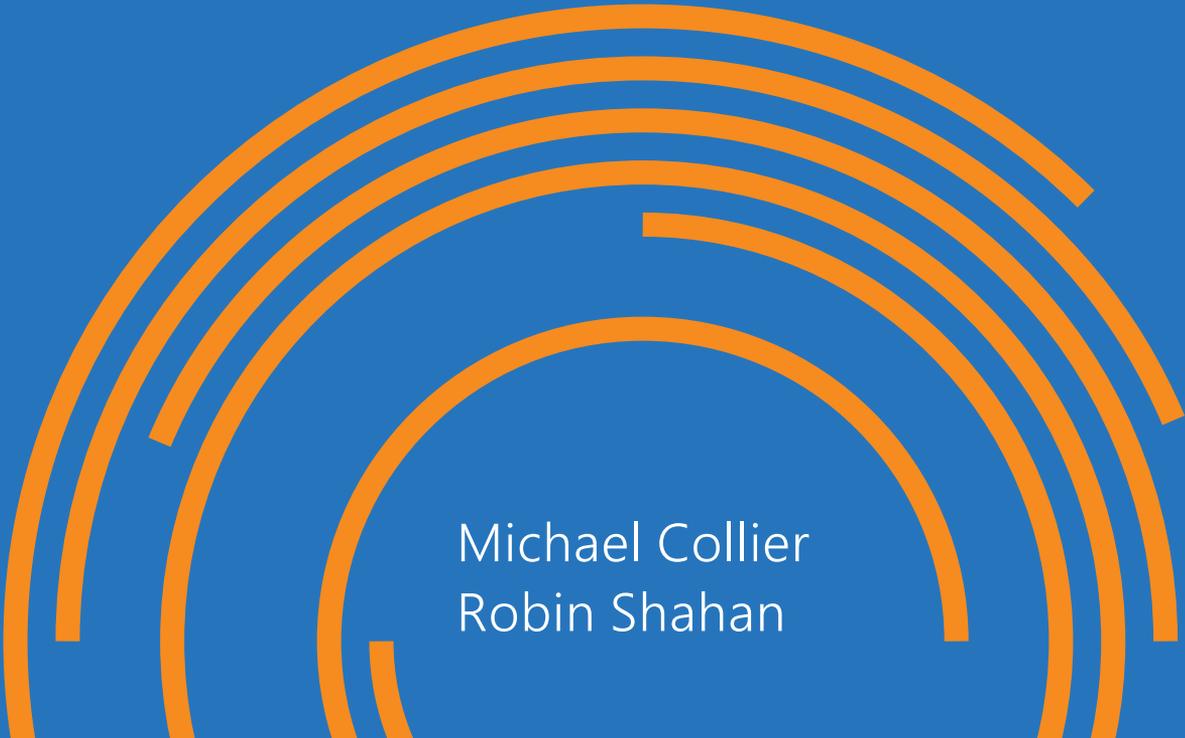


Fundamentals of Azure

Microsoft Azure Essentials



Michael Collier
Robin Shahan

PUBLISHED BY
Microsoft Press
A division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399

Copyright © 2015 Microsoft Corporation. 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: 978-0-7356-9722-5

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://aka.ms/tellpress>.

This book is provided “as-is” and expresses the authors’ views and opinions. The views, opinions, and information expressed in this book, including URL and other Internet website references, may change without notice.

Unless otherwise noted, the companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in examples herein are fictitious. No association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred.

Microsoft and the trademarks listed at <http://www.microsoft.com> on the “Trademarks” webpage are trademarks of the Microsoft group of companies. All other marks are property of their respective owners.

Acquisitions, Developmental, and Project Editor: Devon Musgrave

Editorial Production: nSight, Inc.

Copyeditor: Teresa Horton

Cover: Twist Creative

To my wife, Sonja, and son, Aidan; I could not have written this book without your immense support and patience.

—Michael S. Collier

I dedicate this book to my friend Carol Schultz, who passed away earlier this year. She always believed in me.

—Robin E. Shahan

Table of Contents

Foreword	11
Introduction	12
Who should read this ebook.....	12
Assumptions	12
This ebook might not be for you if.....	13
Organization of this ebook.....	13
Conventions and features in this ebook	14
System requirements.....	14
Acknowledgments.....	15
Errata, updates, & support.....	16
Free ebooks from Microsoft Press	16
We want to hear from you	16
Stay in touch.....	16
Chapter 1 Getting started with Azure	17
What is Azure?	17
Overview of cloud computing	17
Comparison of on-premises versus Azure	18
Cloud offering.....	19
SaaS: Software as a Service.....	19
PaaS: Platform as a Service.....	19
IaaS: Infrastructure as a Service	19
Azure services	20
Introduction to portals	20
Azure Preview Portal.....	21
Blades and journeys.....	22

Customizing the Startboard	27
Azure Management Portal	28
Subscription management and billing.....	32
Available subscriptions.....	32
Share administrative privileges for your Azure subscription	33
Add administrative privileges in the Azure Preview Portal.....	34
Add a co-administrator in Azure Management Portal	38
Pricing calculator.....	39
Billing	41
Chapter 2 Azure Websites and Azure Cloud Services.....	46
Creating and configuring websites	46
What is Azure Websites?	46
Creating a new website.....	47
Using the portal.....	47
Websites gallery.....	51
Configure and scale a website	53
Configuration.....	53
Scaling	59
Deploying and monitoring websites	61
Options for creating websites	61
Notepad or an HTML editor	61
WebMatrix	62
Visual Studio	62
Publishing a website from Visual Studio	63
Monitoring a website	65
PaaS Cloud Services.....	66
Creating a cloud service	66

Publishing a cloud service	69
Scaling and monitoring a cloud service.....	72
SCALE options.....	72
Monitoring.....	74
Miscellaneous points.....	77
Configuration in the portal	77
Production and staging slots in the portal	77
Worker roles.....	77
Chapter 3 Azure Virtual Machines.....	78
What is Azure Virtual Machines?.....	78
VM status.....	79
IP address	80
Billing.....	80
Create and configure virtual machines.....	80
Create a virtual machine with the Azure Preview Portal.....	81
Connect to a virtual machine.....	84
Configure disks.....	84
Caching.....	85
Attach a disk.....	85
Formatting disks	87
Disk performance	89
Endpoints.....	89
Virtual machine management.....	91
Availability set.....	92
Service level agreement.....	93
Scalability.....	94
Image capture.....	97

Chapter 4 Azure Storage	101
Blob storage.....	101
Azure Files (preview)	102
Table storage.....	103
Queues.....	104
Redundancy.....	106
Creating and managing storage	107
Create a storage account	107
Create a container and add blobs.....	112
Create a table and add records.....	116
Chapter 5 Azure Virtual Networks.....	119
What is a virtual network	119
Overview.....	119
Definitions	120
Virtual network address spaces.....	120
Subnets.....	120
DNS servers.....	121
Creating a virtual network	121
Quick Create.....	122
Custom Create.....	122
Using a network configuration file.....	125
Export the network configuration file.....	125
Import the network configuration file.....	126
What ifs	127
What if you add a network?	127
What if you remove a network?	127
What if you change a network?	128

Cross-premises connection options.....	129
Site-to-site connectivity.....	129
Point-to-site connectivity	129
Comparing site-to-site and point-to-site connectivity.....	130
Private site-to-site connectivity (ExpressRoute).....	130
Point-to-site network.....	131
Overview of setup process.....	131
Configuring point-to-site VPN	132
Create a virtual network.....	132
Deploy a virtual machine into the virtual network.....	133
Create the VPN gateway.....	136
Create an authentication certificate for your virtual network	137
Upload the self-signed root certificate	140
Install the client certificate (.pfx) to authenticate the virtual network	140
Install the client VPN package	141
Connect to the virtual network through the VPN client.....	141
Connect to the VM using the internal IP address.....	143
Test the VPN connection	145
Chapter 6 Databases	146
Azure SQL Database.....	146
Administration.....	149
Firewall settings	149
Connect using SQL Server Management Studio	153
Billing.....	154
Business continuity	154
Database Copy	158
Import and Export	158

Standard Geo-Replication	162
Active Geo-Replication	162
Geo-Restore	165
Service Level Agreement.....	167
Applications connecting to Azure SQL Database.....	168
SQL Server in Azure Virtual Machines.....	169
Billing.....	169
Virtual machine configuration	170
Business continuity	170
Comparing Azure SQL Database with SQL Server in Azure Virtual Machines.....	171
Database alternatives.....	173
MySQL.....	173
NoSQL options	175
DocumentDB	175
Table storage	175
Chapter 7 Azure Active Directory.....	176
Overview of Azure Active Directory.....	176
What is Azure Active Directory?	176
Active Directory editions.....	177
Creating a directory.....	178
Custom domains.....	180
Delete a directory.....	186
Users and groups.....	186
Add users.....	187
Add groups.....	194
Multi-Factor Authentication	196
Azure Multi-Factor Authentication.....	197

Multi-Factor Authentication for Azure administrators	199
Application gallery	201
Adding gallery applications	201
Assigning users to applications	204
MyApps	206
Chapter 8 Management Tools.....	208
Management tool overview	208
Visual Studio 2013 and the Azure SDK.....	209
Install the Azure SDK.....	210
Manage resources with Server Explorer.....	212
Create an Azure resource.....	218
Windows PowerShell.....	219
Azure PowerShell cmdlet installation.....	220
Connecting to Azure.....	223
Connecting using a Microsoft account	223
Connect using a management certificate	224
Azure PowerShell modes.....	226
Usage	227
Cross-Platform Command-Line Interface.....	228
Installation.....	228
Installing on Windows.....	228
Installing on Linux.....	231
Connecting to Azure.....	232
Connect using a management certificate	232
Connect using a work or school account.....	233
Usage	233
Chapter 9 Business Cases.....	236

Development and test scenarios.....	236
Hybrid scenarios.....	238
Network connectivity.....	238
Internet connectivity.....	239
Application and infrastructure modernization and migration.....	240
Azure Mobile Services.....	241
About the Authors.....	243

Foreword

I'm thrilled to be able to share these Microsoft Azure Essentials ebooks with you. The power that Microsoft Azure gives you is thrilling but not unheard of from Microsoft. Many don't realize that Microsoft has been building and managing datacenters for over 25 years. Today, the company's cloud datacenters provide the core infrastructure and foundational technologies for its 200-plus online services, including Bing, MSN, Office 365, Xbox Live, Skype, OneDrive, and, of course, Microsoft Azure. The infrastructure is comprised of many hundreds of thousands of servers, content distribution networks, edge computing nodes, and fiber optic networks. Azure is built and managed by a team of experts working 24x7x365 to support services for millions of customers' businesses and living and working all over the globe.

Today, Azure is available in 141 countries, including China, and supports 10 languages and 19 currencies, all backed by Microsoft's \$15 billion investment in global datacenter infrastructure. Azure is continuously investing in the latest infrastructure technologies, with a focus on high reliability, operational excellence, cost-effectiveness, environmental sustainability, and a trustworthy online experience for customers and partners worldwide.

Microsoft Azure brings so many services to your fingertips in a reliable, secure, and environmentally sustainable way. You can do immense things with Azure, such as create a single VM with 32TB of storage driving more than 50,000 IOPS or utilize hundreds of thousands of CPU cores to solve your most difficult computational problems.

Perhaps you need to turn workloads on and off, or perhaps your company is growing fast! Some companies have workloads with unpredictable bursting, while others know when they are about to receive an influx of traffic. You pay only for what you use, and Azure is designed to work with common cloud computing patterns.

From Windows to Linux, SQL to NoSQL, Traffic Management to Virtual Networks, Cloud Services to Web Sites and beyond, we have so much to share with you in the coming months and years.

I hope you enjoy this Microsoft Azure Essentials series from Microsoft Press. The first three ebooks cover fundamentals of Azure, Azure Automation, and Azure Machine Learning. And I hope you enjoy living and working with Microsoft Azure as much as we do.

*Scott Guthrie
Executive Vice President
Cloud and Enterprise group, Microsoft Corporation*

Introduction

Microsoft Azure is Microsoft's cloud computing platform, providing a wide variety of services you can use without purchasing and provisioning your own hardware. Azure enables the rapid development of solutions and provides the resources to accomplish tasks that may not be feasible in an on-premises environment. Azure's compute, storage, network, and application services allow you to focus on building great solutions without the need to worry about how the physical infrastructure is assembled.

This ebook covers the fundamentals of Azure you need to start developing solutions right away. It concentrates on the features of the Azure platform that you are most likely to need to know rather than on every feature and service available on the platform. This ebook also provides several walkthroughs you can follow to learn how to create VMs and virtual networks, websites and storage accounts, and so on. In many cases, real-world tips are included to help you get the most out of your Azure experience.

In addition to its coverage of core Azure services, the ebook discusses common tools useful in creating and managing Azure-based solutions. The ebook wraps up by providing details on a few common business scenarios where Azure can provide compelling and valuable solutions.

Who should read this ebook

This ebook focuses on providing essential information about the key services of Azure for developers and IT professionals who are new to cloud computing. Detailed, step-by-step demonstrations are included to help the reader understand how to get started with each of the key services. This material is useful not only for those who have no prior experience with Azure, but also for those who need a refresher and those who may be familiar with one area but not others. Each chapter is standalone; there is no requirement that you perform the hands-on demonstrations from previous chapters to understand any particular chapter.

Assumptions

We expect that you have at least a minimal understanding of virtualized environments and virtual machines. There are no specific skills required overall for this ebook, but having some knowledge of the topic of each chapter will help you gain a deeper understanding. For example, the chapter on virtual networks will make more sense if you have some understanding of networking, and the chapter on databases will be more useful if you understand what a database is and for what you might use one. Web development skills will provide a good background for understanding websites, and some understanding of identity will be helpful when studying the chapter on Active Directory.

This ebook might not be for you if...

This ebook might not be for you if you are looking for an in-depth developer or architecture-focused discussion on a wide range of Azure features, or if you are looking for details on other public or private cloud platforms.

Organization of this ebook

This ebook explores six foundational features of the Microsoft Azure platform, along with insights on getting started with Azure, management tools, and common business scenarios. There are many services in the Azure platform that are not in the scope of this ebook, such as HDInsight (Azure's Hadoop service), Service Bus, and Azure Automation, to mention just a few. To learn about all of the services available in the Azure platform, start your journey at <http://azure.microsoft.com>.

The topics explored in this book include:

- **Getting started with Azure:** Understand what cloud computing is, visit the management portals, and learn about billing.
- **Websites and Cloud Services:** Learn about Azure Websites, from deployment to monitoring, and gain an understanding of the web and worker roles used in Azure Cloud Services.
- **Virtual Machines:** Explore the basic features of Azure Virtual Machines, including how to create, configure, and manage them.
- **Storage:** Read about the basics of Azure Storage, including blobs, tables, queues, and file shares.
- **Virtual Networks:** Learn the basics of virtual networks, including how to create one, and why a virtual network might be necessary. This also covers site-to-site and point-to-site networking, as well as ExpressRoute.
- **Databases:** Explore two relational database options available in Azure: Azure SQL Database and SQL Server in Azure Virtual Machines.
- **Azure Active Directory:** Explore basic features of Azure AD, including creating a directory, users and groups, and using the application gallery.
- **Management Tools:** Explore three common tools for working with Azure: Visual Studio 2013 and the Azure SDK, Azure PowerShell cmdlets, and the Cross-Platform Command-Line Interface
- **Business Scenarios:** Explore four common scenarios for utilizing Azure features: development and test, hybrid, application and infrastructure modernization, and Azure Mobile Services.

Conventions and features in this ebook

This ebook presents information using conventions designed to make the information readable and easy to follow:

- To create specific Azure resources, follow the numbered steps listing each action you must take to complete the exercise.
- There are currently two management portals for Azure: the Azure Management Portal at <http://manage.windowsazure.com> and the new Azure Preview Portal at <http://portal.azure.com>. It is necessary to move between both portals to explore all Azure features; unless otherwise specified, assume the Azure Preview Portal is used.
- Boxed elements with labels such as "Note" or "See Also" provide additional information.
- A plus sign (+) between two key names means that you must press those keys at the same time. For example, "Press Alt+Tab" means that you hold down the Alt key while you press Tab.
- A right angle bracket between two or more menu items (e.g., File Browse > Virtual Machines) means that you should select the first menu or menu item, then the next, and so on.

System requirements

For many of the examples in this ebook, you need only internet access and a browser (Internet Explorer 10 or higher) to access the Azure portals.

Chapter 2, "Azure Websites and Azure Cloud Services," and Chapter 4, "Azure Storage," use Visual Studio to show some concepts used in developing applications for Azure. For these examples, you will need Visual Studio. The system requirements are:

- Windows 7 Service Pack 1, Windows 8, Windows 8.1, Windows Server 2008 R2 SP1, Windows Server 2012, or Windows Server 2012 R2
- Computer that has a 1.6GHz or faster processor (2GHz recommended)
- 1 GB (32 Bit) or 2 GB (64 Bit) RAM (Add 512 MB if running in a virtual machine)
- 20 GB of available hard disk space
- 5400 RPM hard disk drive
- DirectX 9 capable video card running at 1024 x 768 or higher-resolution display

- DVD-ROM drive (if installing Visual Studio from DVD)
- Internet connection

After installing Visual Studio, you must also install the Azure Tools and SDK version 2.4 or higher. You can do this using the Web Platform Installer at <http://azure.microsoft.com/en-us/downloads/>, or, if preferred, you can manually install and configure the required components by following the instructions here: <http://www.microsoft.com/en-us/download/details.aspx?id=43709> (SDK 2.4) or here: <http://www.microsoft.com/en-us/download/details.aspx?id=44938> (SDK 2.5).

Links to both version 2.4 and version 2.5 are provided here because version 2.5 has some breaking changes. If you have any other Azure solutions that use SDK 2.4, you should read the release notes for version 2.5 before upgrading: <http://msdn.microsoft.com/en-us/library/azure/dn873976.aspx>.

The system requirements for the Azure SDK that are not included in the Visual Studio system requirements are as follows:

- IIS7 with ASP.NET and WCF HTTP Activation, Static Content, IIS Management Console, and HTTP Redirection
- Web Deployment Tools 2.1 or up
- Internet Explorer 10 or higher

Depending on your Windows configuration, you might require Local Administrator rights to install or configure Visual Studio 2013.

Acknowledgments

The Azure community is made up of many people bound together by this one technology. We are honored to be members of this community, and we thank you for your help and support. We would like to especially thank Neil Mackenzie, Mike Martin, Gaurav Mantri, and Fabien Lavocat for their detailed technical reviews and feedback. All of them provided additional insights that greatly enhanced the overall quality and value of this ebook.

Special thanks to the team at Microsoft Press for their unwavering support and guidance on this journey. It was a pleasure to work with our editor, Devon Musgrave. Devon provided immensely helpful advice from the days when this ebook was just an idea, all the way through to final copy.

Most importantly, we are profoundly grateful to our families and friends for their love, encouragement, and patience. Many nights and weekends were sacrificed in the writing of this ebook.

Errata, updates, & support

We've made every effort to ensure the accuracy of this ebook. You can access updates to this ebook—in the form of a list of submitted errata and their related corrections—at:

<http://aka.ms/FundAzure>

If you discover an error that is not already listed, please submit it to us at 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 and hardware is not offered through the previous addresses. For help with Microsoft software or hardware, go to <http://support.microsoft.com>.

Free ebooks from Microsoft Press

From technical overviews to in-depth information on special topics, the free ebooks from Microsoft Press cover a wide range of topics. These ebooks are available in PDF, EPUB, and Mobi for Kindle formats, ready for you to download at:

<http://aka.ms/mspressfree>

Check back often to see what is new!

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://aka.ms/tellpress>

We know you're busy, so we've kept it short with just a few questions. Your answers go directly to the editors at Microsoft Press. (No personal information will be requested.) Thanks in advance for your input!

Stay in touch

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

Chapter 2

Azure Websites and Azure Cloud Services

In this chapter, we take a look at two of the Platform as a Service (PaaS) offerings in Microsoft Azure: Azure Websites and Azure Cloud Services. We talk about what Azure Websites is, how to use the service to create websites, and how to keep them updated. We also look at the options for prebuilt websites offered by Azure.

Cloud Services is a PaaS compute feature in which applications are deployed into instances (virtual machines [VMs]) of server types referred to as web roles and worker roles. The deployment of the instances is fully managed by Microsoft, making it easy to scale applications in and out. This feature is not to be confused with the Cloud Services that are a container for self-deployed VMs, which are covered in Chapter 3, “Azure Virtual Machines.”

Creating and configuring websites

In this section, we take a look at what Azure Websites is, discuss some of its features, and show you how to create, configure, and scale websites.

What is Azure Websites?

Azure Websites is a managed cloud service that allows you to deploy a web application and make it available to your customers on the Internet in a very short amount of time. You don't directly support the VMs on which your website runs; they are managed for you.

Supported languages include .NET, Java, PHP, Node.js, and Python. In addition to creating your own website, there are several web applications available to use as a starting point, such as WordPress, Umbraco, Joomla!, and Drupal.

You can use continuous deployment with Team Foundation Server (TFS), Git, or GitHub so that every time you commit a change, a new version of the website is deployed.

You have the ability to scale the number of instances in and out on demand; you also can configure autoscaling so Azure will scale it in or out for you depending on specific performance measures such as CPU Percentage. If your website has multiple instances, you can configure load balancing to make the most of your resources.

For diagnostics, you can gather performance statistics, application logging, web server logging, IIS

logs, and IIS Failed Request logs. If you're using Microsoft Visual Studio, you can even remotely debug your application while it is running in the cloud.

In short, there are many features available in Azure Websites to make it easy for you to deploy, manage, and troubleshoot a web application.

Creating a new website

Let's create a new website. Later in this section, we will publish content to the website.

Start by logging into the Microsoft Azure Preview Portal (portal.azure.com). At this point, you need an Azure account. If you don't have one, you can sign up for a free trial at azure.microsoft.com.

Using the portal

After logging into the portal, click the big +NEW icon in the lower-left corner of the screen and select Website, as displayed in Figure 2-1.

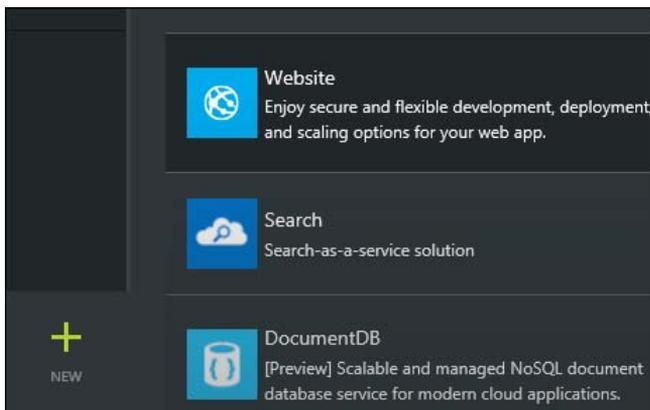
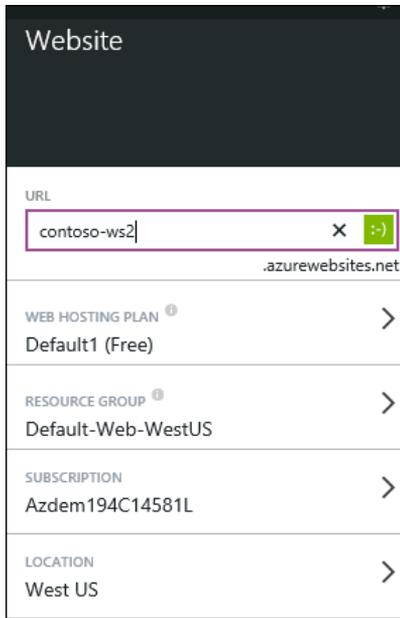


FIGURE 2-1 Add a new website in the Azure Preview Portal.

You should now see something similar to Figure 2-2, with the fields ready to be filled in.



The screenshot shows a 'Website' creation form with the following fields:

- URL:** A text input field containing 'contoso-ws2'. To the right of the input is a green square with a smiley face ':-)'. Below the input field, the text '.azurewebsites.net' is displayed.
- WEB HOSTING PLAN:** A dropdown menu with 'Default1 (Free)' selected and a right-pointing chevron.
- RESOURCE GROUP:** A dropdown menu with 'Default-Web-WestUS' selected and a right-pointing chevron.
- SUBSCRIPTION:** A dropdown menu with 'Azdem194C14581L' selected and a right-pointing chevron.
- LOCATION:** A dropdown menu with 'West US' selected and a right-pointing chevron.

FIGURE 2-2 Create a new website.

The URL must be unique among all of the entries used in Azure Websites. If accepted, there will be a green square with a smiley face in it. Note that whatever prefix is provided here will be appended with *.azurewebsites.net* to create the URL for the website.

SUBSCRIPTION shows the name of the subscription assigned to the Microsoft account with which you logged in. If you administer multiple accounts with the same Microsoft account, you can click SUBSCRIPTION and select the subscription you want to use. LOCATION is the region of the datacenter where the website will be hosted. Select the LOCATION closest to you. Accept the default for RESOURCE GROUP.

WEB HOSTING PLAN defines the allocation of resources for the website, such as number of cores and memory, amount of local storage, and the features available, such as autoscaling and backups.

If you click the selection for WEB HOSTING PLAN, the window in Figure 2-3 will be displayed. You can specify a name for a new web hosting plan and then pick the plan you want. Not all of the plans are displayed on that screen. If you scroll down below the OK button (past what is displayed here), you will see a BROWSE ALL PRICING TIERS option, and you can click that to see all of them. Under that selection is Or Use Existing, which basically says not to create a new web hosting plan. Select the free tier or, if your default is the free tier, select Use Existing.

Web hosting plan

Choose a new or existing web hosting plan. [Learn more](#)

 Create new

NAME

S1 STANDARD 74.40	B1 BASIC 55.80	S3 STANDARD 297.60
--------------------------------	-----------------------------	---------------------------------

1 Core	
1.75 GB RAM	
 Storage	50 GB
 Custom domains / SSL	5 SNI, 1 IP
 Auto scale	Up to 10 instances
 Backup	Daily
 Website staging	5 slots
 Geo availability	Traffic Manager

FIGURE 2-3 Web Hosting Plan selection.

Use the defaults on the rest of the fields, make sure the Add To Startboard check box is selected, and click Create on the bottom of the new website screen (Figure 2-4).

Add to Startboard

FIGURE 2-4 Create a website and add it to the Startboard.

Azure will create your new website, pin it to the Startboard of your portal so you can easily find it, and show the website and its properties, as displayed in Figure 2-5.

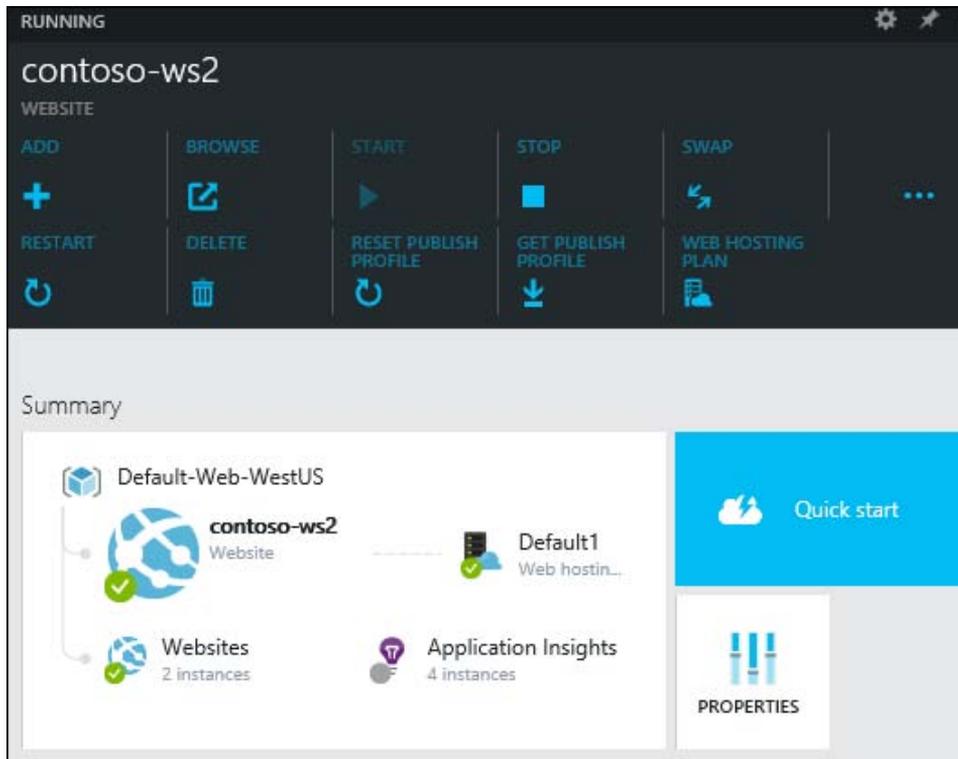


FIGURE 2-5 Website options.

If you click the three dots to the right of SWAP, you can see all of the options:

- ADD adds a new website.
- BROWSE opens your website in the browser. If you haven't published anything yet, it shows a default page directing you to various deployment tools.
- START/STOP starts and stops the website.
- SWAP swaps deployment environments. For example, if you have a production environment and a staging environment, you can publish your website to staging and test it. When you're satisfied with it, you can promote it to production by using the SWAP option and then remove the staging environment that is now the old production version.
- RESTART restarts your website.
- DELETE removes the website from your account.
- RESET PUBLISH PROFILE resets the publishing credentials and invalidates the old credentials;

these are the credentials used for FTP and Git access.

- GET PUBLISH PROFILE retrieves the information needed to publish a website from Visual Studio.
- WEB HOSTING PLAN allows you to change the size, instance count, etc. for the host on which the website is running.

At this point, you've created a new Azure website but you haven't published any content to it; we'll do that in the section "Publishing a website from Visual Studio" later in this chapter.

Websites gallery

While we're looking at the Azure Preview Portal, let's look at some of the website options available from the Azure Marketplace. Click BROWSE on the left side of the page and then select Marketplace. In the Marketplace blade, select Web to display the options for websites.

Here, there are several precreated websites and templates that you can take advantage of. If you scroll down, you can see the categories. At the end of any row, clicking More will show additional options in that category. Here are some of the choices displayed:

- Blogs + CMSs: WordPress, DNN, Joomla!, Umbraco CMS, MonoX, and Drupal
- Starter Sites: ASP.NET, HTML5, Node.js, PHP, some examples like a Bakery website

Select Scalable WordPress—it shows you details on the right side. Click Create at the bottom of that window. This opens a window where you can configure your WordPress site; see Figure 2-6.

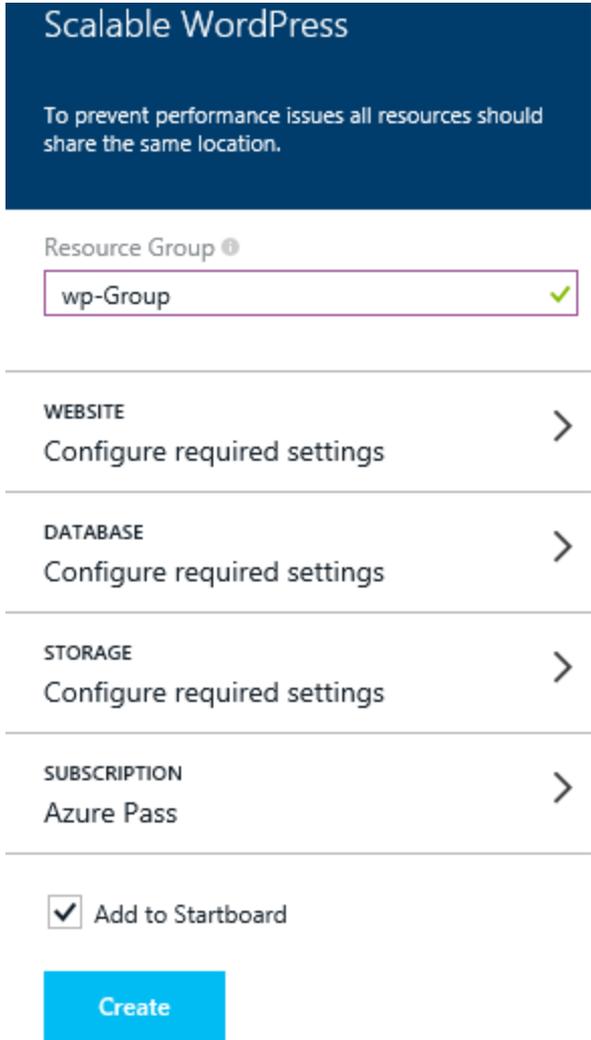


FIGURE 2-6 Set up and configure your WordPress website.

RESOURCE GROUP is a way of grouping multiple resources to be used to see and manage Azure resources that are related to one another, such as a website and a database. Fill that in and under WEBSITE, click Configure Required Settings.

WEBSITE settings include the WEB HOSTING PLAN (the same as discussed previously), the LOCATION (the region in which the datacenter resides), and the WEB APP SETTINGS (including various keys and SALT values).

DATABASE settings include the DATABASE NAME, the pricing tier, LOCATION of the datacenter, and LEGAL TERMS, which basically give Microsoft permission to bill you for the MySQL database service.

STORAGE settings allows you to either create an additional storage account for the website to use or select an existing storage account.

After entering the configuration information, click CREATE. Azure will create the WordPress site for you. You can access it as an administrator and make any additional changes needed.

Configure and scale a website

Let's look at the configuration and scaling options for a website in the Azure Management Portal (*manage.windowsazure.com*). (Not all of the features are available in the Azure Preview Portal yet.) Log into the portal, select WEB SITES in the left column, and then click one of your websites.

Configuration

To find the configuration settings for the website, click the CONFIGURE option on the top of the page; see Figure 2-7.

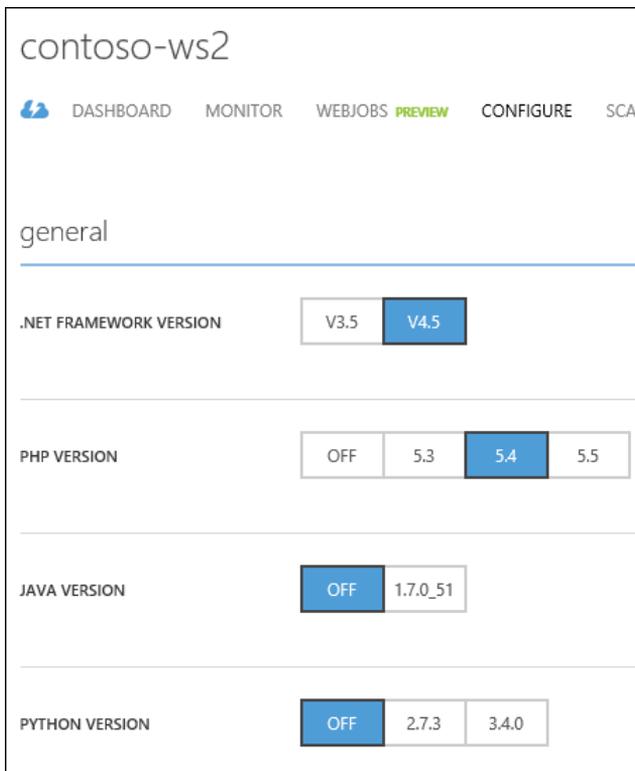


FIGURE 2-7 Configuration settings for the website.

This is the top of the CONFIGURE page. As you can see, you can set the versions for .NET, PHP, Java, and Python here. Scrolling farther down the page, we see additional general options for the website, as shown in Figure 2-8.

- **PLATFORM (32-BIT/64-BIT)** When you set up a Free website, this is set to 32-bit. Once you change your website to Standard, you might want to change this to 64-bit.
- **WEB SOCKETS (ON/OFF)** When this is enabled, you can use real-time request pattern applications that communicate using web sockets, such as chat.
- **ALWAYS ON (ON/OFF)** When this is enabled on a site, Azure will automatically ping your website regularly to ensure that the website is always active and in a warm/running state. Doing this will make sure that a site is always responsive and that the process or app domain has not paged out due to lack of external HTTP requests.
- **EDIT IN VISUAL STUDIO ONLINE (PREVIEW)** If you turn this on, a link to the editor will appear in the quick glance section of the DASHBOARD tab. This enables you to use Visual Studio Online to edit your website while it's live. If you do live editing and you have Deployment From Source Control enabled, if someone checks in a change it will overwrite the live changes you made.

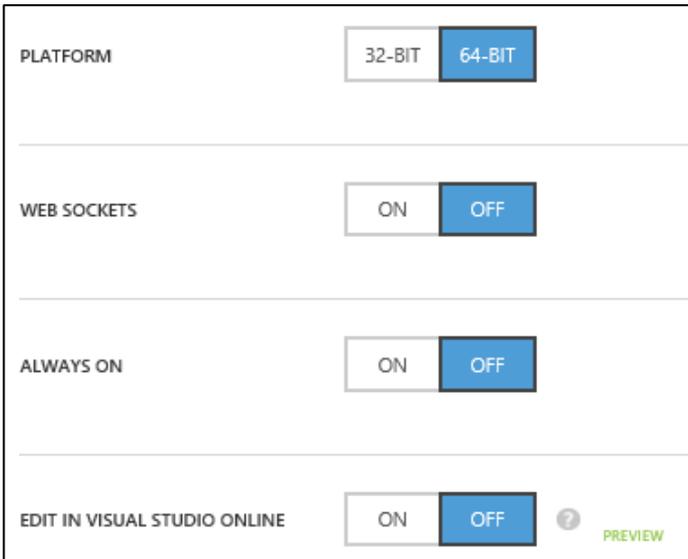


FIGURE 2-8 More general website options.

There are options for uploading certificates, managing domains, and managing your Secure Sockets Layer (SSL) bindings, as shown in Figure 2-9.

- **Certificates** You can upload an SSL certificate here. If you bind your SSL certificate to your custom domain name, end users can access your site using HTTPS.
- **Domain Names** This allows you to use a custom domain like mywebsite.contoso.com instead of mywebsiteatcontoso.azurewebsites.net.

- **SSL Bindings** This is where you bind the SSL certificate to the custom domain name.

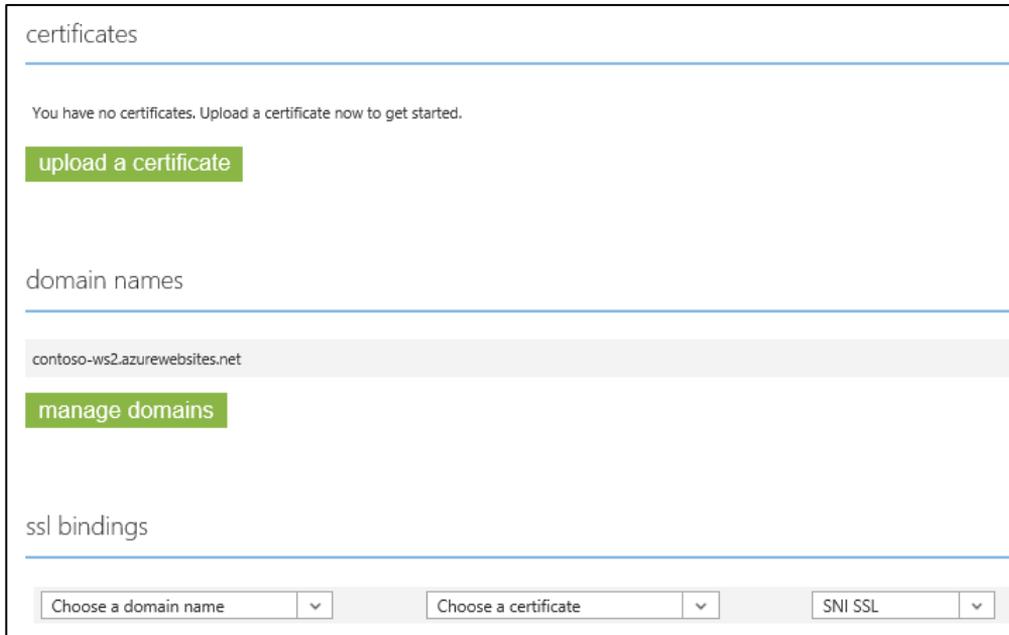


FIGURE 2-9 Manage certificates, domain names, and SSL bindings for the website.

The next section is used to configure application diagnostics; see Figure 2-10. To show as much as possible in the image, all of the settings are enabled.

- **APPLICATION LOGGING (FILE SYSTEM) (ON/OFF)** If this is turned on, then any logging performed by the web application will be written to the file system. You can access the logs by FTPing into the website. Because of the limited amount of disk space available, this will be enabled for 12 hours and then disable itself. The logging levels include Error, Warning, Information, and Verbose.
- **APPLICATION LOGGING (TABLE STORAGE) (ON/OFF)** If this is turned on, any logging performed by the web application will be written to Azure Tables. The logging levels include Error, Warning, Information, and Verbose. If you select this option, you will be prompted for the storage account and table to be used (see Figure 2-11). These logs are never deleted automatically.
- **APPLICATION LOGGING (BLOB STORAGE) (ON/OFF)** If this is turned on, it writes the logs to Azure Blob storage, storing logs for each hour in a separate blob. For these logs, you can specify a retention time in days; if you leave it blank, these logs will never be deleted automatically. If you select this option, you will be prompted for the storage account and container (Figure 2-12).

application diagnostics

Application tracing to the file system will be enabled for 12 hours.

APPLICATION LOGGING (FILE SYSTEM) ON OFF ?

LOGGING LEVEL

APPLICATION LOGGING (TABLE STORAGE) ON OFF ?

LOGGING LEVEL

[manage table storage](#)

APPLICATION LOGGING (BLOB STORAGE) ON OFF ?

LOGGING LEVEL

[manage blob storage](#)

SET RETENTION ?

RETENTION PERIOD days

FIGURE 2-10 Configuring application diagnostics for a website.

Manage Table Storage for Application Diagnostics

STORAGE ACCOUNT 

nightbirdstorage 

WINDOWS AZURE TABLE

Create a new table 

TABLE NAME

wawsapplogtablecontoso-ws2



FIGURE 2-11 Configuring Table storage for application diagnostics.

Manage Blob Storage for Application Diagnostics

STORAGE ACCOUNT 

nightbirdstorage 

WINDOWS AZURE BLOB CONTAINER

Create a new blob container 

BLOB CONTAINER NAME

wawsapplogblobcontoso-ws2 



FIGURE 2-12 Configuring Blob storage for application diagnostics.

The next section is used to configure site diagnostics (see Figure 2-13).

- **WEB SERVER LOGGING (OFF/STORAGE/FILE SYSTEM)** This indicates whether to write the web server (IIS) logs to Azure Tables or to the local file system. You can set the retention time if you choose STORAGE or FILE SYSTEM. For FILE SYSTEM, you also can set the QUOTA, or maximum amount of disk space the logs can take up, which must be between 25 MB and 100 MB.
- **DETAILED ERROR MESSAGES (ON/OFF)** This indicates whether to write summary error messages or detailed error messages.
- **FAILED REQUEST TRACING (ON/OFF)** This indicates whether to write the IIS Failure Logs.

site diagnostics

WEB SERVER LOGGING

QUOTA MB

SET RETENTION

RETENTION PERIOD days

DETAILED ERROR MESSAGES

FAILED REQUEST TRACING

FIGURE 2-13 Configuring site diagnostics.

In the next section, you can configure remote debugging (see Figure 2-14). If you turn this on and publish a debug version of your website, you can use Visual Studio to attach a debugger and debug your website while it's running in Azure.

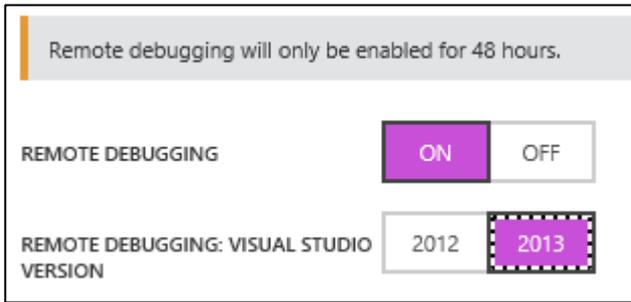


FIGURE 2-14 Configuring remote debugging.

In the next section, you can specify up to two endpoints to be monitored, as shown in Figure 2-15. Configuring this will allow you to monitor the availability of HTTP or HTTPS endpoints from up to three locations, including Chicago (IL), Amsterdam, Singapore, San Jose (CA), San Antonio (TX), Ashburn (VA), Hong Kong, and Dublin. If you have an internationally used application, this can help you pinpoint latency around the world.

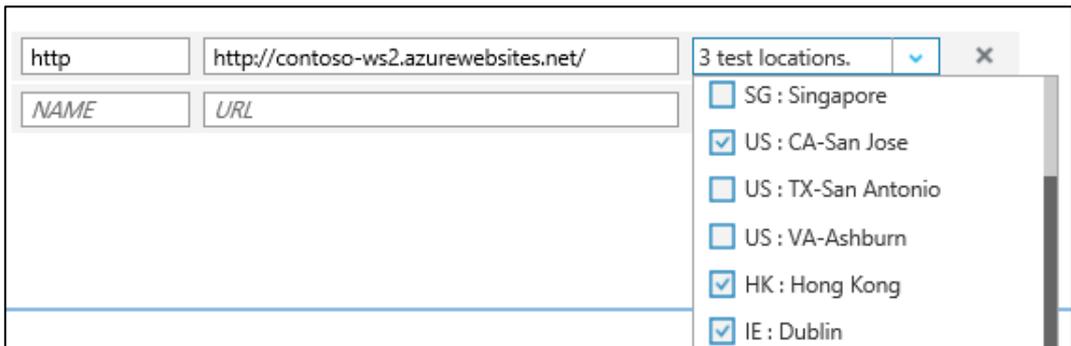


FIGURE 2-15 Configuring endpoint monitoring.

The only thing in the Azure Preview Portal that is not in the Azure Management Portal is the ability to set up and manage multiple deployment slots for your website, such as staging and production. This is under the Configuration section of the website blade in the Azure Preview Portal (portal.azure.com).

Scaling

You cannot scale a Free website; it is restricted to one instance. You can scale a Basic website manually up to three instances. For autoscaling, you must use a Standard website, which allows up to 10 instances. Let's look at the options using the Azure Management Portal (manage.windowsazure.com) because not all of the features have been migrated to Azure Preview Portal yet. First, we need to make sure the web hosting plan is STANDARD.

Log into the Azure Management Portal (manage.windowsazure.com), click WEBSITES in the left column, and then select the website you want to configure or autoscale. Click SCALE at the top of the

screen. You should see something similar to Figure 2-16.

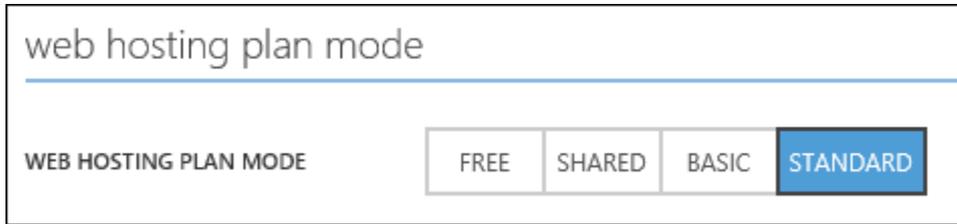


FIGURE 2-16 Web hosting plan.

Select STANDARD to change your plan and then click SAVE at the bottom of the screen. Note that you also can change the Instance Size (number of instances) on this screen.

The Azure Preview Portal has more details about the web hosting plan than the Azure Management Portal does. To check out those options, log into *portal.azure.com* and select your website, and then click WEB HOSTING PLAN in the actions on the top. If you do that now, be sure to come back to the Azure Management Portal to continue.

Now that we have a standard website, the scaling options are visible. First, you can scale according to a schedule. Clicking Set Up Schedule Times will display the entry screen shown in Figure 2-17.

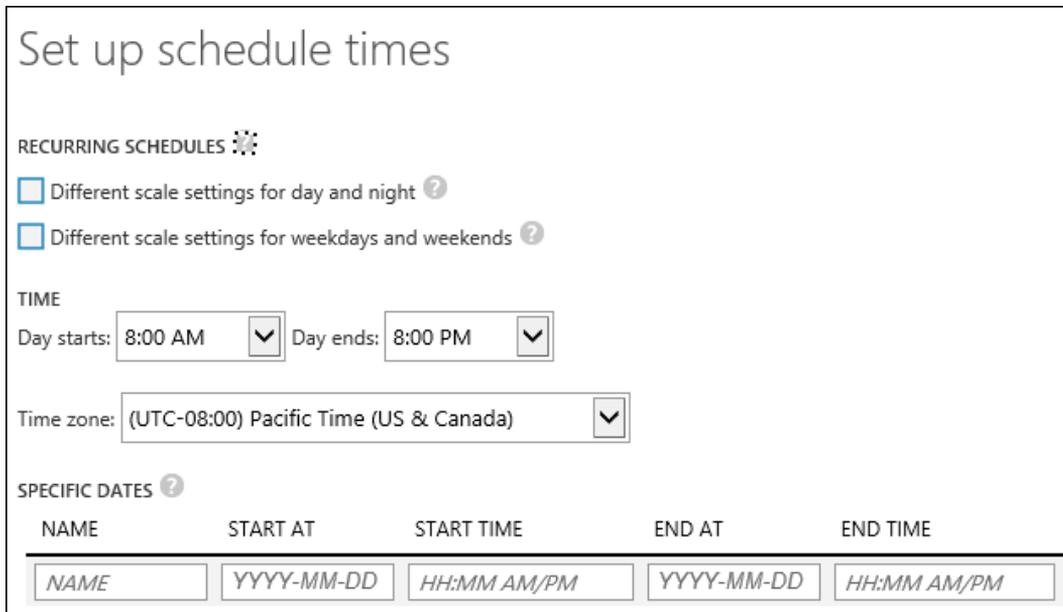


FIGURE 2-17 Scaling by schedule.

As you can see, there are several options. Additionally, you can set a default schedule and then override it for specific dates. For example, if you're scaling up from 8 A.M. to 5 P.M., you might want to

override that on a holiday.

You also can scale by CPU Percentage as shown in Figure 2-18.



FIGURE 2-18 Scale by metric.

This shows a chart of the number of instances for the past week. You can set the minimum and maximum number of instances (INSTANCE COUNT) and then set the TARGET CPU for the scaling. In this example, when the CPU hits 60 percent, it will increase the number of instances to three, and when it goes below 60 percent again, it will decrease them back to one. With Azure Websites, autoscaling takes about five minutes.

Deploying and monitoring websites

In this section, we look at options for creating website content, show how to publish your website from Visual Studio 2013, and look at the monitoring options in the Azure Management Portal.

Options for creating websites

There are multiple options for creating a website and pushing the content up to Azure Websites.

Notepad or an HTML editor

This is a pretty limited way to create a website, but if you're beginning on your journey to learn web development and want to create a simple HTML page, you can do that with Notepad or your favorite

HTML editing tool. After you're done, you can FTP to the website and transfer the files.

To FTP your files up to your website, log into the Azure Management Portal, click Websites, and then select your website. If you have not set up your login credentials yet, click Reset Your Deployment Credentials in the "quick glance" column. When prompted, provide a username and password. This is for access via Git or FTP.

Also under the "quick glance" column, the FTP HOST NAME and the DEPLOYMENT / FTP USER are provided. When you FTP in to your website, you will use these two pieces of information in addition to the password to access your website and place your files.

WebMatrix

This is a free, lightweight, cloud-connected web development tool that will enable you to create, publish, and maintain your websites. You can download this from <http://www.microsoft.com/web/webmatrix/>. Here are some of the features of this application:

- Seamless integration with Azure Websites.
- Can be used with PHP, Node.js, ASP.NET, HTML5, CSS3, and jQuery.
- Enables you to support many of the websites in the Website Gallery in the Azure Management Portal or the Website Marketplace in the Azure Preview Portal. Some examples of the websites available are Umbraco, WordPress, Joomla!, and Drupal.
- Allows management for SQL Server, SQL CE, and MySQL databases.
- Works seamlessly with Git and TFS.
- Can be used to develop websites locally or remotely using FTP or WebDeploy.

You can log in using the Microsoft account you use for Azure, create a new web application using one of the templates available, and publish it. When you log into either of the Azure Management Portals, you will see your new web application. You can make changes, verify the results in the local browser, and then republish. Republishing only deploys the modified files.

Visual Studio

Visual Studio is a full development environment, giving you the ability to create many different kinds of applications including, but not limited to, ASP.NET MVC applications, .NET client applications, Windows Communication Foundation (WCF) services, Web API, and Cloud Services, using languages such as C#, C++, VB, F#, and XAML.

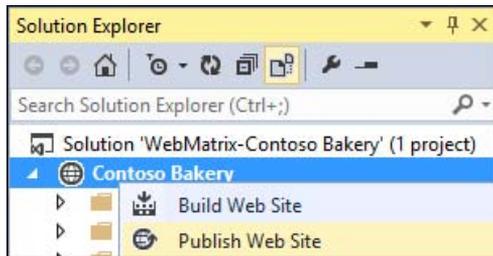
With Visual Studio, you can create a new web application and publish it directly to Azure Websites. We'll see how to do this in the next section.

Publishing a website from Visual Studio

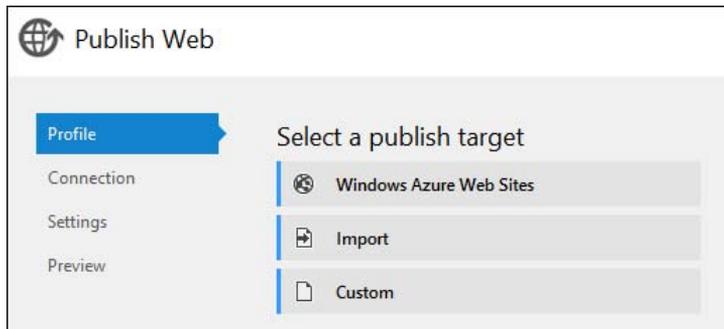
Open one of your web applications in Visual Studio. If you don't have a web application yet, create a new one with Visual Studio by selecting FILE > NEW PROJECT, selecting an ASP.NET Web Application, specifying the folder for the solution, and then selecting MVC Application. This gives you a basic MVC application that runs "as is." You can modify it later to make it your own.

Let's publish the web application to the Azure website we created earlier in this chapter.

1. Open your web application in Visual Studio. Right-click the website and select Publish Web Site.



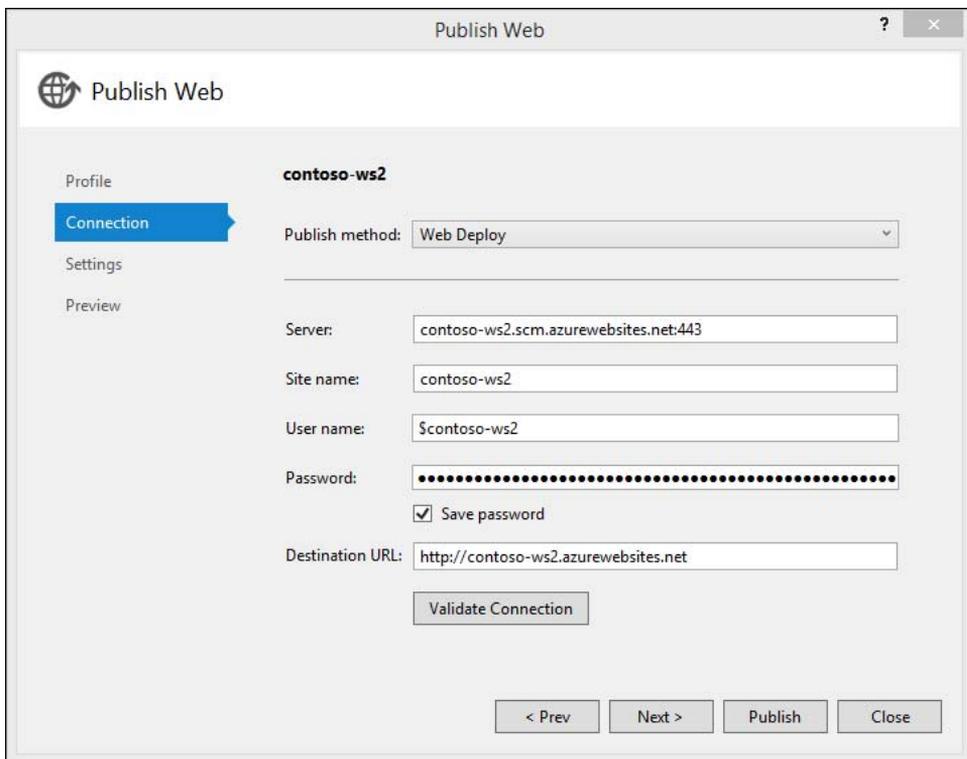
The Publish Web dialog will be displayed.



2. Select Windows Azure Web Sites. You will be asked to sign into your Azure account.
3. After signing in, you will be prompted to select the website to which to deploy. Select your website in the drop-down list and click OK. It retrieves the publishing settings from Azure and displays the connection information.



4. Click Validate Connection to make sure the connection works.



5. Click Next to go to the next screen, where you can select Debug or Release Configuration. Accept the defaults on that screen and click Next to go to the final screen.
6. On the final screen, you can preview the files that will be published. Click Publish to publish the website. All of the files will be deployed to the website, and then it will open the website.

When you make changes to your website and go through this same process to publish the website

again, it will only publish the files that have been added or modified.

Monitoring a website

You can set up many metrics to be monitored for a website. To do this, log into the Azure Management Portal (*manage.windowsazure.com*), select WEBSITES, and then select your website. When it brings up the Quick Start or the DASHBOARD, select MONITOR from the options at the top of the screen.

Initially, you will see only six metrics: CPU Time, Data In, Data Out, Http Server Errors, and Requests. If you set up Endpoint Monitoring on the CONFIGURE screen, you also will see the response times here (Figure 2-19).

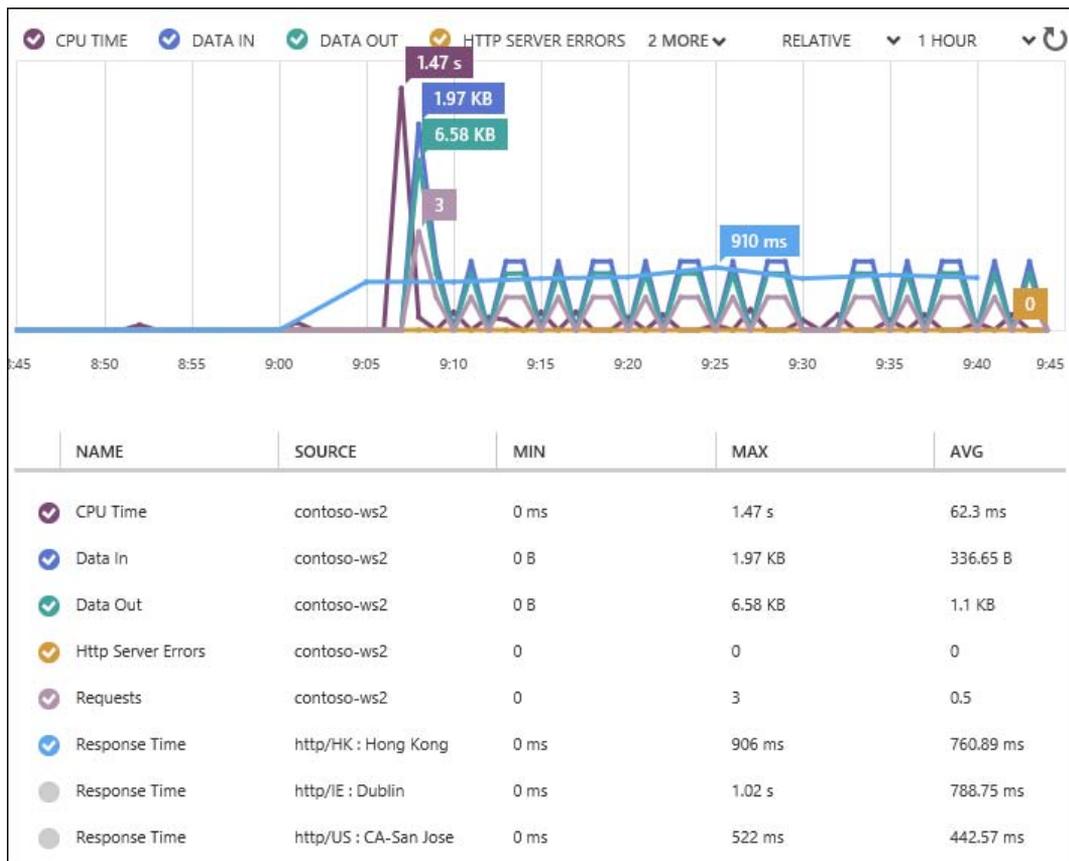


FIGURE 2-19 Monitoring a website.

You can see the endpoints defined earlier for Hong Kong, Dublin, and San Jose, CA. You can request as many metrics as you need to be displayed in the list, but you can select only six metrics to appear on the chart at any given time. You can add metrics by clicking +ADD METRICS at the bottom of the

screen. There are several more that can be selected.

The time frame displayed is 1 HOUR, as selected in the upper-right corner. Note that there is no y-axis. This is because each metric has its own y-axis value; it charts these to make the best of the space available.

There are other monitoring applications available through the Azure Store in the Azure Management Portal, including New Relic and App Dynamics. These can be selected and configured on the CONFIGURE screen in the Developer Analytics section.

PaaS Cloud Services

In Azure, there are two uses of the phrase “cloud service.” One is as a container for VMs that you create and maintain yourself. For example, you might create four VMs that are identical and put them in one cloud service. Then you would use the IP address of the cloud service as the entry point, and Azure would provide automatic load balancing of the four VMs. If you want to update the application running on those VMs, you must deploy it to each of the VMs. You can put them in an availability set to ensure that you always have a minimum number of VMs available. If you want to scale the application up or down, you must manually add or remove VMs to or from the cloud service or stop or start them. These VMs will be covered in Chapter 3.

The second kind of cloud service is one in which Azure maintains and manages your VMs for you. When you want to update all of the VMs, you just publish a new version of the application, and Azure updates each of the VMs, making sure to cycle through them so there’s no downtime. If you want to change the number of VMs, you just log into the Azure Management Portal and change the instance count, and Azure will add or remove the requested number of VMs for you.

This section is about the second kind of cloud service. In these cloud services, you have either web roles or worker roles. The only difference between the two is that web roles have IIS running in them by default. Web roles generally are used for web applications, WCF services, and anything else requiring IIS.

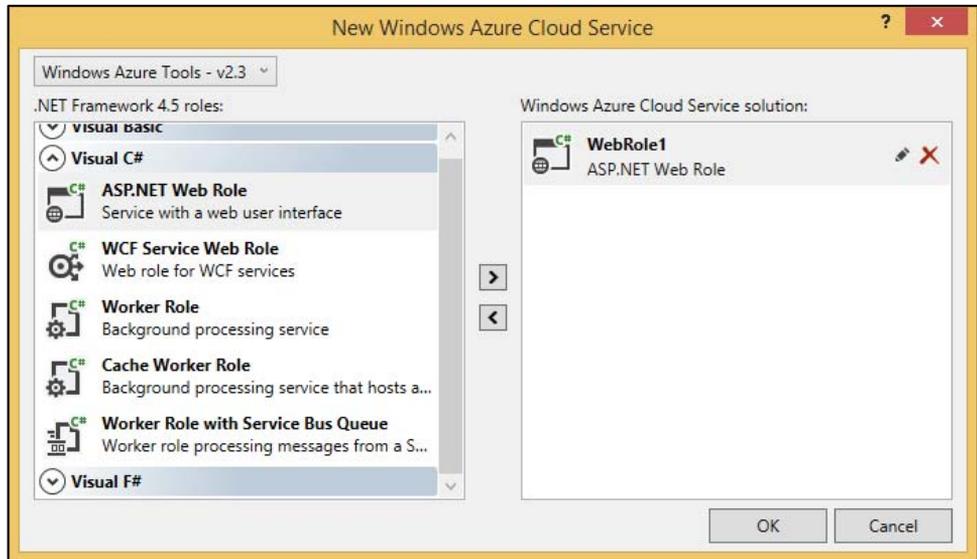
Worker roles generally are used for processing that needs to be continuous. For example, you might have a queue with messages on it that you want processed. The worker role runs an infinite loop that checks for messages on the queue and, if found, retrieves them and processes them. You can do substantial processing of images, video, files, etc. in a worker role.

Creating a cloud service

Let’s create a cloud service with a web role in it. Then we can look at the configurations and see how to publish it.

1. Using Visual Studio, create a new project (FILE > NEW PROJECT). For the type of project, select Windows Azure Cloud Service.

2. Fill in the name of the solution and click OK. (If you don't have Windows Azure Cloud Service in the list of projects, then you need to install the Azure SDK and Tools.) Next you will be prompted to select your role(s).
3. Select the ASP.NET Web Role and click the right-facing arrow to copy it to the right. Hover over it and click the pencil if you want to change the name of your web role before continuing and then click OK. Because you selected an ASP.NET Web project, you will be prompted to select which kind of ASP.NET project. Select MVC and click OK to continue.



Now when you look at your solution, you will see the project with the web role (MVC application) and the project with the cloud service; see Figure 2-20.

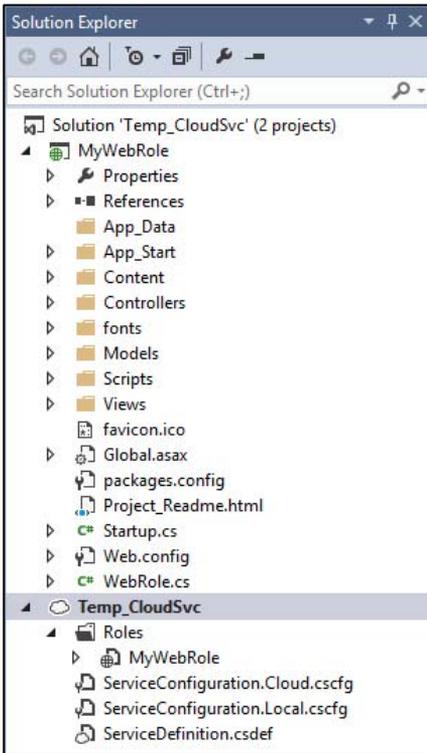


FIGURE 2-20 Web application and cloud service project.

In the web application project, notice there is a `WebRole.cs`. This is what starts up when the web role spins up in Azure. By default, this is blank, but you can put in event handlers for the role starting up and shutting down. For example, if you were going to use an Azure queue, you might want to add code to the role startup to make sure the queue exists and create it if it doesn't.

If you double-click the role, you can access its properties. Here are some of the most commonly used properties:

- **Configuration** VM Size, Instance count, diagnostics configuration, storage connection strings. To qualify for the service level agreement (SLA), a minimum instance count of two is required.
- **Settings** Configuration settings that you can retrieve in code. These can be modified in the Azure Management Portal while the site is running live. You can have settings for multiple configurations, including debug and release.
- **Endpoints** By default, a web role will have an HTTP endpoint open. You can modify that and open other endpoints such as HTTPS.
- **Local Storage** You can configure local storage for each instance. For example, if you wanted a 5 GB drive to use for temporary files, you would configure that here. Each instance would

have its own 5 GB local resource space.

- **Certificates** You can configure certificates to be used for SSL and for Remote Desktop (RDP) access here. You also must upload the certificate to the Azure Management Portal.

The settings are stored in XML format in the ServiceConfiguration.*.cscfg file(s). There are a couple of properties in the XML in that file that are not surfaced through the UI: osFamily and osVersion. osFamily selects the operating system that will be running in the VM. For example, osFamily 4 is Windows Server 2012 R2; osFamily 3 is Windows Server 2012; and osFamily 2 is Windows Server 2008 SP2.

To learn more about these values, check out the article "Azure Guest OS Releases and SDK Compatibility Matrix" at <http://msdn.microsoft.com/en-us/library/ee924680.aspx>. Unless your application requires a specific version, leave the values at the defaults, which are currently osFamily = "4" and osVersion = "*". The asterisk in osVersion means it will use the most recent version of the selected osFamily.

You can have multiple ServiceConfiguration.*.cscfg files. When you publish, it will ask which one you want to use. There is only one ServiceDefinition.csdef file. This has the master list of configuration setting variables and the endpoint definitions. It also has the instance size. Instance sizes range from Extra Small (shared, 1 CPU core, 768 MB memory) to A9 (16 CPU cores, 112 GB memory). Also available is the new D-series VM, with faster processors, a Solid State Drive (SSD) for the temporary disk, and a higher memory-to-core ratio.

For more details, check out "Virtual Machine and Cloud Service Sizes for Azure" at <http://msdn.microsoft.com/en-us/library/azure/dn197896.aspx>.

You can test a cloud service by using the storage and compute emulators. These are installed as part of the Azure SDK and Tools. If you make the cloud service your startup project, you can just hit F5 and the emulator(s) will start up and the application will run in your default browser.

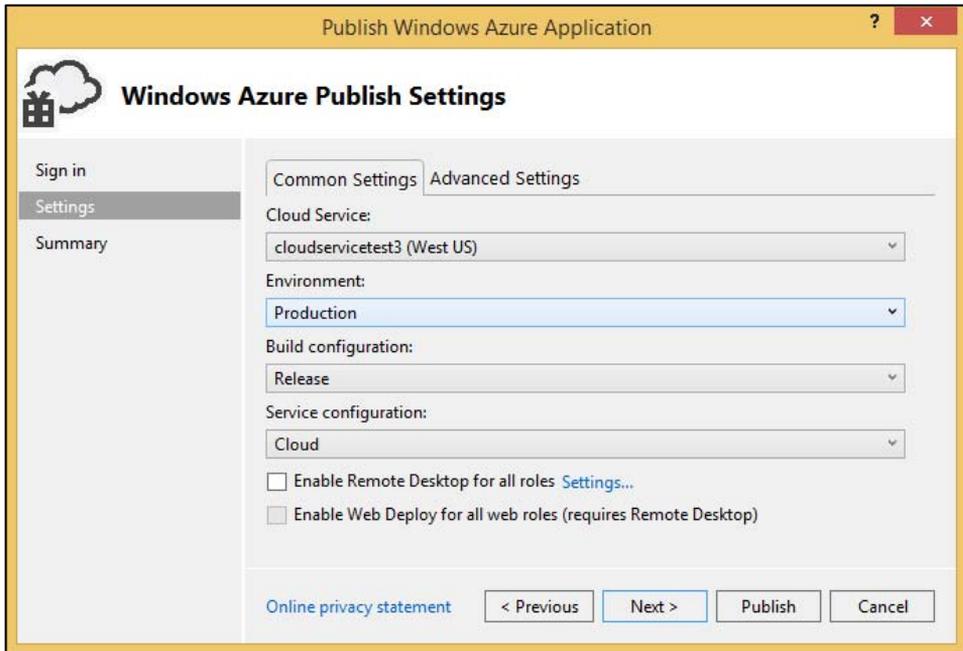
Publishing a cloud service

Let's publish our cloud service from the previous section.

1. Open the solution in Visual Studio.
2. Before publishing your cloud service for the first time, you have to create a cloud service in the portal. To do this, log into the Azure Management Portal (*manage.windowsazure.com*) and click CLOUD SERVICES in the left column.
3. On the bottom of the page, click +NEW > CLOUD SERVICE > QUICK CREATE; you will be prompted for URL and REGION. The URL must be unique among all cloud services. Note that the domain of the URL will be cloudapp.net. Select a region close to you and click CREATE CLOUD SERVICE. Azure will create your cloud service. This has no instances yet; when you publish your application, Azure will start up the instances, install your application, and then

enable them for access.

4. In Visual Studio, right-click the cloud service project and select Publish. You will be prompted for the Microsoft account you use for Azure; log in. Now it shows your subscription in the Publish Windows Azure Application window. Click Next to get to the Settings screen.



5. Select the cloud service you just added via the portal. For Environment, you can select Production or Staging (more on that later). If you are going to do remote debugging, select a Build Configuration of Debug; otherwise, set it to Release. The Service Configuration is a list of configurations from Visual Studio. This one shows Cloud and Local, but you could have one for production, one for staging, and one for development, and select which one to use here. This is a good way to support multiple environments with different configuration settings (such as database connection strings) in a cloud service.

The advanced settings are shown in Figure 2-21.

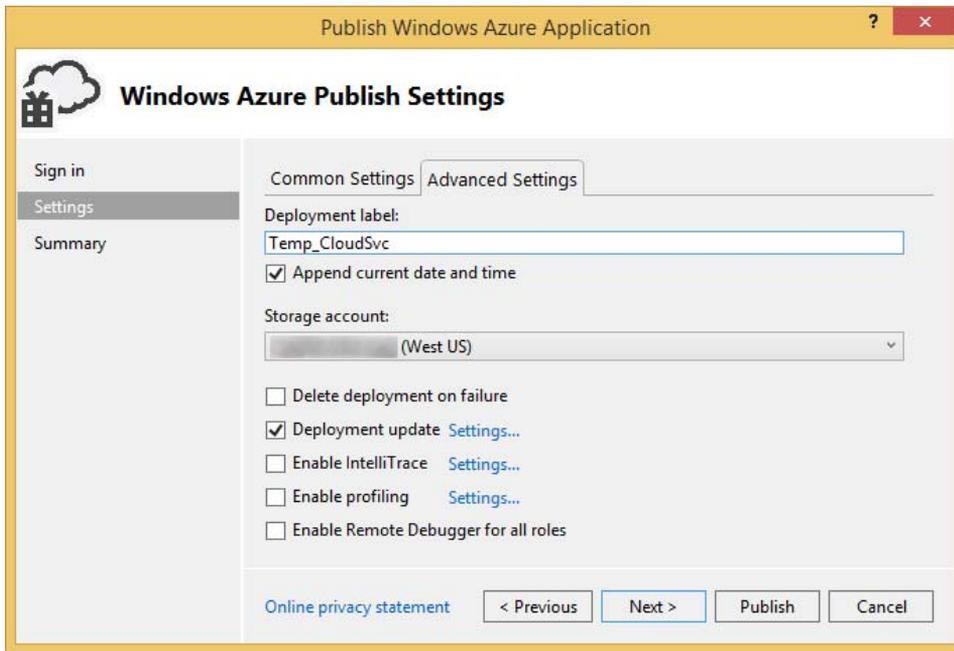


FIGURE 2-21 Advanced settings.

This is where you can set a deployment label. For example, you might put a version number in this field. You can also append the date and time to the deployment label. You will be able to see this in the portal, and it will help you know when it was last deployed.

The storage account is used to retain the package that is uploaded for you. The instances are built from the information in that package. The package consists of two files. One is the zipped versions of the application assemblies; the other is a configuration file specifying application configuration.

Accept the defaults here and click Next.

This will take you to a summary screen that will show all of your selections and let you save this publishing profile to be used again. After saving the publishing profile, click Publish.

Visual Studio will open a Windows Azure Activity Log window and display the progress of the deployment. It will verify the storage account, upload the package, create and start the instances with the right operating system on them, install your software, and then make the application available. This takes 5 to 10 minutes. The activity window will look similar to Figure 2-22 when publishing is completed.

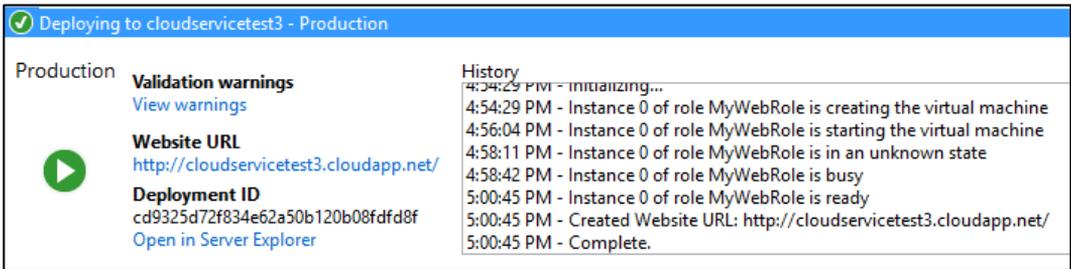


FIGURE 2-22 Publishing completed.

You can click the Website URL, and it will open your website in your default browser.

Another way to publish your cloud service is to right-click the cloud project and select Package and have Visual Studio create the deployment package for you. The package consists of two files. One is the zipped versions of the application assemblies; the other is a configuration file specifying application configuration.

Next you can go to the portal and do an UPDATE on PRODUCTION or select the STAGING environment and select UPLOAD. This will allow you to upload the package, and Azure will do the same thing as publishing from Visual Studio. It's just doing the upload package manually instead of having Visual Studio do it for you. You might want to do this if you have one group create the packages and another deploy them to staging or production.

Scaling and monitoring a cloud service

The easiest way to scale a cloud service is to set the VM Size and Instance count in the service configuration and republish. Another way to do this is to use the scaling options in the Azure Management Portal (*manage.windowsazure.com*). Log into the portal, and let's take a look at the features for the PaaS cloud service.

After logging in, click CLOUD SERVICES in the left column. Note that both kinds of cloud services are displayed here in the same list: the PaaS cloud services and cloud services that you create as a wrapper to one or more VMs, as discussed in the beginning of this section. Select the cloud service that you created and published in the previous section.

SCALE options

To scale a cloud service, go to the SCALE option. Here are the options available:

- SCALE BY SCHEDULED TIMES** To use this feature, click Set Up Schedule Times. This opens the schedule times screen. You can specify the start and end time for a day and scale settings for day versus night and weekdays versus weekends. You also can specify scale start and end times for specific dates. (This is identical to the Scale By Scheduled Times option for websites displayed in Figure 2-17.)

- SCALE BY METRIC: CPU or QUEUE** Several of the options are the same, whether scaling by CPU or QUEUE.

When scaling by CPU, you select the target CPU that will trigger the scaling event. For example, you might want it to scale when the CPU percentage is between 60 percent and 80 percent; see Figure 2-23.

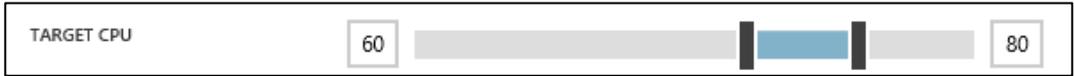


FIGURE 2-23 Setting target when scaling by CPU.

When scaling by queue, you select the storage account and queue name, as well as the target number of messages each instance can handle. If you take the total number of messages in the queue and divide it by the number of messages that each instance can handle, you get the number of instances that are needed. It will try to scale to this number or to the maximum number of instances, whichever is smaller. For example, if you set this to 500 and 2,000 messages come in, it will scale up to the maximum number of instances or four instances (2,000 messages divided by (500 messages/instance) = 4), whichever is smaller (see Figure 2-24).

FIGURE 2-24 Setting queue and queue target when scaling by queue.

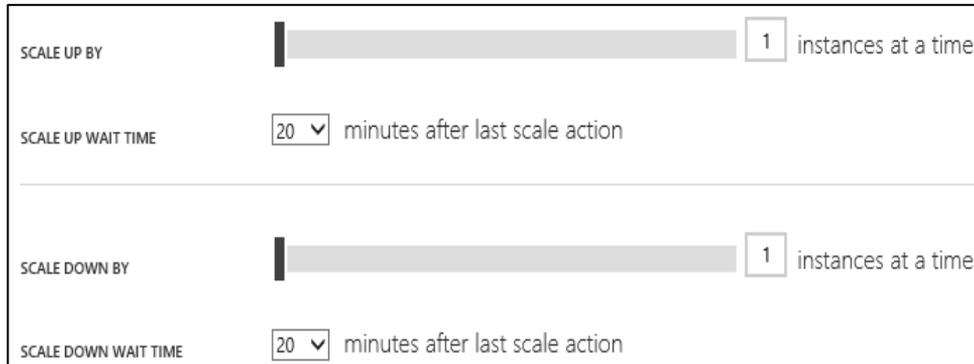
For both options, you can change the instance range. This is the minimum and maximum number of instances you want to have; see Figure 2-25.



FIGURE 2-25 Setting minimum and maximum number of instances.

For both options, you can select the number of instances by which to scale in or out with each autoscale event and how long to wait before scaling again. For example, you might want to scale out two instances at a time, but not more than once every 20 minutes. For scaling down, you might want

to scale in one instance at a time, but not more than once every 30 minutes in case the activity increases again; see Figure 2-26.



The image shows a configuration interface for Azure Autoscale. It is divided into two sections: 'SCALE UP' and 'SCALE DOWN'. Each section contains a slider for the number of instances to scale and a dropdown menu for the wait time. The 'SCALE UP BY' slider is set to 1, and the 'SCALE UP WAIT TIME' dropdown is set to 20 minutes. The 'SCALE DOWN BY' slider is also set to 1, and the 'SCALE DOWN WAIT TIME' dropdown is set to 20 minutes.

SCALE UP BY	<input type="range" value="1"/>	1 instances at a time
SCALE UP WAIT TIME	<input type="text" value="20"/>	minutes after last scale action
SCALE DOWN BY	<input type="range" value="1"/>	1 instances at a time
SCALE DOWN WAIT TIME	<input type="text" value="20"/>	minutes after last scale action

FIGURE 2-26 Setting number of instances to scale up or down with each scaling action.

An important thing to note: the monitoring framework reads about 15 minutes behind and the data it is reading is a 45-minute average, so it takes about an hour for the autoscale event to be triggered. If the CPU spikes to 75 percent and comes right back down, it will not trigger an autoscaling event; it must be a sustained spike.

Monitoring

There are two ways to monitor performance for a cloud service. One is through the Azure Management Portal (MONITOR tab), and the other is through Visual Studio. In the portal, go to the MONITOR tab for the cloud service and click +METRICS on the bottom of the page; see Figure 2-27. There is a minimal list of metrics available.

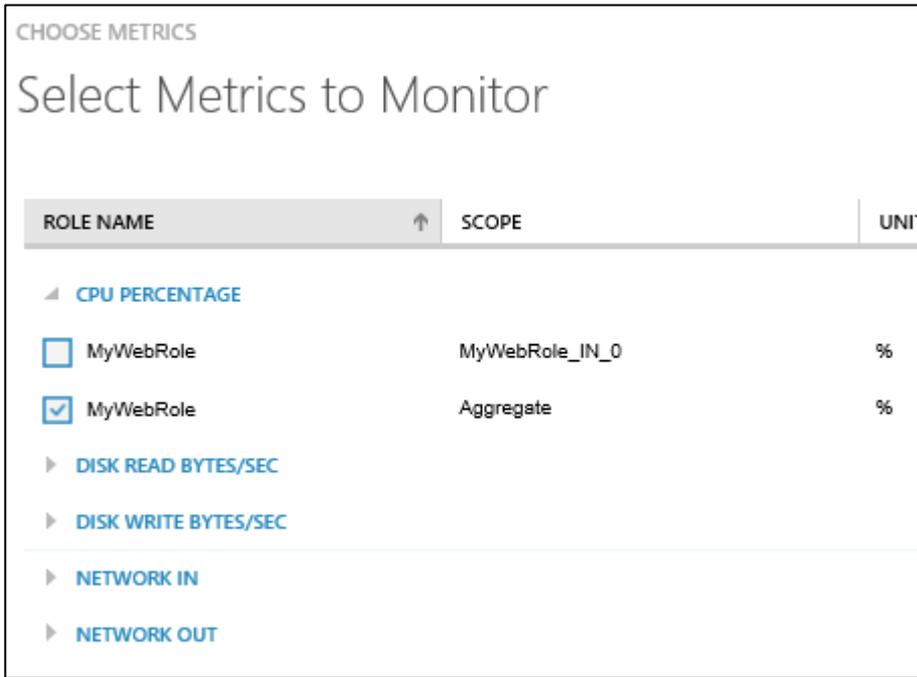


FIGURE 2-27 Selecting metrics to monitor.

If you go to the CONFIGURE tab and change the monitoring level to VERBOSE and let it run a bit, then click +METRICS again in MONITOR, you will find that many more performance metrics are available to add to MONITOR. Azure will write 5-minute, 1-hour, and 12-hour averages to tables in Azure Storage with names like WAD[deploymentid]PT1HRTTable. You can set the data retention time in days.

The other way to add performance metrics is through Visual Studio. Open your solution, go to the cloud service properties, and select the Configuration tab. You will find that you can define a Custom Diagnostics plan; see Figure 2-28.

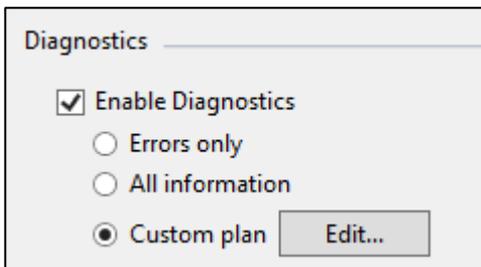


FIGURE 2-28 Enable diagnostics and choose to create a Custom plan.

Select Custom Plan and click Edit to open the Diagnostics configuration. On the Performance Counters tab, you will find many other metrics that you can add. You can even add performance counters not already in the list; see Figure 2-29.

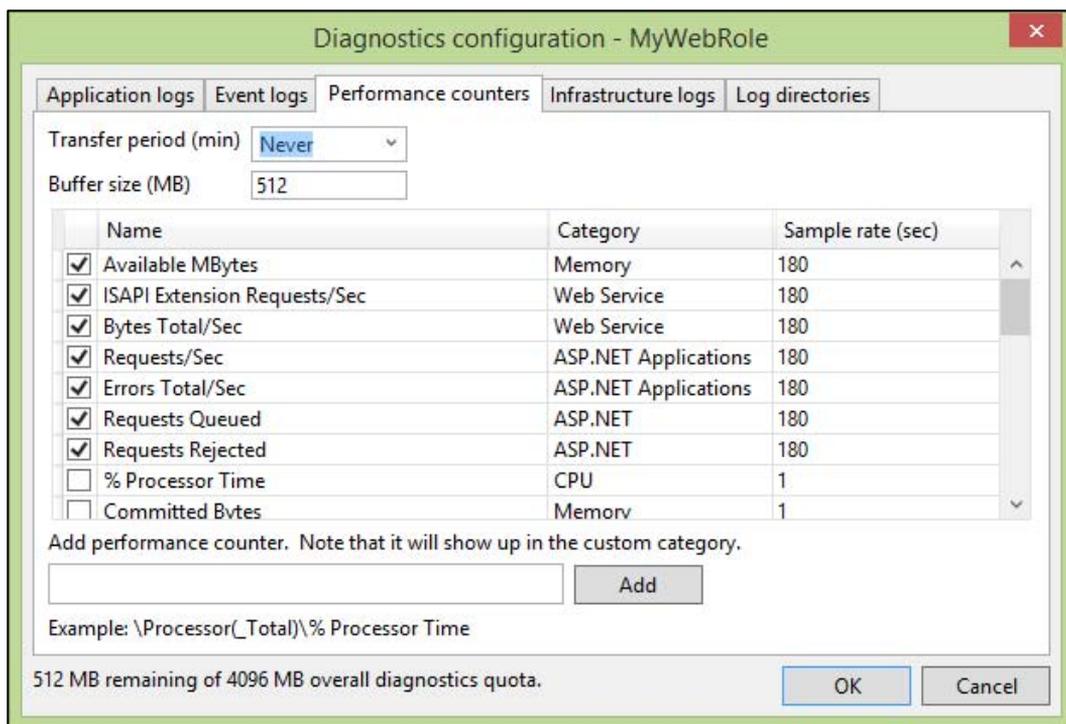


FIGURE 2-29 Adding performance counters using Visual Studio.

These diagnostics are written to the table called WADPerformanceCounterTable, and they are written roughly every five minutes as point-in-time values. There is no automatic retention time for this data; it will stay in the table until you remove it.

Of the two options, I recommend using the configuration in Visual Studio because the other tables have the deployment ID in the table name, and the deployment ID changes each time you deploy a cloud service. This means if you want to graph your metrics over time, you have to search each table. If you make the changes in Visual Studio, all of the metrics are stored in the same table, making it easier to query the information over time.

Miscellaneous points

Here a few items to keep in mind:

Configuration in the portal

You can modify the operating system family and version on the CONFIGURE tab in the Azure Management Portal. If you have configuration settings in your web role or worker role, you can edit those in the portal and save them. In both of these cases, the original package uploaded to Azure is not modified. This means if Azure installs patches on your VMs and reboots them, the manual changes you made through the portal will be lost. So if you make these changes via the portal, be sure to follow up by modifying the Visual Studio solution and deploying a new version.

Production and staging slots in the portal

Each cloud service has both a production and a staging slot. The production slot uses the URL assigned, such as *yourgreatwebapp.cloudapp.net*. The staging slot is assigned a globally unique identifier (GUID) for the URL, which changes each time you deploy a new package to it. This makes it difficult to test intertwined cloud services, such as A calls B calls C. You either have to change the address in A that points to B and the address in B that points to C, or you have to change the domain name system (DNS) entries, which can take time to propagate.

One way around this is to set up multiple cloud services for one web application in the Azure Management Portal. For example, you might set up MyWebApp for production and stMyWebApp for staging. Publish to staging and do your testing. When you are ready to put the changes in production, publish to the staging slot of MyWebApp and do a VIP Swap, which swaps the IP addresses for the two deployment slots, effectively putting the new one in production. Don't forget to delete the old one in the staging slot!

Worker roles

To add a worker role to your cloud project, right-click the Roles and select Add New Worker Role Project. Worker roles generally are used to retrieve messages from a queue and process them and then delete the messages from the queue.

A good example of this is having a web application that lets a customer upload pictures. You want to resize the pictures, so when the upload is completed, you write a message to the queue. The worker role would retrieve the message from the queue, resize the photos, put them in the destination folder, and then delete the message from the queue.

About the authors



Robin E. Shahan is a Microsoft Azure MVP with over 25 years of experience developing complex, business-critical applications for Fortune 100 companies. As the VP of Technology for the startup GoldMail (DBA PointAcross), she re-architected their entire infrastructure and migrated it to Microsoft Azure, reducing their costs by 90%. Robin is the President of Nightbird Consulting, focusing on helping companies architect and develop scalable and efficient solutions utilizing the Azure platform.

Robin regularly speaks about Microsoft Azure at various .NET User Groups and Code Camps and runs the San Francisco Azure meetup. She can be found on Twitter as [@RobinDotNet](https://twitter.com/RobinDotNet), and you can read her articles about Microsoft Azure (and other subjects) at <http://robindotnet.wordpress.com>.



Michael S. Collier is a five-time Microsoft Azure MVP and most recently served as a Principal Cloud Architect with Aditi Technologies. He has over 13 years of experience with various consulting and technology firms where he was instrumental in leading and developing solutions for a wide range of clients. He has a vast amount of experience in helping companies determine the best strategy for adopting cloud computing, and providing the insight and hands-on experience to ensure they are successful. Michael is also a respected technology community leader and is often found sharing his Microsoft Azure insights and experiences at regional and national conferences. You can follow Michael's

experiences with Azure on his blog at <http://www.michaelscollier.com> and on Twitter at [@MichaelCollier](https://twitter.com/MichaelCollier).

Michael lives in Marysville, Ohio. He is a 2003 graduate of The Ohio State University and is a passionate Buckeyes fan. Michael is also an avid golfer, although golf doesn't always like him.