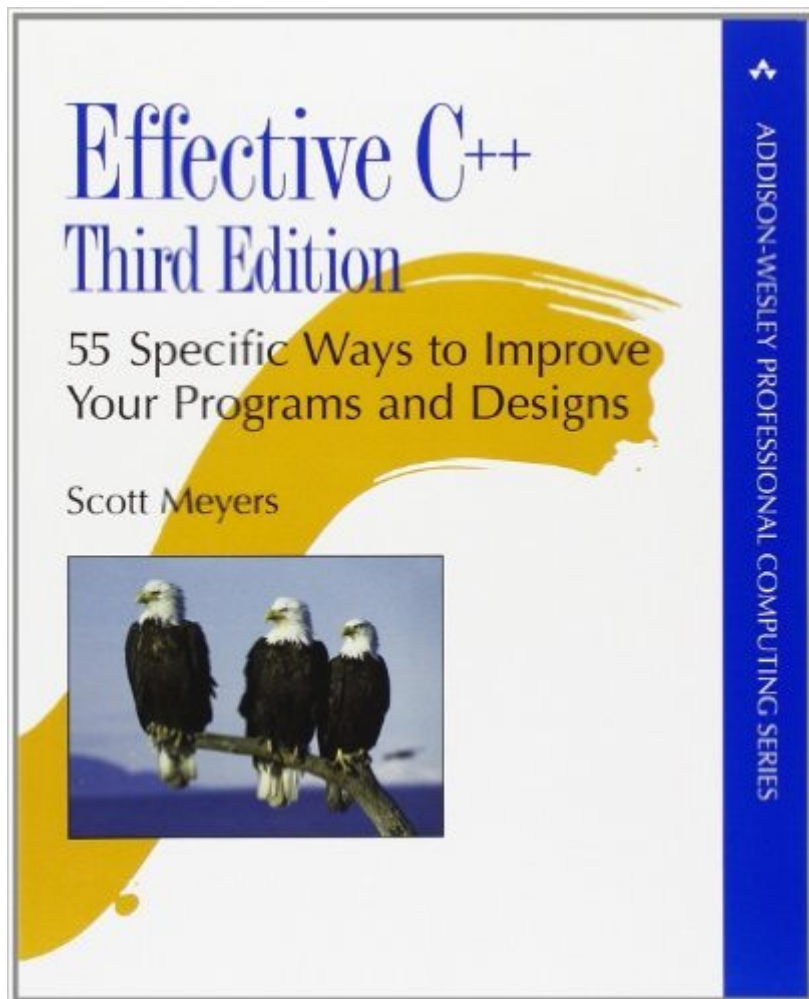


The book was found

Effective C++: 55 Specific Ways To Improve Your Programs And Designs (3rd Edition)



Synopsis

“Every C++ professional needs a copy of Effective C++. It is an absolute must-read for anyone thinking of doing serious C++ development. If you’ve never read Effective C++ and you think you know everything about C++, think again.” Steve Schirripa, Software Engineer, Google

“C++ and the C++ community have grown up in the last fifteen years, and the third edition of Effective C++ reflects this. The clear and precise style of the book is evidence of Scott’s deep insight and distinctive ability to impart knowledge.” Gerhard Kreuzer, Research and Development Engineer, Siemens AG

The first two editions of Effective C++ were embraced by hundreds of thousands of programmers worldwide. The reason is clear: Scott Meyers’ practical approach to C++ describes the rules of thumb used by the experts – the things they almost always do or almost always avoid doing – to produce clear, correct, efficient code. The book is organized around 55 specific guidelines, each of which describes a way to write better C++. Each is backed by concrete examples. For this third edition, more than half the content is new, including added chapters on managing resources and using templates. Topics from the second edition have been extensively revised to reflect modern design considerations, including exceptions, design patterns, and multithreading. Important features of Effective C++ include: Expert guidance on the design of effective classes, functions, templates, and inheritance hierarchies. Applications of new C++ standard library functionality, along with comparisons to existing standard library components. Insights into differences between C++ and other languages (e.g., Java, C#, C) that help developers from those languages assimilate the C++ way of doing things.

Book Information

Paperback: 320 pages

Publisher: Addison-Wesley Professional; 3 edition (May 22, 2005)

Language: English

ISBN-10: 0321334876

ISBN-13: 978-0321334879

Product Dimensions: 7.3 x 0.8 x 9.1 inches

Shipping Weight: 1.4 pounds (View shipping rates and policies)

Average Customer Review: 4.6 out of 5 stars – See all reviews (125 customer reviews)

Best Sellers Rank: #32,400 in Books (See Top 100 in Books) #10 in Books > Computers & Technology > Programming > Microsoft Programming > C & C++ Windows Programming #13 in Books > Computers & Technology > Programming > Languages & Tools > C & C++ > C++

Customer Reviews

How do you improve a book that has been known as a "bible" in the C++ community for years? As Scott mentions at the beginning of the text, he almost threw everything out, and started from scratch. When I first saw the book, I thought that there are five new items that were added to the book, but I couldn't be more wrong. As you might have guessed, C++ has gone thru a number of significant changes over the past decade, and the third edition of this book is updated to take advantage of the new editions to the C++ standard. In reality, almost every item in this book has gone thru a re-write. Many have been consolidated and new chapters, topics and many new items have been added. A few items that did not make sense anymore like items 2, 3 and 4 in the second edition are removed from this third edition. Scott breaks down the C++ language into 4 subparts:

- * The old C subsystem. Before all these advanced programming languages such as Java and .NET came, C was the language of choice. C++ is "translated" to C first, and then compiled and linked to an executable.
- * OO C++, which is C with Classes. This is where the concept of Object Orientation in C++ started. Even though this concept was very much new a decade ago, it is very much part of a programmer's vocabulary.
- * Template C++, which is the newest edition to the C++ standard and it brings with it the concept of Template Metaprogramming. This concept is very much new, and this book has dedicated a whole chapter around templates, and template metaprogramming.
- * STL, which is the C++ Standard Template Library. Again, STL was a new concept a few years back, but it is very much an established notion in C++.

"Effective C++: 55 Specific Ways to Improve Your Programs and Designs" recently came out in its 3rd edition. The first came out in 1991, making this line of book revisions positively ancient in high tech terms. But you should consider the elapsed 14 years an investment in wisdom, not obsolescence. Unlike the 2nd edition, this is a complete rewrite from the ground up. More than that, Scott Meyers begins from first principles, reevaluating what topics are most important to the programmers of 2005. This was a very strategic and insightful move. Not only has the language evolved over that time, but the audience has too. In the first edition, the likely readers were coming from languages like C. Now the likely readers cross over from some other object-oriented language, like Java or C#. Keep in mind that the author focuses strictly on standard C++; he does not address anything that is platform-specific. Like its predecessors, the items can be read stand-alone, and he does cross reference related items. The 55 items are grouped into the following chapters: Chapter 1:

Accustoming Yourself to C++Chapter 2: Constructors, Destructors, and Assignment
OperatorsChapter 3: Resource ManagementChapter 4: Designs and DeclarationsChapter 5:
InheritanceChapter 6: Implementations and Object-Oriented DesignChapter 7: Templates and
Generic ProgrammingChapter 8: Customizing new and deleteChapter 9: MiscellanyI would hate to
have to learn the lessons in this book by hard experience. It would be costly in time and
unfortunately might even cover more than one employment span.

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

Effective C++: 55 Specific Ways to Improve Your Programs and Designs (3rd Edition) Effective
C++: 55 Specific Ways to Improve Your Programs and Designs Effective Objective-C 2.0: 52
Specific Ways to Improve Your iOS and OS X Programs (Effective Software Development Series)
Effective TCP/IP Programming: 44 Tips to Improve Your Network Programs: 44 Tips to Improve
Your Network Programs Effective Modern C++: 42 Specific Ways to Improve Your Use of C++11
and C++14 Effective STL: 50 Specific Ways to Improve Your Use of the Standard Template Library
Improve Your Eyesight Naturally: How To Improve Your Vision Naturally - Learn Super Effective
Eyesight Exercises To Improve Eyesight Without (Vision Therapy, Optometry, Eyesight
Improvement) School-Based Mental Health Services: Creating Comprehensive and Culturally
Specific Programs (Applying Psychology to the Schools) Effective COM: 50 Ways to Improve Your
COM and MTS-based Applications 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) Brooks/Cole Empowerment Series: Social Welfare Policy and Social
Programs (SW 323K Social Welfare Programs, Policies, and Issues) Insider's Guide to Graduate
Programs in Clinical and Counseling Psychology (Insider's Guide to Graduate Programs in Clinical
& Counseling Psychology) Graduate Programs in Business, Education, Information Studies, Law &
Social Work 2017 (Peterson's Graduate Programs in Business, Education, Health, Information
Studies, Law and Social Work) Functional Programming in JavaScript: How to improve your
JavaScript programs using functional techniques Yes, You Can Get Pregnant: Natural Ways to
Improve Your Fertility Now and into Your 40s Only Skin Deep?: An Essential Guide to Effective Skin
Cancer Programs and Resources How Reliable Is Your Product? (Second Edition): 50 Ways to
Improve Product Reliability Home Security Basics: The easiest and most affordable ways to improve
your home security. The Fertility Diet: Groundbreaking Research Reveals Natural Ways to Boost
Ovulation and Improve Your Chances of Getting Pregnant

[Dmca](#)