

## Qualitätsgetriebenes Reengineering



### Fachvorträge im Rahmen des Berlin-Brandenburger Software-Forums

**Zeit:** 10.11.2005, 18:00 bis 20:30 Uhr

**Ort:** Fraunhofer ISST, Mollstraße 1, 10178 Berlin

Ein Großteil der in Betrieb befindlichen Softwaresysteme ist heute 10-15 Jahre alt. 80 Prozent der IT-Budgets fließen nicht in die Entwicklung neuer Software, sondern in die Erweiterung, Anpassung und Korrektur bestehender Systeme. Systematische Reengineering-Aktivitäten zur Restrukturierung, Migration oder Ablösung von Softwaresystemen bzw. zur Analyse ihrer Eigenschaften (Reverse Engineering) gewinnen damit für das langfristig erfolgreiche Management großer Softwaresysteme zunehmend an Bedeutung.

Das Programm sieht folgende Punkte vor:

- **Vorstellung des Virtuellen Software Engineering-Kompetenzzentrums** (Thomas Zehler, BTU Cottbus)
- **Reengineering – Strategien, Techniken, Ursachen** (Dr. Markus Pizka, TU München, itestra GmbH)

Ungeachtet der großen Bekanntheit des Begriffs "Reengineering" besteht nach wie vor wenig Einigkeit darüber, was Reengineering genau bedeutet, wann es notwendig ist, wie man es durchführt oder noch besser vermeiden kann. Nach einer Einordnung und Abgrenzung der verschiedenen Begrifflichkeiten, die in diesem Umfeld eine Rolle spielen, gibt der Vortrag einen Überblick über verschiedene Ansätze des Reengineering und diskutiert dabei sowohl unterschiedliche Strategien, wie "Chicken-Little" versus "Cold-Turkey", als auch die oftmals überbewertete Rolle der gewählten Technologie, s. service-orientierte Architekturen. Da Reengineering in der Regel teuer und riskant ist, erläutert der Vortrag abschließend Aspekte der Kriterien und Maßnahmen des Qualitätsmanagements, die dazu beitragen können, die Notwendigkeit für Reengineering-Maßnahmen zu minimieren.

- **Unsichtbares Reengineering: Chancen zur Risikominimierung und Kostensenkung** (Dr. Frank Simon, SQS Software Quality Systems AG)

Bereits das sichtbare Reengineering, d.h. das Verändern einer Applikation bzgl. neuer Vorgaben wie z.B. ein neues Look&Feel, ist schwer zu motivieren und finanziell zu rechnen. Noch schwieriger ist dies häufig für das unsichtbare Reengineering, in dessen Fokus lediglich technische Interna stehen und deren Änderung i.d.R. keine nach außen sichtbare Wirkung hat. Dies führt in der Industrie häufig zu einem sehr sorglosen Umgang mit technischen Eigenschaften eines Systems, was mittelfristig große Risiken und deutlich überhöhte Aufwände bei der Wartung und Weiterentwicklung bedeuten kann. In diesem Vortrag soll gezeigt werden, was typische unsichtbare Risiken sind, wie sie erkannt werden können und wie ihre Behebung angestoßen und überprüft werden kann. Anhand einiger Praxisbeispiele wird gezeigt, wie ein solches unsichtbares Reengineering nicht nur zu reduzierten Risiken, sondern auch zu deutlich reduzierten Wartungsaufwänden beitragen kann. Gleichzeitig werden typische Fehler und Probleme aufgezeigt, die hierbei in der Praxis häufig auftreten, und mit Lösungsvorschlägen kontrastiert.

- **Corporate Software Renewal** (Reinhard Irsigler, Intervista AG)

Software altert, nicht aus physikalischen Gründen, sondern weil sich die Paradigmen erneuern, was aufgrund von Technologieschüben der Fall ist. Den Unternehmen beschert dieser Umstand nicht unerhebliche Kosten. Diese haben große Summen in Software investiert und auch wenn diese Systeme die Unternehmen in Ihren Kompetenzen gut unterstützen, müssen sich diese früher oder später davon aus vielfältigen Gründen trennen. Wenn inzwischen keine Standardsysteme als Ersatz zur Verfügung stehen, müssen die Altsysteme erneuert werden. Auch bei komplettem Verständnis des Systems ist eine quasi „zeilenweise“ Umsetzung von alt nach neu kein gangbarer Weg mehr. Neue Anforderungen, wie „one face to the customer“, Zugangs- und Skalierungstechniken, durch Application-Servertechnik bereitgestellt, stellen gänzlich neue Realisierungsanforderungen. Da sich im Unternehmen auch die Prozesse verändert haben, sind oft auch nicht einmal die zentralen Algorithmen für eine Migration zu gebrauchen. Worauf soll sich dann der Begriff „Reengineering“ noch beziehen? Es sind die Datenbestände, die es weiter zu verwenden gilt. Diese stellen meist einen größeren Wert dar, als die Programme selbst. Die Techniken, die Daten sinnvoll zu überführen, sind Gegenstand einiger Projekte bei INTERVISTA. Hier werden also nicht Belange des Reengineering als „Refactoring“, als ständiges Verbessern von Software im Sinne einer Wartung, sondern als Ersatz von Altsystemen behandelt. Die Herausforderung der Migration besteht dann besonders darin, den Prozess der System-Erneuerung während des normalen Betriebes im Unternehmen durchzuführen.

- **Lektionen aus einem Reengineering-Projekt** (Mathias Meyer, ASDIS Software AG)

Kürzere Time-to-Market neuer Features und damit die Zufriedenheit des Kunden ist ein entscheidendes Ziel von qualitätsgetriebenem Reengineering. Deshalb entschloss sich die ASDIS Software AG ein existierendes Projekt zu refaktorisieren, um diesem Maßstab in Zukunft besser gerecht werden zu können. Auf Basis von J2EE wurde das Produkt mit einer gesunden Balance geschäftlicher und technischer Aspekte refaktoriert, und konnte innerhalb der vergangenen Monate riesige Fortschritte vorweisen. Die gelernten Lektionen aus dem Projekt haben dazu einen wichtigen Beitrag leisten können.

- **Diskussion.**

Die Teilnahme ist kostenlos. Da die Kapazität begrenzt ist, wird um eine Anmeldung bis zum **08.11.2005** auf der Seite [www.forumbb.de](http://www.forumbb.de) gebeten. Ihre Teilnahmebestätigung berechtigt Sie dann zur Teilnahme.

## Umfrage

In der Vorbereitung zum Themenabend bitten wir Sie um aktive Unterstützung. Die Online-Umfrage "Qualitätsgetriebenes Reengineering" zum Themenabend finden Sie auf unserer Webseite unter dem Menüpunkt "Umfrage" oder über <http://www.forumbb.de/umfrage/reeng.php>. Sie dauert 15 Minuten. Die Antworten werden ausgewertet und Ihnen die Ergebnisse in anonymisierter Form zugesandt. Über Ihre Beteiligung an der Umfrage würden wir uns freuen.

## Referenten

### **Dr. Markus Pizka**

Nach dem Studium der Informatik 1989-1994 promovierte Dr. Markus Pizka 1999 an der Technischen Universität München im Bereich Systemarchitektur zum Thema "Integriertes Management erweiterbarer verteilter Systeme". In dieser Arbeit beschäftigte er sich mit der Entwicklung von Programmiersprachen, Übersetzern und Betriebssystemen. Nach einem Forschungsaufenthalt bei Microsoft Research Ltd. in Cambridge, UK übernahm er 2000 bei einem deutschen Softwarehaus die Leitung kommerzieller Software-Projekte. Motiviert von der Beobachtung, dass sowohl im kommerziellen als auch im akademischen Umfeld, die Fähigkeit zur Änderung, Integration und Erweiterung vorhandener Software von herausragender Bedeutung ist, kehrte er 2001 an die Technische Universität München zurück und schloss sich der Software-Engineering Gruppe von Prof. Broy an. Herr Pizka gründete das Kompetenzzentrum "Software-Maintenance", das in enger Zusammenarbeit mit der Industrie an Fragestellungen der Optimierung der Software-Wartung arbeitet sowie Vorlesungen und Schulungen zu diesem Thema anbietet.

### **Dr. Frank Simon**

Dr. Frank Simon ist seit über 4 Jahren Mitarbeiter der SQS Software-Quality-Systems AG, dem führenden Dienstleister im Bereich des Qualitätsmanagements. Er leitet dort ein Kompetenzzentrum, das Dienstleistungen rund um Code-Qualität und Kennzahlensysteme anbietet und bei vielen Großkunden bereits erfolgreich eingebracht hat. Darüber hinaus leitet er für die SQS seit über 2 Jahren das Forschungsprojekt QBench, das wesentlichen Input für neue Ideen des unsichtbaren Reengineering liefert.

### **Reinhard Irsigler**

Studierte Informatik der TU Berlin und an der UC Santa Barbara. Wissenschaftlicher Angestellter im Hahn-Meitner Institut Berlin. 10 Jahre Geschäftsführer im CAD-Unternehmen System Consult Berlin/München. Danach Gründung und Leitung einer Firma für Order-Management und CRM-Software für den Mobilfunkmarkt. Nun Aufsichtsratsvorsitzender der im Jahr 2000 mitgegründeten INTERVISTA AG.

### **Mathias Meyer**

Nach dem Studium der Informatik hat Mathias Meyer im September 2003 als Software-Entwickler bei der ASDIS Software AG angefangen und betreut seit einigen Monaten das Server-Team des Projektes ASDIS ACM als Projektleiter. Derzeit liegt sein Fokus dabei auf agiler Software-Entwicklung und Projektleitung.

## Weitere Informationen

Weitere Auskünfte über das Berlin-Brandenburger Software-Forum erteilen folgende Ansprechpartner:



*Thomas Zehler (BTU Cottbus), Tel. (03 55) 69 – 38 15,  
[thomas.zehler@informatik.tu-cottbus.de](mailto:thomas.zehler@informatik.tu-cottbus.de)*



*Michael John (Fraunhofer FIRST), Tel. (0 30) 63 92 – 17 82,  
[michael.john@first.fraunhofer.de](mailto:michael.john@first.fraunhofer.de)*



*Andreas Mühlhausen (Fraunhofer ISST), Tel. (0 30) 2 43 06 – 4 42,  
[andreas.muehlhausen@isst.fhg.de](mailto:andreas.muehlhausen@isst.fhg.de)*