

**Stefan Kamphausen · Tim Oliver Kaiser**

# **Clojure**

**Grundlagen, Concurrent Programming, Java**



**dpunkt.verlag**

Stefan Kamphausen  
ska@clojure-buch.de

Tim Oliver Kaiser  
tok@clojure-buch.de

Lektorat: Dr. Michael Barabas  
Copy-Editing: Annette Schwarz, Ditzingen  
Satz: Stefan Kamphausen  
Herstellung: Nadine Thiele  
Titelbild: Elodie Kamphausen  
Umschlaggestaltung: Helmut Kraus, [www.exclam.de](http://www.exclam.de)  
Druck und Bindung: Media-Print Informationstechnologie, Paderborn

Bibliografische Information der Deutschen Nationalbibliothek  
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie;  
detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

ISBN 978-3-89864-684-0

1. Auflage 2010  
Copyright © 2010 dpunkt.verlag GmbH  
Ringstraße 19 B  
69115 Heidelberg

Die vorliegende Publikation ist urheberrechtlich geschützt. Alle Rechte vorbehalten. Die Verwendung der Texte und Abbildungen, auch auszugsweise, ist ohne die schriftliche Zustimmung des Verlags urheberrechtswidrig und daher strafbar. Dies gilt insbesondere für die Vervielfältigung, Übersetzung oder die Verwendung in elektronischen Systemen.

Es wird darauf hingewiesen, dass die im Buch verwendeten Soft- und Hardware-Bezeichnungen sowie Markennamen und Produktbezeichnungen der jeweiligen Firmen im Allgemeinen warenzeichen-, marken- oder patentrechtlichem Schutz unterliegen.

Alle Angaben und Programme in diesem Buch wurden mit größter Sorgfalt kontrolliert. Weder Autor noch Verlag können jedoch für Schäden haftbar gemacht werden, die in Zusammenhang mit der Verwendung dieses Buches stehen.

5 4 3 2 1 0

# Vorwort

Es ist fünf Jahre her, dass Herb Sutter in einem viel beachteten Artikel die »Concurrency Revolution« ausgerufen hat, nicht leugnend, dass das Thema Concurrency (Nebenläufigkeit) seit Jahrzehnten eine wichtige Rolle in der Informatik spiele – bislang aber nur von wenigen Programmierern und Programmiererninnen verwendet, geschweige denn beherrscht werde. Dies werde sich ändern, da die Entwicklung immer schnellerer Prozessoren mit immer höheren Taktfrequenzen an eine physikalische Grenze stoße. In der Tat werden Leistungssteigerungen zunehmend durch den Einsatz von Hyperthreading-Techniken und Mehrkernprozessoren erreicht – ein Trend, der sich auch in der Verwendung massiv paralleler Grafikprozessoren für umfangreiche Rechenaufgaben manifestiert.

Diese Entwicklung macht es wünschenswert, nebenläufige Programmierung im Informatikstudium an Hochschulen möglichst frühzeitig und praxisbezogen einzuführen. Clojure bietet sich als Basis an, indem es von Grund auf für die Behandlung nebenläufiger Probleme geschaffen ist, gleichzeitig jedoch auf Java aufsetzt, welches sich als erste Programmiersprache an vielen Hochschulen etabliert hat. Als positiven Nebeneffekt erlaubt Clojure einen Einblick in die funktionale Programmierung und schlägt damit eine Brücke zur Theoretischen Informatik.

Der Name Clojure weckt vielfältige Assoziationen. Die Nähe zu *Closure* lässt zunächst an eher profane Dinge wie einen mathematischen Abschluss oder ein Konzept funktionaler Programmiersprachen denken. Französisch ausgesprochen mag *Clojure* auch nach einem gutem Wein und einem delikaten Essen klingen. Nicht abwegig – und vielleicht am schönsten – ist jedoch die Verwandtschaft mit *Conjuration*: Möge dieses Buch viele Leserinnen und Leser dazu anregen, mit Clojure zu zaubern.

Prof. Dr. Volker Ahlers  
Fachhochschule Hannover  
im Mai 2010

---

# Geleitwort von Hans Dockter

Nachdem die Java-Community viele Jahre im eigenen Saft schmor-  
te, richtet sie seit einiger Zeit viel Aufmerksamkeit auf andere JVM-  
Sprachen. Die Konzepte letzterer sind ein zentrales Element in der  
Diskussion um die zukünftige Evolution von Java. Zurecht!

Clojure ist eine der wichtigen neuen Sprachen für die JVM. Eine  
Stärke von Clojure ist, dass grundlegende Sprachkonzepte elementa-  
ren Mustern des Software-Designs entsprechen (z. B. Side-Effect-Free-  
Functions, Value- und Entity-Objekte, etc.). Das macht sie u. a. zu  
einer sehr guten Lehrerin und stärkt einen auch darin, diese Muster in  
der diesbezüglich sehr viel verwascheneren Java-Sprache umzusetzen.  
Clojure bietet als Sprache natürlich noch viel mehr. Darüber hinaus  
macht ihre tiefe Integration mit Java sie zu einer realen Option für den  
Unternehmenseinsatz.

Dieses Buch gibt eine sehr sorgfältige Einführung in die Sprache  
und bietet sich mit seiner Detailtiefe immer wieder zum Nachschlagen  
an. Viel Spaß in der Clojure-Wunderwelt!

Hans Dockter  
Gründer des Build-Systems Gradle  
Korsika im August 2010