

# Introduction

C++ Unleashed is a survey of advanced topics in C++. The goal of this book is to provide a focused examination of each of these topics, covering the essential information you need to fully exploit the power of the C++ language.

Many of the topics in this book deserve a book in their own right. Because it is not possible, given the available space, to cover every aspect of some of these subjects, the chapters in this book explain only what is most necessary for you to gain a working understanding of the technologies they describe. Often, you will find that the information provided here is sufficient for your immediate needs. Even if that is not always the case, these chapters provide a useful foundation in these advanced issues that will allow you to quickly gain a more comprehensive understanding of them with further study.

## What Is Covered

### Part I, “Object-Oriented Programming”

We begin with a comprehensive introduction to object-oriented analysis and design. It is my view that C++ is best used to implement a well-designed object-oriented model, rather than to bang out quick-and-dirty code. The significant advantages of object-oriented programming can only be realized once you have done the necessary analysis and put the time in to design a well-conceived product. Chapter 1 will get you started on the difficult but rewarding path of object modeling. Along the way, I’ll teach you the fundamentals of the Unified Modeling Language (UML)—the emerging industry standard.

In Chapter 2, you’ll learn how to implement your object model in C++. This mapping, from design model to code, is essential if you want to use C++ to its fullest potential as an object-oriented programming language.

Chapter 3 continues this theme, focusing on how C++ supports inheritance and polymorphism. This detailed examination of the intricacies of polymorphism will lay the groundwork for creating high-quality commercial C++ applications.

### Part II, “Implementation Issues”

In Chapter 4, we discuss advanced memory management techniques. We’ll consider advanced issues with pointers and references and we’ll discuss auto pointers and smart pointers. In Chapter 5, we’ll discuss application frameworks and, within that context, we’ll consider such advanced topics as multi-threading.

Also in Part II, we'll offer an in-depth introduction to the Standard Template Library. Chapter 6 focuses on the STL container classes and Chapter 7 follows with a discussion of STL iterators and algorithms. In Chapter 8, we move on to one of the newest features of ANSI C++—namespaces—and we consider how namespaces can help you avoid name clashes as you use third-party libraries.

In Chapter 9, we focus on runtime type identification and the new ANSI-style casting operators. Finally, in Chapter 10, we'll consider how you tune your application performance to optimize for speed or code size.

## Part III, “Manipulating Data”

Part III opens Chapter 11—a discussion of advanced techniques using recursion. In Chapter 12, we discuss sorting algorithms, and in Chapter 13, we discuss object-oriented searching. This discussion is rounded out in Chapter 14 with a consideration of hashing and parsing techniques.

## Part IV, “Object Persistence and Encryption”

Chapter 15 considers object persistence and demonstrates how to write your objects to disk and how to manage memory with persistent objects. Chapter 16 returns to the application frameworks and considers ODBC and MFC Database connections. Chapter 17 extends this discussion to consider object persistence using relational databases, and Chapter 18 discusses object-oriented databases. Finally, in Chapter 19, we discuss encryption including Diffie, Hellerman, Hoffman, and Caesar ciphers; public encryption and popular encryption approaches such as Pretty Good Privacy; and DES and Clipper.

## Part V, “Distributed Computing Topics”

Chapter 20 considers CORBA, and Chapter 21 provides an in-depth introduction to COM. Finally, Chapter 22 examines the differences between Java and C++ and considers whether these differences are significant.

## What You Need To Know Already

*C++ Unleashed* assumes you have read at least one good primer (such as *Sams Teach Yourself C++ in 21 Days*) and/or have been programming in C++ for at least six months. More experienced programmers will find detail on subjects they may not have considered before; less experienced programmers will find a host of new ideas, information, and best practices.

## What Software You Need

All of the programs in this book can be created and run with Microsoft Visual C++ or any ANSI-compliant 32-bit compiler. While the example programs in the chapters on the MFC will only compile on a Windows machine (Windows 95 or Windows NT), just about all the other programs in the book will compile on any operating system.

You need no other software—just an editor, compiler, and linker. If you use an integrated development environment such as Visual C++, you are all set. While we’ve endeavored to test all the programs in this book on a number of compilers, we do know that it all works in Microsoft Visual C++, and thus we recommend that compiler if you don’t already have another.

## How To Read This Book

Think of this book as a series of “white papers” on advanced topics in C++. Feel free to jump around among the chapters, dipping into those areas which intrigue you. Again, remember that we made no attempt to be “comprehensive” on each topic; rather, our goal was to provide detailed introductions to these advanced topics. Each of these topics is the subject of one or more advanced books. Our goal here is to provide the *essential* information necessary for you to either start your further study or to obtain a quick and useful overview.

One good way to read this book is as Humpty Dumpty advised: begin at the beginning, proceed to the end, and then stop. As an alternative, you might want to read the first three chapters and then pick and choose among those topics which are of most interest to you.

In any case, enjoy and please let us know how we did. You can reach me, Jesse Liberty, on the Internet at [jliberty@libertyassociates.com](mailto:jliberty@libertyassociates.com). There is support for the book at the Sams Web site (<http://samspublishing.com>) as well as at my own Web site (<http://www.libertyassociates.com>)—click the books and resources link.