

# About This Book

---

This is a guide to the operation of the High C/C++ globally optimizing compiler targeting the PowerPC family of processors.

This guide contains all the system-specific information you need to use the compiler effectively. This initial chapter contains the following sections:

- *Summary of Contents*
- *Document Conventions*
- *Where to Go for More Information*

---

## Summary of Contents

This section provides a summary of the contents of each chapter and appendix in this document.

Chapter 1: *Introducing High C/C++* presents an overview of the C and C++ languages and the High C/C++ compiler implementation.

Chapter 2: *Using the Compiler* describes how to use the compile-and-link driver and customize the compilation process.

Chapter 3: *Using Compiler Options* describes the compiler options and lists format strings for output-file names.

Chapter 4: *Using Compiler Toggles* describes the compiler toggles (switches that turn compiler controls on and off) and how to set them.

Chapter 5: *Using Compiler Pragmas* describes the compiler pragmas and how to set them.

Chapter 6: *Optimizing Program Performance* describes the compiler's levels of optimization and the controls for activating individual optimizations.

Chapter 7: *Language Extensions* describes MetaWare High C/C++ language extensions.

Chapter 8: *C++-Specific Issues* covers use of the built-in C++ inliner, exception handling, run-time type information, templates, and other features of the C++ language and compiler.

Chapter 9: *Data Representations* provides information called for by the ANSI C Standard regarding data types.

Chapter 10: *Storage Mapping* details information needed when you interface C/C++ code with assembly code, or with code written in other languages.

Chapter 11: *PowerPC Run-Time Organization* describes the run-time organization and calling convention assumptions. This information is needed for incorporating assembly code or attempting to optimize your program's run-time efficiency.

Chapter 12: *Assembly-Language Communication* describes the conventions the compiler uses to organize C/C++ and assembly language functions.

Chapter 13: *Debugging and Diagnostic Tools* covers special features provided for tracing function calls and minimizing symbolic debug information.

Appendix A: *Generating List Files* describes the source listings that the compiler can generate and explains how to control the format of the output.

Appendix B: *Configuring the Driver* explains how to change the default values of many compiler controls set in the configuration (.cnf) file

Appendix C: *Using the Optional Inliner* describes an optional compiler pass that replaces calls to functions with the logic contained within the functions.

Appendix D: *Developing Embedded Applications* provides information to assist you in developing embedded applications for the PowerPC.

Appendix E: *Manual Template Instantiation* describes how to use compiler options and pragmas to manually control instantiation of C++ templates.

Appendix F: *The Heap Manager* describes the heap manager. This appendix applies only if you are using the MetaWare-supplied (ANSI) library.

Appendix G: *Loop Unrolling* describes an optimization called *loop unrolling*, by which the compiler replaces loops with equivalent instructions.

Appendix H: *Controlling Diagnostic Messages* describes the conditions likely to produce diagnostic messages and presents ways to control them.

---

## Document Conventions

This section describes the notational, typographical, and terminology conventions used in this document.

---

### Notational and Typographic Conventions

This manual uses several notational and typographic conventions to visually differentiate text.

Convention	Meaning
<code>Courier</code>	Program text, input, output, file names
<b><code>Courier</code></b>	Commands, keywords, literal options
<i>Courier</i>	Formal parameters to be replaced by user-specified names or values; user input on the command line
<i>Italics</i>	Special terms such as <i>Infinity</i> and <i>NAN</i> (Not a Number), which are not part of the C language or of the header files
<code>{ x   y   z }</code>	Select one and only one of the options separated by vertical bars and enclosed in curly braces
<code>[ x   y   z ]</code>	Select none, one, some, or all of the options separated by vertical bars and enclosed in brackets
<code>...</code>	Indicates multiple entries of the same type
<code> </code>	Separates choices within brackets

---

### Terminology Conventions

*“/” versus “\”* UNIX pathnames use a forward slash (“/”) to separate directories, while DOS pathnames use a backslash (“\”). This manual uses the UNIX-style forward slash when specifying pathnames.

*Word and half-word* In this manual, a data word signifies a 32-bit entity and a data half-word signifies a 16-bit entity.

---

## Where to Go for More Information

This Programmer's Guide is written for experienced software developers who have a working knowledge of the C and C++ languages and the specific processor being targeted. This section lists the documentation shipped with the High C/C++ distribution, along with other sources of information.

---

### Documents in the PowerPC Toolset

The file `readme`, in your main High C/C++ directory, identifies last-minute changes and describes any special files.

The **Installation Guide** contains instructions for installing the High C/C++ Software Development Toolset from the distribution media onto your system.

The **New Information** manual summarizes the features that have been added to the High C/C++ distribution since the last version.

The **High C/C++ Language Reference** describes the syntax and semantics of the C and C++ languages, and the MetaWare High C/C++ extensions.

The **High C Library Reference** provides descriptions of C functions.

The **High C++ I/O Streams Library Reference** provides information about C++ input and output streams.

The **ELF Assembler User's Guide** describes how to use the ELF (Executable and Linking Format) Assembler for PowerPC.

The **ELF Linker/Locator and Archiver User's Guide** describes how to use the MetaWare linker, archiver, and other ancillary utilities for object modules and libraries conforming to the Executable and Linking Format (ELF).

---

## C and C++ Programming Documents

For understanding the C language, we recommend the following texts:

Kernigan, Brian W., and Dennis M. Ritchie. **The C Programming Language**, Second Edition. Prentice Hall, Inc., Englewood Cliffs, NJ 07632, 1988.

Plauger, P. J., and J. Brodie. **Standard C**. Microsoft Press, 1989.

Harbison, Samuel P., and Guy L. Steele Jr. **C: A Reference Manual**, Second Edition. Prentice-Hall, Inc., Englewood Cliffs, NJ 07632, 1987.

For a definition of the C++ language, consult the following texts:

Stroustrup, Bjarne., and M. A. Ellis, AT&T Bell Laboratories, **Annotated C++ Reference Manual**, Addison-Wesley Publishing Company, 1990.

Stroustrup, Bjarne, AT&T Bell Laboratories, **The C++ Programming Language**, Second Edition. Addison-Wesley Publishing Company, 1991.

Stroustrup, Bjarne, AT&T Bell Laboratories, **The Design and Evolution of C++**. Addison-Wesley Publishing Company, 1994.

---

## Processor-Specific Documents

For information about a specific PowerPC processor, refer to the manufacturer's documentation.

To order documentation for PowerPC processors manufactured by IBM, contact the IBM Publications Ordering Center at (800) 879-2755.

To order documentation for PowerPC processors manufactured by Motorola Inc., contact the Motorola Inc. Semiconductor Products Sector Technical Responsiveness Center at (800) 879-2755.

---

## Specifications and ABI Documents

*DWARF* **DWARF Debugging Information Format.**

Revision: Version 1.1.0. UNIX International, Programming Languages SIG, October 6, 1992.

*IEEE* IEEE-approved standard for binary floating-point arithmetic, IEEE 754, 1985.

*PowerPC* **PowerPC Embedded Application Binary Interface.**

*Embedded ABI* Version 1.0. Motorola, 1995.

*ANSI C Standard* **Programming Language C.**

American National Standards Institute, 311 First St. NW, Suite 500, Washington DC, 20001. ANSI document X3.159, 1989.

Available from Global Engineering Documents, telephone 1-800-854-7179.

*ANSI C++  
Standard* **Working Paper for Draft Proposed International Standard for  
Information System — Programming Language C++.**

ANSI document X3J16/96-0018, 1996.

This document specifies the form and establishes the interpretation of programs expressed in the programming language C++. Its purpose is to promote portability, reliability, maintainability, and efficient execution of C++ language programs on a variety of computing systems.

*UNIX System V ABI* **System V Release 4 Application Binary Interface.**

Third Edition. UNIX System Laboratories, 1994 (ISBN 0-13-100439-5).

**System V Release 4 Application Binary Interface Supplement, PowerPC**  
Sun Microsystems, September 1995.