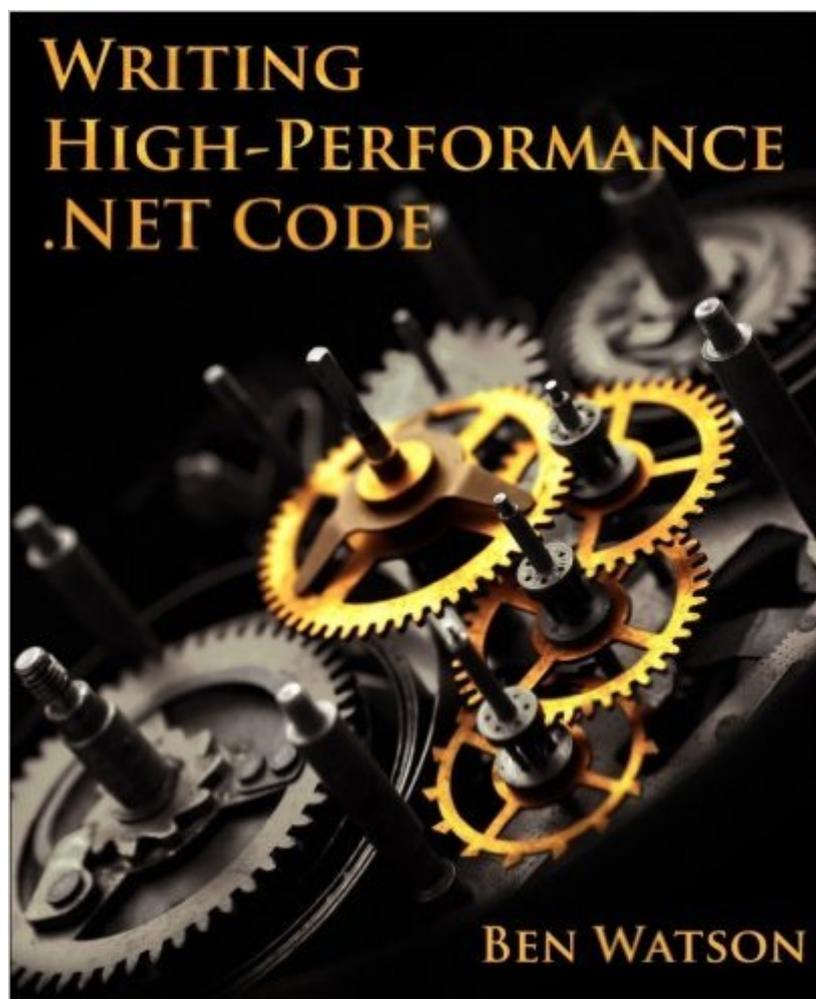The book was found

Writing High-Performance .NET Code

Synopsis

Do you want your .NET code to have the absolute best performance it can? This book demystifies the CLR, teaching you how and why to write code with optimum performance. Learn critical lessons from a person who helped design and build one of the largest high-performance .NET systems in the world. This book does not just teach you how the CLR worksâ "it teaches you exactly what you need to do now to obtain the best performance today. It will expertly guide you through the nuts and bolts of extreme performance optimization in .NET, complete with in-depth examinations of CLR functionality, free tool recommendations and tutorials, useful anecdotes, and step-by-step guides to measure and improve performance. Among the topics you will learn are how to: Choose what to measure and why Use many amazing tools, freely available, to solve problems quickly Understand the .NET garbage collector and its effect on your application Use effective coding patterns that lead to optimal garbage collection performance Diagnose common GC-related issues Reduce costs of JITting Use multiple threads sanely and effectively, avoiding synchronization problems Know which .NET features and APIs to use and which to avoid Use code generation to avoid performance problems Measure everything and expose hidden performance issues Instrument your program with performance counters and ETW events Use the latest and greatest .NET features Ensure your code can run on mobile devices without problems Build a performance-minded team â |and much more.

Book Information

Paperback: 280 pages

Publisher: Ben Watson (July 23, 2014)

Language: English

ISBN-10: 0990583430

ISBN-13: 978-0990583431

Product Dimensions: 7.5 x 0.6 x 9.2 inches

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

Average Customer Review: 4.7 out of 5 starsÂ Â See all reviewsÂ (47 customer reviews)

Best Sellers Rank: #79,696 in Books (See Top 100 in Books)   #15 inÂ Books > Computers & Technology > Programming > Microsoft Programming > .NET   #36 inÂ Books > Computers & Technology > Programming > Microsoft Programming > C & C++ Windows Programming   #24512 inÂ Books > Reference

Customer Reviews

Note: I received an advanced review copy of this book.I first want to comment on the readability of

this book. Most technical references read like an encyclopedia that make you glassy eyed after a few paragraphs. If you ever read "The Soul of a New Machine" and liked it (which you probably did), you'll really appreciate how this book reads. It's engaging, and reads like a technical memoirs of the lessons learned from the genesis of a very large scale .NET service. Readability is great, but the technical content is invaluable.I personally felt that the chapter on GC alone is easily worth the price of the book and the time to read it. It doesn't dive into an exhaustive summary of how the CLR works, but it clearly draws out the profoundly important things you need to understand about how to best manage memory and address GC in the CLR. The takeaway here is memory management is completely controllable. This is knowledge that someone will eventually have to acquire over time when working on large scale systems, so it's an excellent resource to fully understand what you need to know about GC.The book is also peppered with very valuable code snippets and downloadable content with debugging instructions to understand the concepts in this book. Overall, this book does a very good job and breaking down the myths of poor performing managed code, and the optimal strengths managed code has over native.I gave this book 5 stars for several reasons. First, it delivers on the content and readability the book is selling. It's not trying to be an exhaustive summary of .NET or the CLR. The price is also very generous for the content you're getting. Rather it is teaching you some of the most valuable lessons, tips, and resources to writing high performance .NET application. Finally, it reads very quickly. I found myself finishing this book much faster than most computer science books of similar length.

This is not a beginner book; in fact it's almost the archetype of an "advanced" book. But the best advice is to test and measure so you don't have to blindly take the word of the author. This is a great book if you want and need to really understand performance. But like all good lessons except to actually do the work in the book. It will put you at a new level of expertise.

I have a trading system that I develop for a few years already, and this book make a good help for improve some things in the code base. And I am still have the fastest system at the Russian Moscow exchange.I thought that I know everything about the performance in .NET, but this book has proven that I don't.

This is one of the best books in dotnet I've read in the past few years. I'm always been centered in performance programming, and still I could found lots of new info here. For example the windbg stuff (I've never ever ventured myself into the use of the dreadful windbg until now... and i'm

learning a lot about my processes...), or some neat tricks with the ETWs I didn't know about...Being a dotnet programmer for almost 13 years now, I really recommend this book to all advanced and intermediate programmers out there. You want to be serious about dotnet programming? you should be aware of the stuff this book talks about.

This book is on the shortlist of books good enough to change the way I think about programming. I've had optimization of code as a primary or secondary task of my job for at least 8 years, and worked with .Net since pre Beta (ie PDC), but I learned a lot of valuable strategies and ways to think about performance that have me revisiting code I thought I'd squeezed the last drops of optimization out of and laughing at my naivete. Good stuff -- if you want performant code as part of your .Net programming experience, you'll be hard pressed to do better than this book.

Note: I received a review copy of this bookThis book is a must have if you aim to develop any serious .NET-based software that is performant and scalable. The two best things about this book are:1) The clear explanation of how the GC works and how seemingly innocuous implementation decisions can seriously affect your performance2) How to use the freely available performance monitoring tools to debug hot spotsI particularly enjoyed the detailed explanations of the free performance monitoring tools and how my team can leverage ETW to get granular insight into how our .NET services are performing. Before reading this I believed we needed to buy expensive tools from Jet Brains and wasn't even aware we had other options!One thing I would've liked to see in this book is an acknowledgement that designing for performance and building monitoring into the first generation of a product is a tough sell to management. It also gets in the way of quickly iterating on ideas that you are turning into products. How would the author recommend we get over this, and how would he recommend we build such capabilities into the existing product after its initial version? I'm sure he has quite a bit of experience in his time at Bing.Regardless, this book is a must have for any .NET developer worth his or her salt.

Great book filled with insight learned over many years of detailed analysis of .NET code and performance. HIGHLY recommended if you're interested in learning how to measure, analyze and improve the performance of your .NET apps.

Download to continue reading...

Writing High-Performance .NET Code MCAD/MCSD Self-Paced Training Kit: Developing Windows®-Based Applications with Microsoft® Visual Basic® .NET and Microsoft Visual C#®

.NET, Second Ed: ... C#(r) .Net, Second Ed (Pro-Certification) Writing: A Guide Revealing The Best Ways To Make Money Writing (Writing, Writing Skills, Writing Prompts Book 1) Database Programming with Visual Basic .NET and ADO.NET: Tips, Tutorials, and Code 2012 International Plumbing Code (Includes International Private Sewage Disposal Code) (International Code Council Series) ATL Server: High Performance C++ on .NET ASP.NET Core 1.0 High Performance Writing Romance: The Top 100 Best Strategies For Writing Romance Stories (How To Write Romance Novels, Romance Writing Skills, Writing Romance Fiction Plots, Publishing Romance Books) The .NET and COM Interoperability Handbook (Integrated .Net) MCPD Self-Paced Training Kit (Exams 70-536, 70-528, 70-547): Microsoft® .NET Framework Web Developer Core Requirements: Microsoft .Net Framework Web ... Requirements (Microsoft Press Training Kit) Building Web Solutions with ASP.Net and ADO.NET (Developer Reference) ASP.NET For Beginners: The Simple Guide to Learning ASP.NET Web Programming Fast! Pro ASP.NET MVC Framework (Expert's Voice in .NET) Beginning ASP.NET E-Commerce in C#: From Novice to Professional (Expert's Voice in .NET) Programming ASP.NET MVC 4: Developing Real-World Web Applications with ASP.NET MVC Pro Microsoft Speech Server 2007: Developing Speech Enabled Applications with .NET (Expert's Voice in .NET) Ford Total Performance: Ford's Legendary High-Performance Street and Race Cars Sentences and Paragraphs: Mastering the Two Most Important Units of Writing (The Writing Code Series Book 8) High Commitment High Performance: How to Build A Resilient Organization for Sustained Advantage Gardening For Entrepreneurs: Gardening Techniques For High Yield, High Profit Crops (Farming For Profit, Gardening For Profit, High Yield Gardening)