Componentware:

Methodik des evolutionären Architekturentwurfs

Andreas Rausch

Inhaltsverzeichnis

1 Einleitung

Moderne Technologien, neue Beschreibungstechniken, verschiedene Vorgehensmodelle – in den letzten zehn Jahren haben diese Themen den Wandel in der Informatik entscheidend mitbestimmt. Anfang der 90'er Jahre setzte der Siegeszug objektorientierter Softwareentwicklung einen neuen technologischen Umbruch mit einer enormen Eigendynamik in Gang. Ausgehend von dem breiten Erfolg objektorientierter Programmiersprachen entwickelten sich Client/Server-Programmierung, Drei-Schichten-Architekturen und verschiedene Middleware-Ansätze. Inzwischen werden die ersten Application Server, objektorientierten Transaktionsmonitore und Embedded Workflow Engines im industriellen Umfeld eingesetzt. Ein Ende dieser technologischen Innovationswelle ist bis heute nicht in Sicht. Stetig kommen neue Technologien auf den Markt und werden mehr oder weniger erfolgreich in der Industrie eingesetzt.

Nahezu zeitgleich mit dem Aufstieg der objektorientierten Programmierung erschienen die ersten Arbeiten im Bereich der Methodik objektorientierter Analyse und Design. Die bekanntesten stammen von den Hauptautoren der Unified Modeling Language (UML) [BJR98, RJB98]: Grady Booch [Booc94], Jim Rumbaugh [RBP+91] und Ivar Jacobson [Jaco92]. Darüber hinaus haben aber auch eine Vielzahl anderer Autoren maßgeblich die Entwicklungen der ersten objektorientierten Methoden bestimmt, wie zum Beispiel Bertrand Meyer [Meye88], Sally Shlaer und Steve Mellor [SM89], Peter Coad und Ed Yourdon [CY90, CY91], Rebecca Wirfs-Brock [WWW90], Derek Coleman [Cole93] sowie James Martin und Jim Odell [MO92], um nur einige von ihnen zu nennen.

Aus diesen Aktivitäten entstand zuerst eine Fülle unterschiedlicher methodischer Ansätze sowie Beschreibungs- und Modellierungstechniken. Dieser „Wildwuchs" konvergierte in der Entwicklung eines internationalen Standards für objektorientierte Modellierung, der UML. 1997 wurde die Version 1.1 der UML von der Object Management Group (OMG) standardisiert [OMG97]. Die neueste Version des UML Standards ist die Version 1.3 [OMG00a]. Die Vorbereitungen zur Standardisierung der Version 2.0 sind bereits angelaufen [OMG01a].

Zunächst blieb die Frage offen, wie dieser Standard anzuwenden sei. Denn im Gegensatz zu ihren Vorgängern ist die UML keine Methode. Sie definiert nur Notation und Semantik der Modellierungselemente. Mit etwas Verzögerung begannen die Initiatoren der UML unter der Federführung von Ivar Jacobson mit den Arbeiten an einem Vorgehensmodell. Das Ergebnis ist der Unified Software Development Process [JBR99, Kruc00]. Parallel dazu entstanden neue Vorgehensmodelle und vorhandene wurden weiter entwickelt, wie zum Beispiel das V-Modell '97 [DW99], der Catalysis-Ansatz [DW98], der Object Engineering Process [OOSE01] und die Ansätze der Process Patterns Community [Ambl98, Ambl99, BRSV98a, BRSV98b, BRSV98c, GMP+01a, GMP+01b]. Zwar ist die Standardisierung von Vorgehensmodellen problematischer als die von Notation und Semantik und auch nur bedingt sinnvoll, da die meisten Unternehmen ihre spezifischen Vorgehensmodelle definieren wollen, dennoch zeichnet sich auch hier eine gewisse Konvergenz ab.

Mit Hilfe von zwei neuen Konzepten versuchen die Autoren den bekannten Problemen klassischer Verfahren zu begegnen: Eine iterative und inkrementelle Vorgehensweise soll durch frühzeitige Rückkopplung Risiken minimieren. Die anwendungsfallgetriebene und architekturzentrierte Modellierung soll zu einer adäquaten und stabilen Architektur führen.