

## **Modellbasierte Migration von Unternehmensanwendungen durch Kombination eines *Top-down-* und *Bottom-up-*Ansatzes**

M.Sc. Michael Goll, Lehrstuhl für Softwaretechnik, Ruhr-Universität Bochum

Softwaresysteme werden meist länger als geplant eingesetzt. Es können Lebensdauern von über 30 Jahren erreicht werden. Im Laufe der Nutzung müssen Änderungen eingepflegt werden, die oftmals im Vorfeld nicht abzusehen waren. Es können daher Softwaresysteme entstehen, die selbst durch die ursprünglichen Programmierer nur noch partiell verstanden werden. Die Änderungen beziehen sich ebenfalls auf die zugrundeliegende Datenbank. Solche Systeme können nicht zeitlich unbegrenzt im Einsatz bleiben, da die Instandhaltungskosten meist sehr hoch und Erweiterungen äußerst schwierig umzusetzen sind.

Altsysteme können mittels einer Softwaremigration ersetzt werden. Dazu werden in dieser Arbeit einige Anforderungen ermittelt. Es sollen analog zu Softwareentwicklungen Modelle eingesetzt werden, die bereits in der Analysephase dazu dienen, bestimmte Merkmale visuell darzustellen. Weiterhin soll eine Architekturverbesserung angestrebt werden, damit »Atlanten« nicht in das Neusystem übernommen werden. Eine einfache »Übernahme« ohne Änderung erfüllt diese Vorgabe nicht. Zur Umgehung des technisch komplexen Problems der Konsistenz zwischen verschiedenen heterogenen Informationssystemen (z. B. Datenbanken), soll lediglich eine Datenbank produktiv im Einsatz bleiben. Das Altsystem darf zudem nicht verändert werden, da dies ein enormes Risiko darstellt (Änderungen müssen sowohl im Alt- als auch im Neusystem implementiert werden). Bereits migrierte Teilkomponenten sollen nach der Fertigstellung direkt eingesetzt werden können, um Wartungskosten zu minimieren. Das Altsystem und diese Teilkomponenten werden also parallel eingesetzt.

Der in dieser Arbeit präsentierte modellbasierte Ansatz erfüllt die festgelegten Anforderungen. Das Altsystem greift dabei bis zur vollständigen Migration (inkl. der Daten) ohne Veränderung auf die alte Datenbank zu. Das Neusystem greift über eine proprietäre Komponente »ModORM« auf die alte und die neue Datenbank zu (Datenmigration). Die Komponente »ModORM« ist während des Migrationsprozesses das »Bindeglied« zwischen dem Neusystem und der alten und neuen Datenbank. Weiterhin wird mit dieser Komponente die Datenmigration durchgeführt.

Da die Struktur der neuen und alten Datenbank und damit auch des Neusystems (vorliegend als Modell) und der alten Datenbank unterschiedlich sein können, müssen diese Unterschiede identifiziert und überbrückt werden. Dafür können innerhalb der Komponente ModORM Strategien konfiguriert werden, die diese »Unterschiede« durch geeignete Maßnahmen kompensieren sollen (z. B. Ersetzen von Werten, Transformation). Es werden vier Strategieebenen unterschieden: Typ- und Attributstrategien (Top-down-Betrachtung) sowie Tabellen- und Spaltenstrategien (Bottom-up-Betrachtung). Typ-Strategien beziehen sich auf Maßnahmen, die den gesamten Typ betreffen und nicht nur ein einzelnes Attribut. Im Gegensatz dazu beziehen sich Attributstrategien auf ein einzelnes Attribut. Eine Tabellenstrategie führt in der Regel die eigentlichen SQL-Befehle aus (alternative Ermittlung der Daten über andere Quellen möglich). Spaltenstrategien beziehen sich dabei auf eine einzelne Spalte.

Als *Proof-Of-Concept* wurde die Migration eines bestehenden Altsystems (Web-Architektur) anhand zweier Fallstudien durchgeführt. In der ersten Fallstudie (Client-Server-Architektur) konnten die Unterschiede der alten Datenbank und des Neusystems vollständig überbrückt werden (Realisierung aller CRUD-Operationen). Bei der zweiten Fallstudie (Web-Architektur) waren die strukturellen Unterschiede zu groß, daher wurde lediglich eine Datenmigration durchgeführt.