- A description of the Java Foundation Classes (JFC) showing the patterns they illustrate
- A set of case studies where patterns have been helpful

For each pattern, we start with a brief verbal description and then build simple example programs. Each example is a visual program that you can run and examine so as to make the pattern as concrete as possible. All of the example programs and their variations are on the CD-ROM that accompanies this book. In that way, you can run them, change them, and see how the variations that you create work.

All of the programs are based on Java 1.2, and most use the JFC. If you haven't taken the time to learn how to use these classes, there is a tutorial covering the basics in Section 5 where we also discuss some of the patterns that they illustrate.

Since each of the examples consists of a number of Java files for each of the classes we use in that example, we also provide a Visual SlickEdit project file for each example and place each example in a separate subdirectory to prevent any confusion.

As you leaf through the book, you'll see screen shots of the programs we developed to illustrate the design patterns; these provide yet another way to reinforce your learning of these patterns. You'll also see UML diagrams of these programs that illustrate the interactions between classes in yet another way. UML diagrams are just simple box and arrow illustrations of classes and their inheritance structure, with the arrows pointing to parent classes and dotted arrows pointing to interfaces. If you are unfamiliar with UML, we provide a simple introduction in the first chapter.

Finally, since we used JVISION to create the UML diagrams in each chapter, we provide the original JVISION diagram files for each pattern as well, so you can use the demo version of JVISION included on the CD to play with them. We also include the free Linux version of JVISION.

When you finish this book, you'll be comfortable with the basics of design patterns and will be able to start using them in your day to day Java programming work.

James W. Cooper
Wilton, CT
Nantucket, MA
November, 1999