

Praise for *Maximizing ASP.NET*

“Whether you want to improve your existing ASP.NET skills or are looking for a book that will give you the grounding and support you need to get started in ASP.NET development, this is the book for you! Jeff’s approach is simple—he explains new methods in a logical, no-nonsense way and includes real examples that demonstrate the .NET way to perform a traditional activity.”

—**John Timney, Microsoft MVP, ASP.NET**

Web Services Senior Consultant, British Telecom Consulting & Systems Integration

“I was excited about this book from the moment I first heard about it. I strongly believe ASP.NET developers will benefit greatly from understanding object-oriented fundamentals. Jeff has done a great job of introducing important object-oriented concepts clearly and concisely, enhancing the reader’s understanding with some excellent real-life code examples.”

—**Gordon Barrs, Independent Developer and Technical Training Consultant**

“This book covers a great spectrum of ASP.NET topics for those interested in discovering what ASP.NET has to offer. I particularly enjoyed the inclusion of ASP.NET 2.0, IIS 6.0, and Visual Studio 2005 information.”

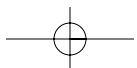
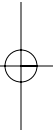
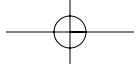
—**Tad Anderson, Enterprise Architect, Corporate Systems & Solutions**

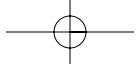
“A great book for ASP developers wanting to learn an object-oriented methodology.

—**Eric Landes, Microsoft MVP, ASP.NET, <http://blogs.aspadvise.com/elandes>**

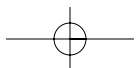
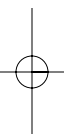
“This is an excellent resource for developers making the move from ASP to ASP.NET, as well as a good read for developers new to coding web pages on the IIS platform. I particularly appreciated the effort the author took to develop all the sample code in both C# and VB.NET.”

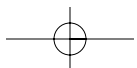
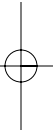
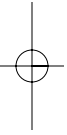
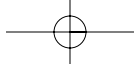
—**William “mac” McLuskie, Senior Solution Architect, Enterprise Consulting Services, Hewlett-Packard, Inc.**

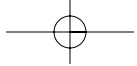




Maximizing ASP.NET



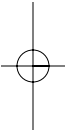
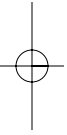




Maximizing ASP.NET

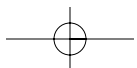
Real World, Object-Oriented Development

Jeffrey Putz



◆ Addison-Wesley

Upper Saddle River, NJ • Boston • Indianapolis • San Francisco
New York • Toronto • Montreal • London • Munich • Paris • Madrid
Capetown • Sydney • Tokyo • Singapore • Mexico City



Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.

The author and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

The publisher offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales, which may include electronic versions and/or custom covers and content particular to your business, training goals, marketing focus, and branding interests. For more information, please contact:

U. S. Corporate and Government Sales
(800) 382-3419
corpsales@pearsontechgroup.com

For sales outside the U. S., please contact:

International Sales
international@pearsoned.com

Visit us on the Web: www.awprofessional.com

Library of Congress Catalog Number: 2004116048

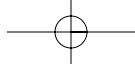
Copyright © 2005 Pearson Education, Inc.

All rights reserved. Printed in the United States of America. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, write to:

Pearson Education, Inc.
Rights and Contracts Department
One Lake Street
Upper Saddle River, NJ 07458

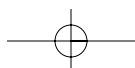
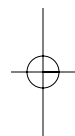
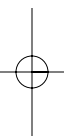
ISBN 0-32-129447-5

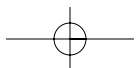
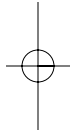
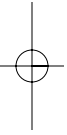
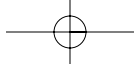
Text printed in the United States on recycled paper at R.R. Donnelley, Crawfordsville, Indiana.
First printing, March, 2005



For Stephanie

*This book, and indeed the wonderful lives we lead, would
not be possible without your constant support and love.
I can't imagine a universe without you.*





Contents

Preface	.xv
Acknowledgments	.xxi
About the Author	.xxiii

Part I: The Leap to Object-Oriented Programming1

Chapter 1: The Object Model3

Breaking Away from the Linear Script Way of Thinking	.3
Classes and Objects	.7
Analogy to a Car	.7
Object Inheritance	.9
Your Classes Are as Special as Microsoft's	.14
Summary	.15

Chapter 2: Classes: The Code Behind the Objects17

Access Modifiers	.18
The Class Declaration	.18
Constructors	.19
Properties and Private Variables	.21
Methods	.23
Member Scope	.26
Overloading	.27
Static (Shared) Members	.28
Inheritance	.29
Interfaces	.30
Abstract Classes	.32
Enumerations	.33
Summary	.34

X Contents

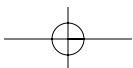
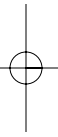
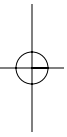
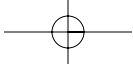
Chapter 3: Class Design	.35
Assessing Your Needs and Requirements	.35
Narrowing Your Focus	.36
Getting Data In and Out	.37
Instantiated Classes Versus Static Methods	.41
Using Classes for Simple Data Structures	.42
Test-Driven Development	.43
Summary	.44
Chapter 4: Application Architecture	.45
The Benefit of Splitting Applications into Layers	.45
The Classic n-tier	.47
Case Study: The POP Forums Architecture	.48
Talking to Other Applications	.59
Assessment of Needs and Requirements in the Real World	.60
Summary	.60
Chapter 5: Object-Oriented Programming Applied: A Custom Data Class	.61
Analyzing Design Requirements	.63
Choosing Our Properties	.64
The Constructors	.69
Create, Update, and Delete Methods	.73
Caching the Data for Better Performance	.77
Getting More than One Record at a Time	.83
Summary	.87
Part II: The ASP.NET Architecture	.89
Chapter 6: The Nuts and Bolts of IIS and Web Applications	.91
IIS: The First Step	.91
Deciding What ASP.NET Will Handle	.95
Where the Code Lives	.95
Pre-Compiled Deployment	.103
Summary	.104

Chapter 7: The ASP.NET Event Model	.105
History	.105
The Postback Process	.107
Server Controls, Their Events and Event Handlers	.110
The Application and Page Events	.111
Postback Events	.113
Wiring Events	.114
Summary	.116
Chapter 8: HttpHandlers and HttpModules	.117
The Built-in Handlers	.117
Handling File Types of Your Own	.119
Example: Protecting Your Images from Bandwidth Leeching	.122
HttpModules: The Replacement for ISAPI	.128
Example: Adding Users to Roles	.130
Summary	.137
Chapter 9: Server Controls	.139
Simple Derived Controls	.139
Building Your Own Server Control from Scratch	.145
Using Viewstate	.155
Efficient Rendering of Your HTML	.155
Composite Controls	.156
Composite Controls Versus User Controls	.162
Summary	.163
Chapter 10: Web Services as Objects	.165
Converting Your Code to a Web Service	.166
Consuming a Web Service	.173
Object-Oriented Programming Without “Owning” the Object	.180
Modifying the Proxy Class	.180
Summary	.181
Chapter 11: Membership and Security	.183
Forms Authentication and Membership	.183
The Built-in Membership Provider	.186

xii Contents

Building Your Own Provider	189
Authentication Controls	202
Summary	203
Chapter 12: Profiles, Themes, and Skins	205
Profiles in Relation to Membership Records	205
Building Your Own Profile Provider	209
Themes	221
Skinning Controls	223
Tying the Look to User Profiles	225
Summary	226
Chapter 13: Declarative Programming	227
Overview	227
Validation Controls	228
Data Controls	229
Web Parts	232
Navigation Controls	233
Expressions	235
Summary	235
Part III: Development Issues	237
Chapter 14: Developing with Visual Studio	239
Versions	239
Default Behavior of the IDE	241
Pitfalls of the Designer	241
Multiple Projects and References	242
The Debugger	246
Summary	247
Chapter 15: Performance, Scalability, and Metrics	249
Terms Defined	249
Fragment Caching	250
Data Caching	250

Managing Viewstate253
Variations in Data Access Methods254
String Concatenation vs. StringBuilder255
Using Trace to Measure Time258
ACT and WAS258
Performance Monitor259
Hardware261
SQL Profiler and Query Analyzer261
Summary264
Chapter 16: Testing, Deployment, and Code Management267
Test, Test, and Retest with All Stakeholders267
Test-Driven Development268
Getting the Bits Out to Production278
Versioning and Splitting Up the Work279
Summary280
Chapter 17: More Advanced Topics283
Streams283
Networking290
Threading296
Index301



Preface

Microsoft has endowed Web developers with a gift. Since its introduction to the masses in beta form in 2001, the .NET Framework and its ASP.NET subset have changed the way we look at building Web applications. Indeed, before this time, many of us didn't even think in terms of "applications" but rather collections of loosely related scripts that together formed a Web site.

Making this transition required a serious leap of faith and a great deal of new learning for a lot of developers. Scripting languages such as ASP 3.0, ColdFusion, and PHP made it fairly easy for someone with little or no programming experience to quickly learn about the platform and start building dynamic Web sites. Years of experience as a developer or an academic background in computer science were not necessary.

That background still isn't necessary, but with ASP.NET, script developers need to make some fairly radical changes in their thinking to get the most out of this amazing platform. This book aims to help you with that transition, or to get you familiar with the platform's architecture if you're already well versed in object-oriented design.

Why Was This Book Written?

In watching the various message boards and newsgroups around the Internet, it became clear to me that a lot of very smart developers are having some problems making the transition to the object-oriented world of ASP.NET. As much as we praise Microsoft for delivering the platform to us, we can also curse them for making it so inviting and safe because it doesn't force you to follow the "best practices" they envisioned. You can do things in almost the same way you did when you were using a scripting platform.

It would be easy to create a straight book on techniques or a “cookbook” of sorts to help you along, but what I’m really after is a guide that helps you understand the underlying concepts and architecture to the platform so that you can apply the same concepts to your own applications. Indeed, I point out in the second chapter that the code you write isn’t any different than the code that Microsoft wrote into the thousands of existing .NET base classes. A cookbook or overview wouldn’t help you understand this.

This book is not meant to evangelize object-oriented techniques. The idea is to show you enough rationale behind the concepts to encourage you to use them. I want light bulbs to go off in your head that say, “Oh yeah, I get it!” Using OOP just for the sake of doing so is not good.

Who Is This Book For?

This book assumes that you’ve already taken the first steps in learning about ASP.NET and have a relatively basic grasp of either the C# or Visual Basic .NET languages. It is also assumed that you have some basic understanding of how to use Microsoft SQL Server and understand what a relational database is. You might be a developer who in a previous life was primarily a script developer and wants to “get” the platform and its strong object-oriented architecture. You might also be a more seasoned developer who just wants to get into some of the platform-specific details of ASP.NET that make things tick under the hood. These developers may choose to skip around a bit.

Regardless of the camp you might be in, this book is all about context. Developers are generally very smart people, and they don’t learn by memorization, they learn by understanding. This book takes the basics of object-oriented programming and applies them to ASP.NET and Visual Studio to give the reader a more rounded skillset that includes application design, test-driven development, code reuse, modularization, and an eye on performance. The book puts Microsoft’s platform into context by moving beyond the “how” and into the “why,” not just explaining the concepts but also selling them as the best way to solve real problems. If you come from a scripting background or want to make your skills more applicable to other areas of .NET programming, this book is for you.

If you are totally new to ASP.NET, this book isn't for you—yet. You should start with one of the many ASP.NET books that have “beginner” or “introduction” in the title. When you've got the basics, come back to this book to round out and strengthen your knowledge. We won't cover basics such as master pages, user controls, connecting to a database, configuring `FormsAuthentication`, and so on.

I want to say up front that the term “best practice” is a bit of a misnomer. Some concepts and methodologies are certainly better than others, but this term implies that universal truths abound in the world of programming. Some might believe that the only real universal truths are death and taxes, so while you read about what I believe are best practices, keep in mind that there's plenty of room for interpretation based on your experience and the business problems you're trying to solve. As with anything else, use what works and adapt it to your needs.

Conventions

Whenever there is a bit of code to show you, it will look like this:

C#

```
public class ThisIsAGreatClass
{
    // lots of code here
}
```

VB.NET

```
Public Class ThisIsAGreatClass
    ' lots of code here
End Class
```

Special points, sidebars, important notes, and other tangents will be separated from the rest of the text like this:

This is something you should take into consideration.

Play-by-Play

The following is a brief outline of what you can expect to find in the rest of this book:

Part I: The Leap to Object-Oriented Programming

Chapter 1, “The Object Model.” Object-oriented programming has been around for ages, but it’s a new concept for developers with a scripting background. You’ll learn that everything is an object in .NET, including your own code. We’ll analogize the concept of object-oriented programming to the classic example of the structure and use of cars.

Chapter 2, “Classes: The Code Behind the Objects.” After you see what an object is and how it contains some kind of functionality, we’ll get into the nuts and bolts of writing a class. You’ll learn about the instantiation of objects, inheritance, protection levels, static methods, enumerations, and interfaces.

Chapter 3, “Class Design.” A class can serve many different purposes in an application. Your classes can execute a block of code, much as the familiar `System.Web.UI.Page` class does, and at times they can do nothing other than group data together. More useful classes can do all these things.

Chapter 4, “Application Architecture.” Apart from ASP.NET, the n-tier approach to application development can make even the largest projects easier to build, deploy, and maintain. We’ll take a look at this common design pattern and address when and when not to use it.

Chapter 5, “Object-Oriented Programming Applied: A Custom Data Class.” This chapter presents you with a practical example of class design that manipulates database data and even caches it for better performance. This model shows you the payoff of object-oriented code, where you write it once and use it many times.

Part II: The ASP.NET Architecture

Chapter 6, “The Nuts and Bolts of IIS and Web Applications.” Regardless of how you code your application, the files, folders, and assemblies need to be in the right places to make everything work.

Chapter 7, “The ASP.NET Event Model.” Every request by users of your application has a fascinating life cycle. By understanding and following that life cycle, you can intervene or perform certain actions at just the

right time to get the most out of your application. The execution of page, application, and control events is covered.

Chapter 8, “HttpHandlers and HttpModules.” ASP.NET can do so much more than produce pages, and HttpHandlers and HttpModules are just the tools you’ll need for special requests. I’ll give you an example of a handler that will protect your images from being bandwidth leeched.

Chapter 9, “Server Controls.” You might find that you use the same combination of controls frequently or that an existing control doesn’t meet your needs. Building your own isn’t that hard, and it’s exactly the same process used by Microsoft to create many of the controls you already use.

Chapter 10, “Web Services as Objects.” It’s easy to create a Web service from your existing code, and it’s easy to create a proxy class that consumes the service. What might not be as obvious is that Web services enable you to use that remote code as if it were an object local to your application.

Chapter 11, “Membership and Security.” ASP.NET has a rich set of classes to help you control access to your site and verify who your users are. You can use little or no code to identify your users, or extend the system to make your own custom solution. You’ll learn how to develop a custom Membership provider to connect the system to your own data.

Chapter 12, “Profiles, Themes, and Skins.” Chances are you’ll want to keep more than just a user’s email and password, and again the ASP.NET team has made it possible to store this data with little effort on your part. You can also take the leap to extend the system with your own provider. Your users’ preferences can be tied into an extensive skinning system so they don’t have to live with your site’s “look.”

Chapter 13, “Declarative Programming.” With all this talk of object-oriented code, what is declarative programming, and where does it fit? We’ll cover some of the common controls and their uses available in ASP.NET.

Part III: Development Issues

Chapter 14, “Developing with Visual Studio.” Heralded by many as the greatest development tool ever made, it’s not without its nuances and quirks. It quite possibly is the greatest tool, but you’ll have to start thinking a little like a desktop application developer. We’ll look at the approach the software takes to making your events happen and how to debug with the best of them.

XX Preface

Chapter 15, “Performance, Scalability, and Metrics.” There are probably hundreds of things you can do to make your code industrial-strength, but there are a few basics you should know and live by. A number of tools and techniques provide the foundation for measuring the performance of your application.

Chapter 16, “Testing, Deployment, and Code Management.” The bigger a project gets, the harder it can be to maintain and deploy. There are a number of ways to keep everything organized while testing and versioning your code. We’ll discuss the basics of test-driven development here.

Chapter 17, “More Advanced Topics.” The .NET Framework has more classes than you can shake a stick at. Here are a few of the more useful ones in a high-level overview, including streams, networking, and threading.

Code Downloads, Errata, and Feedback

You can download the code from this book, check for errata, and give feedback at this site:

<http://www.uberasp.net/books/>

Acknowledgments

There are a lot of people who made this book possible, starting with my wife, Stephanie Gall. Her constant support and encouragement gave me the will to leave a cushy high-paying job to write this book.

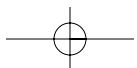
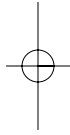
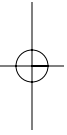
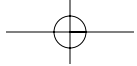
Special thanks go to Walt Schmidt, my partner on *PointBuzz: The Ultimate Guide to Cedar Point* (<http://www.pointbuzz.com>). Through the many discussions we had about our “little” hobby site, my programming advice to Walt was met with the advice that I should write a book.

A lot of people I’ve worked with over the years were crucial to my professional development as an Internet code monkey. Perry Trunick at Penton Media gave me my first job in this role over someone with certifications, all on a gut feeling. Tim Walsh, Bob Eck, Tony D’Avino, Jay Davis, Mike Freeze, Mary Aichlmayr and others had an impact as well. We were involved in really great things during the Internet boom!

Katie and Mark Bruno of Website Design and Development in Wadsworth, Ohio, also helped me realize my potential as a teacher of sorts by giving me the opportunity to teach their staff in the ways of .NET.

I’d like to thank the people who did the editorial reviews for Addison-Wesley, especially Gordon Barrs, for making sure I was covering the right things.

Finally, congratulations to the many people involved with .NET at Microsoft. It’s through their hard work and vision that we all have this great platform to work with. Their products have had a profound impact on my professional life and indirectly on the users of my own software and Web sites. I’d specifically like to acknowledge Brian Goldfarb at Microsoft for keeping me in the loop through changes leading up to production for this book. Writing about software that isn’t yet done was a lot easier with his help.



About the Author

Jeff Putz is the founder of POP World Media, LLC, a company formed originally to organize several hobby Web sites into a bona fide business. Jeff started programming in grade six on a TRS-80 and moved up through an Atari 600XL and Apple II+ later. After flirting with various jobs in the broadcast world, Jeff returned to computers and welcomed the Internet in 1998, working in various programming and management roles for several companies specializing in vertical market content.

Jeff's POP Forums application (<http://www.popforums.com>) has been downloaded tens of thousands of times and was featured in *MSDN Magazine* (Feb. 2004). His company is in the process of developing several products using ASP.NET and continues to operate CoasterBuzz (<http://www.coasterbuzz.com>), the world's most popular roller coaster enthusiast site.

