Introduction

Thank you for picking up *Windows Presentation Foundation Unleashed*! To avoid unsatisfied customers, I want to clarify that this is *not* a book about Microsoft PowerPoint (which many people consider to be the *foundation* of *Windows presentations*)! Windows Presentation Foundation (WPF) is Microsoft's latest technology for creating graphical user interfaces, whether they consist of plain forms, document-centric windows, animated cartoons, videos, immersive 3D environments, or all of the above! This is a technology that makes it easier than ever to create a broad range of applications. For example, WPF makes it relatively straightforward to implement applications similar to Windows Media Player, Microsoft Word (or at least WordPad), and, yes, even Microsoft PowerPoint!

Ever since WPF was publicly announced in 2003 (with the code name of "Avalon"), it has gotten considerable (and deserved) attention for the ways in which it revolutionizes the process of creating software—especially for Windows programmers used to Windows Forms and GDI. It's relatively easy to create fun, useful, and shareable WPF samples that demonstrate all kinds of techniques that are difficult to accomplish in other technologies. But WPF is quite a departure from previous technologies in terms of its programming model, underlying concepts, and basic terminology. Even viewing the source code for WPF (by cracking open its components with a tool like .NET Reflector) is a confusing experience because the code you're looking for often doesn't reside where you'd expect. When you combine all of this with the fact that there are often several ways to accomplish any task, you arrive at a conclusion shared by many: *WPF has a very steep learning curve*.

That's where this book comes in. Five years ago, I wrote *.NET and COM: The Complete Interoperability Guide* because I felt there was a need for an entire book to guide people through such a deep and complex topic. As WPF was developed, it was obvious that there would be no shortage of WPF books in the marketplace. But it wasn't clear to me that the books would have the right balance to guide people through the technology and its unique concepts while showing practical ways to exploit it. Therefore, I wrote *Windows Presentation Foundation Unleashed* with the following goals in mind:

- To provide a solid grounding in the underlying concepts in a practical and approachable fashion
- To answer the questions most people have when learning the technology, and to show how commonly desired tasks are accomplished
- To be an authoritative source, thanks to input from members of the WPF team who designed, implemented, and tested the technology
- To be clear about where the technology falls short, rather than selling the technology as the answer to all problems
- To be an easily navigated reference that you can constantly come back to

I hope you find this book to exhibit all of these attributes.

Who Should Read This Book?

This book is for software developers who are interested in user interfaces. Regardless of whether you're creating line-of-business applications, consumer-facing applications, or reusable controls, this book contains a lot of content that helps you get the most out of the platform. It's designed to be understandable even for folks who are new to .NET. And if you are already well versed in WPF, I'm confident that this book still has things to teach you. At the very least, it should be an invaluable reference for your bookshelf.

Because WPF enables you to create not only standalone Windows applications but also content hosted in a web browser, anyone interested in alternatives to Adobe Flash might find this book interesting. And although the more lightweight and cross-platform Windows Presentation Foundation Everywhere (WPF/E) technology does not have significant coverage in this book, many of the same concepts in this book will apply to WPF/E once it is released.

Although this book's content is not optimized for graphic designers, reading this book can be a great way to understand more of the "guts" behind a product like Microsoft Expression Blend.

To summarize, this book

- Covers everything you need to know about Extensible Application Markup Language (XAML), the new XML-based language for creating declarative user interfaces that can be easily restyled
- Examines the WPF feature areas in incredible depth: controls, layout, resources, data binding, styling, graphics, animation, and more
- Delves into topics that aren't covered by most books: 3D, speech, audio/video, documents, bitmap effects, and more
- Shows how to create popular UI elements, such as features introduced in the 2007 Microsoft Office System: Galleries, ScreenTips, custom control layouts, and more
- Demonstrates how to create sophisticated UI mechanisms, such as Visual Studio-like collapsible/dockable panes
- Explains how to develop and deploy all types of applications, including navigation-based applications, applications hosted in a web browser, and applications with great-looking nonrectangular windows
- Explains how to create first-class custom controls for WPF
- Demonstrates how to create hybrid WPF software that leverages Windows Forms, ActiveX, or other non-WPF technologies
- Explains how to exploit new Windows Vista features in WPF applications, and how to go beyond certain limitations of WPF

This book doesn’t cover every last bit of WPF. (In particular, XML Paper Specification [XPS] documents are only given a small bit of attention.) WPF’s surface area is so large that I don’t believe any single book can. But I think you’ll be pleased with the breadth and depth achieved by this book.

Examples in this book appear in XAML and C#, plus C++/CLI for interoperability discussions. XAML is used heavily for a number of reasons: It’s often the most concise way to express source code, it can often be pasted into tools like XamlPad (in the Windows SDK) to see instant results without any compilation, WPF-based tools generate XAML rather than procedural code, and XAML is applicable no matter what .NET language you use, such as Visual Basic instead of C#. Whenever the mapping between XAML and a language like C# is not obvious, examples are shown in both representations.

Software Requirements

This book targets the final release of version 3.0 of Windows Presentation Foundation, the corresponding Windows SDK, and the October 2006 release of .NET Framework 3.0 extensions to Visual Studio 2005.

The following software is required:

- A version of Windows that supports the .NET Framework 3.0. This can be Windows XP with Service Pack 2 (including Media Center, Tablet PC, and x64 editions), Windows Server 2003 with Service Pack 1 (including the R2 edition), Windows Vista, or later versions.
- The .NET Framework 3.0, which is installed by default starting with Windows Vista. For earlier versions of Windows, you can download it for free from http://msdn.com.

In addition, the following software is recommended:

- The Windows Software Development Kit (SDK), specifically the .NET tools it includes. This is also a free download from http://msdn.com.
- Visual Studio 2005 or later, which can be a free Express edition downloaded from http://msdn.com. If you’re using a 2005 edition of Visual Studio (Express or otherwise), you should download the extensions for .NET Framework 3.0 development available from MSDN. This is not necessary for later versions of Visual Studio.

If you want additional tool support for WPF-based graphic design, Microsoft Expression can be extremely helpful. See the appendix, “Helpful Tools,” for other pieces of software that can be helpful for WPF design and development.

A few examples in Chapter 7, “Structuring and Deploying an Application,” are specific to Windows Vista, but the rest of the book applies equally to all relevant versions of Windows.

Code Examples

The source code for examples in this book can be downloaded via www.samspublishing.com or www.adamnathan.net/wpf.

How This Book Is Organized

This book is arranged into five main parts, representing the progression of feature areas that you typically need to understand to use WPF effectively. But if you're dying to jump ahead and learn about a topic such as 3D or animation, the book is set up to allow for nonlinear journeys as well. The following sections provide a summary of each part.

Part I: Background

The book introduces WPF by comparing it to alternative technologies and helping you make decisions about when WPF is appropriate for your needs. Chapter 2 explores XAML in great depth, giving you the foundation to understand the XAML you'll encounter in the rest of the book and in real life. Chapter 3 highlights the most unique pieces of WPF's programming model above and beyond what .NET programmers already understand.

- Chapter 1: Why Windows Presentation Foundation?
- Chapter 2: XAML Demystified
- Chapter 3: Important New Concepts in WPF

Part II: Building a WPF Application

Part II equips you with the knowledge to assemble and deploy a traditional-looking application (although some fancier effects like transforms, non-rectangular windows, and Aero glass are also covered). It begins by introducing WPF's implementation of controls you'd expect to have available, plus a few that you might not expect. It then devotes two chapters to arranging such controls (and other elements) in a user interface. Chapter 7 ends this part by examining several different ways to take WPF-based user interfaces and package and deploy complete applications. This not only includes traditional standalone Windows applications, but also applications that are more like web pages.

- Chapter 4: Introducing WPF's Controls
- Chapter 5: Sizing, Positioning, and Transforming Elements
- Chapter 6: Layout with Panels
- Chapter 7: Structuring and Deploying an Application

Part III: Features for Professional Developers

The features covered in Part III are not always necessary to use in WPF applications, but they can greatly enhance the development process. Therefore, they tend to be indispensable for professional developers who are serious about creating maintainable and robust applications or components.

- Chapter 8: Resources
- Chapter 9: Data Binding
- Chapter 10: Styles, Templates, Skins, and Themes

Part IV: Going Beyond Today’s Applications with Rich Media

This part of the book covers the features in WPF that typically get the most attention. The support for 2D and 3D graphics, animation, video, and more enable you to create a stunning experience. These features—and the way they are exposed—set WPF apart from previous systems. WPF lowers the barrier to incorporating such content in your software, so you might try some of these features that you never would have dared to in the past!

- Chapter 11: 2D Graphics
- Chapter 12: 3D Graphics
- Chapter 13: Animation
- Chapter 14: Audio, Video, Speech, and Documents

Part V: Advanced Topics

The topics covered in Part V are relevant for advanced application developers, or developers of WPF-based controls. Because existing WPF controls can be radically restyled, the need for creating custom controls is greatly reduced.

- Chapter 15: Interoperability with Win32, Windows Forms, and ActiveX
- Chapter 16: User Controls and Custom Controls
- Chapter 17: Layout with Custom Panels

Conventions Used in This Book

Various typefaces in this book identify terms and other special items. These typefaces include the following:

Typeface	Meaning
Italic	Italic is used for new terms or phrases when they are initially defined, and occasionally for emphasis.
`Monospace`	`Monospace` is used for screen messages, code listings, and command samples, as well as filenames. In code listings, *`italic monospace type`* is used for placeholder text. Code listings are colorized similar to the way they are colorized in Visual Studio. `Blue monospace type` is used for XML elements and C#/C++ keywords, `brown monospace type` is used for XML element names and C#/C++ strings, `green monospace type` is used for comments, `red monospace type` is used for XML attributes, and `teal monospace type` is used for type names in C# and C++.

Throughout this book, you'll find the following sidebar elements:

- FAQ (Frequently Asked Question) sidebars present a question readers might have regarding the subject matter in a particular spot in the book—then it provides a concise answer.
- Digging Deeper sidebars present advanced or more detailed information on a subject than is provided in the text surrounding them. Think of Digging Deeper material as stuff you can look into if you're curious, but can ignore if you're not.
- Tips are bits of information that can help you in real-world situations. They often offer shortcuts or alternative approaches to make a task easier, quicker, or produce better results.
- Warnings alert you to an action or condition that can lead to an unexpected or unpredictable result, and then tell you how to avoid it.