
Java Performance Tuning 2nd Edition

Taking you to the limit in Concurrency, OOP, and the most advanced capabilities of C

Java Performance
 Accelerating MATLAB Performance
 SQL Performance Tuning
 Sun Performance and Tuning
 Java Performance Companion
 The Well-Grounded Java Developer, Second Edition
 Secure Electronic Commerce
 Virtualizing and Tuning Large Scale Java Platforms
 The Well-Grounded Java Developer
 Java Performance Tuning
 Analyzing and Tuning SAP Systems
 WildFly Performance Tuning
 JBoss AS 5 Performance Tuning
 Detecting Anti-Patterns with Open Source Tools
 1001 tips to speed up MATLAB programs
 Lucene in Action
 What's New in Java 7
 A Quantitative Approach
 High Performance Android Apps
 Java Performance and Scalability
 Mule in Action
 EJB 3 in Action
 Java Performance and Scalability: Server-side programming techniques
 Optimizing Java
 High-Performance Java Persistence
 In-Depth Advice for Tuning and Programming Java 8, 11, and Beyond
 Java 8 Pocket Guide
 Clojure High Performance Programming
 IBM Business Process Manager V8.5 Performance Tuning and Best Practices
 Troubleshooting Oracle Performance
 Getting the Most Out of Your Code
 Thinking in Java
 Pro Java EE 5 Performance Management and Optimization
 A Project-Driven Guide to Fundamentals in Java
 Speeding Up the Web
 More than 70 solutions to common Hibernate problems
 Java Platform Performance
 Practical Techniques for Improving JVM Application Performance
 Web Performance Tuning

Java Performance Tuning 2nd Edition

Downloaded from ecobankpayservices.ecobank.com by guest

ASHLEY FINLEY

Taking you to the limit in Concurrency, OOP, and the most advanced capabilities of C Apress

When you use Hibernate in your projects, you quickly recognize that you need to do more than just add @Entity annotations to your domain model classes. Real-world applications often require advanced mappings, complex queries, custom data types and caching. Hibernate can do all of that. You just have to know which annotations and APIs you need to use. *Hibernate Tips - More than 70 solutions to common Hibernate problems* shows you how to efficiently implement your persistence layer with Hibernate's basic and advanced features. Each Hibernate Tip consists of one or more code samples and an easy to follow step-by-step explanation. You can also download an example project with executable test cases for each Hibernate Tip. Throughout this book, you will get more than 70 ready-to-use solutions that show you how to: - Define standard mappings for basic attributes and entity associations. - Implement your own attribute mappings and support custom data types. - Use Hibernate's Java 8 support and other proprietary features. - Read data from the database with JPQL, Criteria API, and native SQL queries. - Call stored procedures and database functions. This book is for developers who are already working with Hibernate and who are looking for solutions for their current development tasks. It's not a book for beginners who are looking for extensive descriptions of Hibernate's general concepts. The tips are designed as self-contained recipes which provide a

specific solution and can be accessed when needed. Most of them contain links to related tips which you can follow if you want to dive deeper into a topic or need a slightly different solution. There is no need to read the tips in a specific order. Feel free to read the book from cover to cover or to just pick the tips that help you in your current project.

Java Performance Addison-Wesley Professional

When you need quick answers for developing or debugging Java programs, this pocket guide provides a handy reference to standard features of the Java programming language and its platform. You'll find helpful programming examples, tables, figures, and lists, as well as Java 8 features such as Lambda Expressions and the Date and Time API. It's an ideal companion, whether you're in the office, in the lab, or on the road. This book also provides material to help you prepare for the Oracle Certified Associate Java Programmer exam. Quickly find Java language details, such as naming conventions, types, statements and blocks, and object-oriented programming Get details on the Java SE platform, including development basics, memory management, concurrency, and generics Browse through information on basic input/output, NIO 2.0, the Java collections framework, and the Java Scripting API Get supplemental references to fluent APIs, third-party tools, and basics of the Unified Modeling Language (UML)

Accelerating MATLAB Performance Dreamtech Press

Troubleshoot the most widespread and pernicious Java performance problems using a set of open-source and freely-available tools that will make you dramatically more productive in finding the root causes of slow performance. This is a brief book that focuses on a small number of performance anti-

patterns, and you'll find that most problems you encounter fit into one of these anti-patterns. The book provides a specific method in a series of steps referred to as the "P.A.t.h. Checklist" that encompasses persistence, alien systems, threads, and heap management. These steps guide you through a troubleshooting process that is repeatable, that you can apply to any performance problem in a Java application. This technique is especially helpful in 'dark' environments with little monitoring. Performance problems are not always localized to Java, but often fall into the realms of database access and server load. This book gives attention to both of these issues through examples showing how to identify repetitive SQL, and identify architecture-wide performance problems ahead of production rollout. Learn how to apply load like an expert, and determine how much load to apply to determine whether your system scales. Included are walk-throughs of a dozen server-side performance puzzles that are ready to run on your own machine. Following these examples helps you learn to: Assess the performance health of four main problems areas in a Java system: The P.A.t.h. Checklist presents each area with its own set of plug-it-in-now tools Pinpoint the code at fault for CPU and other bottlenecks without a Java profiler Find memory leaks in just minutes using heapSpank, the author's open-source leak detector utility that is freely available from heapSpank.org The repeatable method provided in this book is an antidote to lackluster average response times that are multi-second throughout the industry. This book provides a long absent, easy-to-follow, performance training regimen that will benefit anyone programming in Java. What You'll Learn Avoid the 6 most common ways to mess up a load test Determine the exact number of threads to dial into the load generator to test your system's scalability Detect the three most common SQL performance anti-patterns Measure network response times of calls to back-end systems ('alien systems') Identify whether garbage collection performance is healthy or unhealthy and whether delays are caused by problems in the old or new generation, so you know which generation needs to be adjusted Who This Book Is For Intermediate and expert Java developers and architects. Java experts will be able to update their skill set with the latest and most productive, open-source Java performance tools. Intermediate Java developers are exposed to the most common performance defects that repeatedly show up in Java applications, ones that account for the bulk of slow-performing systems. Experts and intermediates alike will benefit from the chapters on load generation.

SQL Performance Tuning Simon and Schuster

Written in Henry Liu's clear, concise style, *Java Performance and Scalability* gets right to the point. With clearly explained concepts, most pertinent theories, precise step-by-step procedures, and large volume of illustrative charts and tables with highly reliable data supporting behind, you gain quickly the necessary knowledge and skills for being able to cope with Java application performance and scalability issues without having to resort to more experienced professionals or expensive external consultants. Specifically, it helps you learn the following knowledge and skills that are essential for you to become more effective in contributing to the success of your organization: * What you need to know at minimum about the architecture of modern hardware so that you can make smart decisions on when you should pour your time on your application and when you can just throw in more advanced hardware to get by. * What you need to know about garbage collection theories in general and how they are implemented with widely used Java Virtual Machines like HotSpot JVMs. * Precise methodologies, procedures, and programs that you can start to use immediately to help you profile and tune your Java applications. * How you can design and build performance and scalability into your product proactively without having to face tough retrofitting decisions or even torrents of customer escalations later on. In addition, the book contains interesting data for your reference, associated with oops compression, CMS garbage collection tuning, DoEscapeAnalysis, G1 versus CMS comparison, etc., all based on full scale, rigorous performance and scalability tests with real products.

Sun Performance and Tuning Packt Publishing Ltd

Java® Performance Companion shows how to systematically and proactively improve Java performance with today's advanced multicore hardware and complex operating system environments. The authors, who are all leading Java performance and Java HotSpot VM experts, help you improve performance by using modern software engineering practices, avoiding common mistakes, and applying tips and tricks gleaned from years of real-world experience. Picking up where Charlie Hunt and Binu John's classic *Java Performance* left off, this book provides unprecedented detail on two powerful Java platform innovations: the Garbage First (G1) garbage collector and the HotSpot VM Serviceability Agent. Coverage includes Leveraging G1 to overcome limitations in parallel, serial, and CMS garbage collection Understanding each stage of G1 GC collections, both young and old Getting under the hood with G1 and efficiently fine-tuning it for your application Identifying potential optimizations, interpreting experimental results, and taking action Exploring the internals of the HotSpot VM Using HotSpot VM Serviceability Agent to analyze, triage, and resolve diverse HotSpot VM issues Troubleshooting out of memory errors, Java level deadlocks, and HotSpot VM crashes Extending the Serviceability Agent, and using the Plugin for VisualVM Mastering useful HotSpot VM command line options not covered in Java™ Performance Java® Performance Companion can help you squeeze maximum performance and value from Java with JDK 8 or 9—for any application, in any environment. Register your product at informit.com/register for convenient access to downloads, updates, and corrections as they become available.

Java Performance Companion "O'Reilly Media, Inc."

Threads are a fundamental part of the Java platform. As multicore processors become the norm, using concurrency effectively becomes essential for building high-performance applications. Java SE 5 and 6 are a huge step forward for the development of concurrent applications, with improvements to the Java Virtual Machine to support high-performance, highly scalable concurrent classes and a rich set of new concurrency building blocks. In *Java Concurrency in Practice*, the creators of these new facilities explain not only how they work and how to use them, but also the motivation and design patterns behind them. However, developing, testing, and debugging multithreaded programs can still be very difficult; it is all too easy to create concurrent programs that appear to work, but fail when it matters most: in production, under heavy load. *Java Concurrency in Practice* arms readers with both the theoretical underpinnings and concrete techniques for building reliable, scalable, maintainable concurrent applications. Rather than simply offering an inventory of concurrency APIs and mechanisms, it provides design rules, patterns, and mental models that make it easier to build concurrent programs that are both correct and performant. This book covers: Basic concepts of concurrency and thread safety Techniques for building and composing thread-safe classes Using the concurrency building blocks in `java.util.concurrent` Performance optimization dos and don'ts Testing concurrent programs Advanced topics such as atomic variables, nonblocking algorithms, and the Java Memory Model *The Well-Grounded Java Developer, Second Edition* Thoughts on Java

First book to address and assess performance of enterprise Java-based applications using the new Java EE 5 Presents Java EE 5 Performance Management as a proven methodology, featuring a set of common problems that have been observed in real-world customer environments Presents "wait-based" performance tuning methodology, the most efficient Java EE 5 tuning methodology, but one previously neglected in the Java EE 5 space *Secure Electronic Commerce* O'Reilly Media

Unique and clever ideas are important when building a hot-selling Android app, but the real drivers for success are speed, efficiency, and power management. With this practical guide, you'll learn the major performance issues confronting Android app developers, and the tools you need to diagnose problems early. Customers are finally realizing that apps have a major role in the performance of their Android devices. Author Doug Sillars not only shows you how to use Android-specific testing tools from companies including Google, Qualcomm, and AT&T, but also helps you explore potential remedies. You'll discover ways to build apps that run well on all 19,000 Android device types in use. Understand how performance issues affect app sales and retention Build an Android device lab to maximize UI, functional, and performance testing Improve the way your app interacts with device hardware Optimize your UI for fast rendering, scrolling, and animations Track down memory leaks and CPU issues that affect performance Upgrade communications with the server, and learn how your app performs on slower networks Apply Real User Monitoring (RUM) to ensure that every device is delivering the optimal user experience

Virtualizing and Tuning Large Scale Java Platforms Apress

PLEASE PROVIDE DESCRIPTION

The Well-Grounded Java Developer "O'Reilly Media, Inc."

Become an expert at writing fast and high performant code in Clojure 1.7.0 About This Book Enhance code performance by using appropriate Clojure features Improve the efficiency of applications and plan their deployment A hands-on guide to designing Clojure programs to get the best performance Who This Book Is For This book is intended for intermediate Clojure developers who are looking to get a good grip on achieving optimum performance. Having a basic knowledge of Java would be helpful. What You Will Learn Identify performance issues in Clojure programs using different profiling tools Master techniques to achieve numerical performance in Clojure Use Criterium library to measure latency of Clojure expressions Exploit Java features in Clojure code to enhance performance Avoid reflection and boxing with type hints Understand Clojure's concurrency and state-management primitives in depth Measure and monitor performance, and understand optimization techniques In Detail Clojure treats code as data and has a macro system. It focuses on programming with immutable values and explicit progression-of-time constructs, which are intended to facilitate the development of more robust programs, particularly multithreaded ones. It is built with performance, pragmatism, and simplicity in mind. Like most general purpose languages, various Clojure features have different performance characteristics that one should know in order to write high performance code. This book shows you how to evaluate the performance implications of various Clojure abstractions, discover their underpinnings, and apply the right approach for optimum performance in real-world programs. It starts by helping you classify various use cases and the need for them with respect to performance and analysis of various performance aspects. You will also learn the performance vocabulary that experts use throughout the world and discover various Clojure data structures, abstractions, and their performance characteristics. Further, the book will guide you through enhancing performance by using Java interoperability and JVM-specific features from Clojure. It also highlights the importance of using the right concurrent data structure and Java concurrency abstractions. This book also sheds light on performance metrics for measuring, how to measure, and how to visualize and monitor the collected data. At the end of the book, you will learn to run a performance profiler, identify bottlenecks, tune performance, and refactor code to get a better performance. Style and approach An easy-to-follow guide full of real-world examples and self-sufficient code snippets that will help you get your hands dirty with high performance programming with Clojure.

Java Performance Tuning "O'Reilly Media, Inc."

Coding and testing are generally considered separate areas of expertise. In this practical book, Java expert Scott Oaks takes the approach that anyone who works with Java should be adept at understanding how code behaves in the Java Virtual Machine—including the tunings likely to help performance. This updated second edition helps you gain in-depth knowledge of Java application performance using both the JVM and the Java platform. Developers and performance engineers alike will learn a variety of features, tools, and processes for improving the way the Java 8 and 11 LTS releases perform. While the emphasis is on production-supported releases and features, this book also features previews of exciting new technologies such as ahead-of-time compilation and experimental garbage collections. Understand how various Java platforms and compilers affect performance Learn how Java garbage collection works Apply four principles to obtain best results from performance testing Use the JDK and other tools to learn how a Java application is performing Minimize the garbage collector's impact through tuning and programming practices Tackle performance issues in Java APIs Improve Java-driven database application performance

Analyzing and Tuning SAP Systems Simon and Schuster

Have you thought about building games for your cell phone or other wireless devices? Whether you are a first time wireless Java developer, or an experienced professional *Beginning J2ME, Third Edition* brings exciting wireless and mobile Java application development right to your door!

WildFly Performance Tuning "O'Reilly Media, Inc."

Coding and testing are often considered separate areas of expertise. In this comprehensive guide, author and Java expert Scott Oaks takes the approach that anyone who works with Java should be equally adept at understanding how code behaves in the JVM, as well as the tunings likely to help its performance. You'll gain in-depth knowledge of Java application performance, using the Java Virtual Machine (JVM) and the Java platform, including the language and API. Developers and performance engineers alike will learn a variety of features, tools, and processes for improving the way Java 7 and 8 applications perform. Apply four principles for obtaining the best results from performance testing Use JDK tools to collect data on how a Java application is performing Understand the advantages and disadvantages of using a JIT compiler Tune JVM garbage collectors to affect programs as little as possible Use techniques to manage heap memory and JVM native memory Maximize Java threading and synchronization performance features Tackle performance issues in Java EE and Java SE APIs Improve Java-driven database application performance *JBoss AS 5 Performance Tuning* Simon and Schuster

"Newcomers will appreciate the clear explanations of the origins and development of secure e-commerce. More experienced developers can move straight to the detailed technical material. Anyone who is involved in e-commerce design, management, or operation will benefit from Secure Electronic Commerce."--BOOK JACKET.

Detecting Anti-Patterns with Open Source Tools Prentice Hall Professional

Java 7 has a number of features that will please developers. Madhusudhan Konda provides an overview of these, including strings in switch statements, multi-catch exception handling, try-with-resource statements, the new File System API, extensions of the JVM, support for dynamically-typed languages, and the fork and join framework for task parallelism.

1001 tips to speed up MATLAB programs Prentice Hall

Performance tuning is an experimental science, but that doesn't mean engineers should resort to guesswork and folklore to get the job done. Yet that's often the case. With this practical book, intermediate to advanced Java technologists working with complex technology stacks will learn how to tune Java applications for performance using a quantitative, verifiable approach. Most resources on performance tend to discuss the theory and internals of Java virtual machines, but this book focuses on the practicalities of performance tuning by examining a wide range of aspects. There are no simple recipes, tips and tricks, or algorithms to learn. Performance tuning is a process of defining and determining desired outcomes. And it requires diligence. Learn how Java principles and technology make the best use of modern hardware and operating systems Explore several performance tests and common anti-patterns that can vex your team Understand the pitfalls of measuring Java performance numbers and the drawbacks of microbenchmarking Dive into JVM garbage collection logging, monitoring, tuning, and tools Explore JIT compilation and Java language performance techniques Learn performance aspects of the Java Collections API and get an overview of Java concurrency

Lucene in Action Pearson Education

This handbook is for anyone responsible for a Web site, from the person running a personal site off a Linux PC at home up to large corporate site managers who wants to improve their performance right now.

Related with Java Performance Tuning 2nd Edition:

© [Java Performance Tuning 2nd Edition Honeywell Heat Pump Thermostat Wiring Diagram](#)

© [Java Performance Tuning 2nd Edition Honda Civic Manual 1999](#)

© [Java Performance Tuning 2nd Edition Honors Algebra 2 Curriculum](#)

What's New in Java 7 IBM Redbooks

Build faster, more efficient enterprise Java applications.

[A Quantitative Approach](#) "O'Reilly Media, Inc."

Software -- Programming Languages.

High Performance Android Apps Packt Publishing Ltd

Summary Mule in Action, Second Edition is a totally-revised guide covering Mule 3 fundamentals and best practices. It starts with a quick ESB overview and then dives into rich examples covering core concepts like sending, receiving, routing, and transforming data. About the Technology An enterprise service bus is a way to integrate enterprise applications using a bus-like infrastructure. Mule is the leading open source Java ESB. It borrows from the Hohpe/Woolf patterns, is lightweight, can publish REST and SOAP services, integrates well with Spring, is customizable, scales well, and is cloud-ready. About the Book Mule in Action, Second Edition is a totally revised guide covering Mule 3 fundamentals and best practices. It starts with a quick ESB overview and then dives into rich examples covering core concepts like sending, receiving, routing, and transforming data. You'll get a close look at Mule's standard components and how to roll out custom ones. You'll also pick up techniques for testing, performance tuning, and BPM orchestration, and explore cloud API integration for SaaS applications. Written for developers, architects, and IT managers, this book requires familiarity with Java but no previous exposure to Mule or other ESBs. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. What's Inside Full coverage of Mule 3 Integration with cloud services Common transports, routers, and transformers Security, routing, orchestration, and transactions About the Authors David Dossot is a software architect and has created numerous modules and transports for Mule. John D'Emic is a principal solutions architect and Victor Romero a solutions architect, both at MuleSoft, Inc. Table of Contents PART 1 CORE MULE Discovering Mule Processing messages with Mule Working with connectors Transforming data with Mule Routing data with Mule Working with components and patterns PART 2 RUNNING MULE Integration architecture with Mule Deploying Mule Exception handling and transaction management with Mule Securing Mule Tuning Mule PART 3 TRAVELING FURTHER WITH MULE Developing with Mule Writing custom cloud connectors and processors Augmenting Mule with orthogonal technologies