# Preface

*He who does not plow,*
*must write.*
*– Martin A. Hansen*

The ACM HOPL-2 conference on the History of Programming Languages asked me to write a paper on the history of C++. This seemed a reasonable idea and a bit of an honor, so I started writing. To get a more comprehensive and balanced view of C++'s growth, I asked a few friends from the early days of C++ for their recollections. That caused news of this project to travel through the grapevine. There, the story mutated, and one day I received a message from a friend asking where he could buy my new book on the design of C++. That email message is the real origin of this book.

Traditional books about programming and programming languages explain *what* a language is and how to use it. However, many people are also curious about *why* a language is the way it is and how it came to be that way. This book answers these last two questions for C++. It explains how C++ evolved from its first design to the language in use today. It describes the key problems, design aims, language ideas, and constraints that shaped C++, and how they changed over time.

Naturally, C++ and the ideas about design and programming that shaped it didn't just mutate by themselves. What really evolved was the C++ users' understanding of their practical problems and of the tools needed to help solve them. Consequently, this book also traces the key problems tackled using C++ and the views of the people who tackled them in ways that influenced C++.

C++ is still a young language. Some of the issues discussed here are yet unknown to many users. Many implications of decisions described here will not become obvious for years to come. This book presents my view of how C++ came about, what it is, and what it ought to be. I hope this will be of help to people trying to understand how best to use C++ and in the continuing evolution of C++.

The emphasis is on the overall design goals, practical constraints, and people that shaped C++. The key design decisions relating to language features are discussed and put into their historical context. The evolution of C++ is traced from C with Classes through Release 1.0 and 2.0 to the current ANSI/ISO standards work. The explosion of use, interest, commercial activity, compilers, tools, environments, and libraries is described. C++'s relationship to C and Simula is discussed in detail. C++'s relationship to other languages is discussed briefly. The design of major language facilities such as classes, inheritance, abstract classes, overloading, memory management, templates, exception handling, run-time type information, and namespaces are discussed in some detail.

The primary aim of this book is to give C++ programmers a better idea of the background and fundamental concepts of their language and hopefully to inspire them to experiment with ways of using C++ that are new to them. This book can also be read by experienced programmers and students of programming languages and might help them decide whether using C++ might be worth their while.

**Acknowledgments**

I am very grateful to Steve Clamage, Tony Hansen, Lorraine Juhl, Peter Juhl, Brian Kernighan, Lee Knight, Doug Lea, Doug McIlroy, Barbara Moo, Jens Palsberg, Steve Rumsby, and Christopher Skelly for reading complete drafts of this book. Their constructive comments caused major changes to the contents and organization of this book. Steve Buroff, Martin Carroll, Sean Corfield, Tom Hagelskjæ r, Rick Hollinbeck, Dennis Mancl, and Stan Lippman helped by commenting on selected chapters. Also, thanks to Archie Lachner for asking for this book before I had thought of writing it.

Naturally, I owe thanks to the many people who helped make C++. In a sense, this book is a tribute to them and some of their names can be found throughout the chapters and in the index. Should I single out individuals, it must be Brian Kernighan, Andrew Koenig, Doug McIlroy, and Jonathan Shopiro, each of whom has been a steady source of help, encouragement, and ideas for more than a decade. Also, thanks to Kristen Nygaard and Dennis Ritchie as the designers of Simula and C from which the key ingredients of C++ were borrowed. Over the years, I have come to appreciate them not only as brilliant and practical language designers, but also as gentlemen and thoroughly likable individuals.

*Murray Hill, New Jersey*                                                    *Bjarne Stroustrup*

Living languages change and grow. C++ is no exception to this rule. When this book was recently (2005) translated into Japanese, I was asked to give my perspective on C++'s futher evolution. That, together with some thoughts about where the language is today and where C++ might be going in the next ten years, can be found on the book's web page at http://www.awprofessional.com/title/0201543303.

*College Station, Texas*                                                    *Bjarne Stroustrup*