



And now for something completely different

▶ In „Jim Knopf und Lukas der Lokomotivführer“ ist von dem Scheinriesen Tur Tur die Rede, der aus weiter Distanz riesig wirkt, aber schrumpft, sobald sich der Beobachter ihm nähert. Bei eingebetteten oder mobilen Systemen verhält es sich ähnlich. Aus der Nähe betrachtet, erscheinen die Geräte zunächst ziemlich „mickrig“. Erst, wenn sie der kritische Betrachter aus objektiver Distanz analysiert, wird ihr Potenzial umso deutlicher. Es herrscht also eine Art „Heisenbergsche Unschärferelation“, derzufolge sich der Marktimpuls einer Technologie und ihre technische Bedeutung nicht gleichzeitig messen lassen.

Auch die Java-Technologie begann als jugendlicher, unscheinbarer David, um nur wenig später die trägen Goliaths der IT-Industrie vor sich herzutreiben. Ursprünglich als Programmierplattform für eingebettete Systeme konzipiert, erfolgte der Siegeszug allerdings über Umwege. Zunächst mittels Webanwendungen, über Serveranwendungen bis hin zu Desktop-Applikationen. Mit Ausnahme von vereinzelt, zaghaften und eher stiefmütterlichen Versuchen war die Java-Technologie daher schon Lichtjahre von ihren Wurzeln entfernt, als die Entwickler von Android sich auf die inhärenten Stärken des Java-Ökosystems besannen. Mit Hilfe der JVM können Entwickler Android-Apps entwickeln, die auf der Vielzahl Android-basierter Smartphone- und Tablet-Hardware funktionieren – Write-Once-Run-Anywhere fast in Reinkultur.

Inzwischen gibt es eine neue „Grass Roots“-Bewegung, über die Experten wie einst auch über Java nur müde lächeln. Schließlich handelt es sich doch

um Spielzeug für technologieverliebte Enthusiasten und moderne IT-Zeit-Menschen, die zumindest im Falle männlicher Exemplare mit langen Bärten und Sandalen im dunklen, unaufgeräumten Hobbykeller ihrem Erfindergeist frönen. Doch weit gefehlt. Die „Do-it-yourself“-Welle der Achtziger Jahre hat in der Maker-Bewegung ihren würdigen Nachfolger gefunden. „Makers“ sind engagierte Amateure und Profis, die selten allein und meistens gemeinsam mit Gleichgesinnten Hardware konstruieren, und dann die Bauanleitung frei ins Netz stellen. Wer einmal Beispiele im Web durchstöbern möchte, kann das unter anderem im Thingiverse tun (<http://www.thingiverse.com/>). Der (Nach-)Bau erfolgt dann entweder in der eigenen Werkstatt oder in einer lokalen Maker-Werkstatt, die eine größere Anzahl von Makern gemeinsam finanziert haben. Als Beispiel sei an dieser Stelle der Verein Attraktor in Hamburg genannt (<http://blog.attraktor.org/>). Wer keinen Zugriff auf 3D-Drucker hat, kann auf die zahlreichen Anbieter im Internet zurückgreifen, die aus kundenspezifischen 3D-Grafiken – im standardisierten Format STL (Surface Tesselation Language) – die entsprechenden Bauteile auch in sehr kleinen Stückzahlen oder sogar einzeln produzieren.

Zwei der Hauptingredienzen des Maker-Schaffens sind dabei 3D-Drucker zum Herstellen von Gehäusen und mechanischen Teilen sowie standardisierte Mikrocontroller-Platinen, auf Neudeutsch „Physical Computing“-Plattformen, als Grundlage von eigener Elektronik oder Mechatronik. Einige Plattformen firmieren sogar als Open-Source-Hardware, sodass beliebige Dritthersteller entsprechende Platinen ohne Lizenzkosten entwickeln und anbieten dürfen. Dadurch können wahre Macher inzwischen auf ein breites Spektrum diverser Platinen und Hardwareerweiterungen zurückgreifen, und das zu passablen Preisen. Die Platinen rangieren dabei von Standardformaten bis zu Kleinstgrößen, die sich sogar in Kleidung integrieren lassen.

Die Hardware umfasst ein einfaches I/O-Board mit Mikrocontroller sowie digitalen und analogen Eingängen und Ausgängen. Zu den verbreitetsten Plat-

nen gehören das Arduino-Board und die Raspberry Pi-Hardware, gefolgt von Android-Boards, dem Beagleboard und dem .NET-basierten Netduino-System. Nicht zu ignorieren sind dabei die Mindstorms-Baukästen von Lego, die Erwachsene angeblich ihren Kindern, aber in Wirklichkeit meistens sich selbst schenken.

Und an dieser Stelle kommt der Hauptakteur unserer Zeitschrift, also Java ins Spiel. So beruht die Programmierumgebung für Arduino auf der in Java programmierten Processing-Umgebung – was übrigens eine weitere interessante Technologie darstellt. Aber auch Java-APIs zum Zugriff auf die Funktionalitäten eines Arduino stehen dem Entwickler mittlerweile zur Verfügung. Ein Höhepunkt wären freilich JVMs für die Arduino-Hardware – so dachten viele. Und schon sprangen letztes Jahr die Jungspunde HaikuVM und µJ aus dem Zauberhut.

Für den Linux-basierten Raspberry Pi sowie für das Beagleboard existieren sowohl JDK-Entwicklungsplattformen als auch Bibliotheken zum Zugriff von Java-Programmen auf die nativen System-Bibliotheken. Lego-Mindstorms-Entwickler erschließen sich durch LeJOS als Java-basiertes Betriebssystem neue Wege, um eigene Roboter-Kreationen zu steuern. Das ist gewissermaßen die „Bright Side of Life“. Wer nun glaubt, nur Hobbyisten wären der Open-Source-Hardware verfallen, täuscht sich gewaltig. In einigen kommerziellen Produkten stecken heute schon Arduino-Boards.

Mittels der Open-Source-Hardware und Maker-Bewegung könnte sich also Java in Zukunft „back to the roots“ entwickeln. Längst vorbei die Zeiten, als Entwickler eingebetteter Systeme eine JVM als Plattform für utopisch hielten. Statten Sie also ruhig einmal Tur Tur, dem Scheinriesen, einen Besuch ab. Ach ja, und heizen Sie schon einmal den LötKolben auf.

Sich informieren können Sie allerdings schon heute. Sie finden in diesem Heft dazu einige interessante Artikel. Viel Spaß bei der Lektüre wünscht Ihnen

Ihr Prof. Dr. Michael Stal