Inside Windows Debugging (Developer Reference)
**Synopsis**

Use Windows debuggers throughout the development cycle—and build better software. Rethink your use of Windows debugging and tracing tools—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, you’ll apply expert debugging and tracing techniques—and sharpen your C++ and C# code analysis skills—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**

File Size: 38332 KB
Print Length: 592 pages
Simultaneous Device Usage: Up to 5 simultaneous devices, per publisher limits
Publisher: Microsoft Press; 1 edition (May 15, 2012)
Publication Date: May 15, 2012
Sold by: Digital Services LLC
Language: English
ASIN: B00JDMPHN6
Text-to-Speech: Enabled
X-Ray: Not Enabled
Word Wise: Not Enabled
Lending: Not Enabled
Enhanced Typesetting: Not Enabled
Best Sellers Rank: #139,729 Paid in Kindle Store (See Top 100 Paid in Kindle Store) #5 in Books > Computers & Technology > Programming > Languages & Tools > Debugging #9 in Books > Computers & Technology > Programming > Languages & Tools > Compiler Design
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.

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.

Download to continue reading...
