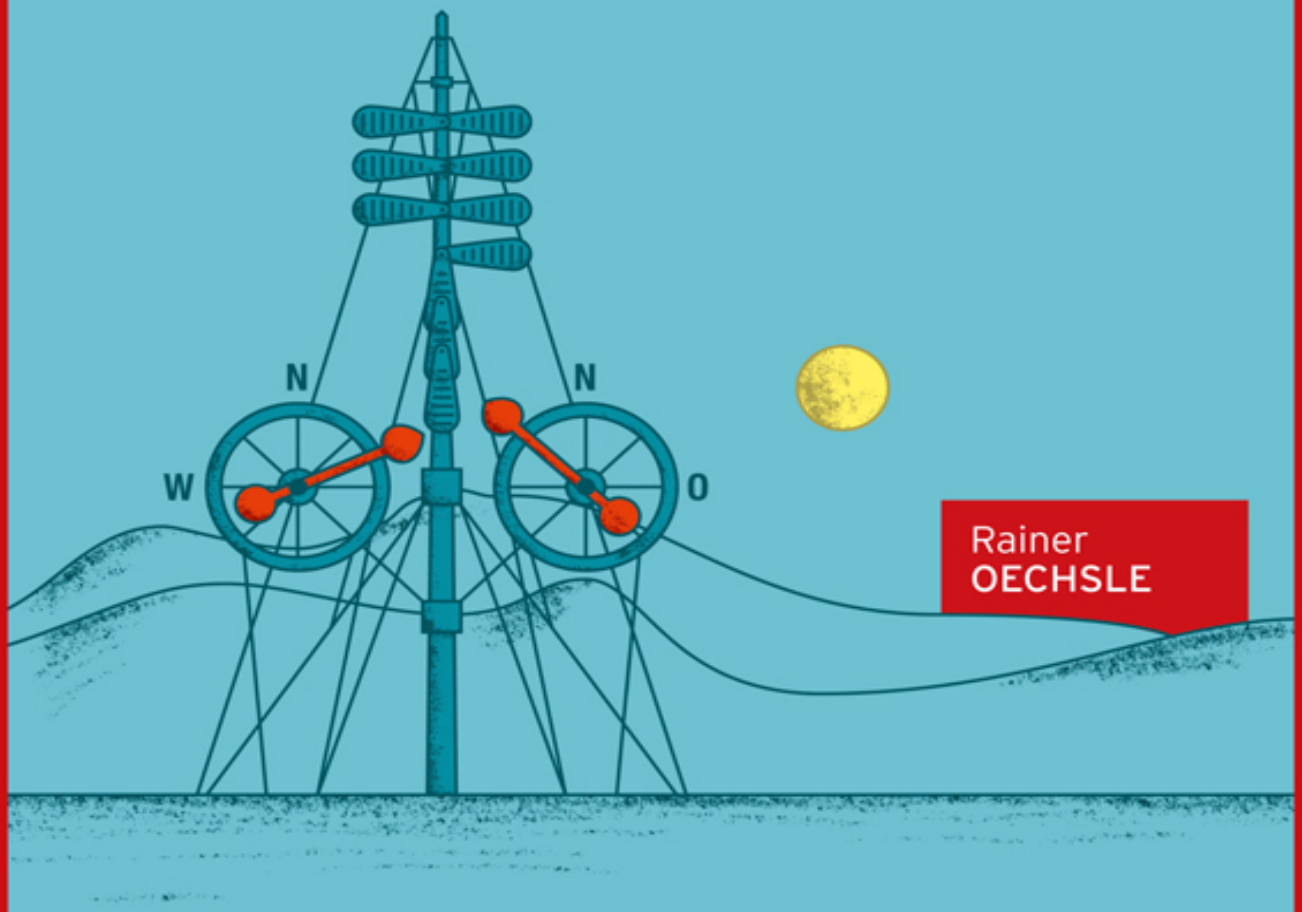
6., überarbeitete und erweiterte Auflage



Rainer
OECHSLE

Parallele und verteilte Anwendungen in JAVA

Alle Beispielprogramme zum Download auf plus.hanser-fachbuch.de

HANSER

HANSER

Rainer Oechsle

Parallele und verteilte Anwendungen in Java

6., aktualisierte Auflage

Ihr Plus – digitale Zusatzinhalte!

Auf unserem Download-Portal finden Sie zu diesem Titel kostenloses Zusatzmaterial.

Geben Sie auf **plus.hanser-fachbuch.de** einfach diesen Code ein:

plus-n72ed-6shfd

Der Autor:

Prof. Dr. Rainer Oechsle, Hochschule Trier

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.

Internet: www.hanser-fachbuch.de
Lektorat: Dipl.-Ing. Natalia Silakova-Herzberg
Herstellung: Frauke Schafft
Covergestaltung: Max Kostopoulos

Coverkonzept: Marc Müller-Bremer, www.rebranding.de, München
Titelbild: © Max Kostopoulos

Print-ISBN 978-3-446-46919-8
E-Book-ISBN 978-3-446-47348-5
E-Pub-ISBN 978-3-446-47504-5

Vorwort zur 6. Auflage

Dieses Buch handelt von der Entwicklung paralleler und verteilter Anwendungen in Java. Nach einem einleitenden Kapitel, in dem wichtige Begriffe wie Programme, Prozesse und Threads auch anhand einer Metapher aus dem täglichen Leben erläutert werden, lassen sich die folgenden acht Kapitel in drei Teile gruppieren:

- Entwicklung paralleler Anwendungen in Java (Kapitel 2 und 3): Das Buch beginnt mit einer relativ ausführlichen Einführung in das Gebiet der parallelen Programmierung. Die Sachverhalte sind für Neulinge oft anspruchsvoll, denn Programmcode, der bei rein sequenzieller Ausführung korrekt ist, kann im Fall einer parallelen Nutzung fehlerbehaftet sein. Der Einstieg in das Thema Parallelität wird im Zusammenhang mit Objektorientierung für viele noch problematischer. Denn zum einen muss man verstehen, dass es Thread-Objekte gibt, dass diese aber nicht identisch mit den parallelen Aktivitäten, den Threads selbst, sind. Zum anderen muss man begreifen lernen, dass es mehrere Objekte einer Klasse geben kann, dass aber ein einziges Objekt (quasi) gleichzeitig von mehreren Threads verwendet werden kann, d. h., dass dieselbe und

unterschiedliche Methoden auf einem Objekt gleichzeitig parallel ausgeführt werden können. In diesem Buch wird in die Gedankenwelt der Parallelität mit zahlreichen Programmbeispielen behutsam eingeführt (Kapitel 2). Es werden dann aber auch die grundlegenden Ideen weiterer anspruchsvoller Konzepte aus der Java-Concurrent-Bibliothek wie das Fork-Join-Framework, sequenzielles und paralleles Data-Streaming sowie CompletableFutures behandelt, ohne auf alle Details dieser Konzepte einzugehen (Kapitel 3).

- Entwicklung von Anwendungen mit grafischer Benutzeroberfläche in Java (Kapitel 4): Grafische Benutzeroberflächen scheinen auf den ersten Blick nichts mit parallelen und verteilten Anwendungen zu tun zu haben. Bei näherem Hinsehen erkennt man aber durchaus Zusammenhänge. So wird zum Beispiel in diesem Buch ausführlich erläutert, welche negativen Effekte es bei naiver Programmierung für Anwendungen mit grafischer Benutzeroberfläche gibt, wenn eine länger dauernde Aktivität aufgrund einer Interaktion mit der grafischen Benutzeroberfläche gestartet wird. Insbesondere bei verteilten Anwendungen können länger dauernde Aktivitäten immer bei einer Kommunikation zwischen einem Client und einem Server über das Internet vorkommen. Die Probleme lassen sich mithilfe von Parallelität lösen. Da Client-Programme oft und Server-Programme manchmal eine grafische Benutzeroberfläche haben, spielt also das Thema Parallelität bei verteilten Anwendungen mit grafischer Benutzeroberfläche eine Rolle. Aber auch wichtige Strukturierungsprinzipien für lokale Programme mit grafischer Benutzeroberfläche wie das MVP-

Architekturmuster (MVP: Model View Presenter) lassen sich auf verteilte Anwendungen übertragen.

- Entwicklung verteilter Anwendungen in Java (Kapitel 5 bis 9): Verteilte Anwendungen folgen häufig dem Client-Server-Prinzip. Auch hier besteht wieder ein enger Zusammenhang zur Parallelität, denn auf Server-Seite ist fast immer Parallelität notwendig, um mehrere Clients (quasi) gleichzeitig zu bedienen und somit die Bedienung eines Clients nicht beliebig lange durch die Bearbeitung eines länger dauernden Auftrags eines anderen Clients zu verzögern. Um die parallele Bearbeitung von Client-Aufträgen zu erreichen, müssen die Threads bei der Programmierung eines Servers auf Socket-Basis (Kapitel 5) selbst explizit erzeugt werden. Wenn RMI (Kapitel 6) oder Servlets und Java Server Faces (Kapitel 8) benutzt werden, dann werden Threads implizit (d. h. nicht im Programmcode der Anwendung) erzeugt. Dies muss man wissen und den Umgang damit beherrschen, wenn man korrekte Server-Programme schreiben will. Wenn die Kommunikation zwischen den Rechnern nicht mehr direkt, sondern über einen Vermittler (Broker) erfolgt, erreicht man eine losere Kopplung zwischen den kommunizierenden Parteien mit einigen Vorteilen (Kapitel 7). Viele verteilte Anwendungen nutzen heutzutage Cloud-Dienste. Es werden abschließend einige Cloud-Anwendungen unter Nutzung der Cloud-Dienste von Amazon präsentiert (Kapitel 9).

In vielen Lehrbüchern werden Parallelitätsaspekte bei Programmen mit grafischer Benutzeroberfläche oder bei Server-Programmen nicht genügend oder überhaupt nicht berücksichtigt. So habe ich einige Beispiele in Lehrbüchern gefunden, die das Thema Parallelität vollständig ignorieren.

Folglich sind solche Programme mit fehlender Synchronisation oftmals nicht korrekt, was beim oberflächlichen Ausprobieren in der Regel (zum Glück oder leider?) nicht auffällt. In diesem Buch wird dagegen durchgängig für alle Anwendungen ein besonderes Augenmerk auf Parallelitätsaspekte gelegt.

Dieses Buch ist weder ein Handbuch mit allen Details, die man bei der Software-Entwicklung benötigt, noch ist es ein Überblicksbuch, in dem eine Fülle von Themen nur angerissen wird. Stattdessen versucht es seinem Charakter als Lehrbuch gerecht zu werden, indem es die Grundprinzipien zentraler Konzepte herausarbeitet. Der Fokus liegt auf den beiden eng miteinander verzahnten Themen Parallelität (Nebenläufigkeit) und Verteilung. Bei dem Thema parallele Programmierung ist es in der Praxis sicher ratsam, nach Möglichkeit die Klassen aus der Java-Concurrent-Klassenbibliothek zu verwenden, die in Kapitel 3 auch behandelt werden. Allerdings wird die Mehrzahl der Beispiele aus didaktischen Gründen ohne Nutzung dieser Bibliothek entwickelt. Ähnliches trifft auf Servlets und JSF (Java Server Faces) zu: In der Praxis wird man eher JSF nutzen, im Buch werden die meisten Beispiele mit Servlets entwickelt, um den Leserinnen und Lesern besser verständlich zu machen, was bei der Ausführung des Programms passiert. Auch in dieser Auflage behandle ich immer noch RMI sehr intensiv. Man mag der Meinung sein, dass RMI inzwischen veraltet ist, aber aus meiner Sicht ist RMI weiterhin eine sehr elegante und konsequent zu Ende gedachte Realisierung einer Client-Server-Kommunikation. Ich bin überzeugt davon, dass die intensive Beschäftigung mit RMI elementar wichtige Aspekte der Informatik wie zum Beispiel den Unterschied zwischen Call-by-value und Call-by-result gut verständlich macht. So ist die Beschäftigung mit den hier vorhandenen Inhalten nicht nur dazu da, um aktuell notwendige

Kenntnisse und Fertigkeiten für die Berufswelt zu erlernen, sondern vor allem zum Erlernen grundlegender Informatikkonzepte. Aus diesem Grund ist die hier verwendete Programmiersprache Java auch nur ein Vehikel zur Darstellung unterschiedlicher Aspekte aus dem Bereich der Programmierung paralleler und verteilter Anwendungen. So wie dieselben Inhalte dieses Buchs statt in Deutsch in einer anderen Sprache wie Englisch oder Französisch vermittelt werden könnten, so ließen sich viele der vorgestellten Konzepte auch durch Programmbeispiele in anderen Programmiersprachen wie etwa C++ oder C# illustrieren.

Bei der Auswahl des Lehrstoffs ist es immer auch schmerzlich, viele interessante und relevante Themen nicht tiefer oder gar nicht behandeln zu können aufgrund des begrenzten Umfangs des Buchs. So könnte das neue Kapitel 9 zum Thema Cloud Computing auch leicht so ausgedehnt werden, dass es ein ganzes Buch füllen würde. Zu den leider gar nicht behandelten Themen gehören beispielsweise die Bereiche Spring Boot und Kubernetes. Bei der Stoffauswahl muss man eben Kompromisse eingehen.

Für diese sechste Auflage wurden neben der Korrektur von Fehlern, die in der fünften Auflage bemerkt wurden, folgende wesentlichen Änderungen vorgenommen:

- Das Kapitel 7 zur indirekten Kommunikation wurde neu geschrieben.
- Außerdem ist Kapitel 9 zum Thema Cloud Computing neu dazugekommen.
- In Kapitel 5 wird im neu geschriebenen Abschnitt 5.7 nun auch verschlüsselte Kommunikation über SSL (Secure

Socket Layer) bzw. TLS (Transport Layer Security) behandelt.

- Da Java EE (Enterprise Edition) von der Firma Oracle nicht mehr weiterentwickelt wird, sondern dies von der Eclipse Foundation übernommen wurde und nun den Namen Jakarta EE trägt, wurden Bezüge auf die Enterprise Edition von Java EE in Jakarta EE geändert.
- Der Abschnitt 6.8 (Laden von Klassen über das Netz) im RMI-Kapitel der fünften Auflage wurde ersatzlos gestrichen, da der dort benutzte Security Manager seit der Java-Version 17 „deprecated“ ist (d. h. nicht mehr benutzt werden sollte, da er in einer späteren Version nicht mehr vorhanden sein könnte).
- Ferner wurden mehrere kleinere Anpassungen vorgenommen wie z. B. die Ergänzung des Abschnitts 2.1.4 über parallele Abläufe oder die Ergänzung des Abschnitts 5.6.4 über horizontale Skalierung.

Die Beispielprogramme folgen gängigen Programmierkonventionen für Java bezüglich der Groß- und Kleinschreibung von Bezeichnern und dem Einrücken. Alle Bezeichner für Klassen, Schnittstellen, Methoden und Attribute sind einheitlich in Englisch geschrieben. Die Ausgaben, die von den Programmen erzeugt werden, sind jedoch alle in deutscher Sprache. In den abgedruckten Programmen wurden alle Package-Anweisungen entfernt (bis auf eine Ausnahme in Kapitel 9, Listing 9.6, weil bei diesem Beispiel auf den Package-Namen explizit Bezug genommen wird). Beachten Sie aber bitte, dass in der elektronischen Version, die Sie von einer der Webseiten zum Buch *www.hochschule-trier.de/puva6* (puva: **p**arallele **u**nd **v**erteilte **A**nwendungen) und *plus.hanser-fachbuch.de* beziehen

können, die Klassen und Schnittstellen kapitelweise in unterschiedliche Packages gruppiert wurden (chapter2, chapter3 usw.). Alle Java-Programme wurden mit einem Java-Compiler der Version 14 übersetzt und ausprobiert.

Von den soeben bereits erwähnten Webseiten *www.hochschule-trier.de/puva6* und *plus.hanser-fachbuch.de* können Sie nicht nur alle Programme des Buchs in Form einer ZIP-Datei herunterladen. Auch nachträglich entdeckte Fehler werde ich mitsamt ihren Richtigstellungen und den Namen der Entdeckenden wie für die vorhergehende Auflage auf dieser Seite veröffentlichen. Ich habe zwar für diese Auflage alle entdeckten Fehler korrigiert, aber es ist nicht auszuschließen, dass bisher unentdeckte alte Fehler noch zutage treten werden, und ich bin mir sehr sicher, dass ich bei der Überarbeitung der alten Texte und dem Schreiben der neuen Texte unabsichtlich neue Fehler eingebaut habe. Ich bin allen Leserinnen und Lesern dankbar für alle Arten von Fehlermeldungen, sowohl für die Meldung gravierender Fehler als auch einfacher Komma-, Tipp- oder Formatierungsfehler. Kommentare, Verbesserungsvorschläge und weitere Programmbeispiele, die Sie mir gerne senden können, werde ich ebenfalls auf dieser Webseite veröffentlichen, sofern sie mir für einen größeren Kreis interessant erscheinen.

Meinen Wunsch, geschlechtsneutrale Formulierungen zu verwenden, habe ich so umgesetzt, dass ich an manchen Stellen die männliche und weibliche Form angebe, an anderen Stellen nur die männliche oder nur die weibliche Form. Ich hoffe, dass sich dadurch Lesende beiderlei Geschlechts in gleicher Weise angesprochen fühlen.

Sollten Sie tiefer in die Thematik dieses Buches einsteigen wollen, dann empfehle ich Ihnen, das Modul „Fortgeschrittene

Programmiertechniken (FOPT)“ im Rahmen des Informatik-Fernstudiums an der Hochschule Trier zu belegen. Hier können Sie zu den Themen dieses Buches Einsendeaufgaben bearbeiten, an zusätzlichen Tutorien (per Videokonferenz) teilnehmen sowie ein einwöchiges Präsenzpraktikum absolvieren. Nähere Informationen hierzu, insbesondere über die Voraussetzungen für die Belegung, über die Kosten sowie über die weiteren Module des Fernstudiums, finden Sie auf den Webseiten des Fachbereichs Informatik der Hochschule Trier (*www.hochschule-trier.de/informatik*).

Diese sechste Auflage hätte ohne die Hilfe der nachfolgend genannten Personen nicht bzw. nicht in dieser Form erscheinen können. Ich bedanke mich daher sehr gerne

- bei den für dieses Buch verantwortlichen Mitarbeiterinnen des Hanser-Verlags, Natalia Silakova und Christina Kubiak, für das Ergreifen der Initiative zur sechsten Auflage dieses Buchs, für die Möglichkeit der Erweiterung des Umfangs des Buchs um ca. 20 %, für die Umsetzung meines Vorschlags eines Semaphors als Titelbild sowie für die jederzeit schnelle Klärung aller meiner Fragen,
- bei Daniel Aggintus, Andreas Daum, Fabian Gibert, Robin Grell, Yanik Kaypinger, Marc Kutscher, Frank Seeger, Gunnar Sperveslage, Timo Vollmert und Thomas Zehrer für ihre Hinweise auf entdeckte Fehler und ihre Verbesserungsvorschläge, die alle auf der Webseite *www.hochschule-trier.de/puva5* veröffentlicht und in dieser sechsten Auflage berücksichtigt wurden,
- bei Stefan Bodenschatz, der mit mir zusammen an der Hochschule Trier eine Lehrveranstaltung zum Thema Cloud Computing aufgebaut und mehrfach durchgeführt hat, für

seine zahlreichen Verbesserungsvorschläge zu einer früheren Fassung von Kapitel 9,

- und schließlich bei meiner Frau Ingrid für die gewährte Zeit zur Überarbeitung des Buchs.

Über positive und negative Bemerkungen zu diesem Buch, Hinweise auf Fehler und Verbesserungsvorschläge würde ich mich auch dieses Mal wieder freuen. Senden Sie Ihre Kommentare bitte in Form einer elektronischen Post an *oechsle@hochschule-trier.de*.

Konz-Oberemmel, im Februar 2022

Rainer Oechsle

Inhaltsverzeichnis

9 Verteilte Anwendungen in der Cloud

9.1 Cloud Computing

9.2 AWS (Amazon Web Services)

9.2.1 AWS-Infrastruktur

9.2.2 AWS-Dienste

9.2.3 Nutzung der AWS-Dienste

9.3 Nutzung der AWS-Dienste von außerhalb der Cloud

9.3.1 Nutzung des AWS-Dienstes S3

9.3.2 Nutzung des AWS-Dienstes DynamoDB

9.3.3 Nutzung des AWS-Dienstes Translate

9.4 Nutzung von EC2 als Server

9.5 Nutzung von ECS als Server

9.5.1 Isolationsstufen

9.5.2 Linux-Grundlagen für die Realisierung von Containern

9.5.3 Docker