
High Performance Java Persistence

PentaBAR Webinar // High-Performance Java Persistence High-Performance Java Persistence and Cloud Native QBasic VLAD MIHALCEA - Java Persistence and Hibernate Tips that can boost up your application performance High-Performance Hibernate (Vlad Mihalcea) The Best Way to Fetch Data with Java Persistence and Hibernate Workshop by Vlad Mihalcea DevNation: The Show (Oct 20, 2020) JDK IO 2018 - Vlad Mihalcea - High-Performance Hibernate Excellent discussion with Catalin Tudose on Java Persistence Migrating from (Spring Data) JPA to Spring Data JDBC by Jens Schauder @ Spring I/O 2024 Spring Office Hours: S3E41 - Spring's Hidden Powers with Greg Turnquist Building Fast and Scalable Persistence Layers with Spring Data JPA The best way to fetch entities with JPA and Hibernate Modern Java in Action Top 6 Most Popular API Architecture Styles 10,000 Java performance tips over 15 years - what did I learn? by Jack Shirazi Parallel Streams, CompletableFuture, and All That: Concurrency in Java 8 VICTOR RENTEA - Functional Programming Patterns with Java8 This is why understanding database concurrency control is important High-

Performance Hibernate Sven Ruppert — Hidden pearls for high-performance-persistence in Java
High-Performance Hibernate - Vlad Mihalcea
VDB17 • Vlad Mihalcea - Concurrency Control Patterns
Vlad Mihalcea - High-Performance Hibernate
Vlad Mihalcea - Transactions and Concurrency Control Patterns
Mastering Java Persistence: Best Practices for Cloud-Native
Modernization | Otavio Santana (EN)
Java Persistence: From DB over JDBC to Transactions
Best 5 books for Java programmers | I have been reading these Java books
Hibernate Performance Tuning Done Right
JPA & Hibernate: Using the Optimal Query Approach & Projection
Basic Concepts
Mastering Enterprise JavaBeans
Beginning jOOQ
Hands-On Cloud-Native Applications with Java and Quarkus
What Every Superuser Should Know
Spring Boot Persistence Best Practices
Java Performance: The Definitive Guide
Harnessing Hibernate
The Java EE 6 Tutorial
Hibernate Tips
Spring Data
High Performance Spark
How Linux Works, 2nd Edition
Optimize Java Persistence Performance in Spring Boot Applications
Getting the Most Out of Your Code
Spring Boot Persistence Best Practices

Java Concurrency in Practice
A Developer's Notebook
In-Depth Advice for Tuning and Programming Java
8, 11, and Beyond
High Performance Python
Techniques for scaling and optimizing Spring and
Spring Boot applications
Pro Spring 5
An In-Depth Guide to the Spring Framework and
Its Tools

*High
Performance
Java
Persistence* OMB No.
9146527834710
edited by

**SWEENEY
HESS**

**BASIC
CONCEPTS**

Simon and
Schuster
Get started
with
Hibernate, an
open source
Java
persistence
layer and gain
a clear
introduction to
the current
standard for

object-
relational
persistence in
Java. This
updated
edition
includes the
new Hibernate
6.0 framework
which covers
new
configuration,
new object
relational
mapping
changes, and
enhanced
integration
with the more
general
Spring, Boot

and Quarkus
and other Java
frameworks.
The book
keeps its
focus on
Hibernate
without
wasting time
on
nonessential
third-party
tools, so you'll
be able to
immediately
start building
transaction-
based engines
and
applications.
Experienced

authors Joseph Ottinger with Dave Minter and Jeff Linwood provide more in-depth examples than any other book for Hibernate beginners. They present their material in a lively, example-based manner—not a dry, theoretical, hard-to-read fashion. What You'll Learn Build enterprise Java-based transaction-type applications that access complex data with Hibernate

Work with Hibernate 6 using a present-day build process Integrate into the persistence life cycle Search and query with the new version of Hibernate Keep track of versioned data with Hibernate Envers Who This Book Is For Programmers experienced in Java with databases (the traditional, or connected, approach), but new to open-source, lightweight Hibernate.

MASTERING ENTERPRISE JAVABEANS

Simon and Schuster
This guide walks the reader through the ins and outs of using Hibernate, from installation and configuration, to complex associations and composite types.

BEGINNING JOOQ

"O'Reilly Media, Inc."
This book is a collection of developer code recipes and best practices for

persisting data using Spring, particularly Spring Boot. The book is structured around practical recipes, where each recipe discusses a performance case or performance-related case, and almost every recipe has one or more applications. Mainly, when we try to accomplish something (e.g., read some data from the database), there are several approaches to

do it, and, in order to choose the best way, you have to know the implied trades-off from a performance perspective. You'll see that in the end, all these penalties slow down the application. Besides presenting the arguments that favor a certain choice, the application is written in Spring Boot style which is quite different than plain Hibernate. Persistence is an important set of

techniques and technologies for accessing and using data, and this book demonstrates that data is mobile regardless of specific applications and contexts. In Java development, persistence is a key factor in enterprise, ecommerce, cloud and other transaction-oriented applications. After reading and using this book, you'll have the fundamentals to apply these persistence

solutions into your own mission-critical enterprise Java applications that you build using Spring. What You Will Learn Shape *-to-many associations for best performances Effectively exploit Spring Projections (DTO) Learn best practices for batching inserts, updates and deletes Effectively fetch parent and association in a single SELECT Learn how to inspect Persistent

Context content Dissect pagination techniques (offset and keyset) Handle queries, locking, schemas, Hibernate types, and more Who This Book Is For Any Spring and Spring Boot developer that wants to squeeze the persistence layer performances. [Hands-On Cloud-Native Applications with Java and Quarkus](#) Apress Unlike some operating

systems, Linux doesn't try to hide the important bits from you—it gives you full control of your computer. But to truly master Linux, you need to understand its internals, like how the system boots, how networking works, and what the kernel actually does. In this completely revised second edition of the perennial best seller How Linux Works, author Brian Ward makes the concepts behind Linux

internals accessible to anyone curious about the inner workings of the operating system. Inside, you'll find the kind of knowledge that normally comes from years of experience doing things the hard way. You'll learn:

- How Linux boots, from boot loaders to init implementations (systemd, Upstart, and System V)
- How the kernel manages devices, device drivers, and processes

- How networking, interfaces, firewalls, and servers work
- How development tools work and relate to shared libraries
- How to write effective shell scripts

You'll also explore the kernel and examine key system tasks inside user space, including system calls, input and output, and filesystems. With its combination of background, theory, real-world examples, and

patient explanations, *How Linux Works* will teach you what you need to know to solve pesky problems and take control of your operating system.

What Every Superuser Should Know Apress

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

**SPRING
BOOT
PERSISTENCE
BEST
PRACTICES**

"O'Reilly Media, Inc." Scaling Java enterprise

applications beyond just programming techniques-- this is the next level. This volume covers all the technologies Java developers need to build scalable, high-performance Web applications. The book also covers servlet-based session management, EJB application logic, database design and integration, and more. Java Performance: The Definitive Guide Addison-Wesley

Professional Your Python code may run correctly, but you need it to run faster. Updated for Python 3, this expanded edition shows you how to locate performance bottlenecks and significantly speed up your code in high-data-volume programs. By exploring the fundamental theory behind design choices, High Performance Python helps you gain a deeper understanding of Python's implementation

n. How do you take advantage of multicore architectures or clusters? Or build a system that scales up and down without losing reliability? Experienced Python programmers will learn concrete solutions to many issues, along with war stories from companies that use high-performance Python for social media analytics, productionized machine learning, and more. Get a better grasp of NumPy,

Cython, and profilers Learn how Python abstracts the underlying computer architecture Use profiling to find bottlenecks in CPU time and memory usage Write efficient programs by choosing appropriate data structures Speed up matrix and vector computations Use tools to compile Python down to machine code Manage multiple I/O and computational operations

concurrently Convert multiprocessing code to run on local or remote clusters Deploy code faster using tools like Docker **Harnessing Hibernate** Pearson Education If you are a professional or enthusiast who has a basic understanding of graphs or has basic knowledge of Neo4j operations, this is the book for you. Although it is targeted at an advanced user base, this

book can be used by beginners as it touches upon the basics. So, if you are passionate about taming complex data with the help of graphs and building high performance applications, you will be able to get valuable insights from this book.

The Java EE 6 Tutorial No Starch Press Summary Java Persistence with Hibernate, Second Edition explores Hibernate by developing an application

that ties together hundreds of individual examples. In this revised edition, authors Christian Bauer, Gavin King, and Gary Gregory cover Hibernate 5 in detail with the Java Persistence 2.1 standard (JSR 338). All examples have been updated for the latest Hibernate and Java EE specification versions. About the Technology Purchase of the print book includes a free eBook in PDF,

Kindle, and ePub formats from Manning Publications. Persistence—the ability of data to outlive an instance of a program—is central to modern applications. Hibernate, the most popular Java persistence tool, offers automatic and transparent object/relational mapping, making it a snap to work with SQL databases in Java applications. About the Book Java Persistence with Hibernate,

Second Edition explores Hibernate by developing an application that ties together hundreds of individual examples. You'll immediately dig into the rich programming model of Hibernate, working through mappings, queries, fetching strategies, transactions, conversations, caching, and more. Along the way you'll find a well-illustrated discussion of best practices in database design and optimization techniques. In this revised edition, authors Christian Bauer, Gavin King, and Gary Gregory cover Hibernate 5 in detail with the Java Persistence 2.1 standard (JSR 338). All examples have been updated for the latest Hibernate and Java EE specification versions. What's Inside Object/relational mapping concepts Efficient database application design Comprehensive Hibernate and Java Persistence reference Integration of Java Persistence with EJB, CDI, JSF, and JAX-RS * Unmatched breadth and depth About the Reader The book assumes a working knowledge of Java. About the Authors Christian Bauer is a member of the Hibernate developer team and a trainer and consultant. Gavin King is

the founder of the Hibernate project and a member of the Java Persistence expert group (JSR 220). Gary Gregory is a principal software engineer working on application servers and legacy integration.	STRATEGIES Mapping persistent classes Mapping value types Mapping inheritance Mapping collections and entity associations Advanced entity association mappings Complex and legacy schemas	WRITING QUERIES Creating and executing queries The query languages Advanced query options Customizing SQL
Table of Contents	PART 3	Hibernate Tips O'Reilly Media
1 GETTING STARTED WITH ORM	TRANSACTIONAL DATA PROCESSING	Spring 5.0 brings major advancements in the rich APIs provided by the Spring framework and thus creates a need for developers to master its tools and techniques to achieve high-performing applications. This book will help you
Understanding object/relational persistence	Managing data	
Starting a project	Transactions and concurrency	
Domain models and metadata	Fetch plans, strategies, and profiles	
PART 2	Filtering data	
MAPPING	PART 4	

improve the speed of your code and optimize the performance of your apps. **Spring Data** Apress
 How can you bring out MySQL's full power? With High Performance MySQL, you'll learn advanced techniques for everything from designing schemas, indexes, and queries to tuning your MySQL server, operating system, and hardware to their fullest potential. This guide also

teaches you safe and practical ways to scale applications through replication, load balancing, high availability, and failover. Updated to reflect recent advances in MySQL and InnoDB performance, features, and tools, this third edition not only offers specific examples of how MySQL works, it also teaches you why this system works a. **High Performance**

Spark Packt Publishing Ltd
 A high-performance data access layer must resonate with the underlying database system. Knowing the inner workings of a relational database and the data access frameworks in use can make the difference between a high-performance enterprise application and one that barely crawls. This book is a journey into Java data access performance tuning. From

connection management, to batch updates, fetch sizes and concurrency control mechanisms, it unravels the inner workings of the most common Java data access frameworks. The first part aims to reduce the gap between application developers and database administrators . For this reason, it covers both JDBC and the database fundamentals that are of paramount importance when reducing

transaction response times. In this first part, you'll learn about connection management, batch updates, statement caching, result set fetching and database transactions. The second part demonstrates how you can take advantage of JPA and Hibernate without compromising application performance. In this second part, you'll learn about the most efficient

Hibernate mappings (basic types, associations, inheritance), fetching best practices, caching and concurrency control mechanisms. The third part is dedicated to jOOQ and its powerful type-safe querying capabilities, like window functions, common table expressions, upsert, stored procedures and database functions. [How Linux Works, 2nd Edition](#) Routledge The Java EE 6 Tutorial: Basic Concepts,

Fourth Edition, is a task-oriented, example-driven guide to developing enterprise applications for the Java Platform, Enterprise Edition 6 (Java EE 6). Written by members of the Java EE 6 documentation team at Oracle, this book provides new and intermediate Java programmers with a deep understanding of the platform. Starting with expert guidance on web tier technologies, including JavaServer Faces and Facelets, this book also covers building web services using JAX-WS and JAX-RS, developing business logic with Enterprise JavaBeans components, accessing databases using the Java Persistence API, securing web and enterprise applications, and using Contexts and Dependency Injection for the Java EE platform. This edition contains extensive new material throughout, including detailed introductions to the latest APIs and platform features, and instructions for using the latest versions of GlassFish Server Open Source Edition and NetBeans IDE. Key platform features covered include Convention over configuration, so developers need specify only those aspects of an application that vary from

the convention Annotated POJOs (Plain Old Java Objects) with optional XML configuration Simplified but more flexible packaging Lightweight Web Profile that is ideal for developing web applications The Java Series...from the Source Since 1996, when Addison-Wesley published the first edition of The Java Programming Language by Ken Arnold and James Gosling, this series has

been the place to go for complete, expert, and definitive information on Java technology. The books in this series provide the detailed information developers need to build effective, robust, and portable applications and are an indispensable resource for anyone using the Java platform. [Optimize Java Persistence Performance in Spring Boot Applications](#) "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. **Getting the Most Out of Your Code**. Pearson Education. Understand the newest trend in database programming for developers working in Java, Kotlin, Clojure, and other JVM-based languages. This book introduces Reactive Relational Database

Connectivity (R2DBC), a modern way of connecting to and querying relational databases from Java and other JVM languages. The book begins by helping you understand not only what reactive programming is, but why it is necessary. Then building on those fundamentals, the book takes you into the world of databases and the newly released Reactive Relational Database

Connectivity (R2DBC) specification. Examples in the book are worked using the freely available MariaDB database along with MariaDB's vendor-implementation of the R2DBC service-provider interface (SPI). Following along with the examples and the provided example code helps prepare you to work with any of the growing number of R2DBC implementations for popular

enterprise databases such as Oracle Database and SQL Server. You'll be well prepared for what is becoming the future of database access from Java and other languages built on the JVM. What You Will Learn Understand why R2DBC was created and how it utilizes the Reactive Streams API Understand the components of the R2DBC service-provider interface Create and

manage reactive database connections and connection pools using an R2DBC client Programmatically execute queries on a relational database using an R2DBC client Effectively utilize transactions using an R2DBC client Build relational database-driven applications that are event-driven and non-blocking Who This Book Is For Software developers

building solutions using JVM languages and the JVM ecosystem, and developers who need an introduction to the R2DBC specification and reactive programming with relational databases and want to understand what Reactive Relational Database Connectivity is and why it came about. This book includes practical examples of using the R2DBC specification with Java and

MariaDB that will provide developers with the knowledge they need to create their own solutions. [Spring Boot Persistence Best Practices](#) Apress Build robust and reliable Java applications that works on modern infrastructure, such as containers and cloud, using the new features in Quarkus 1.0 Key Features Build apps with faster boot time and low RSS memory using the latest

Quarkus 1.0 features Seamlessly integrate imperative and reactive programming models to build modern Java applications Discover effective solutions for running Java on serverless apps, microservices, containers, FaaS, and the cloud Book Description Quarkus is a new Kubernetes-native framework that allows Java developers to combine the power of

containers, microservices, and cloud-native to build reliable applications. The book is a development guide that will teach you how to build Java-native applications using Quarkus and GraalVM. We start by learning about the basic concepts of a cloud-native application and its advantages over standard enterprise applications. Then we will quickly move on to application development, by installing

the tooling required to build our first application on Quarkus. Next, we'll learn how to create a container-native image of our application and execute it in a Platform-as-a-Service environment such as Minishift. Later, we will build a complete real-world application that will use REST and the Contexts and Dependency injection stack with a web frontend. We will also learn how to add

<p>database persistence to our application using PostgreSQL. We will learn how to work with various APIs available to Quarkus such as Camel, Eclipse MicroProfile, and Spring DI. Towards the end, we will learn advanced development techniques such as securing applications, application configuration, and working with non-blocking programming models using Vert.x. By the</p>	<p>end of this book, you will be proficient with all the components of Quarkus and developing blazing fast applications leveraging modern technology infrastructure. What you will learn Build a native application using Quarkus and GraalVM Secure your applications using Elytron and the MicroProfile JWT extension Manage data persistence with Quarkus using PostgreSQL Use a non-blocking</p>	<p>programming model with Quarkus Learn how to get Camel and Infinispan working in native mode Deploy an application in a Kubernetes-native environment using Minishift Discover Reactive Programming with Vert.x Who this book is for The book is for Java developers and software architects who are interested in learning a promising microservice architecture for building reliable and robust</p>
--	--	--

applications. Knowledge of Java, Spring Framework, and REST APIs is assumed.

Java Concurrency in Practice
Packt Publishing Ltd
Have you ever felt frustrated working with someone else's code? Difficult-to-maintain source code is a big problem in software development today, leading to costly delays and defects. Be part of the solution. With this practical book, you'll learn 10 easy-to-follow guidelines for delivering Java software that's easy to maintain and adapt. These guidelines have been derived from analyzing hundreds of real-world systems. Written by consultants from the Software Improvement Group (SIG), this book provides clear and concise explanations, with advice for turning the guidelines into practice. Examples for this edition are written in Java, while our companion C# book provides workable examples in that language. Write short units of code: limit the length of methods and constructors. Write simple units of code: limit the number of branch points per method. Write code once, rather than risk copying buggy code. Keep unit interfaces small by extracting parameters into objects. Separate concerns to avoid building large classes. Couple architecture

components loosely
 Balance the number and size of top-level components in your code
 Keep your codebase as small as possible
 Automate tests for your codebase
 Write clean code, avoiding "code smells" that indicate deeper problems
[A Developer's Notebook](#)
 Apress
 Passive House Details introduces the concepts, principles, and design processes of building

ultralow-energy buildings. The objective of this book is to provide design goals, research, analysis, systems, details, and inspiring images of some of the most energy-efficient, carbon-neutral, healthy, and satisfying buildings currently built in the region.
 Other topics included: heat transfer, moisture management, performance targets, and climatic zones.

Illustrated with more than 375 color images, the book is a visual catalog of construction details, materials, and systems drawn from projects contributed from forty firms.
 Fourteen in-depth case studies demonstrate the most energy-efficient systems for foundations, walls, floors, roofs, windows, doors, and more.
[In-Depth Advice for](#)

Tuning and Programming Java 8, 11, and Beyond

Thoughts on Java
Includes more than 30 percent revised material and five new chapters, covering the new 2.1 features such as EJB Timer Service and JMS as well as the latest open source Java solutions
The book was developed as part of TheServerSide.com online EJB community, ensuring a built-in audience

Demonstrates how to build an EJB system, program with EJB, adopt best practices, and harness advanced EJB concepts and techniques, including transactions, persistence, clustering, integration, and performance optimization
Offers practical guidance on when not to use EJB and how to use simpler, less costly open source technologies in place of or in conjunction with EJB

HIGH PERFORMANCE PYTHON

Packt Pub Limited
This book is written for users experienced in using Java with databases but inexperienced in the use of the open source, lightweight Hibernate, the most popular de-facto object-relational mapping and database-oriented application development framework.
The book has plentiful examples and

handy reference sections, including a comprehensive reference for Hibernate O/R mapping strategies. Beginning Hibernate 3 is packed with brand-new information on the latest release of the Hibernate persistence layer and provides a clear introduction to the de facto standard for object relational persistence in Java. Readers will get started right away with building transaction-based engines and applications.

Related with High Performance Java Persistence:

[© High Performance Java Persistence Free Printable Letter R Worksheets For Preschool](#)

[© High Performance Java Persistence Free Printable Line Plot Worksheets](#)

[© High Performance Java Persistence Free Printable Tracing Numbers 1 10 Worksheets Pdf](#)