
Vorwort

*Ich unterrichte es nur; ich habe nicht gesagt, dass
ich etwas davon verstehe.*

Robin Williams in Good Will Hunting

Was macht eigentlich eine Programmiersprache aus? Die Frage ist schwerer zu beantworten, als es auf den ersten Blick scheinen mag. An der Oberfläche ist eine Sprache definiert durch ihre Syntax und Semantik. Das heißt, man muss wissen, welche Konstrukte sie enthält, mit welchen Schlüsselworten diese Konstrukte notiert werden und wie sie funktionieren.

Aber ist das schon die Sprache? Bei einfachen Sprachen mag das so sein. Aber bei größeren professionellen Sprachen ist das nur ein Bruchteil des Bildes. Ein typisches Beispiel ist JAVA. Der Kern von JAVA, also die Syntax und Semantik, ist relativ klein und überschaubar. Ihre wahre Mächtigkeit zeigt die Sprache erst in ihren Bibliotheken. Dort gibt es Hunderte von Klassen mit Tausenden von Methoden. Diese Bibliotheken erlauben es dem Programmierer, bei der Lösung seiner Aufgaben aus dem Vollen zu schöpfen und sie auf hohem Niveau zu konzipieren, weil er viel technischen Kleinkram schon vorgefertigt geliefert bekommt.

Doch hier steckt auch eine Gefahr. Denn die Kernsprache ist (hoffentlich) wohl definiert und vor allem standardisiert. Bei Bibliotheken dagegen droht immer Wildwuchs. Auch JAVA ist nicht frei von diesem Problem. Zwar hat man sich grundsätzlich große Mühe gegeben, die Bibliotheken einigermaßen systematisch und einheitlich zu gestalten. Aber im Laufe der Jahre sind zahlreiche Ergänzungen, Nachbesserungen und Änderungen entstanden, die es immer schwerer machen, sich in dem gewaltigen Wust zurechtzufinden.

Aber da ist noch mehr. Zu einer Sprache gehört auch noch eine Sammlung von Werkzeugen, die das Arbeiten mit der Sprache unterstützen. Auch hier glänzt JAVA mit einem durchaus beachtlichen Satz von Tools, angefangen vom Compiler und Interpreter bis hin zu Dokumentations- und Archivierungshilfen.

Und auch das ist noch nicht alles. Denn eine Sprache verlangt auch nach einer bestimmten Art des Umgangs mit ihr. Es gibt Techniken und Methoden des Programmierens, die zu der Sprache passen und die man sich zu Eigen machen muss, wenn man wirklich produktiv mit ihr arbeiten will. Und es gibt Arbeitsweisen, die so konträr zur Sprachphilosophie sind, dass nur Schauriges entstehen kann.

Irgendwie müssen sich alle diese Aspekte in einem Buch wiederfinden. Und gleichzeitig soll es im Umfang noch überschaubar bleiben. Bei JAVA kommt das der Quadratur des Kreises gleich.

So gibt es zum Beispiel zwei Bücher mit den schönen Titeln „Java in a Nutshell“ [15] und „Java Foundations Classes in a Nutshell“ [14]. Beides sind reine Nachschlagewerke, die nichts enthalten als Aufzählungen von JAVA-Features, ohne die geringsten didaktischen Ambitionen. Das erste behandelt nur die grundlegenden Packages von JAVA und hat 700 Seiten, das andere befasst sich mit den Packages zur grafischen Fenster-Gestaltung und hat 800 Seiten. Offensichtlich muss es viele Dinge geben, die in einem Einführungsbuch nicht stehen können.

Das vorliegende Buch hat das Programmierenlernen als Thema und JAVA als Vehikel. Und es geht um eine Einführung, nicht um eine erschöpfende Abhandlung über alles und jedes. Deshalb muss vieles unbehandelt bleiben. Alles andere wäre auch hoffnungslos.

Aus diesem Blickwinkel heraus war es ein echtes Problem, dass während des Schreibens des Buches das sog. Beta-Release der neuen Version **Java 1.5** erschien. Im Gegensatz zu den früheren Versionen sind hier wirkliche Neuerungen enthalten. Vor allem aber sind unter diesen Neuerungen auch einige, die echte Lücken im bisherigen Sprachdesign schließen. Deshalb habe ich mich entschlossen, die wichtigsten Erweiterungen von JAVA 1.5 in den Text aufzunehmen. Das kleine Risiko, dass sich vom Beta-Release zur endgültigen Version noch etwas ändern kann, scheint tolerierbar.

Jedes Einführungsbuch in JAVA hat mit einem Problem zu kämpfen: JAVA ist für erfahrene Programmierer konzipiert worden, nicht für Anfänger. Deshalb begannen die ersten JAVA-Bücher meist mit einem Kapitel der *Was ist anders als in C?* Inzwischen hat die Sprache aber einen Reife- und Verbreitungsgrad gefunden, der diese Form des Einstiegs überflüssig macht. Deshalb findet man heute vorwiegend drei Arten von Büchern:

- Die eine Gruppe bietet einen Einstieg in JAVA. Das heißt, es werden die elementaren Konzepte von JAVA vermittelt. Deshalb wenden sich diese Bücher vor allem an JAVA-Neulinge oder gar Programmier-Neulinge.
- Die zweite Gruppe taucht erst in neuerer Zeit auf. Diese Bücher konzentrieren sich auf fortgeschrittene Aspekte von JAVA und wenden sich daher an erfahrene JAVA-Programmierer. Typische Beispiele sind [50] und [33].
- Die dritte Gruppe sind Nachschlagewerke. Sie erheben keinen didaktischen Anspruch, sondern listen nur die JAVA-Features für bestimmte Anwendungsfelder auf. In diese Gruppe gehören z. B. die schon erwähnten Titel

[14] und [15], sowie das umfangreiche Handbuch [27], aber auch das erfreulich knappe Büchlein [45].

Das vorliegende Buch gehört in die erste Gruppe. Es beschränkt sich aber nicht darauf, nur eine Einführung in JAVA zu sein. Vielmehr geht es darum, Prinzipien des Programmierens vorzustellen und sie in JAVA zu repräsentieren. Auf der anderen Seite habe ich große Mühe darauf verwendet, nicht einfach die klassischen Programmieretechniken von PASCAL auf JAVA umzuschreiben (was man in der Literatur leider allzu oft findet). Stattdessen werden die Lösungen grundsätzlich im objektorientierten Paradigma entwickelt und auf die Eigenheiten von JAVA abgestimmt.

Weil JAVA für erfahrene Programmierer konzipiert wurde, fehlen in der Sprache leider einige Elemente, die den Einstieg für Anfänger wesentlich erleichtern würden. Das ist umso bedauerlicher, weil die Hinzunahme dieser Elemente leicht möglich gewesen wäre. Wir haben an der TU Berlin aber davon abgesehen, sie in Form von Präprozessoren hinzuzufügen, weil es wichtig ist, dass eine Sprache wie JAVA in ihrer Originalform vermittelt wird.

Damit wird das Lehren von JAVA für Anfänger aus didaktischer Sicht eine ziemliche Herausforderung. Dieser Herausforderung gerecht zu werden, war ein vorrangiges Anliegen beim Schreiben dieses Buches.

Das Buch ist aus einer zweisemestrigen Vorlesung an der Technischen Universität Berlin hervorgegangen, die vor allem Studierenden der Elektrotechnik und auch Wirtschaftsingenieuren eine Einführung in die Informatik geben soll. Die Erfahrungen, die in dieser Vorlesung über mehrere Jahre hinweg mit JAVA gewonnen wurden, haben die Struktur des Buches wesentlich geprägt.

Mein besonderer Dank gilt den Mitarbeitern, die während der letzten Jahre viel zur Gestaltung der Vorlesung und damit zu diesem Buch beigetragen haben, insbesondere Michael Cebulla, Martin Grabmüller, Thomas Nitsche und Baltasar Trancón y Widmann. Martin Grabmüller hat viel Mühe darauf verwendet, die Programme in diesem Buch zu prüfen und zu verbessern.

Die Mitarbeiter des Springer-Verlags haben durch ihre kompetente Unterstützung viel zu der jetzigen Gestalt des Buches beigetragen.

Berlin, im August 2004

Peter Pepper