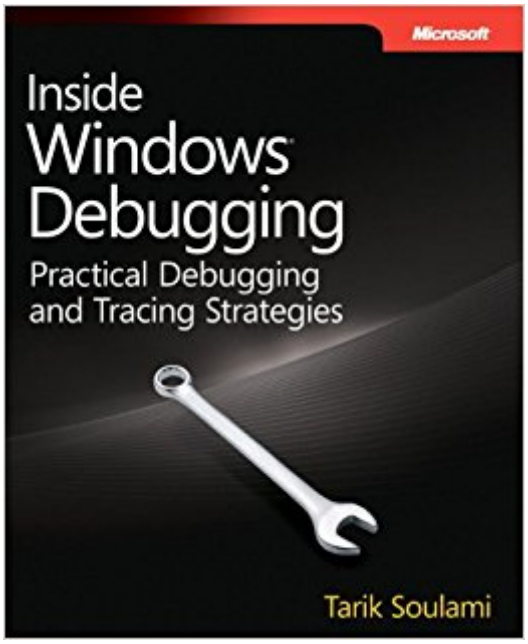
Inside Windows Debugging (Developer Reference)



Synopsis

Use Windows debuggers throughout the development cycleÃ¢â ¬â •and build better software Rethink your use of Windows debugging and tracing toolsÃ¢â ¬â •and learn how to make them a key part of test-driven software development. Led by a member of the Windows Fundamentals Team at Microsoft, youÃ¢â ¬â„¢ll apply expert debugging and tracing techniquesÃ¢â ¬â •and sharpen your C++ and C# code analysis skillsÃ¢â ¬â •through practical examples and common scenarios. Learn why experienced developers use debuggers in every step of the development process, and not just when bugs appear. Discover how to: Go behind the scenes to examine how powerful Windows debuggers work Catch bugs early in the development cycle with static and runtime analysis tools Gain practical strategies to tackle the most common code defects Apply expert tricks to handle user-mode and kernel-mode debugging tasks Implement postmortem techniques such as JIT and dump debugging Debug the concurrency and security aspects of your software Use debuggers to analyze interactions between your code and the operating system Analyze software behavior with Xperf and the Event Tracing for Windows (ETW) framework

Book Information

Series: Developer Reference

Paperback: 592 pages

Publisher: Microsoft Press; 1 edition (May 25, 2012)

Language: English

ISBN-10: 0735662789

ISBN-13: 978-0735662780

Product Dimensions: 7.3 x 1.2 x 8.8 inches

Shipping Weight: 2 pounds (View shipping rates and policies)

Average Customer Review: 4.5 out of 5 stars 9 customer reviews

Best Sellers Rank: #447,395 in Books (See Top 100 in Books) #21 inÃ Â Books > Computers & Technology > Programming > Languages & Tools > Compiler Design #22 inÃ Â Books > Computers & Technology > Programming > Languages & Tools > Debugging #40 inÃ Â Books > Computers & Technology > Programming > Parallel Programming

Customer Reviews

Tarik Soulami is a principal development lead on the Windows Fundamentals Team at Microsoft.

A Programmers toolset is important, and what’s more important is knowning how to use the toolset.

Windows Debugging, and debugging in general was not something that I every learned in school. The windows Debugger is powerful but some complain there is a learning curve to it. The author takes you through very practical real world problems, with accompanying examples to teach the reader how to not only use Windows debugging tools mainly windbg, but also to teach the reader how to debug.Overall excellent book, well worth adding to the library.

Since most of the core/kernel elements of Windows haven’t changed since NT in the late 80’s, most of the "new" stuff is in the form of API’s. Soulami assumes a basic working knowledge of C/C++ or C#, but doesn’t start at such a high level that you get lost in either the debugger or the tracer. This book is REALLY up to date on windows, and will catch you up even if you are still working on an NT base.Chapters include: 1. How to develop software for Windows 2. Getting started (debugging for fun and profit section) 3. How debuggers work (pretty basic but very complete, covers both User and Kernel modes) 4. Postmortem Debugging (JIT vs. dump techniques. Goes much deeper than the day to day systems engineer will usually go) 5. Beyond the Basics (the real meat of the book-- awesome-- data vs. code breakpoints, scripts, etc.) 6. Code analysis tools (fair to C/++ and sharp, with many actual/not just pseudo/ code examples that are well thought out and RUN); 7. Expert Debugging Tricks (we finally get to the fun and profit piece-- many techniques that are effective but unusual, and probably wouldn’t be attempted by the usual coder without this book’s help on avoiding potholes); 8 and 9 are a whole collection of very cool "scenarios" covering all the NIGHTMARES created by threads and multiprocessors such as race conditions, deadlocks, stack/heap and access problems, etc. These two chapters are worth the price of the whole book; 10 gets into the console subsystem and concludes this section.Section two (about 120 pages) switches themes with three chapters about Xperf. In short, if you try to run traces as you develop your software using just ETW (event tracing for Windows), you’ll soon get overwhelmed and give it up. This means you’re losing one of the best "secret sauces" of the Windows 7 SDK (a way to integrate what’s already been perfected, instead of reinventing every wheel, with proven code connected with an already debugged ETW web). The way to tap into that secret sauce IS Xperf.The two excellent appendices give user and kernel debug quick start examples that make this book as much as a reference and tutorial as a step by step learning guide.Beyond debugging, there is a LOT of information on how to develop superior software USING the debugger, not for debugging, but for software analysis, code vs. operating system, security, and development cycle issues like static vs. runtime analysis. Any good or prospective windows developer will benefit from this wealth of info. This is over 500 pages PACKED with wisdom and experience, well worth the price as a career

enhancer or builder.

Great reference for windbg or great for people getting started in Reverse engineering. I would for sure suggest this for people new to this topic and veteran Reverse engineers for a reference.

very good tips. it is always good to have insider information from Microsoft.

This is a very good book.

There is nothing in this book that is surprising it covers the usual things, stack tracing, heap corruption etc.,.The obvious things are covered, but I hoped for more detail 'inside' debugging for windowsIt could also do with a better explanation for how a process is put together, the information is all there, but scattered around the place.

Inside Windows Debugging is the latest book on in-depth debugging and tracing strategies written by an author with an inside look into core techniques of Windows; some of which he worked on directly.With all the latest programming languages and integrated development environments aimed at making writing software applications more accessible, creating applications has never been easier than today. Unfortunately, creating an application is only one part of the equation, getting it to work correctly is the other - usually much harder - part. This book focuses on exactly that harder part; identifying, tracing and resolving bugs in your application as well as preventing them in the first place.Many still think of debugging as an activity after a software application has been finished and users start to experience issues that require investigation. This, however, is far from the truth these days since many development idioms such as test-driven development (TDD) actually promote debugging during development phases.The book is divided into three parts, the first providing a bit of background about the evolution and architecture of Windows, the Windows Developer Interface as well as the Microsoft Developer Tools. The second part introduces the basics of debugging, how the Windows debuggers actually work and debugging your application after a crash (postmortem) before moving to more advanced techniques such as scripting the debugger, debugging the WOW64 environment, code analysis tools, debugging system internals as well as looking at common debugging scenarios. The third part introduces strategies to trace and analyze application behavior using different mechanism and tools such as the Event Tracing for Windows (ETW) and the accompanying Windows Performance Analysis Tool (Xperf). Finally, two appendices provide a

quick start on how to use the WinDbg debugger to accomplish both user-mode and kernel-mode debugging tasks.This book is not aimed at the novice developer by any means since a general understanding of C++ and/or C# as well as the Win32 platform and/or the .NET framework is required. The author does provide an excellent job by introducing basic concepts prior to moving to more advanced topics so that nobody really should get lost while moving from chapter to chapter. By not just preaching the theories but also presenting real-world debugging scenarios, the author also manages to provide developers with methods and tools they immediately can use in their daily routine.I have always been a fan of most books coming from Microsoft Press and this one is no exception: a wealth of information using an inside look into the underlying mechanics and paired with an engaging writing style makes for another book every serious developer should have on his/her shelf.

Alaska (Adventure Guide to the Inside Passage & Coastal Alaska) (Adventure Guide to Coastal Alaska & the Inside Passage) Of Russia: A Year Inside (Of China: A Year Inside, Of Iraq: A Year Inside Book 1)