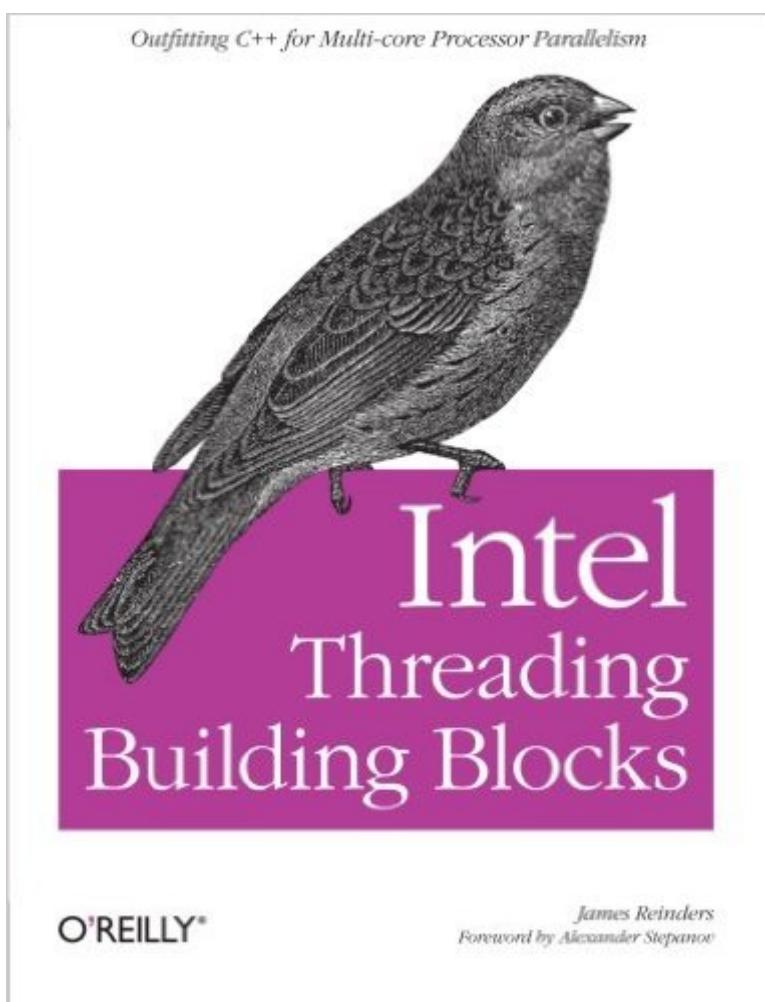


The book was found

Intel Threading Building Blocks: Outfitting C++ For Multi-core Processor Parallelism



Synopsis

Multi-core chips from Intel and AMD offer a dramatic boost in speed and responsiveness, and plenty of opportunities for multiprocessing on ordinary desktop computers. But they also present a challenge: More than ever, multithreading is a requirement for good performance. This guide explains how to maximize the benefits of these processors through a portable C++ library that works on Windows, Linux, Macintosh, and Unix systems. With it, you'll learn how to use Intel Threading Building Blocks (TBB) effectively for parallel programming -- without having to be a threading expert. Written by James Reinders, Chief Evangelist of Intel Software Products, and based on the experience of Intel's developers and customers, this book explains the key tasks in multithreading and how to accomplish them with TBB in a portable and robust manner. With plenty of examples and full reference material, the book lays out common patterns of use, reveals the gotchas in TBB, and gives important guidelines for choosing among alternatives in order to get the best performance. You'll learn how Intel Threading Building Blocks: Enables you to specify tasks instead of threads for better portability, easier programming, more understandable source code, and better performance and scalability in general. Focuses on the goal of parallelizing computationally intensive work to deliver high-level solutions. Is compatible with other threading packages, and doesn't force you to pick one package for your entire program. Emphasizes scalable, data-parallel programming, which allows program performance to increase as you add processors. Relies on generic programming, which enables you to write the best possible algorithms with the fewest constraints. Any C++ programmer who wants to write an application to run on a multi-core system will benefit from this book. TBB is also very approachable for a C programmer or a C++ programmer without much experience with templates. Best of all, you don't need experience with parallel programming or multi-core processors to use this book.

Book Information

Paperback: 336 pages

Publisher: O'Reilly Media; 1 edition (July 22, 2007)

Language: English

ISBN-10: 0596514808

ISBN-13: 978-0596514808

Product Dimensions: 7 x 0.7 x 9.2 inches

Shipping Weight: 1.1 pounds (View shipping rates and policies)

Average Customer Review: 3.8 out of 5 stars [See all reviews](#) (6 customer reviews)

Best Sellers Rank: #713,648 in Books (See Top 100 in Books) #25 in Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Performance Optimization #71 in Books > Computers & Technology > Programming > Parallel Programming #415 in Books > Computers & Technology > Programming > Languages & Tools > C & C++ > C++

Customer Reviews

There are many views of how best to implement multi-threading and with multi-core processors now common the topic is becoming ever more relevant. Of course the best implementation method really depends on what you are trying to do. Are you implementing something that waits for a price update message from a derivatives exchange, parses the message, updates cached data, writes a log and forwards the message on to an algorithmic trading engine? If so, you'll probably be attracted to techniques different from someone calculating risk, bond prices etc. There is no one technique that suits every job. Anyone working with multi-threading (and parallelism in general) keeps an eye on what techniques other people are using to see whether they may be useful. That's why I bought this book (Intel Threading Building Blocks) and TBB certainly looks like it will be of use for some multi-threaded applications that I develop, although not for all. Starting with the most important question - have I learnt anything from reading this book. Yes, and I've learnt a couple of bits from reading the TBB source code too (downloadable from the web). Now for the detail... Having read the book, am I now about to start using the Intel Threading Building Blocks library (downloadable from Intel)? The answer is yes, for some applications on some hardware architectures, but not for all of my multi-threaded applications. Importantly in terms of this book though, the first reason I looked at the TBB source code was to answer questions raised when reading the book, and that is my main issue with this book. For a developer who already has extensive multi-threading experience, this book raises quite a few questions that it doesn't answer.

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

Intel Threading Building Blocks: Outfitting C++ for Multi-core Processor Parallelism Food Processor Greats: Delicious Food Processor Recipes, The Top 100 Food Processor Recipes High Performance Parallelism Pearls Volume One: Multicore and Many-core Programming Approaches Tupac Shakur: Multi-platinum Rapper: Multi-Platinum Rapper (Lives Cut Short) Efficient Android Threading: Asynchronous Processing Techniques for Android Applications Block Print: Everything you need to know for printing with lino blocks, rubber blocks, foam sheets, and stamp sets Guaranteed to Last: L.L. Bean's Century of Outfitting America Victory in Spiritual Warfare: Outfitting Yourself for the Battle Home Automation with Intel Galileo Introduction to 64 Bit Intel Assembly

Language Programming for Linux: Second Edition Intel Galileo Essentials Assembly Language for Intel-Based Computers (3rd Edition) Assembly Language for Intel-Based Computers (4th Edition) Parallel Programming with Intel Parallel Studio XE Software Optimization Cookbook: High-Performance Recipes for the Intel Architecture Intel Trinity, The: How Robert Noyce, Gordon Moore, and Andy Grove Built the World's Most Important Company Building a Multi-Use Barn: For Garage, Animals, Workshop, or Studio Building Blocks, The Whitney Museum of American Art SCIENCE EXPLORER C2009 BOOK K STUDENT EDITION CHEMICAL BUILDING BLOCKS (Prentice Hall Science Explore) Sand and Soil: Earth's Building Blocks (Rocks, Minerals, and Resources)

[Dmca](#)