



□ Michael Deynet

(E-Mail: michael.deynet@tu-clausthal.de)

ist wissenschaftlicher Mitarbeiter am Lehrstuhl für Software Systems Engineering an der Technischen Universität Clausthal. Er ist dort an der Weiterentwicklung des V-Modell XT, der verbindlich vorgeschriebene Entwicklungsprozess für IT-Projekte der Bundesrepublik Deutschland, beteiligt. Herr Deynet hat in diesem Rahmen sowohl industrielle Projekte als auch Projekte der öffentlichen Hand betreut. Er beschäftigt sich wissenschaftlich u. a. mit modellbasierter Software-Entwicklung, modellbasierten Spezifikationstechniken und der Modellierung von Software-Entwicklungsprozessen.



□ Sebastian Herold

(E-Mail: sebastian.herold@tu-clausthal.de)

ist wissenschaftlicher Mitarbeiter und Doktorand am Lehrstuhl für Software Systems Engineering an der Technischen Universität Clausthal. Seit 2007 beschäftigt er sich dort in Forschung und Projektarbeit mit den Themen Softwarearchitektur und modellbasierte Softwareentwicklung. In seiner Promotion befasst er sich mit der Überprüfung von Sollarchitekturen während Design und Implementierung.

objektspektrum themenspecial: architekturen

Das weite Feld Softwarearchitektur

Der Bereich Softwarearchitektur ist eines der facettenreichsten Teilgebiete des Software Engineerings. Wer Softwarearchitekten aus der Praxis nach ihren Aufgaben befragt oder gewünschte Fähigkeiten in Stellenausschreibungen liest, muss glauben, dass der ideale Softwarearchitekt ein wolpertingerartiges Mischwesen aus Projektleiter, Consultant, Technologieexperte und Chefprogrammierer ist. Dies liegt zum einen an den unterschiedlichsten Vorstellungen dessen, was eigentlich eine Softwarearchitektur ist, was durch sie beschrieben werden sollte, und somit auch, was eigentlich die Aufgaben eines Softwarearchitekten (und zahlreiche andere, synonym benannter IT-Spezialisten) sind. Zum anderen liegt der Facettenreichtum einfach am Wesen der Softwarearchitektur. Softwarearchitekturen sind in allen Entwicklungsstufen eines Projektes relevant und sollten durchgängig betrachtet werden – von der Anforderungsanalyse über die Implementierung bis zum Betrieb, der Wartung und einer eventuellen Stilllegung. Auch hieraus ergibt sich die Vielfalt des Aufgabenfeldes eines Softwarearchitekten.

So haben Peter Hruschka und Gernot Starke in der Ausgabe 04/09 des OBJEKTSpektrums den Softwarearchitekten sehr passend als den Zehnkämpfer der IT bezeichnet.

Dieses Bild beibehaltend, hat der IT-Zehnkämpfer aber im Vergleich mit zusätzlichen Widrigkeiten zu kämpfen – schließlich ändern sich für ihn häufig die zu bewältigenden „Sportarten“ oder zumindest die Regeln. Dies lässt sich am Forschungsfeld der Softwarearchitektur erkennen, aber auch am „State of the Practice“. Wird von Forschung und Praxis eine neue Art von Systemen betrachtet, wie seit einiger Zeit die „Ultra-Large-Scale Systems“ oder die IT-Ökosysteme, stellen sich immer wieder die Fragen nach geeig-

neten Softwarearchitekturen, ihrer systematischen Erstellung und Beschreibung. Ähnlich verhält es sich beim Aufkommen von neuen Technologien und Frameworks zur Implementierung von Systemen und bei neuartigen Programmierparadigmen. Stets sind Softwarearchitekten gefragt, um deren Eignung für bestimmte Systeme oder Auswirkungen auf die Struktur von Systemen aus architektonischer Sicht zu beurteilen.

So gesehen wird Softwarearchitektur auch weiterhin ein hochdynamisches und -interessantes Feld bleiben. Auch dieses Online-Themenspecial des OBJEKTSpektrums zum Thema Architekturen spiegelt die Vielfalt des Themenbereichs Softwarearchitektur wider.

Ernst Sikora und Professor Klaus Pohl befassen sich in ihrem Artikel mit der Frage, wie Requirements Engineering und Architekturentwurf besser verzahnt werden können. Schließlich ist ein schlechtes Zusammenspiel zwischen beiden Bereichen eine mögliche Ursache für teure Fehler bei der Systemerstellung. In ihrem Artikel stellen die Autoren einen integrierten Entwicklungsansatz vor.

Viele performancebezogene Softwareprobleme haben ihre Wurzeln in Architekturentscheidungen, die bereits während der Phase der Anforderungsanalyse getroffen werden. Je weiter das Projekt fortschreitet, desto aufwendiger wird es, diese Entscheidungen rückgängig zu machen. Viele dieser Probleme lassen sich aber

bereits frühzeitig in der Entwicklung feststellen und vermeiden. **Andreas Grabner** gibt einen Einblick, wie Architektur- bzw. Designentscheidungen in Bezug auf Performance und Skalierbarkeit frühzeitig validiert werden können.

Colin Hood und **Knut Salomon** stellen in ihrem Artikel eine Standard-Architektur eines Automobil-Zulieferers vor, die es ermöglicht, mit agilen Vorgehensweisen in kürzerer Zeit komplexe und dennoch anpassbare Systeme zu entwickeln.

Mit komponentenbasierter Entwicklung beschäftigen sich **Ralf Kruthoff-Brüwer**, **Professor Frank M. Thiesing** und **Frank Nordemann**. Sie beschreiben wie das Managed Extensibility Framework genutzt

werden kann, um verschiedenartige Komponentenarchitekturen zu verbinden. **Marcus Powarzynski** diskutiert einen Architektur-aspekt für eine spezifische Klasse von Systemen. In seinem Artikel geht es um die effiziente Anbindung von eingebetteten Datenbanken – insbesondere ein Thema in ressourcenbeschränkten Systemen und Umgebungen.

Welche Möglichkeiten gibt es, Architekturrichtlinien und Programmierichtlinien während der Implementierung zu überprüfen? Diese Frage zur Gewährleistung der Einhaltung einer gewünschten Sollarchitektur behandelt der Artikel von **Boris Wehrle**.

Softwaresysteme und damit auch deren Architekturen sind, auch vor ihrer

Einführung, einem steten Wandel unterzogen. **Albert Wilhelm** und **Hubert Zenner** beschreiben in ihrem Artikel, wie man durch geeignete Methoden und Werkzeuge auf diese Änderungen reagieren kann.

Oftmals ist es nötig, Softwarearchitekturen während des Betriebes zu überwachen und zu analysieren. **Eric Schaumlöffel** stellt in seinem Beitrag ein Werkzeug vor, mit dem Softwarearchitekturen applikationsübergreifend geprüft und gesteuert werden können. Dies betrifft sowohl die technische als auch die fachliche Sicht auf die Softwarearchitekturen.

Wir wünschen Ihnen eine interessante und vielfältige Lektüre! ■