

C++ Standard Library: A Tutorial and Reference, The

By [Nicolai M. Josuttis](#)

Publisher : Addison Wesley

Pub Date : August 06, 1999

ISBN : 0-201-37926-0

Pages : 832

- [Table of Contents](#)

C++ Standard Library provides a set of common classes and interfaces that greatly extend the core C++ language. The library, however, is not self-explanatory. To make full use of its components-and to benefit from their power-you need a resource that does far more than list the classes and their functions.

The C++ Standard Library not only provides comprehensive documentation of each library component, it also offers clearly written explanations of complex concepts, describes the practical programming details needed for effective use, and gives example after example of working code.

This thoroughly up-to-date book reflects the newest elements of the C++ standard library incorporated into the full ANSI/ISO C++ language standard. In particular, the text focuses on the Standard Template Library (STL), examining containers, iterators, function objects, and STL algorithms. You will also find detailed coverage of special containers, strings, numerical classes, internationalization, and the IOSTream library. Each component is presented in depth, with an introduction to its purpose and design, examples, a detailed description, traps and pitfalls, and the exact signature and definition of its classes and their functions. An insightful introduction to fundamental concepts and an overview of the library will help bring newcomers quickly up to speed.

Comprehensive, detailed, readable, and practical, The C++ Standard Library is the C++ resource you will turn to again and again.

EEen

777

[Copyright](#)

[Preface](#)

[Acknowledgments](#)

[Chapter 1. About this Book](#)

[1.1 Why this Book](#)

[1.2 What You Should Know Before Reading this Book](#)

[1.3 Style and Structure of the Book](#)

[1.4 How to Read this Book](#)

[1.5 State of the Art](#)

[1.6 Example Code and Additional Information](#)

[1.7 Feedback](#)

[Chapter 2. Introduction to C++ and the Standard Library](#)

[2.1 History](#)

[2.2 New Language Features](#)

[2.3 Complexity and the Big-O Notation](#)

[Chapter 3. General Concepts](#)

[3.1 Namespace](#)

[3.2 Header Files](#)

[3.3 Error and Exception Handling](#)

[3.4 Allocators](#)

[Chapter 4. Utilities](#)

[4.1 Pairs](#)

[4.1.1 Convenience Function](#)

[4.1.2 Examples of Pair Usage](#)

[4.2 Class](#)

[4.3 Numeric Limits](#)

[4.4 Auxiliary Functions](#)

[4.5 Supplementary Comparison Operators](#)

[4.6 Header Files](#)

[Chapter 5. The Standard Template Library](#)

[5.1 STL Components](#)

[5.2 Containers](#)

[5.3 Iterators](#)

[5.4 Algorithms](#)

[5.5 Iterator Adapters](#)

[5.6 Manipulating Algorithms](#)

[5.7 User-Defined Generic Functions](#)

[5.8 Functions as Algorithm Arguments](#)

[5.9 Function Objects](#)

[5.10 Container Elements](#)

[5.11 Errors and Exceptions Inside the STL](#)

[5.12 Extending the STL](#)

[Chapter 6. STL Containers](#)

[6.1 Common Container Abilities and Operations](#)

[6.2 Vectors](#)

[6.3 Deques](#)

[6.4 Lists](#)

[6.5 Sets and Multisets](#)

[6.6 Maps and Multimaps](#)

[6.7 Other STL Containers](#)

[6.8 Implementing Reference Semantics](#)

[6.9 When to Use which Container](#)

[6.10 Container Types and Members in Detail](#)

[Chapter 7. STL Iterators](#)

[7.1 Header Files for Iterators](#)

[7.2 Iterator Categories](#)

[7.3 Auxiliary Iterator Functions](#)

[7.4 Iterator Adapters](#)

[7.5 Iterator Traits](#)

[Chapter 8. STL Function Objects](#)

[8.1 The Concept of Function Objects](#)

[8.2 Predefined Function Objects](#)

[8.3 Supplementary Composing Function Objects](#)

[Chapter 9. STL Algorithms](#)

[9.1 Algorithm Header Files](#)

[9.2 Algorithm Overview](#)

[9.3 Auxiliary Functions](#)

[9.4 The](#)

[9.5 Nonmodifying Algorithms](#)

[9.6 Modifying Algorithms](#)

[9.7 Removing Algorithms](#)

[9.8 Mutating Algorithms](#)

[9.9 Sorting Algorithms](#)

[9.10 Sorted Range Algorithms](#)

[9.11 Numeric Algorithms](#)

[Chapter 10. Special Containers](#)

[10.1 Stacks](#)

[10.2 Queues](#)

[10.3 Priority Queues](#)

[10.4 Bitsets](#)

[Chapter 11. Strings](#)

[11.1 Motivation](#)

[11.2 Description of the String Classes](#)

[11.3 String Class in Detail](#)

[Chapter 12. Numerics](#)

[12.1 Complex Numbers](#)

[12.2 Valarrays](#)

[12.3 Global Numeric Functions](#)

[Chapter 13. Input/Output Using Stream Classes](#)

[13.1 Common Background of I/O Streams](#)

[13.2 Fundamental Stream Classes and Objects](#)

[13.3 Standard Stream Operators << and >>](#)

[13.4 State of Streams](#)

[13.5 Standard Input/Output Functions](#)

[13.6 Manipulators](#)

[13.7 Formatting](#)

[13.8 Internationalization](#)

[13.9 File Access](#)

[13.10 Connecting Input and Output Streams](#)

[13.11 Stream Classes for Strings](#)

[13.12 Input/Output Operators for User-Defined Types](#)

[13.13 The Stream Buffer Classes](#)

[13.14 Performance Issues](#)

[Chapter 14. Internationalization](#)

[14.1 Different Character Encodings](#)

[14.2 The Concept of Locales](#)

[14.3 Locales in Detail](#)

[14.4 Facets in Detail](#)

[Chapter 15. Allocators](#)

[15.1 Using Allocators as an Application Programmer](#)

[15.2 Using Allocators as a Library Programmer](#)

[15.3 The Default Allocator](#)

[15.4 A User-Defined Allocator](#)

[15.5 Allocators in Detail](#)

[15.6 Utilities for Uninitialized Memory in Detail](#)

[Internet Resources](#)

[Bibliography](#)

Copyright

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book and Addison Wesley Longman Inc., was aware of a trademark claim, the designations have been printed in initial caps or all caps.

The authors and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

The publisher offers discounts on this book when ordered in quantity for special sales. For more information, please contact:

AWL Direct Sales

Addison Wesley Longman, Inc

One Jacob Way

Reading, Massachusetts 01867

(781) 944-3700

Visit AW on the Web: www.awl.com/cseng/

Library of Congress Cataloging-in-Publication Data

Josuttis, Nicolai M.

The C++ standard library: a tutorial and reference / Nicolai M.

Josuttis.

p. cm.

Includes bibliographical references and index.

1. C++ (Computer program language) I. Title.

QA76.73.C153J69 1999

005.13'3--dc21 99-24977

CIP

Copyright © 1999 by Addison Wesley Longman, Inc.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher. Printed in the United States of America. Published simultaneously in Canada.

1 2 3 4 5 6 7 8 9 -CRW- 0302010099

First printing, July 1999

Preface

In the beginning, I only planned to write a small German book (400 pages or so) about the C++ standard library. That was in 1993. Now, in 1999 you see the result — an English book with more than 800 pages of facts, figures, and examples. My goal is to describe the C++ standard library so that all (or almost all) your programming questions are answered before you think of the question. Note, however, that this is not a complete description of all aspects of the C++ standard library. Instead, I present the most important topics necessary for learning and programming in C++ by using its standard library.

Each topic is described based on the general concepts; this discussion then leads to the specific details needed to support every-day programming tasks. Specific code examples are provided to help you understand the concepts and the details.

That's it — in a nutshell. I hope you get as much pleasure from reading this book as I did from writing it. Enjoy!

Acknowledgments

This book presents ideas, concepts, solutions, and examples from many sources. In a way it does not seem fair that my name is the only name on the cover. Thus, I'd like to thank all the people and companies who helped and supported me during the past few years.

First, I'd like to thank Dietmar Kühl. Dietmar is an expert on C++, especially on input/output streams and internationalization (he implemented an I/O stream library just for fun). He not only translated major parts of this book from German to English, he also wrote sections of this book using his expertise. In addition, he provided me with invaluable feedback over the years.

Second, I'd like to thank all the reviewers and everyone else who gave me their opinion. These people endow the book with a quality it would never have had without their input. (Because the list is extensive, please forgive me for any oversight.) The reviewers for the English version of this book included Chuck Allison, Greg Comeau, James A. Crotinger, Gabriel Dos Reis, Alan Ezust, Nathan Meyers, Werner Mossner, Todd Veldhuizen, Chichiang Wan, Judy Ward, and Thomas Wikehult. The German reviewers included Ralf Boecker, Dirk Herrmann, Dietmar Kühl, Edda Lörke, Herbert Scheubner, Dominik Strasser, and Martin Weitzel. Additional input was provided by Matt Austern, Valentin Bonnard, Greg Colvin, Beman Dawes, Bill Gibbons, Lois Goldthwaite, Andrew Koenig, Steve Rumsby, Bjarne Stroustrup, and David Vandevoorde.

Special thanks to Dave Abrahams, Janet Cocker, Catherine Ohala, and Maureen Willard who reviewed and edited the whole book very carefully. Their feedback was an incredible contribution to the quality of this book.

A special thanks goes to my "personal living dictionary" — Herb Sutter — the author of the famous "Guru of the Week" (a regular series of C++ programming problems that is published on the `comp.std.C++.moderated` Internet newsgroup).

I'd also like to thank all the people and companies who gave me the opportunity to test my examples on different platforms with different compilers. Many thanks to Steve Adamczyk, Mike Anderson, and John Spicer from EDG for their great compiler and their support. It was a big help during the standardization process and the writing of this book. Many thanks to P. J. Plauger and Dinkumware, Ltd, for their early standard-conforming implementation of the C++ standard library. Many thanks to Andreas Hommel and Metrowerks for an evaluative version of their Code Warrior Programming Environment. Many thanks to all the developers of the free GNU and egcs compilers. Many thanks to Microsoft for an evaluative version of Visual C++. Many thanks to Roland Hartinger from Siemens Nixdorf Informations Systems AG for a test version of their C++ compiler. Many thanks to Topjects GmbH for an evaluative version of the ObjectSpace library implementation.

Many thanks to everyone from Addison Wesley Longman who worked with me. Among others this includes Janet Cocker, Mike Hendrickson, Debbie Lafferty, Marina Lang, Chanda Leary, Catherine Ohala, Marty Rabinowitz, Susanne Spitzer, and Maureen Willard. It was fun.

In addition, I'd like to thank the people at BREDEX GmbH and all the people in the C++ community, particularly those involved with the standardization process, for their support and patience (sometimes I ask really silly questions).

Last but not least, many thanks and kisses for my family: Ulli, Lucas, Anica, and Frederic. I definitely did not have enough time for them due to the writing of this book.

Have fun and be human!