

Foreword

When Jeffrey approached me about writing the foreword for this book, I enthusiastically accepted. First, I remember finding the first edition of the book, which was titled *Advanced Windows NT*, in the early '90s when I started developing Windows applications. It served as both a tutorial and a reference for me as I developed applications like Filemon, Regmon, and HandleEx (now named Process Explorer). The first edition was to Windows systems programming what Charles Petzold's book, *Programming Windows*, was to user-interface programming.

What also makes writing the foreword especially rewarding for me is that Jeffrey has turned the focus of his attention to managed code in the last few years. He's become the leading authority on the .NET CLR and systems programming in C#. He embarked on his transformation as Microsoft was designing .NET, and has constantly ribbed David Solomon, my co-author for the Windows Internals series of books, and me that the sun was fading on native code programming. David and I have always held fast to the belief that native programming, with its different API, failure semantics, deployment, and run-time characteristics, will continue filling the requirements for a class of applications, or for at least a portion of them. A significant percentage of Windows client applications are still native, and even Microsoft's flagship server applications, including IIS, SQL Server, and Exchange, are written almost entirely as native programs, or have native components. I therefore see this book as Jeffrey's validation, if only unspoken, of our belief.

As Windows has evolved, so has this book. It's gone through a number of intermediate editions and even name changes. For example, the edition prior to this one was called *Programming Applications for Microsoft Windows*. Regardless of the title, however, each edition has earned a spot on my bookshelf. (It helps that Jeffrey has signed each one for me with the flattering comment of my choice, like "I learned everything from Mark.") This edition brings not only a new title, but also information on the latest APIs introduced by Windows Vista and Windows Server 2008, including threading, security, exception handling, and I/O.

What has always made this book especially useful is the fact that it's filled with programming samples that compactly and efficiently demonstrate key concepts. It's unfortunate, but many authors who share code snippets don't acknowledge that many developers will simply cut and paste the code into shipping applications. Unlike code that you might find in Web searches, Jeffrey and Christophe have ensured that the code is written to the highest standards of correctness.

I'm confident that, whether you're new to or a veteran of Windows application programming, writing fully native applications or P/Invoking APIs not provided by the .NET Framework, you'll find that this book will unlock powerful platform features and their intricacies to help you make more scalable, reliable, secure, and performant applications. It's definitely heading to my bookshelf, hopefully signed with "I learned everything from Mark" by both Jeffrey and Christophe.

Mark Russinovich
Technical Fellow
Microsoft Corporation