page performance, monitoring, 311–312
- pages / sec, of memory, 309
- paging file
 - monitoring, 308
 - total bytes /sec, 310
 - % used and % used peak, 309
- parallel site collection upgrades, limiting downtime
 - using, 268–269
- Parser Profiles, 383
- pass-through authentication, in BCS, 202
- pass-through disks, in virtual servers, 322
- passwords
 - adding to web test fixed, 403

path-based site collections

- managing through SharePoint Managed Account, 238–239
- path-based site collections, 7, 10–11
- path-based sites, 198
- peak concurrent user percentage, in setting load test standards, 396
- People Picker control
 - about, 206–207
 - SPTTrustedClaimProvider handling, 188
- % free space, disk, 309
- % idle time, disk, 309
- % processor time, 309, 310
- performance counters, monitoring, 308–311
- Performance Monitor (PerfMon), 315
- PerformancePoint service application, 31
- PerformancePoint service (PPS) application, 107
- PerformancePoint services
 - c2WTS and, 200, 201
 - database, 144
 - SSS and, 202
- performance testing environments, 388–393, 423–424
- Performance Web Test, creating, 401–403
- permissions
 - configuring user, 43–44
 - configuring user policies across zones, 204
 - managing app, 191
 - setting for service applications, 123
 - turning on Use Self-Service Site Creation right, 199–200
 - understanding group permissions, 240
- Peschka, Steve, 186
- physical architecture, in implementation project, 74
- physical spindle activity, 315
- pipebinding, 51–52
- pipe (|) symbol, piping output object to cmdlet, 51–52
- platform security
 - about, 213–214
 - encrypting communications
 - about, 218
 - between client and server, 219–222
 - Central Administration communications, 241–243
 - enabling SSL encryption between applications and SQL Server, 225–229
 - obtaining Certification Authorities, 219–220
 - planning for configuring SQL Server TDE, 231–233
 - server to server, 223
 - using IPsec and IKEv2, 229–231
 - installing SharePoint using least privileges, 233–239
- SharePoint installations
 - preventing, 214–215
 - reporting, 217
 - tracking, 215–217
- SharePoint Managed Accounts and
 - about, 238–240
 - understanding roles and group permissions, 239–240
 - user accounts and, 235–237
- Player.vimeo.com, 15
- Policy folder, 22
- port allocation, verifying, 380–381
- port numbers, reusing for web applications, 7
- ports and protocols chart for SharePoint 2013, 226
- ports critical to SharePoint servers, 223–224
- PowerPivot
 - database, 143–144
 - using with PPS, 107
- PowerShell
 - about, 37–39
 - associating security certificate using, 222
 - backup farm cmdlet, 344–345
 - benefits of, 39–41
 - blocking SharePoint installations using, 214–215
 - changing search service topology, 57
 - configuration of SQL Server aliases, 136–137
 - configuring People Picker control, 206
 - configuring ReadOnlyMaintenanceLink, 350
 - configuring user permissions, 43–44
 - converting Classic to Claims authentication mode, 181–182
 - creating Foundation Subscription Settings service application, 31
 - creating initial farm using, 29–30
 - creating web application using FBA, 186
 - deploying service applications by, 92, 94
 - deploying service applications into production environment, 96
 - discovering SharePoint-specific cmdlets, 52–55
 - disposal of objects, 56
 - enabling c2WTS, 201
 - enabling RevertToSelf, 203
 - enhancements, 42
 - exporting trust certificates, 120
 - health-focused cmdlets, 299

- implementing service applications, 95
- index of cmdlets, 54–55
- launching SQL Server Client Network Utility, 29
- management shell, 44–46
- managing users in WSS_SHELL_ACCESS database role, 239
- provisioning Usage and Health Data Collection service application, 304–305
- registering or deleting SharePoint Managed Accounts through, 238
- reporting SharePoint installations using, 217
- role of, 38–39
- scheduling background jobs, 42
- syntax of, 39
- tracking SharePoint installations using, 215–217
- troubleshooting using, 333
- updating cmdlet help references, 50–51
- upgrading My Site host, 254, 287, 288–290
- using comments, 58
- using in Central Administration, 10–11
- working with cmdlets, 48–52
- PowerShell for SharePoint Command Builder, Windows, 53–54
- PowerShell Integrated Scripting Environment (ISE)
 - about, 38, 45
 - Command Add-Ons for, 48–49, 52
- PPS (PerformancePoint service) application, 107
- % used and % used peak, paging file, 309
- processing
 - % processor time, 310
 - working set, 310
- processor
 - monitoring, 308
 - % processor time, 309
- Products Configuration Wizard, SharePoint 2013, 41
- Profile database, 30, 112, 142, 248, 264–266
- profiles, selecting for Windows PowerShell ISE, 47–49
- Project Server 2013, 31
- proxies, service application, 93, 94–95
- \$PSModuleAutoLoadingPreference variable, 42

Q

- query processing component, of Search service application, 110
- questions to ask in gathering implementation project requirements, 64, 83–88
- quota templates, in site collections, 12

R

- RBS (Remote BLOB Storage), 140, 147, 262, 340
- ReadOnlyMaintenanceLink, configuring, 350
- ReadOnlyMaintenanceLink property, 373
- Read-only mode
 - limiting downtime using, 267, 271–272
 - using in upgrading service application upgrades, 280
- Read-Only right, 191
- Read Only status bar, 253
- Realm component, of SAML token-based authentication, 186
- Recovery Point Objectives (RPO)
 - about, 340–341
 - in failover clustering, 354
- Recovery Time Objectives (RTO)
 - about, 340–341
 - in failover clustering, 354
- Recycle Bin feature, 348–349, 373
- red zone
 - in setting load test standards, 396
 - setting threshold values, 411
- Remote BLOB Storage (RBS), 140, 147, 262, 340
- RemoteSigned permissions policy, 44
- Remove-SPDiagnosticsCounter cmdlet, 311
- Remove-SPShellAdmin cmdlet, 240
- replication support, 126
- reporting
 - creating deploying and managing, 300
 - Site Settings Storage Metrics report, 313
 - Slowest Pages report, 311
 - troubleshooting using Health Analyzer, 336, 338
 - View Analytics reports option, 307
 - viewing health reports, 299
- Reporting Server TempDB, 144
- Report Server Alerting database, 145
- Report Server Catalog database, 144, 145
- Request Management (RM) service
 - about, 329
 - configuration elements, 330–331
 - issues concerning configuring of, 338
 - SPRequestModule handling, 34
- Requests Per Second (RPS)
 - calculating number of, 387–388
 - in setting load test standards, 396, 398–399
 - monitoring, 299, 299–303
- requests queued, ASP.NET, 310
- requests rejected, ASP.NET, 310
- requests wait time, ASP.NET, 310

requirements, gathering implementation project

- requirements, gathering implementation project
 - about, 62–64
 - business intelligence requirements, 86
 - deliverables
 - about, 66–67
 - content types, 68
 - functional requirements, 67
 - governance plan (Internet usage policy) and documentation, 69, 84
 - site structure, 68–69
 - taxonomy and metadata, 68
 - usage scenarios, 67
 - hardware and software requirements, 75–76
 - hybrid architectures
 - considerations, 77–80
 - requirements, 81–82
 - identifying business processes, 85
 - identifying stakeholders, 64–65
 - information architecture, 70–72, 84–85
 - logical architecture, 72–73
 - mapping objectives to functionality, 66
 - Office client role, 87
 - organizational objectives, 65–66
 - performance and reliability requirements, 87–88
 - physical architecture, 74
 - planning strategy
 - about, 63–64
 - for SharePoint solution, 83
 - prioritizing goals, 66
 - questions to ask in, 64, 83–88
 - security requirements, 85–86
 - SharePoint Online
 - considerations, 76–77
 - requirements, 82–83
 - tools and process, 66
 - understanding system requirements, 75
- Resources folder, 22
- resource throttling
 - backward-compatible event handlers, 314
 - change log, 314
 - daily time window for large queries, 314
 - http request monitoring and throttling, 314
 - list unique permissions threshold, 314
- restore features
 - about, 343–345
 - native, 344
 - recovering data from unattached content database, 348
- Recycle Bin, 348–349
- Restore-SPSite cmdlet, 346
- Restore-SPDeletedSite cmdlet, 349
- Restore-SPSite cmdlet, 346
- Restricted permissions policy, 44
- result codes, in performance testing using IE Dev Tools, 392
- result types, search features for site collection administrators, 13–14
- .resx entries, 21
- ReuseThread option, 48
- RevertToSelf authentication, in BCS, 203
- Review Problems And Solutions screen, 304
- RM (Request Management) service
 - about, 329
 - configuration elements, 330–331
 - issues concerning configuring of, 338
 - SPRequestModule handling, 34
- roles, understanding, 239
- routers, configuring, 321
- Routing Machines, Request Management
 - Configuration element, 330–331
- Routing Rules, Request Management Configuration element, 330–331
- Routing Targets (Servers), Request Management Configuration element, 330–331
- RPO (Recovery Point Objectives)
 - about, 340–341
 - in failover clustering, 354
- RPS (Requests Per Second)
 - calculating number of, 387–388
 - in setting load test standards, 396, 398–399
 - monitoring, 299–303
- RP-STS (Relying Party Security Token Service), 186
- RTO (Recovery Time Objectives)
 - about, 340–341
 - in failover clustering, 354
- Run Settings window, in New Load Test Wizard, 410

S

- S2S (Server-to-server) authentication
 - about, 193
 - encrypting, 223
 - using STS, 116
- Samaj, Marek, 224
- SAML (Security Assertion Markup Language)
 - authentication, 186–189
 - c2WTS and, 201
 - sticky sessions and, 185
 - token-based authentication, 186

- SAML Token 1.1, 184
- sample site (site collection), creating, 57
- sandbox solutions
 - about, 27–28
 - extending rights within site collections to, 13
 - joining farm solutions and, 26
- SAN storage
 - monitoring capabilities of specific, 315
 - support for, 147
 - testing network drives, 166
- SCSI disks, using virtual, 322
- search administration component, of Search service application, 110
- Search Administration database, 30, 110, 140, 280
- Search Center, upgrading, 281–282
- Search farm, incorporating after SharePoint 2013 federated, 292
- Search Index database, 248
- search query rules, search features for site collection administrators, 14
- search result sources, search features for site collection administrators, 14
- search schema, search features for site collection administrators, 13–14
- Search service application, 96, 108–110, 117
- Search service application, SharePoint 2013, 41, 57, 140, 280–281
- search settings, search features for site collection administrators, 13–14
- search topology, showing current status of, 57
- secondary stakeholders, 68
- Secure Sockets Layer (SSL)
 - Central Administration over, 238, 240–241, 243
 - certificates, 219–220, 220
 - configuring web application to use, 7
 - enabling encryption between applications and SQL Server, 225–229
- Secure Store database, 141, 248
- Secure Store Service (SSS) application
 - about, 30
 - BCS and, 202–203
 - mapping as hybrid architecture requirement in implementation project, 82
 - service federation and, 117
- Secure Token Service (STS)
 - as WCF service endpoint, 94, 115
 - consuming federation metadata endpoint in JSON format, 194
 - user authentication using, 115–116
- security
 - for lists, 17
 - platform
 - about, 213–214
 - enabling SSL encryption between applications and SQL Server, 225–229
 - encrypting Central Administration communications of, 241–243
 - encrypting communication between client and server, 219–222
 - encrypting communication server to server communication, 223
 - encrypting communication using IPsec and IKEv2, 229–231
 - installing SharePoint using least privileges, 233–239
 - obtaining Certification Authorities, 219–220
 - planning for configuring SQL Server TDE, 231–233
 - preventing SharePoint installations, 214–215
 - reporting SharePoint installations, 217
 - SharePoint Managed Accounts and, 238–240
 - tracking SharePoint installations, 215–217
 - understanding roles and group permissions, 239–240
 - user accounts and, 235–237
- Security Assertion Markup Language (SAML)
 - authentication, 186–189
 - claims and c2WTS, 201
 - token-based authentication and sticky sessions, 185
- security requirements, in implementation project, 85–86
- SecurityTokenServiceApplicationPool, 234
- Self-Hosted App deployment model, 27
- Self-Service Site Creation feature, 11
- Self-Service Site Creation (SSSC) feature, enabling, 199–200
- Self-Signed Certificate, 219
- servers, SharePoint
 - accessing SharePoint Management Shell, 45–46
 - configuring for delegation, 177
 - encrypting communications, 223–224
 - encrypting communications between client and, 219–222
 - IPsec encrypting traffic to and from, 229
 - ports critical to, 223–224, 226
 - running cmdlets on, 45–48
- Server-to-server (S2S) authentication
 - about, 193
 - encrypting, 223

Service Account, changing

- using STS, 116
- Service Account, changing, 324
- service accounts, Central Administration's
 - Configure, 201
- service application endpoints, STS as WCF, 94, 115–116
- service application proxy, 92
- service applications
 - about, 92
 - Access Services web service application, 100–101, 105
 - Application Management, 101–102
 - application pool for, 9
 - architecture
 - about, 91
 - key concepts of, 92–93
 - Business Data Connectivity (BDC) service, 31, 105
 - communication sites for, 93
 - components of, 92–93
 - databases and, 96
 - databases for, 30–32
 - Excel Services, 106
 - exploring service federation for, 117–119
 - implementing, 95
 - Machine Translation, 102–103
 - Managed Metadata, 106–107
 - mapping to proxy groups, 93, 94–95
 - Office Web Applications server and, 96–98
 - PerformancePoint service, 107
 - provisioning and publishing, 122
 - Search, 108–110
 - Secure Token, 115–116
 - setting permissions for, 123
 - Topology, 115
 - upgrading, 254, 255, 280–281, 283–287
 - User Profile Application, 111–112
 - Visio Graphics, 112–113
 - web content applications and, 9
 - Word Automation, 113–114
 - workflows for, 98–100
 - Work Management, 103–104
- service consumer, 92
- Service-Level Agreements (SLAs), 340–341, 370
- service machine instance, 92
- Service Master Key, 231–232
- Service Principal Names (SPNs), guidelines for configuring, 177–178
- service proxy groups, 92
- services
 - about, 92
 - configuring, 93
 - deploying, 92
- services.cnf files, 21
- services farm
 - building, 119–124
 - implementing, 117–119
- Set-SPCentralAdmin cmdlet, 242
- Set-SPFarmConfig cmdlet, 217
- Set-SPRequestManagementSettings cmdlet, 331
- Set-SPUsageApplication cmdlet, 305
- Set-SPUsageService cmdlet, 305
- Shared Services Farm, upgrading to SharePoint 2013, 255–256
- SharePoint
 - about, 61
 - as trusted subsystem, 171
 - hardware and software requirements in
 - implementation project, 75–76
 - installations
 - preventing, 214–215
 - reporting, 217
 - tracking, 215–217
 - Operating Systems versions supported by
 - new, 353
 - pipeline, 32–35
 - ports and protocols chart, 226
- SharePoint14Module, 34
- SharePoint 2013 PowerShell cmdlets, 37
- SharePoint 2013 Products Configuration Wizard, 41
- SharePoint 2013 Search service application, 41, 57, 96, 108, 117, 140
- SharePoint API, using in Central Administration, 10–11
- SharePoint App model
 - about, 17, 27
 - joining farm and sandbox solutions, 27
- SharePoint Central Administration v4 application pool, 234
- SharePoint_Config database, 29
- SharePoint_Content_Admin database, 29
- SharePoint databases. *See also* databases
 - about, 10
 - Analytics Reporting, 31, 110, 141
 - App Management, 140
 - attach upgrade, 247–248
 - Business Data Connectivity, 140

- Central Administration content, 139, 146
- configuration, 139, 146
- content, 139–140, 146, 256–257, 280, 287–288
- Crawl, 31, 110, 141
- Link, 31, 110, 141
- Machine Translation Services, 143
- Managed Metadata, 143
- PerformancePoint services, 144
- PowerPivot, 143–144
- Profile, 30, 112, 142, 264, 285
- Search Administration, 30, 110, 140
- Search service application, 140
- Secure Store, 141
- Social, 143, 264–266, 285
- SQL Server aliases, 29, 133–137
- State Service, 143
- Subscription Settings, 142
- supported, 247–248
- Synch, 30, 112, 142, 248, 264–266, 285
- system databases, 29–30
- unsupported, 248
- Usage, 142, 307, 308
- user credentials stored in LDAP, 185
- SharePoint Developer Dashboard
 - about, 379–380
 - analyzing performance of webpages using, 300
 - enabling/disabling, 312
 - monitoring page performance using, 311–312
- SharePoint Farm Configuration Wizard
 - for configuring usage and health provider settings, 304
 - using, 41
- SharePoint farms. *See* farm
- SharePoint Foundation 2013
 - databases
 - about, 138
 - Analytics Reporting, 31, 110, 141
 - App Management, 140
 - Business Data Connectivity, 140
 - Central Administration content, 139, 146
 - configuration, 139, 146
 - content, 139–140, 146, 256–257, 280, 287–288
 - Crawl, 31, 110, 141
 - Link, 31, 110, 141
 - Search Administration, 30, 110, 140
 - Search service application, 140
 - Secure Store, 141
 - Subscription Settings, 142
 - Usage, 142, 307, 308
 - SharePoint Foundation Subscription Settings service
 - application, 31
 - SharePoint Foundation Web Application (SPFWA)
 - service, 329
 - SharePoint Health Analyzer
 - about, 299
 - correcting index fragmentation, 319
 - rules, 303–304, 318
 - troubleshooting using reports from, 336, 338
 - SharePoint-Hosted App deployment model, 27
 - SharePoint Managed Accounts, for application pool, 9
 - SharePoint Online, in implementation project
 - considerations for, 76–77
 - requirements, 82–83
 - SharePoint Online Management Shell environment,
 - setting up, 44–46
 - SharePoint Online Secure Store service, 82
 - SharePoint Products and Configuration Wizard,
 - creating initial farm using, 29–30
 - SharePoint Products Network Monitor Parsers, Office and, 383
 - SharePoint Root folder, IIS folders located in, 22–23
 - SharePoint Search service application, databases
 - for, 30
 - SharePoint Secure Store application pool, 234
 - SharePoint Server 2010
 - upgrading to SharePoint 2013
 - about, 245–247
 - attach upgrade, 247–248
 - authentication upgrade, 247
 - Classic mode authentication, 266–267
 - cleaning content databases, 256–257
 - cleaning environment, 257–258
 - content database upgrade, 256–257, 280, 287–288
 - deferred site collection upgrade, 249
 - deploying .wsp files, 287
 - documenting current settings, 258–260
 - documenting environment, 258
 - documenting environment settings, 260
 - documenting services, 263–266
 - documenting service settings, 260
 - email notifications during, 253
 - evaluation site collections upgrade, 250–251
 - feature fallback behavior, 249
 - federating search services, 282–283
 - finding .wsp files, 262
 - implementing upgrade, 271–273
 - in-place upgrades, 247

SharePoint servers

- language packs and, 267
 - learning from deployment, 271
 - limiting downtime during, 267–269, 271–272
 - managing customizations, 261–262
 - manually deploying code and updates, 262
 - monitoring upgrade process, 273
 - process overview of, 253–256
 - RBS and, 262
 - Search Center upgrade, 281–282
 - service application upgrades, 255, 280–281, 283–287
 - Shared Services Farm and, 255
 - Site Collection Health Check, 249–250, 290–292
 - Site Collection Upgrade Throttling, 250–251
 - stages and responsibilities, 254
 - System Status Bars, 252–253
 - testing upgrade, 269–271, 278–279
 - third-party products and, 262
 - troubleshooting upgrade, 274
 - ULS logging format support, 253
 - upgrading production farm, 272–273
 - validating upgrade, 274
- Visual upgrade, 249
- SharePoint Server 2013
- databases
 - about, 138, 142
 - Analytics Reporting, 31, 110, 141
 - App Management, 140
 - attach upgrade of, 247–248
 - Business Data Connectivity, 140
 - Central Administration content, 139, 146
 - configuration, 139, 146
 - content, 139–140, 146, 256–257, 280, 287–288
 - Crawl, 31, 110, 141
 - Link, 31, 110, 141
 - Machine Translation Services, 143
 - Managed Metadata, 143
 - PerformancePoint services, 144
 - PowerPivot, 143–144
 - Profile, 142, 264–266, 285
 - Search Administration, 30, 110, 140
 - Search service application, 140
 - Secure Store, 141
 - Social, 143, 264–266, 285
 - State Service, 143
 - Subscription Settings, 142
 - supported, 247–248
 - Synch, 30, 112, 142, 248, 264–266, 285
 - Synchronization, 142
 - unsupported, 248
 - Usage, 142, 307, 308
 - upgrading to
 - attach upgrade of, 247–248
 - authentication upgrade, 247
 - Classic mode authentication, 266–267
 - cleaning content databases, 256–257
 - cleaning environment, 257–258
 - content database upgrade, 256–257, 280, 287–288
 - deferred site collection upgrade, 249
 - deploying .wsp files, 287
 - documenting current settings, 258–260
 - documenting environment, 258
 - documenting environment settings, 260
 - documenting services, 263–266
 - documenting service settings, 260
 - email notifications during, 253
 - evaluation site collections upgrade, 250–251
 - feature fallback behavior, 249
 - federating search services, 282–283
 - finding .wsp files, 262
 - implementing upgrade, 271–273
 - in-place upgrades, 247
 - language packs and, 267
 - learning from deployment, 271
 - limiting downtime during, 267–269, 271–272
 - managing customizations, 261–262
 - manually deploying code and updates, 262
 - monitoring upgrade process, 273
 - process overview of, 253–256
 - RBS and, 262
 - Search Center upgrade, 281–282
 - service application upgrades, 255, 280–281, 283–287
 - Shared Services Farm and, 255
 - Site Collection Health Check, 249–250, 290–292
 - Site Collection Upgrade Throttling, 250–251
 - stages and responsibilities, 254
 - supported databases, 247–248
 - System Status Bars, 252–253
 - testing upgrade, 269–271, 278–279
 - third-party products and, 262
 - troubleshooting upgrade, 274
 - ULS logging format support, 253
 - unsupported databases, 248
 - upgrading production farm, 272–273

- validating upgrade, 274
- SharePoint servers
 - accessing SharePoint Management Shell, 45–46
 - configuring for delegation, 177
 - encrypting communications, 223–224
 - encrypting communications between client and, 219–222
 - IPsec encrypting traffic to and from, 229
 - ports critical to, 226
 - running cmdlets on, 45–48
- SharePoint_Shell_Access role, adding user to, 43
- SharePoint snap-in, loading, 45
- SharePoint ULS Log Viewer, 333
- SharePoint Web Content Default application pool, 234
- SharePoint Web Services Default, application pool, 234
- sharing feature, AuthZ, 208
- Show-Command cmdlet, 49–50
- Shredded Storage, 160, 161, 262
- shrink operation, SQL Server, 319
- SignInURL parameter, 189
- Single Sign On (SSO) feature, 78, 79
- site architecture, in implementation project, 71
- Site Collection Administration options, Site Settings
 - Storage Metrics report in, 313
- site collection administrator
 - granting user rights, 13–14
 - HTML field security, 15
 - picking objects for web application policies, 206
 - recycle bin, 349
 - search features for, 13
- Site-collection backup feature, 345–346
- Site Collection Health Check, 249–250, 290–292
- Site Collection Recycle Bin, 348–349
- site collections
 - about, 10–14
 - host headers, 7, 200
 - path-based, 7
 - quota templates in, 12
 - sites and, 15
 - users rights to, 13–14
- site collection (sample site), creating, 57
- Site Collection Upgrade Throttling, 251–252
- Site Pages library, creating landing page, 373
- Site Settings Storage Metrics report, 313
- Site/Site collection, as hybrid architecture
 - requirement in implementation project, 82
- site structure
 - as implementation project deliverable, 68–69
 - in information architecture, 70
- sites (webs)
 - export of, 347
 - import of, 347
 - in site collection, 15
 - vs. site collections, 16
- SiteTemplates folder, 23
- Skydrive.live.com, 15
- SLAs (Service-Level Agreements), 340–341, 370
- Slowest Pages report, 311
- snap-in, SharePoint, loading, 45
- Social database, 143, 248, 264–266
- Social Tagging database, 30, 112
- software and hardware requirements, in
 - implementation project, 75–76
- spAdmin, user account, 235
- spC2Wts, user account, 235
- spConten, user account, 235
- spCrawl, user account, 236, 237
- spFarm, user account, 235, 236
- spInstall, user account, 235, 236
- SPNs (Service Principal Names), guidelines for
 - configuring, 177–178
- SPRequestModule, 34
- SPRequest objects, 56
- spServices, user account, 235, 237
- SPSite objects, disposal of, 56
- spSSS, user account, 236
- spSuperReader, user account, 236
- spSuperUser, user account, 236
- SPTTrustedClaimProvider, handling People Picker
 - control, 188
- SPTTrustedIdentityTokenIssuer component, of SAML
 - token-based authentication, 186
- SPTTrustedIdentityTokenIssuer, diagram of
 - object, 189–190
- spUnattend, user account, 236, 237
- spUPS, user account, 235
- SPWeb objects, disposal of, 56
- SQL Client Configuration Utility (cliconfg.exe),
 - configure SQL Server aliases using, 133–134
- SQL databases, prerequisites for Access Services web
 - service application, 101
- SQL folder, 23
- sqlInstall, user account, 235
- SQLIO
 - about, 162–168
 - analyzing I/O subsystem using, 145, 315

- stress testing SQL Server drives, 168
- testing iSCSI network drives, 166–167
- testing SQL Server drives, 167–168
- testing SQL Server internal drives, 162
- SQL Server
 - aliases, 29–30, 133–137
 - AlwaysOn Availability feature of, 363–367
 - business intelligence
 - enabling, 147–165
 - support for, 127
 - creating Foundation Subscription Settings service application, 31
 - database mirroring, 9
 - enabling SSL encryption between applications and, 225–229
 - HA/DR options using, 126–128
 - initial optimization and configuration of
 - about, 128
 - autocreating statistics, 132
 - collation, 128–130
 - managing growth of data and log files, 137–138
 - MDOP, 130–132
 - monitoring storage and performance, 138
 - using aliases, 133–137
 - installation for implementation project, 128
 - IPsec encrypting traffic to and from, 229
 - maintenance plan, 319–320
 - optimization of, 315–317
 - performance counters, 310
 - planning for configuring transparent data encryption on, 231–233
 - shrinking data files on, 319
 - storage and IOPS, 145–147
 - stress testing drives on, 168
 - tempdb database, 145
 - testing drives on, 167–168
 - testing internal drives, 162
 - using aliases, 135–136
 - versions supported by new SharePoint, 353
 - versions supported by SharePoint, 126–128
- SQL Server BI (Business Intelligence)
 - enabling, 147–165
 - support for, 127
- SQL Server Client Network Utility, launching, 29
- SQL Server Configuration Manager
 - configure SQL Server alias using, 135–136
 - viewing protocols for MSSQLSERVER
 - properties, 227–228
- SQL Server Management Studio (SSMS), testing
 - aliases using, 135
- SQL Server PowerPivot service application, 31, 143–144
- sqlService, user account, 235
- SSL (Secure Sockets Layer)
 - Central Administration over, 238, 240–241, 243
 - certificates, 219–220, 220
 - configuring web application to use, 7
 - enabling encryption between applications and SQL Server, 225–229
- SSMS (SQL Server Management Studio), testing
 - aliases using, 135
- SSO (Single Sign On) feature, 78, 79
- SSRS Report Server Catalog database, 144
- SSRS (SQL Server Reporting Services),
 - Microsoft, 144–145, 300
- SSSC (Self-Service Site Creation) feature,
 - enabling, 199–200
- SSS (Secure Store Service) application
 - about, 31, 117
 - BCS and, 202, 202–203, 203
 - mapping as hybrid architecture requirement in implementation project, 82
 - service federation and, 117
- stakeholders, identifying implementation, 64–65
- standard SSL certificates, 219–220
- Start-SPAssignment -Global cmdlet, 56
- State service application, 31, 304–307
- State Service database, 143
- Step Load test settings, in New Load Test Wizard, 416
- Stop-SPAssignment- Global cmdlet, 56
- storage, SharePoint, monitoring, 138, 312–313
- stress testing environments
 - about, 414
 - creating, 415–417
 - error in, 415
- stsadm.exe command-line utility, 37, 40
- STS (Secure Token Service)
 - as WCF service endpoint, 94, 115
 - consuming federation metadata endpoint in JSON format, 194
 - user authentication using, 115–116
- Subscription service application, 304–307
- Subscription Settings database, 142
- Sun ONE, 185
- switches, configuring, 321
- Synchronization Connections, 335

Synchronization (SYN) database, 30, 112, 142, 248, 264–266, 285
 syntax, of cmdlets, 39
 System Center 2012, 300
 System Center Management Pack, Operations Manager with, 300
 system requirements, understanding in implementation project, 75
 System Status Bars and notifications, 252–253

T

Task Scheduler tasks, 42
 taxonomy
 and metadata as implementation project deliverable, 68
 in information architecture, 70
 TDE (Transparent Data Encryption), planning for configuring SQL Server, 231–233
 tempdb database, SQL Server, 145
 TEMPLATE folder, 22
 test farm, upgrading, 269–270
 testing
 iSCSI network drives, 166–167
 SQL Server drives, 167–168
 SQL Server internal drives, 162
 upgrade to SharePoint 2013, 269–271, 278–279
 testing environments, 383–416
 about, 383
 documenting effectiveness of changes, 428–429
 IIS logs
 combing through, 385–388
 inspecting, 384–385
 load
 about, 394
 building demo, 417–418
 building load test, 403–413, 424–427
 creating Performance Web Test, 401–403
 determining capacity of farm, 397–398
 notification when breaching thresholds, 411
 options for, 394–395
 setting standards for, 396
 setting up Visual Studio Ultimate 2012 in, 398–400
 Visual Studio Ultimate in, 417
 performance, 388–393, 423–424
 stress
 about, 414
 creating, 415–417
 error in, 415
 verifying sites, 420–421
 Test Mix Model, selecting in New Load Test Wizard, 405, 416
 Test Mix window, in New Load Test Wizard, 407
 Test-SPContentDatabase cmdlet, 256–257, 282, 287, 289
 THEMES folder, 23
 Think Time Profile, selecting in New Load Test Wizard, 404
 throttling, resource, 314–315
 Throttling Rules, Request Management Configuration element, 330–331
 Throttling, Site Collection Upgrade, 251–252
 timer jobs, 299
 Timings window, in performance testing, 393
 Token-signing certificate component, of SAML token-based authentication, 186, 188
 Topology serv2ice, 121
 Topology service, 115
 total bytes /sec, paging file, 310
 total user count, in setting load test standards, 396
 Transparent Data Encryption (TDE), planning for configuring SQL Server, 231–233
 troubleshooting
 help with VSTS, 418
 Kerberos tickets, 382
 SharePoint environment, 332–333
 upgrading to SharePoint 2013, 274
 Usage And Health Data Collection Service application, 379
 using Fiddler, 381
 trust, creating between farms, 118, 120–121, 282–283
 Trusted Identity Provider authentication, 8
 trusted subsystem authentication, 171
 turning and optimizing environment
 fine-tuning network performance, 320–322
 planning and configuring caching, 322–329
 Request Management service, 329–331
 resource throttling, 314–315
 SQL optimization, 315–317. *See also* optimizing, SQL Server
 using database maintenance plan, 317–320
 two-way hybrid search architecture, 79–80
 two-way server-to-server trust, in hybrid architecture, 78, 79

U

- UCC Certificates, 219
- ULS Log Viewer, 333
- ULS (Unified Logging Service) logs
 - about format, 253
 - in Central Administration, 22
 - troubleshooting using, 336
 - using Developer Dashboard with, 312, 379–380
 - viewing, 376–379
- unattached content database, recovering data
 - from, 348
- Unicast method, 369
- UPA (User Profile Application)
 - about, 111–112
 - as hybrid architecture requirement in
 - implementation project, 82
 - database, 30–31
 - dependency of S2S on, 193
 - service application as, 96
 - service federation and, 117
- Update-Help cmdlet, 42, 50–51
- Update-SPDistrib cmdlet, 323–324
- Upgrade Available status bar, 252
- Upgrade Evaluation Site Collection status bar, 253
- Upgrade In Progress status bar, 253
- Upgrade-SPSite cmdlet, 289–290
- upgrading to SharePoint 2013
 - about, 245–247
 - attach upgrade, 247–248
 - authentication upgrade, 247
 - Classic mode authentication, 266–267
 - cleaning content databases, 256–257
 - cleaning environment, 257–258
 - deferred site collection upgrade, 249
 - documenting
 - current settings, 258–260
 - environment, 258
 - environment settings, 260
 - services, 263–266
 - service settings, 260
 - email notifications during, 253
 - evaluation site collections upgrade, 250–251
 - feature fallback behavior, 249
 - federating search services, 282–283
 - implementing upgrade, 271–273
 - in-place upgrades, 247
 - language packs and, 267
 - learning from deployment, 271
 - limiting downtime during, 267–269, 271–272
 - managing customizations
 - about, 261–262
 - content database upgrade, 256–257, 280, 287–288
 - deploying .wsp files, 287
 - finding .wsp files, 262
 - manually deploying code and updates, 262
 - RBS and, 262
 - third-party products and, 262
 - monitoring upgrade process, 273
 - process overview of, 253–256
 - Search Center upgrade, 281–282
 - service application upgrades, 255, 280–281, 283–287
 - Shared Services Farm and, 255
 - Site Collection Health Check, 249–250, 290–292
 - Site Collection Upgrade Throttling, 250–251
 - stages and responsibilities, 254
 - supported and unsupported databases, 247–248
 - System Status Bars, 252–253
 - testing upgrade, 269–271, 278–279
 - troubleshooting upgrade, 274
 - ULS logging format support, 253
 - upgrading production farm, 272–273
 - validating upgrade, 274
- UPN (User Principal Name), c2WTS and, 200
- UPS (User Profile Service)
 - importing .bin file and, 265
 - upgrading, 264
- URLs
 - application pool and, 8
 - host headers allowing for specification of, 7
 - in web application, 8
- Usage And Health Data Collection Service
 - application, 30, 142, 304, 379
- Usage database, 142, 307, 308
- usage scenarios
 - as implementation project deliverable, 67
 - in information architecture, 70
- UserAccountDirectoryPath, 207
- user accounts
 - platform security and, 235–237
 - recommended, 235–236
- user authentication. *See also* authentication
 - using STS, 115–116

UserCode folder, 24
 user credentials, storing in LDAP database, 185
 usernames, adding to web test fixed, 403
 user permission policy, allowing configuring of user policies across zones, 204
 user permissions, verifying, 102
 User Principal Name (UPN), c2WTS and, 200
 User Profile Application (UPA)
 about, 111–112
 as hybrid architecture requirement in implementation project, 82
 database, 30–31
 dependency of S2S on, 193
 service application as, 96
 service federation and, 117
 Synch database, 142
 User Profile Service (UPS)
 importing .bin file and, 265
 upgrading, 264
 User Profile Synchronization Server, importing .bin file and, 265
 users rights, to site collections, 13–14
 users, training user on metadata, 72
 Use Self-Service Site Creation right, turning on, 199–200

V

View Analytics reports option, 307
 Virtual Machines (VMs), as web servers, 321
 virtual servers, 321–322
 Visio
 refreshing diagrams in web browser, 112–113
 Visio Graphics service (VGS)
 about, 112–113
 c2WTS and, 200, 201
 SSS and, 202
 Visual Round Trip Analyzer (VRTA), 389–391
 Visual Studio Team System (VSTS) 2008 Team Suite,
 using for demo test load, 417–418
 Visual Studio (VS) Ultimate 2012
 in load testing environments, 417
 setting up in testing environment, 398–400, 423
 Visual upgrade, SharePoint Server 2010, 249
 VMs (Virtual Machines), as web servers, 321
 VRTA (Visual Round Trip Analyzer), 389–391
 VSTS (Visual Studio Team System) 2008 Team Suite,
 using for demo test load, 417–418

W

Warm standby redundancy method, 358
 WAS (Word Automation Service) application, 32, 113–114
 WAWS (Windows Azure Workflow Server), 98–100, 100
 WCF service endpoint, STS as, 94, 115
 web application policies, SharePoint, 203–205
 web applications
 about, 4–9
 adding content databases, 139
 configuring SPNs for, 178
 databases, 10
 in business intelligence dashboards, 107
 modifying HTTP modules, 33–34
 proxy groups assigned through service applications to, 93, 94–95
 site collections, 10–14
 web browser. *See also* Internet Explorer (IE)
 publishing within Excel documents within, 106
 refreshing Visio diagrams in web browser, 112–113
 using Get-Help cmdlet to open topics in, 42
 WebClient folder, 24
 web.config files
 about, 21
 changing, 19
 importance of, 32–35
 in WebServices folder, 24
 making changes in IIS web application zone, 198
 web content applications, service applications and, 9
 webpages
 analyzing performance of, 300
 debugging webpages, 381
 web parts, query to find, 337
 web requests, verifying, 380–381
 web servers, virtual machines as, 321
 web service applications
 Access Services, 100–101
 Office Web Applications server, 96–98
 WebServices folder, 24
 web sessions, recording, 417
 webs (sites)
 export of, 347
 import of, 347
 in site collection, 15
 vs. site collections, 16
 .webtest files, 401

weekly processes, in monitoring environment

- weekly processes, in monitoring environment, 302–303
- WIF (Windows Identity Foundation)
 - about, 184
 - c2WTS as feature to extract UPN of, 200
- wildcard SSL certificates, 219–220
- Wilén, Wictor, 194–195
- WinDiff, 262–263
- Windows authentication
 - AD DS and, 184–185
 - Basic protocol, 172–173
- Windows Azure Access Control Service (ACS)
 - about, 27
 - authorization server for cloud-hosted apps, 191–193
 - for SharePoint Online, 83
 - SSS and, 202
- Windows Azure Workflow Server (WAWS), 98–100, 100
- Windows Command Prompt (cmd.exe), 40
- Windows Event Log, troubleshooting using, 335
- Windows Identity Foundation (WIF)
 - about, 184
 - c2WTS as feature to extract UPN of, 200
- Windows Management Framework 3.0, 44
- Windows Management Instrumentation (WMI), 300
- Windows NT LAN Manager (NTLM) credentials
 - about, 8
 - credentials in Windows, 173–174
 - pass-through authentication, 202
 - search crawl the web application with AD DS, 198
 - vs. Kerberos, 320–321
- Windows NT Server 4.0, IP load balancer with, 368–369
- Windows Operating Systems, versions supported by SharePoint 2013, 353
- Windows PowerShell
 - about, 37–39
 - associating security certificate using, 222
 - benefits of, 39–41
 - blocking SharePoint installations using, 214–215
 - changing search service topology, 57
 - configuration of SQL Server aliases, 136–137
 - configuring People Picker control, 206
 - configuring ReadOnlyMaintenanceLink, 350
 - configuring user permissions, 43–44
 - converting Classic to Claims authentication mode, 181–182
 - creating Foundation Subscription Settings service application, 31
 - creating web application using FBA, 186
 - deploying service applications by, 92, 94
 - deploying service applications into production environment, 96
 - discovering SharePoint-specific cmdlets, 52–55
 - disposal of objects, 56
 - enabling c2WTS, 201
 - enabling RevertToSelf, 203
 - enhancements, 42
 - exporting trust certificates, 120
 - health-focused cmdlets, 299
 - implementing service applications, 95
 - index of cmdlets, 54–55
 - management shell, 44–46
 - managing users in WSS_SHELL_ACCESS database role, 239
 - New-SPTrustedSecurityTokenIssuer cmdlet, 193
 - New-SPWebApplicationExtension cmdlet, 197
 - provisioning Usage and Health Data Collection service application, 304–305
 - registering or deleting SharePoint Managed Accounts through, 238
 - reporting SharePoint installations using, 217
 - role of, 38–39
 - scheduling background jobs, 42
 - syntax of, 39
 - tracking SharePoint installations using, 215–217
 - troubleshooting using, 333
 - updating cmdlet help references, 50–51
 - upgrading My Site host, 254, 287, 288–290
 - using comments, 58
 - using in Central Administration, 10–11
 - working with cmdlets, 48–52
- Windows PowerShell for SharePoint Command Builder, 53–54
- Windows PowerShell Integrated Scripting Environment (ISE)
 - about, 45, 45–48
 - Command Add-Ons for, 48–49, 52
- Windows Script Host (WSH) scripts, 40
- Windows Task Scheduler, calling Windows PowerShell scripts, 344
- wkpsstd.aspx page, 23
- WMI (Windows Management Instrumentation), 300
- WMS (Work Management service) application, 103–104
- Word Automation Service (WAS) application, 31, 113–114
- Word Conversion database, 31
- Word, conversion of documents in, 113–114

- workflow
 - components, 99
 - service, 98
- WorkflowActivities folder, 23
- working set, processing, 310
- Work Management service (WMS), 103–104
- wpresources/_wpresources folders, 21
- Wreply parameter, 189
- Write right, 191
- WS-Federation 1.1, 184
- WS-Federation Passive Requestor Profile (WS-F RPR),
 - in SAML token-based authentication, 186
- .wsp files
 - deploying, 287
 - exporting, 262
- WSS_ADMIN_WPG group, 240
- WSS_Admin_WPG group, add user to, 43
- WSS_CONTENT_APPLICATION_POOLS database
 - role, 239
- WSS_Content database, 29
- WSS_RESTRICTED_WPG, 240
- WSS_SHELL_ACCESS database role, 239
- WSS_WPG group, 240
- WS-Trust 1.4, 184

Y

- Young, Shane, exporting .wsp files, 262
- Youtube.com, 15
- Youtube-nocookie.com, 15

Z

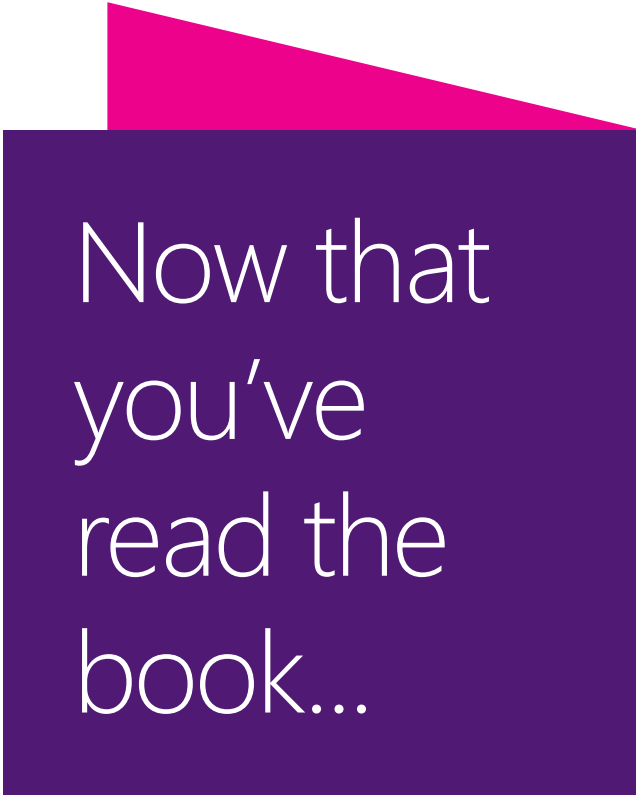
- zones, SharePoint
 - alternate access mappings and, 198
 - associated to web application, 8
 - authentication, 197–198
 - configuring user policies across, 204

About the authors

SHANNON BRAY, MCM, MCT, MCS, MCSE, MCDBA, MCITP, MCPD, is a Microsoft Certified Solution Master (MCSM) for Microsoft SharePoint. Shannon has been working with the SharePoint platform since 2006 and has presented SharePoint topics at Microsoft's TechED, TechReady, and SharePoint Conference. Professionally, Shannon serves his company as the chief SharePoint architect and has vast experience in both the infrastructure and development sides of the product. This is Shannon's second book; he coauthored *Automating SharePoint 2010 Administration with Windows PowerShell 2.0*. You can reach Shannon on Twitter: @noidentity29. Shannon is a former military diver, and when he is not working with SharePoint, he enjoys diving off the coast of North Carolina and exploring shipwrecks.

PATRICK CURRAN, MCT, MCTS, MCP, MCITP, MCPD, MCSA, is currently the director for the Federal Group at Planet Technologies, where he is also a senior SharePoint architect. Patrick has been working with SharePoint since 2003, doing everything from administration, branding, and development to upgrading and troubleshooting. Patrick works with clients from all over the world and has been designing and deploying farms of varying complexity over the years, especially within the Federal realm, where nothing is small or simple. Patrick has spoken at SharePoint Saturday events all over the country, and he can be reached through Twitter: @PCfromDC. Patrick still enjoys getting out onto the football (soccer) pitch and kicking the ball around with the wee-ones when possible.

MIGUEL WOOD, MCM, MCT, MCITP, MCPD, MCSE, MCS, MCCDBA, is an MCSM for SharePoint and currently fills the roles of director, Central Region, and chief architect for Planet Technologies. Prior to his current roles, he served as president and CEO of TekFocus, a worldwide provider of specialized Microsoft training and services. With his technical and business experience, he provides enterprise expertise to clients of all sizes. Miguel is an active presenter, delivering keynotes to technical sessions at many Microsoft technology conferences, covering SharePoint, SQL, ASP.NET, and other topics. Miguel currently resides in Dallas, Texas, with his wife and two sons. Miguel is a U.S. Army Airborne Infantry veteran, studied electrical and computer engineering at LSU, and enjoys singing with live rock bands when not working or learning new technologies. Miguel can be reached on Twitter, @miguelwood, and other social media sites, including Facebook and LinkedIn.



Now that
you've
read the
book...

Tell us what you think!

Was it useful?

Did it teach you what you wanted to learn?

Was there room for improvement?

Let us know at <http://aka.ms/tellpress>

Your feedback goes directly to the staff at Microsoft Press,
and we read every one of your responses. Thanks in advance!



Microsoft

