

Martin Stabauer / Johann Höller

Ein integrativer Ansatz zur semantischen Modellierung von Rechtstexten

Das Zusammenwirken von Softwareentwicklung und Ontologien

The semantic representation of legal texts has for a long time been a significant object of research. Approaches from both, software engineering and semantic modeling, have led to impressive results. However, some gaps still remain. The paper tries to bridge one of these gaps between the two approaches, namely the one between generalizability and applicability. For this purpose, a hybrid framework for drafting and modeling legislation is being presented, which consists of a semantic base model and a matching general development process. This framework then was tested in an extensive case study, which is also part of this paper.

Category: Articles

Field of law: Artificial Intelligence Law

Region: Austria

Citation: Martin Stabauer / Johann Höller, Ein integrativer Ansatz zur semantischen Modellierung von Rechtstexten, in: Jusletter IT 11 December 2014

Inhaltsübersicht

- I. Einführung
- II. Verwandte Forschung
- III. Semantisches Grundmodell und Vorgehensmodell
 1. Semantisches Grundmodell
 2. Automatische Generierung von Rechtstext
 3. Vorgehensmodell
- IV. Fallstudie: Curriculumserstellung
 1. Umwelt und Setting
 2. Implementierung des Grundmodells
 - 2.1. Ontologien
 - 2.2. Semantisches Regelwerk
 - 2.3. Grafische Benutzeroberflächen
 3. Ergebnisse
- V. Fazit

I. Einführung

[Rz 1] Die Aufgabe der Übermittlung von Regeln für das geordnete Gemeinschaftsleben wird traditionell auf schriftliche Art und Weise unter Zuhilfenahme von Rechtstexten gelöst. Die Verfasser des Rechtstextes möchten die gewünschten Regelungsinhalte an die Adressaten kommunizieren und bedienen sich daher des althergebrachten Werkzeugs des Rechts- bzw. Gesetzestextes. Es stellt sich nun die Frage, ob die schriftliche Form der Kommunikation tatsächlich die einzig mögliche und optimale Variante darstellt. Gerade aus heutiger Sicht, wo Rechtstexte häufig die Grundlage für IT-Systeme bilden und sie selbst ebenso auf anderen IT-Systemen erstellt werden, ist dies anzuzweifeln. Es entstehen Medienbrüche, die durch eine direktere Art der Kommunikation vermieden werden könnten.

[Rz 2] Aktuellen Gesetzestexten wird aus Gründen ihrer Komplexität und der Mehrdeutigkeit von verwendeten Begrifflichkeiten und Formulierungen insbesondere von juristischen Laien häufig eine unzureichende Lesbarkeit und Verständlichkeit unterstellt.¹ Hier können Ontologien mit ihrer eindeutigen Begriffswelt deutliche Vorteile bieten. Forschungen auf diesem Gebiet haben vielversprechende Ergebnisse auf Basis von formaler Logik und künstlicher Intelligenz gebracht.² Allerdings tun sich auch neue Problemfelder auf, wie etwa die Frage, wer zur Modellierung von Recht berechtigt ist.³

[Rz 3] Als Vorteile von semantischen Systemen in der Legistik gegenüber der traditionellen schriftlichen Übermittlung von Regelungsinhalten können folgende Punkte gesehen werden:⁴

- Höhere Qualität des Rechtstextes an sich durch das ermöglichte schnellere Aufdecken von

¹ Vgl. z.B. MUHR, RUDOLF (2012): Zur Bürgerfreundlichkeit und Verständlichkeit alltagsnaher österreichischer Rechtstexte. In: Moraldo, Sandro M. (Hrsg.): Sprachenpolitik und Rechtssprache — Methodische Ansätze und Einzelanalysen, S. 117—140.

² Vgl. z.B. KOWALSKI, ROBERT A. (1995): Legislation as logic programs. In: Informatics and the Foundations of Legal Reasoning, Springer, S. 325—356. Oder auch BOURCIER, DANIELE (1995): La décision artificielle: le droit, la machine et l'humain. Presse universitaires de France. Oder auch RISSLAND, EDWINA ET AL. (2003): AI and Law — a fruitful synergy. In: Artificial Intelligence 150(1).

³ Vgl. LAURITSEN, MARC (2013): Are we free to code the law? In: Communications of the ACM 56(8), S. 60—66.

⁴ Vgl. HÖLLER, JOHANN/IPSMILLER, DORIS (2009): Semantik — was wir immer schon wussten: Semantische Technologien in der Unterstützung von Rechtsetzungsprozessen. In: Schweighofer, Erich (Hrsg): Semantisches Web und Soziale Netzwerke im Recht: Tagungsband des 12. Internationalen Rechtsinformatik Symposions, S. 144ff.

Unvollständigkeiten und inneren Widersprüchen.

- Verbesserter Entwicklungsprozess des Rechtstextes in Hinsicht auf ein gemeinsames Verständnis und gemeinsame Begrifflichkeiten. Unterschiedliche Wissensstände der Beteiligten können durch gezielte Informationen ausgeglichen werden.
- Verbesserte Kommunikation. Worte haben häufig unterschiedliche Bedeutungen, was zu Missverständnissen und schlecht verständlichen Texten führen kann. Semantische Modelle hingegen erheben den Anspruch der Eindeutigkeit.
- Im Fall von Änderungen in den Rechtsnormen wird das Durchführen einer Wirkungsanalyse wesentlich vereinfacht und verbessert.
- Die Qualität von eventuellen auf Grund des Rechtstextes implementierten IT-Systemen wird durch eine direktere Informationsübermittlung verbessert.

[Rz 4] Auf der Grundlage der bisher in diesem Gebiet durchgeführten Forschungen versucht dieses Paper nun, die trotz der bisherigen bereits beeindruckenden Ergebnisse bestehende Lücke zwischen den traditionellen Ansätzen der Softwareentwicklung und jenen der semantischen Modellierung zu schließen. Hierzu wird ein Framework vorgestellt, das aus einem semantischen Grundmodell und einem zu dessen Anwendung geeigneten Vorgehensmodell besteht.

[Rz 5] Der Anspruch der vollständigen Automatisierung der Erzeugung von Rechtstexten, wie sie beispielsweise in Winkels & Den Haan⁵ beschrieben ist, soll hierbei nicht erfüllt werden. Vielmehr geht es darum, den Erstellern von Modellen nach dem definierten Standard ein Werkzeug in die Hand zu geben, mit dem sich die Vorteile von Rechtstexten mit «state-of-the-art»-Technologien der Softwareentwicklung und den Fortschritten von semantischen Modellen kombinieren lassen.

II. Verwandte Forschung

[Rz 6] Als ein Vorreiter von IT-Systemen, die unmittelbar der Umsetzung von gesetzlichen Bestimmungen dienen, kann in Österreich der Pendlerrechner des Bundesministeriums für Finanzen gesehen werden. Dieser dient gem. § 3 Abs. 1 der Pendlerverordnung (PV) der «Ermittlung der Entfernung zwischen Wohnung und Arbeitsstätte ... und für die Beurteilung, ob die Benützung eines Massenbeförderungsmittels zumutbar oder unzumutbar ist». Das Ergebnis der Berechnungen des Pendlerrechners gilt als amtlicher Vordruck im Sinne des § 16 Abs. 1 Z 6 lit. g des Einkommenssteuergesetzes (EStG) 1988. Somit wird auf in Österreich bislang einzigartige Weise die Erfüllung von Kriterien einer Verordnung ausschließlich mittels eines IT-Systems überprüft.

[Rz 7] Derartige Szenarien werden wohl in Zukunft weiter zunehmen, schon längere Zeit werden ihnen eine stark steigende Bedeutung zugemessen: «Der ... Strukturierung von Normen in XML und ihrer Kontextualisierung in RDF/OWL wird die Zukunft, soweit sie überhaupt absehbar ist, gehören»⁶. In Umfeldern mit hohem Grad an IT-Integration bräuchte eine semantische Umsetzung

⁵ Vgl. WINKELS, RADBOUD/DEN HAAN, NIENKE(1995): Automated legislative drafting — Generating paraphrases of legislation. In: Proceedings of the 5th international conference on Artificial intelligence and law, ACM, S. 112—118.

⁶ SCHEFBECK, GÜNTHER(2009): Per Anhalter durch das Legiversum — Rechts- und Legislativinformatik 2.0. In: Schweighofer, Erich (Hrsg): Semantisches Web und Soziale Netzwerke im Recht: Tagungsband des 12. Internationalen Rechtsinformatik Symposions, S. 55.

nachvollziehbarerweise den höchsten Zusatznutzen.

[Rz 8] Ontologien für legistische Anwendungsgebiete wurden bislang eher mit dem Anspruch entwickelt, dass sie für jede denkbare Anwendung in jedem denkbaren Rechtsgebiet passen sollen. Dies führt zu einem hohen Grad der Abstraktion, der wiederum häufig darin resultiert, dass die Ontologien nicht immer direkt einsetzbar sind. Eine der am weitesten verbreiteten Sprachen dieser sehr wertvollen Beiträge ist LKIF, das «Legal Knowledge Interchange Format», welches vom EU-weit agierenden Estrella Konsortium («European project for Standardized Transparent Representations in order to Extend Legal Accessibility») entwickelt wurde⁷.

[Rz 9] LKIF erlaubt es, rechtliches Wissen in einer OWL-basierten Ontologie abzubilden. Dabei werden unter anderem spezifische Terminologie, Regeln und normative Aussagen berücksichtigt. Die Regeln in LKIF erweitern den «Semantic Web Rule Language» Standard (SWRL)⁸, der selbst wiederum eine Kombination aus OWL und RuleML⁹ darstellt. Das später in diesem Beitrag vorgestellte semantische Grundmodell versucht, mittels Mappings einen engen Bezug zu LKIF beizubehalten, um so Kompatibilität zu internationalen Ansätzen herzustellen.

[Rz 10] Von diesen Frameworks der Rechtsinformatik abgesehen, können zwei grundlegende Kategorien von IT-Systemen im juristischen Zusammenhang unterschieden werden¹⁰: Zum einen das Abspeichern und spätere Abrufen von rechtlicher Information und zum anderen Rechtsanalysensysteme. Versuchte man eine Einordnung des hier vorgestellten Ansatzes, fiel er wohl eher unter die erste Kategorie.

III. Semantisches Grundmodell und Vorgehensmodell

[Rz 11] Das Hauptproblem bei vielen heute verwendeten semantischen Modellen zur Erstellung und Pflege von Rechtstexten liegt darin, dass sie häufig entweder sehr abstrakt oder andernfalls nicht gut wiederverwendbar sind. Wie diese Lücke durch unser integriertes Modell geschlossen wird, soll in diesem Abschnitt dargestellt werden, Abbildung 1 zeigt einen Überblick. Während die direkten Herangehensweisen von semantischen Modellierungswerkzeugen und Softwareentwicklungsmethoden für sich genommen zu einer Reihe von Unzulänglichkeiten führen, erreicht das integrierte Modell die gewünschten Ziele mit wesentlich geringerem Aufwand. Um die Vorteile dieser Strategie näher aufzuzeigen, wird im nächsten Abschnitt eine umfangreiche Fallstudie vorgestellt.

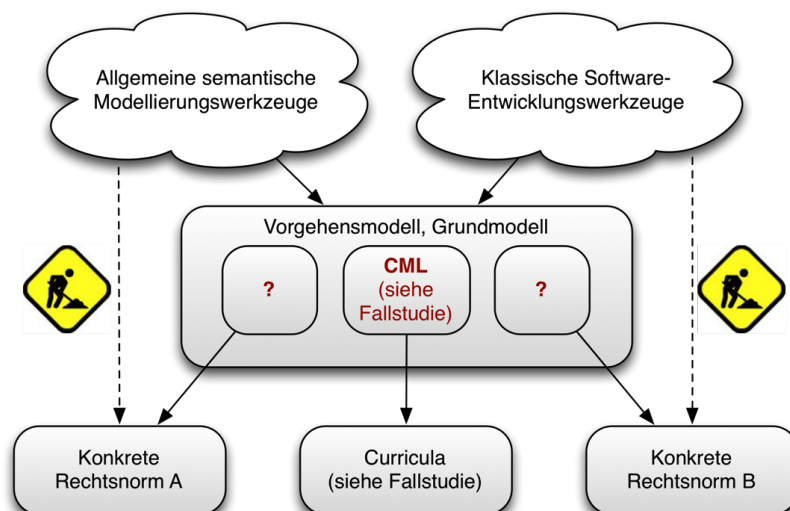
Abbildung 1: Forschungslücke

⁷ Vgl. <http://www.estrellaproject.org>.

⁸ Vgl. HORROCKS, IAN ET AL.(2004): SWRL: A Semantic Web Rule Language — Combining OWL and RuleML, <http://www.w3.org/Submission/SWRL>.

⁹ Vgl. BOLEY, HAROLD ET AL.(2001): Design Rationale of RuleML — A Markup Language for Semantic Web rules. In: Proceedings of the Semantic Web Working Symposium, S. 381—401.

¹⁰ Vgl. z.B.: MCCARTY, L. THORNE/SRIDHARAN, N.S.(1982): A computational theory of legal argument. Oder auch BING, JON(1984): Handbook of legal information retrieval.



1. Semantisches Grundmodell

[Rz 12] Das semantische Grundmodell stellt den Kern der Repräsentation von Gesetzestexten nach dem in diesem Paper vorgestellten Ansatz dar. Es wurde als Sammlung von allgemein verwendbaren Klassen in einer Ontologie konzipiert, die wiederum mit Ontologien höherer Ordnung verknüpft werden können. Das zugrunde liegende Konzept wurde von LKIF inspiriert, siehe hierzu Abschnitt II. Das Modell ist weitestgehend wiederverwendbar und verallgemeinerbar, insofern, als die darin enthaltenen Basiskomponenten für die Modellierung von Gesetzestexten aller Art anwendbar sind und dann je nach Anwendungs- bzw. Rechtsgebiet erweitert werden können.

[Rz 13] Diese fachspezifischen Ergänzungen der Terminologie und Ontologie sind je nach Rechtsgebiet unterschiedlich umfangreich, können aber durchaus den größten Teil des Endresultats ausmachen. Dies ergibt sich durch die Unterschiedlichkeit der verschiedenen in Frage kommenden Rechtsgebiete und bedingt auch einen recht unterschiedlich hohen Aufwand zu deren Implementierung. Je strukturierter und isolierter eine bestimmte Rechtsnorm ist, umso einfacher ist sie in das Grundmodell zu integrieren.

[Rz 14] Das Modell enthält sowohl die inhaltlichen Regelungen der Rechtsnorm, als auch den Rechtstext an sich in seiner konzeptuellen Struktur. Daher ist es nicht nur möglich, das in der Ontologie modellierte Wissen der Rechtsnorm in verschiedensten Anwendungsgebieten zu nutzen, sondern auch, den korrekten und vollständigen Rechtstext in für Menschen lesbarer Form zu erzeugen bzw. wiederherzustellen.

[Rz 15] Des Weiteren erlaubt das Modell Verknüpfungen zu externen Wissensbasen in Form von Ontologien bzw. Frameworks über passende semantische Beziehungen. Diese folgen den Prinzipien von Linked Open Data, wie sie beispielsweise in Bizer et al. beschrieben werden¹¹. Bereits im Grundmodell kann LKIF wahlweise als Basisframework eingebunden werden, die entspre-

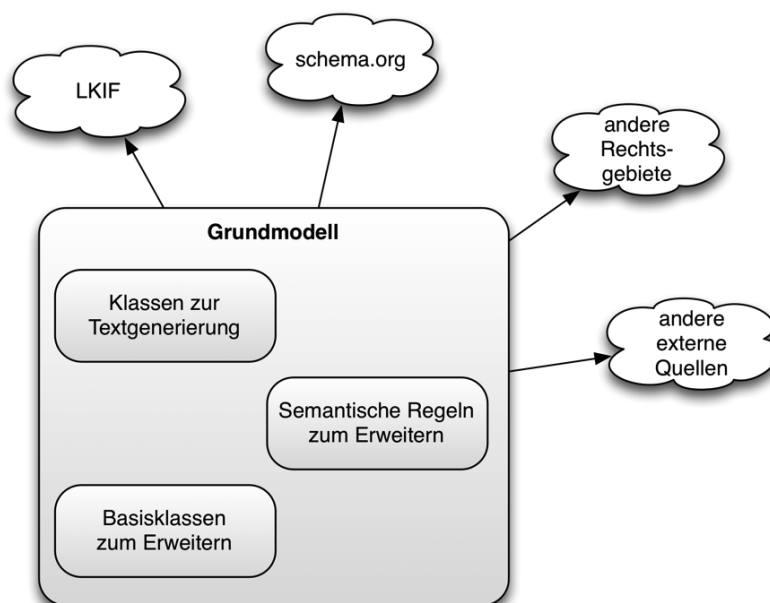
¹¹ Vgl. BIZER, CHRISTIAN ET AL.(2008): Linked data on the web. In: Proceedings of the 17th international conference on World Wide Web, ACM, S. 1265—1266.

chenden Verknüpfungen werden als Extension bereit gestellt. Auch hier gilt, dass das Mapping einzelner Komponenten fallspezifisch durchgeführt werden muss, da sich die entsprechenden Zuordnungen stark unterscheiden und kaum im Vorfeld für unterschiedliche Rechtsgebiete definiert werden können.

[Rz 16] Eine äußerst weitreichende Frage war jene, welche Variante bzw. Sprachebene von OWL¹² zum Einsatz kommen soll. Die Entscheidung fiel zu Gunsten von OWL DL, das einen vernünftigen Kompromiss aus Ausdrucksstärke und Entscheidbarkeit bietet. Da zukünftige Implementierungen des Grundmodells möglicherweise nicht alle Ausdrücke und Konstrukte einsetzen möchten, die bspw. OWL 2 Full bietet, scheint es eine gute Idee zu sein, das Grundmodell selbst auf eine bestimmte Grundmenge an Ausdrücken einzuschränken und dafür garantierte Entscheidbarkeit und die Verfügbarkeit von produktiv einsetzbaren Reasoning-Algorithmen anzubieten.

[Rz 17] Abbildung 2 zeigt schematisch den Aufbau des Grundmodells.

Abbildung 2: Semantisches Grundmodell



2. Automatische Generierung von Rechtstext

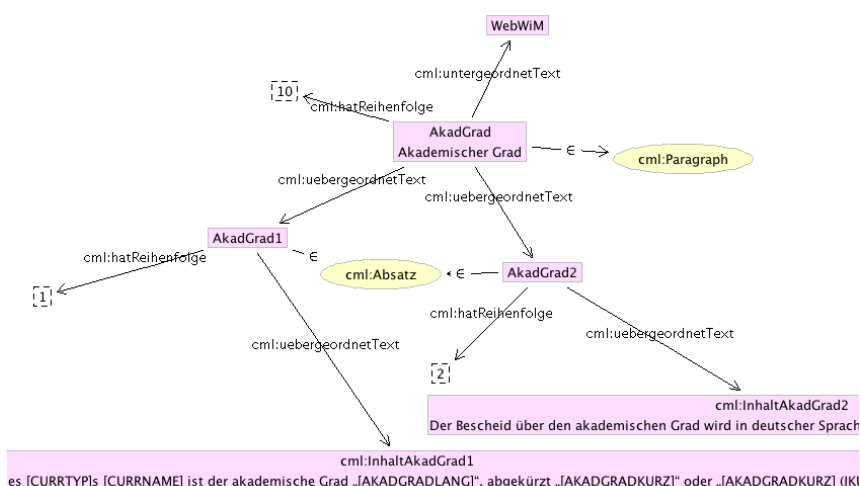
[Rz 18] Die automatische Generierung von semantisch korrektem Rechtstext wird hauptsächlich durch eine spezielle semantische Struktur erreicht, die als Input für Algorithmen in der Programmiersprache Java dient. Diese Algorithmen konstruieren dann eine menschenlesbare Version der Ontologie in der Form eines klassischen Rechtstextes. Da die zugrunde liegende konzeptuelle Struktur des konkreten Rechtsgebiets Teil der modellierten Ontologie ist, kann sie verwendet werden, um die Struktur und Hierarchie des Rechtstextes zu erzeugen.

¹² Vgl. HITZLER, PASCAL ET AL.(2012): OWL 2 Web Ontology Language Primer (Second Edition), <http://www.w3.org/TR/owl2-primer>.

[Rz 19] Um dieses Ziel zu erreichen, wird im Grundmodell ein Satz von Basisklassen angeboten, die für eine große Bandbreite von Rechtsgebieten eingesetzt werden können. Im Wesentlichen bilden diese Klassen die grundlegenden Bausteine von Rechtstexten ab, wie beispielsweise Paragraphen, Absätze, Ziffern, Litera etc. und deren hierarchische Beziehungen zueinander. Gemeinsam mit einer zusätzlichen Helferklasse namens *Inhalt* sind sie alle Subklassen der Klasse *Textbaustein*, die die essentiellen Grundfunktionalitäten für Hierarchisierung, Ordnung und Organisation von Rechtstext bereit stellt. Diese Funktionalität wird mittels integrierter Metainformationen und Beziehungen zu anderen *Textbausteinen* modelliert.

[Rz 20] Das folgende Beispiel illustriert die Generierung von Rechtstext anhand eines ausgewählten Paragraphen eines Mastercurriculums an der Johannes Kepler Universität Linz (JKU). Abbildung 3 zeigt die Struktur dieses Paragraphen in der Ontologie, die unter Anwendung verhältnismäßig einfacher Java-Algorithmen dann in lesbaren Rechtstext umgewandelt werden kann. Diese erzeugen die Hierarchie von Paragraphen, Absätzen, etc. auf rekursive Art und Weise und platzieren die korrekten Objekte und Beziehungen an den entsprechenden Positionen.

Abbildung 3: Automatische Textgenerierung



[Rz 21] Die Hauptalgorithmen für die Textgenerierung innerhalb der Klassen des Rechtstextes selbst und in allen Varianten von *Textbaustein* enthalten spezifische Formatierungsanweisungen und eine Rekursion ähnlich der folgenden:

```
StringBuffer sb = new StringBuffer(); Collections.sort(textbausteine); for(Textbaustein b : textbausteine) { sb.append(b.toString()); }
```

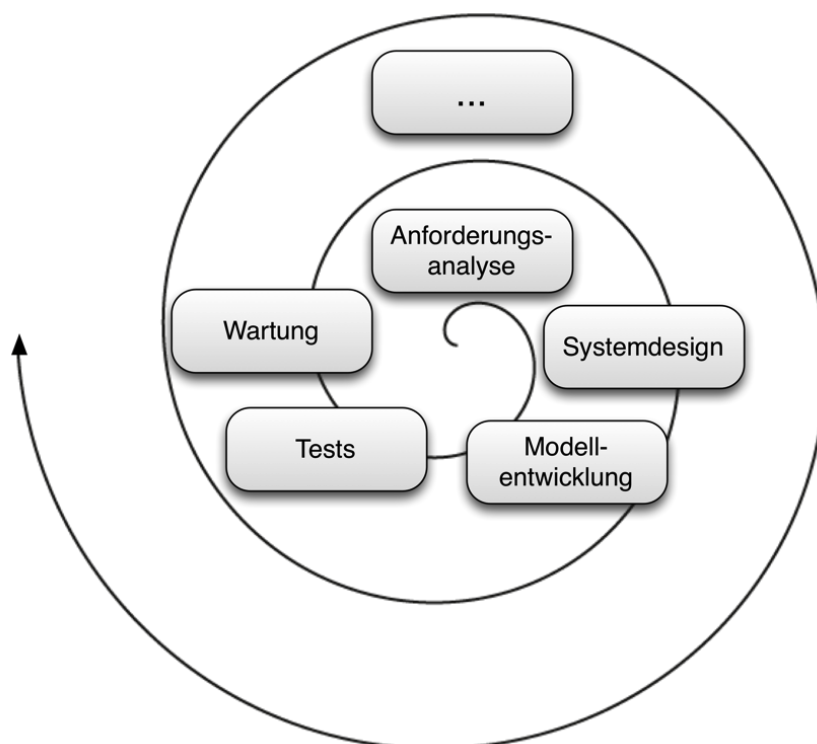
[Rz 22] Im nächsten Schritt wird ein Vorgehensmodell beschrieben, das die korrekte Funktionsweise aller Algorithmen, semantischen Regeln und Konsistenzprüfungen sicherstellt und für Kompatibilität der erstellten Modelle mit dem Grundmodell sorgt.

3. Vorgehensmodell

[Rz 23] Das in diesem Abschnitt beschriebene Vorgehensmodell konzentriert sich darauf, den Modellierer durch die Entwicklung eines semantischen Modells eines konkreten Anwendungs-

gebiets in Form eines bestimmten Rechtstextes zu führen. Es lässt hierbei Erfahrungen aus der Softwareentwicklung einfließen, indem es auf dem vielfach bewährten Spiralmodell¹³ aufbaut. Dessen großer Vorteil in diesem Kontext liegt darin, dass es die Gegebenheit von sich regelmäßig ändernden Rechtsvorschriften in einem schrittweisen Ansatz abdeckt. Jedes Mal, wenn die Schritte des Modells erneut von vorne abgearbeitet werden, startet eine neue Iteration, die auf den Ergebnissen des vorherigen Durchgangs aufbaut und diese den aktuellen Gegebenheiten anpasst.

Abbildung 4: Vorgehensmodell



[Rz 24] Abbildung 4 zeigt das Vorgehensmodell, das aus den folgenden Schritten besteht:

- Anforderungsanalyse Im ersten Schritt müssen die Bedürfnisse und Anforderungen aller am Rechtssetzungsprozess Beteiligten erhoben und analysiert werden. Häufig sind mehrere Benutzergruppen mit unterschiedlichem technischen und juristischen Wissensstand involviert. Zur Strukturierung kann etwa folgende Vorgehensweise eingesetzt werden¹⁴:
 - Anforderungserhebung
 - Anforderungsanalyse
 - Anforderungsspezifikation
 - Anforderungsbewertung

¹³ Vgl. BOEHM, BARRY W.(1988): A spiral model of software development and enhancement. In: IEEE Computer 21(5), S. 61–72.

¹⁴ Vgl. BOURQUE, PIERRE/FAIRLEY, RICHARD E.(2014): SWEBOOK 3.0 — Guide to the Software Engineering Body of Knowledge, IEEE Computer Society.

Was die Beteiligten betrifft, so sind jedenfalls Verfasser und Empfänger der Rechtsnorm zu berücksichtigen, in vielen Anwendungsgebieten wird es jedoch noch weitere mit einzubeziehende Stakeholder geben. Weiters sind Anforderungen hinsichtlich rechtlicher Voraussetzungen zu untersuchen, zudem können spezielle Bedürfnisse wie etwa Mehrsprachigkeit, Versionierung, Barrierefreiheit oder ähnliche bestehen.

- Systemdesign Der nächste Schritt beinhaltet das Design des Gesamtsystems und dient insbesondere der Komplexitätsreduktion und Minimierung von Risiken hinsichtlich Fehlentwicklungen. Beispielsweise sind hier folgende Fragen zu beantworten:
 - Soll das gesamte Rechtsgebiet oder nur bestimmte Teile modelliert werden?
 - Wie soll die Struktur des Modells aussehen, wie werden einzelne Teilbereiche der Rechtsnorm abgebildet? Gibt es Beziehungen zu anderen Rechtsnormen und wie sollen sie ggf. modelliert werden?
 - Gibt es externe Quellen wie etwa bereits modellierte Rechtsnormen, die wiederverwendet werden können?
 - Welche semantische Sprachebene wird benutzt, welche Regelsprache, welche Programmiersprache? Welche Frameworks können verwendet werden?
 - Braucht es eine oder ev. mehrere Benutzeroberflächen zur Unterstützung der Anwender?
- Modellentwicklung Hier erfolgt die eigentliche Implementierung des semantischen Modells sowie eventueller Regelwerke und Benutzeroberflächen. Auf Basis des zuvor entwickelten Systemdesigns werden die ermittelten Anforderungen programmatisch umgesetzt. In dieser Phase sind die Unterschiede zwischen verschiedenen Rechtsgebieten am höchsten, dies liegt an höchst unterschiedlichen rechtlichen Voraussetzungen, externen Beziehungen und Strukturen. Nichtsdestotrotz versucht das Grundmodell, die unterschiedlichen Implementierungen zueinander kompatibel zu halten, indem gewisse Grundrichtlinien eingehalten werden müssen.
- Tests Als ein erster Test kann die Konkretisierung des Modells für eine spezifische Anwendung gesehen werden. Eine bestimmte Rechtsnorm wird mit Hilfe des eben entwickelten semantischen Modells abgebildet. Gelingt dies, stellt es ein Indiz für eine erfolgreiche Implementierung dar. Dies wird beispielsweise in der Fallstudie im nächsten Abschnitt gezeigt. Jedenfalls ist es sinnvoll, noch weitere Tests durchzuführen, hierzu gibt es je nach konkreter Umsetzung in der Literatur ausreichend erprobte Möglichkeiten.
- Wartung Unter Wartung fällt «die Veränderung eines Softwareprodukts nach dessen Auslieferung, um Fehler zu beheben, Performanz oder andere Attribute zu verbessern oder Anpassungen an die veränderte Umgebung vorzunehmen»¹⁵. Im vorliegenden Fall ist also zu unterscheiden zwischen:
 - Wartung zum Zweck der Verbesserung der Software, wenn etwa Fehler im Modell ausgebessert oder auch Algorithmen und Regeln korrigiert bzw. neue Funktionalitäten eingeführt werden.
 - Wartung zum Zweck der Anpassung an externe Änderungen. Manche Rechtsnormen werden mehr oder weniger regelmäßig angepasst. Wenn das entsprechende Modell mit diesen Änderungen umgehen kann, verlängert dies seine Lebenszeit erheblich.

¹⁵ IEEE Standard Glossary of Software Engineering Technology 610.12-1990, S. 46.

IV. Fallstudie: Curriculumserstellung

[Rz 25] Als ein Beispiel für das eben beschriebene semantische Grundmodell wurde eine konkrete Umsetzung entwickelt. Diese Implementierung fungiert auch als «Proof-of-Concept» für das Grund- und das Vorgehensmodell. Sie zeigt, dass es möglich ist, vollständige Rechtsnormen mit einer Integration der zuvor gezeigten Techniken semantisch zu modellieren und dann wieder lesbaren Rechtstext zu erzeugen.

1. Umwelt und Setting

[Rz 26] Das Universitätswesen im europäischen Raum und so auch in Österreich durchläuft seit Beginn der 1990er große Veränderungen. Mit dem Universitätsgesetz 2002 wurden die österreichischen Universitäten vollrechtsfähig, dies brachte unter anderem das Erfordernis zur Erlassung einer universitären Satzung mit sich. IT-Systeme für Curricula müssen so auch diesen Rechtsnormen entsprechen. Weiters gab es umfangreiche Änderungen im Studienrecht, beispielsweise wurden die Weichen in Richtung Bologna-Architektur gestellt, vornehmlich also ein dreistufiges System aus Bachelor-, Master- und Doktoratsstudien. Auch wurde ein einheitliches Leistungspunktesystem eingeführt, das European Credit Transfer System (ECTS)¹⁶.

[Rz 27] Die Veränderungen der letzten Jahre haben also einerseits zu Vereinheitlichungen wie den ECTS-Punkten geführt, andererseits gibt es zwischen den einzelnen Bildungseinrichtungen nach wie vor große Unterschiede, was die studienrechtliche Regelungen in den jeweiligen Satzungen betrifft. Das in dieser Fallstudie implementierte Modell geht insbesondere auf die Gegebenheiten an der Johannes Kepler Universität Linz ein, versucht jedoch auch, anderen österreichischen Universitäten und Bildungseinrichtungen die Anpassung an ihre eigenen Bedürfnisse möglichst einfach zu gestalten.

2. Implementierung des Grundmodells

[Rz 28] Als ein Proof-of-Concept des Grundmodells wurde eine Reihe von Ontologien implementiert. Sie wurden mit Curriculum Modeling Language (CML) benannt und erlauben die Modellierung von Curricula nach österreichischem Recht.

2.1. Ontologien

[Rz 29] Tabelle 1 bietet einen Überblick über die in CML modellierten Ontologien:

Ontologie	Inhalt
cml.owl	Grundstruktur, bindet die anderen Ontologien ein.
cml_uni.owl	Modelliert Universitäten und andere Bildungseinrichtungen.
cml_curr.owl	Modelliert ein Curriculum; stellt sicher, dass alle nötigen Bestandteile vorhanden sind.

¹⁶ Vgl. ECTS-Leitfaden (2009), http://ec.europa.eu/education/tools/docs/ects-guide_de.pdf.

cml_templates.owl	Musterdaten, Vorlagen für verschiedene Arten von Curricula, Vorschläge für Rechtstext, englische Übersetzungen.
cml_rules.owl	Enthält die in SWRL modellierten Regeln zur automatisierten Prüfung von Curricula.
cml_extended.owl	Stellt die Verbindung zu LKIF her, Mapping von externen Konzepten.

Tabelle 1: CML Ontologien

2.2. Semantisches Regelwerk

[Rz 30] CML macht starken Gebrauch von SWRL. Drei grundlegende Arten von Regeln können zur Validierung eines modellierten Curriculums unterschieden werden:

- Bedingungen, die erfüllt sein müssen, damit ein Curriculum als gültig und korrekt bezeichnet werden kann. Ein Bachelor-Curriculum an der JKU muss beispielsweise mindestens eine Lehrveranstaltung beinhalten, «die Inhalte der Frauen- und Geschlechterforschung abdecken; dies sind mindestens a) eine Pflichtlehrveranstaltung im Umfang von 3 ECTS in Bachelor- und Diplomstudien.»¹⁷ Dies kann mit folgender SWRL-Regel überprüft werden, die korrekte Curricula als *CurriculumMitGenderLV* kategorisiert:

$$\text{Curriculum}(? curr) \wedge \text{LV}(? lv) \wedge \text{indirektUntergeordnet}(? lv, ? curr) \wedge \text{hatGenderbezug}(? lv, true) \Rightarrow \text{CurriculumMitGenderLV}(? curr)$$

- Regeln, die die Aufmerksamkeit der für die Prüfung des Curriculums verantwortlichen Person auf eine Irregularität lenkt, deren Erfüllung jedoch keine unbedingte Voraussetzung für ein korrektes Curriculum ist. Beispielsweise umfassen Bachelor-Curricula üblicherweise 180 ECTS, von diesem Wert kann jedoch in begründeten Fällen abgewichen werden¹⁸. Die folgende Regel überprüft, ob der Standardwert der Kategorie des Curriculums eingehalten wird, andernfalls wird das Curriculum als *CurriculumSpezialECTS* kategorisiert.

$$\text{Curriculum}(? curr) \wedge \text{hatCurriculumTyp}(? curr, ? typ) \wedge \text{ects}(? curr, ? ects) \wedge \text{ects}(? typ, ? standard) \wedge \text{swrlb: notEqual}(? ects, ? standard) \Rightarrow \text{CurriculumSpezialECTS}(? curr)$$

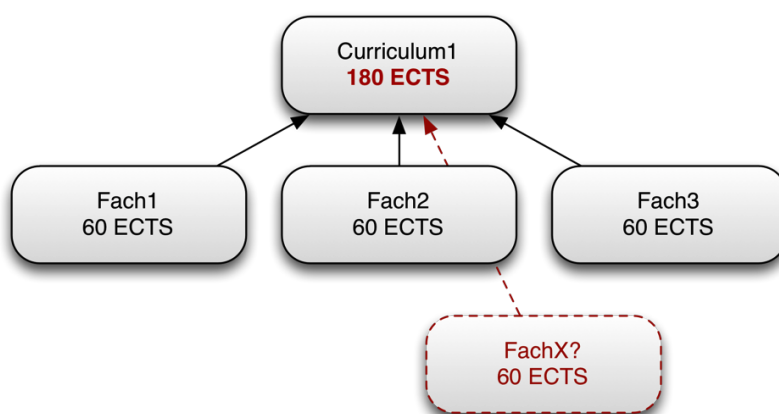
- Die dritte Kategorie beinhaltet jene Regeln, die nicht oder nur mit hohem Aufwand mit semantischen Technologien umgesetzt werden können. In vielen Fällen liegt dies an der in OWL gültigen Open World Assumption (OWA). Deren Grundidee ist es, dass, wenn ein bestimmtes Stück Wissen nicht von der Menge an bestehendem Wissen abgeleitet werden kann, nichts daraus geschlossen werden kann. Somit kann zwischen nicht vorhandenem und negativem

¹⁷ § 19 Abs. 1 Z 11 der Satzung der JKU, Satzungsteil Studienrecht.

¹⁸ Vgl. § 54 Abs. 3 des Universitätsgesetzes (UnivG).

Wissen unterschieden werden¹⁹. Viele Regeln dieser Kategorie sind summenbasiert, so wie das folgende Beispiel: Es soll festgestellt werden, ob die Summe der ECTS der Fächer eines Curriculums den Gesamt-ECTS des Curriculums entspricht. Abbildung 5 zeigt die Problematik: In OWA kann man nicht wissen, ob die momentan untergeordneten Fächer tatsächlich die einzigen sind. Es könnte immer ein weiteres Fach auftauchen, das bisher nicht berücksichtigt wurde.

Abbildung 5: Problematik OWA



Eine Möglichkeit, dieses Problem zu lösen, ist die Einführung von sogenannten Closure Axiomen, die eine künstliche Closed World Umgebung erzeugen. Diese Vorgehensweise erweitert die Ontologie um eine sehr große Anzahl von Axiomen, zudem muss jede Änderung an mehreren Stellen passieren. Eine weitere Möglichkeit wäre der Einsatz von SQWRL (Semantic Query-Enhanced Web Rule Language), welches zeitpunktbezogene Abfragen in einer Ontologie bietet, wie sie in einem Closed World System möglich wären²⁰. Im folgenden Beispiel wird SQWRL eingesetzt, um eine Summe der ECTS der Fächer zu bilden und diese mit dem Curriculum zu vergleichen:

$$\begin{aligned} & Curriculum(?curr) \wedge Baustein(?baustein) \wedge ects(?curr, ?currents) \wedge \\ & untergeordnet(?baustein, ?curr) \wedge Baustein(?baustein) \wedge \\ & ects(?baustein, ?ects) \wedge \mathbf{sqwrl:sum}(?sum, ?ects) \wedge \\ & \mathbf{swrlb:notEqual}(?sum, ?currents) \Rightarrow CurriculumFalschECTS(?curr) \end{aligned}$$

Da SQWRL noch nicht standardisiert ist und von vielen Frameworks bzw. Reasonern nicht voll unterstützt wird, verwendet unser Ansatz Java-Algorithmen, um die gewünschten Überprüfungen durchzuführen.

¹⁹ Vgl. GRIMM, STEPHAN(2010): Knowledge Representation and Ontologies. In: Scientific Data Mining and Knowledge Discovery — Principles and Foundations, Springer, S. 210ff.

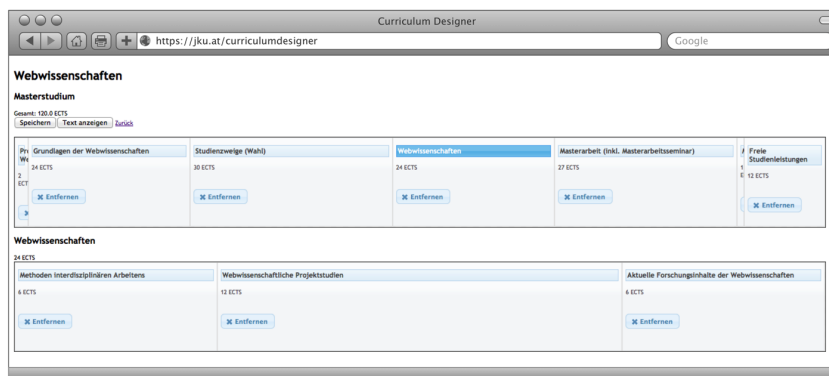
²⁰ Vgl. O'CONNOR, MARTIN J./DAS, AMAR K.(2009): SQWRL: A Query Language for OWL. In: OWLED'09.

2.3. Grafische Benutzeroberflächen

[Rz 31] Obwohl es möglich ist, eine Ontologie konform zu CML mit generischen Werkzeugen wie Protégé²¹ zu erstellen und zu bearbeiten, stellt deren Komplexität und große Funktionsvielfalt viele am Rechtssetzungsprozess Beteiligte vor große Herausforderungen. Eine speziell für deren Aufgaben implementierte grafische Benutzeroberfläche kann hier Abhilfe schaffen. Im Idealfall sind die Bedürfnisse aller Beteiligten (vgl. Abschnitt III 3.) zu berücksichtigen und in entsprechenden Benutzeroberflächen umzusetzen.

[Rz 32] Exemplarisch wurde im Zuge der Fallstudie eine Benutzeroberfläche umgesetzt. Sie bietet Studienkommissionen, die mit der Erstellung eines neuen Curriculums betraut wurden, eine einfache Weboberfläche, mit der die groben Blöcke eines Curriculums auf intuitive Art und Weise zusammengesetzt werden können und dieses Grobgerüst dann in einer zu CML kompatiblen Ontologie abgespeichert werden kann. Dieser sogenannte «Curriculum Designer» ist in Java und JSP geschrieben, Abbildung 6 zeigt einen Screenshot.

Abbildung 6: Curriculum Designer



3. Ergebnisse

[Rz 33] Die Ergebnisse der Fallstudie sind äußerst vielversprechend. Exemplarisch wurden mehrere Curricula in semantische Modelle umgesetzt und daraus wieder menschenlesbarer Rechtstext generiert. Die Basisstruktur mit den wichtigsten Informationen eines typischen österreichischen Masterstudiums mit etwa 300 bis 400 Fächern und Lehrveranstaltungen ergibt eine Ontologie mit etwa 4000 bis 5000 Axiomen. Ontologien dieser Größenordnung können von den beschriebenen Algorithmen und Frameworks problemlos abgearbeitet werden. Verständlicherweise erhöht sich die Zahl der Axiome, wenn mehr Informationen über die einzelnen Lehrveranstaltungen hinzugefügt werden oder komplexere Regelwerke implementiert werden. Dieser Größenanstieg ist jedoch moderat und erfolgt linear, sodass sich in den Tests keinerlei Performanceprobleme gezeigt haben.

²¹ Vgl. <http://protege.stanford.edu>.

V. Fazit

[Rz 34] In diesem Beitrag wurde ein Ansatz zur Integration von semantischen Modellen und traditionellem Software Engineering zur Modellierung von Rechtstexten vorgestellt. Dieser erlaubt es, die Lücke zwischen sehr abstrakten und generischen Repräsentationen zum einen und sehr spezifischen Implementierungen mit relativ geringen Möglichkeiten zur Wiederverwendung zum anderen zu schließen. In der Fallstudie wurde die Umsetzbarkeit des Grundmodells und des Vorgehensmodells in einem verhältnismäßig komplexen Umfeld gezeigt, welches Verfassungsrecht, Gesetze, statutarisches Recht und universitäre Regulationen vereint. Somit zeigt dieses Experiment auch die Fähigkeit unseres Ansatzes, mit verschiedenen Hierarchien von legislativem und statutarischem Recht umzugehen.

[Rz 35] Zusammengefasst lässt sich sagen, dass die beschriebene Methodik eine semantische Repräsentation von Rechtstext signifikant vereinfacht und die produktive Einsetzbarkeit erhöht.

MARTIN STABAUER / JOHANN HÖLLER, Institut für Datenverarbeitung in den Sozial- und Wirtschaftswissenschaften, Johannes Kepler Universität Linz.