

Extracted from:

Programming Clojure

This PDF file contains pages extracted from Programming Clojure, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.com>.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printer versions; the content is otherwise identical.

Copyright © 2009 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

The
Pragmatic
Programmers

Programming Clojure



Stuart Halloway

Edited by Susannah Davidson Pfalzer



Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

Our Pragmatic courses, workshops, and other products can help you and your team create better software and have more fun. For more information, as well as the latest Pragmatic titles, please visit us at

<http://www.pragprog.com>

Copyright © 2009 Stuart Halloway.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

Printed in the United States of America.

ISBN-10: 1-934356-33-6

ISBN-13: 978-1-934356-33-3

Printed on acid-free paper.

P2.0 printing, September 2009

Version: 2009-10-27

Foreword

We are drowning in complexity. Much of it is incidental—arising from the way we are solving problems, instead of the problems themselves. Object-oriented programming seems easy, but the programs it yields can often be complex webs of interconnected mutable objects. A single method call on a single object can cause a cascade of change throughout the object graph. Understanding what is going to happen when, how things got into the state they did, and how to get them back into that state in order to try to fix a bug are all very complex. Add concurrency to the mix, and it can quickly become unmanageable. We throw mock objects and test suites at our programs but too often fail to question our tools and programming models.

Functional programming offers an alternative. By emphasizing pure functions that take and return immutable values, it makes side effects the exception rather than the norm. This is only going to become more important as we face increasing concurrency in multicore architectures. Clojure is designed to make functional programming approachable and practical for commercial software developers. It recognizes the need for running on trusted infrastructure like the JVM and supporting existing investments made by customers in Java frameworks and libraries, as well as the immense practicality of doing so.

What is so thrilling about Stuart’s book is the extent to which he “gets” Clojure, because the language is targeted to professional developers just like himself. He clearly has enough experience of the pain points Clojure addresses, as well as an appreciation of its pragmatic approach. This book is an enthusiastic tour of the key features of Clojure, well grounded in practical applications, with gentle introductions to what might be new concepts. I hope it inspires you to write software in Clojure that you can look back at and say, “Not only does this do the job, but it does so in a robust and simple way, and writing it was fun too!”

—Rich Hickey
Creator of Clojure

The Pragmatic Bookshelf

The Pragmatic Bookshelf features books written by developers for developers. The titles continue the well-known Pragmatic Programmer style and continue to garner awards and rave reviews. As development gets more and more difficult, the Pragmatic Programmers will be there with more titles and products to help you stay on top of your game.

Visit Us Online

Programming Clojure's Home Page

<http://pragprog.com/titles/shcloj>

Source code from this book, errata, and other resources. Come give us feedback, too!

Register for Updates

<http://pragprog.com/updates>

Be notified when updates and new books become available.

Join the Community

<http://pragprog.com/community>

Read our weblogs, join our online discussions, participate in our mailing list, interact with our wiki, and benefit from the experience of other Pragmatic Programmers.

New and Noteworthy

<http://pragprog.com/news>

Check out the latest pragmatic developments, new titles and other offerings.

Buy the Book

If you liked this eBook, perhaps you'd like to have a paper copy of the book. It's available for purchase at our store: pragprog.com/titles/shcloj.

Contact Us

| | |
|-----------------------|--|
| Online Orders: | www.pragprog.com/catalog |
| Customer Service: | support@pragprog.com |
| Non-English Versions: | translations@pragprog.com |
| Pragmatic Teaching: | academic@pragprog.com |
| Author Proposals: | proposals@pragprog.com |
| Contact us: | 1-800-699-PROG (+1 919 847 3884) |