Netty IN ACTION

Norman Maurer
Marvin Allen Wolfthal

Foreword by Trustin Lee

MANNING

Netty in Action

NORMAN MAURER
MARVIN ALLEN WOLFTHAL

MANNING
SHELTER ISLAND

For online information and ordering of this and other Manning books, please visit www.manning.com. The publisher offers discounts on this book when ordered in quantity. For more information, please contact

Special Sales Department
Manning Publications Co.
20 Baldwin Road
PO Box 761
Shelter Island, NY 11964
Email: orders@manning.com

Manning Publications Co.
20 Baldwin Road
PO Box 761
Shelter Island, NY 11964

Development editors: Jeff Bleiel, Jennifer Stout
Technical development editor: Mark Elston
Copyeditor: Andy Carroll
Proofreader: Elizabeth Martin
Technical proofreaders: David Dossot, Neil Rutherford
Typesetter: Dennis Dalinnik
Cover designer: Marija Tudor

ISBN: 9781617291470
Printed in the United States of America
1 2 3 4 5 6 7 8 9 10 – EBM – 20 19 18 17 16 15

brief contents

contents

foreword

There was a day when people believed that web application servers would let us forget about how we write HTTP or RPC servers. Unfortunately, this daydream did not last long. The amount of load and the pace of the functional changes we are dealing with continue to increase beyond the extent that a traditional three-tier architecture can afford, and we are being forced to split our application into many pieces and distribute them into a large cluster of machines.

Running such a large distributed system leads to two interesting problems—the cost of operation and latency. How many machines could we save if we improved the performance of a single node by 30%, or by more than 100%? How could we achieve minimal latency when a query from a web browser triggers dozens of internal remote procedure calls across many different machines?

In *Netty in Action*, the first-ever book about the Netty project, Norman Maurer, one of the key contributors to Netty, gives you the definitive answers to such questions by showing you how to build a high-performance and low-latency network application in Netty. By the time you get to the end of this book, you'll be able to build every imaginable network application, from a lightweight HTTP server to a highly customized RPC server.

What's impressive about *Netty in Action* is not only that it is written by the key contributor who knows every single bit of Netty, but that it also contains case studies from several companies—Twitter, Facebook, and Firebase to name a few—that use Netty in their production systems. I'm confident that these case studies will inspire you by

showing you how the companies that use them were able to unleash the power of their Netty-based applications.

You might be astonished to learn that Netty started as my personal project back in 2001 when I was an undergraduate student (http://t.motd.kr/ko/archives/1930), and that the project is still alive and kicking today, thanks to enthusiastic contributors like Norman, who spent many sleepless nights devoted to the project, (http://netty.io/community.html). I hope this book opens up another aspect of the project by inspiring its readers to contribute and continue to "open the future of network programming."

TRUSTIN LEE
FOUNDER OF NETTY

preface

Looking back, I still can't believe that I did it.

When I started to contribute to Netty in late 2011, I would never have imagined that I'd be writing a book about Netty and be one of the core developers of the Framework itself.

It all started when I got involved in 2009 with the Apache James project, a Java-based mail server developed under the Apache Software Foundation.

Apache James, like many applications, required a solid networking abstraction to build upon. While investigating the field of projects that provide networking abstractions, I stumbled upon Netty and immediately fell in love with it. As I became more familiar with Netty from a user's perspective, I started to turn my gaze on improvements and giving back to the community.

Despite my first contributions being of limited scope, it became obvious very quickly how beneficial making contributions and the related discussions with the community, especially with Trustin Lee, the founder of the project, were to my personal growth. This experience grabbed hold of me, and I enjoyed spending my free time becoming more engaged in the community. I found myself helping on mailing lists and joining discussions on the IRC channel. Working on Netty began as a hobby but it quickly evolved into a passion.

My passion for Netty eventually led to my employment at Red Hat. This was a dream come true, as Red Hat paid me to work on the project I had come to love. I eventually came to know Claus Ibsen who was working on Apache Camel at the time (and still does). Claus and I came to the realization that Netty had a solid user base

and good JavaDocs, but it lacked a higher level of documentation. Claus is the author of *Camel in Action* (Manning Publications, 2010), and he turned me on to the idea of writing a similar book for Netty. I thought about the idea for a few weeks, and eventually I was sold. This is how *Netty in Action* got off the ground.

While writing *Netty in Action*, my involvement in the community continued to grow. I eventually became the second most active contributor (after Trustin Lee) with over 1,000 commits. I found myself speaking about Netty at conferences and technical meetups around the world. Eventually Netty opened another employment opportunity at Apple Inc. where I am currently employed as a Senior Software Engineer on the Cloud Infrastructure Engineering Team. I continue to work on Netty and make regular contributions back to the community, while helping to drive the project.

NORMAN MAURER
CLOUD INFRASTRUCTURE ENGINEERING, APPLE

My work as a Dell Services consultant at Harvard Pilgrim Health Care in Wellesley, MA, has focused primarily on creating reusable infrastructure components. Our goal is to expand the common code base in a way that not only benefits the software process in general, but also relieves application developers of responsibility for plumbing code that can be both pesky and mundane.

At one point it came to my attention that two related projects were working with a third-party claims-processing system that supported only direct TCP/IP communications. One of the projects needed to reimplement in Java a somewhat underdocumented legacy COBOL module built on the vendor's proprietary delimited format. This module was ultimately replaced by the other project, which would use a more recent XML-based interface to the same claims system (but still using straight sockets, no SOAP!).

This seemed to me an ideal opportunity to develop a common API, and an interesting one at that. I knew there would be stringent throughput and availability requirements and also that the design was still evolving. Clearly, the underlying networking code had to be completely decoupled from the business logic in order to support rapid cycles of iteration.

My research into high-performance networking frameworks for Java led me straight to Netty. (The hypothetical project you'll read about at the start of chapter 1 is pretty much taken from real life.) I soon became convinced that Netty's approach, using dynamically configurable encoders and decoders, was a perfect fit for our needs: both projects would employ the same API, deployed with the handlers needed for the specific data format in use. I became even more convinced when I discovered that the vendor's product was also based on Netty!

Just at that time I learned that the book *Netty in Action* I had been hoping for was actually in the works. I read the early drafts and soon contacted Norman with questions and a few suggestions. In the course of our conversations, we spoke often about

the need to keep the end user's point of view in mind, and since I was currently involved in a live Netty project, I was happy to take on this role.

I hope that with this approach we've succeeded in meeting the needs of developers. If you have any suggestions on how we can make this book more useful, please contact us at https://forums.manning.com/forums/netty-in-action.

MARVIN ALLEN WOLFTHAL
DELL SERVICES

acknowledgments

The Manning team made the work on this book a joy, and they never complained when the joy went on a bit longer than anticipated. From Mike Stephens, who made it happen, and Jeff Bleiel, from whom we learned something new about collaboration, to Jennifer Stout, Andy Carroll, and Elizabeth Martin, who exhibited a level of calm patience that we can only envy, all have upheld standards of professionalism and quality that inspire authors to do their best.

Thanks to the people who helped to review this book, whether by reading the early release chapters and posting corrections in the Author Online forum, or by reviewing the manuscript at various stages during its development. You are part of it and should be proud. Without you, the book would not be the same. Special thanks to the following reviewers: Achim Friedland, Arash Bizhan Zadeh, Bruno Georges, Christian Bach, Daniel Beck, Declan Cox, Edward Ribeiro, Erik Onnen, Francis Marchi, Gregor Zurowski, Jian Jin, Jürgen Hoffmann, Maksym Prokhorenko, Nicola Grigoletti, Renato Felix, and Yestin Johnson. Also to our excellent technical proofreaders, David Dossot and Neil Rutherford.

We are very grateful to and would like to acknowledge Bruno de Carvalho, Sara Robinson, Greg Soltis, Erik Onnen, Andrew Cox, and Jeff Smick for contributing the case studies you will find in chapters 14 and 15.

Last but not least, thanks to all the people who support Netty and OpenSource in general; without you, the community, this project wouldn't have been possible. Through the community we have met new friends, talked at conferences all over the world, and grown both professionally and personally.

Norman Maurer

I'd like to thank my former coworker and friend Jürgen Hoffmann (aka Buddy). Jürgen helped me find my way into the OpenSource world and showed me what cool stuff can be built when you are brave enough to jump in. Without him there is a chance I would have never started programming and thus never found my true professional passion.

Another big thank you goes out to my friend Trustin Lee, founder of Netty, who helped and encouraged me to contribute to the Netty Project in the first place and who penned the foreword to our book. I'm honored to know you and to be able to call you a friend! I'm confident that by continuing to work together, Netty will stay awesome and be around for a long time!

Also I want to thank my coauthor Marvin Wolfthal. Despite Marvin joining the project late in its lifecycle, he helped to improve the overall structure and content by a considerable amount. Without him, the book would not have been possible in its current form. Which brings me to the Manning team itself, who were always helpful and gave the right guidance to make the idea of writing a book reality.

Thanks to my parents, Peter and Christa, for always supporting my visions and me.

Most importantly, thanks to my wife Jasmina and my kids Mia Soleil and Ben, for all the support they showed during the process of writing this book. Without you this book would not have been possible.

Marvin Wolfthal

First of all, I want to thank Norman Maurer, my coauthor, for both his fine work and his kindness. Though I came late to the project, I was made to feel as if I had been part of it from day one.

To my colleagues past and present at Dell Services and Harvard Pilgrim Health Care, I offer sincere thanks for their help and encouragement. They have created that rare environment where new ideas are not only expressed, but realized. To Deborah Norton, Larry Rapisarda, Dave Querusio, Vijay Bhatt, Craig Bogovich, and Sharath Krishna, special thanks for their support, and even more, their trust—I doubt many software developers have been granted the creative opportunities I have enjoyed in the past four years, including adding Netty to our toolkit.

But most of all, thanks to my beloved wife, Katherine, who never lets me forget the things that really matter.

about this book

Netty is a Java framework for the rapid development of high-performance network applications. It encapsulates the complexities of network programming and makes the most recent advances in networking and web technologies accessible to a broader range of developers than ever before.

Netty is more than a collection of interfaces and classes; it also defines an architectural model and a rich set of design patterns. But until now, the lack of a comprehensive and systematic user's guide has been an obstacle to getting started with Netty, a situation *Netty in Action* aims to remedy. Beyond explaining the details of the framework components and APIs, this book will show how Netty can help you to write more efficient, reusable, and maintainable code.

Who should read this book?

This book assumes you are comfortable with intermediate Java topics such as generics and multithreading. Experience with advanced network programming is not required, but familiarity with the basic Java networking APIs will prove very helpful.

Netty uses Apache Maven as its build management tool. If you have not used Maven, the appendix will provide the information you need to run the book's sample code. You'll also be able to reuse the sample Maven configurations as starting points for your own Netty-based projects.

Roadmap

Netty in Action has four parts and an appendix.

Part 1: Netty concepts and architecture

Part 1 is a detailed guide to the framework, covering its design, components, and programming interfaces.

Chapter 1 begins with a brief overview of the blocking and non-blocking network APIs and the corresponding JDK interfaces. We introduce Netty as a toolkit for building highly scalable, asynchronous and event-driven networking applications. We take a first look at the basic building blocks of the framework: channels, callbacks, futures, events, and handlers.

Chapter 2 explains how to configure your system for building and running with the book's sample code. We test it out with a simple application, a server that echoes the messages it receives from connected clients. We introduce bootstrapping—assembling and configuring all the components of an application at runtime.

Chapter 3 begins with a discussion of the technical and architectural aspects of Netty. The core components of the framework are introduced: `Channel`, `EventLoop`, `ChannelHandler`, and `ChannelPipeline`. The chapter concludes with an explanation of the differences between bootstrapping servers and clients.

Chapter 4 discusses network transports and contrasts the use of blocking and non-blocking transports with the JDK APIs and with Netty. We study the interface hierarchy underlying Netty's transport API and the transport types they support.

Chapter 5 is devoted to the framework's data handling API—`ByteBuf`, Netty's byte container. We describe its advantages over the JDK's `ByteBuffer`, and the ways in which the memory used by a `ByteBuf` can be allocated and accessed. We show how to manage memory resources using reference counting.

Chapter 6 focuses on the core components `ChannelHandler` and `ChannelPipeline`, which are responsible for dispatching application processing logic and moving data and events through the network layer. Additional topics include the role of `ChannelHandlerContext` in implementing advanced use cases and the sharing of `ChannelHandlers` among multiple `ChannelPipelines`. The chapter concludes with an illustration of handling exceptions triggered by inbound and outbound events.

Chapter 7 provides a general overview of threading models and covers Netty's threading model in detail. We examine `interface EventLoop`, which is the principal component of Netty's concurrency API, and explain its relationship with threads and `Channels`. This information is essential for understanding how Netty implements asynchronous, event-driven networking. We show how to perform task scheduling using `EventLoop`.

Chapter 8 explores bootstrapping in depth, starting with the `Bootstrap` class hierarchy. We revisit the basic use cases as well as some special ones, such as bootstrapping a client connection within a server application, bootstrapping datagram channels, and adding multiple channels during the bootstrapping phase. The chapter concludes with a discussion of how to shut down an application gracefully and release all resources in an orderly fashion.

Chapter 9 is about unit testing `ChannelHandlers`, for which Netty provides a special `Channel` implementation, `EmbeddedChannel`. The examples show how to use this class with JUnit to test both inbound and outbound handler implementations.

Part 2: Codecs

Data conversion is one of the most common operations in network programming. Part 2 describes the rich set of tools Netty provides to simplify this task.

Chapter 10 begins by explaining decoders and encoders, which transform sequences of bytes from one format to another. A ubiquitous example is converting an unstructured byte stream to and from a protocol-specific layout. A codec, then, is a component that combines both an encoder and a decoder in order to handle conversions in both directions. We provide several examples to show how easy it is to create custom decoders and encoders with Netty's codec framework classes.

Chapter 11 examines the codecs and `ChannelHandlers` Netty provides for a variety of use cases. These classes include ready-to-use codecs for protocols such as SSL/TLS, HTTP/HTTPS, WebSocket, and SPDY, and decoders that can be extended to handle almost any delimited, variable length or fixed-length protocol. The chapter concludes with a look at framework components for writing large volumes of data and for serialization.

Part 3: Network protocols

Part 3 elaborates on several network protocols that have been touched on briefly earlier in the book. We'll see once again how Netty makes it easy to adopt complex APIs in your applications without having to be concerned with their internal complexities.

Chapter 12 shows how to use the WebSocket protocol to implement bidirectional communications between a web server and client. The example application is a chat room server that allows all connected users to communicate with one another in real time.

Chapter 13 illustrates Netty's support for connectionless protocols with a server and client that utilize the broadcast capabilities of the User Datagram Protocol (UDP). As in the previous examples, we employ a set of protocol-specific support classes: `DatagramPacket` and `NioDatagramChannel`.

Part 4: Case studies

Part 4 presents five case studies submitted by well-known companies that have used Netty to implement mission-critical systems. These examples illustrate not only real-world usages of the framework components we have discussed throughout the book, but also the application of Netty's design and architectural principles to building highly scalable and extensible applications.

Chapter 14 has case studies submitted by Droplr, Firebase, and Urban Airship.

Chapter 15 has case studies submitted by Facebook and Twitter.

Appendix: Introduction to Maven

The primary goal of the appendix is to provide a basic introduction to Apache Maven so that you can compile and run the book's sample code listings and extend them to create your own projects as you begin to work with Netty.

The following topics are presented:

- The primary goals and uses of Maven
- Installing and configuring Maven
- Basic Maven concepts: the POM file, artifacts, coordinates, dependencies, plugins, and repositories
- Example Maven configurations, POM inheritance, and aggregation
- Maven's command-line syntax

Code conventions and downloads

This book provides copious examples that show how you can make use of each of the topics covered. Source code in listings or in text appears in a `fixed-width font like this` to separate it from ordinary text. In addition, class and method names, object properties, and other code-related terms and content in text are presented using `fixed-width font`.

Occasionally, code is italicized, as in *`reference`*`.dump()`. In this case *`reference`* should not be entered literally but replaced with the content that is required.

The book's source code is available from the publisher's website at www.manning.com/books/netty-in-action and at GitHub: https://github.com/normanmaurer/netty-in-action. It is structured as a multimodule Maven project, with a top-level POM and modules corresponding to the book's chapters.

About the authors

Norman Maurer is one of the core developers of Netty, a member of the Apache Software Foundation, and a contributor to many OpenSource Projects over the past years. He's a Senior Software Engineer for Apple, where he works on Netty and other network-related projects as part of the iCloud Team.

Marvin Wolfthal has been active in many areas of software development as a developer, architect, lecturer, and author. He has been working with Java since its earliest days and assisted Sun Microsystems in developing its first programs dedicated to promoting distributed object technologies. As part of these efforts he wrote the first cross-language programming courses using C++, Java, and CORBA for Sun Education. Since then his primary focus has been middleware design and development, primarily for the financial services industry. Currently a consultant with Dell Services, he is engaged in extending methodologies that have emerged from the Java world to other areas of enterprise computing; for example, applying the practices of Continuous Integration to database development. Marvin is also a pianist and composer whose work is