Developing Solutions for Microsoft Azure

THIRD EDITION

Exam Ref AZ-204

Santiago Fernández Muñoz

FREE SAMPLE CHAPTER |

Microsoft

Exam Ref AZ-204 Developing Solutions for Microsoft Azure, Third Edition

Santiago Fernández Muñoz

Exam Ref AZ-204 Developing Solutions for Microsoft Azure, Third Edition

Published with the authorization of Microsoft Corporation by:
Pearson Education, Inc.

Hoboken, New Jersey

ISBN-13: 9780138312138
ISBN-10: 0138312133

Library of Congress Control Number: 2024938412

$PrintCode

CREDITS

EDITOR-IN-CHIEF
Brett Bartow

EXECUTIVE EDITOR
Loretta Yates

ASSOCIATE EDITOR
Shourav Bose

DEVELOPMENT EDITOR
Songlin Qiu

MANAGING EDITOR
Sandra Schroeder

SENIOR PROJECT EDITOR
Tracey Croom

COPY EDITOR
Dan Foster

INDEXER
Timothy Wright

PROOFREADER
Barbara Mack

TECHNICAL EDITOR
Doug Holland

EDITORIAL ASSISTANT
Cindy Teeters

COVER DESIGNER
Twist Creative, Seattle

COMPOSITOR
codeMantra

GRAPHICS
codeMantra

Contents at a glance

Contents

About the Author

SANTIAGO FERNÁNDEZ MUÑOZ is a Senior Solution Architect involved in developing international projects. He started his career as a Systems Engineer and jumped into the professional development world, attracted by the possibilities and complexities of distributed computing. He runs his own company, mainly focused on providing services to the industrial environment, specializing in photovoltaic environments and industrial cybersecurity.

Introduction

Most books take a very low-level approach, teaching you how to use individual classes and accomplish fine-grained tasks. Through this book, we review the main technologies that Microsoft offers for deploying different kinds of solutions into Azure. From the most classic and conservative approaches using Azure Virtual Machines to the latest technologies implementing event-based or message-based patterns with Azure Event Grid or Azure Service Bus, this book reviews the basics for developing most types of solutions using Azure services. The book also provides code examples that illustrate how to implement most of the concepts covered. This book should be used as an introduction to implementing more complex solutions. Although the book covers some basic concepts, you should have basic programming experience using ASP.NET, .NET Framework, or .NET Core, as well as using Git.

This book covers every major topic area found on the exam, but it does not cover every exam question. Only the Microsoft exam team has access to the exam questions, and Microsoft regularly adds new questions to the exam, making it impossible to cover specific questions. You should consider this book a supplement to your relevant real-world experience and other study materials. If you encounter a topic in this book that you do not feel completely comfortable with, use the "Need more review?" links you'll find in the text to find more information and take the time to research and study the topic. Great information is available on MSDN, TechNet, and in blogs and forums.

Organization of this book

This book is organized by the "Skills measured" list published for the exam. The "Skills measured" list is available for each exam on the Microsoft Learn website: *learn.microsoft.com/en-us/credentials/certifications/resources/study-guides/az-204*. Each chapter in this book corresponds to a major topic area in the list, and the technical tasks in each topic area determine a chapter's organization. If an exam covers six major topic areas, for example, the book will contain six chapters.

Preparing for the exam

Microsoft certification exams are a great way to build your resume and let the world know about your level of expertise. Certification exams validate your on-the-job experience and product knowledge. Although there is no substitute for on-the-job experience, preparation through study and hands-on practice can help you prepare for the exam. This book is not designed to teach you new skills.

We recommend that you augment your exam preparation plan by using a combination of available study materials and courses. For example, you might use the *Exam Ref* and another study guide for your at-home preparation and take a Microsoft Official Curriculum course for the classroom experience. Choose the combination that you think works best for you. Learn more about available classroom training, online courses, and live events at *microsoft.com/learn*.

Note that this *Exam Ref* is based on publicly available information about the exam and the author's experience. To safeguard the integrity of the exam, authors do not have access to the live exam.

Microsoft certifications

Microsoft certifications distinguish you by proving your command of a broad set of skills and experience with current Microsoft products and technologies. The exams and corresponding certifications are developed to validate your mastery of critical competencies as you design and develop, or implement and support, solutions with Microsoft products and technologies both on-premises and in the cloud. Certification brings a variety of benefits to the individual and to employers and organizations.

NEED MORE REVIEW? **ALL MICROSOFT CERTIFICATIONS**

For information about Microsoft certifications, including a full list of available certifications, go to *www.microsoft.com/learn*.

Access the Exam Updates chapter and online references

The final chapter of this book, "AZ-204 developing solutions for Microsoft Azure exam updates," will be used to provide information about new content per new exam topics, content that has been removed from the exam objectives, and revised mapping of exam objectives to chapter content. The chapter will be made available from the link below as exam updates are released.

Throughout this book are addresses to webpages that the author has recommended you visit for more information. Some of these links can be very long and painstaking to type, so we've shortened them for you to make them easier to visit. We've also compiled them into a single list that readers of the print edition can refer to while they read.

The URLs are organized by chapter and heading. Every time you come across a URL in the book, find the hyperlink in the list to go directly to the webpage.

Download the Exam Updates chapter and the URL list at *MicrosoftPressStore.com/ERAZ2043e/downloads*.

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:

MicrosoftPressStore.com/ERAZ2043e/errata

If you discover an error that is not already listed, please submit it to us at the same page.

For additional book support and information, please visit *MicrosoftPressStore.com/Support*.

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 *support.microsoft.com*.

Stay in touch

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

CHAPTER 2

Develop for Azure storage

All applications work with information or data. Applications create, transform, model, or operate with that information. Regardless of the type or volume of data that your application uses, eventually you need to save it persistently so that it can be used later.

Storing data is not a simple task, and designing storage systems for that purpose is even more complicated. Perhaps your application must deal with terabytes or even petabytes of information, or you might work with an application that needs to be accessed from different countries, and you need to minimize the time required to access it. Cost efficiency is also a requirement in any project. In general, many requirements make designing and maintaining storage systems difficult.

Microsoft Azure offers different storage solutions in the cloud to satisfy your application storage requirements. Azure offers solutions for making your storage cost-effective and for minimizing latency.

Skills covered in this chapter:

- Skill 2.1: Develop solutions that use Cosmos DB storage
- Skill 2.2: Develop solutions that use Blob Storage

Skill 2.1: Develop solutions that use Cosmos DB storage

Cosmos DB is a premium storage service that Azure provides for satisfying your need for a globally distributed, low-latency, highly responsive, and always-online database service. Cosmos DB has been designed with scalability and throughput in mind. One of the most significant differences between Cosmos DB and other storage services offered by Azure is how easily you can scale your Cosmos DB solution across the globe by merely clicking a button and adding a new region to your database.

Another essential feature that you should consider when evaluating this type of storage service is how you can access this service from your code and how hard it would be to migrate your existing code to a Cosmos DB–based storage solution. The good news is that Cosmos DB offers different APIs for accessing the service. The best API for you depends on the type of data that you want to store in your Cosmos DB database. You store your data using Key-Value, Column-Family, Documents, or Graph approaches. Each of the different

APIs that Cosmos DB offers allows you to store your data with different schemas. Currently, you can access Cosmos DB using SQL, Cassandra, Table, Gremlin, and MongoDB APIs.

This skill covers how to

- Perform operations on containers and items by using the SDK
- Set the appropriate consistency level for operations
- Implement change feed notifications

Perform operations on containers and items by using the SDK

When working with Cosmos DB, you have several layers in the hierarchy of entities managed by the Cosmos DB account. The first layer is the Azure Cosmos DB account, where you choose the API you want to use to access your data. Remember that this API has implications for how the data is stored in the databases.

The second layer in the hierarchy is the database. You can create as many databases as you need in your Cosmos DB account. Databases are a way of grouping containers; you can think of databases like namespaces. At this level, you can configure the throughput associated with the containers included in the database.

When planning how to store the information that your application needs to work, you must consider the structure you need to use for storing that information. You may find that some parts of your application need to store information using a key-value structure. In contrast, others may need a more flexible, schema-less structure in which you save the information into documents. One fundamental characteristic of your application might be that you need to store the relationship between entities and use a graph structure for storing your data.

Cosmos DB offers a variety of APIs for storing and accessing your data, depending on the requirements of your application:

- **NoSQL** This is the core and default API for accessing your data in your Cosmos DB account. This core API allows you to query JSON objects using SQL syntax, which means you don't need to learn another query language. Under the hood, the SQL API uses the JavaScript programming model for expression evaluation, function invocations, and typing systems. You use this API when you need to use a data structure based on documents.
- **Table** You can think of the Table API as the evolution of the Azure Table Storage service. This API benefits from the high-performance, low-latency, and high-scalability features of Cosmos DB. You can migrate from your current Azure Table Storage service with no code modification in your application. Another critical difference between Table API for Cosmos DB and Azure Table Storage is that you can define your own indexes in your tables. In the same way you do with the Table Storage service, the Table API allows you to store information in your Cosmos DB account using a data structure based on documents.

- **Cassandra** Cosmos DB implements the wire protocol for the Apache Cassandra database into the options for storing and accessing data in the Cosmos DB database. This allows you to forget about operations and performance-management tasks related to managing Cassandra databases. In most situations, you can migrate your application from your current Cassandra database to Cosmos DB using the Cassandra API by merely changing the connection string. Azure Cosmos DB Cassandra API is compatible with the CQLv4 wire protocol. Cassandra is a column-based database that stores information using a key-value approach.
- **MongoDB** You can access your Cosmos DB account by using the MongoDB API. This NoSQL database allows you to store the information for your application in a document-based structure. Cosmos DB implements the wire protocol compatible with MongoDB 3.2. This means that any MongoDB 3.2 client driver that implements and understands this protocol definition can connect seamlessly with your Cosmos DB database using the MongoDB API.
- **PostgreSQL** This service is built on top of native PostgreSQL, which means you can use your code directly with Azure Cosmos DB for PostgreSQL without any substantial modification. This is a managed service, so Microsoft takes care of all the details regarding performance, availability, geo-replication, and all the features that the Cosmos DB service offers.
- **Gremlin** Based on the Apache TinkerPop graph transversal language or Gremlin, this API allows you to store information in Cosmos DB using a graph structure. This means that instead of storing only entities, you store:
  - **Vertices** You can think of a vertex as an entity in other information structures. In a typical graph structure, a vertex could be a person, a device, or an event.
  - **Edges** These are the relationships between vertices. A person can know another person, a person might own a type of device, or a person might attend an event.
  - **Properties** These are attributes you can assign to a vertex or an edge.

Beware that you cannot mix these APIs in a single Cosmos DB account. You must define the API you want to use for accessing your Cosmos DB account when creating the account. Once you have created the account, you won't be able to change the API to access it.

Azure offers SDKs for working with the different APIs you can use to connect to Cosmos DB. Supported languages are .NET, Java, Node.js, and Python. Depending on the API you want to use for working with Cosmos DB, you can also use other languages such as Spring Data, Spark V3, or Golang.

NOTE **AZURE COSMOS DB EMULATOR**

You can use the Azure Cosmos DB emulator during the development stage of your application. You should keep in mind that there are some limitations when working with the emulator instead of a real Cosmos DB account. The emulator is only supported on Windows platforms or Docker for Windows. You can review all characteristics of the Cosmos DB emulator at *https://docs.microsoft.com/en-us/azure/cosmos-db/local-emulator*.

Index

A

B

C

D

E

F

G

H

I

J-K

L

M

N

O

P

Q

R

S

T

U

V

W-X-Y-Z