



developer

// Step by step

# Microsoft Azure SQL Database

Intermediate



Leonard G. Lobel  
Eric D. Boyd

# Microsoft<sup>®</sup> Azure<sup>™</sup> SQL Database Step by Step

Leonard G. Lobel  
Eric D. Boyd

**PUBLISHED BY**

Microsoft Press  
A Division of Microsoft Corporation  
One Microsoft Way  
Redmond, Washington 98052-6399

Copyright © 2014 by Leonard G. Lobel and Eric D. Boyd

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.

Library of Congress Control Number: 2014940679  
ISBN: 978-0-7356-7942-9

Printed and bound in the United States of America.

First Printing

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](mailto:mspinput@microsoft.com). Please tell us what you think of this book at <http://aka.ms/tellpress>.

Microsoft and the trademarks listed at <http://www.microsoft.com/en-us/legal/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 Editor:** Devon Musgrave

**Project Editor:** Rosemary Caperton

**Editorial Production:** Waypoint Press, [www.waypointpress.com](http://www.waypointpress.com)

**Technical Reviewer:** Scott Klein; Technical Review services provided by Content Master, a member of CM Group, Ltd.

**Copyeditor:** Roger LeBlanc

**Indexer:** Christina Yeager

**Cover:** Twist Creative • Seattle and Joel Panchot

*To my partner of 20 years, Mark, and our children, Adam, Jacqueline, Joshua, and Sonny. With all my love, I thank you guys, for all of yours.*

—LEONARD LOBEL

*For my loving wife, Shelly, and our wonderful boys, Jaxon and Xander.*

—ERIC BOYD

*This page intentionally left blank*

# Contents at a glance

	<i>Introduction</i>	<i>xiii</i>
CHAPTER 1	Getting started with Microsoft Azure SQL Database	1
CHAPTER 2	Configuration and pricing	31
CHAPTER 3	Differences between SQL Server and Microsoft Azure SQL Database	57
CHAPTER 4	Migrating databases	63
CHAPTER 5	Security and backup	97
CHAPTER 6	Cloud reporting	123
CHAPTER 7	Microsoft Azure SQL Data Sync	173
CHAPTER 8	Designing and tuning for scalability and high performance	217
CHAPTER 9	Monitoring and management	261
CHAPTER 10	Building cloud solutions	289
	<i>Index</i>	<i>357</i>

*This page intentionally left blank*

# Contents

*Introduction* ..... *xiii*

## **Chapter 1 Getting started with Microsoft Azure SQL Database 1**

Cloud computing: The concept .....	1
Instant dynamic provisioning.....	2
The Microsoft Azure cloud.....	2
Getting signed up for SQL Database.....	5
Creating a Microsoft account.....	5
Creating a Microsoft Azure subscription .....	7
Creating a server.....	8
Creating a SQL Database instance.....	13
Using the SQL Database management portal .....	15
Designing tables and relationships.....	17
Inserting data .....	21
Querying the database .....	24
Exploring additional portal capabilities.....	27
Summary.....	30

## **Chapter 2 Configuration and pricing 31**

Using the Microsoft Azure platform management portal.....	31
Creating a new database .....	31
Setting firewall rules .....	35
Obtaining connection strings .....	38
Deleting a database.....	40
Using SQL Server Management Studio.....	40
Connecting to SQL Database.....	41
Creating a new database .....	43

---

**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](https://microsoft.com/learning/booksurvey)

Changing the database edition and maximum size. . . . .	44
Deleting a database. . . . .	44
Using PowerShell. . . . .	44
Installing the Microsoft Azure PowerShell cmdlets . . . . .	44
Using the PowerShell Integrated Scripting Environment . . . . .	46
Configuring PowerShell for your Microsoft account . . . . .	46
Creating a new server . . . . .	47
Creating a new database . . . . .	48
Deleting a database. . . . .	50
Budgeting for SQL Database. . . . .	50
SQL storage. . . . .	50
Client bandwidth . . . . .	51
Backup storage space . . . . .	51
Backup storage bandwidth. . . . .	52
Support . . . . .	53
Optimizing your costs . . . . .	54
Configuring the database edition and size . . . . .	55
Summary. . . . .	56
<b>Chapter 3 Differences between SQL Server and Microsoft Azure SQL Database</b>	<b>57</b>
Size limitations. . . . .	58
Connection limitations. . . . .	58
Unsupported features . . . . .	59
Summary. . . . .	62
<b>Chapter 4 Migrating databases</b>	<b>63</b>
Making the case for data migration . . . . .	63
Migrating data using Transact-SQL scripts. . . . .	64
Setting up a local SQL Server database. . . . .	64
Creating the T-SQL scripts . . . . .	67
Generating T-SQL scripts . . . . .	68
SQL Data-Tier Applications . . . . .	70
Creating a Microsoft Azure Storage account . . . . .	71
Exporting a BACPAC to Microsoft Azure Storage . . . . .	74
Importing a BACPAC to Microsoft Azure SQL Database. . . . .	77

SQL Server Bulk Copy (bcp) . . . . .	80
Migrating Schema . . . . .	81
Exporting data . . . . .	83
Importing data . . . . .	84
SQL Database Migration Wizard . . . . .	86
Downloading the tool . . . . .	87
Migrating a database . . . . .	88
Summary . . . . .	95

## **Chapter 5 Security and backup 97**

Addressing major cloud concerns . . . . .	97
Security responsibilities of the public cloud vendor . . . . .	98
Shared security responsibilities . . . . .	98
Security in Microsoft Azure . . . . .	99
Securing SQL Database . . . . .	100
Creating a SQL Database . . . . .	100
Configuring SQL Database Firewall . . . . .	101
Authenticating and authorizing users . . . . .	105
Backing up SQL Database . . . . .	112
Copying a database . . . . .	112
Monitoring the progress of a database copy operation . . . . .	113
Exporting a BACPAC . . . . .	115
Importing a BACPAC . . . . .	117
Scheduling BACPAC exports . . . . .	120
Summary . . . . .	122

## **Chapter 6 Cloud reporting 123**

Creating a SQL Server Reporting services virtual machine . . . . .	125
Creating the virtual machine from the image gallery . . . . .	126
Configuring SSRS in the virtual machine . . . . .	128
Opening firewall access to the report server . . . . .	130
Creating the sample database . . . . .	132
Using Report Builder . . . . .	135
Installing Report Builder . . . . .	135
Creating a report using Report Builder . . . . .	137

Using Visual Studio Report Server projects . . . . .	150
Installing AdventureWorks2012 for SQL Database . . . . .	152
Installing SSDT Business Intelligence for Visual Studio 2012 . . . . .	154
Creating a report using Visual Studio. . . . .	156
Implementing report security. . . . .	170
Shutting down the SSRS virtual machine . . . . .	171
Summary. . . . .	171
<b>Chapter 7 Microsoft Azure SQL Data Sync</b>	<b>173</b>
Getting to know SQL Data Sync. . . . .	173
Exporting data from SQL Server to SQL Database. . . . .	175
Importing data from SQL Database to SQL Server . . . . .	175
Sharing data between multiple locations . . . . .	176
Scaling out. . . . .	178
Creating the SQL Database . . . . .	180
Working with SQL Data Sync . . . . .	182
Creating a sync group. . . . .	182
Creating sync rules. . . . .	188
Running a manual sync. . . . .	191
Establishing conflict resolution . . . . .	196
Creating an automated sync schedule. . . . .	200
Creating a local SQL Server database. . . . .	202
Creating a sync agent . . . . .	204
Pitfalls and best practices . . . . .	215
Summary. . . . .	216
<b>Chapter 8 Designing and tuning for scalability and high performance</b>	<b>217</b>
Achieving high performance in the cloud . . . . .	218
Creating a RESTful web API. . . . .	218
Creating the sample database. . . . .	219
Creating a new solution . . . . .	221
Creating an ASP.NET Web API project . . . . .	222
Adding an Entity Framework Code First Web API controller . . . . .	223
Testing the Wine Web API . . . . .	228

Adding an ADO.NET Web API controller . . . . .	230
Testing the Customer Web API . . . . .	233
Managing SQL Database connections . . . . .	234
Opening late, closing early . . . . .	234
Pooling connections . . . . .	234
Recovering from connection faults . . . . .	234
Adding the Transient Fault Handling Application Block . . . . .	235
Using the Transient Fault Handling Application Block with ADO.NET . . . . .	237
Using the Transient Fault Handling Application Block with Entity Framework . . . . .	239
Reducing network latency . . . . .	243
Keeping services close . . . . .	243
Minimizing round trips . . . . .	243
Effectively using SQL Database . . . . .	244
Using the best storage service . . . . .	244
Optimizing queries . . . . .	245
Scaling up SQL Database . . . . .	245
Partitioning data . . . . .	250
Scaling out with functional partitions . . . . .	250
Scaling out with shards . . . . .	251
Summary . . . . .	260

**Chapter 9 Monitoring and management 261**

Creating the sample database . . . . .	261
Monitoring . . . . .	263
Using the management portal . . . . .	264
Microsoft Azure Service Dashboard . . . . .	269
SQL Database management portal . . . . .	270
Dynamic management views and functions . . . . .	275
Programming the Service Management REST API . . . . .	281
Summary . . . . .	288

## Chapter 10 Building cloud solutions 289

Creating the SQL Database .....	292
Extending the SQL Database .....	294
Creating a new solution .....	294
Creating a SQL Server Database project .....	295
Setting the target platform .....	296
Importing from SQL Database into the project .....	297
Adding a new column to the <i>Wine</i> table .....	300
Deploying the project to Microsoft Azure SQL Database .....	301
Creating the <i>Order</i> table .....	305
Creating stored procedures for the <i>Order</i> table .....	307
Creating the data access layer .....	312
Introducing the Entity Data Model .....	313
Creating the Data Access Layer project .....	314
Creating an Entity Data Model .....	315
Creating the website .....	321
Creating an ASP.NET web application project .....	321
Referencing the data access layer .....	323
Creating the user interface .....	324
Testing the website locally .....	328
Deploying the website to Microsoft Azure .....	331
Creating the ASP.NET Web API services .....	336
Adding a Web API controller .....	337
Testing the Web API .....	339
Deploying the Web API .....	340
Creating the Windows Phone application .....	341
Installing the Windows Phone SDK 8.0 .....	341
Creating the Windows Phone Project .....	343
Adding Json.NET .....	343
Creating the App's main page .....	344
Testing the Windows Phone application .....	353
<i>Index</i> .....	357

---

### 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](https://microsoft.com/learning/booksurvey)

# Introduction

Microsoft Azure SQL Database is the cloud version of Microsoft SQL Server, which is Microsoft's well-established on-premises relational database engine platform. Despite some noteworthy differences, SQL Database (the short name for Microsoft Azure SQL Database) is largely compatible with SQL Server, so for the most part, any experience you have working with SQL Server can be directly and immediately applied to SQL Database. If you are a software professional looking to consider the cloud as a platform for the database in your next application, SQL Database can be just the right tool for you. And if you want to get up to speed quickly with this emerging platform, with or without SQL Server experience, this is just the right book for you.

*Microsoft Azure SQL Database Step by Step* provides an organized walkthrough of the SQL Database platform. Our goal was to produce an end-to-end treatment of SQL Database that balances coverage and depth. In the first chapter, you will quickly create your first SQL Database on Microsoft Azure. By the last chapter, you will create a full multitiered solution in the cloud—including a website and a Windows Phone 8 app—all layered on top of SQL Database. And in every chapter in between, you will explore other facets of SQL Database and many of its orbiting technologies. SQL Database is a huge topic, but we carefully crafted each chapter to tackle one piece at a time, with easy-to-follow procedures that put digestible concepts to immediate applied use. Your knowledge will build in each chapter, as you learn about configuration, migration, security, backup, reporting, and more.

One big difference between on-premises software and cloud services is that the latter can be updated and enhanced much more frequently than the former, given that no installation or customer infrastructure is required in the cloud case. Cloud services are subject to frequent changes in pricing as well. As such, features, limitations, costs, the tooling user interface, or even the branding of Microsoft Azure SQL Database, as described in this book, may have evolved by the time you read it. For example, shortly before going to press, the platform formerly branded as Windows Azure was changed to Microsoft Azure. (Although the book title and textual references were updated accordingly, many screen shots still show the older name, *Windows Azure*.) Regardless of the potential for such changes, the principles and techniques covered throughout this book will help you achieve comfort with and mastery of Microsoft Azure SQL Database.



**Note** As Azure evolves, we evolve with it. Even as this first edition goes to press, we are busy planning the next edition with expanded coverage of the recently announced Basic, Standard, and Premium editions. These new service tiers (which have limited preview availability at the time of this writing) can support larger and more scalable databases than the current Web and Business editions offer. Our next edition will also be revised for the upcoming release of a new management portal currently being developed by Microsoft.

## Who should read this book

---

This book exists to help software developers and database professionals understand the core concepts of Microsoft Azure SQL Database and its related technologies. Many readers will have little or no prior experience with either SQL Database or SQL Server, and that's perfectly fine. This book starts with square one, and gets you off to a good start even if you have no prior knowledge of SQL Server or relational database concepts.

The book is also useful for those familiar with on-premises SQL Server and are interested in creating new applications to work with SQL Database, or those who would like to migrate existing applications that currently work with on-premises SQL Server to work with SQL Database as well.

## Assumptions

No prior knowledge or experience with Microsoft Azure and cloud computing is assumed or required. Furthermore, although experience with Microsoft SQL Server is certainly useful, that too is not required.

Several chapters involve .NET programming. Here, too, prior experience with Microsoft Visual Studio and C# is helpful but not required. The procedures in these chapters include complete code listings, and clear explanations of the code are provided.

## This book might not be for you if...

---

This book might not be for you if you already have extensive knowledge and experience with SQL Database, and are seeking to delve deeper into internals or other specialized focus areas not covered in this book. Still, this book contains useful information even for experienced users. Therefore, we recommend that you take a quick glance at the chapter descriptions in the next section. Doing so should help you quickly determine if there are specific areas of interest we cover that you would like to learn more about.

## Organization of this book

---

This book is composed of ten chapters, each of which focuses on a different aspect of Microsoft Azure SQL Database. Most readers will probably benefit by starting with Chapter 1, but by no means does this book need to be read in any particular order. Read it from start to finish if you want, or jump right in to just those chapters that suit your needs or pique your interests. Either way, you'll find practical guidance and walkthroughs to help get your job done with SQL Database.

- **Chapter 1—Getting started with Microsoft Azure SQL Database** The opening chapter gets you acquainted with the SQL Database platform. After a brief overview of cloud computing with SQL Database, you'll create a Microsoft account (if you don't already have one) and a Microsoft Azure subscription. Then you will learn how to use the Microsoft Azure management portal to create servers and databases. You'll move on to use the SQL Database management portal, where you'll design tables, views, and stored procedures, and then populate and query tables in the database. Absolutely no local tools are required to follow along with the procedures in this chapter; all you need is a web browser and Internet access.
- **Chapter 2—Configuration and pricing** With the basics covered, this chapter explains additional options for configuring SQL Database, beyond the browser-based portals introduced in Chapter 1. You will learn how to connect to SQL Database using familiar local tools, such as SQL Server Management Studio (SSMS) and SQL Server Data Tools (SSDT) inside Visual Studio. You will also learn how to configure and manage SQL Database using PowerShell, by downloading the Microsoft Azure PowerShell cmdlets. The chapter concludes with an explanation of how SQL Database pricing is structured on Microsoft Azure, and it provides tips to help you budget for a SQL Database solution.

- **Chapter 3—Differences between SQL Server and Microsoft Azure SQL Database** Readers with prior SQL Server experience will want to know about the important differences between the on-premises relational engine they are familiar with and the SQL Database implementation on Microsoft Azure. This brief chapter enumerates these differences and explains the rationale behind them. Where possible, we suggest workarounds for SQL Server features that are not supported in SQL Database.
- **Chapter 4—Migrating databases** When building systems on Microsoft Azure, there is often a need to migrate databases from existing on-premises SQL Servers to SQL Database. There are numerous techniques and tools you can use to migrate databases and data to SQL Database. In this chapter, you will learn about and use Transact-SQL scripts, SQL Data-Tier Applications, bulk copy, and the SQL Database Migration Wizard to migrate databases to SQL Database.
- **Chapter 5—Security and backup** Security, availability, and disaster recovery top the list of concerns when customers consider new data centers and public cloud providers. In this chapter, you will learn about security in Microsoft Azure and how to secure your SQL Database with firewall rules, as well as users and permissions. In addition to gaining knowledge about security, you will learn how to handle disaster recovery with SQL Database backup techniques.
- **Chapter 6—Cloud reporting** When you have data in a database, it's only a matter of time before you also have reporting requirements related to that data. And when that database is hosted in the cloud on Microsoft Azure, it's only natural to consider using the Azure cloud to host a reporting solution as well. In this chapter, you will learn how to create an Azure virtual machine (VM) to host SQL Server Reporting Services (SSRS) in the cloud. (No prior SSRS experience is needed.) Once the VM is configured, you will learn how to build SSRS reports using two report authoring tools: Report Builder and SSDT Business Intelligence for Visual Studio. After building and previewing reports locally, you will learn how to deploy them to the VM for a complete reporting solution in the cloud.
- **Chapter 7—Microsoft Azure SQL Data Sync** In this chapter, you will learn how to use the SQL Data Sync service available on Microsoft Azure to replicate data between multiple databases. You will learn about the hub-and-spoke architecture upon which the service is based, and see how SQL Data Sync can be used to implement solutions for a variety of scenarios, including one-way or bi-directional replication across a set of databases in multiple locations. The procedures in this chapter walk you through the process of configuring the SQL Data Sync service and creating sync groups that replicate between multiple databases

hosted both in the cloud (on Microsoft Azure SQL Database) and on-premises (using SQL Server). You will also learn how to establish a conflict-resolution strategy and set up an automated synchronization schedule.

- **Chapter 8—Designing and tuning for scalability and high performance** Applications and systems intended for real production use need to provide responsive experiences and good performance. In this chapter, you will optimize and tune database performance for SQL Database. Next, you will improve application reliability by managing database connections and connection errors using both ADO.NET and Entity Framework. Finally, you will explore how to scale databases in SQL Database using a special partitioning technique known as *sharding*.
- **Chapter 9—Monitoring and management** Services used by production applications must provide monitoring and management capabilities. In this chapter, you will learn how to monitor the health of SQL Database using the management portal, the Service Dashboard, and dynamic management views and functions. You will also learn how to automate SQL Database operations programmatically, using the REST-based Service Management API.
- **Chapter 10—Building cloud solutions** In the book’s closing chapter, you will learn how to build a complete solution in the cloud on top of Microsoft Azure SQL Database. Specifically, you will create a Visual Studio solution that includes a SQL Server Database project, an Entity Framework data-access layer, ASP.NET MVC, and ASP.NET Web API. The solution provides a website, web services, and a Windows Phone 8 app with functionality for users to retrieve and update data stored in SQL Database.

## Conventions and features in this book

---

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

- Each procedure consists of a series of tasks, presented as numbered steps (1, 2, and so on) listing each action you must take to complete the exercise.
- Boxed elements with labels such as “Note” provide additional information or alternative methods for completing a step successfully.
- Text that you type (apart from code blocks) appears in bold.

- 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 the Tab key.
- A vertical bar between two or more menu items (for example, File | Close) means that you should select the first menu or menu item, then the next, and so on.

## System requirements

---

At a minimum, there are no special system requirements for working with SQL Database. The Microsoft Azure management portal requires only a web browser and Internet access. Similarly, the SQL Database management portal requires only a browser with the Silverlight plug-in.

Some chapters walk you through procedures that use local tools—typically, SQL Server Management Studio (SSMS) and Visual Studio 2013. To complete these procedures, you will need to have those tools installed as well, which requires the following:

- One of Windows 7, Windows 8, Windows Server 2008 with Service Pack 2, Windows Server 2008 R2, or Windows Server 2012.
- Visual Studio 2013, any edition. (Multiple downloads may be required if using Express Edition products.)
- SQL Server 2012 Express Edition or higher, with SQL Server Management Studio 2012 Express or higher. (Included with Visual Studio, Express Editions require separate download.)

Depending on your Windows configuration, you might require Local Administrator rights to install or configure Visual Studio 2013 and SQL Server 2012 products.

Chapter 4, “Migrating databases,” and Chapter 7, “Microsoft Azure SQL Data Sync,” include procedures that require a local SQL Server instance on which you have permissions to create a database. If you don’t have access to a local SQL Server instance, you can install SQL Server Express Edition (the free version of SQL Server) by following the instructions shown in the next section.

Finally, several individual chapters work with additional software that gets installed locally. These chapters include detailed procedures for downloading and installing the necessary software so that you can follow along with the rest of the chapter.

## Downloads: SQL Server Express Edition

---

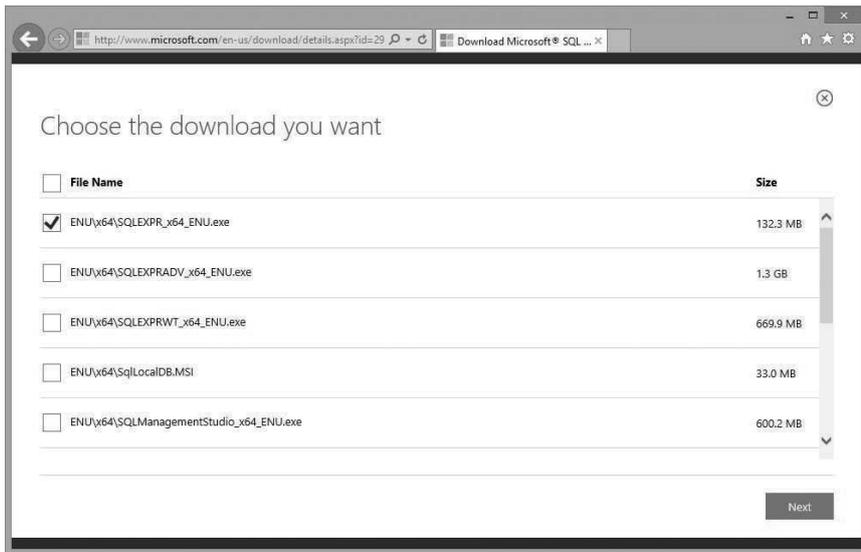
There are several SQL Server Express Edition downloads available on the Microsoft site, and they are available in both 32-bit and 64-bit versions. You can choose to install just the SQL Server Express database engine (and nothing else), or you can choose one of two other (larger) downloads: Express With Tools (which includes SQL Server Management Studio [SSMS]) or Express With Advanced Services (which includes SSMS, Full Text Search, and Reporting Services). There are also separate downloads for SSMS and LocalDB, but these do not include the SQL Server Express database engine needed to host local databases.

To install the SQL Server Express Edition database engine, follow these steps:

1. Open Internet Explorer, and navigate to <http://www.microsoft.com/en-us/download/details.aspx?id=29062>.
2. Click the large orange Download button.
3. Select the appropriate download for your system, as shown in Figure I-1:
  - a. For 64-bit systems, choose *ENU\x64\SQLEXPRESS\_x64\_ENU.exe*.
  - b. For 32-bit or 64-bit WoW systems, choose *ENU\x86\SQLEXPRESS\_x86\_ENU.exe*.
  - c. For 32-bit systems, choose *ENU\x86\SQLEXPRESS\_x86\_ENU.exe*.

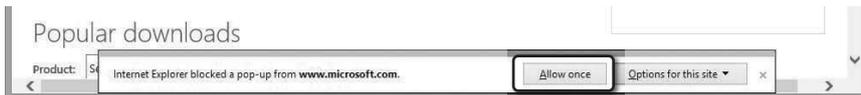


**Note** If you need to download SQL Server Management Studio (SSMS) as well, choose the Express With Tools file instead, which is the one that includes *WT* in the filename.



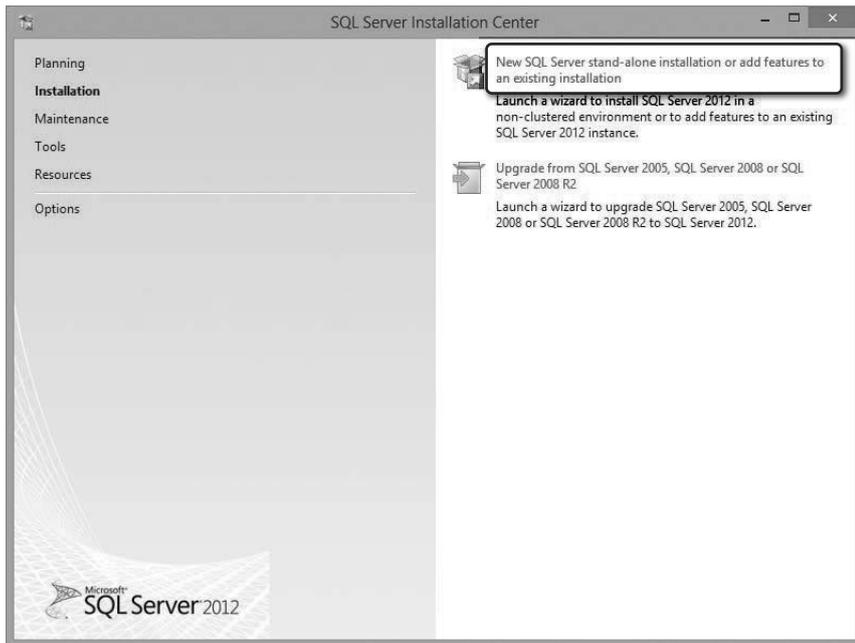
**FIGURE I-1** Downloading SQL Server 2012 Express Edition (64-bit version)

4. Click Next.
5. If you receive a pop-up warning, click Allow Once, as shown in Figure I-2.



**FIGURE I-2** Temporarily allowing pop-ups to enable the download, if necessary

6. When prompted to run or save the file, choose Run. This starts and runs the download.
7. If the User Account Control dialog appears after the download files are extracted, click Yes.
8. In the SQL Server Installation Center, click New SQL Server Stand-Alone Installation, as shown in Figure I-3.



**FIGURE I-3** Choosing a new SQL Server installation

- 9.** In the SQL Server 2012 Setup wizard, do the following:
  - a.** On the License Terms page, select I Accept The License Terms and click Next.
  - b.** On the Product Updates page, allow the wizard to scan for updates, and then click Next.
  - c.** On the Install Setup Files page, wait for the installation to proceed.
  - d.** On the Feature Selection page, Click Next.
  - e.** Continue clicking Next through all the remaining pages until the Installation Progress page, and wait for the installation to proceed.
  - f.** On the Complete page indicating a successful setup (shown in Figure I-4), click Close.

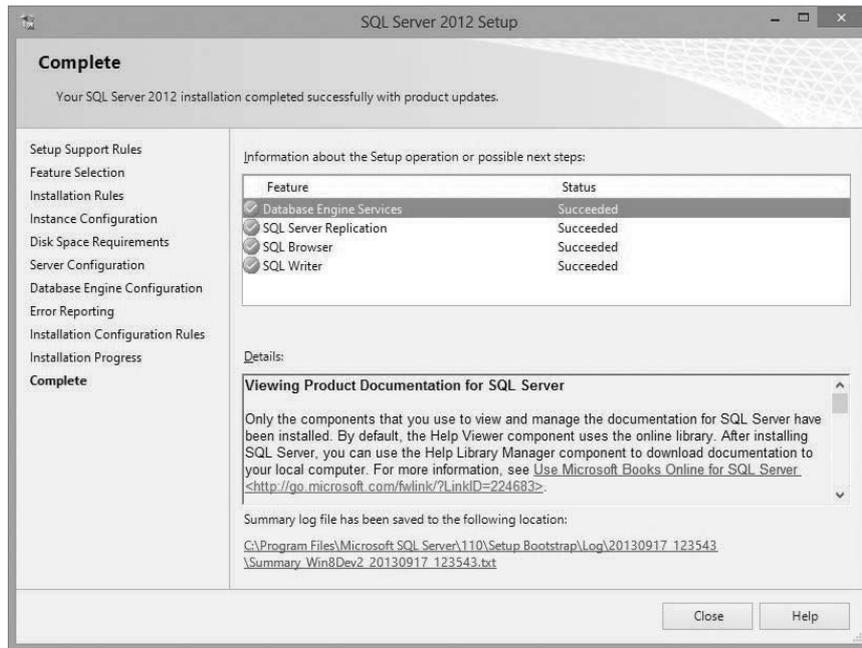


FIGURE I-4 SQL Server Express installation in progress

## Downloads: Code samples on the book's companion website

Many chapters have procedures in which you will write actual code. In most cases, there is only a small amount of code, but you may still find it helpful to download the completed code listings from the book's companion website. Doing so can help you work through the procedures, particularly those few procedures that have a bit more code than others. All the code found in this book can be downloaded at the following page:

[http://aka.ms/AzureSQLDB\\_SBS](http://aka.ms/AzureSQLDB_SBS)

## Installing and using the code samples

Simply download the zip file to your local machine and extract it. You will find one folder for the code in each chapter, although note that there is no code for Chapters 2 and 3. The code folders contain listing files that correspond to the listing numbers

found in each of the chapters. Finally, the code folders for Chapters 6, 8, and 10 also include the completed Visual Studio solutions for the exercises found in those chapters.

## Acknowledgments

---

I was first asked to write a book on SQL Azure—back when it was still *called* SQL Azure—nearly two years ago. It’s been a long road since then, and despite seismic shifts both in the Azure product platform and in the book publishing ecosystem (not to mention an unexpected curve ball or two), I am extremely delighted to finally publish!

This is my third technical book, and although each experience has been unique, I’ve learned the same lesson in each case: I could not have even contemplated the challenge without the aid of numerous other talented and caring individuals. These are folks who deserve special recognition—people who lent their generous support out in so many different ways that it’s impossible to mention names in any prescribed order.

So I’ll start with Andrew Brust. If not for Andrew (who himself is a well-established leader in the software industry), I would never have started down the book-writing path in the first place. I am grateful for our personal friendship, as well as our working relationship writing books and presenting workshops together. These experiences truly help me thrive and grow.

I’m also fortunate to have teamed up with my colleague and co-author Eric Boyd, who produced four excellent chapters on several advanced topics. Eric is an extremely talented software professional, whose expertise and passion for technology comes through clearly in his writing.

Russell Jones, my pal at O’Reilly Media, gets special mention of course, because he’s the one who asked me to write this book in the first place. I thank Russell, not only for offering me the opportunity, but for his expert guidance and assistance during the transition to Microsoft Press. More thanks go out to Roger LeBlanc for his copyediting review, and to Scott Klein for his technical review. Special thanks as well to Devon Musgrave and Rosemary Caperton at Microsoft Press, and Steve Sagman of Waypoint Press. Their guidance has been vital to the successful production of this book, and it has been an absolute pleasure working with each one of them.

I would like to give special mention to the Microsoft MVP program, which was an indispensable resource during the writing of this book. So thank you Microsoft, and to my MVP lead Simon Tien as well, for his constant encouragement.

This book could not have been written, of course, without the love and support of my family. I owe an enormous debt of gratitude to my wonderful partner Mark, and our awesome kids Adam, Jacqueline, Josh, and Sonny, for being so patient and tolerant with me throughout this project.

And greatest thanks of all go out to my dear Mom, bless her soul, for always encouraging me to write with “expression.”

—*Leonard Lobel*

I have been developing software professionally for almost 20 years and I am grateful for being blessed with deep interest and excitement for this industry, the ability learn and understand what are sometimes very complex concepts, and the support of my family, friends, mentors and peers throughout my career. Writing a book like this requires lots of guidance and help from many people, and I have many people to thank.

First and foremost, I want to thank God for everything: for life, salvation, family, friends, talents, abilities and everything.

Working on this project over the past year has been a lot of fun, but it has also been a lot of work. My family has been extremely supportive, even when I had to block off nights and weekends to write. I owe so much to my wife, Shelly, for everything that she does for our family. And I’m so thankful for our two wonderful boys who enjoy sitting next to me in my office and cuddling up next to me with my laptop in the living room, when my evenings and weekends get occupied with writing.

In addition to family, I want to thank friends and co-workers who have also been very supportive during this project, even when I bring my laptop to their living rooms, kitchens and dining rooms so that I can write a few more words, paragraphs and pages.

I want to thank Lenni Lobel who invited me to join him on this project. Lenni has been a fantastic co-author and has done a great job leading this project and driving it to completion. His guidance, editing and feedback has been extremely valuable for me personally and for the project. I’m also very appreciative of his patience throughout this project.

Last, but certainly not least, thank you to everyone at Microsoft and Microsoft Press who have helped with this project both directly and indirectly, this list includes Scott Klein, Dora Chan, Mark Brown, Devon Musgrave, Rosemary Caperton, Steve Sagman, Conor Cunningham, the Azure CAT team, and so many more.

—*Eric Boyd*

## Errata, updates, & book support

---

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

*[http://aka.ms/AzureSQLDB\\_SBS](http://aka.ms/AzureSQLDB_SBS)*

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](mailto: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>*.

## 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>*

*This page intentionally left blank*

# Differences between SQL Server and Microsoft Azure SQL Database

—Leonard Lobel

One of the most attractive aspects of Microsoft Azure SQL Database is that it shares virtually the same codebase and exposes the same tabular data stream (TDS) as on-premises Microsoft SQL Server. Thus, to a great extent, the same tools and applications that work with SQL Server work just the same and just as well with SQL Database. Notice that we said *to a great extent*, because despite their commonality, there are quite a few SQL Server features that SQL Database does not support. In this brief chapter, we discuss how and why these two platforms differ from one another, and we explain the SQL Database constraints you need to be aware of if you have previous experience with SQL Server.

SQL Server and SQL Database differ in several ways—most notably, in terms of size limitations, feature support, and T-SQL compatibility. In many cases, these constraints are simply the price you pay for enjoying a hassle-free, self-managing, self-healing, always-available database in the cloud. That is, Microsoft cannot responsibly support features that impair its ability to quickly replicate, relocate, and scale a SQL Database instance. This is why SQL Database places limits on database size and doesn't support certain specialized features, such as FILESTREAM.

Another common reason why a particular feature or T-SQL syntax might not be supported in SQL Database is that it's simply not applicable. With SQL Database, administrative responsibilities are split between Microsoft and you. Microsoft handles all the physical administration (such as disk drives and servers), while you manage only the logical administration (such as database design and security). This is why any and all T-SQL syntax that relates to physical resources (such as path names) are not supported in SQL Database. For example, you don't control the location for primary and log file groups. This is why you can't include an *ON PRIMARY* clause with a *CREATE DATABASE* statement, and indeed, why SQL Database does not permit a file group reference in *any* T-SQL statement. Plainly stated, everything pertaining to physical resources (that is, infrastructure) is abstracted away from you with SQL Database.

Yet still, in some cases, a certain SQL Server feature or behavior might be unsupported merely because Microsoft has just not gotten around to properly testing and porting it to SQL Database. Azure is constantly evolving, so you need to keep watch for updates and announcements. This small

chapter is a great starting point, but the best way to stay current is by reviewing the “Guidelines and Limitations” section of the SQL Database documentation on the MSDN website. (See <http://msdn.microsoft.com/en-us/library/ff394102.aspx>.)

## Size limitations

---

With the exception of the free, lightweight Express edition of SQL Server, there is no practical upper limit on database size in any edition of SQL Server. A SQL Server database can grow as large as 524,272 terabytes. (For SQL Server Express edition, the limit is 10 gigabytes.)

In contrast, SQL Database has very particular size limitations. As explained in Chapter 2, “Configuration and pricing,” you can set the maximum size by choosing between the Web and Business editions. With a Web edition database, you can set the maximum database size to either 1 or 5 gigabytes (GB). With a Business edition database, the maximum database size can range from 10 to 150 GB. The absolute largest supported database size is 150 GB, although partitioning strategies can be leveraged for scenarios that require databases larger than 150 GB (as explained in Chapter 8, “Designing and tuning for scalability and high performance”).



**Note** At the time this book went to press, Microsoft announced the Preview availability of SQL Database Premium. This is a more costly option than the standard Web and Business editions of SQL Database (which have been rebranded as Basic and Standard), and it supports a maximum database size of 500 GB. As we explain in Chapter 8, SQL Database Premium also lets you scale up for performance using dedicated CPU and memory.

## Connection limitations

---

SQL Database is far less flexible than SQL Server when it comes to establishing and maintaining connections. Keep the following in mind when you connect to SQL Database:

- SQL Server supports a variety of client protocols, such as TCP/IP, Shared Memory, and Named Pipes. Conversely, SQL Database allows connections only over TCP/IP.
- SQL Database does not support Windows authentication. Every connection string sent to SQL Database must always include a login user name and password.
- SQL Database often requires that `@<server>` is appended to the login user name in connection strings. SQL Server has no such requirement.
- SQL Database communicates only through port 1433, and it does not support static or dynamic port allocation like SQL Server does.
- SQL Database *does* fully support Multiple Active Result Sets (MARS), which allows multiple pending requests on a single connection.

- Because of the unpredictable nature of the Internet, SQL Database connections can drop unexpectedly, and you need to account for this condition in your applications. Fortunately, several options are available to cope with this:
  - The latest version of the Entity Framework (EF6, Microsoft’s recommended data access API for .NET) has a new Connection Resiliency feature, which automatically handles the retry logic for dropped connections.
  - The Microsoft Enterprise Library Transient Fault Handling Application Block, covered in Chapter 4, lets you define and implement retry strategies to deal with dropped connections.
  - The ADO.NET *SqlConnection* class has an *OpenWithRetry* extension method that handles the retry logic based on the default retry policy (which must be defined using the Microsoft Enterprise Library Transient Fault Handling Application Block).

## Unsupported features

---

This section lists many SQL Server capabilities that are not supported in SQL Database, and here we suggest workarounds where possible. Again, because this content is subject to change, we recommend you check the MSDN website for the latest information. (See <http://msdn.microsoft.com/en-us/library/ff394102.aspx>.)

- **Agent Service** You cannot use the SQL Server Agent service to schedule and run jobs on SQL Database.
- **Audit** The SQL Server auditing feature records server and database events to either the Windows event log or the file system, and it is not supported in SQL Database.
- **Backup/Restore** Conventional backups with the *BACKUP* and *RESTORE* commands are not supported with SQL Database. However, SQL Database supports an automated backup schedule that creates transactionally consistent backups in the form of BACPAC files created in Azure storage. You can also create BACPAC files manually; however, this does not provide transactional consistency for changes made during the export operation. To ensure transactional consistency for a manual backup, you can either set the database as read-only before exporting it to a BACPAC, use the Database Copy feature to create a copy of the database with transactional consistency and then export that copy to a BACPAC file. See Chapter 5, “Security and backup,” for more information.
- **Browser Service** SQL Database listens only on port 1433. Therefore, the SQL Server Browser Service, which listens on various other ports, is unsupported.
- **Change Data Capture (CDC)** This SQL Server feature monitors changes to a database, and it captures all activity related to change tables. CDC relies on a SQL Server Agent job to function and is unsupported in SQL Database.

- **Common Language Runtime (CLR)** The SQL Server CLR features (often referred to simply as *SQL CLR*) allow you to write stored procedures, triggers, functions, and user-defined types in any .NET language (such as Microsoft C# or Visual Basic) as an alternative to using traditional T-SQL. In SQL Database, only T-SQL can be used; SQL CLR is not supported. Note, however, that this limitation does not apply to SQL Server data types implemented internally using the CLR (such as *xml*, *geography*, and *geometry*, all of which are supported in SQL Database).
- **Compression** SQL Database does not support the data-compression features found in SQL Server, which you use to compress tables and indexes.
- **Database object naming convention** In SQL Server, multipart names can be used to reference a database object in another schema (with the two-part name syntax *schema.object*), in another database (with the three-part name syntax *database.schema.object*), and (if you configure a linked server) on another server (with the four-part name syntax *server.database.schema.object*). In SQL Database, two-part names can also be used to reference objects in different schemas. However, three-part names are limited to reference only temporary objects in *tempdb* (that is, where the database name is *tempdb* and the object name starts with a # symbol); you cannot access other databases on the server. And you cannot reference other servers at all, so four-part names can never be used.
- **Extended events** In SQL Server, you can create extended event sessions that help to troubleshoot a variety of problems, such as excessive CPU usage, memory pressure, and deadlocks. This feature is not supported in SQL Database.
- **Extended stored procedures** You cannot execute your own extended stored procedures (which are typically custom-coded procedures written in C or C++) with SQL Database. Only conventional T-SQL stored procedures are supported.
- **File streaming** SQL Server native file-streaming features, including FILESTREAM and FileTable, are not supported in SQL Database. Instead, you can consider using Azure Blob Storage containers for unstructured data files, but it will be your job at the application level to establish and maintain references between SQL Database and the files in blob storage, though note that there will be no transactional integrity between them using this approach.
- **Full-Text Searching (FTS)** The FTS service in SQL Server that enables proximity searching and querying of unstructured documents is not supported in SQL Database. However, there is a third-party text search engine library available from Lucene that does work with SQL Database. For more information, visit <http://www.lucene.net>.
- **Mirroring** SQL Database does not support database mirroring, which is generally a non-issue because Microsoft is ensuring data redundancy with SQL Database, so you don't need to worry about disaster recovery. This does also mean that you cannot use SQL Database as a location for mirroring a principal SQL Server database running on-premises. However, if you want to consider the cloud for this purpose, you can host SQL Server inside an Azure virtual machine (VM) against which you can mirror an on-premises principal database. This solution requires that you also implement a virtual private network (VPN) connection between

your local network and the Azure VM, although it will work even without the VPN if you use server certificates.

- **Partitioning** With SQL Server, you can partition tables and indexes horizontally (by groups of rows) across multiple file groups within a database, which greatly improves the performance of very large databases. SQL Database has a maximum database size of 150 GB (or 500 GB, for the newly announced Premium edition) and gives you no control over file groups, thus it does not support table and index partitioning.
- **Replication** SQL Server offers robust replication features for distributing and synchronizing data, including merge replication, snapshot replication, and transactional replication. None of these features are supported by SQL Database; however, SQL Data Sync can be used to effectively implement merge replication between a SQL Database and any number of other SQL Databases on Microsoft Azure and on-premises SQL Server databases. See Chapter 7, “Microsoft Azure SQL Data Sync,” for more information.
- **Resource Governor** The Resource Governor feature in SQL Server lets you manage workloads and resources by specifying limits on the amount of CPU and memory that can be used to satisfy client requests. These are hardware concepts that do not apply to SQL Database, so the Resource Governor is unsupported.
- **Service Broker** SQL Server Service Broker provides messaging and queuing features, and it is not supported in SQL Database.
- **System stored procedures** SQL Database supports only a few of the system stored procedures provided by SQL Server. The unsupported ones are typically related to SQL Server features and behaviors not supported by SQL Database. At the same time, SQL Database provides a few new system stored procedures not found in SQL Server that are specific to SQL Database (for example, *sp\_set\_firewall\_rule*).
- **Tables without a clustered index** Every table in a SQL Database must define a clustered index. By default, SQL Database will create a clustered index over the table’s primary key column, but it won’t do so if you don’t define a primary key. Interestingly enough, SQL Database will actually let you create a table with no clustered index, but it will not allow any rows to be inserted until and unless a clustered index is defined for the table. This limitation does not exist in SQL Server.
- **Transparent Data Encryption (TDE)** You cannot use TDE to encrypt a SQL Database like you can with SQL Server.
- **USE** In SQL Database, the *USE* statement can refer only to the current database; it cannot be used to switch between databases as it can with SQL Server. Each SQL Database connection is tied to a single database, so to change databases, you must connect directly to the database.
- **XSD and XML indexing** SQL Database fully supports the *xml* data type, as well as most of the rich XML support that SQL Server provides, including XML Query (XQuery), XML Path (XPath), and the *FOR XML* clause. However, XML schema definitions (XSD) and XML indexes are not supported in SQL Database.

## Summary

---

In this brief chapter, you learned about the important differences between on-premises SQL Server and SQL Database on Microsoft Azure. We explained the SQL Database limitations on size, as compared to a virtually unlimited database size supported by SQL Server. We also discussed connection limitations, and important considerations to keep in mind with respect to dropped connections, which occur with relative frequency in SQL Database. The chapter concluded by enumerating the many SQL Server features that are either unsupported or have limited support in SQL Database, and offered workarounds where possible.

The information in this chapter will help you decide whether or not SQL Database is suitable for your particular scenario. Of course, if you determine that it is not, always remember that you can run on-premises SQL Server in an Azure VM (we show you how in Chapter 6). This IaaS approach provides you with full SQL Server functionality in the cloud, compared to the PaaS approach of going with SQL Database.

*This page intentionally left blank*

# Index

## A

- abstraction, 3–4, 217, 313
- access control, 12, 35–38, 106–111, 170
  - least privilege principle, 108–109, 111
  - with stored procedures, 307–312
- action methods, 321, 326–327, 338
- Active Directory, adding users, 170
- ADO.NET, 217
  - connection pooling, 234
  - data access layer, building, 312–313
  - database connections, managing, 234
  - Dataset* objects and data adapters, 313
  - encapsulating, 239
  - raw objects, 312–313
  - sharding with, 254–259
  - SqlConnection* class *OpenWithRetry* extension method, 59
  - transient fault handling, 237–238
  - Web API controllers, building, 230–233
- AdventureWorks database, installing, 151–154
- agent keys, 204
- Agent service, 59
- agents, 174
- aging policies, 56
- allowed IP addresses, 12
  - See also* firewall rules; SQL Database Firewall
- ALTER DATABASE statement, 44
- applications
  - auditing, 99
  - building, 289–354
  - in the cloud, 2
  - connection faults, recovering from, 235–243
  - data synchronization, 179
  - geographic locations, 179, 243
  - hackers, protecting from, 99
  - health, monitoring, 261
  - load balancing, 179–180
  - physical distance from databases, 55, 179, 243
  - regulatory requirements, complying with, 99
  - repository pattern, 314
  - scaling, designing for, 218
  - tables, indirect access to, 307–312
  - volume purchase plan, 55
- ASP.NET MVC websites, 321–335
- ASP.NET Web APIs *See* Web API controllers; Web API services; Web APIs
- attack surface, reducing, 108. *See also* security
- auditing, 59, 99
- authentication
  - with management certificates, 281, 283
  - selecting type, 41
  - SQL Database support for, 58
  - of users, 105–111
  - Windows Integrated authentication, 170
- authorization of users, 105–111, 170
- automated BACPAC export schedule, 120–121
- automated data synchronization schedules, 200–202
- Automated Export feature, 120–121
- Azure *See* Microsoft Azure

## B

- backup storage, 51–53
- backup-and-restore process, 59, 112–121
- backups, 112–121
  - automated, 59
  - geo-replicated storage for, 55
  - removal strategy, 55
  - transactional consistency, 59
- BACPAC files, 30
  - for backups and restores, 112, 115–121
  - creating, 34–35, 59
  - data migration with, 70–80
  - database objects for export, 75
  - exporting, 34–35, 74–77, 115–117, 120–121

## BACPAC files

- BACPAC files (*continued*)
  - importing to SQL Database instance, 77–80
  - transactional consistency of, 120–121
- bandwidth-based fees, 9, 79
- basic authentication, 170
- bcp (SQL Server Bulk Copy), 80–86, 153
- bi-directional binding, 350
- bi-directional synchronization, 195–196
- blob containers, 73–74
- blob storage, 54, 60, 115, 244–245
- block blobs, 115
- Browser role, 170
- Browser Service, 59
- browsers
  - content negotiation, 230
  - MVC websites, running from, 335
  - reports, running from, 149–150, 168–169
  - Web APIs, testing, 228–230
- Business Intelligence Developer Studio (BIDS), 151
- Business Intelligence project templates, installing, 151, 154–156
- Business (Standard) edition of SQL Database, 55, 58

## C

- Cache service, 244–245
- caching, client-side, 244
- catalog views, 280
- Change Data Capture (CDC), 59
- change scripts, 301, 303
- check constraints, 310
- circular references, 338–339
- class library projects, 314–315, 323, 324
- client agent service, 204
- client bandwidth pricing, 51
- client connections, allowing through firewall, 101–105
- client protocols, SQL Database support for, 58
- clients, pushing data changes to, 174
- client-side storage and caching, 244
- cloud computing, 1–4, 218
- cloud databases. *See also* databases; SQL Database
  - location of, 215
  - synchronizing, 182–202
- cloud reporting, 3, 123–171
- Cloud Service Fundamentals in Windows Azure*, 259
- cloud solutions, 289–354
  - ASP.NET Web API services, creating, 336–341
  - data access layer, creating, 312–321
  - delivery of, 3
  - deploying to SQL Database, 301–304, 312
  - importing into SQL Database, 297–300
  - iterative development cycle, 305
  - layered architecture, 289–292
  - sample database, building, 292–293
  - sample solution stack, 289–291
  - scaling, 2–3
  - SQL Server Database project, creating, 295–296
  - stored procedures, creating, 307–312
  - table columns, adding, 300–301
  - tables, creating, 305–307
  - target platform, setting, 296–297
  - website, creating, 321–335
  - Windows Phone application, creating, 341–354
- cloud storage
  - browsing, 78
  - Microsoft Azure Storage Account for, 35
- cloud-based Microsoft Azure SQL Database. *See* SQL Database
- CLR (Common Language Runtime), 60
- clustered indexes, 61
- code in the cloud, 2
- codebases, 57, 63–64
- collations, 31–34
- columns, table
  - adding, 300–301
  - check constraints on, 310
  - null values, 300–301
- command prompt, opening, 46, 83
- Common Language Runtime (CLR), 60
- compatibility issues, 88–94
- compliance, 98–99
- compression, 60
- compute resources, 218, 234
  - load distribution across, 251–259
  - partitioning, 250–259
  - reserving, 245
- conflict resolution in data synchronization procedures, 196–200, 214–215
- Connect To Server dialog, 41–42
  - advanced version, 110–111
- Connect to Server errors, 109–110
- connected development, 294
- connection pooling, 234
- connection strings
  - @server requirement, 58
  - entity, locating, 324
  - passwords in, 316
  - quick-start link to, 38–39
  - viewing, 38–39
- connections
  - dropped, 59
  - faults, recovering from, 234–236
  - limitations on, 58–59, 234
  - managing, 234–243

connections (*continued*)

- monitoring with DMVs and DMFs, 277–280
- multiple pending, 58
- opening late, closing early, 234
- pooling, 234
- prevention of, 102 *See also* firewall rules
- specifying database, 110–111
- throttling, 235
- Transient Fault Handling Application Block, 235–243
- transient faults, 234–243 viewing, 278, 280

console applications, creating, 283–287

constraints, disabling during import operations, 85

Content Manager role, 170

content negotiation, 230

controllers, MVC, 321

- action methods, 326–327
- extending, 327
- Home controller, 321, 325–326
- scaffolding, 325–326
- URLs, mapping to, 321

controllers, Web API, 223–228, 230–233, 336–339

copy database operation, 112–115

COUNT function, 141

CREATE DATABASE statement, 43, 57

CREATE DATABASE...AS COPY OF statement, 113–114

credentials, user, 99

CRUD actions, 336–337

custom authentication, 170

custom code in reports, 124

Custom Create database creation option, 31, 33–34

customers of cloud vendors, security responsibilities, 98–99

CustomerShard class, 254, 256

cyber attacks, preventing, 98

## D

DACPAC (Data-tier Application Component Package) files, 30, 70, 80

DAL. *See* data access layer (DAL)

data

- check constraints, 310
- in cloud, 2
- formatting in reports, 162–163
- geo-replication, 71
- incremental changes and updates, 63
- referential integrity, 19
- regulatory compliance, 99
- scripting, 69
- service layer over, 27
- summarizing at database and report levels, 157
- validation rules, 307–312

data access, 106–111, 170

- encapsulating in stored procedures, 243–244, 307–312
- with Entity Framework, 313
- over HTTP, 336
- performance, 217
- securing, 98

data access layer (DAL)

- ADO.NET, building with, 312–313
- class library projects, creating, 314–315
- creating, 290–291, 312–321
- referencing, 323–324
- repository pattern, 314

data aggregation with SQL Data Sync, 177

data compression, 60

data entry, 21–24

data importing *See* importing

data migration, 63–86, 88–94

data partitioning, 13, 58, 61, 250–259, 276

data redundancy, 60–61

data sources, report, 125, 135, 138

- creating, 139–140, 158–159

data stores, maintaining, 63–64

data synchronization *See* SQL Data Sync; sync groups; synchronizing data

data validation, 307–312

database access, 12, 35–38, 106–111, 170, 307–312

database administration, 30

- See also* SQL Database management portal

database collations, 31–34

database connections *See* connections

Database Copy feature, 59, 112–113

database dynamic management views (DMVs), 275–278

database objects

- developing, deploying, and managing, 70
- mapping to .NET objects, 217
- naming conventions, 60
- scripting, 68
- T-SQL scripts, generating for, 91–92, 94

database queries *See* queries

database-level permissions, 108–109

databases. *See also* on-premises SQL Server databases; SQL Database

- changes, discovering and replicating, 173
- changing between, 61
- configuring, 44
- connection strings *See* connection strings
- copying, 112–113
- creating, 13–15, 184
- creating in Microsoft Azure portal, 31–35
- creating in PowerShell, 48–49
- creating in SSMS, 43

## databases

### databases (*continued*)

- creating with Service Management API, 283–287
- current number, 265
- deleting, 40, 44, 50, 77
- edition, 33–34, 44, 55–56
- exporting, 115–117
- firewall rules *See* firewall rules
- fragmentation information, 276
- free space, 56, 270
- importing, 31, 34–35, 297–300
- incremental changes and updates, 63, 191
- index statistics, 276
- I/O, locking, latching, and access method activity, 276
- load distribution across, 251–259
- local, 64–66 *See also* on-premises SQL Server databases
- migrating to SQL Database, 63–94
- mirroring, 60–61
- monitoring, 263–280
- new query window, 43, 65, 106–107, 109
- offline management, 300
- operational metrics, 265–269
- page and row-count information, 276
- partitioning, 13, 250–259
- physical distance from applications, 243
- populating, 21–24
- querying, 24–26 *See also* queries
- reporting *See* reporting; reports
- saving data to, 22
- scripting, 68 *See also* Transact-SQL (T-SQL)
- service layer over, 307
- sharding, 13, 252–254
- size, 31, 33, 44, 55–58, 270, 276–278
- synchronizing, 173–174 *See also* SQL Data Sync; synchronizing data
- transactional consistency, 112
- updating, 194–195
- upgrading to SQL Database Premium, 249–250
- usage metrics, 265–269
- varbinary(max)* or *image* file storage, 54
- waits statistics, 276

*Dataset* objects and data adapters, 313

datasets, data synchronization, 188–191, 215

datasets, report, 135, 138, 140–143

Data-tier Application Component Package (DACPAC) files, 30, 70, 80

Data-Tier Applications (DACs), 70, 74–77, 80

data-transfer pricing, 51

*dbmanager* security role, 107

deleting

- data in Windows Phone 8 apps, 352
- databases, 40, 44, 50, 77, 113

- firewall rules, 104

- tables, 81

Developer Command Prompt, 281

disaster recovery, 60–61, 97, 112–121

DMFs *See* dynamic management functions (DMFs)

DMVs *See* dynamic management views (DMVs)

*DROP DATABASE* statement, 44, 113

*DROP TABLE* statement, 81

dropped connections, 59

dynamic management functions (DMFs), 275–280

dynamic management views (DMVs), 275–280

## E

EDM *See* Entity Data Model (EDM)

EDMX files, 223, 313

EF *See* Entity Framework (EF)

endpoints, VM, 131–132

entities, 318–320, 325

entity connection string, 324

Entity Data Model (EDM), 313–314

- class library projects, creating, 314–315

- conceptual schema, 313

- creating, 315–321

- data connection, 315

- data source, 316

- .edmx files, 313

- entity classes, 318

- mapping, 313, 320

- metadata, 313

- model contents, 315

- as model in MVC applications, 323–324

- navigation properties of entities, 318–319

- result bindings, 319

- storage schema, 313

- stored procedures, 317, 319–320

- tables, selecting, 317

- version setting, 317

Entity Data Model (EDM) designer, 223

Entity Framework (EF), 217

- Code First feature, 223–228, 313

- Connection Resiliency feature, 59

- conventions, 224

- data access layer, building, 312–321

- data context, 226–227

- database first approach, 313

- database initialization strategy, 224, 228

- database resource consumption, 234

- Entity Data Model, 313–314

- performance overhead, 217

- pluralization naming strategy, 224–225, 227

- POCO (Plain Old CLR Object) classes, 223

Entity Framework (*continued*)

- RESTful Web APIs, building, 218–228
- schemas, mapping between, 314
- sharding, 259
- Transient Fault Handling Application Block and, 239–243
- Visual Studio scaffolding feature and, 321
- Web API controllers, adding, 226–227
- Web APIs, creating, 225–228

entity models, 223–224, 230

error code 40544, 276

errors, 109–110, 235, 276, 329–330

event tables, 280

exception handling, 329–331

*ExecuteAction* method, 239–243

execution dynamic management views (DMVs), 278–280

execution performance, monitoring, 277–280

execution speed, optimizing, 217

Export Database feature, 115–117

Export Data-Tier Application wizard, 74–77

exporting, 35, 83–84

- Automated Export feature, 120–121
- BACPAC files, 115–117
- export files, 120–121

extended events, 60

extended stored procedures, 60

Extensible Application Markup Language (XAML)

- binding features, 345

## F

F5 key, 43

fan-out queries, 256, 259

file streaming, 60

firewall rules, 9, 12, 35–38, 48, 102–105

- See also* SQL Database Firewall
- for local network access, 35–36
- quick-start link, 36
- for remote IP address access, 35–36
- report server access, 130–132

foreign-key table relationships, 65, 306, 318–319

forms-based authentication, 170

FROM clause, 140–141

Full-Text Searching (FTS), 60

fully qualified names, 134

functional partitions, 250–251

## G

Generate And Publish Scripts wizard, 68–69

geographic region, 8–9

of applications, 179, 243

of cloud databases, 215

of Microsoft Azure Storage account, 71

pricing and, 51

synchronization and, 176–179

geo-replication, 52–53, 55, 71

GET requests, 326

GROUP BY clause, 140–141

GUI management tools, 261. *See also* Microsoft Azure management portal; SQL Database management portal; SQL Server Data Tools (SSDT); SQL Server Management Studio (SSMS)

## H

hackers, protecting against, 99

hardware, hypervisor virtualization technology on, 2–3

high availability, 112

horizontal partitioning, 251–259

HTTP data access over, 336

HTTP requests and responses, 223, 230, 283–287, 326, 351–352

hub databases, 174, 186

Huey, George, 86

hypervisor virtualization technology, 2–3

## I

IaaS (Infrastructure as a Service), 3–4

*image* file storage, 54

Import Database feature, 31, 77–80, 117–119

importing

- BACPAC files, 117–119
- databases, 34–35, 77–80
- nonclustered indexes, triggers, constraints, 85
- into SQL Database instance, 84–86
- with SQL Server Integration Services, 63

incremental changes and updates, 63

index partitioning, 61

index statistics, 276

Infrastructure as a Service (IaaS), 3–4

instance names of local SQL Server database, 64

instance objects, 70

Internet Explorer, 150

- See also* browsers

Internet hosting, cloud computing service level, 1

IP addresses

- allowed, 12, 35
- current client IP address, 104
- firewall rules for, 101–104
- learning remote addresses, 37

## JavaScript Object Notation (JSON)

### J

JavaScript Object Notation (JSON), 336, 338–339  
Json.NET, 343–344

### L

language settings, customizing, 31  
latency, 9, 215, 243–244  
least privilege principle, 108–109, 111  
legacy systems, data migration from, 63–94  
local machines, 2, 12  
    *See also* servers  
local networks, enabling database access, 35–36  
local SQL Server databases, 64–68, 83–84  
local sync agent, 204–211  
locks, monitoring, 280  
logical administration, 57  
login user names @*server* requirement, 58  
*loginmanager* security role, 107  
lookup data, client-side storage of, 244  
Lucene text search engine library, 60

### M

maintenance, 4, 63–64  
malicious attack prevention, 98  
management, 99, 261, 275–287  
    GUI tools for, 261. *See also* Microsoft Azure  
    management portal; SQL Database management  
    portal; SQL Server Data Tools (SSDT); SQL Server  
    Management Studio (SSMS)  
management certificates, 281–283  
manual syncs, 191–196  
mapping  
    objects, 217  
    between schemas, 314  
    stored procedures to entities, 319–320  
    URLs to controllers and actions, 321  
*master* database, 102, 106–110, 134–135, 280  
matrixes, report, 157, 160–161  
metadata, 244, 278  
Microsoft account, 5–7, 46–47  
Microsoft Azure, 1–4  
    ASP.NET Web Application projects, deploying to,  
    331–335  
    authentication for, 281, 283  
    compute resources, scaling up, 218  
    free trial subscription, 7–8  
    management certificates, 281–283  
    preview services, 120–121, 174, 246  
    pricing, 7, 9, 50–54, 79

    Queue storage, 244–245  
    scaling, 3. *See also* scalability  
    security, 99  
    SQL Reporting, 123–124  
    staying current with, 57–58  
    storage services for nonrelational data, 244–245  
    subscribing to, 7–8  
    Table storage, 244–245  
    volume purchase plan, 55  
Microsoft Azure Blob Storage, 35, 54, 60, 115, 244–245  
Microsoft Azure Cache, 244–245  
Microsoft Azure cloud, 2–4  
    *See also* cloud computing; cloud solutions  
Microsoft Azure Customer Advisory Team, 235, 259  
Microsoft Azure datacenters, 2, 51, 55  
Microsoft Azure Management Application Programming  
    Interface (API), 261  
Microsoft Azure management portal, 9, 31–40  
    BROWSE CLOUD STORAGE dialog box, 78  
    CONFIGURE link, 37  
    connection strings, obtaining, 38–39  
    copy database operation, monitoring, 114–115  
    CREATE A SQL DATABASE SERVER link, 9–10  
    CREATE SERVER dialog, 10–11  
    Custom Create database option, 33–34  
    database usage metrics, 265–266  
    databases, creating, 31–35  
    databases, deleting, 40, 77  
    databases, importing, 77–80  
    Export Database Settings dialog, 115–117  
    firewall rules, managing, 35–38, 102  
    IMPORT DATABASE dialog, 77–79  
    local agent service, configuring, 207–208  
    logging in, 9  
    main portal page, 9  
    MANAGE URL link, 15–16  
    management certificates, 281–283  
    Microsoft Azure Storage account, creating, 71–74  
    monitoring with, 264–269  
    publish profile for websites, 333–334  
    Quick Create database option, 31–33  
    quick-start links for databases, 36  
    server usage overview, 264–265  
    servers, creating, 9–11  
    SQL Data Sync configuration, 174  
    SQL Database management portal, accessing, 15,  
    192–193  
    SQL Database Monitor page, 267–269  
    SQL Database Premium, signing up for, 246–247  
    SQL Database Premium quota, requesting, 248  
    SQL Database Premium quota request status,  
    248–249

- Microsoft Azure management portal (*continued*)
  - SQL Database servers, list of, 102–103
  - SSRS, configuring in VM, 128–130
  - storage access keys, generating, 71
  - SYNC link, 188
  - VM endpoints, creating, 131–132
  - VMs, creating, 126–128
  - websites, creating, 331–334
- Microsoft Azure PowerShell cmdlets, installing, 44–45
- Microsoft Azure Service Dashboard, 269–270
- Microsoft Azure Service Management API, 71, 281–288
- Microsoft Azure services *See* services
- Microsoft Azure SQL Data Sync, 173–216
- Microsoft Azure SQL Database *See* SQL Database
- Microsoft Azure Storage, 35, 71–80
  - browsing, 78, 117–118
  - geo-replication, 52
  - pricing, 51–52
- Microsoft Azure Traffic Manager, 179–180
- Microsoft Azure Trust Center, 99
- Microsoft Silverlight, 15–16, 82, 183, 341, 343, 351
- Microsoft Visual Studio *See* Visual Studio
- Microsoft.AzureCat.Patterns.Data.SqlAzureDalSharded* library, 259
- migrating data, 63–94
- migration wizard, 86–94
- mirroring, 60–61
- models, entity, 223–224, 230
- Model-View-Controller (MVC) applications, 222–223, 323–331
- Model-View-Controller (MVC) framework, 291–292
- Model-View-Controller (MVC) websites, 321–335
- Model-View-ViewModel (MVVM) pattern, 344
- monitoring, 261, 263–280
  - database operations, 265–266
  - database size, 276–278
  - database usage, 265
  - dynamic management views and functions, 275–280
  - metrics, displaying, 267–269
  - with Microsoft Azure management portal, 264–269
  - Microsoft Azure Service Dashboard, 269–270
  - query performance, 271–275
  - sample database, creating, 261–263
  - server usage, 264–265
  - services health, 269–270
  - with SQL Database management portal, 270–275
- MSDN subscribers, Azure pricing for, 7
- multipart names for database objects, 60
- Multiple Active Result Sets (MARS), 58
- multitenant services, 98
- MVC. *See* Model-View-Controller (MVC) applications; Model-View-Controller (MVC) websites

## N

- naming conventions for database objects, 60
- navigation properties of entities, 318–319
- .NET Framework 3.5, 153
- network bandwidth fees, 9, 51–53, 79
- network latency, reducing, 243–244
- network round trips, minimizing, 243–244
- network traffic, reducing, 244
- new query window, 43, 65, 106–107, 109
- Newtonsoft.Json* assembly, 336, 343–344
  - importing namespace for, 350–351
- nonclustered indexes, 85
- nondeterministic behavior, 310
- nonrelational data, 244–245
- NuGet Package Manager, 235–236
  - Json.NET, downloading and adding, 344
- numeric data, formatting in reports, 162–163

## O

- Object Explorer, 40, 42
- object relational mapping (ORM), 217, 312
  - See also* Entity Framework (EF)
- objects
  - developing, deploying, and managing, 70
  - mapping to .NET objects, 217
  - naming conventions, 60
  - packaging as DACPAC files, 70
  - scripting, 68
  - T-SQL scripts, generating for, 91–92, 94
- offline projects, 294
- online transactional processing (OLTP), 178–179
- on-premises SQL Server databases, 64–66. *See also*
  - databases
    - abstraction levels, 3
    - BACPAC files, creating, 34–35
    - instances, 1
    - latency, 243
    - migrating to SQL Database, 63–94
    - mirroring, 60–61
    - nonrelational and binary data in, 244
    - size, calculating, 51
    - sync agent, registering with, 209–211
    - sync group, adding to, 211–214
    - synchronizing, 174–176, 202–215 *See also* synchronizing data
- on-premises SQL Server servers, 2
  - See also* servers
- ORM (object relational mapping), 217, 312

## PaaS (Platform as a Service)

### P

- PaaS (Platform as a Service), 4, 218
- partitioning data, 58, 61, 250–259
  - page and row-count information, 276
  - sharding, 13, 251–259
- password policy, 107
- passwords in connection strings, 316
- performance
  - in the cloud, 218
  - of cloud reporting, 124
  - of data access, 217
  - database connections overhead, 234
  - latency, reducing, 243–244
  - optimizing and tuning, 217, 244–245
  - of queries, 245, 271–275
- permissions, 108–111, 170
- physical administration, 57
- physical data centers, 98, 99
- physical storage with PaaS, 4
- Plain Old CLR Object (POCO) classes, 223–224, 230
- Platform as a Service (PaaS), 4, 218
- port 80, 130–132
- port 1433, 58
- port allocation, 58
- POST requests, 326, 352
- PowerShell, 44–50
  - administrator credentials, 48
    - Collation* switch, 49
  - configuring for Microsoft Azure, 46–47
  - context associated with server and credentials, 48
  - database collation, changing, 49
  - database edition, changing, 49
  - database size, changing, 49
  - databases, creating, 48–49
  - databases, deleting, 50
  - databases, viewing, 49
    - Edition* switch, 49
  - firewall rules, creating, 48
    - Force* switch, 50
  - management, automating, 281
  - master databases, 49
    - MaxSizeGb* switch, 49
  - server access, 47–48
  - servers, creating, 47–48
- PowerShell Integrated Scripting Environment (ISE), 46
- PowerShell Microsoft Azure cmdlets, 44–45, 48, 49
- Premium edition of SQL Database, 58, 245–250
- preview release software, 120–121, 174, 246
- pricing for Microsoft Azure
  - bandwidth, 9, 51–53, 79
  - Blob Storage, 54

- data transfer, 51
- file storage, 54
- geographic location and bandwidth pricing, 51
- online pricing calculator, 50
- pricing structures, 7
- storage, 50–51
- storage space, 51–52
- support plans, 53–54

- primary keys, 67
- primary transactional database, replicating, 178–179
- privacy, 98–99
- public cloud platforms *See* Microsoft Azure
- public cloud vendors
  - auditing services, 99
  - compliance certifications, 99
  - hackers, protecting against, 99
  - malicious attack prevention, 98
  - multitenancy management, 98
  - security responsibilities, 98–99
- publish process, 301–303, 312
- .publishsettings* file, 47
- PUT requests, 352

### Q

- queries, 24–26
  - editing, 273
  - encapsulating, 27, 243–244
  - error code 40544, 276
  - execution plans, 273–275
  - fan-out, 256, 259
  - memory grants, 278
  - optimizing, 245
  - performance, 271–275
  - query plan details, 272–273
  - query plans, cached, 278
  - query plans, showing, 278
  - running, 193–194
  - in stored procedures, 141, 243–244
- query code, 140–141, 157
- query filters, 25–26
- Queue storage, 244–245
- Quick Create database creation option, 31–33
- quick-start links, 36, 38–39

### R

- RDL authoring tools, 125
- .rdl* files, 169
- .rdp* files, 128
- read-only permissions, 108–111

- records, aging policies, 56
  - Red-Gate Data Compare, 63
  - Red-Gate SQL Compare, 63
  - redundancy, data, 60–61
  - reference databases, 174, 184, 186–187
  - referential integrity of data, 19
  - regulatory requirements, complying with, 99
  - relational data, storage in SQL Database, 244–245
  - relational databases, 250
    - See also databases; SQL Database
  - reliability, 217, 234–244
  - Remote Desktop sessions, 128
  - remote IP addresses, enabling database access, 37–38
  - replication, 61
  - Report Builder, 123, 125
    - browser, running report from, 149–150, 168–169
    - data sources for reports, 138–140
    - datasets for reports, 138, 140–143
    - date and time of report execution, 138
    - Getting Started dialog, 137
    - images, adding to reports, 139
    - installing, 135–137
    - launching, 137
    - layout of report, 143–145
    - matrix control, 138–139
    - previewing reports, 135, 145–147
    - Report Data pane, 138–139
    - report parameters, defining, 139
    - reports, creating, 137–150
    - reports, saving and running locally, 145–147
    - Row Groups and Column Groups panes, 139
    - SSRS, deploying reports to, 147–149
    - table control, 138–139
    - user interface, 138
  - Report Definition Language (RDL), 123
  - Report Designer, formatting data, 162–163
  - Report Manager, 130, 149, 170
  - report server, 124, 130–132
  - Report Server Database Configuration Wizard, 129–130
  - Report Server Project Wizard, 158–162
  - Report Server projects, 150–169
  - reporting, 123–171
    - Report Builder, 135–150
    - requests, performance impact of, 179
    - sample database, creating, 132–135
    - security of reports, 170
    - service URL, 149
    - virtual directory, creating, 129
    - Visual Studio Report Server projects, 150–169
    - VM, creating, 125–132
    - VM, shutting down, 171
  - Reporting Services See SQL Server Reporting Services (SSRS)
  - reports
    - bar charts, 164–166
    - creating with Report Builder, 137–150
    - creating with Visual Studio, 156–162
    - data source, 125, 135, 138–140, 158–159
    - dataset, 125, 135, 138, 140–143
    - date and time of execution, 138
    - deploying to SSRS, 135, 147–149, 166–168
    - drill-down capabilities, 163
    - executing, 146
    - formatting, 162–163
    - layout, 125, 135, 143–145
    - matrixes in, 157, 160
    - previewing, 145–147, 163
    - queries, defining, 159–160
    - RDL format, 125
    - running from browser, 149–150, 168–169
    - running locally, 145–146
    - saving, 145
    - security, 170
    - summarizing data in, 157
    - title, 143
    - user role assignment, 170
  - repository pattern, 314
  - requests. See HTTP requests and responses
    - monitoring with DMVs and DMFs, 277–280
  - Resource Governor, 61
  - REST services, 336
  - RESTful Web APIs, 218–228
  - restore database operations, 117–119
  - RetryPolicy* class *ExecuteAction* method, 239–243
  - round trips, minimizing, 243–244
  - routing, 179
    - in MVC applications, 321
- ## S
- SaaS (Software as a Service), 4, 289
  - scalability, 2–4, 218, 245–250
    - functional partitions, 250–251
    - OLTP activity, separating from reporting activity, 184
    - partitioning for, 58
    - sharding, 251–259
    - synchronizing between multiple locations, 178–179
  - SCC (source code control), 294–295
  - schemas, 156, 313
    - changing before migration, 80
    - compare operations, 301–303
    - deploying to SQL Database, 91, 94
    - mapping, 313–314

## schemas

- schemas (*continued*)
  - migrating, 77–83
  - referencing objects in, 60
  - scripting, 69
  - storage, 313
- searching, 60
- security, 97–121, 170, 216
  - firewall, 101–105 *See also* SQL Database Firewall
  - users, authenticating and authorizing, 99, 105–111
- server certificates, 61
- server logins, 105–107
- server-level permissions, 107–108
- server-level principals, 106–107
- server-level security roles, 107–108
- servers, 2, 8–12, 47–49, 183
  - accessibility to cloud services, 10
  - authentication method, 41
  - firewall settings, 12, 35–38, 102, 103 *See also* firewall rules; SQL Database Firewall
  - fully qualified names, 41
  - geographic region, 8–9
  - health, monitoring, 263–280
  - instance names, 64
  - physical access, securing, 98
  - PowerShell access, 47–48
  - usage, viewing, 264–265
- Service Broker, 61
- service levels, 3
- Service Management API, 281–288
- service outages, 269
- services, 2–4
  - Agent service, 59
  - Allowed Services list, 105
  - asynchronous service calls, 351
  - auditing, 99
  - Browser Service, 59
  - Cache service, 244–245
  - client agent service, 204
  - data center regions, 243
  - geographic region, 8–9
  - health, monitoring, 261, 269–270
  - local agent service, 207–208
  - Microsoft Azure services, 105, 120–121, 174, 244–246
  - multitenant, 98
  - outages, 269
  - reporting *See* SQL Server Reporting Services (SSRS)
  - REST services, 336
  - Service Broker, 61
  - Service Management API, 281–288
  - SQL Server Integration Services, 63
  - Web API services, 336–341
- sessions, monitoring with DMVs and DMFs, 277–280
- Shard* class, 254–256
- sharding, 13, 251–259
- ShardRoot* class, 254
- Software as a Service (SaaS), 4, 289
- source code control (SCC), 294–295
- spoke databases, 174
- SQL Data Sync, 61, 173–216
  - automated sync schedules, 200–202
  - best practices, 215–216
  - clients, 174
  - conflict resolution, 196–200
  - direction settings, 187
  - exporting data from SQL Server to SQL Database, 175
  - importing data from SQL Database to SQL Server, 175–176
  - incremental change tracking, 191
  - manual syncs, 191–196
  - multiple cloud databases, synchronizing, 178–180
  - multiple locations, sharing data between, 176–177
  - one-way and two-way synchronization, 174
  - performance, 215
  - reference databases, 174
  - sample database, creating, 180–182
  - security measures, 216
  - SQL Server on-premises database, creating, 202–203
  - sync agents, 204–215
  - sync groups, 174, 182–188
  - sync rules, 188–191
  - Traffic Manager and, 179–180
- SQL Database, 1
  - applications, co-locating with, 55
  - Automated Export feature, 120–121
  - backing up, 112–121
  - Business edition, 55, 58
  - communication port, 58
  - comparison to SQL Server, 57–62
  - connecting from SSMS, 41–42
  - connection limitations, 58–59
  - constraints of, 57
  - cost-related items, 50–56
  - database connections, managing, 234–243
  - editions, switching between, 13
  - exporting data *See* exporting
  - firewall rules *See* firewall rules
  - Generally Available features, 120
  - high-availability features, 112
  - importing, 297–300 *See also* importing
  - Microsoft account, 5–7
  - Microsoft Azure subscription, 7–8
  - migrating databases to, 63–94

- SQL Database (*continued*)
  - monitoring health and usage, 263–280
  - monthly charges, 56
  - networking requirements, 101
  - online pricing calculator, 50
  - as PaaS solution, 4
  - performance, 244–245 *See also* performance
  - port number, 42
  - Premium edition, 55, 58, 249–250
  - Preview features, 120–121
  - pricing, 50–56, 215
  - SaaS solutions, 4
  - scaling up, 245–250
  - scripts compatible with, 69
  - securing, 100–111 *See also* security
  - signing up for, 5–8
  - size limitations, 57–58
  - sizes, switching between, 13
  - SQL Service Database projects, deploying to, 301–304, 312
  - synchronization *See* SQL Data Sync; synchronizing data
  - unsupported features, 59–61
  - upgrading to SQL Database Premium, 249–250
  - Web edition, 31, 55, 58
- SQL Database APIs, 281
- SQL Database Firewall, 101–105, 184
  - See also* firewall rules
- SQL Database instances, 13–30, 100–101
  - BACPAC files, importing, 77–80
  - importing data into, 84–86
  - naming, 79
  - server for hosting, 79
- SQL Database management portal, 15–17
  - accessing from Microsoft Azure portal, 192–193
  - ad-hoc query window, 24–25
  - connections and sessions, monitoring, 279–280
  - data entry, 21–24
  - database administration features, 30
  - database size, checking, 277–278
  - database summary page, 270–271
  - dropping and re-creating databases, 82
  - foreign-key designer, 20
  - logging in, 16, 82
  - monitoring performance and usage with, 270–275
  - Query Performance page, 271–272
  - Query Plan Details area, 272–273
  - stored procedures, creating, 27–29
  - Summary view, 17
  - table designer, 18–20
  - T-SQL scripts, executing, 70
  - views, creating, 27
- SQL Database Management REST API, 105
- SQL Database Migration Wizard, 87–94
  - See also* migrating data
- SQL Database servers
  - See also* servers
  - AdventureWorks database, deploying to, 153
  - Allow Windows Azure Services To Access The Server option, 11, 105
  - connecting to using SQL Database Migration Wizard, 91–92
  - databases, deleting from, 77
  - master database, 106
  - permissions, 107–109
  - SQL Database Premium quota, adding, 248
- SQL Database Service Dashboard, 270
- SQL Data-Tier Applications, 70, 74–77, 80
- SQL Server, 1
  - See also* on-premises SQL Server databases
  - exporting data, 175
  - Express edition, 58, 64
  - Express With Advanced Services edition, 126
  - hosting in Azure virtual machines, 3, 60–61
  - importing data, 175–176
  - vs. SQL Database, 3–4, 57–62
  - synchronization service, 174 *See also* SQL Data Sync
- SQL Server Authentication, 41, 105
- SQL Server Bulk Copy (bcp), 80–86, 153
- SQL Server Data Tools For Visual Studio 2012, 158
- SQL Server Data Tools (SSDT), 41, 151
  - BACPAC file import restrictions, 80
  - bi-directional editing, 300–301
  - Business Intelligence add-in for Visual Studio, 123
  - Business Intelligence project templates, installing, 151, 154–156
  - connected development, 294
  - DACPAC files, defining and deploying, 30
  - DACPAC files, importing into, 80
  - database projects, testing, 295
  - databases, creating, 180–182, 292–293
  - disconnected development, 294
  - table designer, 300–301
  - T-SQL scripts, building and running, 64
- SQL Server Database projects
  - creating, 295–296
  - database definition, 295
  - dbo folder, 300
  - deploying, 301–305, 312
  - importing from SQL Database, 297–300
  - offline changes, 305
  - target platform, setting, 296–297
- SQL Server instances, 65–66, 202–203
- SQL Server Integration Services (SSIS), 63

## SQL Server Management Objects (SMO)

- SQL Server Management Objects (SMO), 40–41
  - SQL Server Management Studio (SSMS), 40–44
    - advanced Connect to Server dialog, 110–111
    - Connect To Server dialog, 41–42, 64–65
    - Database Engine, connecting to, 110
    - databases, configuring, 44
    - databases, creating, 43
    - databases, deleting, 44
    - downloading, 40
    - exporting Data-Tier Applications, 74–77
    - fully qualified server names, 38, 41
    - Generate And Publish Scripts wizard, 68
    - graphical designers and dialogs support, 40–41
    - launching, 41, 65, 134
    - new query window, 43, 65
    - Object Explorer, 40
    - server logins, creating, 106–107
    - sharded databases, creating, 252–254
    - SQL Database, connecting to, 41–42
    - T-SQL scripts, generating automatically, 68–70
    - T-SQL scripts, running, 65
    - T-SQL scripts, viewing, 69–70
  - SQL Server Object Explorer, 304
  - SQL Server Reporting Services (SSRS), 3, 123–124
    - authentication of users and applications, 170
    - Configuration Manager, 128–129
    - configuring in virtual machine, 128–130
    - database, creating, 129–130
    - deploying reports to, 147–149
    - formatting capabilities, 150
    - free version, 126
    - Report Manager, running reports from, 149–150
    - Report Manager virtual directory, 129–130
    - security in, 170
    - Visual Studio as authoring tool, 151
    - VM SSRS service URL, 166–167
    - VM to host, creating, 125–132
  - .sqlexpress* instance name, 64
  - SSDT *See* SQL Server Data Tools (SSDT)
  - SSMS *See* SQL Server Management Studio (SSMS)
  - SSRS *See* SQL Server Reporting Services (SSRS)
  - SSRS-in-a-VM solutions, 124
  - statements result set metadata, 278
  - storage, 50–51, 244
    - See also* Microsoft Azure Storage
  - stored procedures, 27–29, 60, 243–244, 307–312
    - mapping to entities, 319–320
    - performance statistics, 278
    - queries in, 141
    - system, 61
    - testing, 29–30
  - strong password policy, 107
  - sync agents, 204–215
  - sync groups, 174, 182–189, 192
    - conflict resolution behavior, 174, 196–200, 214–215
    - on-premises SQL Server databases, adding, 211–214
    - ready status, 187, 191
    - scheduling syncs, 200–202, 215
    - sync loops, 215
    - sync rules, 188–191
  - sync loops, 215
  - sync rules, 188–191
  - synchronizing data, 215
    - automated schedules, 200–202
    - conflict resolution, 196–200
    - costs, 179
    - dataset, defining, 188–191
    - direction settings, 187
    - exporting data from SQL Server to SQL Database, 175
    - importing data from SQL Database to SQL Server, 175–176
    - manual, 191–196
    - multiple locations, sharing data between, 176–177
    - one-way and two-way, 174–177
    - with SQL Data Sync, 61, 173 *See also* SQL Data Sync
    - sync agents, creating, 204–215
    - sync groups, creating, 182–188
    - sync rules, creating, 188–191
    - testing, 195–196
  - sys.database\_connection\_stats* view, 80
  - sys.dm\_database\_copies* view, 114
  - sys.event\_log* view, 280
  - system stored procedures, 61
- ## T
- table partitioning *See* partitioning data
  - Table storage, 244–245
  - tables, 17–19, 30, 305–307
    - access control, 307–312
    - aggregating rows, 141
    - aliasing, 140–141
    - change tracking, 191
    - check constraints on, 310
    - clustered indexes, 61
    - columns, adding, 18, 300–301
    - data, inserting, 21–24, 67–68
    - deleting, 81–82
    - foreign keys, 19, 22–23, 306
    - ID* primary key column, 18
    - Is Identity? option, 18, 21
    - joining, 24–26, 140–141
    - migrating data into and out of, 83–86

- tables (*continued*)
    - NULL* values, 22
    - partitioning, 251–259
    - populating using T-SQL script, 67–68
    - primary keys, 21–22
    - relationships between, defining, 19–20
    - scripting, 68
    - service layer over, 307
    - updating, 304
    - validation rules, 307–312
  - table-valued parameters, batching, 244
  - tabular data stream (TDS), 57
  - Task Parallel Library, 256, 259
  - TCP port 80, 130–132
  - TCP port 1433, 58
  - tenants of cloud vendors, privacy from, 98
  - text search engine library, 60
  - throttling, 235
  - Traffic Manager, 179–180
  - transaction dynamic management views (DMVs), 280
  - transactional consistency, 59, 112, 120–121
  - Transact-SQL (T-SQL)
    - change scripts, 301, 303
    - FROM* clause, 24, 26
    - CREATE DATABASE* statement, 43
    - data migration with, 64–70
    - database engine type property, 69
    - for database queries, 24
    - databases, creating, 180–181, 219–221, 292
    - DROP TABLE* statement, 81
    - editing, 300–301
    - EXEC* statements, 29
    - executing, 43
    - foreign-key table relationships, 65
    - generating automatically, 68–70
    - IDENTITY\_INSERT* setting, 67
    - left outer joins, 24, 26, 28
    - local SQL Server database, creating, 65
    - populating local databases, 67–68
    - query filters, 25–26
    - Remove-AzureSqlDatabase* cmdlet, 50
    - reporting, preparing databases for, 133
    - role assignment management, 107–108
    - server logins, creating, 107
    - sharded databases, 252
    - SQL Database support, 57
    - SQL Database use of, 60
    - types of data property, 69
    - user management, 107–108
    - WHERE* clause, 25–26
  - transient connection faults, 234–235, 237–243
  - transient error conditions, 239–243
  - Transient Fault Handling Application Block, 59, 235–243
  - Transparent Data Encryption (TDE), 61
  - triggers, 85, 278, 310
  - tuning queries, 245
  - two-part names for database objects, 60
  - two-way synchronization, 176–177
- ## U
- unstructured data, 60, 244–245
  - UPDATE statement, 194–195
  - URLs
    - managing, 15–16
    - mapping to controllers and actions, 321
  - usage, monitoring, 263–280
  - USE* statement, 61, 135
  - users
    - authenticating and authorizing, 105–111
    - Browser role, 170
    - Content Manager role, 170
    - creating, 108
    - routing to closest data center, 179–180
    - security credentials, 99
- ## V
- validation, data, 307–312
  - varbinary(max)* file storage, 54
  - vendors, security responsibilities, 98–99
  - views, database, 27
  - views, dynamic management (DMVs), 275–280
  - views (in MVC applications), 321, 327–328
  - virtual machines (VMs), 60–61, 126–132
    - globally unique name, 147
    - with IaaS, 3
    - saving reports to, 148
    - shutting down, 128, 171
    - SSRS service URL, 166–167
  - virtual private networks (VPNs), 60–61
  - virtualization layer, 2–3
  - Visual Studio
    - Add Scaffold dialog, 325
    - ASP.NET Web Application project type, 222–223
    - bi-directional editing, 300–301
    - Business Intelligence Developer Studio version, 151
    - console applications, creating, 286–287
    - Data Tools Operations window, 303
    - database definitions, maintaining in, 295
    - databases, creating, 180–182
    - Dataset* designer, 313
    - EDM designer, 223, 313, 318

## Visual Studio

### Visual Studio (*continued*)

- Entity Data Model, creating, 315–321
  - importing databases into database projects, 297–300
  - Mapping Details window, 320
  - NuGet package manager, 235–236
  - publishing databases, 301–303
  - references, adding, 323
  - scaffolding feature, 321, 325, 336–337
  - Solution Explorer, viewing, 222
  - solutions, building, 226, 228
  - solutions, creating, 221–222, 294–295
  - source code control, 294–295
  - SQL Server Database projects, creating, 295–296
  - SQL Server Object Explorer, 304
  - SSDT Business Intelligence add-in, 123
  - SSDT Business Intelligence project templates, installing, 151, 154–156
  - stored procedures, creating, 310–312
  - tables, creating, 30
  - target platform for database projects, setting, 296–297
  - Views folder, 327
  - Web API projects, creating, 222–223
  - Web API projects, deploying, 334–335, 340–341
  - Windows Phone apps, creating, 343
  - Windows Phone apps, testing, 353–354
  - Windows Phone SDK, 341–342
- Visual Studio Report Server projects, 150–169
- See also* reporting; reports
  - AdventureWorks database, installing, 151–154
  - bar charts, adding to reports, 164–166
  - creating reports, 156–162
  - data sources, creating, 158–159
  - deploying, 150–169
  - designing, 150–169
  - formatting reports, 162–163
  - previewing reports, 163
  - project properties, 166–167
  - queries, defining, 159–160
  - Report Manager home page, 168
  - Report Server Project Wizard, 158–162
  - running reports from browser, 168–169
  - with SSDT Business Intelligence add-in, 125
  - SSDT Business Intelligence project templates, installing, 151, 154–156
  - VM SSRS service URL, 166–167

## W

- waits statistics, 276
- Web API controllers, 223–230, 336–339
- Web API services, 336–341
- Web APIs
  - connection strings, setting, 227
  - content negotiation, 230
  - controllers, adding, 226–227, 230–233
  - creating, 222–223, 225–228, 321–323
  - data access layer, referencing, 323–324
  - deploying from Visual Studio, 334–335
  - deploying to Microsoft Azure, 331–335
  - entity connection string, 324
  - exception handling, 329–331
  - model class, adding, 225
  - referencing data access layer project, 323
  - referencing MVC and Web API assemblies, 322–323
  - RESTful Web APIs, building, 218–228
  - sample database, creating, 219–221
  - scaffolds, 226
  - services, defining in, 336
  - sharding with, 254–259
  - testing, 228–230, 233, 328–331
  - Transient Fault Handling Application Block, adding, 235–236
  - user interface, creating, 324–328
- Web (Basic) edition of SQL Database, 31, 55, 58
- web services, creating, 336–341
- Web.config* file, copying into entity connection string, 324
- websites, 4, 321–335
- whatismyipaddress.com*, 37
- Windows authentication, 58, 105
- Windows Azure, 2
- Windows Firewall With Advanced Security, 130
- Windows Integrated authentication, 170
- Windows Live ID, 5
- Windows Phone 8 apps, 341–354
- Windows Phone Databound App template, 344
- Windows Phone SDK, installing, 341–342
- Windows PowerShell *See* PowerShell

## X

- X.509 v3 certificates, creating, 281
- XAML binding features, 345
- XML, SQL Database support, 61
- xml* data type, 61