Arbeitskreis „Generative und komponentenbasierte Softwareentwicklung“ der GI-Fachgruppe 2.1.9 – Tagungsbericht

Ulrich Breymann, Hochschule Bremen
brey@informatik.hs-bremen.de

Am 10. September fand in Erfurt die erste Tagung des Arbeitskreises „Generative und komponentenbasierte Softwareentwicklung“ im Rahmen der STJA '97 (Smalltalk und Java in Industrie und Ausbildung) statt. Der Arbeitskreis wurde innerhalb der Fachgruppe 2.1.9 (Objektorientierte Softwareentwicklung) der „Gesellschaft für Informatik“ auf der OOP '97 in München von Prof. Dr. Manfred Broy (TU München), Dipl.-Inform. Krzysztof Czarnecki (Daimler-Benz AG) und Prof. Dr. Ulrich Eisenecker (FH Heidelberg) gegründet.

Die generative Softwareentwicklung legt die Grundlagen für die automatisierte Komposition von Software-Systemen aus abstrakten und konkreten Bausteinen. Ziele sind die Verbesserung der Wiederverwendbarkeit und Anpaßbarkeit, die Beherrschung hoher Komplexität und die Handhabung einer großen Variantenvielfalt. Es gibt einige, gut bekannte Ansätze zur Generierung von Software, das Gebiet ist aber durch die Verbindung zur Objekttechnologie stark in Bewegung. Die generative Softwareentwicklung hat Bezüge zu Domain Engineering, Intentional Programming, Aspect Oriented Programming und weiteren Ansätzen, die hier nicht alle aufgezählt werden können (siehe Internetadresse unten).

Dem Thema wurden zwei der fünf Keynotes und einer der vier Tracks der STJA gewidmet. Die erste Keynote von Karl Lieberherr (Northeastern University, USA) „Demeter and Aspect Oriented Programming“ zeigte, wie Aspekte automatisch in eine Applikationssoftware hineingewoben („aspect weaver“) werden können. Unter „Aspekt“ werden hier realisierungsbezogenen Dinge verstanden, zum Beispiel landesspezifische Vorschriften oder die Währung eines Landes, die separat von der primären Funktionalität wie Geschäftsobjekte und -regeln zu sehen sind. Das Demeter-System erreicht das Hineinweben von Aspekten durch eine Traversierung des Klassenbaums und der entsprechenden Erzeugung von Code (C++ oder Java).

Der Hauptvortrag von Mark Simos (Synquiry Technologies Ltd., Boston, USA) „Organization Domain Modeling and OO Analysis and Design“ stellte die Methode Organization Domain Modeling (ODM) vor, welche die Modellierung der Applikationen einer Domäne besonders im Hinblick auf das Erkennen von Gemeinsamkeiten und Unterschieden erleichtern soll. Dadurch wird eine anwendungsübergreifende Wiederverwendung von Modellen, Dokumenten und Code bereits in frühen Projektphasen möglich. Die Methode betont das soziale und kommunikative Moment von Modellierungsprozessen.

Darüber hinaus gab es viele weitere interessante Vorträge, von denen hier nur zwei stellvertretend genannt seien. Das vollständige Programm der STJA ist unter *http://www-stja.tu-ilmenau.de* zu finden.

Der Vortrag von Martin Schütze befaßte sich mit der musterbasierten Komponentengenerierung am Beispiel der Gebäudesimulation, ein Teilprojekt des Sonderforschungsbereichs 501 an der Universität Kaiserslautern. Mehmet Aksit (Universität Twente, Niederlande) stellte die Wiederverwendung und Komposition von Komponenten durch „Composition Filters“ vor. In einer Demonstration zeigte er anhand eines Email-Systems die automatische Realisierung verschiedener Aspekte wie zum Beispiel verschiedene Leseberechtigungen für Empfänger und Mail-System, Synchronisation und automatische Überwachung auf doppelt gesendete Mails.

Dieser Tagungsbericht kann nur einen oberflächlichen Eindruck vermitteln. Nähere Hinweise und Informationen sind unter

http://nero.prakinf.tu-ilmenau.de/~czarn/generate

zu finden. Insbesondere ist dort eine Liste mit ca. 110 Quellenhinweisen und WWW-Links abgelegt, die einen guten Einstieg bieten.

Den Organisatoren der STJA und des AK ist ein sehr interessantes Programm gelungen. Dank der Sponsoren gab es freie Verpflegung und niedrige Teilnahmegebühren, die Studierenden und Hochschulangehörigen sogar noch erlassen wurden. Einen herzlichen Dank an die Veranstalter und die Sponsoren!