

Preface



“... the highest simplicity of structure is produced, not by a few elements, but by the highest complexity.”

—Ralph Waldo Emerson, “Goethe; or, the Writer,”
Representative Men, Chapter 8

In November 2005, version 2.0 of Microsoft’s .NET Framework and ASP.NET was released along with a new version of its Visual Studio development environment. This new version of ASP.NET significantly increased its power as well as its complexity. When I first began teaching Web application development back in 1999 with classic ASP, I could teach my students the essentials of ASP in just two weeks. Of course, to create a sample application of even moderate complexity in ASP required the students to do a great deal of coding. Now, with ASP.NET 2.0, it requires almost two-thirds of a semester to teach my students ASP.NET. The students, however, now can create a sample application of substantial complexity. That is, although ASP.NET 2.0 can be complex, it can dramatically improve a Web developer’s productivity. Thus, to paraphrase Emerson, after the developer has grasped and comprehended the seeming initial complexity of ASP.NET, he may very well be struck by its ultimate simplicity, even its beauty.

This book endeavors to help the reader make the transition from complexity to simplicity. That is, it tries to make the process of learning how to create realistic Web applications using ASP.NET 2.0 less daunting for readers who are unfamiliar with ASP.NET, as well as for readers who are somewhat familiar with ASP.NET but want to learn how to use it more effectively.

As part of the process of learning how to create realistic Web applications, this book also endeavors to stress the importance of proper programming and design principles. When first learning ASP.NET, a developer is often tempted to focus all of her attention on using the many different Web server controls along with the Visual Studio Designer. This is quite understandable given the range and power of these controls and the simplicity and functionality of Visual Studio. However, as you create more complex “real-world” Web applications, other considerations, such as maintainability, scalability, and adaptability, become progressively more important. As a consequence, this book’s ultimate aim is to help the reader (you) become not only proficient with ASP.NET 2.0, but also to help you become a better Web application developer by also focusing on contemporary best practices in Web application development.

Target Audience

This book is intended first and foremost for professional developers who desire to learn how to create Web applications using the latest version of Microsoft’s ASP.NET. Because I teach Web development at a college, this book is also intended for potential use in the classroom for upper-level students taking a course in Web application development using ASP.NET.

Prerequisites

This book assumes that the reader already knows the basics of HTML and CSS. It does not assume any knowledge of ASP.NET or C#. The book does assume that you are familiar with programming using an object-oriented language. As a result, this book does not provide detailed coverage of C# (for that, see Stephen Perry’s *Core C# and .NET* from Prentice Hall, 2006); instead, the book illustrates how to use C# in conjunction with ASP.NET.

The book also contains the occasional UML diagrams. Although knowledge of the UML (Unified Modeling Language) may increase your understanding, it is by no means a necessity for this book. This book also assumes that you are familiar in general with databases and XML.

Approach

This book tries to provide you with a clear path to learning how to effectively and realistically use ASP.NET 2.0 for creating Web applications. Due to the sheer size of ASP.NET 2.0, a fair amount of book space is used just to teach the very basics of

ASP.NET. This book's approach is to verge on the side of conciseness in regard to the very basics in order to spend more time with the issues you typically face after you master those basics.

As should be no surprise in a book that is about software development, there is a fair amount of programming code in this book. Much of the code consists of very short code snippets. There are, however, the occasional longer code listings. These listings provide a more complex completed example, such as an RSS reader, a file manager for uploading and downloading files, or a sample business object. These listings are all heavily commented so that you can learn not only from the book's text, but also from these longer code listings. Most chapters also contain a few walk-through exercises. These are a set of step-by-step instructions for accomplishing some task in ASP.NET.

There are several possible pathways through each chapter. A reader could focus principally on the main text and its code snippets, and skip over the longer code listings and the walkthrough exercises. Other readers might prefer to first work through a chapter's walkthrough exercises, and then read through the text to extract a fuller understanding of the chapter's content. Other readers might glance through the chapter text, and then "read" the longer code listings.

The book is structured so that a reader with no knowledge of ASP.NET can progress linearly through the chapters, in that material in one chapter builds on knowledge of the material presented in earlier chapters. However, the material is presented in such a way that a reader can take a more "random" approach, skipping forward and backward to the material that is of interest to her.

The approach and sequence of topics in this book were chosen principally as a result of my experience teaching ASP.NET in the classroom to undergraduates, as well as to professional developers. It was also influenced by my own experiences using ASP.NET professionally for real-world clients.

Overview

This book is broken into three principal parts. The first part is "Core ASP.NET," and consists of the first seven chapters of the book. These chapters introduce and explore the key fundamental features of ASP.NET. The second part encompasses the next five chapters: "Working with Data." It focuses on perhaps the most important aspect of any Web application: representing, extracting, manipulating, and presenting data to the user. The third and final part contains four chapters: "Implementing Web Applications." Its focus is the more advanced side of application development with ASP.NET: security, personalization, Web services, and localization and deployment.

Chapter 1 introduces ASP.NET and the .NET Framework. It examines the different components of the .NET platform, compares ASP.NET to other Web

development environments, describes the ASP.NET compilation model, examines the event system in ASP.NET, and illustrates how to create simple ASP.NET pages using Visual Studio 2005.

Chapter 2 continues the coverage of the basics of ASP.NET. This chapter examines in depth how ASP.NET works. It describes the event system in ASP.NET, the page lifecycle, and the essential mechanisms of postback and view state. It also covers some more advanced topics that could be skipped and returned to after you become more comfortable with ASP.NET. These topics include the ASP.NET compilation model, the `Page` class, as well as the application lifecycle.

Chapter 3 provides an overview of ASP.NET's Web server control architecture, covers Web forms syntax, examines how to use the common features of all Web server controls, and provides and illustrates how to use a core subset of the standard Web server controls. Because ASP.NET 2.0 now has so many Web server controls, some of the less frequently used core Web server controls are covered in Chapter 4.

Chapter 4 continues the coverage of the standard Web server controls. The controls covered in this chapter are more complicated. Some of the controls covered in this chapter include the `Panel`, `MultiView`, `Wizard`, `FileUpload`, and `Xml` controls. Several of the longest code listings in the entire book are in this chapter.

Chapter 5 covers one of the most important facets of Web application development, namely, how to deal with exceptions, both at the language level and at the ASP.NET level. It also illustrates how to use the ASP.NET validation controls.

Chapter 6 examines how to create complex user interfaces using styles, themes, skins, and master pages. The chapter also covers the creation of your own user controls.

Chapter 7 examines how to describe and create a site's navigation system using the ASP.NET site navigation controls.

Chapter 8 is the first chapter of the second part of the book. ASP.NET 2.0 introduces a new way of working with data and this chapter's focus is on the different ways that data can be represented. It covers data binding, arrays, collections, generics, and data sets.

Chapter 9 continues the material from Chapter 8 by examining how to programmatically and declaratively work with data in databases. This chapter begins by examining how to access and modify data within databases in ASP.NET using the classes of ADO.NET. The chapter also covers the codeless approach to accessing data using the data source controls introduced in version 2.0 of ASP.NET.

Chapter 10 illustrates how to use the various data controls in ASP.NET. It illustrates the use of the `Repeater`, `DataList`, `FormView`, `DetailsView`, and `GridView` controls. Each of these controls uses data binding to display (and for some even edit) multiple sets of data in different ways.

Chapter 11 shifts the focus away from individual controls and classes and instead examines some of the issues involved in creating a more complex Web application with ASP.NET. It begins with the design of Web applications and some

common layering models for Web applications, and then moves on to implement two sample layered architectures.

Chapter 12 covers an aspect of ASP.NET that is vital for any Web application: managing state. This chapter begins with the various types of ASP.NET state whose data is stored on the client, such as view state and cookies. It then moves on to those state mechanisms whose data is stored in the server: session state, application state, and finally the ASP.NET cache.

Chapter 13 is the first chapter of the final part of the book. It covers security, one of the most important aspects of any Web application. It discusses authentication and authorization in the context of ASP.NET, illustrates how to use the various login control as well as the new provider system, including the membership and role management providers.

Chapter 14 examines two mechanisms in ASP.NET 2.0 for integrating user personalization into a Web application: namely, the profile system and the Web part framework. The profile system allows an application to persist user information across time and multiple pages. The Web part framework provides the developer with a mechanism for creating pages in which the user can customize the placement, appearance, and possibly the behavior of the page's content.

Chapter 15 looks at how to synchronously and asynchronously consume Web services in ASP.NET. The chapter also demonstrates how to construct Web services.

Chapter 16 demonstrates how to plan and adapt an ASP.NET application for an international audience, as well as the various ways to deploy a completed ASP.NET Web application.

The **Appendix** provides a preliminary examination of ASP.NET AJAX, which up until the fall of 2006 was known by the code-name Atlas. ASP.NET AJAX is a free framework from Microsoft that adds Asynchronous JavaScript and XML (AJAX) support to ASP.NET. ASP.NET AJAX encompasses a fairly large set of functionality that integrates client script (Javascript) libraries with ASP.NET server-based pages. It provides an API for working with Javascript, a declarative alternative to working with Javascript, rich client-script components, along with special Atlas server-side controls.

Supplementary Materials

The Web site for this book is <http://www.randyconnolly.com/core>. It contains

- The source code for all the examples in the book
- All databases, images, and style sheets used in the chapter examples
- Solutions to the practice exercises that are at the end of each chapter
- A list of known errors in the book and the code
- A form for submitting corrections and suggestions
- Downloadable versions of additional or updated appendices

This book is also intended for potential use in the classroom for upper-level students taking a course in Web application development using ASP.NET. For educators who adopt this book for their courses, the following material is available from this same site:

- Powerpoint lectures for each chapter
- Recommended syllabi and detailed lesson plans for half-semester courses
- Assignments and course projects
- Multiple-choice, short-answer, and long-answer examination questions
- Laboratory tutorials

Prentice Hall also maintains a book Web page that contains additional information: www.prenhallprofessional.com/title/0321419502.