

# Entwurf domänenspezifischer Modelle im Web mit Oryx

Matthias Kunze, Mathias Weske

Hasso Plattner Institut an der Universität Potsdam  
Prof.-Dr.-Helmert-Straße 2-3, 14482 Potsdam  
{matthias.kunze,mathias.weske}@hpi.uni-potsdam.de

Die stetige Zunahme der Komplexität von Informationssystemen erschwert den Systementwurf mit herkömmlichen Methoden. Universelle Modellierungssprachen wie UML stoßen an ihre Grenzen, da sich die fachlichen Anforderungen und Eigenschaften eines derart entworfenen Systems oft in der Menge technischer Details verlieren.

Der Einsatz domänenspezifischer Modellierungssprachen verspricht Abhilfe: Der Systementwurf wird auf fachliche Konzepte statt seiner Implementierung abstrahiert und Domänenexperten werden befähigt, aktiv an Entwurf und Dokumentation des Systems teilzunehmen. Um domänenspezifische Modellierungssprachen effektiv einsetzen zu können, benötigt man jedoch Werkzeuge mit denen sich entsprechende Modelle entwickeln lassen. In dem Artikel wird die erweiterbare Modellierungsumgebung Oryx vorgestellt, die eine Vielzahl von Modellierungssprachen unterstützt, sowie die Umsetzung domänenspezifischer Modellierungssprachen in Oryx.

Eine Modellierungssprache besteht aus den Konzepten Notation, Syntax und Semantik [JS06]. Üblicherweise werden Syntax und Semantik durch formale bzw. semiformale Festlegungen definiert, z.B. durch das UML-Metamodell. Im Oryx Editor existieren *Stencil Sets* – textuell definierte Beschreibungen der graphischen Repräsentation der spezifischen Konzepte der Domäne (Notation) und Kompositionsregeln (Syntax) von Modellelementtypen. Semantik wird als Abbildung der Domänenkonzepte im Modell auf eine bestimmte Bedeutung verstanden. Dies wird in Oryx durch *Plugins* realisiert, mit denen sich Modelle beispielsweise abwickeln oder, im Rahmen modellgetriebener Ansätze, in andere Sprachen transformieren lassen.

In Oryx sind sowohl Anwendung als auch Modelle Ressourcen im Web. Das vereinfacht Installation und Datenhaltung und öffnet die Modellierungsplattform einem breiten Benutzerspektrum. Oryx besteht aus zwei Komponenten: Der *Editor* dient zur Entwicklung von Modellen verschiedener Modellierungssprachen. Mithilfe des *Repository*s können entwickelte Modelle verwaltet, markiert (tagging) und bewertet (rating) werden. Auf dessen Grundlage wird auch die Zusammenarbeit mit Dritten (sharing) realisiert, indem Benutzern Rechte zur Betrachtung oder Bearbeitung von Modellen eingeräumt werden.

[JS06] Ethan K. Jackson und Janos Sztipanovits. Towards a formal foundation for domain specific modeling languages. In *EMSOFT '06: Proceedings of the 6th ACM & IEEE International conference on Embedded software*, Seiten 53–62, New York, NY, USA, 2006. ACM.