

Business Driven Architecture

Fachliche Muster im Einsatz

Während in Industrie der Einsatz von standardisierten Komponenten und Konstruktionen bei der Fertigung selbstverständlich ist, halten diese Ideen in der Softwareentwicklung erst langsam Einzug.

Software-Entwicklung läuft immer in drei Phasen ab: Anforderungserfassung, Realisierung, Betrieb. Je nach Entwicklungsmethode wird dieser Zyklus einmal oder mehrfach durchlaufen. Existieren für den Betrieb bereits seit langem etablierte Standards und *Best Practices*, trifft dies nur eingeschränkt oder gar nicht auf die beiden anderen Phasen zu. Bei der Realisierung wird heute häufig auf Muster (*Pattern*) zurückgegriffen. Ein Muster ist eine Lösung für ein Problem in einem bestimmten Kontext, die sich als stabil und sinnvoll erwiesen hat. Muster im Kontext Software-

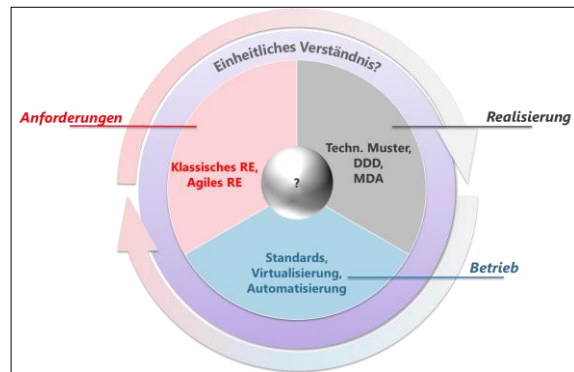


Abbildung 1: Software-Entwicklungsprozess

Realisierung sind z.B. die bekannten **GoF Patterns** (Gamma, Helm, Johnson, & Vlissides, 1995), **EAI-Patterns** (Fowler, 2002) oder die taktischen und strategischen Patterns aus dem **Domain Driven Design** (Evans, 2003), (Vernon, 2013). Mit diesen Mustern stehen Architekten und Entwicklern Kataloge von Lösungen zur Verfügung, die sie – wie in der industriellen Serienfertigung – verwenden können, um damit neue, komplexere Lösungen aufzubauen. Die Gesamtlösung erhält damit eine klarere Struktur. Entwicklungsaufwände und -zeiten lassen sich besser abschätzen, eine Automatisierung der Software-Erstellung wird damit prinzipiell machbar. Bei der Erfassung von Anforderungen wird heute immer noch primär auf textuelle Beschreibungen gesetzt, sei es das *800-seitige Anforderungsdokument* oder die *476 User Stories*. Zum einen ist es sehr schwierig oder sogar unmöglich sicherzustellen, dass alle die benannten Anforderungen vollständig und widerspruchsfrei sind, zum anderen gibt es immer einen Bruch zwischen textueller Beschreibung und der erstellten Software. Die Frage „wo müssen Änderungen an der Software vorgenommen werden, wenn die Anforderung 'X' geändert werden muss“ ist die Frage, wie die Software die fachlichen Anforderungen repräsentiert. Fachliche Anforderungen lassen sich jedoch ebenso als fachliche Muster mit entsprechenden Ontologien (Hruby, 2006) formulieren wie technische Anforderungen als

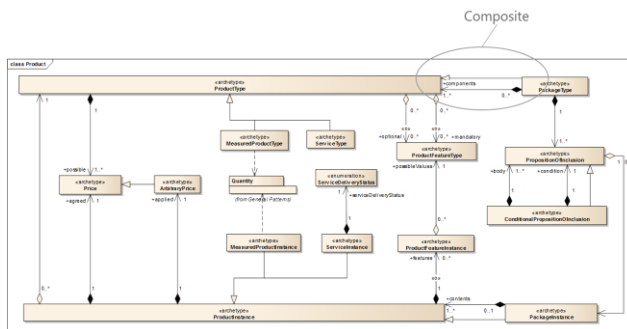


Abbildung 2: Ausschnitt eines Produktmodells

komplexere Lösungen aufzubauen. Die Gesamtlösung erhält damit eine klarere Struktur. Entwicklungsaufwände und -zeiten lassen sich besser abschätzen, eine Automatisierung der Software-Erstellung wird damit prinzipiell machbar.

Bei der Erfassung von Anforderungen wird heute immer noch primär auf textuelle Beschreibungen gesetzt, sei es das *800-seitige Anforderungsdokument* oder die *476 User Stories*. Zum einen ist es sehr schwierig oder sogar unmöglich sicherzustellen, dass alle die benannten Anforderungen vollständig und widerspruchsfrei sind, zum anderen gibt es immer einen Bruch zwischen textueller Beschreibung und der erstellten Software. Die Frage „wo müssen Änderungen an der Software vorgenommen werden, wenn die Anforderung 'X' geändert werden muss“ ist die Frage, wie die Software die fachlichen Anforderungen repräsentiert. Fachliche Anforderungen lassen sich jedoch ebenso als fachliche Muster mit entsprechenden Ontologien (Hruby, 2006) formulieren wie technische Anforderungen als

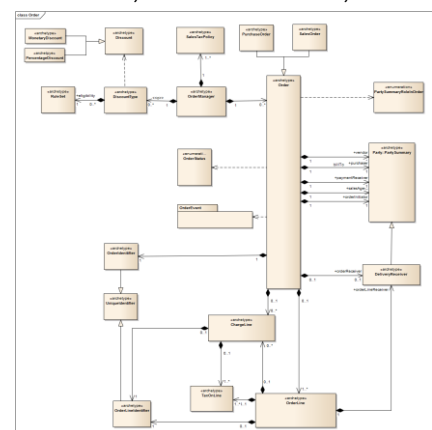


Abbildung 3: Auftragsmodell



technische Muster. In (fast) jedem Projekt tauchen in den Anforderungen Konzepte wie *Kunde, Produkt, Auftrag, Vertrag, Konto* etc. auf, ebenso wie *Verkaufen, Herstellen*, etc. Für diese existieren allgemeine Modelle (s. Abbildungen), die entsprechend der aktuellen Anforderungen angepasst aber nicht jedes Mal neu entwickelt werden müssen. Für die Umsetzung der fachlichen Muster werden dann die geeigneten technischen Muster genutzt. Diese Fachmodelle fungieren als Kommunikationsmittel zwischen Fachanforderern und Entwicklern, bringen Struktur in die Kommunikation und schaffen so ein einheitliches Verständnis. Der Übergang vom Fachmodell zum technischen Modell läuft ohne Bruch. Die Modelle können in der Phase der Anforderungserfassung als Richtlinie dienen festzustellen, ob alle Anforderungen erfasst wurden.

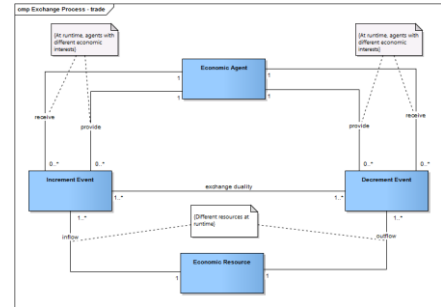


Abbildung 1: REA Modell - Verkauf

Fazit: Fachanforderer erstellen gemeinsam mit Entwicklern ein Modell ihrer Fachanforderungen auf der Basis von Fachmustern. Die Entwickler erstellen aus diesem Fachmodell unter Nutzung von technischen Mustern ein technisches Modell und setzen es um.

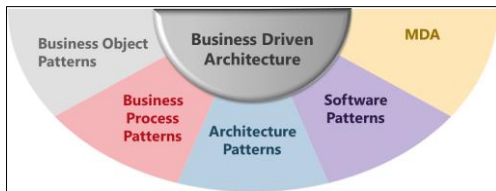


Abbildung 2: Business Driven Architecture

Die entstehende Architektur ist von den fachlichen Anforderungen getrieben. Wiederkehrende Fach- und Technismuster werden in einem sich ständig erweiternden Katalog abgelegt und stehen für weitere Projekte zur Verfügung. Damit ist ein Softwareentwicklungsprozess analog zu einem

industriellen Fertigungsprozess. Diese Vorgehensweise ermöglicht Entwicklern fachliche Standardprobleme mit passenden fachlichen Mustern beschreiben und mit passenden technischen Mustern und Standardmethoden einfach zu lösen und damit dort Raum für neue Entwicklungen zu schaffen, wo keine Standardlösung existiert.

Wir bieten:

- Beratung auf der Basis der Business Driven Architecture bei Migrations-, Entwicklungs- und Integrationsprojekten
- Unterstützung bei Projekten
- Training und Coaching

Literaturverzeichnis

Evans, E. (2003). *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Amsterdam: Addison-Wesley Longman.
 Fowler, M. (2002). *Patterns of Enterprise Application Architecture*. Amsterdam: Addison-Wesley Longman.
 Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design Patterns. Elements of Reusable Object-Oriented Software*. Addison-Wesley.
 Hruby, P. (2006). *Model-Driven Design Using Business Patterns*. Springer.
 Vernon, V. (2013). *Implementing Domain Driven Design*. Amsterdam: Addison-Wesley Longman.

Kontakt	<p><i>Peter Welkenbach</i> Principal Consultant Lyoner Strasse 15, Atricom D-60528 Frankfurt am Main Tel. +49-69-264 93 300 Fax +49-69-264 93 3019 peter.welkenbach@trivadis.com www.trivadis.com</p>	<p><i>Dr. Michael Faden</i> Senior Consultant Werdener Strasse 4 D-40227 Düsseldorf Tel. +49-211-58 6664 70 Fax +49-211-58 6664 71 michael.faden@trivadis.com www.trivadis.com</p>
----------------	---	--