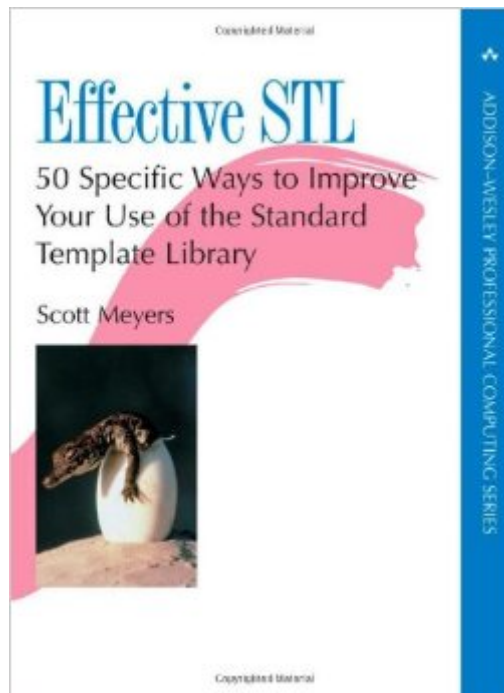


The book was found

Effective STL: 50 Specific Ways To Improve Your Use Of The Standard Template Library



Synopsis

“This is Effective C++ volume three – it’s really that good.” – Herb Sutter, independent consultant and secretary of the ISO/ANSI C++ standards committee

“There are very few books which all C++ programmers must have. Add Effective STL to that list.” – Thomas Becker, Senior Software Engineer, Zephyr Associates, Inc., and columnist, C/C++ Users Journal

C++’s Standard Template Library is revolutionary, but learning to use it well has always been a challenge. Until now. In this book, best-selling author Scott Meyers (*Effective C++* , and *More Effective C++*) reveals the critical rules of thumb employed by the experts – the things they almost always do or almost always avoid doing – to get the most out of the library. Other books describe what’s in the STL. *Effective STL* shows you how to use it. Each of the book’s 50 guidelines is backed by Meyers’ legendary analysis and incisive examples, so you’ll learn not only what to do, but also when to do it – and why. Highlights of *Effective STL* include: Advice on choosing among standard STL containers (like `vector` and `list`), nonstandard STL containers (like `hash_set` and `hash_map`), and non-STL containers (like `bitset`). Techniques to maximize the efficiency of the STL and the programs that use it. Insights into the behavior of iterators, function objects, and allocators, including things you should not do. Guidance for the proper use of algorithms and member functions whose names are the same (e.g., `find`), but whose actions differ in subtle (but important) ways. Discussions of potential portability problems, including straightforward ways to avoid them. Like Meyers’ previous books, *Effective STL* is filled with proven wisdom that comes only from experience. Its clear, concise, penetrating style makes it an essential resource for every STL programmer.

Book Information

Paperback: 288 pages

Publisher: Addison-Wesley Professional; 1 edition (June 16, 2001)

Language: English

ISBN-10: 0201749629

ISBN-13: 978-0201749625

Product Dimensions: 7.3 x 0.9 x 9 inches

Shipping Weight: 1.3 pounds (View shipping rates and policies)

Average Customer Review: 4.5 out of 5 stars – See all reviews (52 customer reviews)

Best Sellers Rank: #131,739 in Books (See Top 100 in Books) #77 in Books > Computers & Technology > Programming > Microsoft Programming > C & C++ Windows Programming #91

Customer Reviews

There we go - with his well-known sharpness and diligence, Scott absorbed STL in all detail, taught it in seminars, chewed on the ensuing experience, and distilled it all in this book. I was one of the reviewers and in the beginning I thought that reviewing a book on STL is going to be an easy enough task. I was wrong. I learned lots of new things on using STL effectively: why ``empty()`` is better than ``size() == 0``, when and how to write custom allocators, how `std::string` might be implemented, how associative containers distinguish between equality and equivalence, how to implement associative containers as sorted vectors (that's a gem!), and many, many other things that I either had a blurry understanding of, or simply didn't know about. Now I'm glad I do, because my understanding of the STL and the practical use of it are much better. The book went through an extensive review process; it is really combed and distilled to its finest. I recommend it to any C++ programmer who uses STL - which should be, any C++ programmer, period. Five well deserved stars.

I'm a professional software engineer. I write code all day long and have lots of experience with C++, but I hadn't used STL much until recently. If you're in a similar situation--decent C++ knowledge but not an STL expert--this book is for you. I haven't even read the whole thing yet, and already I am using patterns from the book to write more effective code. Before I started this book, I thought STL was kind of neat. It had some useful containers. It was nice to be able to use a list or map or string class that had already been tested. Boy, was I underestimating the power of STL. This book has made me a big STL fan, but I'm not reviewing the STL now so I'll leave that topic alone... Thanks to Scott Meyers, I have a much better grasp of the capabilities and limitations of STL. I can use it to do a lot more. I write more concise code that's easier to read and debug. I make better choices about which containers to use. I recognize situations where I can use an STL algorithm instead of many lines of my own code. In short, I look at the STL code I wrote before and laugh... I mean, it all works, but the Meyers book has taken my use of (and appreciation for) the STL to a whole new level. I recommend this book for any C++ developer who isn't already an STL expert. An update, 2 years after the above text was written: I still recommend this book to people and still think it's the best STL book I've read.

This is a truly useful book. It explains lots of little "gotchas" that I didn't know about previously, and Scott does his usual excellent job at explaining *why* it's important to do things a certain way (and no other). One part that I found particularly interesting is about the futility of writing container-independent code; not only does that section show why this is a bad idea, it also serves as a splendid illustration of the idiosyncracies of the various containers. The chapter on iterators is priceless, as are the tips about writing comprehensible code and debugging. The presentation is very much up-to-date (even to the point where it anticipates some of the forthcoming updates to the C++ standard). The writing style is clear and precise without sounding academic or condescending, and the book has an index that actually works. "Effective STL" is every bit as good as "Effective C++" and "More Effective C++". No C++ programmer should still be writing code without the STL, and no-one writing code with the STL should be without this book. Buy it!

I'm a great fan of Meyer's Effective C++, but this book is not in that league. Maybe it was a good book when it came out (in 2001!), but the C++ compilers and the STL have moved on --- but the glowing reviews on this page date from 2001. There was no need for workarounds for broken compilers (e.g. the section on containers of `auto_ptr`) in 2011, and with first TR1 and then C++11 out this book is seriously out-of-date -- no-one uses (or should use) `ptr_fun` and friends any more. Come on Scott, write a second edition!

I was very pleased to see this book come out as I am a huge fan of Meyers' Effective C++ and More Effective C++. Both of those books offer many concrete suggestions on how to improve your C++ coding and do things you didn't think you could do. Effective STL, on the other hand, offers mainly suggestions of what not to do. You'll find most of the items tell you that you should not use a specific technique because it is not portable, not a clean design, or simply because it will not work. All of this is good to know, but I didn't find it as useful as his other two Effective books. Other than the use of the built in looping functions like `for_each` the book didn't really provide me with many new "tricks" for my "bag." However, all the information in the book is useful, and the intermediate level STL programmer will probably learn a lot of ways to avoid problems by reading this book. Hopefully there will be a second edition in the future that will give the book a little more utility.

[Download to continue reading...](#)

Effective STL: 50 Specific Ways to Improve Your Use of the Standard Template Library Data Structures in C++: Using the Standard Template Library (STL) Effective Objective-C 2.0: 52 Specific Ways to Improve Your iOS and OS X Programs (Effective Software Development Series) Effective

Modern C++: 42 Specific Ways to Improve Your Use of C++11 and C++14 Effective C++: 55 Specific Ways to Improve Your Programs and Designs (3rd Edition) Effective C++: 55 Specific Ways to Improve Your Programs and Designs Improve Your Eyesight Naturally: How To Improve Your Vision Naturally - Learn Super Effective Eyesight Exercises To Improve Eyesight Without (Vision Therapy, Optometry, Eyesight Improvement) Effective TCP/IP Programming: 44 Tips to Improve Your Network Programs: 44 Tips to Improve Your Network Programs Effective COM: 50 Ways to Improve Your COM and MTS-based Applications Data Structures with C++ Using STL (2nd Edition) Brain Training: Power Brain! - Secret Techniques To: Improve Memory, Focus & Concentration (Brain teasers, Improve memory, Improve focus, Concentration, Brain power) Effective Perl Programming: Ways to Write Better, More Idiomatic Perl (Effective Software Development Series) Node JS: Learn Node In The Most Simple and Effective Way Possible. Learn to use Node in several ways.: With Events, Async Functions, API calls, Push Notifications, Command Line Input, Web Server The NEW Knitter's Template: Your Guide to Custom Fit and Style How to Use Tea Tree Oil - 90 Great Ways to Use Natures "Medicine Cabinet in a Bottle" - Acne, Boils, Head Lice, Nail Fungus, Ringworm, Skin Tags, - Health ... Dilutions and Lots More! (What Is? Book 2) Yes, You Can Get Pregnant: Natural Ways to Improve Your Fertility Now and into Your 40s Linux PCI Device Driver - A Template (Linux Driver Development) Linux Char Device Driver - A Template (Linux Driver Development) Weekly Teacher Planner Template: Undated Lesson Plan Book For Teachers. 40 weeks,5 Day View 7 Periods, With Classroom Management, Goals, Substitute Teacher Info & Record Pages (Teaching Resources) Blank Grade Book Template: Record Notebook/Pad For Teachers. With Attendance Sheets, Grading Sheets And More. 35 Names. 8.5in by 11in. 102 Pages (Teaching Resources)

[Dmca](#)