

OOP 2010: Prozesse statt Programmierung

# Die Sprachen-Renaissance

Kersten Auel, Alexander Neumann

Objektorientierung ist längst nicht mehr das zentrale Thema der OOP. Das bekräftigte nicht zuletzt eine der Keynotes, deren Thema die funktionale Programmierung war.

War die OOP, die vom 25. bis 29. Januar in ihrer 19. Auflage in München stattfand, früher eine Pflichtveranstaltung für den klassischen Programmierer, ist sie über die Jahre hinweg mehr und mehr zur Konferenz für Projektleiter und Softwarearchitekten geworden. Die letzten Jahre schien der Veranstalter mit der Entscheidung, von technischen Themen eher abzurücken und auf Projektmanagementvorträge zu setzen, richtig gelegen zu haben. Der Teilnehmer fand eine nahezu unveränderte Anzahl Besucher vor, und die Ausstellung führte zahlreiche große und kleine Unternehmen und war vor allem gut besucht. Dieses Jahr waren spürbar weniger Teilnehmer vor Ort, und die Messe zählte mit rund 70 Ständen etwas weniger als das letzte Jahr, wenn auch die Organisatoren sie wieder als „ausverkauft“ meldeten. Bei aller Spekulation mögen die Gründe wohl in der finanziell

angespannten Lage vieler IT-Abteilungen zu finden sein und nicht allein in der thematischen Ausrichtung der Konferenz, denn Letztere war durchaus ansprechend.

## Vom Buzzword in den Alltag

Neben zu erwartenden Hype-thesen wie Cloud Computing, Multi-Core-Programmierung oder agile Softwareentwicklungsmethoden dürften für viele die große Menge an praxisnahen Erfahrungsberichten wertvoll gewesen sein, die dem Motto der Konferenz „Productivity: People, Process, and Technology“ in ihrer ganzheitlichen Sicht am nächsten gekommen sind.

Clouds scheinen nicht mehr nur ein neues Buzzword zu sein. Einen Einblick in das „Distributed Computing“ bei Google gab beispielsweise Gregor Hohpe, der erläuterte,

warum man sich dort für bestimmte Techniken entschieden hat. Unter der Devise „Less is More“ etwa hat man sich mit dem Einsatz des hauseigenen Datenbanksystems BigTable gegen Transaktionen entschieden und setzt auf einfache Datentypen. Da laut Hohpe die Frage nicht sei, ob ein System irgendwann ausfällt, sondern wann, gibt es drei Replikate der im Google File System (GFS) gespeicherten Daten.

Visionären und zugleich retrospektiven Charakter hatten zwei der insgesamt sieben Keynotes. Bei beiden ging es um die Renaissance von Programmiersprachen. Dürfen die letzten Jahre eher als Epoche der Werkzeuge oder der Frameworks gelten, liegt das Augenmerk der jüngeren Vergangenheit wieder verstärkt auf den Sprachen selbst. „Newcomer“ wie Scala, Clojure und F# bringen neue Begeisterung in die Zukunft der Entwickler.

Auf sie gingen auch Klaus Alfert und Bernd Löchner von Zühlke in ihrer Keynote zur funktionalen Programmierung ein. Clojure ist ein Lisp-Dialekt, der Code für die Java Virtual Machine kompiliert und sich vor allem für die nebenläufige Programmierung eignet, und Microsofts F# geht auf den ML-Ableger OCaml zurück. Scala bezeichneten die beiden Zühlke-Angestellten als Hybrid aus funktionaler und objektorientierter Programmiersprache, das zudem die Java- und die .Net-Plattform anspricht.

## Pragmatik statt Dogmatik

Das Interesse an der funktionalen Programmierung liegt in der Multi-Core-Programmierung begründet. Die Übergabe von Werten statt Objekten soll Inkonsistenzen verhindern, die sich dadurch ergeben, dass zwei Objekte gleichzeitig in einen Thread schreiben. Als Treiber sind Themen wie Serverkonsolidierung, Green-IT, steigender Admin-Aufwand und das Bedürfnis zu nennen, mehr aus sequenziellen Prozessen herausholen zu müssen. Die Konzepte funktionalen Programmierens sind nicht neu, doch im Gegensatz zu vor 20 Jahren liegen heute reifere Implementierungen und gute



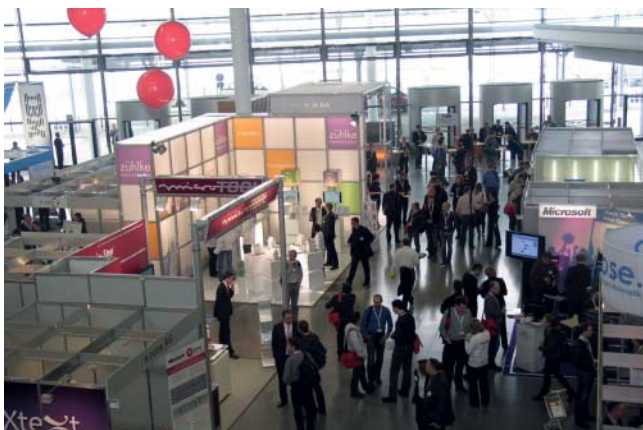
„Uncle Bob“ wirbt für testgetriebene Softwareentwicklung.

Plattformintegrationen vor. Von einem Paradigmenwechsel rede allerdings niemand mehr. Anders als früher sei nicht mehr das Allheilmittel gefragt. Vielmehr stehen Entwickler heute vor der Aufgabe, aus einer Vielzahl etablierter und neuer Sprachen die für ein Projekt jeweils richtige zu wählen.

Mit Programmiersprachen setzte sich auch Robert C. Martin auseinander. „Uncle Bob“ hat sich einen Namen gemacht als Gründer von Object Mentor. In seiner Keynote „The Polylgot Craftsman“ gab er mit großem schauspielerischen Talent einen historischen Überblick zu den Hauptströmungen bei den Programmiersprachen. Er sieht ihre Evolution in Skriptsprachen wie Ruby, Python und Groovy münden, deren Vorteil es sei, dass man mit ihnen testgetrieben entwickle, sodass man nicht mehr schlechten Code schreiben könne. Zudem sieht er bei ihnen das Potenzial, fünfmal schneller entwickeln zu können.

Die OOP war eine in der Auswahl der Themen sowie in der Qualität und professionellen Darbietung der Vorträge gelungene Veranstaltung. Die Konferenz mag zwar weniger Besucher gezogen zu haben, leer waren die Flure aber noch lange nicht, und auch das für Konferenzen dieser Art wichtige Networking ist für die meisten Teilnehmer sicherlich nicht zu kurz gekommen. Es bleibt zu wünschen, dass sich die OOP zum Jubiläum im nächsten Jahr treu bleibt und wieder ein größeres Publikum findet. Die OOP 2011 findet vom 24. bis 28. Januar statt.

(ane/ka)



Ausstellung als Networking-Anlaufstelle