

## VOM GESCHÄFTSPROZESS ZUM ANWENDUNGSSYSTEM: MODELLGETRIEBENE ENTWICKLUNG BETRIEBS- WIRTSCHAFTLICHER SOFTWARE

Betriebswirtschaftliche Anwendungen dienen einem wesentlichen Zweck: Sie sollen die Geschäftsprozesse eines Unternehmens optimal unterstützen. Eine unzureichende Berücksichtigung dieser Tatsache im Rahmen eines Softwareentwicklungsprojekts führt bestenfalls zu nicht optimalen Ergebnissen und schlimmstenfalls zu dessen Scheitern. Dieser Artikel zeigt, wie man mit Hilfe von ARIS UML Designer klassische Geschäftsprozessmodellierung als elementaren Bestandteil in ein Softwareentwicklungsprojekt integriert. Das Ergebnis ist ein von der Geschäftsprozessanalyse bis hin zum Systemdesign durchgehend modellbasierter und integrierter Ansatz zur Entwicklung betriebswirtschaftlicher Anwendungen.

Während Modellierung in der Softwareentwicklung bislang insgesamt eher zögerlich und oft nur zur (nachträglichen) Dokumentation einzelner Teilaspekte eingesetzt wurde, hat sie sich in den Fachabteilungen vieler Unternehmen längst als Standard etabliert. Modellbasiertes Erfassen, Überwachen und kontinuierliches Optimieren von Geschäftsprozessen ist zur Erlangung und Sicherung von Wettbewerbsvorteilen unabdingbar geworden. Mit Einführung der *Model Driven Architecture* (MDA) durch die OMG und ihrer eher pragmatischen Variante – der Modellgetriebenen Softwareentwicklung (MDS) – gewinnt das Thema Modellierung jedoch in jüngster Zeit auch in der Softwareentwicklung zunehmend an Bedeutung.

Es existieren bereits einige Veröffentlichungen zur Integration von Geschäftsprozessanalyse und objektorientierter Softwareentwicklung (z. B. [OOGPM]), die in der Regel dadurch gekennzeichnet sind, dass sie die in der Softwareentwicklung verwendeten Modellierungsmethoden auf die Geschäftsprozessmodellierung übertragen. Dieser Artikel stellt einen Ansatz vor, bei dem eine Integration auf Basis der in den Fachabteilungen der Unternehmen verbreiteten klassischen Geschäftsprozessmodellierung erfolgt.

### Überblick

Abbildung 1 zeigt schematisch das Vorgehen bei modellgetriebener Softwareentwicklung unter Einbeziehung der für das Projekt relevanten Geschäftsprozesse und

der Verwendung eines Template-basierten Code-Generators [OAW].

Die Geschäftsprozesse werden in Form eines Geschäftsprozessmodells erfasst und dienen als Grundlage für die Anfor-

▶ der autor



Thomas Andres (E-Mail: [thomas.andres@ids-scheer.com](mailto:thomas.andres@ids-scheer.com)) ist Manager bei IDS Scheer und dort im Entwicklungsbereich verantwortlich für die Themen UML, modellgetriebene Softwareentwicklung und Code-Generierung.

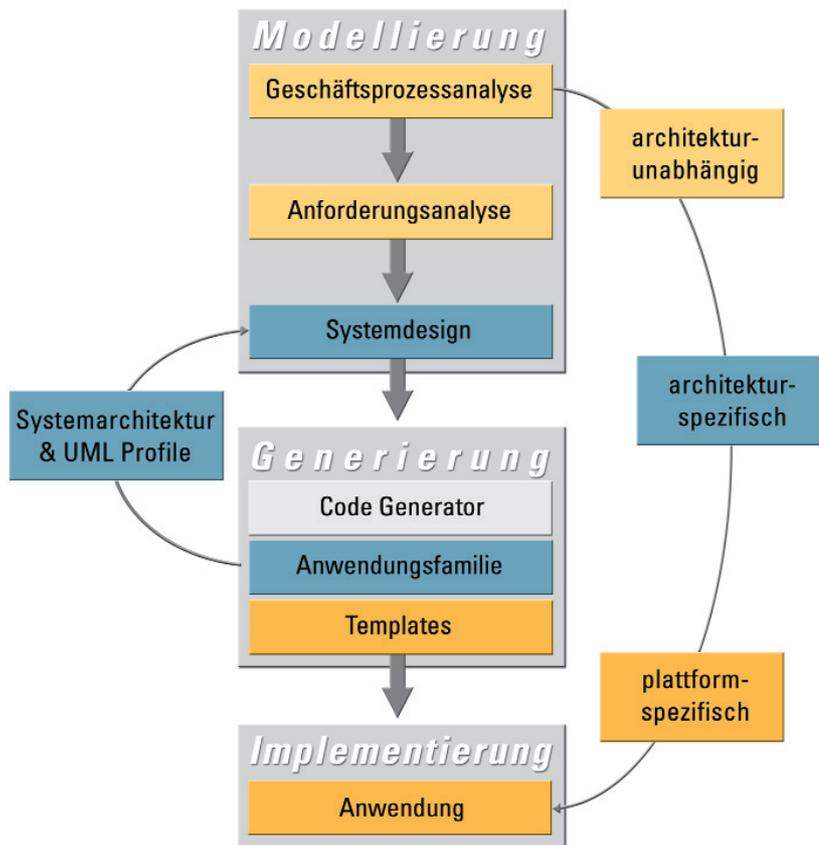


Abb. 1: Modellgetriebene Entwicklung betriebswirtschaftlicher Anwendungen

derungsanalyse. Dabei entstehen Modelle, die sowohl von der gewählten Systemarchitektur als auch von der Implementierungsplattform unabhängig sind.

Das Systemdesign erfolgt auf Basis des Analysemodells und der zur Code-Generierung ausgewählten Anwendungsfamilie, welche die Systemarchitektur und das zu verwendende UML-Profil vorgibt. Der Code-Generator setzt das Designmodell mit Hilfe der Anwendungsfamilie in Quell-Code und andere Artefakte um, die dann teilweise durch manuelle Implementierung noch vervollständigt werden müssen. Was genau aus dem Designmodell heraus generiert wird, wird durch die Templates der Anwendungsfamilie festgelegt. Idealerweise entsteht erst durch die Templates eine Abhängigkeit von der Implementierungsplattform, sodass das Designmodell lediglich von der Systemarchitektur abhängig ist.

Zu den Bereichen Systemdesign, Code-Generierung und Implementierung existieren bereits zahlreiche Veröffentlichungen (z.B. [MDS]). Viele Entwickler sind jedoch mit dem Thema Geschäftsprozessmodellierung nicht vertraut. Daher konzentriert sich dieser Artikel darauf, wie man aus einem Geschäftsprozessmodell ein objektorientiertes Analysemodell ableiten kann.

## Geschäftsprozessanalyse

Vom Standpunkt der Softwareentwicklung aus dient die Geschäftsprozessanalyse hauptsächlich der Kommunikation mit den Fachabteilungen und als Grundlage für die Anforderungsanalyse der zu erstellenden Software. Aus Sicht der Fachabteilungen ist Softwareentwicklung nur eines der vielen Anwendungsszenarien für Geschäftsprozessanalyse.

Weitere Anwendungsszenarien sind beispielsweise Geschäftsprozessoptimierung, Einführung von ERP-Systemen, Prozesskostenrechnung oder ISO-9000-Zertifizierung.

In der Praxis hat sich ARIS [ARIS] weltweit als Quasi-Standard zur Geschäftsprozessmodellierung etabliert. Einer der Kerndiagrammtypen von ARIS ist die ereignis-gesteuerte Prozesskette – kurz EPK genannt. Die EPK visualisiert, wie in einem Unternehmen Ereignisse dazu führen, dass bestimmte Funktionen angestoßen werden, die dann ihrerseits wieder Ereignisse zur Folge haben können. Den einzelnen Funktionen kann man unter anderem die ausführenden Organisationseinheiten und die zu bearbeitenden Daten zuordnen.

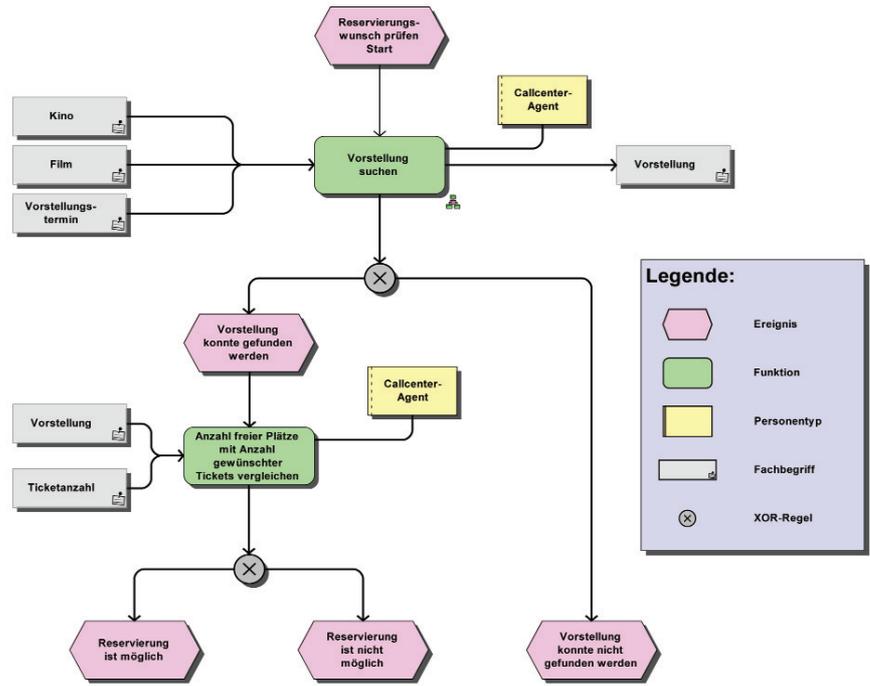


Abb. 2: Ereignis-gesteuerte Prozesskette (EPK)

Die EPK in Abb. 2 zeigt, wie im Callcenter einer großen Kinokette ein Reservierungswunsch für Kinokarten geprüft wird. Sämtliche in dem Diagramm visualisierten Elemente und Beziehungen sind in einer Datenbank abgelegt. So taucht in diesem Beispiel zweimal der Callcenter-Agent als ausführende Organisationseinheit einer Funktion auf. Dabei handelt es sich jedoch um ein einziges Datenbankelement. Dies ermöglicht eine Vielzahl von Auswertungen wie z.B.: „Welche Funktionen werden von einer Organisationseinheit im Unternehmen ausgeführt?“ und ist auch Voraussetzung für die Verknüpfung von Geschäftsprozess- und Anforderungsanalyse.

Für den Einsatz von ARIS zur Geschäftsprozessanalyse im Rahmen der Softwareentwicklung sprechen folgende Argumente:

- ARIS bietet eine modellbasierte Beschreibung aller Aspekte eines Unternehmens in einer zentralen Datenbank. Es entstehen grafische Modelle, die sich wesentlich leichter auf Konsistenz und Vollständigkeit überprüfen lassen, als eine Sammlung von Word-Dokumenten oder PowerPoint-Folien.
- Die ARIS-Methodik ist im Gegensatz zur UML sowohl für Mitarbeiter von Fachabteilungen als auch für Softwareentwickler gleichermaßen leicht verständlich. Sie fördert die Kommunikation zwischen den Beteiligten und eignet sich somit hervorragend als Grundlage für eine gemeinsame Anforderungsanalyse.
- Die Fachabteilungen modellieren ihre Geschäftsprozesse selbst und nehmen so eine aktivere Rolle im Softwareentwicklungsprozess ein.

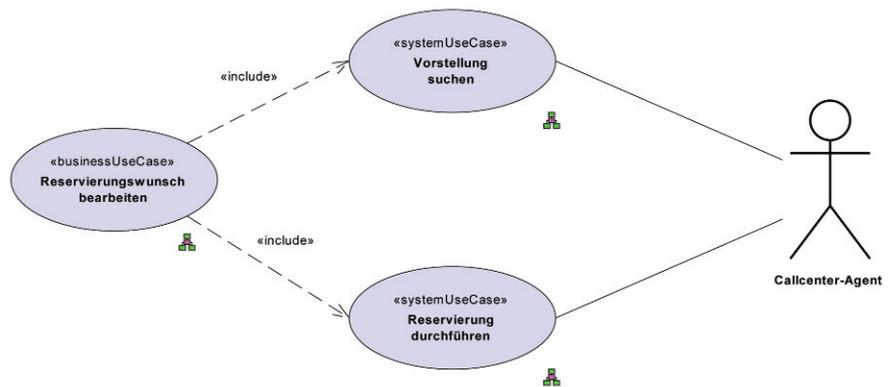


Abb. 3: System- und Geschäftsanwendungsfälle



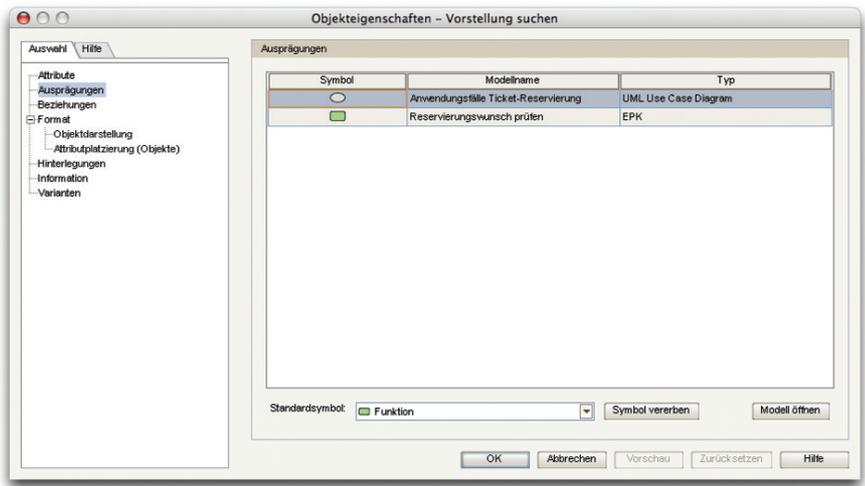


Abb. 4: ARIS-Eigenschaften der Funktion "Vorstellung suchen"

■ Oftmals ist in einem Unternehmen bereits ein detailliertes und mit ARIS erstelltes Geschäftsprozessmodell vorhanden. In diesem Fall kann man wesentlich schneller mit der Anforderungsanalyse beginnen.

## Anforderungsanalyse

So wie sich ARIS als Modellierungsmethode für Geschäftsprozesse etabliert hat, gilt dies für die UML im Umfeld objektorientierter Softwareentwicklung. Die Herausforderung besteht nun darin, beide Modellierungswelten zu integrieren und methodisch sinnvoll miteinander zu verknüpfen. Im Folgenden werden zwei

sich ergänzende Wege vorgestellt, wie man aus dem Geschäftsprozessmodell ein UML-Analysemodell ableiten kann.

## Anwendungsfallmodell

Zunächst werden in den Geschäftsprozessen alle Funktionen identifiziert, die in einem Softwaresystem implementiert werden sollen. Diese Funktionen werden in der UML-Welt als Anwendungsfälle betrachtet und in ein UML-Anwendungsfalldiagramm kopiert. ARIS stellt zu diesem Zweck eine vordefinierte Transformation zur Verfügung, die zusätzlich alle der Funktion zugeordneten Organisationseinheiten ermittelt und als UML-

Akteure in das Anwendungsfalldiagramm einfügt.

Abbildung 3 zeigt das Ergebnis einer solchen Transformation. IT-relevante Funktionen sind als Systemanwendungsfälle («systemUseCase») stereotypisiert. Zusätzlich enthält das Diagramm auch den übergeordneten Geschäftsprozess als Geschäftsanwendungsfall («businessUseCase»). Die Hinterlegungssymbole in Abb. 3 rechts unterhalb der Anwendungsfälle deuten an, dass diese durch weitere Diagramme verfeinert sind. Wesentlich bei dieser Transformation ist, dass keine neuen Elemente erzeugt, sondern vielmehr die bereits im Geschäftsprozessmodell existierenden Elemente wieder verwendet wurden. Verdeutlicht wird dies durch den in Abb. 4 gezeigten Eigenschaftsdialog der Funktion "Vorstellungen suchen". Man erkennt, dass diese Funktion sowohl in der EPK als auch im UML-Anwendungsfalldiagramm visualisiert wird.

Die Vorteile dieser Integration liegen auf der Hand. Wenn beispielsweise die Funktion im Geschäftsprozessmodell umbenannt wird, so wird der neue Name auch automatisch im UML-Anwendungsfalldiagramm angezeigt. Ferner ist auf diese Weise eine einfache und bidirektionale Navigation zwischen korrespondierenden Inhalten im Geschäftsprozess- und UML-Analysemodell möglich.

## Analyseklassenmodell

Einen weiteren Anknüpfungspunkt für die Anforderungsanalyse stellen die in Abb. 2 dargestellten Ein- und Ausgabeparameter der EPK-Funktionen – die so genannten Fachbegriffe – dar. Diese werden im Rahmen der Geschäftsprozessmodellierung üblicherweise in einem Fachbegriffsmodell zusammengefasst, um eine Verwendung einheitlicher Fachbegriffe im gesamten Projekt zu gewährleisten. Bei der Anforderungsanalyse untersucht man diese Fachbegriffe dahingehend, welche von ihnen als Kandidaten für Fachklassen und welche eher als Fachattribute in Frage kommen. Die so ermittelten Fachklassen und Fachattribute werden im gleichen Diagramm platziert und mit dem jeweiligen Fachbegriff verknüpft.

Abbildung 5 zeigt einen Ausschnitt eines solchen Fachbegriffsmodells. Auch hier stehen Transformationen zur Verfügung, mit denen sich diese Schritte automatisieren lassen.

Im nächsten Schritt werden die Fachklassen in ein UML-Klassendiagramm

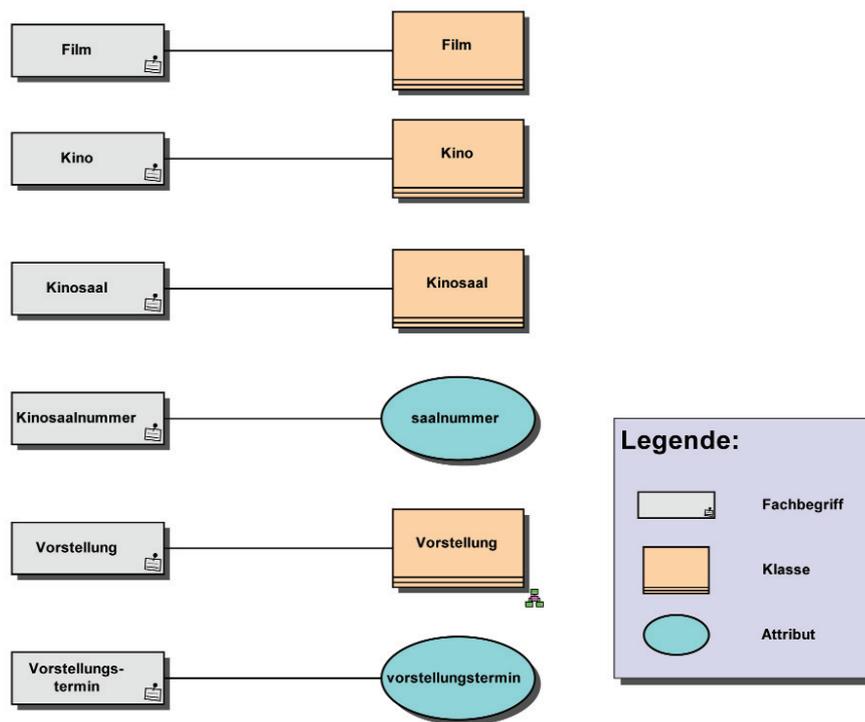


Abb. 5: Fachbegriffsmodell mit Fachklassen und Fachattributen

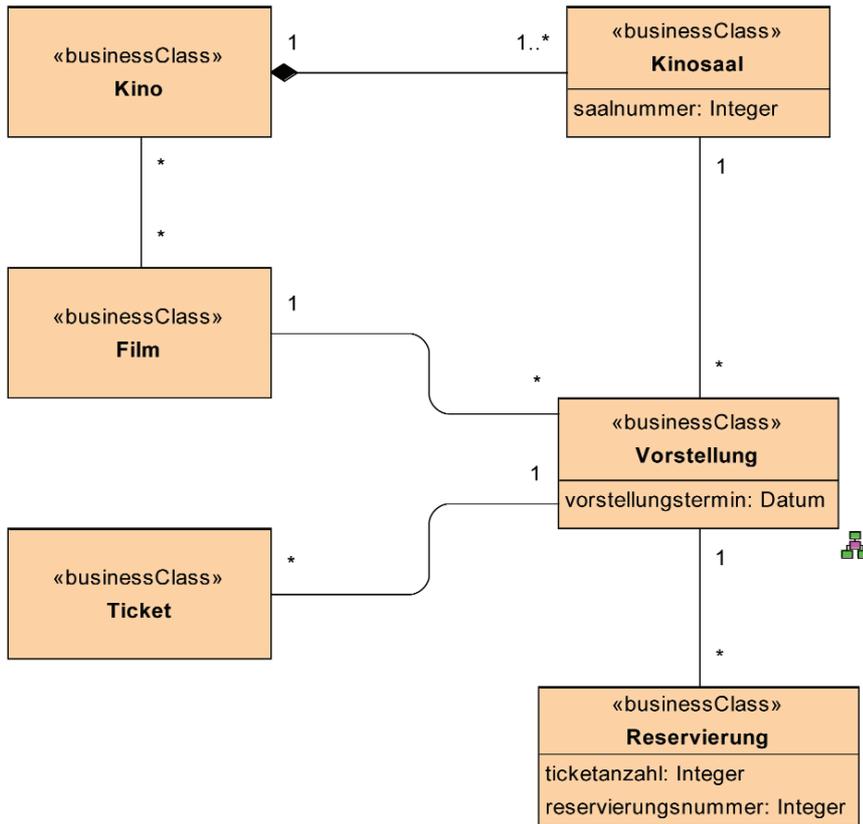


Abb. 6: Analyseklassendiagramm

kopiert, um ihnen dort die Fachattribute zuzuordnen und die Klassen mittels binärer Assoziationen in Beziehung zu setzen. Das Ergebnis ist in Abb. 6 dargestellt. Die Fachklassen können jetzt noch um weitere fachliche Attribute und Operationen ange-reichert werden.

Die Ableitung des Analysemodells aus dem Geschäftsprozessmodell erfolgt in enger Zusammenarbeit von Fach- und IT-Abteilung. Während beispielsweise die Fachabteilung besser beurteilen kann, welche EPK-Funktionen und Fachbegriffe für die zu entwickelnde Software überhaupt relevant sind, kann der objektorientierte Analytiker eher entscheiden, ob ein Fachbegriff einer Fachklasse oder einem Fachattribut entspricht. Auch bei Ausarbeitung des UML-Analysemodells ist die Mitarbeit der Fachabteilung hilfreich, da diese besser beurteilen kann, wie die einzelnen Fachklassen miteinander assoziiert sind.

ARIS UML Designer ermöglicht die Verknüpfung zwischen UML- und klassischer ARIS-Modellierung dadurch, dass beiden Modellierungsmethoden das gleiche Objektmodell zugrunde liegt. Für die UML-Modellierung ist lediglich ein UML-Objektmodell auf das gemeinsame ARIS-Objektmodell aufgesetzt. Zur Bearbeitung der UML-Elemente stehen daher jeweils

ein spezifischer UML-Eigenschaftsdialog (siehe Abb. 7) und ein klassischer ARIS-Eigenschaftsdialog (siehe Abb. 8) zur Verfügung. Neben der in Abb. 4 gezeigten Ausprägungsliste enthält dieser Dialog auch eine Liste aller Beziehungen, die das Element zu anderen Elementen hat, und ermöglicht dadurch eine Navigation vom UML-Element zum verknüpften ARIS-Element (im gezeigten Beispiel von der Fachklasse zum Fachbegriff).

Neben den hier vorgestellten Möglichkeiten zur Verknüpfung klassischer

Geschäftsprozessmodellierung mit UML-Modellierung existieren noch etliche weitere. So lassen sich beispielsweise EPK-Funktionen auch als UML-Operationen oder UML-Aktivitäten (ActionState), EPK-Ereignisse als Trigger von Zustandsübergängen oder ERM-Attribute als UML-Attribute wieder verwenden. Grundsätzlich erfolgt die Verknüpfung beider Modellwelten entweder durch die direkte Wiederverwendung eines Geschäftsprozesselements als UML-Element oder durch die Verknüpfung mit einem UML-Element. Darüber hinaus besteht die Möglichkeit, fachliche Elemente durch UML-Diagramme zu verfeinern. In allen Fällen ist eine einfache bidirektionale Navigation zwischen beiden Modellwelten gewährleistet.

## Systemdesign und Generierung

Das UML-Analysemodell dient als Grundlage für ein Designmodell, auf dessen Basis die Code-Generierung erfolgt. ARIS UML Designer bietet zwei Möglichkeiten für die Anbindung von Code-Generatoren: via XML-Schnittstelle oder integriert über Plugins. Letztere haben den Vorteil, dass die Code-Generierung direkt auf dem internen UML-Objektmodell arbeiten kann und der Umweg über das Dateisystem entfällt. Mit der aktuellen Version stehen dem Anwender ein entsprechendes Plugin für den Open-source-Generator openArchitectureWare [OAW] (siehe Abb. 9) und ein SDK zur Entwicklung eigener Plugins zur Verfügung.

## Fazit

Geschäftsprozessanalyse ist für den Erfolg von Softwareentwicklungsprojekten im betriebswirtschaftlichen Umfeld essentiell.

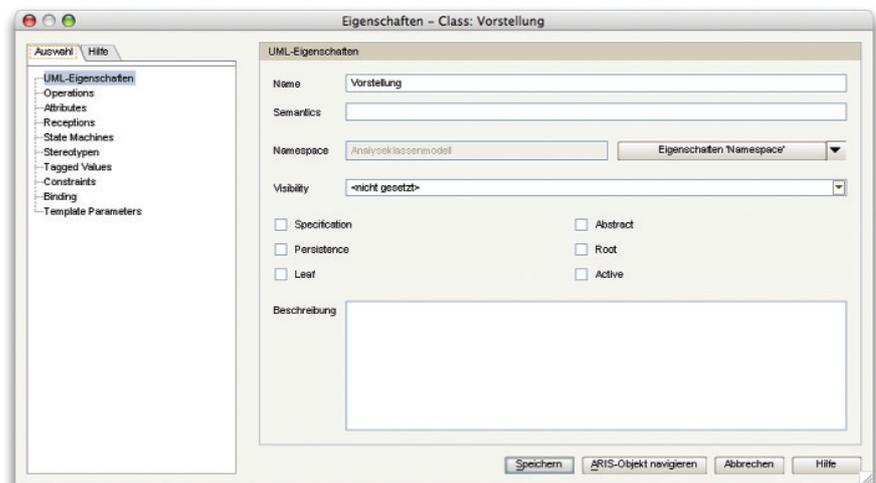


Abb. 7: UML-Eigenschaften der Analyseklasse "Vorstellung"



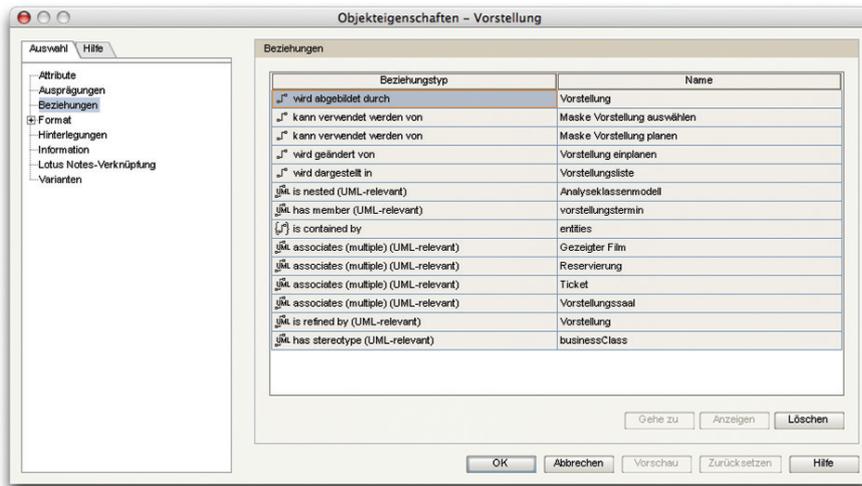


Abb. 8: ARIS-Eigenschaften der Analyseklasse "Vorstellung"

Erfolgt diese modellbasiert, kann die Anforderungsanalyse auf Basis einer formalen Geschäftsprozessbeschreibung stattfinden. Erfolgt die Geschäftsprozessanalyse darüber hinaus mit Methoden, mit denen die Fachabteilungen vertraut sind, führt dies zu einem vollständigeren und fachlich konsistenten Geschäftsprozessmodell.

Die technische Integration von Geschäftsprozess- und UML-Modellierung ermöglicht eine Ableitung des UML-Analysemodells aus dem Geschäftsprozessmodell unter Verwendung von Transformationen. Die methodische Integration gewährleistet darüber hinaus eine Verknüpfung der jeweils korrespon-

dierenden Modellinhalte und ermöglicht eine bidirektionale Navigation zwischen beiden Modellwelten.

MDA und MDSD zeichnen sich dadurch aus, dass Änderungen stets auf der Abstraktionsebene durchgeführt werden sollten, mit der die Änderung semantisch korrespondiert. Bei betriebswirtschaftlichen Anwendungen betrifft dies oftmals die Ebene der Geschäftsprozesse. Mit ARIS UML Designer steht hierfür eine durchgehende und integrierte Lösung zur Verfügung, die alle Aspekte von der klassischen Geschäftsprozessmodellierung über objektorientierte Analyse und Design bis hin zur Code-Generierung abdeckt. ■

## Links

- [ARIS] ARIS UML Designer und ARIS Platform, siehe: <http://www.aris-platform.com>
- [MDSD] T. Stahl, M. Völter, Modellgetriebene Softwareentwicklung, dPunkt 2005, siehe auch: <http://www.mdsd-buch.de/>
- [OAW] openArchitectureWare, siehe: <http://www.architectureware.org/>
- [OOGPM] B. Oestereich, Ch. Weiss, C. Schröder, T. Weilkens, A. Lenhard, Objektorientierte Geschäftsprozessmodellierung mit UML, dPunkt 2003, siehe auch: <http://www.oogpm.de>
- [P2A] A.-W. Scheer, W. Jost, K. Wagner, IDS Scheer AG (Hrsg.), Von Prozessmodellen zu lauffähigen Anwendungen, Springer 2005, sieht auch: <http://www.ids-scheer.com/p2a-buch>

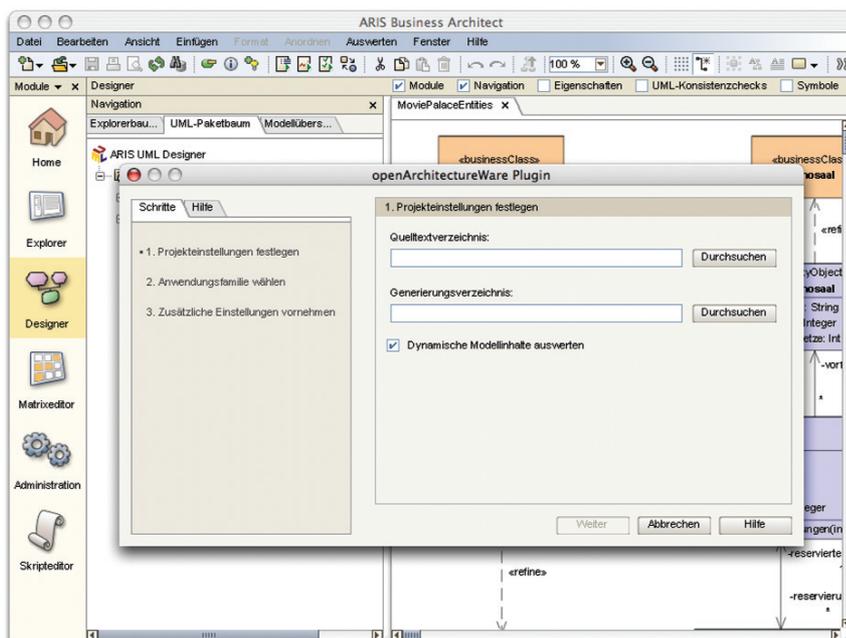


Abb. 9: openArchitectureWare-Plugin