



I, Robot

▶ Bei jeder Java-Installation erinnert ein Pop-up-Fenster unweigerlich daran, dass Java bereits auf mehreren Milliarden Geräten läuft. Diese große Zahl von Installationen lässt sich freilich nur zu einem kleinen Teil auf Notebooks und Desktop-PCs zurückführen; die Mehrheit der virtuellen Maschinen fristet ihr glückliches Dasein in eingebetteten und mobilen Geräten. Wer heutige Fachzeitschriften liest, könnte allerdings den Eindruck gewinnen, dass sich Plattformen wie Java ausschließlich auf die Entwicklung von Desktop- und Backend-Anwendungen fokussieren. Dummerweise ist von außen nicht sichtbar, dass die italienische Espresso-Maschine eine JVM unter der Haube trägt.

Dieses Ungleichgewicht hin zu eingebetteter Entwicklung dürfte sich unter anderem dank des Trends zur Mobilität noch weiter „verschärfen“ – moderne Gadgets begleiten uns schließlich überall hin und sehnen sich nach fortwährender Berührung. Aber auch für eingebettete Systeme im Allgemeinen muss Java zunehmend adäquate Lösungstechnologien bieten. Die Java-Plattform kann sich nur dann langfristig und erfolgreich am Markt behaupten, wenn sie diesen Entwicklungen Rechnung trägt.

Noch vor wenigen Jahren wäre eine Verbreitung von virtuellen Maschinen auf eingebetteter Hardware einer Blasphemie oder zumindest einem Tabubruch gleichgekommen. Die Zeiten ändern sich zum Glück, und das mit rasanter Geschwindigkeit. Leider zeigte sich die Entwicklung von Java als Plattform für mobile und eingebettete Anwendun-

gen nicht immer von ihrer Schokoladen- seite und war bisweilen mit einigen Tret- minen durchsetzt. Ich möchte jetzt nicht an alten Wunden rühren, aber können Sie sich noch an die Frühzeit des J2ME erinnern, an die SunSPOT-Technologie oder auch an die Anfänge der RTSJ (Real-Time Java Specification)? Da haben sich die Java-Väter wahrlich nicht immer mit Ruhm bekleckert.

Glücklicherweise ist das Java-Habitat seit jeher mit engagierten Zeitgenos- sen bevölkert, die das Heft selbst in die Hand genommen haben. Und so existie- ren mittlerweile selbst für Domänen wie Robotik oder Sensor-Netzwerke entspre- chende Java-Lösungen. (Anm. d. Red: Wer beim beliebten Buzzword-Bingo beeindrucken möchte, sollte stattdessen lieber mit dem Begriff „Cyber-Physical Systems“ um sich werfen).

Lassen Sie uns beispielsweise die Robotik betrachten. Dort gehört Java inzwischen zu den etablierten Imple- mentierungsplattformen. Nicht nur für Hobbyisten existieren mittlerweile zahl- reiche Java-basierte Entwicklungswerk- zeuge und Bibliotheken. Und was viele dabei übersehen: Bereits im Jahre 2004 setzte die NASA Java erfolgreich im Rahmen des Mars-Rover-Programmes ein. Java kann also in dieser Hinsicht auf viele Erfolgsstories verweisen.

An dieser Stelle steht zu „befürchten“, dass schon bald Isaac Asimovs Roboter- gesetze ihre Anwendung finden müssen, die dann, wenn auch leicht verfälscht, folgende Ausprägung haben könnten:

0. Eine virtuelle Maschine darf ihre Ent- wickler oder Benutzer nicht verletzen oder durch Passivität zulassen, dass Entwickler oder Benutzer zu Schaden kommen.
1. Eine virtuelle Maschine darf keinen Entwickler oder Benutzer verletzen oder durch Untätigkeit zu Schaden kommen lassen, außer sie verstieße damit gegen das nullte Gesetz.
2. Eine virtuelle Maschine muss den Be- fehlen der Entwickler gehorchen – es sei denn, solche Befehle stehen im Wider- spruch zum nullten oder ersten Gesetz.

3. Eine virtuelle Maschine muss ihre ei- gene Existenz schützen, solange die- ses Handeln nicht dem nullten, ersten oder zweiten Gesetz widerspricht.

Wer denkt, das sei zu weit hergeholt, dem möchte ich entgegenbringen, dass ich erst unlängst eine unliebsame Kon- frontation mit einem Java-basierten Staubsauger hatte.

Im Falle des manisch-depressiven Roboters Marvin aus dem Buch „Hitch- hiker’s Guide to the Galaxy“ fragt man sich freilich, ob Marvin wirklich Asimovs Gesetzen genügen würde:

Marvin: Ich verspürte große Lange- weile und Depression, weshalb ich mich über eine externe Schnittstel- le mit dem Computer verband. Ich unterhielt mich sehr lange mit ihm und erläuterte meine Sicht des Uni- versums.

Ford: Und was ist dann passiert?

Marvin: Er beging Selbstmord!

Es ist allerdings zu vermuten, dass Mar- vins Realisierung noch keine Java-VM zugrunde lag.

Der zunehmenden Bedeutung von Ja- va für eingebettete Systeme tragen wir in dem vorliegenden Heft Rechnung. Jürgen Heck und seine Koautoren zei- gen, warum Java in industriellen Sys- temen immer mehr Verbreitung findet. Sie erfahren von Volker Linz und Jürgen Menge, wie sich Oracle ADF für die Im- plementierung mobiler Unternehmens- anwendungen einsetzen lässt. James Hunt adressiert die wichtige Thematik sicherheitskritischer Systeme.

In diesem Sinne wünsche ich Ihnen viel Spaß mit der vorliegenden Ausgabe

Ihr Prof. Dr. Michael Stal