

# Pro ADO.NET Data Services

Working with RESTful Data



John Shaw and Simon Evans

## **Pro ADO.NET Data Services: Working with RESTful Data**

**Copyright © 2009 by John Shaw and Simon Evans**

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN-13 (pbk): 978-1-4302-1614-8

ISBN-13 (electronic): 978-1-4302-1615-5

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Lead Editor: Tony Campbell

Technical Reviewer: Damien Foggon

Editorial Board: Clay Andres, Steve Anglin, Mark Beckner, Ewan Buckingham, Tony Campbell, Gary Cornell, Jonathan Gennick, Michelle Lowman, Matthew Moodie, Duncan Parkes, Jeffrey Pepper, Frank Pohlmann, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh

Project Manager: Sofia Marchant

Copy Editors: Ami Knox, Heather Lang

Associate Production Director: Kari Brooks-Copony

Production Editor: Laura Cheu

Compositor: Kinetic Publishing Services, LLC

Proofreader: Nancy Bell

Indexer: Broccoli Information Management

Artist: Kinetic Publishing Services, LLC

Cover Designer: Kurt Krames

Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail [orders-ny@springer-sbm.com](mailto:orders-ny@springer-sbm.com), or visit <http://www.springeronline.com>.

For information on translations, please contact Apress directly at 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705. Phone 510-549-5930, fax 510-549-5939, e-mail [info@apress.com](mailto:info@apress.com), or visit <http://www.apress.com>.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales—eBook Licensing web page at <http://www.apress.com/info/bulksales>.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at <http://www.apress.com>.

*To Lori*

*Thanks for being understanding while I worked on this during  
our honeymoon . . . it proves to me how much you love me.*

*—John Shaw*

*To Angharad*

*Your soul makes me complete and your true friendship  
makes me love you more each passing day.*

*—Simon Evans*



# Contents at a Glance

Foreword .....	xiii
About the Authors. ....	xv
About the Technical Reviewer .....	xvi
Acknowledgments .....	xvii

## **PART 1 ■ ■ ■ ADO.NET Data Service Fundamentals**

■ <b>CHAPTER 1</b>	The Foundations of ADO.NET Data Services. ....	3
■ <b>CHAPTER 2</b>	Addressing RESTful ADO.NET Data Services .....	25
■ <b>CHAPTER 3</b>	ADO.NET Data Services Development .....	71

## **PART 2 ■ ■ ■ ADO.NET Data Services in the Real World**

■ <b>CHAPTER 4</b>	Exposing Existing WCF SOAP Services and .NET APIs Through ADO.NET Data Services.....	121
--------------------	--	-----

## **PART 3 ■ ■ ■ ADO.NET Data Services from the Outside**

■ <b>CHAPTER 5</b>	ASP.NET and AJAX Solutions. ....	155
■ <b>CHAPTER 6</b>	Mashing Up Data Services .....	195
■ <b>CHAPTER 7</b>	Silverlight 2.0 Solutions .....	235
■ <b>CHAPTER 8</b>	Using ADO.NET Data Services with BizTalk .....	257

## **PART 4 ■ ■ ■ The Future of ADO.NET Data Services**

■ <b>CHAPTER 9</b>	Conclusion.....	287
■ <b>INDEX</b>	.....	305



# Contents

Foreword .....	xiii
About the Authors. ....	xv
About the Technical Reviewer .....	xvi
Acknowledgments .....	xvii

## PART 1 ■ ■ ■ ADO.NET Data Service Fundamentals

■ CHAPTER 1	<b>The Foundations of ADO.NET Data Services.....</b>	<b>3</b>
	The Concept of Set-Based Logic.....	3
	The Concept of Object Orientation .....	4
	The Concept of Service Orientation .....	5
	RESTful Thinking .....	5
	The ABCs of Windows Communication Foundation.....	6
	Addresses .....	7
	Bindings.....	8
	Contracts .....	10
	Endpoint = Address + Binding + Contract .....	11
	Hosting Endpoints .....	11
	Don't Forget the Client .....	12
	Putting It All Together .....	12
	Standing on the Shoulders of Giants .....	13
	The Universal Service Contract .....	13
	Atom and JSON .....	14
	Data Service Hosting.....	18
	Data Service Clients.....	18
	Putting It All Together .....	18
	Reports of My Death Have Been Greatly Exaggerated .....	19
	LINQ to Something.....	21
	Defining LINQ .....	21
	LINQ to Local Data .....	22
	LINQ to Remote Data.....	23
	Summary.....	24

<b>CHAPTER 2</b>	<b>Addressing RESTful ADO.NET Data Services</b>	25
	Brief History of the Web Service	26
	HTTP Verb Usage	27
	Scoping Information	28
	What Makes a Service RESTful?	29
	HTTP Method	30
	Resources	30
	Addressability	30
	Relationships	30
	Statelessness	30
	Getting Started: Setting Up the Environment	31
	Visual Studio 2008 Components	31
	Development Approach	32
	Trying the New Data Service	40
	Addressing Data	46
	Entities (Resources)	48
	Entity Keys (Resource Keys)	50
	Entity Associations (Resource Navigations)	51
	Service Description	52
	Data Representations	54
	Formulating Queries	55
	Sorting Data	56
	Page Operations	56
	Expanding Related Data	57
	Filtering	61
	Hosting Data Services	64
	Surfacing Custom Data Sources Using CLR Classes	67
	Summary	70
 <b>CHAPTER 3</b>	 <b>ADO.NET Data Services Development</b>	 71
	ADO.NET Data Services Assemblies	72
	Debugging Data Services Using Fiddler	75
	Client Support	80
	DataServiceContext	81
	DataServiceQuery	82
	Creating Local Client Types	83
	Language Integrated Query Support	88
	Creating, Updating, and Deleting Data (CUD Operations)	90



Entity Associations . . . . .	92
Merging Entities . . . . .	93
Lazy Loading . . . . .	93
Asynchronous Actions . . . . .	94
Try Before You Buy . . . . .	96
More Data Services Features . . . . .	96
Scalar Values . . . . .	97
Data Interceptors . . . . .	99
Query Interceptors . . . . .	99
Change Interceptors . . . . .	100
Service Operations . . . . .	101
Batching . . . . .	103
Implementing Concurrency . . . . .	106
Securing Services . . . . .	111
Inheritance of Host Authentication . . . . .	111
Resource Visibility . . . . .	112
Intercepting Data . . . . .	114
Service Operations . . . . .	114
Error Handling Support . . . . .	115
Handled Exception . . . . .	115
Unhandled Exception . . . . .	116
HTTP Status Code Mapping . . . . .	118
Summary . . . . .	118

## PART 2 ■ ■ ■ ADO.NET Data Services in the Real World

CHAPTER 4	<b>Exposing Existing WCF SOAP Services and .NET APIs Through ADO.NET Data Services . . . . .</b>	<b>121</b>
	The Legacy Customer Service . . . . .	122
	Drivers for REST Support in the Customer Service . . . . .	128
	Refactoring the Customer Service for ADO.NET Data Services . . . . .	128
	Adding an ADO.NET Data Service Endpoint to the Refactored Service . . . . .	135
	Implementing Write Operations in the Customer Data Service . . . . .	141
	Summary . . . . .	151

## PART 3 ■ ■ ■ ADO.NET Data Services from the Outside

<b>CHAPTER 5</b>	<b>ASP.NET and AJAX Solutions</b>	155
	The Evolution of Web Applications	155
	The ASP.NET Server-Side Model	157
	Commonly Used ASP.NET Controls	158
	Forms and Postbacks	159
	State Management	159
	The Page Life Cycle	160
	Data Binding	161
	Partial Postbacks with the Update Panel	162
	Putting It All Together	163
	Using ADO.NET Data Services with ASP.NET Server Side	163
	ASP.NET Security	175
	Authentication	175
	Authorization	175
	Authentication and Authorization in ADO.NET Data Services	176
	The ASP.NET AJAX Client-Side Model	176
	Common AJAX Library Namespaces	177
	Introducing the Script Manager	178
	The Importance of Retail Mode	178
	Calling ADO.NET Data Services Using the Microsoft	
	AJAX Library	179
	Binding Data	180
	Public and Private Services	187
	Mixing ASP.NET Client and Server Models	192
	Summary	192
<b>CHAPTER 6</b>	<b>Mashing Up Data Services</b>	195
	Let's Do the Mash	195
	Sources of Mashups	197
	Intersecting Data	198
	Mashup Environments	198
	Using Microsoft Popfly	202
	Blocks	202
	Mashup Design Surface	204
	Web Page Creation	204
	Sharing Mashups	205

Block Development .....	206
XML Definition Files .....	208
JavaScript Class Definition .....	211
Handling Errors .....	214
Developing Popfly Blocks with Data Services .....	214
Data Retrieval Methods .....	214
Creating a Data Services Block .....	215
Creating a Restaurant Rating Mashup .....	217
Enterprise Mashups .....	232
Security .....	233
Data Services .....	233
SharePoint .....	233
Summary .....	233

## CHAPTER 7

### Silverlight 2.0 Solutions .....

The Web Application Spectrum .....	235
Light Up the Web .....	236
Silverlight Controls .....	237
Layout Controls .....	239
Input Controls .....	239
Shape Controls .....	240
Data Binding .....	240
Binding Single Object Properties .....	240
Binding to Collections .....	241
Consuming Services .....	242
Consuming SOAP Services .....	242
Consuming ADO.NET Data Services .....	243
Consuming Other REST and POX Services .....	244
Handling Cross-Domain Requests .....	245
Web Page Interoperability .....	246
Accessing the HTML DOM from Managed Code .....	246
Accessing Silverlight Managed Code from JavaScript .....	247
Debugging Silverlight Code .....	248
Summary .....	256

## CHAPTER 8

### Using ADO.NET Data Services with BizTalk .....

Why BizTalk? .....	257
Publish-and-Subscribe Architecture .....	259
Message Standards .....	261
Transformations (Mapping) .....	261

Pipelines .....	262
Orchestrations .....	262
Adapters .....	263
Configuring Your Environment.....	264
BizTalk 2006 R2.....	264
BizTalk 2009.....	266
BizTalk and ADO.NET Data Services .....	266
Review List of BizTalk Adapters .....	266
Configuring the WCF-Custom Adapter .....	270
Summary.....	282

## PART 4 ■ ■ ■ The Future of ADO.NET Data Services

■ CHAPTER 9	<b>Conclusion.....</b>	287
	Where Are We? .....	287
	Cloud Computing .....	288
	Benefits of Cloud Computing .....	288
	Drawbacks of Cloud Computing .....	289
	Main Players .....	290
	Where Does ADO.NET Data Services Fit In? .....	295
	One for the Road .....	296
	SQL Server Data Services .....	297
	Oslo .....	299
	Entity Framework V2.0 .....	300
	Astoria vNext .....	301
	Enriched Data Source Plug-in to Support High Scale Needs .....	301
	Friendly Feeds .....	302
	Offline ADO.NET Data Services .....	303
	Alignment with SQL Server Data Services.....	303
	ASP.NET DataServiceDataSource Server Control.....	304
	Summary.....	304
■	<b>INDEX .....</b>	305

# Foreword

It's amazing to see the things you can build with a mashup today. You run into a mashup tool, grab a couple of data sources, feed them into one of these fancy display options, and you end up with a working application that would have been really hard to build just a few years ago. Statistical data, visual maps, real-time activity of social networks, all right there, consolidated and displayed nicely after just 5 minutes of work.

When you think about the example of mashups, the technology to move data around and display it in nice visual representations has been around for a long time. The key element that enables all of these scenarios is availability of data. Ubiquity and simplicity add to the broad availability to complete the picture of a game-changing scenario where mashups are not only possible, but also trivial to build and effective in their application.

While this may sound easy and natural, sharing data effectively at the scale of the Web comes with its challenges. First of all, the Web is a big place, with lots of people and computers in it, so scalability is a big challenge. It's also a very diverse place, where keeping things simple and using low coupling is important. Finally, it's made of a bunch of smaller pieces, making composability and layering essential to make things work.

A number of folks have spent a lot of time reflecting about all this and documenting what makes the Web work well and what gets in the way of that. Things such as the REST architectural style observe and articulate the traits of applications that can satisfy the scale, layering, and long-term availability requirements of good web applications and services.

The goal of Project Astoria, which has become the ADO.NET Data Services framework, was to bridge the world of data locked in databases and other traditionally proprietary data sources with the world of the Web. We wanted to make it easy to create systems that expose data in a way that naturally followed the principles that make the Web work. We picked a resource model such that items of interesting granularity would become *resources* and the associations between those resources would be *links*. Every resource in the system is addressable with a URL, like any other web resource. Agents act on resources by using well-known verbs that client and server systems, as well as intermediaries, can understand and consider.

We tried hard not to invent anything. That way we could reuse everything out there, from proxies to authentication to representations. This could open the door to more data being available on the Web for applications to build on top of. Sometimes these will be simple applications in closed systems, sometimes large systems made of clients and servers exchanging data with minimum coupling, and sometimes they will involve shared data for anybody out there to use.

A small set of core principles is what makes the Web tick. We tried hard to capture them in the data services framework. But in order to successfully build applications that are good web citizens and benefit from its architecture, you must know these principles explicitly and know how the various APIs and options in the data services framework help you implement them.

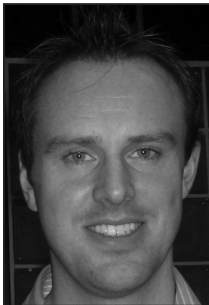
In this book you'll find a discussion of these principles followed by a practical description of what the ADO.NET Data Services framework can do, how it does it, and what it means for your application. Hopefully the combination of principles, framework, and the guidance in this book will help you in your next data application or service undertaking, and you'll have the Web and its goodness working for you.

Pablo Castro  
*Software Architect*  
*Microsoft Corporation*

# About the Authors



**SIMON EVANS** has worked for ten years as a software developer and technical architect. He has architected enterprise-scale applications and services for clients that are household names. Simon is a managing consultant for Conchango in the U.K., part of EMC Consulting Services. He is an expert in .NET development, and more specifically in WCF and ASP.NET, having participated in several Microsoft early adoption programs. Simon believes deeply that a broad understanding of key technology concepts is an essential foundation to being a gifted designer and builder of solutions. Simon is a regular contributor to the blogosphere through his blog at <http://blogs.conchango.com/simonevans>, where he writes about all .NET technologies including ADO.NET Data Services.



**JOHN SHAW** has worked in business processing and integration for over ten years and has architected some of the largest Microsoft integration projects in the U.S., U.K., and Australia. His technical expertise includes implementing Microsoft BizTalk and messaging standards for a variety of clients in finance, supply chain, and other sectors. Most recently, he founded SphereGen, a niche company that provides a blended model of low-cost onshore and offshore solutions for its clients. Prior to that, John worked as the U.S. BizTalk Practice Lead for Conchango Plc., where he obtained BizTalk Virtual Technical Specialist (VTS) status within Microsoft.

John is a regular contributor to the blogosphere through his blog at <http://blogs.spheregen.com/johnshaw>, where he writes about all .NET technologies including ADO.NET Data Services.

# About the Technical Reviewer

■ **DAMIEN FOGGON** is a freelance developer and technical author based in Newcastle, England. When not wondering why the Newcastle Falcons can never win at home or away, he spends his spare time writing, playing rugby, scuba diving, or pretending that he can cook.

If he could be consistent (or interesting), his blog might not be three months out of date. You never know, you may be lucky. See for yourself at <http://www.littlepond.co.uk>.



# Acknowledgments

I would like to thank my partner, Angharad, for her continuous help and support throughout the highs and lows of writing this book. I would also like to thank my parents for buying me my first computer at the age of eight and helping me learn to program it. Without this investment, I would not be in my chosen career.

Several people have helped in reading and providing opinion on the content of this book. Beyond everyone at Apress who have turned a rambling text into a coherent body of work, I would also like to thank my colleagues at Conchango who have provided valuable opinion. In particular I would like to thank Owain Wragg for providing an extra pair of eyes over much of the content of this book.

Simon Evans

I would like to thank my wife, Lori, for her love and support and her inspiration. Thanks also to my family and friends who have stuck by me over the years, particularly my parents, who helped me emerge from a little-known town called Warrington. Similar to Simon's parents, mine bought me a ZX spectrum when I was five years old, providing a solid programming foundation.

Additionally, I would like to thank Mark Beckner, Damien Foggon, Ami Knox, and the rest of the team at Apress, as well as Mike Taulty and Jon Flanders, who provided valuable early insights into ADO.NET Data Services. Lastly I would like to thank the Astoria team, particularly Pablo Castro and Mike Flasko, who have put together a great piece of software.

John Shaw

