

Romain Manni-Bucau

Java EE 8 High Performance

Master techniques such as memory optimization, caching, concurrency, and multithreading to achieve maximum performance from your enterprise applications



Packt>

Java EE 8 High Performance

The ease with which we write applications has been increasing, but with this comes the need to address their performance. A balancing act between easily implementing complex applications and keeping their performance optimal is a present-day need. In this book, we explore how to achieve this crucial balance while developing and deploying applications with Java EE 8.

The book starts by analyzing various Java EE specifications to identify those potentially affecting performance adversely. Then, we move on to monitoring techniques that enable us to identify performance bottlenecks and optimize performance metrics. Next, we look at techniques that help us achieve high performance: memory optimization, concurrency, multithreading, scaling, and caching. We also look at fault tolerance solutions and the importance of logging. Lastly, you will learn to benchmark your application and implement solutions for continuous performance evaluation.

By the end of the book, you will have gained insights into various techniques and solutions that will help create high-performance applications in the Java EE 8 environment.

Things you will learn:

- Identify performance bottlenecks in an application
- Locate application hotspots using performance tools
- Understand the work done under the hood by EE containers and its impact on performance
- Identify common patterns to integrate with Java EE applications
- Implement transparent caching on your applications
- Extract more information from your applications using Java EE without modifying existing code
- Ensure constant performance and eliminate regression

Java EE 8 High Performance

Romain Manni-Bucau



Java EE 8 High Performance

Master techniques such as memory optimization, caching, concurrency, and multithreading to achieve maximum performance from your enterprise applications

Romain Manni-Bucau



BIRMINGHAM - MUMBAI

Java EE 8 High Performance

Copyright © 2018 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author, nor Packt Publishing or its dealers and distributors, will be held liable for any damages caused or alleged to have been caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

Commissioning Editor: Richa Tripathi
Acquisition Editor: Aiswarya Narayanan
Content Development Editor: Vikas Tiwari
Technical Editor: Diwakar Shukla
Copy Editor: Muktikant Garimella
Project Coordinator: Ulhas Kambali
Proofreader: Safis Editing
Indexer: Aishwarya Gangawane
Graphics: Tania Dutta
Production Coordinator: Shantanu Zagade

First published: January 2018

Production reference: 1250118

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham
B3 2PB, UK.

ISBN 978-1-78847-306-4

www.packtpub.com

This book is dedicated to my family and closest ones, to all the wonderful people I met, to the people who inspire me everyday, Mark, David, Theresa, Jean-Louis, Jean-François, Jean-Baptiste, and to all my other Apache Software Foundation friends.

*Finally I would like to dedicate this book to you, reminding you the quote assigned to Margaret Fuller:
"Today a reader, tomorrow a leader!"*



mapt.io

Mapt is an online digital library that gives you full access to over 5,000 books and videos, as well as industry leading tools to help you plan your personal development and advance your career. For more information, please visit our website.

Why subscribe?

- Spend less time learning and more time coding with practical eBooks and Videos from over 4,000 industry professionals
- Improve your learning with Skill Plans built especially for you
- Get a free eBook or video every month
- Mapt is fully searchable
- Copy and paste, print, and bookmark content

PacktPub.com

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at www.PacktPub.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at service@packtpub.com for more details.

At www.PacktPub.com, you can also read a collection of free technical articles, sign up for a range of free newsletters, and receive exclusive discounts and offers on Packt books and eBooks.

Contributors

About the author

Romain Manni-Bucau is a senior software engineer who has been working in the Java EE space for 7 years and, more recently, with big data. He joined the Apache EE family in 2011 (TomEE, Johnzon, OpenWebBeans, BatchEE, OpenJPA, and Meecrowave). He mainly contributes to open source projects (CukeSpace, Arquillian, Apache Beam, and others) and aims to make development easy and efficient, letting people create what they have in mind without caring for the technical challenges. You can follow Romain on twitter at `@rmannibucau`.

About the reviewer

Luigi Fugaro has been a developer since he was a little child, doing experiments and trying out any new technologies—that is, video games! Growing up in the Java environment taught him how great and valuable the open source world is, but only when he worked at Red Hat he did get the time and tools to really participate in this big family, the open source community. He has been working for Red Hat for 6 years as an EMEA Middleware Architect.

He has authored two books, *WildFly Cookbook* and *Mastering JBoss EAP 7*, both published by Packt.

Packt is searching for authors like you

If you're interested in becoming an author for Packt, please visit authors.packtpub.com and apply today. We have worked with thousands of developers and tech professionals, just like you, to help them share their insight with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Table of Contents

| | |
|--|----|
| Preface | 1 |
| Chapter 1: Money – The Quote Manager Application | 7 |
| Setting up the environment | 7 |
| The application architecture | 9 |
| Application project highlights | 10 |
| The build | 10 |
| The persistence layer | 11 |
| The service layer | 13 |
| The JAX-RS layer | 15 |
| The WebSocket layer | 16 |
| Provision some data | 20 |
| Application summary | 24 |
| The application business | 24 |
| The application design | 24 |
| The application server | 25 |
| Testing the application | 26 |
| Get a quote price the JAX-RS way | 26 |
| Get the price, the WebSocket way | 28 |
| Setting up MySQL | 28 |
| Conclusion | 31 |
| Summary | 31 |
| Chapter 2: Looking Under the Cover – What is This EE Thing? | 33 |
| Context and Dependency Injection – what did you do to my beans? | 33 |
| IoC – a pretty simple example | 34 |
| The main features of CDI | 37 |
| Injections | 37 |
| Scopes | 38 |
| Interceptors/decorators | 39 |
| Events | 44 |
| Dynamic lookups | 45 |
| JAX-RS – the servlet router | 46 |
| The router | 47 |
| Marshalling | 47 |
| Filter and interceptors | 48 |

| | |
|---|-----|
| @Suspended or asynchronous operation | 49 |
| JPA – the database link | 50 |
| From JPA to the database | 52 |
| Model and implications | 57 |
| The Java Transaction API | 58 |
| Server resources | 59 |
| DataSource configuration | 61 |
| Java EE and performances | 65 |
| Ensure you know the server | 66 |
| Ensure you know your application | 68 |
| Ensure you know your resources | 68 |
| Summary | 69 |
| Chapter 3: Monitor Your Application | 71 |
| Java tools to know what my application is doing | 71 |
| The jcmd command – the small command line utility that does a lot | 72 |
| Thread.print | 74 |
| Memory | 79 |
| JVisualVM – the UI for JVM monitoring | 81 |
| How to connect remotely | 86 |
| Java Mission Control | 87 |
| GlassFish ad hoc monitoring | 92 |
| Libraries monitor your application | 95 |
| Counters, gauges, timers, and more | 96 |
| Instrumenting the code | 98 |
| CDI instrumentation | 99 |
| Implementing your own configurable monitoring interceptor | 101 |
| Javaagent – a complex but powerful instrumentation | 106 |
| Sirona javaagent | 106 |
| SQL Monitoring | 112 |
| Data aggregation | 113 |
| Tracing | 114 |
| APM | 115 |
| Summary | 115 |
| Chapter 4: Application Optimization – Memory Management and Server Configuration | 117 |
| Java and the memory | 117 |
| Garbage collector | 119 |
| Garbage collector algorithms | 121 |
| The serial collector | 121 |

| | |
|---|-----|
| Parallel and parallel compacting collectors | 125 |
| Concurrent Mark Sweep (CMS) | 127 |
| Garbage First (G1) | 128 |
| Common memory settings | 129 |
| Debugging GC behavior | 130 |
| Heap dump | 135 |
| Java EE and the resources | 138 |
| DataSource | 139 |
| Java EE and pools | 142 |
| Java EE and HTTP pools | 143 |
| Java EE implicit features | 143 |
| Summary | 144 |
| Chapter 5: Scale Up – Threading and Implications | 145 |
| Java EE threading model | 145 |
| Thread data and consistency | 150 |
| Concurrent data access | 150 |
| Java and thread safety | 152 |
| Synchronized | 153 |
| Locks | 153 |
| java.util.concurrent | 156 |
| The volatile data | 156 |
| Java EE and threads | 157 |
| CDI asynchronous events | 157 |
| EJB @Asynchronous | 159 |
| EE concurrency utilities for Java EE | 159 |
| ManagedExecutorService | 162 |
| ManagedScheduledExecutorService | 163 |
| Java EE threads | 164 |
| ContextService | 164 |
| EJB @Lock | 165 |
| HTTP threads | 166 |
| Server side | 166 |
| Client side | 170 |
| Reactive programming and Java EE | 171 |
| Message passing and Java EE | 174 |
| Summary | 175 |
| Chapter 6: Be Lazy; Cache Your Data | 177 |
| Caching challenges | 177 |
| How does the cache work? | 179 |
| The cache key | 179 |
| Eviction policy | 183 |

| | |
|--|-----|
| Least Recently Used (LRU) | 183 |
| First In First Out (FIFO) | 184 |
| Random | 185 |
| Least Frequently Used (LFU) | 185 |
| Cache eviction trigger | 185 |
| Cache storage – memory or not | 186 |
| Consistency of the data | 188 |
| HTTP and caching | 189 |
| Cache-Control | 190 |
| ETag | 191 |
| Vary | 192 |
| HTTP caching and the Java EE API | 193 |
| HTTP 2 promise | 195 |
| JCache – the standard cache of Java EE | 195 |
| Setting up JCache | 196 |
| Programmatic API | 197 |
| JCache configuration | 198 |
| CDI integration | 199 |
| Cache configuration | 200 |
| @CacheDefaults | 205 |
| Cache key | 206 |
| Cache once | 208 |
| Summary | 210 |
| Chapter 7: Be Fault-Tolerant | 211 |
| It will fail, no doubt! | 211 |
| Load balancing – pick the best one | 213 |
| Transparent load balancing – client versus server | 214 |
| Common strategies | 215 |
| The round-robin algorithm | 215 |
| Random load balancing | 216 |
| Link to failover | 217 |
| Sticky session | 218 |
| The scheduling algorithm | 219 |
| Load balancing or proxy – additional features | 220 |
| Failover | 221 |
| Switching to another system | 222 |
| Local fallback | 224 |
| Fallback code structure | 225 |
| Microprofile to the rescue | 226 |
| Failover extension | 226 |
| Fallback handling – caching, an alternative solution | 232 |
| Circuit breaker | 237 |

| | |
|--|-----|
| Bulk head | 238 |
| Timeouts | 240 |
| Timeouts for code without timeouts | 241 |
| Summary | 242 |
| Chapter 8: Loggers and Performances – A Trade-Off | 243 |
| I log, you log, they log | 244 |
| Logging frameworks and concepts | 248 |
| Logger | 249 |
| Logger Factory | 249 |
| LogRecord | 250 |
| Handler | 251 |
| Filter | 252 |
| Formatter | 252 |
| Level | 252 |
| Logging invocations | 253 |
| Filters | 254 |
| Formatters | 255 |
| Using handlers | 256 |
| Logging components and Java EE | 257 |
| Logging patterns | 257 |
| Testing your level | 257 |
| Using templates in your messages | 259 |
| Asynchronous or not? | 260 |
| Logging implementations – which one to pick | 261 |
| Logging facade – the good bad idea | 261 |
| Logging frameworks | 262 |
| Log4j | 263 |
| Log4j2 | 264 |
| Logback | 265 |
| JUL | 266 |
| Choosing an implementation | 267 |
| Summary | 268 |
| Chapter 9: Benchmarking Your Application | 269 |
| Benchmarking – validating your application against your SLA | 270 |
| Benchmark, benchmark, benchmark | 270 |
| Preparing your benchmark | 272 |
| Defining the benchmark criteria | 272 |
| Defining the metric | 273 |
| Defining acceptance criteria | 274 |

| | |
|--|-----|
| Defining the scenario | 276 |
| Defining the environment | 276 |
| The machines | 276 |
| The network | 277 |
| Databases and remote services | 278 |
| The server | 279 |
| Automatize your scenarios | 280 |
| JMeter | 280 |
| Gatling | 283 |
| Other tools | 287 |
| Ensuring you have support | 287 |
| Time boxing your optimization benchmark | 288 |
| Installing and resetting procedures | 288 |
| Benchmark iterations | 289 |
| Iterating | 289 |
| Changing a single setting at a time | 290 |
| Resetting any state between runs | 290 |
| Warming up your system | 291 |
| After your benchmark | 291 |
| Writing a report | 292 |
| Updating your default configurations | 293 |
| Reviewing the conclusions | 293 |
| Enriching your project backlog | 294 |
| Summary | 294 |
| Chapter 10: Continuous Performance Evaluation | 295 |
| Writing performance tests – the pitfalls | 295 |
| Writing performance tests | 296 |
| JMH – the OpenJDK tool | 296 |
| Writing a benchmark with JMH | 297 |
| The state | 298 |
| Creating your first JMH benchmark | 299 |
| BlackHole to be a star | 302 |
| Running the benchmarks | 302 |
| Integrating JMH with JUnit | 303 |
| ContiPerf | 311 |
| Arquillian and ContiPerf – the diabolic pair | 312 |
| JMeter and build integration | 316 |
| Gatling and continuous integration | 318 |
| Deploying your (benchmark) application | 322 |
| Continuous integration platform and performance | 324 |
| Summary | 325 |

| | |
|-----------------------------------|-----|
| Another Book You May Enjoy | 327 |
| Index | 329 |
