

# Methoden- und Werkzeugunterstützung für ontologiebasierte Software-Entwicklung (OBSE)

Andrej Bachmann

Fachbereich Mathematik und Informatik  
Philipps-Universität Marburg  
Hans-Meerwein Str. 10  
35032 Marburg  
rodionov@mathematik.uni-marburg.de

**Abstract:** Dieser Artikel liefert einen Überblick über die Aufgabenstellung und die Lösungsansätze, die im Rahmen meiner Dissertation erarbeitet werden sollen. Sie ist ein Bestandteil unseres laufenden Projekts zur *ontologie-basierten Software-Entwicklung (OBSE)*. Dieses wird zunächst kurz vorgestellt, um das Umfeld der Aufgabe zu skizzieren. Die Strukturierung der Domänen-Ontologien und deren Integration in die klassische Software-Entwicklung werden ausführlicher betrachtet, da dieser Bereich der *OBSE*-Prozess-Unterstützung der Schwerpunkt meiner Arbeit ist.

## 1 Einführung

In den letzten Jahren ist das Interesse für die Definition und die praktische Anwendung von Domain-Ontologien stark gestiegen. Ausgelöst wurde diese Entwicklung vor allem durch die Suche nach neuen Wissensrepräsentationsformen in dem Bereich Künstliche Intelligenz. Beschleunigt wurde sie durch die *Semantic Web*-Initiative. Zurzeit werden Möglichkeiten der Verwendung von Ontologien<sup>1</sup> in dem Bereich der Software-Technik und der damit verbundene Nutzen intensiv untersucht. Dabei sollen die Domänen-Ontologien vor allem die projektübergreifende Wiederverwendung der Elemente des Untersuchungsbereichs und die Erstellung einer gemeinsamen Wissensbasis unterstützen.

Die Palette der Sprachen, die sich als Beschreibungsmittel für Domäne-Ontologien eignen, ist inzwischen sehr breit gefächert. Das Spektrum reicht von formalen Sprachen wie *Frame Logic (F-Logic)* über aus der *Semantic Web*-Initiative hervorgegangene Sprachen *Resource Description Framework (RDF)* oder *Ontology Web Language (OWL)* bis hin zu Modellierungssprachen wie *UML*. Unterschiedliche Einsatzbereiche dieser Sprachen und ihre Eigenschaften führen zu der Frage, welche aus dieser Palette sich für die Beschreibung von Domäne-Ontologien für Software-Entwicklungsprozesse anbieten würden. Ist eine passende Sprache gefunden, dann muss definiert werden, wie

---

<sup>1</sup> Der Begriff Ontologie wird in diesem Artikel entsprechend der aktualisierten Definition von Herrn T. Gruber verwendet. Für weitere Informationen siehe <http://tomgruber.org/writing/ontology-definition-2007.htm>

die Integration der Domänen-Ontologien in einen Software-Entwicklungsprozess erfolgen soll. Im folgenden Kapitel wird ein Verfahren vorgestellt, dass sich mit diesen Fragen befasst.

## 2 Der OBSE-Prozess

Die zentrale Idee der *ontologie-basierten Software-Entwicklung* [BHR+07] ist die Anreicherung der konventionellen Softwareentwicklung um die Möglichkeit in früheren Phasen, meistens zeitnah an der Anforderungsanalyse, einen Wissenstransfer **aus** der Domäne-Ontologie und im weiteren Projektverlauf zurück **in** die Domäne-Ontologie zu ermöglichen und zu unterstützen. Die Vorteile dieser zyklischen Anbindung liegen sowohl an der Seite der Projekte, indem sie auf dem schon vorhandenen Domäne-Wissen aufbauen können, als auch auf der Seite der Domäne-Ontologie, da diese durch Informationen aus den laufenden oder vor allem abgeschlossenen Projekten zuerst entsteht und dann weiter ausgebaut oder aktualisiert wird.

Einen Überblick über den *OBSE*-Prozess liefert die Abbildung 1. Ausgehend von den Projektanforderungen findet zunächst eine semi-automatische Transformation von diesen in ein konzeptuelles Modell (Daten- und Ablaufmodell) statt. Mit Hilfe einer so genannten *Import-Brücke* ist es möglich, Elemente dieses Modells durch weitere Elemente aus der Domänen-Ontologie anzureichen. Das konzeptuelle Modell dient als Ausgangsbasis für die weitere System-Umsetzung. Während deren Durchführung ist in den meisten Fällen zu erwarten, dass am System-Modell weitere Änderungen durchgeführt werden. Aus diesem Grund sieht der Prozess vor, dass diese Änderungen in das konzeptuelle Modell übertragen werden. Wurden im Laufe des Projekts Modellelemente erarbeitet, die die Domäne-Ontologie erweitern oder genauer spezifizieren, besteht die Möglichkeit, diese Elemente über die *Export-Brücke* in die Ontologie zu integrieren.

Aus der Abbildung wird außerdem sichtbar, dass sowohl die Domänen-Ontologie als auch das konzeptuelle Modell die gleiche sprachliche Grundlage haben. Die *Klagenfurt Conceptual Predesign Method (KCPM)* wurde zunächst als eine zusätzliche konzeptuelle Ebene zwischen den Anforderungen und der System-Modellierung definiert [M-K 02]. Damit sollte der konzeptuelle Abstand zwischen den Anforderungen und Modellierung (die meistens in *UML* erfolgt) verkleinert werden. In späteren Untersuchungen wurde festgestellt, dass *KCPM* sich zusätzlich als eine Sprache für Beschreibung von Domänen-Ontologien eignet [KMZ 04]. Somit kann *KCPM* sowohl auf der Projekt- als auch auf



Abbildung 1 Überblick über den OBSE-Prozess

der Ontologie-Seite zum Einsatz kommen. Dies vereinheitlicht das Prozessmodell und führt zu Vorteilen bei der Konzeption der Export- und Import-Brücken.

### 3 Aufgabenstellung und Lösungsansätze

Bei der Definition des *OBSE*-Prozesses wurden in vorhergegangenen Arbeiten unserer Arbeitsgruppen schon die Entscheidungen bezüglich der geeigneten Ontologie-Sprache und des passenden Vorgehensmodells getroffen (siehe [BHR+07], [BHR 07]). Zusätzlich wurde Vorarbeit im Bereich der Integration der *KCPM*-Glossare geleistet [V-M 05], in der Regeln definiert wurden, die bei der Integration angewendet werden. Ganz aktuell beschäftigen wir uns mit Möglichkeiten der Strukturierung von Domäne-Ontologien.

Aufbauend auf den vorhandenen Forschungsergebnissen soll im Rahmen meiner Arbeit die Verknüpfung zwischen der Domänen-Ontologie und dem konzeptuellen Modell eines Projekts ausgearbeitet werden. Insbesondere gehören dazu Prozesse, die mit der Benutzung der Import- und Export-Brücken verbunden sind. Welche Aktivitäten werden bei diesen Vorgängen ausgeführt? Welche Möglichkeiten für die Automatisierung der Abläufe - bedingt durch die Integrationseinschränkungen - stehen zur Verfügung? Auch die Auswirkungen von unterschiedlicher Strukturierung der Glossare auf der Seite der Domänen-Ontologie auf die Import- und Export-Brücken sollen untersucht werden. Gibt es Varianten, die Wissenstransfer erleichtern und zum Beispiel zu weniger Konflikten führen?

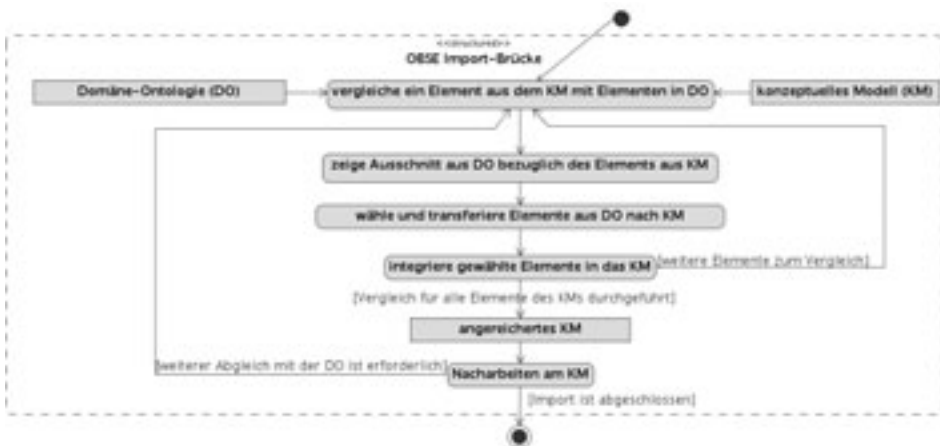


Abbildung 2: Import-Brücke

Mein erster Vorschlag für die Abläufe im Rahmen der Import-Brücke ist in der Abbildung 2 als ein Auszug aus einem Aktivitätsdiagramm dargestellt. Das hier abgebildete Verfahren ist auf den Rückmeldungen der Benutzer basiert und sieht vorerst noch keine Automatismen wie z. B. das Übertragen und Integrieren von mehreren Elementen aus der Domäne-Ontologie in einem Schritt vor.

Die Elemente aus dem konzeptuellen Modell des Projekts (entstanden durch Extrahieren aus den Projektanforderungen) werden benutzt, um nach entsprechenden Elementen in der Domänen-Ontologie zu suchen. Falls ein ähnliches Element entdeckt wurde, wird es zusammen mit seiner Umgebung, die über Assoziationen ausgehend aus diesem Element ermittelt wird, dem Benutzer angezeigt. Er kann dann zu importierende Elemente bestimmen und den Import starten. Wurden diese Schritte für alle Elemente aus dem konzeptuellen Modell des Projektes ausgeführt, entsteht ein durch Elemente der Domänen-Ontologie angereichertes konzeptuelles Modell im Projekt, das anschließend weiterbearbeitet werden kann.

Als interessant im Zusammenhang zu meiner Arbeit und generell dem *OBSE*-Prozess sind noch folgende Methoden zur Domäne-Engineering zu erwähnen: *Feature Oriented Domain Analysis (FODA)*, *Reuse-driven Software Engineering Business (RSEB)* und *Ontology-based Domain Engineering (ODE)*. Bei der Definition von Domäne-Ontologien sind außerdem Konzepte wie *Foundational Ontologies* [Gui 06] und *Formal Ontology, Conceptual Analysis and Knowledge Representation* [Gua 95] interessant, die ich in meiner Arbeit berücksichtigen will. Im Rahmen meiner Arbeit entsteht ein auf der *Rich Client Plattform (RCP)* basierendes Werkzeug zu Prozessunterstützung. Dieses ist schon jetzt in der Lage, *KCPM*-Glossare zu bearbeiten und sie nach *UML* und zurück zu konvertieren. Als ein weiteres Inkrement sind die Umsetzung der Import- und Export-Brücken und die Strukturierung der Domäne-Ontologie geplant.

Die voraussichtliche Fertigstellung der Arbeit ist zum Ende 2009 anvisiert.

## Literaturverzeichnis

- [BHR+07] A. Bachmann, W. Hesse, A. Russ, Ch. Kop, H.C. Mayr, J. Vöhringer: *OBSE – an Approach to Ontology-based Software Engineering in the Practice*. Proc. EMISA conference, St. Goar, pp. 129-142, LNI, Koellen-Verlag, 2007
- [BHR 07] A. Bachmann, W. Hesse, A. Russ: *Coupling ontology and software development processes – a rendez-vous approach*. Proc. WOMSDE, João Pessoa, Brazil, 2007
- [Gua 95] Guarino, N.: *Formal ontology in conceptual analysis and knowledge representation.*, Int. Journal of Human and Comp. Stud., 43:625–640, 1995.
- [Gui 06] Guizzardi, G.: *The Role of Foundational Ontology for Conceptual Modeling and Domain Ontology Representation*, Companion Paper for the Invited Keynote Speech, 7th International Baltic Conference on Databases and Information Systems, Vilnius, Lithuania, 2006.
- [Hes 05] W. Hesse: *Ontologies in the Software Engineering process*. In: R. Lenz et al. (Hrsg.): *EAI 2005 - Tagungsband Workshop on Enterprise Application Integration*, GITO-Verlag Berlin 2005
- [KMZ 04] Ch. Kop, H.C. Mayr, T. Zavinska: *Using KCPM for Defining and Integrating Domain Ontologies*. Proc. Int. Workshop on Fragmentation versus Integration - Perspectives of the Web Information Systems Discipline, Brisbane Australia. LNCS, Springer 2004
- [M-K 02] H.C. Mayr, Ch. Kop: *A User Centered Approach to Requirements Modeling*. Proc. Modellierung 2002, pp. 75-86. LNI p-12, Springer 2002
- [V-M 05] J. Vöhringer, H.C. Mayr: *Integration of schemas on the pre-conceptual level using the KCPM-approach*. Proc. 16th Int. Conference on Information Systems Development ISD2005. LNCS Springer 2005