PRO WPF IN C# 2008:
WINDOWS PRESENTATION FOUNDATION
WITH .NET 3.5 (2ND ED.)
 Matthew MacDonald, Apress, 2008, \$54.99,
 ISBN: 1590599551.
 WPF (Windows Presentation Founda-
 tion) is a new Microsoft Windows

high-level API using DirectX as its graphics technology. Introduced with Windows Vista and .NET 3.0/3.5, WPF dramatically increases GUI possibilities, making one feel like a six-year-old who has just graduated from an eight-color box of crayons to the 120-color box.

There is an almost overwhelming amount of material presented in this book, much of it ideas and ways of thinking that will be new to many readers. Fortunately, Matthew MacDonald is an excellent guide.

To illustrate one fundamental difference from Windows Forms GUI programming, most WPF GUI programming is in XAML (Extensible Application Markup Language), which follows XML (Extensible Markup Language) rules. Although drag-and-drop tools are used to create much of XAML, the programmer can tweak it and also generate some “code behind” code. The C# code for the GUI is generated at runtime from the XAML code, usually from the BAML (Binary Application Markup Language) file, a tokenized version of the XAML code. The XAML files are used for much more than static GUI components. The possibilities are enormous, both for innovative, intuitive, and stunning interfaces and, in the wrong hands, for tasteless and horrible interfaces.

Most of this book’s 27 chapters comprise a careful trip through the new namespaces and classes introduced for WPF and focus on the important ideas, methods, classes, and their uses. Most of the code is in snippets, and much is in XAML. The complete code is available for download.

This book is part of the “Pro” series and assumes an advanced knowledge of C#. It is best suited for a skilled C# programmer who needs richer GUI capabilities or who desires better data binding and document display and more powerful printing capabilities than previously available. The book is well written with just the right amount of detail for its intended audience. I highly recommend it.

—David Naugler

EATING THE IT ELEPHANT:
MOVING FROM GREENFIELD DEVELOPMENT
TO BROWNFIELD (1ST ED.)
 Richard Hopkins and Kevin Jenkins, IBM Press,
 2008, \$29.99, ISBN: 0137130120.
 The success rate of big IT projects is 30
 percent. That number is simply unac-

ceptable, and the focus of *Eating the IT Elephant* is on helping IT organizations dramatically improve their chances of success on big—that is, elephantine—IT projects.

Regarding the book’s subtitle: greenfield development of IT projects occurs in environments where interfaces with existing systems, if they exist at all, are simple; however, most development, brownfield included, occurs within an already complex IT environment.

Complexity is the bane of huge IT projects. The book describes three types of complexity: functional, nonfunctional, and constraint. It then delves into how to build an “elephant eater” to deal with such complexity in bite-sized chunks. The architecture of an elephant eater starts with views—the perspectives of the large number of participants in a brownfield project. These formal, bounded perspectives are key components of any large project; as the number of connections among views increases, so does complexity. Authors Richard Hopkins and Kevin Jenkins describe how to deal with the three key communication issues that arise when using views—namely, inconsistency, ambiguity, and parochialism.

The authors believe that brownfield is more a change of philosophies than just a specific set of technologies. The book also covers a number of approaches to brownfield development: among them, discussions on MDA (model-driven architecture) and a comparison of Agile and waterfall development.

Anyone who is considering a large IT project, or is already involved in such a project, should seriously consider reading this book. The ideas and concepts presented could prove invaluable. No guarantee of success exists for large IT projects, but improving their chances is well worth the time investment necessary to read this book.

—David G. Hill

Reprinted from Computing Reviews, © 2008 ACM, <http://www.reviews.com>