

# “Software-Architektur und Migration”

## Workshopbericht

Rainer Gimmich (IBM Frankfurt)  
Michael Goedicke (Universität Duisburg-Essen)  
Wilhelm Hasselbring (Universität Oldenburg)  
Maritta Heisel (Universität Duisburg-Essen)  
Ralf Reussner (Universität Karlsruhe)  
Andreas Winter (Universität Mainz)

im Rahmen der Software-Engineering 2007 in Hamburg

27. März 2007

Software-Architekturen sind wichtige Artefakte zur Unterstützung der Software-Entwicklung und Software-Evolution. Ziel der Entwicklung von Software-Architekturen ist es, Beschreibungen für grundlegende Strukturen von Softwaresystemen oder Familien von Softwaresystemen zu erhalten, die einerseits die Anforderungen an das System oder an Produktfamilie reflektieren und andererseits, neben weiteren Möglichkeiten zur Nutzung (z.B. Planung), auch eine Basis für eine strukturierte Implementierung und Weiterentwicklung bieten. Die zahlreichen Ansätze zur Modell-basierten Entwicklung dokumentieren dies. Darüber hinaus beschreiben Software-Architekturen auch bestehende Softwaresysteme und dienen als Ausgangspunkt zur Transformation und Migration. Das Finden adäquater Software-Architekturen vor allem für existierende Systeme und deren Übertragung auf andere Plattformen hat sich als eine bedeutende Herausforderung herausgestellt.

Ziel des Workshop *Software-Architektur und Migration*, der am 27. März 2007 im Rahmen der Software-Engineering 2007 (SE07) in Hamburg durchgeführt wurde, war es, neuere Ansätze zur flexiblen Gestaltung, Beschreibung und Anwendung von Software-Architekturen zu diskutieren. Besondere Berücksichtigung sollen aktuelle Herausforderungen wie die Analyse, Synthese, Komposition und Evolution von Architekturen erhalten.

Insgesamt wurden sechs Arbeiten präsentiert und intensiv diskutiert:

**Alexander Stuckenholtz** (FernUniversität Hagen) *Synthese komponentenbasierter Konfigurationen*

Die Analyse der Abhängigkeiten zwischen Versionen von Komponenten nutzt Alexander Stuckenholtz zur Ermittlung von möglichen sinnvollen Konfigurationen. In seinem Beitrag stellt er unterschiedliche Ansätze zur Integration von Komponenten vor und vergleicht diese anhand einer Fallstudie. Die vorgestellten Verfahren unterstützen sowohl die Bewer-

tung vorhandener Konfigurationen als auch die automatische Generierung von Konfigurationsvorschlägen.

**Klaus Krogmann, Steffen Becker** (Universität Karlsruhe) *A Case Study on Model-Driven and Conventional Software Development: The Palladio Editor*

Klaus Krogmann vergleicht anhand der Entwicklung eines graphischen Editors konventionelle mit Modell-getriebener Software-Entwicklung. Im konkreten Beispiel konnten deutliche Vorteile bezüglich Entwicklungsdauer und Erweiterbarkeit für den Modell-getriebenen Ansatz festgestellt werden. Diese Tendenz ist jedoch in breiter angelegten und vergleichbareren Studien noch zu validieren.

**Tanja Schmedes** (OFFIS Oldenburg) *Modellgetriebene Dezentralisierung von Prozessen, Services und Daten am Beispiel des Energiemanagements*

Tanja Schmedes behandelt die Migration von Energiemanagement-Systemen in eine Service-orientierte Architektur. Die Funktionalität des Alt-Systems wird hierzu in Services gekapselt und unter Berücksichtigung der nötigen Topologie und Autonomie in flexiblen und anpassungsfähigen Strukturen bereitgestellt.

**Jürgen Rückert, Barbara Paech** (Universität Heidelberg) *Referenz-Architektur und nichtfunktionale Anforderungen adaptiver Dialogkerne*

Die Einbindung von Benutzungsschnittstellen in Service-orientierte Systeme wird im Beitrag von Jürgen Rückert behandelt. Benutzungsschnittstellen bündeln die bereitgestellte Funktionalität gegenüber den Anwendern und müssen an die jeweils genutzten Dienste angepasst werden können. Hierzu wird

eine Referenz-Architektur eines adaptiven Dialogkerns vorgeschlagen, die auch prototypisch mit bereits vorhandenen Komponenten realisiert ist.

**Bernhard Humm, Marc Lohmann, Markus Voß,**

**Johannes Willkomm** (sd&m Research) *Ein praxiserprobtes Rahmenwerk für die technische Anwendungsintegration*

Markus Voß präsentierte ein umfassendes Rahmenwerk zur Anwendungsintegration, das die Erfahrungen der sd&m-AG zur Entwicklung und Evolution von Anwendungslandschaften dokumentiert. Die Integration von Anwendungslandschaften erfolgt auf Ebene der Präsentation, der angebotenen Services, und der benutzten Daten. Referenzarchitekturen auf allen drei Integrationsebenen und ihr wiederholter Abgleich mit Produktlandschaften zur Beschreibung konkreter Lösungsansätze unterstützt die Evolution komplexer Anwendungslandschaften.

**Dominikus Herzberg** (Fachhochschule Heilbronn) *Was ist Software-Architektur? Ein Abgleich mit der Praxis*

Dominikus Herzberg stellte Begriffsdefinitionen für Software-Architektur aus der Wissenschaft der Begriffsverwendung in der industriellen Praxis kontrovers gegenüber. Insbesondere arbeitete er heraus, dass im industriellen Umfeld dokumentierte Software-Architekturen kaum konsultiert werden und Änderungen/Erweiterungen direkt im Code durchgeführt werden. Hieraus leitete er ab, dass Software-Architekturen nicht von Anforderungen an Softwaresystem determiniert werden, sondern eher durch die Implementierungen bestimmt sind.

Langfassungen der Beiträge sind im Workshop-Band zur SE07 (Bleek, W.-G., Schwentner H., Züllighoven, H. (Hrsg.): Software Engineering 2007, Beiträge zu den Workshops, LNI 106, Bonn, 2007, S. 151ff.) publiziert.

Zentrales Thema der Diskussion zu den Beiträgen des Workshops "Software-Architektur und Migration" war die Bedeutung von Architekturen in der Evolution von Softwaresystemen. Ansätze und Erfahrungen im Umgang mit der Evolution von Softwaresystemen auf Architekturbasis fanden sich in allen Beiträgen. Architekturen von Softwaresystemen stellten hier sowohl Ausgangspunkte der Untersuchungen als auch die Ziele von Migrationsvorhaben dar.

Der Vergleich mit der Praxis stellte jedoch deutlich heraus, dass gegenwärtige Architekturen nur geringe Auswirkungen auf die konkrete Weiterentwicklung von Softwaresystemen haben. Dieses könnte daran liegen, dass aktuelle Architekturbeschreibungen (noch) nicht für die Unterstützung der Weiterentwicklung geeignet sind und folglich Fragestellungen der Software-Evolution in der Architektur-Beschreibung deutlicher berücksichtigt werden sollten. Ebenso ist zu untersuchen, ob die Weiterentwicklung von Softwaresystemen wirklich ohne die explizite Betrachtung der Ursprungs- und Quell-Architektur durchgeführt werden, oder ob das Architekturwissen nicht implizit genutzt wird.

Es ist geplant, die Bedeutung der Software-Architektur in einem Folgeworkshop vertieft zu behandeln.

Der Workshop "Software-Architektur und Migration" wurde gemeinsam von den GI-Fachgruppen *Software-Architektur* (<http://sdq.ipd.uka.de/research/fgswarch>) und *Software-Reengineering* (<http://www.uni-koblenz.de/sre/>) durchgeführt. Die Organisatoren danken herzlich den Referenten und Autoren, die ihre Ideen und Lösungsansätze im Workshop zur Diskussion stellten, den Teilnehmern, die sich aktiv an den Diskussionen zu den Beiträgen beteiligten und den Organisatoren der Software-Engineering 2007, die die Durchführung des Workshops möglich machten und eine optimale Diskussionsumgebung schufen.