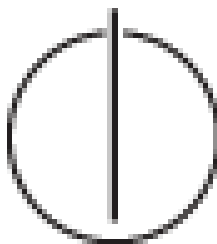


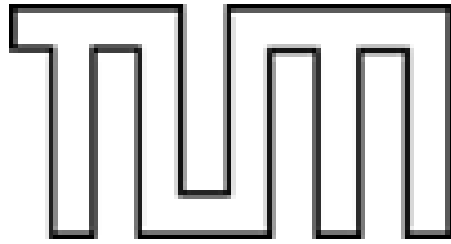
FAKULTÄT FÜR INFORMATIK
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Bachelorarbeit in Wirtschaftsinformatik

**Analyse und Gegenüberstellung
bestehender Wiki-Systeme
zur Pflege semi-strukturierter Informationen**

Martin Bolko Stange





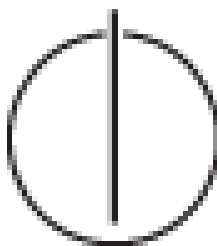
FAKULTÄT FÜR INFORMATIK
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Bachelorarbeit in Wirtschaftsinformatik

**Analyse und Gegenüberstellung
bestehender Wiki-Systeme
zur Pflege semi-strukturierter Informationen**

**Analysis and Comparison
of existing Wiki-Systems
for Maintaining semi-structured Information**

Bearbeiter:	Martin Bolko Stange
Aufgabesteller:	Prof. Dr. Florian Matthes
Betreuer:	Dipl.-Inf. Christian M. Schweda
Abgabedatum:	13. Februar 2009



Ich versichere, dass ich diese Bachelorarbeit selbständig verfasst
und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

13. Februar 2009

Martin Bolko Stange

Zusammenfassung

Semantische Wikis stehen seit 2005 im Fokus vieler Forschungs- und Entwicklungsaktivitäten. Sie versuchen die von Tim Bernes-Lee propagierte Vision des des semantischen Webs, also die Möglichkeit die Bedeutung einer Information beschreiben zu können, mit den Merkmalen der kollaborativen und einfachen Wissenssammlung und Inhaltsgenerierung von Wiki-Systemen zu verbinden. Hierbei gibt es vielfältige Verwendungsszenarien, technische Ansätze und Realisierungen. Im Rahmen dieser Arbeit erfolgt die Betrachtung von fünf sehr unterschiedlichen Ansätzen zur Realisierung eines semantischen Wikis. Dabei handelt es sich um das *Semantic MediaWiki (SMW)*, *NEPOMUK*, die *Hypertext Knowledge Workbench (HKW)*, *AceWiki* sowie das *DITF-MR Recherchesystem*. SMW ist eines der bekanntesten semantischen Wiki-Systeme und stellt eine Erweiterung zum MediaWiki dar, welches durch die Verwendung beim den Wikipedia-Projekten einen hohen Bekanntheitsgrad erlangt hat. Bei NEPOMUK und HKW handelt es sich vorrangig um sogenannte *Social Semantic Desktop* Systeme, welche dem Benutzer die Strukturierung von Information auf verschiedensten Abstraktionsniveaus erlauben. AceWiki verwendet zur Modellierung und Darstellung der Wiki-Inhalte die strukturierte natürliche Sprache *Attempto Controlled English*. Das DITF-MR-Recherchesystem wurde vorrangig zur Suche und Recherche in gespeicherten Webmaschineneinstellung entwickelt, wird mittlerweile auch anderweitig verwendet stellt ein streng strukturiertes Wiki dar. Die unterschiedlichen Ansätze beschreiben sehr gut das Spektrum der Varianten und Möglichkeiten ein semantisches Wiki-System zu erstellen. Aufbauend auf den Erkenntnissen und Erfahrungen aus der Evaluation der Wiki-Systeme werden diese anhand einzelner Merkmale verglichen und bewertet. Hieraus ergeben sich die Empfehlungen für die Weiterentwicklung eines am Lehrstuhl sebis eingesetzten Wiki-Systems.

Inhaltsverzeichnis

1 Einführung.....	1
1.1 Motivation.....	1
1.2 Aufgabenstellung.....	2
1.3 Aufbau der Arbeit.....	3
2 Verwandte Arbeiten – Semantische Wikis.....	4
2.1 Semantic MediaWiki (SMW).....	4
2.1.1 Steckbrief.....	4
2.1.2 Strukturierung.....	5
2.1.3 Abfragesprache.....	9
2.1.4 Export.....	11
2.1.5 Import	11
2.1.6 Berechtigungskonzept.....	11
2.1.7 Useability.....	12
2.2 NEPOMUK.....	13
2.2.1 Steckbrief.....	14
2.2.2 Strukturierung.....	14
2.2.3 Abfragesprache.....	21
2.2.4 Export.....	22
2.2.5 Import.....	23
2.2.6 Berechtigungskonzept.....	24
2.2.7 Useability.....	24
2.3 Hypertext Knowledge Workbench.....	25
2.3.1 Steckbrief.....	25
2.3.2 Strukturierung.....	25
2.3.3 Abfragesprache.....	31
2.3.4 Export.....	31
2.3.5 Import.....	31
2.3.6 Berechtigungskonzept.....	31
2.3.7 Useability.....	31
2.4 AceWiki.....	32
2.4.1 Steckbrief.....	32
2.4.2 Strukturierung.....	32
2.4.3 Abfragesprache.....	40
2.4.4 Export.....	41
2.4.5 Import.....	41
2.4.6 Berechtigungskonzept.....	41
2.4.7 Useability.....	41
2.5 DITF-MR-Recherchesystem.....	42
2.5.1 Steckbrief.....	42
2.5.2 Strukturierung.....	42
2.5.3 Abfragesprache.....	49
2.5.4 Export.....	55
2.5.5 Import.....	55
2.5.6 Berechtigungskonzept.....	56
2.5.7 Useability.....	57

3 Diskussion der Konzepte / Empfehlung.....	58
3.1 Modell / Struktur.....	58
3.1.1 Vorlagen und Typensystem.....	58
3.1.2 Kategorien und Tags.....	59
3.1.3 Beziehungskonzept.....	60
3.2 Benutzeroberfläche.....	61
3.2.1 Eingabearten.....	61
3.3 Suchen / Abfragen.....	63
3.3.1 Semantische Suche.....	63
3.3.2 Suchergebnisse in Wiki-Seiten einbetten.....	63
3.3.3 Merkliste	63
3.3.4 Synonyme.....	64
3.4 Weitere Funktionen.....	64
3.4.1 Maßeinheiten mit Umrechnung.....	64
3.4.2 Sichten.....	64
3.4.3 Bewertung von Inhalten.....	64
3.4.4 Export / Import.....	65
4 Zusammenfassung und Ausblick.....	66
4.1 Kritische Würdigung.....	66
4.2 Künftige Forschungsfragen.....	66
Quellen:.....	68

Abbildungsverzeichnis

Abbildung 1: SMW Kategorien.....	5
Abbildung 2: SMW Kategorie-Seite.....	5
Abbildung 3: SMW Datentyp VL_Umfang.....	6
Abbildung 4: SMW Attribut.....	7
Abbildung 5: SMW Template und Properties.....	8
Abbildung 6: SMW Klassendiagramm.....	9
Abbildung 7: SMW Online Services.....	9
Abbildung 8: SMW Suche.....	10
Abbildung 9: SMW Inline Query.....	11
Abbildung 10: NEPOMUK SWCM Datenmodell in Anlehnung an [Vö08b S.5 Fig. 3].....	15
Abbildung 11: NEPOMUK Beziehungshierarchie in Anlehnung an [Vö08a S.13 Figure 3]	16
Abbildung 12: NEPOMUK PSEW PIMO.....	17
Abbildung 13: NEPOMUK Vorlesung.....	17
Abbildung 14: PSEW Vorlesung bearbeiten I.....	18
Abbildung 15: PSEW Vorlesung bearbeiten II.....	18
Abbildung 16: PSEW Annotieren.....	18
Abbildung 17: PSEW Property hinzufügen.....	19
Abbildung 18: PSEW Property value.....	19
Abbildung 19: PSEW Beziehungstyp.....	19
Abbildung 20: PSEW Property Domain ändern.....	19
Abbildung 21: NEPOMUK Kalenderblatt.....	20
Abbildung 22: PSEW Instanz bearbeiten.....	20
Abbildung 23: PSEW Instanz bearbeiten II.....	20
Abbildung 24: PSEW inverse Beziehung I.....	21
Abbildung 25: PSEW inverse Beziehungen II.....	21
Abbildung 26: PSEW inverse Beziehungen III.....	21
Abbildung 27: PSEW semantische Abfrage.....	22
Abbildung 28: NEPOMUK Export.....	23
Abbildung 29: NEPOMUK Import.....	23
Abbildung 30: HKW CDS Datenmodell in Anlehnung an [Vö08c Fig.1]	26
Abbildung 31: HKW Layout und NameItem.....	27
Abbildung 32: HKW ContentItem.....	28
Abbildung 33: HKW ContentItem bearbeiten.....	28
Abbildung 34: HKW Statement anlegen.....	29
Abbildung 35: HKW Statement.....	29
Abbildung 36: HKW Relation.....	30
Abbildung 37: HKW NameItem Vorlesung.....	30
Abbildung 38: AceWiki create Noun.....	33
Abbildung 39: AceWiki create Proper Name.....	33
Abbildung 40: AceWiki Noun Article.....	33
Abbildung 41: AceWiki Noun References.....	33
Abbildung 42: AceWiki Noun Hierarchy.....	33
Abbildung 43: AceWiki Noun Individuals.....	34
Abbildung 44: AceWiki Proper Name Assignments.....	34
Abbildung 45: AceWiki create Transitive Adjective.....	34
Abbildung 46: AceWiki Adjectiv Article.....	34
Abbildung 47: AceWiki Adjective References.....	34
Abbildung 48: AceWiki Adjective Trans. Adjective.....	34

Abbildung 49: AceWiki create Verb.....	35
Abbildung 50: AceWiki create Of-Construct.....	35
Abbildung 51: AceWiki Verb Article.....	36
Abbildung 52: AceWiki Verb Verb.....	36
Abbildung 53: AceWiki Of-Construct.....	36
Abbildung 54: AceWiki Verb References.....	36
Abbildung 55: AceWiki Funktionswörter.....	36
Abbildung 56: AceWiki exportierbarer und nicht exportierbarer Satz.....	37
Abbildung 57: AceWiki Sentence Editor I.....	39
Abbildung 58: AceWiki Sentence Editor II.....	39
Abbildung 59: AceWiki Sentence Editor III.....	39
Abbildung 60: AceWiki Sentence Editor IV.....	39
Abbildung 61: AceWiki Sentence Editor V.....	39
Abbildung 62: AceWiki Sentence Editor VI.....	39
Abbildung 63: AceWiki Sentence Editor VII.....	40
Abbildung 64: AceWiki erstellter Satz.....	40
Abbildung 65: AceWiki Satzstruktur.....	40
Abbildung 66: AceWiki Satz-Ontologie.....	40
Abbildung 67: AceWiki Sentence Editor Frage.....	41
Abbildung 68: AceWiki Wiki-Seite mit Fragen.....	41
Abbildung 69: DITF-MR-RS Informationsobjekttyp anlegen.....	43
Abbildung 70: DITF-MR-RS Beziehungstyp Bezeichnungen.....	43
Abbildung 71: DITF-MR-RS weiterer Beziehungstyp	43
Abbildung 72: DITF-MR-RS Attributauswahl.....	44
Abbildung 73: DITF-MR-RS Attributbezeichnung.....	44
Abbildung 74: DITF-MR-RS Attributtyp einzeliges Textfeld.....	45
Abbildung 75: DITF-MR-RS Attributtyp Zeitpunkt.....	45
Abbildung 76: DITF-MR-RS Attributtyp Auswahlbox.....	46
Abbildung 77: DITF-MR-RS Attributtyp Relation.....	46
Abbildung 78: DITF-MR-RS Informationsobjekttyp.....	47
Abbildung 79: DITF-MR-RS Informationsobjekt anlegen.....	47
Abbildung 80: DITF-MR-RS Beziehung erstellen.....	48
Abbildung 81: DITF-MR-RS Rich Text Editor.....	48
Abbildung 82: DITF-MR-RS UML Klassendiagramm.....	49
Abbildung 83: DITF-MR-RS Attribut-Ansichten.....	50
Abbildung 84: DITF-MR-RS Attribut-Ansicht erstellen.....	50
Abbildung 85: DITF-MR-RS Standard-Attribut-Ansicht.....	51
Abbildung 86: DITF-MR-RS Attribut-Ansicht Tabelle.....	51
Abbildung 87: DITF-MR-RS Attribut-Ansicht Text.....	51
Abbildung 88: DITF-MR-RS Volltextsuche.....	52
Abbildung 89: DITF-MR-RS Recherche.....	52
Abbildung 90: DITF-MR-RS Recherche Attributauswahl.....	53
Abbildung 91: DITF-MR-RS Recherche Attributwert.....	53
Abbildung 92: DITF-MR-RS Recherche Muss-Kriterien.....	53
Abbildung 93: DITF-MR-RS Recherche Nicht spezifiziert-Kriterien.....	54
Abbildung 94: DITF-MR-RS Recherche Übersicht Filterkriterien.....	54
Abbildung 95: DITF-MR-RS Rechercheergebnis.....	54
Abbildung 96: DITF-MR-RS Vergleich Personen.....	55
Abbildung 97: DITF-MR-RS Backup.....	55
Abbildung 98: DITF-MR-RS Informationsobjekttyp - Zugriffsrechte.....	56

Tabellenverzeichnis

Tabelle 1: AceWiki Funktionalitätsvergleich ACE und AceWiki.....	38
--	----

1 Einführung

„Alle 5 – 7 Jahre verdoppelt sich das weltweit verfügbare Wissen.“ [Sc01] Dieses Wissen zu speichern stellt somit eine gewaltigen Aufgabe dar. Eine bestimmte Information wiederzufinden und vor allem aus dem vorhandenen Wissen neue, bisher unbekannte Zusammenhänge zu erschließen, ist jedoch eine zweite, nicht minder anspruchsvolle Aufgabe. Immer mehr Informationen sind parallel oder gar ausschließlich im Internet abrufbar und stehen so an jedem Ort mit Anschluss an das weltweite Datennetz zur Verfügung. Wikis stellen eine der vielen Möglichkeiten dar um Informationen im Internet zu speichern und für einen Menschen auswertbar darzustellen. Zur Verknüpfung und maschinellen Bearbeitung von Information aber auch zum besseren Verständnis für eine Nutzer bedarf es zusätzlicher Metadaten. Einen Beitrag zu diesen Aufgaben können semantische Wikis leisten. Mit ihnen beschäftigt sich diese Arbeit.

1.1 Motivation

Die Idee der Vernetzung von Information ist kein Kind der Entwicklung des Internets. Wie [Sc08a] beschreibt, hat schon 1934 der belgische Visionär Paul Otlet die Idee eines Wissensnetzes dargelegt. „Otlet wollte zum Beispiel Informationshappen nicht nur einfach verlinken wie im World Wide Web. Er schlug vielmehr intelligente Links vor, die zusätzlich auch Informationen beinhalten über Wahrheitsgehalt und Kontext.“ [Sc08a]

Seit der Entwicklung von HTML in Verbindung mit dem ersten Browser und dem ersten Webserver durch Tim Bernes-Lee Anfang der 1990er Jahre [Wi09d] sowie der immer größer werdenden Nutzerzahl dieses World Wide Webs seit Mitte der 1990er Jahre ist die Anzahl der Webseiten laut [Ne09] nunmehr auf über 180 Millionen angewachsen. Dies stellt eine für Menschen unüberblickbare Menge an Informationen und Daten dar, welche auf Grund fehlender Beschreibungsdaten zu diesen Informationen zudem nur rudimentär maschinell verarbeitet werden können.

Das Ziel des ebenfalls von Tim Bernes-Lee vorgeschlagenen Semantischen Webs [BHL01] besteht in der Erweiterung des World Wide Webs um die Möglichkeit die Bedeutung einer Information beschreiben zu können. Dies erlaubt es, neben dem Menschen, auch Computern die Daten zu interpretieren und vorallem weiterverarbeiten zu können [Wi09e]. „The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation.“ [BHL01]

Die Entwicklung des ersten Wikis namens „WikiWikiWeb“ durch Ward Cunningham im Jahr 1994 [SBB07] legte die Grundlage dafür, „[...] dass die Benutzer von einfachen Lesern zu Mitarbeitern, zu Mitschöpfern, werden [...]“ [VKV07 S.790]. Diese soziale Komponente des „Mitmach-Webs“, sich selbst einbringen und mitwirken zu können, hat zu einem ungeahnten Wachstum sowohl bei Anzahl als auch beim Inhalt der Wiki-Seiten geführt. Das beste Beispiel ist das Wikipedia-Projekt mit all seinen Ablegern. „Es wird also mehr Wissen gesammelt als jemals zuvor, und es stellt sich die Frage, wie man die Informationsmenge noch organisieren und sinnvoll zusammenführen kann.“ [VKV07 S.790]

Ein semantisches Wiki „[...] erweitert ein [normales] Wiki um Metadaten in Form von semantischen Annotationen [...]“ [Sc09] zu den Strukturelementen. Hierbei kann es sich um Wiki-Seiten, Abschnitte, Sätze oder gar einzelne Wortgruppen oder Wörter handeln. „Semantische Annotationen korrespondieren in der Regel mit einer Ontologie, in der definiert wird, welche Eigenschaften welchen Objekttypen zugeordnet werden können. Die erstellten Wissensmodelle werden meist in

den Ontologiesprachen RDF/S bzw. OWL repräsentiert.“ [Sc09] Grundlegende Eigenschaften semantischer Wikis sind einfache Möglichkeiten zur Annotation von Links, Wiki-Seiten oder anderen Inhalten sowie eine semantische Suche nach Zusammenhängen zwischen Inhalten. [Sc09] Unter Umständen existiert auch ein Mechanismus zur „[...] möglicherweise automatische[n] oder halbautomatische[n] Extraktion von Metadaten aus den Wiki-Artikeln, um die Annotation zu erleichtern.“ [Sc09]

Im Folgenden sind zum Begriff *Semantisches Wiki* einige Definitionsversuche und Kurzbeschreibungen aufgeführt:

„A Semantic Wiki is a Wiki enhanced with technologies developed by the Semantic Web community in order to encode more knowledge than just structured text and hyperlinks. Usually this extra knowledge is available in a formal language, so that machines can (at least partially) process it. In particular, machines can calculate new facts from the given facts.“ [Sw09c]

„Ein Semantisches Wiki ist ein Wiki, welches ein Modell des in ihm gespeicherten Wissens hat. Normale Wikis verwenden strukturierten Text und nicht typisierte Hyperlinks. Semantische Wikis ermöglichen, Wissen über Seiten (Metadaten) und ihre Relationen festzuhalten. Die Wissensbasis eines semantischen Wikis ist üblicherweise in einer formalen Sprache, so dass Maschinen es zumindest teilweise automatisch verarbeiten können. Die Technologien aus dem Bereich des Semantischen Web bilden dabei die Basis für automatische Schlussfolgerungen in der Wissensbasis.“ [Wi09f]

„A semantic wiki is a wiki that has an underlying model of the knowledge described in its pages. Regular wikis have structured text and untyped hyperlinks (such as the links in this article). Semantic wikis allow the ability to capture or identify further information about the pages' (metadata) and their relations.“ [Wi09g]

„Semantic Wikis try to combine the strengths of Semantic Web (machine processable, data integration, complex queries) and Wiki (easy to use and contribute, strongly interconnected, collaborativeness) technologies.“ [Se09]

Diese Definition stellen fast alle auf die Unterstützung der maschinellen Verarbeitung der Daten in den Vordergrund. Unbeachtet bleibt meist, dass Strukturierungen und Ontologien auch für den Benutzer Mensch hilfreich sein können, da dieser Strukturen häufig als gedankliches Ordnungsschema verwendet.

1.2 Aufgabenstellung

Es existiert gegenwärtig eine große Anzahl von Anwendungen, welche die Strukturierung von Inhalten auf unterschiedlichste Art und Weise verwirklichen. Dabei ist zu beobachten, dass die sich die Konzepte auf einige grundsätzliche Ansätze beschränken. Normale Wiki-Systeme verwenden den Ansatz des *Text first*. Der Inhalt als die zentrales Element der Modellierung wird durch die Benutzer gemeinschaftlich erstellt und gepflegt. Die Strukturierung erfolgt ausschließlich im und mit dem Text. Dies hat zur Folge, dass Wiki-Seiten zu Objekten gleichen Typs, zum Beispiel Vorlesungen, einen grundverschiedenen Aufbau haben können. Der Ansatz *Schema first*, zum Beispiel verwendet bei Datenbanken, beruht auf der Definition eines Modells zur Strukturierung und vorallem der Vereinheitlichung der Dateneingabe. Der umgekehrte Ansatz wird als *Instance first* bezeichnet. Hierbei wird als erstes eine Instanz der zu strukturierenden Objekte erstellt. Dies geschieht ohne vorher ein festes Schema festgelegt zu haben. Das Schema entwickelt sich bei der Erstellung der Instanzen.

In dieser Arbeit sind Systeme mit den unterschiedlichen Ansätzen zu betrachten sowie die verschiedenen Eigenschaften anhand eines durchgängigen Beispiels zu evaluieren. Anschließend sind die gewonnenen Erkenntnisse zu diskutieren sowie in Empfehlungen für ein Wiki-System mit semistrukturierten Inhalten des Lehrstuhls sebis zu überführen.

Angestrebt wird hierbei die vorteilhafte Verbindung der oben genannten Ansätze, besonders die Schemaevolution des *Instance first* Ansatzes mit dem sich immer weiter verbreitenden Ansatz der Vorlagenverwendung zur semistrukturierten Datenerhebung.

1.3 Aufbau der Arbeit

Diese Arbeit gliedert sich neben der Einführung in drei weitere Kapitel.

Unterschiedlichste Realisierungsansätze semantischer Wikis werden in Kapitel zwei untersucht und beschrieben. Die hierfür ausgewählten Systeme sind *Semantic MediaWiki*, *NEPOMUK*, die *Hypertext Knowledge Workbench*, *AceWiki* sowie das *DITF-MR Recherchesystem*. Es werden die Strukturierung der Inhalte, Eingabeformen der Metadaten, Abfragesprachen und Suchfunktionen, Import- und Exportfunktionalitäten, die Useability sowie Besonderheiten der genannten Systeme betrachtet. Die Eigenschaften und Funktionen der gewählten Wiki-Systeme werden anhand eines Beispiels verdeutlicht.

Ein Vergleich der untersuchten Systeme anhand der genannten Eigenschaften erfolgt in Kapitel drei. Aus dieser Diskussion ergeben sich die Empfehlungen für die Entwicklung eines am Lehrstuhl sebis eingesetzten Wiki-Systems.

In Kapitel vier wird die geleistete Arbeit zusammengefasst und kritisch bewertet. Ideen und Anregungen für weitere Arbeiten und Forschungsfragen auf dem Gebiet der semantischen Wikis sind ebenfalls Inhalt dieses Kapitels.

2 Verwandte Arbeiten – Semantische Wikis

Im nun folgenden Kapitel werden unterschiedliche Ansätze semantischer Wikis betrachtet. Die Hauptgliederungspunkte sind ein kurzer Steckbrief, die Strukturen für Darstellung und Speicherung semantischer Inhalte, die Erstellung semantischer Abfragen, die Export- und Importfunktionalität, die Ausgestaltung des Berechtigungskonzepts und die Useability.

Der Steckbrief enthält für jeweiligen Wiki-Engine einige charakteristische Eigenschaften, welche sich an den Attributen für semantische Wikis der Website *semanticweb.org* orientieren. Neben dem Status, der Nummer und dem Veröffentlichungsdatum der untersuchten Version werden auch das Lizenzmodell und die Projekthomepage angegeben. In welcher Form der Benutzer Inhalte in das Wiki einbringt und pflegt gibt das Bearbeitungsmodell an [Sw08g]. Als annotierbare Elemente werden die Komponenten eines Wikis aufgeführt, zu denen sich Informationen hinterlegen lassen, zum Beispiel: Dokumente, Abschnitte, Sätze und Worte [Sw08h]. Welche Form diese Informationen annehmen können, erklärt der Annotationstyp [Sw08i]. Weiterhin wird ausgewertet, ob eine Unterscheidung zwischen Wiki-Seiten und Konzepten erfolgt, d.h. das es für Wiki-Seiten klare Strukturierungsvorgaben oder Vorlagen gibt [Sw08j]. Können weitere Informationen wie Datum oder Ersteller zu den Annotationen gespeichert werden, gibt dies das Merkmal Kontextinformationen an [Sw08k]. Außerdem werden Aussagen über die Export- und Importmöglichkeit von Schemas und Inhalten getroffen.

Eine tabellarische Merkmalsübersicht der hier beschriebenen sowie weiterer Wiki-Engines befindet sich unter [Sw08f].

Die im Folgenden untersuchten Wiki-Engines werden am Beispiel einer zu universitären Vorlesung erklärt. Dabei wird versucht mit den Möglichkeiten des jeweiligen Wikis alle Beziehungen und Eigenschaften einer Vorlesung darzustellen.

2.1 Semantic MediaWiki (SMW)

Bei SWM [Se08a] handelt es sich um eine Erweiterung der Wiki-Engine MediaWiki [Me08a], welche unter anderem durch das Wikipedia-Projekt (<http://wikipedia.org>) verwendet wird.

2.1.1 Steckbrief

Status:	stable
Version:	1.4.0
Veröffentlichungsdatum:	2008-11-21
Lizenzmodell:	GPL
Homepage:	www.semantic-mediawiki.org
Bearbeitungsmodell:	Wiki-Syntax, Formularfelder
Programmiersprache:	PHP
Annotierbare Elemente:	Dokument
Unterscheidung zwischen Wiki-Seiten und Konzepten:	Ja
Annotationstyp:	Zeichen, Seiten, URI's
Kontextinformationen über Annotationen:	Nein
Instanzenimport:	nur mit externen Skripten
Ontologieimport:	nur mit externen Skripten
Instanzenexport:	Ja
Ontologieexport:	Ja
[Sw08a]	

2.1.2 Strukturierung

SMW besitzt *Kategorien* (*Categories*), *Attribute* (*Properties*), *Typen* (*Types*) und *Vorlagen* (*Templates*) als Strukturelemente. Diese werden nachfolgend im Einzelnen beschrieben.

2.1.2.1 Kategorien

Kategorien [Me08b] als Strukturelemente sind schon Bestandteil des MediaWiki Projekts. Eine Wiki-Seite lässt sich, wie in Abbildung 1 zu sehen, frei definierbaren Kategorien zuordnen. Kategorien selbst können wiederum in Kategorien zusammengefasst werden. Es muss dabei keine Hierarchie eingehalten werden, d.h. der entstehende Kategoriegraph ist nicht notwendigerweise azyklisch und bildet nicht zwingend einen Baum. Es liegt hier vollkommen in der Hand des Benutzers eine gewisse Ordnung einzuhalten. Die Zuordnung einer Wiki-Seite zu einer Kategorie erfolgt mit dem Eintrag `[[Category:KategorieName]]`.

The screenshot shows the 'Weblinks' section of a SMW page. It includes a 'Commons' link for Opel, a list of links to Opel's website, mobile site, and press releases, and three expandable sections: 'Aktuelle Opel-Modelle', 'Nicht mehr produzierte Opel-Modelle', and 'Automobilmarken von General Motors'. At the bottom, there is a list of categories: Opel, Automobilhersteller, Omnibushersteller, Fahrradhersteller, Motorradhersteller, Nutzfahrzeughersteller, Traktorenhersteller, Markenname, Unternehmen (Rhein-Main), Hoflieferant, and Rüsselsheim.

Abbildung 1: SMW Kategorien

Für jede Kategorie kann eine eigene Wiki-Seite angelegt werden, auf der die Kategorie beschrieben wird. Wie in Abbildung 2 dargestellt, enthält diese Seite eine automatisch generierte Auflistung der diese Kategorie verwendenden Wiki-Seiten.

The screenshot shows the Wikipedia page for the category 'Motorradhersteller'. It features the Wikipedia logo, a search bar, and a navigation menu. The main content area displays the category title, a list of subcategories, and a list of member pages. The member pages are organized into three columns: 'A', 'E (Fortsetzung)', and 'M (Fortsetzung)'. The 'A' column lists A.C.E. Corporation, A.J.S., and A.K.D. The 'E' column lists Elsterwerdaer Fahrradfabrik, Ermag, and Express Werke. The 'M' column lists Minerva (Auto), Minerva (Motorrad), Mobylette, Monark, and Montesa.

Abbildung 2: SMW Kategorie-Seite

2.1.2.2 Attribute und Typen

Mittels *Attributen* [Se08b] lassen sich Links zu anderen Artikeln weiter beschreiben. Die entsprechende Syntax lautet:

- für einen normalen Link `[[property::Link]]`,
- für einen Link mit alternativer Beschriftung `[[property::Link|Text]]`,
- für eine versteckte Property `[[property::Link| ]]` (Leerzeichen als Alternativtext) und
- für einen Link mit mehreren Properties `[[property1::property2::Link]]`.

Für jede verwendete Property kann eine extra Wiki-Seite erstellt werden, welche die Bedeutung dieser Property erklären sollte.

Auch jeder beliebige andere Zeichenkette kann als Wert einer Property definiert werden: `[[property::value]]`. Aufgrund der Defaulteinstellung würde SMW diese Property zunächst als Link behandeln. Durch die Angabe eines *Typs* auf der Wiki-Seite der Property ist es möglich dies zu verhindern. Der Ausdruck dazu lautet `[[has type::TypeName]]`. Standarddatentypen sind `String` für Zeichenketten, `Number` optional mit Exponent für ganze und rationale Zahlen und `Page`, als Defaulttyp, für Links. Für den Typ `Number` können zusätzlich Maßeinheiten angegeben werden. Hierzu wird dem Benutzer gestattet eigene Datentypen [Se08c] zu erstellen und zu beschreiben, wie die Maßeinheiten ineinander umzurechnen sind. Dies erlaubt sowohl dem Ersteller als auch den Lesern einer Wiki-Seite ihre bevorzugten Maßeinheiten zu verwenden ohne diese von Hand umrechnen zu müssen. Abbildung 3 zeigt eine Type-Seite inklusive der `[[corresponds to:: number unit1, unit2, ... ]]` Tags zur Umrechnung der Maßeinheiten. Spezielle Datentypen sind zum Beispiel für Datumsangaben, geografische Koordinaten oder Temperaturen schon vorhanden. Seit Version 1.0 des SMW sind auch Properties mit mehreren Typen [Se08d] zulässig.

Editing Datentyp:VL Umfang

Preview

This is only a preview; changes have not yet been saved!

Number

1 SWS, Semesterwochenstunde
2 StudWS
0,5 NWS

Fakten zu VL Umfang

RDF-Feed

Datentyp Zahl

Entspricht 1 SWS, Semesterwochenstunde, 2 StudWS und 0,5 NWS



```
[[has type::Number]]  
  
[[corresponds to:: 1 SWS, Semesterwochenstunde]]  
[[corresponds to:: 2 StudWS]]  
[[corresponds to:: 0,5 NWS]]
```

Abbildung 3: SMW Datentyp VL_Umfang

In Abbildung 4 wird die Wiki-Seite des Attributs VL_Umfang dargestellt, welches den selbst erstellten Typ VL_Umfang verwendet. Hier sind auch die das Attribut verwendenden Wiki-Seiten aufgelistet.



Abbildung 4: SMW Attribut

Eine Propertyhierarchie kann mit dem Konstrukt `[[subproperty of::Subproperty:Property]]` erstellt werden. Dieser Eintrag erfolgt auf der Wiki-Seite des annotierten Objekts. Inverse Properties werden nicht unterstützt. [Se08i]

Mit Hilfe der MediaWiki – Suchseite können sowohl vorhandene Properties als auch vorhandene Typen gesucht werden.

2.1.2.3 Semantische Vorlagen

Zur Beschreibung mehrerer gleichartiger Objekte lassen sich in SWM semantische *Vorlagen* [Se08e] erstellen. Diese Templates fassen entsprechende Properties zusammen. Der Ersteller einer Wiki-Seite muss somit nicht mehr die einzelnen Attribute zur Beschreibung eines Objektes kennen. Es reicht aus das entsprechende Template auf übliche Weise zu verwenden um entsprechende strukturierte Informationen zu einem Objekt zur Verfügung zu stellen. Ein semantisches Template „Vorlesungsinfo“ könnte diese Gestalt haben:

```
{| class="infobox" border="1"
!Titel:
| [[VL_Titel::{{{Titel}}}]]
|-
!Modulnr:
| [[VL_Modulnr::{{{Modulnr}}}]]
|-
!Veranstalter:
| [[VL_Veranstalter::Person::{{{Veranstalter}}}]]
|-
!Betreuer:
| [[VL_Betreuer::Person::{{{Betreuer}}}]]
|-
!ECTS:
| [[ECTS::{{{ECTS}}}]]
|-
!Umfang:
| [[VL_Umfang::{{{Umfang}}}]]
|-
!Ort:
| [[Coordinates::{{{Ort}}}]]
|-
!Zeit:
| [[VL_Zeit::{{{Zeit}}}]]
```

```

|-
!Prüfung:
| [[VL_Pruefung::{{{Pruefung}}}}]]
| }

```

In die Wiki-Seite einer Vorlesung würde zur Verwendung des Templates eingefügt:

```

{{Vorlesungsinfo
|Titel=SEBA Bachelor
|Modulnr=IN2085
|Veranstalter=Prof. Dr. Florian Matthes
|Betreuer=Christian Neubert
|ECTS=5
|Umfang= 2 SWS
|Ort=48°15'44'';11°40'07''
|Zeit= Mo 12.00 - 13:00
|Pruefung=schriftlich
}}

```

Durch die Definition der semantischen Vorlage werden die angegebenen Werte zu Properties. Wie das oben erstellte Template bei Verwendung dargestellt wird und wie SMW die einzelnen Attribute in der sogenannten Factbox zusammenfasst, zeigt Abbildung 5.

The screenshot shows a MediaWiki page titled "SEBA Bachelor". At the top, there is a navigation bar with links: Page, Discussion, Edit, History, Delete, Move, Protect, Watch, Refresh. Below the title, there is a table generated by the {{Vorlesungsinfo}} template:

Titel:	SEBA Bachelor
Modulnr:	IN2085
Veranstalter:	Prof. Dr. Florian Matthes
Betreuer:	Christian Neubert
ECTS:	5
Umfang:	2 SWS
Ort:	48°15'44";11°40'07"
Zeit:	Mo 12.00 - 13:00
Prüfung:	schriftlich

Below the table, it says "weitere Betreuer: Max Mustermannn".

Below that is a section titled "Fakten zu SEBA Bachelor" with a "RDF-Feed" icon. It lists various properties and their values, each with a link icon:

- Coordinates: 48°15'44"N, 11°40'07"O
- ECTS: 5
- Person: Prof. Dr. Florian Matthes, Christian Neubert und Max Mustermannn
- VL Betreuer: Christian Neubert und Max Mustermannn
- VL Modulnr: IN2085
- VL Pruefung: schriftlich
- VL Titel: SEBA Bachelor
- VL Umfang: 2 SWS (4 StudWS, 1 NWS)
- VL Veranstalter: Prof. Dr. Florian Matthes
- VL Zeit: Mo 12.00 - 13:00

At the bottom, there is a "Category: Vorlesung" link.

Abbildung 5: SMW Template und Properties

Das Zusammenspiel der einzelnen Strukturelemente stellt Abbildung 6 als Klassendiagramm dar.

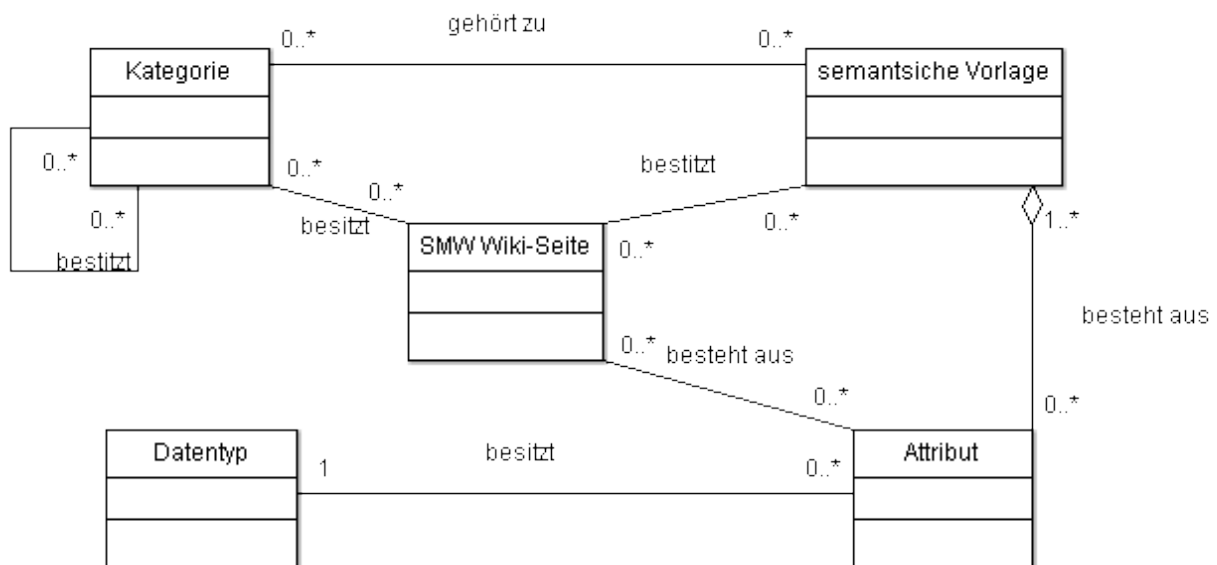


Abbildung 6: SMW Klassendiagramm

2.1.2.4 Links zu Online Services

Als besonderes Feature stellt SMW zur weiteren Recherche für spezielle Attribute (zum Beispiel geografische Daten) eine Möglichkeit zur Verfügung Online Services (zum Beispiel Google Maps) direkt über die Factbox einer Wiki-Seite abrufen zu können (Abbildung 7). Wie die Links zu Online Services einzurichten sind beschreibt [Se08o].

Umfang:	2 SWS
Ort:	48°15'44"N; 11°40'07"
Zeit:	Mo 12.00 - 13.00
Prüfung:	schriftlich

weitere Betreuer: Max Mustermann

Find online maps, Google maps, Mapquest

Fakten zu SEBA Bachelor

Coordinates 48°15'44"N, 11°40'07"O + 🔗

ECTS 5 + 🔗

Person Prof. Dr. Florian Matthes + 🔗, Christian Neubert + 🔗 und Max Mustermann + 🔗

VL Betreuer Christian Neubert + 🔗 und Max Mustermann + 🔗

Abbildung 7: SMW Online Services

2.1.3 Abfragesprache

SMW verfügt über mehrere Möglichkeiten zur Suche und Verwendung semantischer Informationen.

2.1.3.1 Suchseite

Zur semantischen Suche [Se08f] stellt SMW eine Spezialseite (Abbildung 8) bereit. Hier können Suchabfragen nach Kategorien oder Properties abgesetzt werden. Das Suchergebnis setzt sich aus den Wiki-Seiten zusammen, welche die geforderten Kategorien bzw. Properties enthalten. Zur Eingrenzung oder Erweiterung des Ergebnisses können Operatoren und Platzhalter (Wildcard) verwendet werden:

- < kleiner,
- > größer,

- ! nicht gleich,
- ~ <<like>> für Stringvergleiche und
- + alle Elemente.

Es können außerdem für die Trefferliste zusätzlich anzuzeigende Properties angegeben werden.

The screenshot shows the SMW Query interface. At the top, there is a 'Special' tab. Below it, the 'Query' section contains the query: `[[Kategorie:Vorlesung]] [[ECTS::>2]]`. To the right, under 'Additional printouts (optional)', there are two printouts: 'VL Titel' and '?ECTS'. Below the query section, there are links for '[Add sorting condition]', 'Find results', 'Hide query', and 'Querying help'. The results section shows a table with two columns: 'VL Titel' and 'ECTS'. The table contains two rows: 'SEBA Bachelor' with '5' ECTS, and 'SEBA Master' with '5' ECTS. Navigation links like 'Previous', 'Results 1-2', 'Next', and a range '(20 | 50 | 100 | 250 | 500)' are visible above and below the table.

VL Titel	ECTS
SEBA Bachelor	5
SEBA Master	5

Abbildung 8: SMW Suche

2.1.3.2 Inline-Queries

Im Quelltext einer Wiki-Seiten können Abfragen [Se08g] eingefügt werden. Dies ermöglicht die Erstellung dynamischer Wiki-Seiten unter Zuhilfenahme strukturierter Informationen. Die doppelte Pflege von Informationen auf der Wiki-Seite selbst und dazugehörigen Übersichtsseiten ist nicht mehr notwendig. Ein Query setzt sich aus folgenden Elementen zusammen:

- der `#ask`: Parserfunktion,
- der eigentlichen Abfrage (wie bei der Suchseite),
- den anzuzeigenden Properties [Se08h] optional mit einer alternativen Spaltenbeschriftung sowie Maßeinheit beim Typ Number und
- diverse Sortier- und Beschränkungsanweisungen (ebenfalls optional).

In Abbildung 9 sind eine solche Abfrage und die dazugehörige Ausgabe zu sehen.

Page

Discussion

Edit

History

Delete

Move

Protect

Watch

Refresh

Editing Vorlesungen

Preview

This is only a preview; changes have not yet been saved!

max. 10 Vorlesungen > 2 ECTS sortiert nach dem Titel

	Titel	Veranstalter	Umfang in NWS	Raum	ECTS
SEBA Bachelor	SEBA Bachelor	Prof. Dr. Florian Matthes	1 NWS	48°15'44"N, 11°40'7"O	5
SEBA Master	SEBA Master	Prof. Dr. Florian Matthes	1 NWS	48°15'44"N, 11°40'7"O	5

B

/

Ab

A

max. 10 Vorlesungen > 2 ECTS sortiert nach dem Titel

```

{{#ask:
[[Kategorie:Vorlesung]] [[ECTS::>2]]
|?VL_Titel = Titel
|?VL_Veranstalter = Veranstalter
|?VL_Umfang#NWS = Umfang in NWS
|?Coordinates = Raum
|?ECTS
|link=all | limit=10 | sort=VL_Titel | order=asc
}}

```

Abbildung 9: SMW Inline Query

Bei der Suche bezieht SMW auch Unterkategorien mit ein. Suchanfragen nach einer Property berücksichtigen eventuell vorhandene Subproperties ebenfalls.

2.1.3.3 Concepts

Als *Concepts* [Se08j] werden in SMW gespeicherte und auf anderen Seiten wiederverwendbare Queries bezeichnet. Für sie besteht ein eigener Namensraum, in dem Wiki-Seiten für Concepts erstellt werden können. Jedes Concept bedarf einer eigenen Wiki-Seite.

2.1.4 Export

SMW ermöglicht den Export [Se08k] der angelegten Ontologien und Daten im OWL/RDF Format. Eine Spezialseite stellt diese Funktion bereit. Ebenso ist der Export in RDF mit Hilfe verschiedener externer Scripts und Tools möglich [Se08l].

2.1.5 Import

Zum Import von Properties [Se08m] und Ontologien [Se08n] sind in SMW Ansätze vorhanden. Diese befinden sich aber zum jetzigen Zeitpunkt im Beta-Status, da sie noch nicht vollständig ausgereift und voll funktionsfähig sind.

2.1.6 Berechtigungskonzept

Semantic MediaWiki bringt kein eigenes Berechtigungskonzept mit. Als Erweiterung zu MediaWiki verwendet SMW auch dessen Nutzer- und Rechteverwaltung [Me08d]. Benutzer werden unter anderem in nicht registrierte Benutzer (alle), registrierte / angemeldete Benutzer und Administratoren unterteilt. Grundsätzlich und standardmäßig darf jeder eine Wiki-Seite oder eine Diskussion ansehen und erstellen. Angemeldete Benutzer dürfen zu dem unter anderem Seiten verschieben und Dateien hochladen. Administratoren unterliegt die Verwaltung des Systems. Dazu

zählt die Benutzerverwaltung, das Löschen von Wiki-Seiten oder auch die Wiederherstellung gelöschter Seiten. Außerdem ist es ihnen gestattet das Editieren von Wiki-Seiten auf bestimmte Nutzerkreise einzuschränken oder ganz zu verbieten [Me08e]. Gruppenrechte lassen sich durch die Bearbeitung der richtigen Konfigurationsdatei aber auch ändern [Me08f].

2.1.7 Useability

Die Kategorien, welche ein Benutzer zur Beschreibung einer Wiki-Seite verwenden möchte, müssen ihm bekannt sein. Es gibt während des Editieren einer Seite keine Kategorienauswahl oder ähnliches. Das MediaWiki Projekt „Meta-Wiki“ [Me08c] bietet hier eine einfache Lösung: es listet auf einer separaten Seite alle Kategorien sowie deren Unterkategorien auf.

Die entsprechende Wiki-Seite zu selbst definierten Properties sollten vor der ersten Verwendung der Property erstellt werden. Sonst muss eine die Property bereits verwendende Wiki-Seite erneut gespeichert bzw. aktualisiert werden um die strukturierten Informationen als solches sichtbar zu machen. Eine direkte Auswahl oder Hilfe beim Editieren einer Wiki-Seite zum Einfügen von Properties bzw. der Angabe von Types ist nicht vorhanden. Es sind aber Spezialseiten mit einer Übersicht über alle Properties und Types verfügbar.

Die Erstellung von Abfragen bedarf genauer Kenntnis der verwendeten Properties. Auch hier existiert keine Unterstützung oder Führung des Erstellers durch SMW.

Wichtig ist die Beachtung der Strukturelementbezeichnung. Wird ein Element zum Beispiel in deutsch erstellt, sind auch bei Abfragen etc. die deutschen Bezeichnungen zu verwenden (zum Beispiel Category vs. Kategorie).

2.2 NEPOMUK

Bei NEPOMUK [Ne08a] handelt es sich nicht um ein semantisches Wiki sondern um einen „sozialen semantischen Desktop“ im eigentlichen Sinne [Ne08a], der ein Wiki enthält. NEPOMUK ist ein Akronym und steht für „Networked Environment for Personalized, Ontology-based Management of Unified Knowledge“.

Das Projekt wird vom Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI GmbH www.dfki.de) betrieben und von der EU mit 11,5 Millionen Euro gefördert. Der Projektzeitraum erstreckt sich von Anfang 2006 bis Ende 2008 [Ne08c]. Als bekannteste industrielle Partner sind IBM (www.ibm.com), SAP (www.sap.com) und HP (www.hp.com) zu nennen [Ne08d]. Wissenschaftliche Partner sind unter anderem das Forschungszentrum Informatik der Universität Karlsruhe (www.fzi.de) sowie weitere Universitäten aus Irland, Griechenland, der Schweiz und Schweden [Ne08d].

Das Ziel des NEPOMUK - Projektes ist die Entwicklung einer umfassenden Lösung zur Integration des Endbenutzer - Arbeitsplatzes in eine Kollaborationumgebung [Ne08b]. Das Hauptaugenmerk liegt auf der besseren Organisation und Verbreitung von Informationen, die von einzelnen Personen oder Gruppen erstellt wurden und werden. Dieses Ziel soll durch die Förderung der Online - Kollaboration und die Verbesserung des persönlichen Datenmanagements erreicht werden [Ne08b].

Ein sogenannter „Social Semantic Desktop“ [Wi08a] wird in diesem Zusammenhang als Lösungsansatz betrachtet. Es handelt sich dabei um einen erweiterten persönlichen Arbeitsplatz, der Informationen mit wohldefinierten Bedeutungen versieht, diese somit für Computer verwertbar macht und zusätzlich den Austausch mit anderen Arbeitsplätzen und Benutzern solcher ermöglicht [Ne08b].

Zur Realisierung des semantischen Arbeitsplatzes werden Datenstrukturen, Methoden und Dienste benötigt, die frei wählbar Informationen beschreiben und über Dateiformate, Medientypen und Anwendungen hinweg verbinden. NEPOMUK verwendet dazu Methoden und Datenstrukturen des Semantic Web und passt diese geeignet an. Zur Gliederung und Darstellung der Ideen und Gedanken eines Benutzers werden Wiki-Technologien mit Annotationsmechanismen verknüpft. Die Erstellung und Bearbeitung von Inhalten soll auf die Arbeitsweise des Benutzer abgestimmt werden. Hierzu werden agile Prozessmodelle mit Konzepten der Informationserstellung und -strukturierung verbunden [Ne08b].

Der soziale Anteil des Desktops soll Anwendungen zum Aufbau sozialer Beziehungen und zum Informationsaustausch in sozialen Gruppen enthalten. Bei der Verbreitung von Daten und Dokumenten sollen keine isolierten Informationselemente sondern alle relevanten Information inklusive dem entsprechenden Kontext und den teilhabenden Gruppen ausgetauscht werden. Ebenso sollen Methoden zur Suche in und Speicherung von verteilten Informationen entwickelt werden. So soll eine verteilte Wissensbasis für Interessengruppen entstehen [Ne08b].

Mit Hilfe der Open Source Community sind bis dato die Architektur und die dazu implementierte Middleware für einen „Social Semantic Desktop“ mit klar definierten Schnittstellen entwickelt worden. Die Ergebnisse wurden als Open Source Software veröffentlicht und sollen dem Semantic Web Standardisierungsprozess von Gesellschaften wie OASIS (www.oasis-open.org) oder W3C (www.w3.org) zugeführt werden. NEPOMUK wird von verschiedenen Nutzergruppen angepasst und weiterentwickelt. Referenzinstallationen zum Beispiel bei SAP haben erste Erfahrungsberichte hervorgebracht [Ne08b] [Ne08c].

2.2.1 Steckbrief

Status:	beta
Version:	1.0 beta
Veröffentlichungsdatum:	täglich, benutzter Build 2008-12-06
Lizenzmodell:	BSD style
Homepage:	http://nepomuk.semanticdesktop.org/
Bearbeitungsmodell:	Formularfelder
Programmiersprache:	Java, basierend auf Eclipse Rich Client
Annotierbare Elemente:	Element
Unterscheidung zwischen Wiki-Seiten und Konzepten:	Ja
Annotationstyp:	Element
Kontextinformationen über Annotationen:	Ja
Instanzenimport:	Ja
Ontologieimport:	Ja
Instanzenexport:	Ja
Ontologieexport:	Ja

[Ne08e]

NEPOMUK ist bisher bereits in der grafischen Benutzeroberfläche für Unixsysteme 'K Desktop Environment' (KDE, www.kde.org) ab Version 4 implementiert [Ne08f]. Eine weitere Implementierung basiert auf der Eclipse Rich Client Platform (http://wiki.eclipse.org/index.php/Rich_Client_Platform) und wird als *P2P Semantic Eclipse Workbench* (NEPOMUK PSEW) bezeichnet [Ne08g]. In Zukunft sollen auch Mozilla – Produkte wie Firefox und Thunderbird von der NEPOMUK Technologie profitieren. Ein entsprechendes Projekt wurde bereits gestartet [Ne08h].

Zur Veranschaulichung der Konzepte und Möglichkeiten des NEPOMUK – Frameworks werden sich nachfolgende Ausführungen und Abbildungen auf NEPOMUK PSEW als fortgeschrittene Referenzimplementierung beziehen.

2.2.2 Strukturierung

NEPOMUK basiert auf dem Konzept der *Conceptual Data Structures* (CDS) [Vö08a S.31]. Bei CDS handelt es sich um ein Model, welches sich zur Darstellung von Datenstrukturen persönlichen Wissens mit unterschiedlichsten Formalisierungsgraden besonders eignet [Vö08a S.2] [Sw08b]. Es setzt sich aus der Datenmodellebene, auch *Semantic Web Content Model* (SWCM) genannt, und der Ontologieebene (CDS ontology) zusammen [Vö08a S.2].

2.2.2.1 Semantic Web Content Model

Das SWCM besteht aus den Elementen *Model*, *Item*, *NameItem*, *Relation* und *Statement* [Vö08a S.10f]. In Abbildung 10 ist das Zusammenspiel der SWCM Elemente dargestellt.

2.2.2.1.1 Model

Ein Model besteht aus den Items einer Wissensbasis [Vö08a S.10f].

2.2.2.1.2 Item

Jedes in SWCM enthaltene Objekt ist ein Item. Zur globalen Adressierbarkeit besitzt jedes Item einen eindeutigen Uniform Resource Identifier (URI) [Wi08j]. Zusätzlich können einem Item Inhalte binärer oder textueller Art angefügt werden. Zu beachten ist, dass diese Inhalte immer Items zuzuordnen sind und nicht allein existieren dürfen. Genauso gehört jedes Item zu einem Model [Vö08a S.10f].

2.2.2.1.3 NameItem

Ein Spezialisierung des Items stellt das NameItem dar. Es darf im Gegensatz zum Item nur textuellen Inhalt besitzen, der der Anforderung der Eindeutigkeit in einer Wissensbasis genügen muss. Es darf also nur ein NameItem mit dem selben Textinhalt existieren. Nur so ist es möglich Items mit für Menschen verständlichen Name zu versehen und zu adressieren. SWCM erzwingt damit aber nicht Name – Inhalt – Zuweisungen wie es herkömmliche Wikis tun [Vö08a S.10f].

2.2.2.1.4 Relation

Hierbei handelt es sich um ein spezielles NameItem. Beziehungen dienen zur Erläuterung von Statements. Eine Besonderheit der Beziehungen bei SWCM ist die zwingende Existenz einer inversen Beziehung [Vö08a S.10f].

2.2.2.1.5 Statement

Ein Statement ist immer ein Tripel bestehend aus Item – Relation – Item. Da es sich bei einem Statement selbst um ein Item handelt, lassen sich auch Statements mit weiteren Statements versehen. Aufgrund der inversen Beziehungen lassen sich Statements immer in zwei Richtungen darstellen: (Item1, Relation, Item2) und (Item2, inverse Relation, Item1) [Vö08a S.10f].

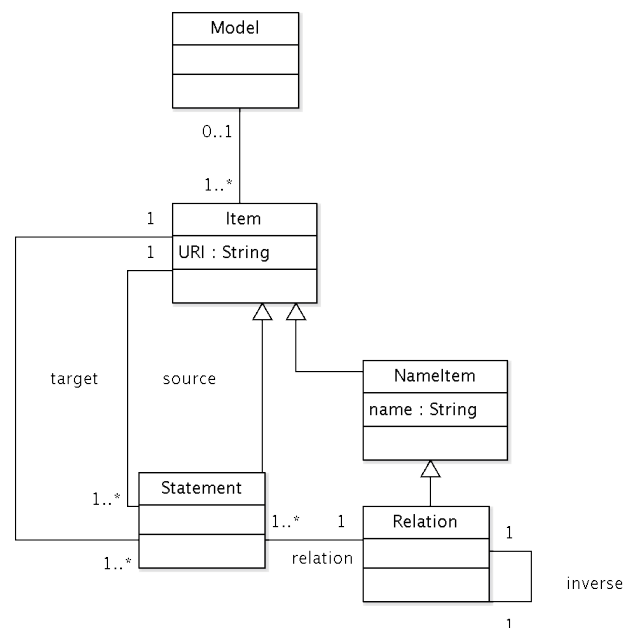


Abbildung 10: NEPOMUK SWCM Datenmodell in Anlehnung an [Vö08b S.5 Fig. 3]

2.2.2.2 CDS Ontology

„O“ und „M“ im Akronym NEPOMUK bedeuten, dass das Management der Daten in NEPOMUK auf Ontologien basiert. Während also SWCM die Wissensrepräsentation ermöglicht, erweitert die CDS Ontology SWCM um eine Beziehungshierarchie [Vö08a S.12]. Diese Hierarchie leitet von der allgemeingültigen Beziehung *related to* alle weiteren Beziehungstypen ab. Es gibt Beziehungen, die auf die Gleichheit von Items hinweisen: *similar to*, *same as* und *alias for*. Weiterhin gibt es einen Typ für Verweise: *hasTarget* mit der inversen Beziehung *hasSource*. Verwendung findet dieser Typ zum Beispiel bei Links im World Wide Web, Referenzen in Dokumenten oder Verweisen im Dateisystem. Ordnungsbeziehungen werden durch die Typen *hasAfter* und *hasBefore* dargestellt. Sie sind Subtypen von *hasTarget* und *hasSource* und erlauben auch zyklische Beziehungen. Weiter Subtypen der gerade genannten sind *hasDetail* und *hasContext*. Mit ihnen ist es möglich Hierarchien abzubilden. Zur Beschreibung von Annotationen stehen *hasAnnotation* und *hasAnnotationMember* zur Verfügung. Sie sind von *hasTarget* und *hasSource* abgeleitet. Von den Annotationstypen leiten sich wiederum die Typen für das im Web 2.0 weit verbreitete Tagging *hasTag* und *hasTagMember* ab. Schlussendlich gibt es noch Beziehungen zur Typisierung, wie zum Beispiel zwischen *Item* und *NameItem*. Hierzu stehen *hasType* und *hasInstance* zur Verfügung, welche von den Typen für das Tagging ableiten. [Vö08a S.13f] Abbildung 11 zeigt eine schematische Darstellung der Zusammenhänge von Beziehungstypen in der CDS Ontology.

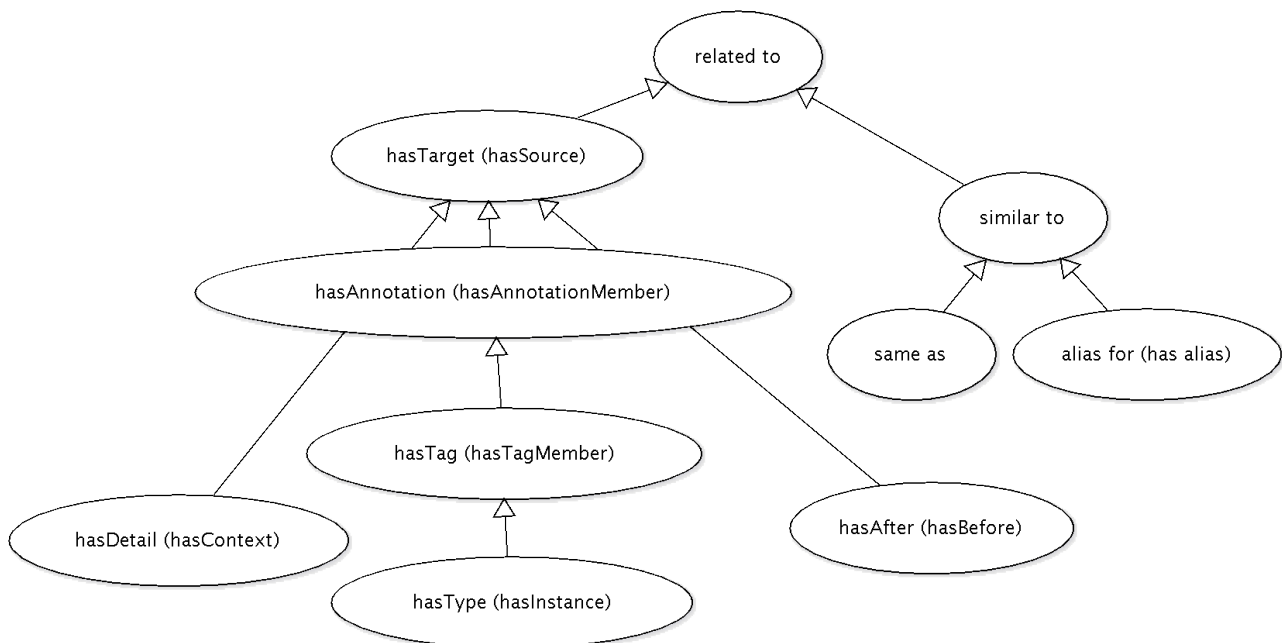


Abbildung 11: NEPOMUNK Beziehungshierarchie in Anlehnung an [Vö08a S.13 Figure 3]

2.2.2.3 Beispiel

Im folgenden wird gezeigt, wie eine Vorlesung mit diversen Eigenschaften modelliert werden kann. Verwendet wird dazu das Personal Information Model (PIMO) des NEPOMUK PSEW. Abbildung 12 zeigt die Oberfläche der genannten Anwendung. Wie zu erkennen, sind unter *My Things* bereits die notwendigen Elemente zum Informationsmanagement vorhanden. Sie werden aus dafür entwickelten Ontologien bezogen.

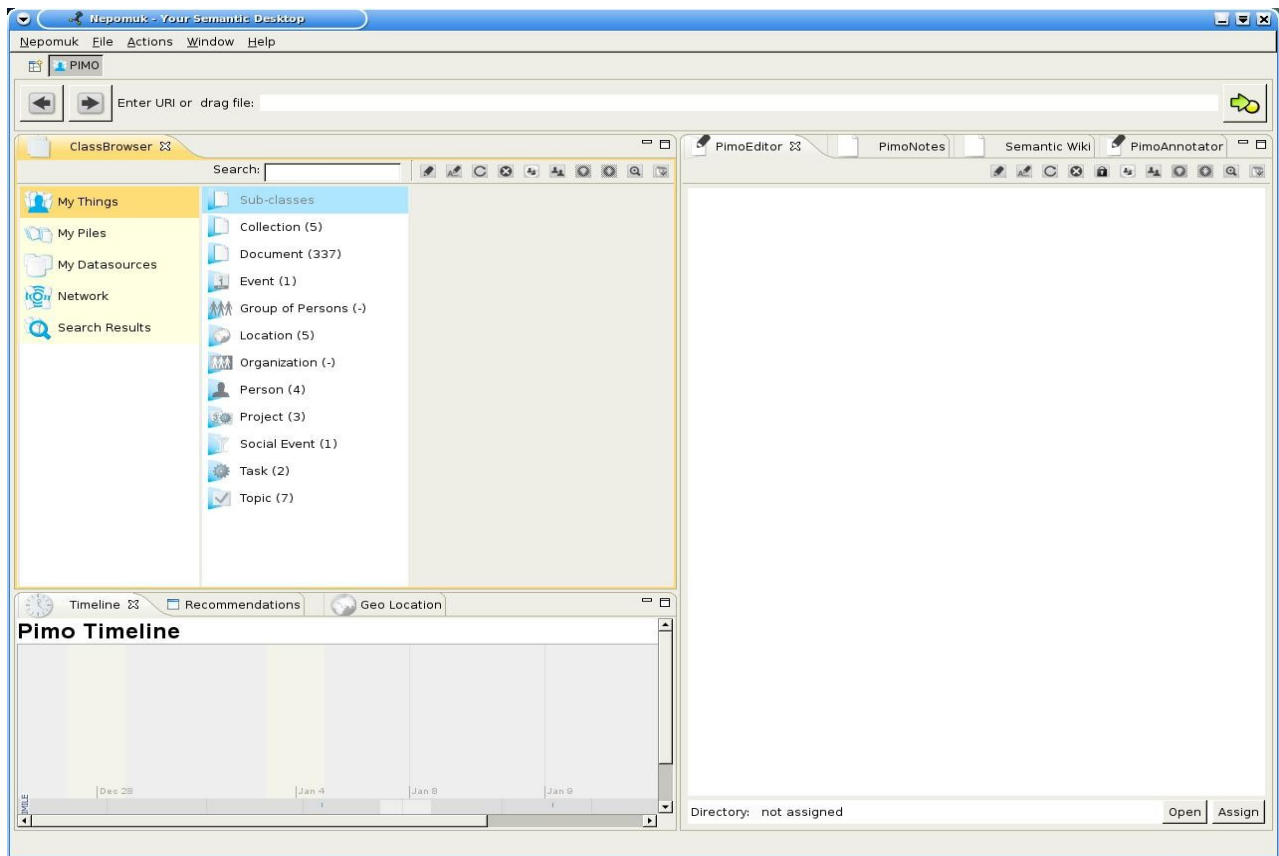


Abbildung 12: NEPOMUK PSEW PIMO

Zur Modellierung der Vorlesung werden *Project* für die Vorlesung selbst, *Person* für die beteiligten Personen, *Location* für den Hörsaal und *Event* für die Klausur zur Vorlesung verwendet. Wie in Abbildung 13 zu sehen, wurde die Klasse *Vorlesung* von *Project* abgeleitet und enthält als Instanzen die zwei beispielhaften Vorlesungen.

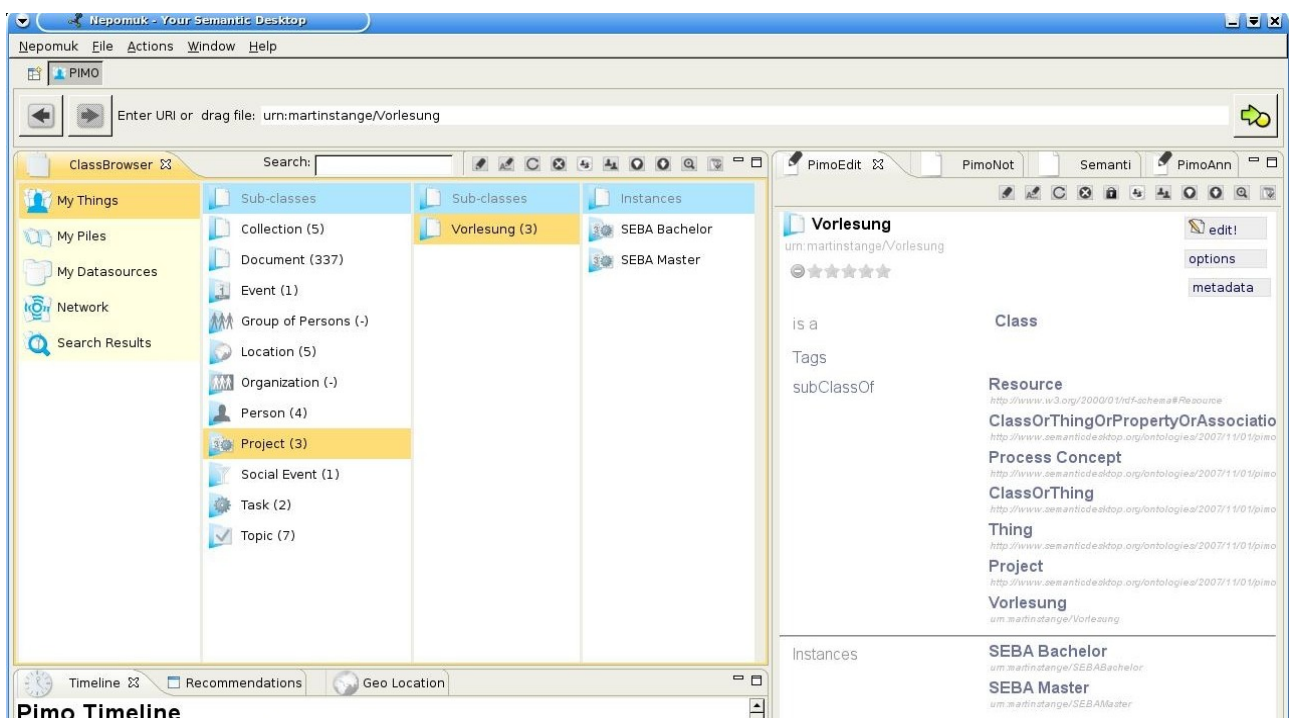


Abbildung 13: NEPOMUK Vorlesung

Abbildung 14 und 15 zeigen die Bearbeitungsmaske für das Item Vorlesung. Zu sehen sind zum einen die Standardattribute und -beziehungen und zum anderen die hinzugefügten Attribute.

Abbildung 14: PSEW Vorlesung bearbeiten I

Abbildung 15: PSEW Vorlesung bearbeiten II

Die Standardattribute eines Elements lassen sich verändern und erweitern. Auch das Inbeziehungsetzen eines Items ist mittels *Annotate* möglich. Hier können aber nur Standardbeziehungen ausgewählt werden (Abbildung 16). Eine neue Annotation wird immer mit dem Typ *has topic* angelegt.

Abbildung 16: PSEW Annotieren

Zur Erstellung spezieller Beziehungen sowie weiterer Eigenschaften ist die Funktion *add property* vorgesehen. Beim Eintragen eines Begriffes in das Textfeld unterhalb von *add property* prüft NEPOMUK, ob Attribute, welche mit den eingegebenen Buchstaben beginnen, schon existieren und bietet diese zur Auswahl an. Ist dies nicht der Fall, öffnet sich nach der Auswahl von *create new* der in Abbildung 17 zu sehende Dialog, um eine neue Eigenschaft zu erstellen.

Als Datentyp (*possible Value*) lassen sich *a Thing*, *Text*, *Integer* und *Date and Time* auswählen (Abbildung 18). *A Thing* ist jedes in NEPOMUK definierte Element, dass sich in irgend einer Weise von der Klasse *Thing* ableitet. Zusätzlich ist ein Beziehungstyp verfeinernd anzugeben (Abbildung 19). Weiterhin ist der Name der anzulegenden inversen Beziehung zu definieren. Der Editor belegt dieses Feld standardmäßig mit *inverse of <property name>* vor.

Abbildung 17: PSEW
Property hinzufügen

Abbildung 18: PSEW
Property value

Abbildung 19: PSEW
Beziehungstyp

Bei den anderen drei Datentypen *Text*, *Integer* und *Date and Time* können keine Beziehungstypen gewählt werden.

Soll die Verwendbarkeit eines Attributes eingeschränkt werden, ist die *Domain* desselben zu ändern. Hierzu wird das Attribut mittels seines URI aufgerufen und kann dann editiert werden. Generell wird ein Attribut beim Erstellen der *Domain Class* zugeordnet. Durch das Eintragen des URI der neuen Domain (zum Beispiel *Vorlesung*) und dem Löschen der alten Domain ist das so geänderte Attribut nur noch in Elementen verwendbar, die der Subklasse *Vorlesung* angehören. Abbildung 20 zeigt diese Änderung.

Abbildung 20: PSEW Property Domain ändern

Von der angelegten Klasse *Vorlesung* kann nun eine Instanz angelegt werden. Die bereits definierten Attribute, deren Domain, wie oben beschrieben, von Class auf ein anderes in der entsprechenden Hierarchie befindliches Item geändert wurde, werden direkt angezeigt. Andere Attribute können wie oben beschrieben hinzugefügt oder neu angelegt werden. Zu den Attributen lassen sich je nach Datentyp Werte zuordnen. Für *Text* und *Number* ist dies direkt möglich. Bei *Number* erfolgt bis jetzt allerdings keine Typüberprüfung. Beim Typ *Date* wird ein Kalenderblatt zur Auswahl angezeigt (Abbildung 21). Attribute vom Typ *Thing* lassen sich mit definierten Elementen belegen (Abbildung 22). Es ist auch möglich erst bei der Wertezuordnung neue Instanzen einer bestimmten Klasse anzulegen (Abbildung 23).



Abbildung 21:
NEPOMUK
Kalenderblatt

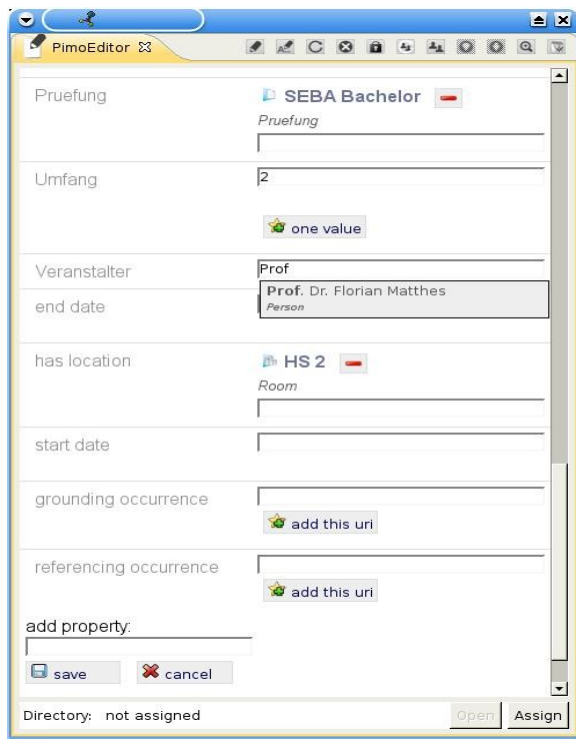


Abbildung 22: PSEW Instanz bearbeiten

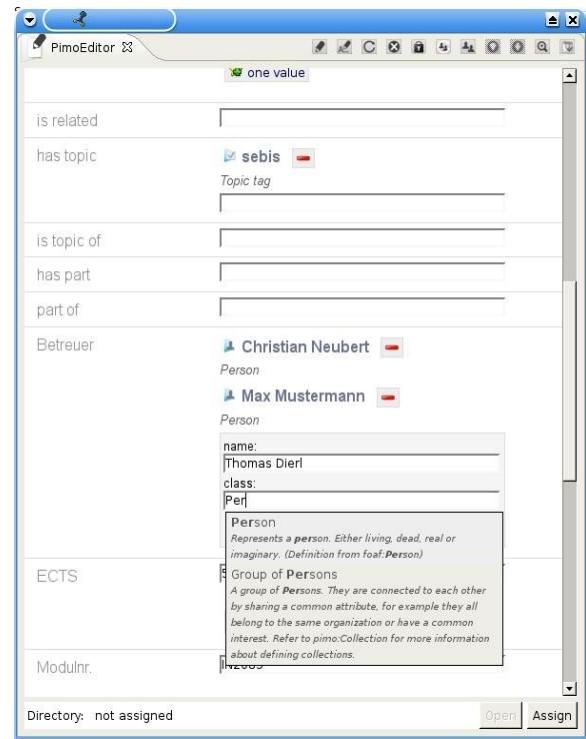


Abbildung 23: PSEW Instanz bearbeiten II

Ohne zusätzlichen Aufwand ist es in NEPOMUK möglich Beziehungen zwischen Items auch von der entgegengesetzten Seite zu betrachten. Dies gelingt mit Hilfe der angelegten inversen Beziehungen. Die Abbildungen 24, 25 und 26 zeigen die inversen Beziehungen zu den in Abbildung 22 und 23 definierten Beziehungen.

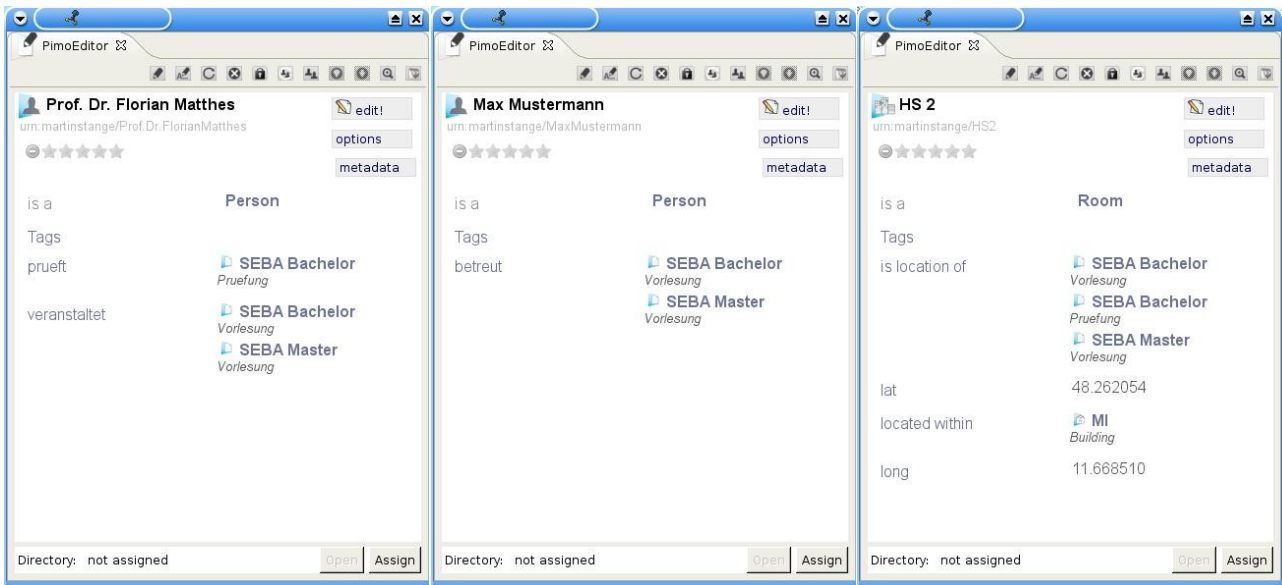


Abbildung 24: PSEW inverse Beziehung I

Abbildung 25: PSEW inverse Beziehungen II

Abbildung 26: PSEW inverse Beziehungen III

2.2.3 Abfragesprache

Zur Suche nach Elementen bietet PSEW eine einfaches Eingabefeld an. Wildcards wie ? für ein einzelnes Zeichen und * für beliebige Zeichen werden akzeptiert. Gefunden werden durch diese Suche aber nur Instanzen und keine Klassen oder Subklassen.

Für eine umfassende und semantische Suche wird die *Simple Protocol and RDF Query Language* (SPARQL) [Wi08b] verwendet. Diese verwendet eine SQL – ähnlichen Syntax. Weitergehende Einführungen in SPARQL bieten unter anderem [Xm08a] und [Be08a]. Entscheidend bei SPARQL ist, dass ein Statement in bei der Bedingungsangabe (WHERE) immer aus drei Elementen *Subject – Prädikat – Objekt* besteht. Das Subjekt ist immer das gesuchte Element, das Prädikat die Beziehungen zwischen Subjekt und Objekt und das Objekt ein weiteres Element. Zur Suche werden grundsätzlich die URI der Elemente und Beziehungen verwendet. Zur einfacheren Schreibweise lassen sich für die normalerweise recht langen URIs Präfixe einsetzen. In Abbildung 27 ist eine solche Abfrage sowie deren Ergebnis abgebildet.

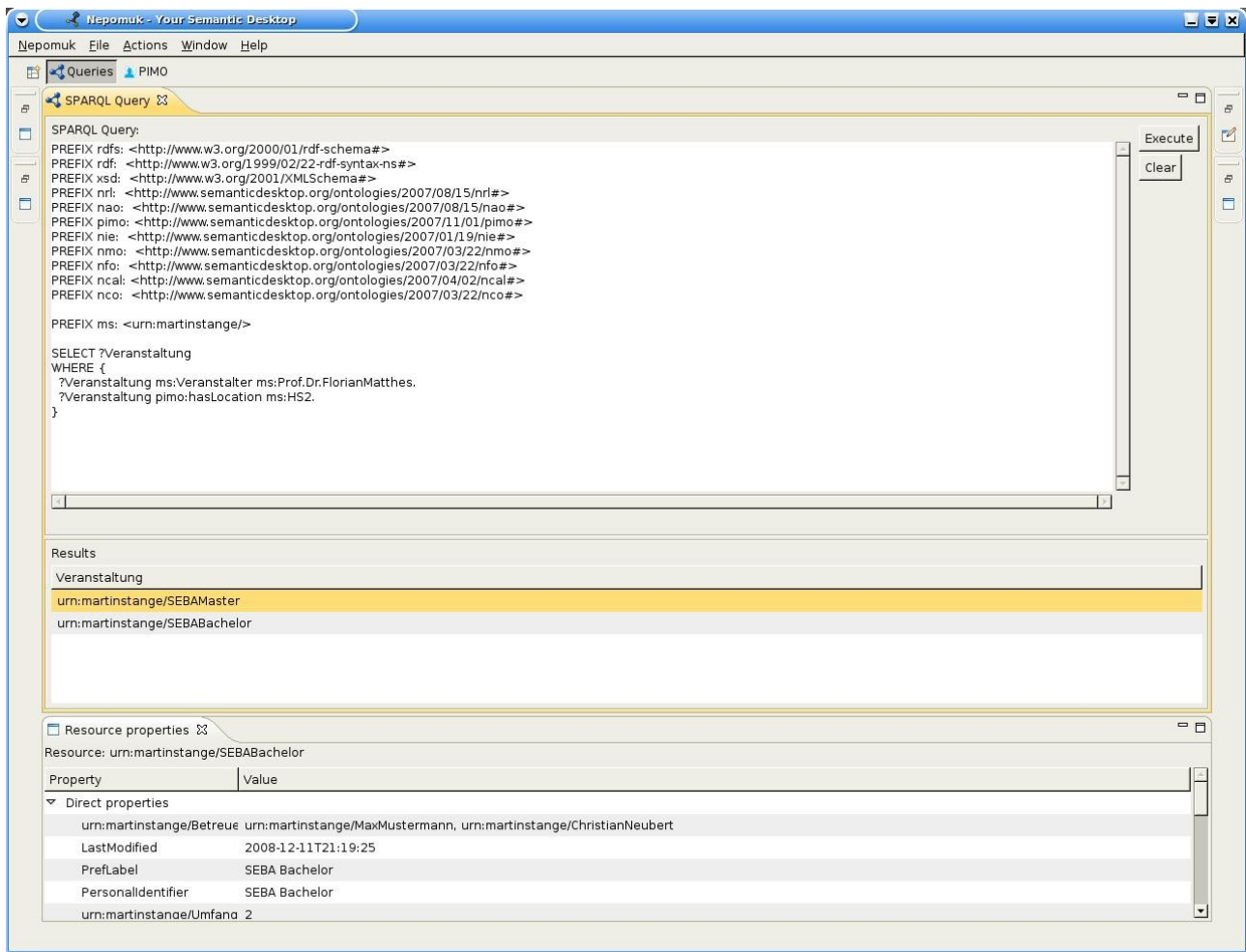


Abbildung 27: PSEW semantische Abfrage

2.2.4 Export

Der Export einzelner Elemente aus dem PSEW heraus in verschiedenste Formate ist möglich. Hierzu zählen RDF/XML, Turtle, N-triples, Trix und Trig. Über die Serverkomponente lassen sich sowohl Abfragen erstellen als auch das ganze Repository in die genannten Formate exportieren (Abbildung 28).

Construct query

Query language sparql

Query

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX nrl: <http://www.semanticdesktop.org/ontologies/2007/08/15/nrl#>
PREFIX nao: <http://www.semanticdesktop.org/ontologies/2007/08/15/nao#>
PREFIX pimo: <http://www.semanticdesktop.org/ontologies/2007/11/01/pimo#>
PREFIX nie: <http://www.semanticdesktop.org/ontologies/2007/01/19/nie#>
PREFIX nmo: <http://www.semanticdesktop.org/ontologies/2007/03/22/nmo#>
PREFIX nfo: <http://www.semanticdesktop.org/ontologies/2007/03/22/nfo#>
PREFIX ncal: <http://www.semanticdesktop.org/ontologies/2007/04/02/ncal#>
PREFIX nco: <http://www.semanticdesktop.org/ontologies/2007/03/22/nco#>
```

RDF format rdxml

Fulltext query

Fulltext querying means that a Lucene Fulltext query is embedded into a SPARQL query. Here is an example that searches for "person". Fulltext search is still under construction. Read more about [LuceneSail at the dev wiki](#).

Query language sparql

Query

```
PREFIX search: <http://www.openrdf.org/contrib/lucenesail#>
SELECT ?x ?score ?snippet WHERE { ?x search:matches ?match.
?match search:query "person";
search:score ?score;
search:snippet ?snippet. }
```

run

Dump repository

Dump the contents of this repository as RDF stream. Opens a new browser window.

RDF format rdxml

☐ return content with browser-friendly text/plain content-type.

dump

Abbildung 28: NEPOMUK Export

2.2.5 Import

Über die Serverkomponente lassen sich Daten in den beim Export aufgezählten Formaten importieren. Der Import kann sowohl aus einer Datei, von einer URL oder auch als direkte Eingabe erfolgen (Abbildung 29).

Add graph from a file

File to add Browse...

Base URI

RDF format rdxml

Named graph (opt) rdxml

Add graph from a URL

URL to add

Base URI

RDF format rdxml

Named graph (opt)

add web document

Add RDF from clipboard

RDF to add

Base URI

RDF format rdxml

Named graph (opt)

add RDF

Abbildung 29: NEPOMUK Import

2.2.6 Berechtigungskonzept

Ein explizites Berechtigungskonzept ist bei PSEW nicht notwendig und somit auch nicht vorhanden, da es sich um eine Desktopanwendung handelt. Der Benutzer kann in einem Kollaborationsnetz selbst festlegen, welchen Personen oder Gruppen er welche Teile seiner Elemente zur Verfügung stellt.

2.2.7 Useability

Für den hier beschriebenen semantischen Desktop ist ein Gesamtverständnis von Ontologien Voraussetzung zur erfolgreichen und adäquaten Handhabung. Aufgrund der guten Gliederung und der vorhandenen Strukturen lässt sich PSEW nach ein kurzen Eingewöhnung schnell nutzen. Einige Aktionen, wie zum Beispiel das Bearbeiten von Properties gestalten sich etwas umständlich. Auch die Ladezeiten der gesamten Anwendung und größerer Abfragen sind relativ hoch. Da es sich bei PSEW aber noch um eine in der Entwicklung befindlichen Anwendung handelt, ist von der Behebung der Unzulänglichkeiten auszugehen. Eine detaillierte Anleitung zur Nutzung von PSEW stellt [Le08] bereit.

2.3 Hypertext Knowledge Workbench

Bei der Hypertext Knowledge Workbench (HKW) handelt es sich wie bei dem semantischen Desktop NEPOMUK um ein „Werkzeug für das persönliche Wissensmanagement“ [Sw08e] in Form einer Webanwendung. Ziel der Anwendung ist es „über den Umweg eines Werkzeugs mit sich selbst [zu kommunizieren] um so kognitive Grenzen (Langzeitgedächtnis, Kurzzeitgedächtnis) zu verschieben“ [Sw08e]. Die vom Forschungszentrum Informatik (FZI) der Universität Karlsruhe (www.fzi.de) entwickelte HKW, verwendet zur Datenstrukturierung genauso wie NEPOMUK die *Conceptual Data Structures* (CDS) [Sw08d]. Diese Gemeinsamkeit ist nicht zufällig, da das FZI auch an der Entwicklung von NEPOMUK beteiligt war.

2.3.1 Steckbrief

Status:	unbekannt
Version:	0.0.7
Veröffentlichungsdatum:	2008-09-24
Lizenzmodell:	BSD
Homepage:	http://semweb4j.org/site/cds.gwt
Bearbeitungsmodell:	Wikisyntax
Programmiersprache:	Java und GWT
Annotierbare Elemente:	Dokument
Unterscheidung zwischen Wiki-Seiten und Konzepten:	Nein
Annotationstyp:	Seite
Kontextinformationen über Annotationen:	Ja
Instanzenimport:	Ja
Ontologieimport:	Ja
Instanzenexport:	Ja
Ontologieexport:	Ja

[Sw08d]

2.3.2 Strukturierung

Wie in der Einleitung schon angedeutet, verwendet die HKW zur Datenstrukturierung die *Conceptual Data Structures* (CDS). Allerdings wird in [Vö08c] eine andere Version der CDS als Datenmodell der HKW beschrieben. Nach Aussage von Max Völkel in einer Antwort auf eine E-Mail-Anfrage handelt es sich dabei um eine neuere Version der CDS, die sowohl in der HKW als auch in NEPOMUK verwendet wird. Die vorliegende Dokumentation NEPOMUKs [Vö08a] erwähnt allerdings noch ein älteres Modell. Auf die erfolgten Änderungen wird im Folgenden eingegangen.

Die CDS besteht aus dem *CDS Data Model* und der *CDS Relation Hierarchy*. Das *CDS Data Model* ist der Nachfolger des bei NEPOMUK vorgestellten *Semantic Web Content Model* (SWCM). Die *CDS Relation Hierarchy* ist identisch mit der bei NEPOMUK beschriebenen *CDS Ontology*.

2.3.2.1 CDS Data Model

Der Unterschied zum beim NEPOMUK vorgestellten SWCM besteht aus den zwei neuen Elementen *ContentItem* und *Triple* sowie einigen veränderten Beziehungen der Elemente untereinander [Vö08c]. In Abbildung 30 ist ein UML-Klassendiagramm des CDS Data Model dargestellt.

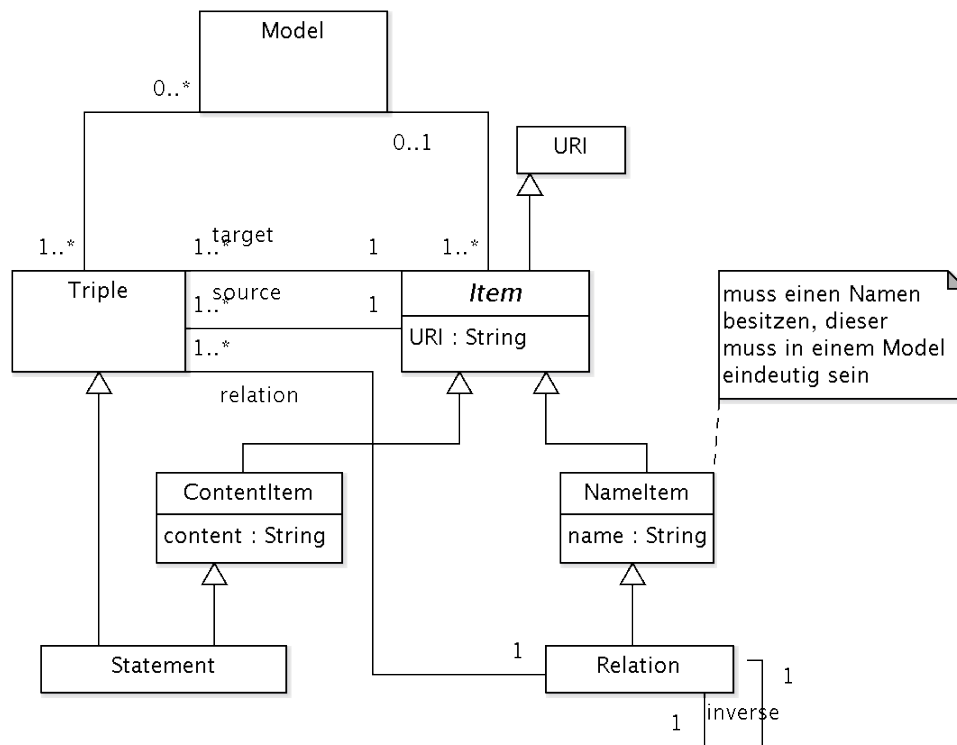


Abbildung 30: HKW CDS Datenmodell in Anlehnung an [Vö08c Fig.1]

2.3.2.1.1 ContentItem

Bei einem *ContentItem* handelt es sich um einen Textabschnitt, vergleichbar mit dem Inhalt einer Wiki-Seite. Der Inhalt ist frei wählbar, kann verschieden lang sein und in Wiki-Syntax erstellt werden. Erstellungsdatum, Änderungszeitpunkt und Autor werden vom System automatisch aufgezeichnet. Zur eindeutigen Referenzierbarkeit besitzt jedes *ContentItem* einen eigenen und sich nicht ändernden Uniform Resource Identifier (URI). [Vö08c S.4f]

2.3.2.1.2 Triple

Ein *Triple* ist vergleichbar mit einem semantischen Link in einem Wiki. Es erzeugt eine Verbindung zwischen zwei *Items* und beschreibt diese mit einer *Relation*. Da es sich bei Triples nicht um Items handelt, besitzen sie keine Metadaten. Es ist daher notwendig ein *Triple* zu einem *Statement* zu erweitern um diesem Metadaten und Annotationen hinzufügen zu können. [Vö08c S.5]

Durch die zwei hinzugefügten Elemente *Triple* und *ContentItem* ergeben sich auch für die anderen Bestandteile des CDS Data Model Änderungen.

2.3.2.1.3 Model

Ein *Model* besteht nun nicht nur aus Items sondern enthält zusätzlich Triples. Es bleibt weiterhin der Container für alle Elemente. Jedes *Model* besitzt einen eindeutige URI und wird in RDF durch einen benannten Graph repräsentiert. [Vö08c S.4]

2.3.2.1.4 Item

Ein *Item* ist eine abstrakte, nicht instantiierbare Klasse, die von *ContentItem* und *NameItem* realisiert wird.

2.3.2.1.5 NameItem

Bei einem NameItem handelt es sich um ein Element mit einem Namen, wie beispielsweise der Name einer Wiki-Seite oder eines Ordners im Dateisystem. Es besitzt außer seinem Namen keinen weiteren Inhalt. Ein NameItem muss genau einen Inhalt (den Namen) besitzen. Dieser Inhalt darf nur aus einfachem Text bestehen, Zeilenumbrüche oder Wiki-Syntax sind nicht zulässig. Die Eindeutigkeit des Inhalts innerhalb eines Modells muss gewährleistet sein. Es ist daher nicht zulässig, dass NameItems mit dem selben Inhalt aber unterschiedlichen URIs existieren. All diese Bedingungen gelten für ContentItems nicht. [Vö08c S.5f]

2.3.2.1.6 Relation

Durch die Einführung des Triple erläutert die Relation nicht mehr das Statement sondern ein Triple näher. [Vö08c S.6]

2.3.2.1.7 Statement

Ein Statement ist sowohl ein Triple als auch ein ContentItem. Es besitzt somit Metadaten, lässt sich annotieren und kann über Textinhalte verfügen. [Vö08c S.6]

2.3.2.2 CDS Relation Hierarchy

Die CDS Relation Hierarchy ist identisch mit der bei NEPOMUK beschriebenen CDS Ontology. Deshalb wird hier auf eine erneute Beschreibung verzichtet.

2.3.2.3 Beispiel

Im Folgenden wird in der Hypertext Knowledge Workbench das schon bekannte Beispiel der Vorlesung modelliert. Es gibt bei der HKW außer den Elementen und Beziehungen des CDS Data Model und der CDS Relation Hierarchy keine weiteren vorgegebenen Strukturen. Es liegt daher in der Hand des Anwenders den darzustellenden Sachverhalt auf geeignete Weise abzubilden.

In Abbildung 31 ist das grundsätzliche Layout der HKW dargestellt. In dem grünen Kasten ist das aktuelle Item (hier RootItem) zu sehen, in den weißen und blauen Kästen die unterschiedlichen vordefinierten Beziehungstypen. Beim *RootItem* handelt es sich um eine *NameItem*, was an dem gelb unterlegten *N* links neben *Edit* zu erkennen ist. Mit Hilfe des Eingabefeldes *Create/Go to:* können existierende Elemente aufgerufen und neue Items angelegt werden.

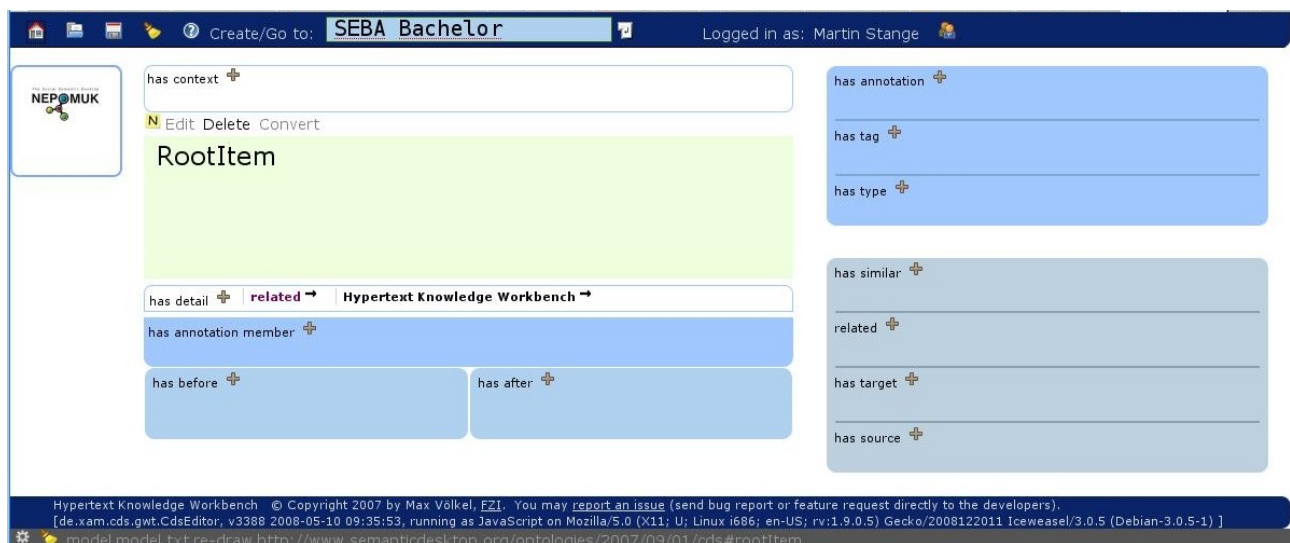


Abbildung 31: HKW Layout und NameItem

Mit der Schaltfläche *Convert* lassen sich NameItems und ContentItems ineinander umwandeln. Ein ContentItem, wie in Abbildung 32 zu sehen, ist an dem gelben Viereck (es stellt einen Notizzettel dar) neben dem *Edit* Button zu erkennen.

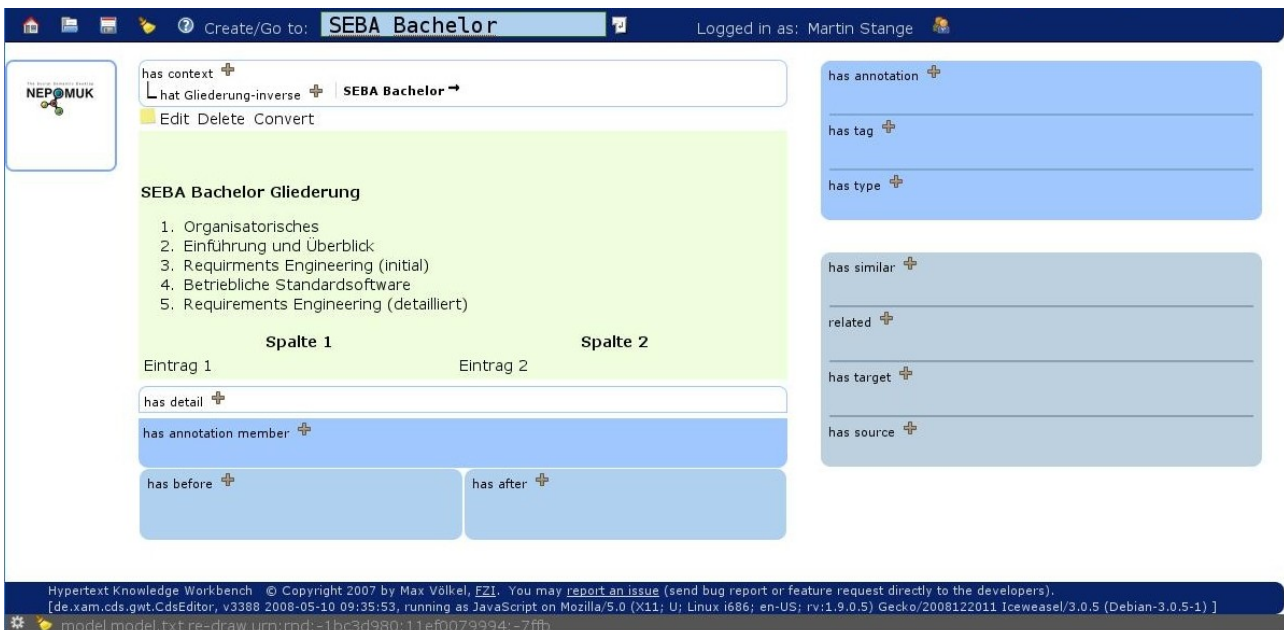


Abbildung 32: HKW ContentItem

Zur Bearbeitung des Inhalts eines ContentItems steht ein einfacher Editor zur Verfügung. Hier kann die in [Vö08a S.63ff] beschriebene Wiki-Syntax zur Aufbereitung des Inhalts verwendet werden (Abbildung 33).

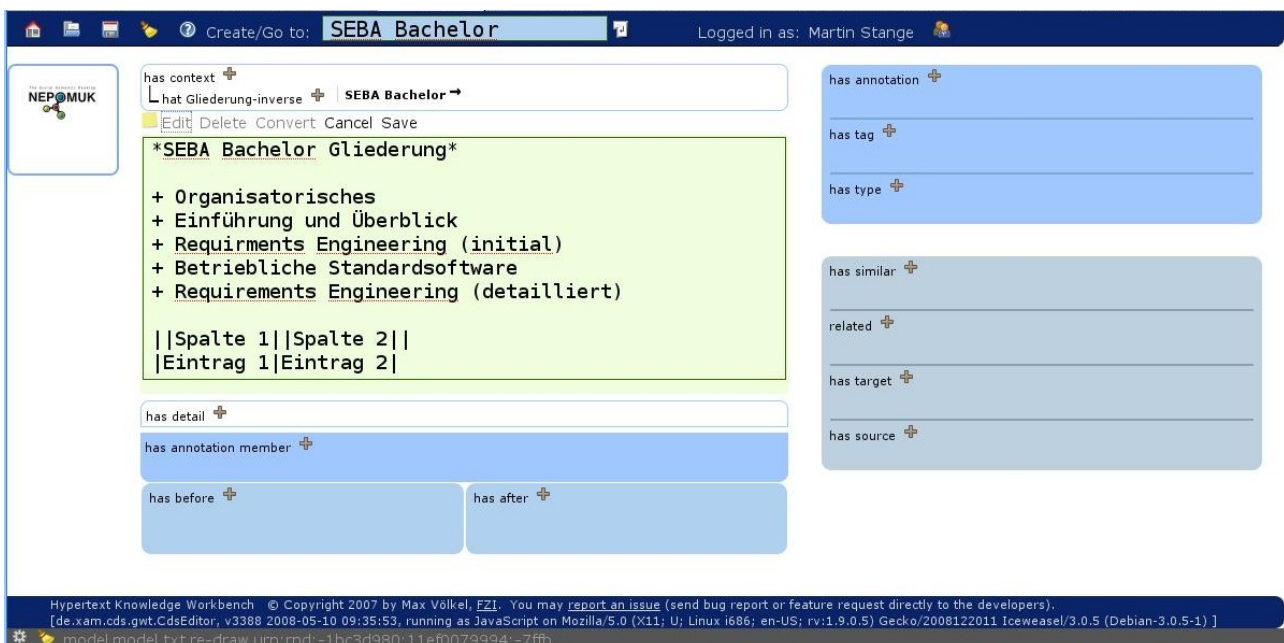


Abbildung 33: HKW ContentItem bearbeiten

Zum Anlegen eines Statements, sprich der Beziehung zwischen zwei Items, bedarf es der Auswahl des hinter den abgebildeten Beziehungen stehenden +. Die Auswahl einer vordefinierten Beziehung ist entscheidend, da neu erstellte Beziehungen von der gewählten abgeleitet werden. Den

entsprechende Dialog zum Anlegen eines Statements zeigt Abbildung 34. Im ersten Eingabefeld kann eine bestehende Relation ausgewählt werden. Ist eine neue Relation zu erstellen, wird diese einfach eingegeben. Mit der Erstellung des Statements werden dann die Relation sowie die zugehörige inverse Relation mit der Bezeichnung *<relationname>-inverse* angelegt. Der Name einer Relation lässt sich im Nachhinein ändern. Das zweite Eingabefeld dient der Angabe des zu verknüpfenden Items. Auch hier kann entweder ein bestehendes gewählt oder ein neues angelegt werden. Ein neues NameItem wird in der Regel dann angelegt, wenn die Eingabe einzeilig erfolgt und der Name noch nicht vergeben ist. Der Hintergrund des Eingabefeldes bleibt weiß. Wird ein Zeilenumbruch eingefügt, vergrößert sich das Eingabefeld und die Hintergrundfarbe wechselt von weiß nach grün. Das neue Item wird als ContentItem angelegt.

Abbildung 34: HKW Statement anlegen

Das angelegte Statement, zu erkennen an dem gelb hinterlegten Pfeil neben dem *Edit* Button, kann nachträglich verändert werden. In Abbildung 35 ist auch zu erkennen, dass ein Statement mit den gleichen Beziehungen und Annotationen versehen werden kann, wie jedes andere Item auch.

Abbildung 35: HKW Statement

Auch Relationen, gekennzeichnet durch einen roten Pfeil, lassen sich nachträglich verändern und bearbeiten. Wie Abbildung 36 zeigt, wird auch immer die inverse Relation mit angegeben.

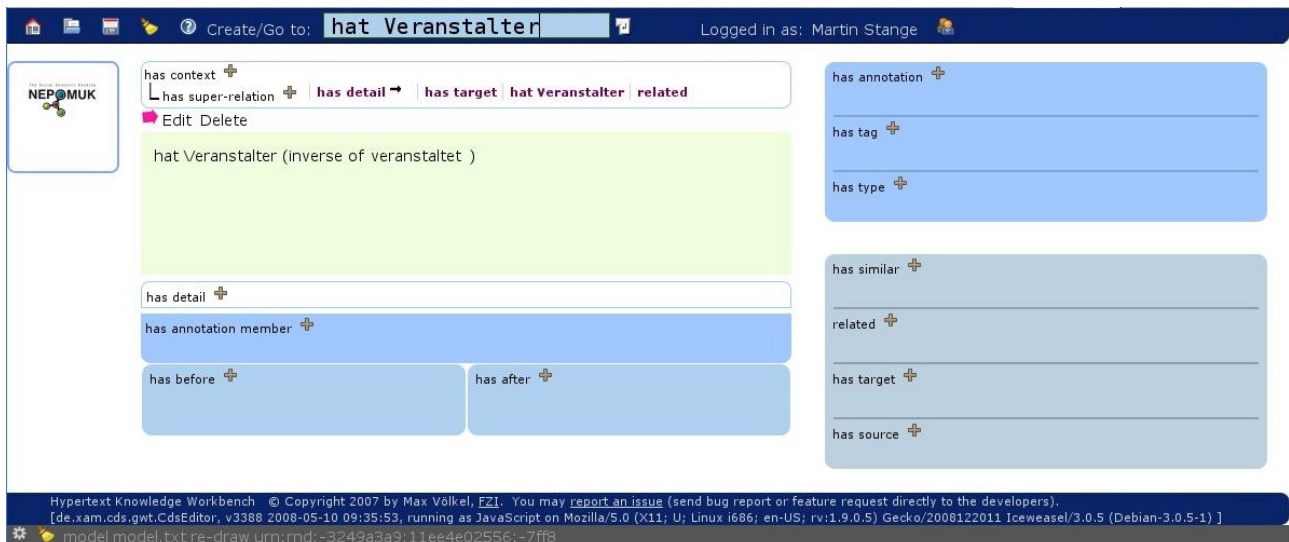


Abbildung 36: HKW Relation

Ein Beispiel für eine bearbeitete und mit Attributen versehene Vorlesung ist in Abbildung 37 zu sehen.

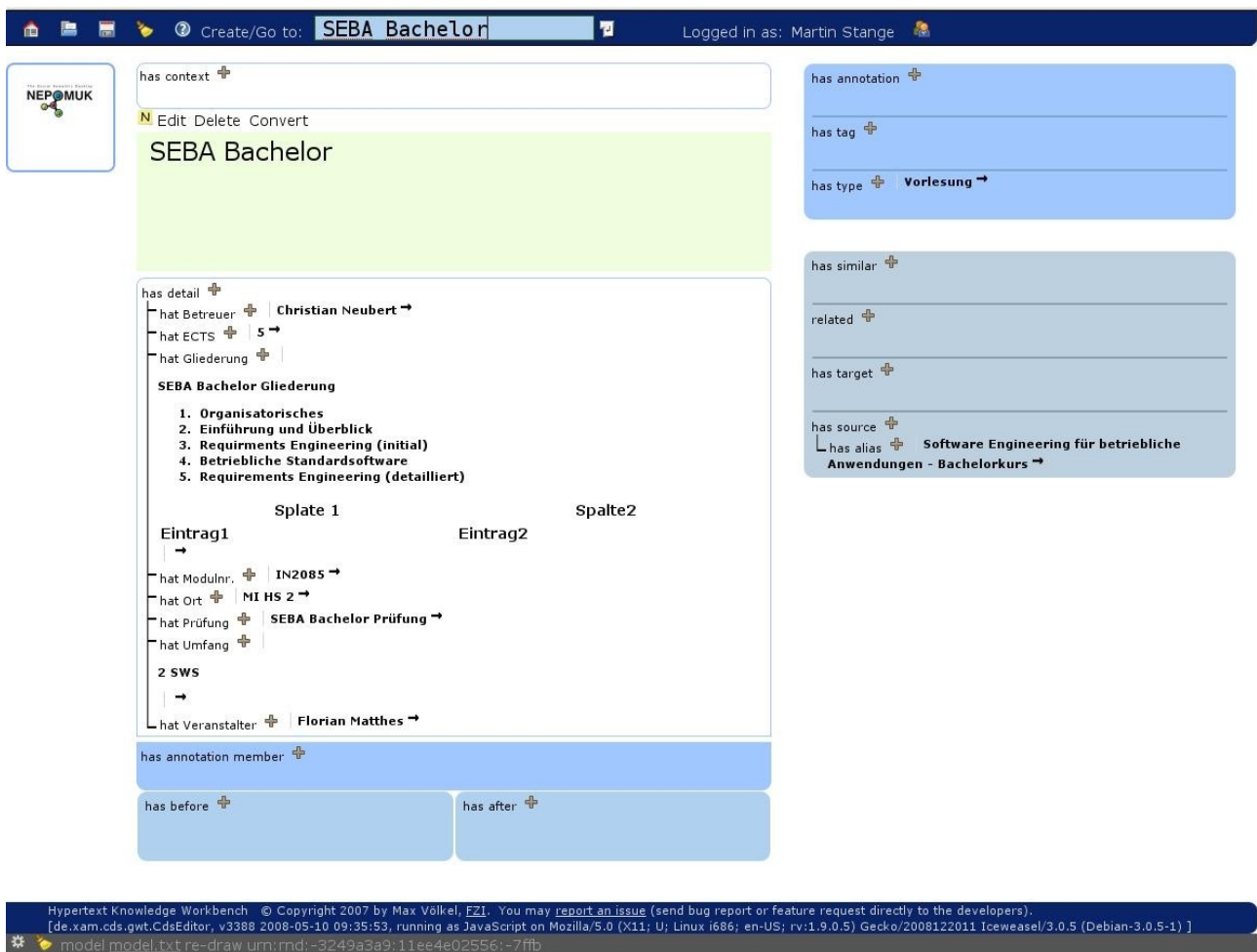


Abbildung 37: HKW NameItem Vorlesung

2.3.3 Abfragesprache

Eine Abfragemöglichkeit ist in der Oberfläche bisher nicht vorgesehen. Mit Hilfe des Eingabefeldes *Create/Go to* werden durch die Eingabe einer Zeichenkette alle Elemente aufgelistet, welche diesen Suchbegriff enthalten. Das gesuchte Element kann dann ausgewählt und aufgerufen werden. Die direkten Beziehungen eines Items zu anderen sind zwar immer sichtbar, komplexere Zusammenhänge können ohne eine Abfragemöglichkeit aber nicht dargestellt werden.

2.3.4 Export

In der Weboberfläche konnte eine direkte Exportfunktionalität nicht aufgefunden werden. Das Datenmodell sowie die enthaltenen Elemente lassen sich dennoch mit Hilfe eines Links anzeigen. Die Ausgabe erfolgt in *Notation3* (N3), einer von Tim Bernes-Lee entwickelten Sprache für das Semantische Web [Wi08i]. Die Spezifikation zu N3 sowie ein Tutorial sind unter [Be06a] und [Be04a] zu finden. Die Oberfläche stellt weiterhin die Möglichkeit zur Speicherung eines Modells zur Verfügung. In der aktuellen Version ist diese Funktion jedoch nicht nutzbar, da zwar ein Dateiname für das zu speichernde Modell angegeben werden kann, eine solche Datei nach dem Abspeichern aber nicht vorhanden ist.

2.3.5 Import

[Sw08d] weist für die HKW Importfunktionalität aus. Es existiert in der Webanwendung eine Funktion zum Laden eines Modells, welche allerdings auf Grund des nicht funktionierenden Exports nicht verifiziert werden konnte.

2.3.6 Berechtigungskonzept

Ein Berechtigungskonzept ist nicht Bestandteil der HKW. Beim Aufruf der Webanwendung wird zwar ein Benutzername abgefragt, der allerdings frei gewählt werden kann und nur zur Dokumentation der Veränderungen dient. Ein Passwort wird nicht verlangt. Dem Benutzer können auch keine Rechte oder Rollen zugeordnet werden. Da es sich um eine Anwendung zum persönlichen Wissensmanagement handelt, wird auf ein Berechtigungskonzept bewusst verzichtet.

2.3.7 Useability

Die Benutzung der HWK ist zu Beginn gewöhnungsbedürftig. Eine gute Kenntnis der Struktur mit NameItems, ContentItems und Statements sowie der Beziehungshierarchie ist sehr vorteilhaft. In Anbetracht der Tatsache, dass sich die HKW in einem sehr frühen Entwicklungsstadium befindet, ist für die Zukunft eine interessante Anwendung zu erwarten.

2.4 AceWiki

Im Gegensatz zu den meisten semantischen Wikis nutzt der von AceWiki [Ac08a] gewählte Ansatz zur Beschreibung semantischer Inhalte kontrollierte natürliche Sprache, genauer das Attempto Controlled English (ACE) [At08a], um Informationen abzubilden und nutzbar zu machen. Ziel des AceWiki ist es normale Menschen ohne Kenntnis formaler Sprachen Begriffsgeflechte in intuitiver und gemeinschaftlicher Weise erstellen zu lassen ohne dazu eine spezielle Anwendung zu installieren. „The goal of AceWiki is to enable ordinary people with no background in formal languages to create expressive ontologies in a collaborative and intuitive way without the need of installing an application.“ [Ku08c S.1]

2.4.1 Steckbrief

Status:	alpha
Version:	0.2.11 alpha
Veröffentlichungsdatum:	2008-12-22
Lizenzmodell:	GNU Lesser General Public
Homepage:	http://attempto.ifi.uzh.ch/acewiki/
Bearbeitungsmodell:	Attempto Controlled English (ACE)
Programmiersprache:	Java
Annotierbare Elemente:	Dokument
Unterscheidung zwischen Wiki-Seiten und Konzepten:	Ja
Annotationstyp:	Satz
Kontextinformationen über Annotationen:	Nein
Instanzenimport:	Nein
Ontologieimport:	Nein
Instanzenexport:	Ja
Ontologieexport:	Ja

[Sw08c]

2.4.2 Strukturierung

Da AceWiki zur Darstellung und Verknüpfung von Sachverhalten ein im Umfang reduziertes Englisch verwendet, sind klassische Strukturelemente semantischer Wikis hier nicht vordergründig zu finden. Vielmehr dienen die Elemente natürlicher Sprachen zur Strukturierung. Zu nennen sind hier Eigennamen (*proper names*), Substantive (*nouns*), Verben (*transitive verbs*, *of-constructs*), Adjektive (*transitive adjectives*) sowie vordefinierte Funktionswörter [Ku08a S.2] [Ku08c S.1]. Sie bilden zusammen Sätze, die im AceWiki sowohl von menschlichen Benutzern als auch von Wiki-Systemen verstanden werden.

2.4.2.1 Nominalphrase

Hierbei handelt es sich um eine Gruppe von Wörtern deren Kern ein Substantiv (z.B. die interessante Vorlesung) bzw. ein Stellvertreter (z.B. sie) für ein Substantiv ist [Wi08c]. Unterschieden werden können zählbare Nominalphrasen in Einzahl (*a/the/1/one lesson*) und Mehrzahl (*3/three lessons*) sowie nicht zählbare Nominalphrasen (*some water*). Dieser Gruppe gehören auch Eigennamen (*Germany*), ganze und reelle Zahlen, arithmetische Ausdrücke, Zeichenketten sowie Mengen und Listen an. Des weiteren sind hier Pronomen (*he, she, it*), Reflexivpronomen (*himself, herself, itself*), Indefinitpronomen (*somebody, nobody*), begrenzende Angaben (*at least, less than*), Maßeinheiten (*kg, m*) und Variablen vertreten. Die Verwendung von Variablen ist dabei ebenso möglich wie die Erstellung von Genitivkonstrukten (*a customer of John*,

John's customer). [At08b] Die hier genannten Sprachkonstrukte stellen den Funktionsumfang von ACE dar. AceWiki verwirklicht davon bisher nur eine Teilmenge. Einen Vergleich der Funktionsumfänge von ACE und AceWiki ist in Tabelle 1 auf Seite 38 zu sehen. Abbildung 38 und Abbildung 39 zeigen die Erstellungsdialoge für Substantive und Eigennamen.

Abbildung 38: AceWiki create Noun

Abbildung 39: AceWiki create Proper Name

Die Wiki-Seite eines Substantivs teilt sich in die Kategorien *Article*, *Noun*, *References*, *Individuals* und *Hierarchy*. Unter *Article* sind alle erstellten Sätze zu sehen (Abbildung 40). Diese können hier verändert, gelöscht und detaillierter betrachtet werden. Zudem ist das Hinzufügen neuer Sachverhalte möglich. *Noun* präsentiert die Definition des Substantivs, welche sich an dieser Stelle auch verändern und löschen lässt. Alle Sätze, in denen das Substantiv verwendet wird, erscheinen unter *References* (Abbildung 41). Die Kategorie *Individuals* stellt alle Objekte (Eigennamen) dar, die diesem Substantiv zugeordnet wurden (Abbildung 43). *Hierarchy* zeigt die Substantive der nächsthöheren und der nächsttieferen Hierarchieebene an, die mittels der Aussage *Every A is a B* definiert werden (Abbildung 42).

Abbildung 40: AceWiki Noun Article

Abbildung 41: AceWiki Noun References

Abbildung 42: AceWiki Noun Hierarchy

Article, *Proper Name*, *References* und *Assignments* sind die Kategorien einer Wiki-Seite für Eigennamen. *Article* und *References* zeigen die gleichen Information wie bei den Substantiven. *Proper Name* zeigt die Definition des Eigennamens. *Assignments* zeigt alle Substantive, dem dieser Eigenname zugeordnet ist (Abbildung 44). Es ist also die Umkehrung der *Individuals* -Darstellung.

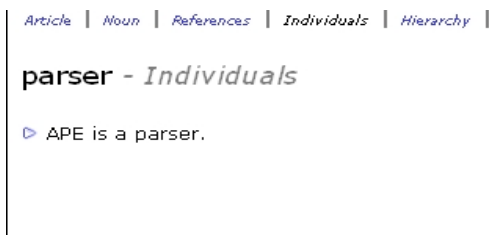


Abbildung 43: AceWiki Noun Individuals

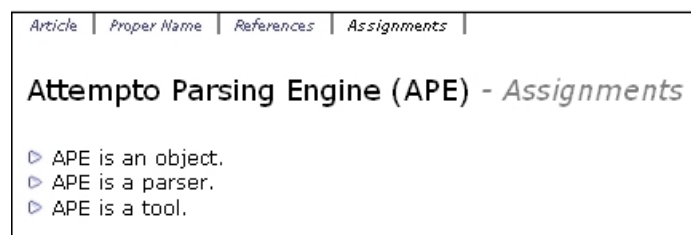


Abbildung 44: AceWiki Proper Name Assignments

2.4.2.2 Adjektive

Nominalphrasen können durch Adjektive näher bestimmt werden. Eine Verkettung von Adjektiven mit *and* ist möglich. Einem Substantiv können Bedingungen folgen (*a customer who is rich*) welche auch mittels *and* verkettet werden können. [At08b] Von AceWiki werden bisher aber nur transitive Adjektive unterstützt. Sie stellen Beziehungen zwischen Nominalphrasen her und bestehen aus einem Adjektiv sowie einer Präposition. Den Erstellungsdialog für transitive Adjektive zeigt Abbildung 45.

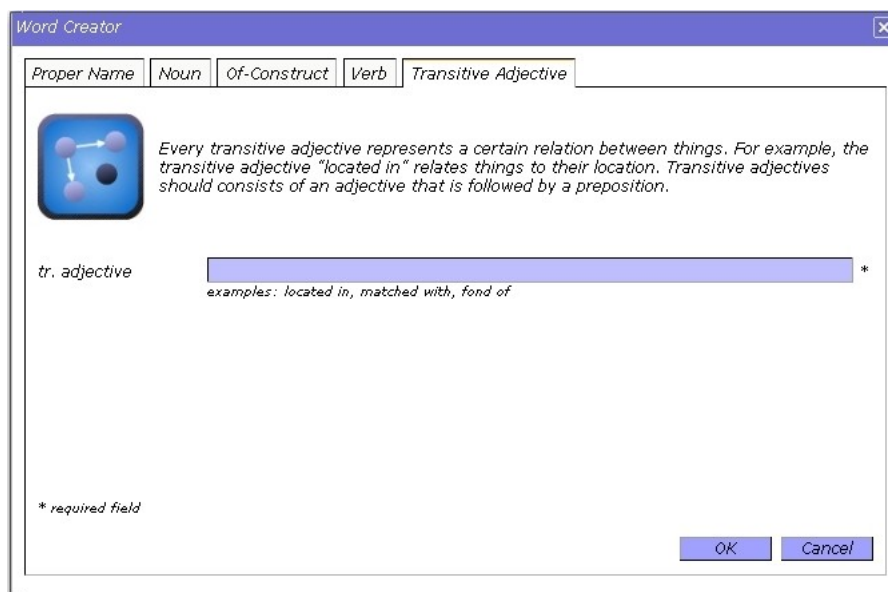


Abbildung 45: AceWiki create Transitive Adjective

Die Wiki-Seite zu einem Adjektiv enthält die Kategorien *Article* (Abbildung 46), *Transitive Adjective* (Abbildung 48) und *References* (Abbildung 47). Diese enthalten die Beschreibung des Adjektivs, die Definition und alle das Adjektiv enthaltenden Sätze.

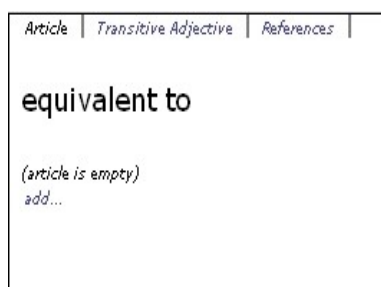


Abbildung 46: AceWiki Adjektiv Article

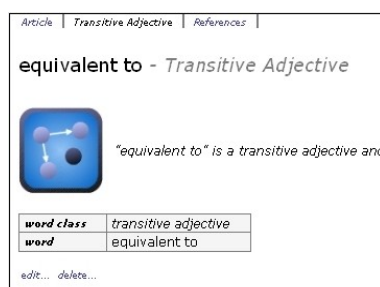


Abbildung 48: AceWiki Adjektiv Trans. Adjective

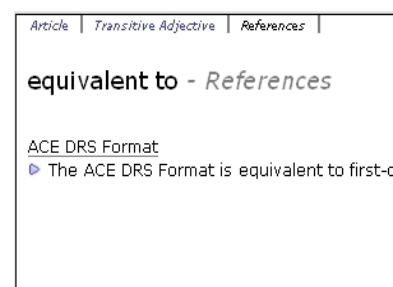


Abbildung 47: AceWiki Adjektiv References

2.4.2.3 Verben

Verben werden bei ACE immer mit der dritten Person (Singular oder Plural) benutzt. Sie müssen bezeichnend sein, in der Gegenwartsform (Simple Present) stehen und können sowohl im Aktiv als auch im Passiv verwendet werden. Unterteilt werden Verben in transitiv [Wi08d], intransitiv [Wi08e] und ditransitiv [Wi08f]. Dies wird benötigt um die Äquivalenz aktiver und passiver Aussagen nachzuweisen oder zu negieren. [At08b]

Das Bindewort *to be* besitzt eine Sonderstellung. Es kann vor Nominalphrasen (*John is a rich customer.*), Adjektiven (*John is rich.*) und Präpositionen (*John is in a garden.*) stehen. Vergleiche sind mit Hilfe der Konstrukte *as ... as* (*John is as rich as Mary.*) und *more ... than* (*John is more fond-of Mary than Bill.*) möglich und in ACE vorgesehen, werden aber bisher von AceWiki nicht zur Auswahl angeboten. Zur Negation der Aussage eines Verbs dienen die Konstrukte *does/do not* (*John does not enter a code.*) und *is/are not* (*The men are not tired.*). Das Ausbleiben einer Handlung oder eines Zustandes drückt *does/do/is/are not provably* (*The men do not provably work.*) aus. Möglichkeiten lassen sich mittels *can/cannot/can not/can't* (*John can wait.*) und Notwendigkeiten respektive Verbote mit Hilfe von *must/have to/does not have to* (*John has to wait.*) ausdrücken. Die letztgenannten Konstrukte sind zwar Bestandteil der ACE Definition werden aber von AceWiki bisher nicht unterstützt. Verben lassen sich mit *and* und *or* verbinden. [At08b] In Abbildung 49 ist die Eingabemaske zur Erstellung eines neuen Verbs abgebildet.

Of-constructs bestehen aus eine Substantiv und dem Wort *of*. Sie dienen der Darstellung von Beziehungen zwischen Dingen und werden im Satz mit *to be* verwendet. Abbildung 50 zeigt die entsprechende Eingabemaske.

Abbildung 49: AceWiki create Verb

Abbildung 50: AceWiki create Of-Construct

Mit der Ausnahme von *to be* lassen sich Verben mit Adverbien genauer beschreiben (*A customer manually inserts a card.*). Auch Steigerungsformen der Adverbien können verwendet werden (*A customer waits patiently/waits more patiently/most patiently waits.*). [At08b] Adverbien sind zwar Bestandteil von ACE, werden aber von AceWiki in der vorliegenden Version bisher nicht unterstützt.

Die Kategorien der Wiki-Seiten zu Verben bzw. Of-constructs lauten *Article* (Abbildung 51), *Verb* (Abbildung 52) bzw. *Of-Construct* (Abbildung 53) und *References* (Abbildung 54). Sie enthalten, wie bei den Adjektiven und Substantiven, Beschreibung, Definition und enthaltende Sätze des entsprechenden Wortes.

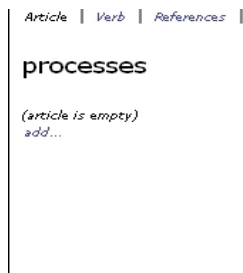


Abbildung 51: AceWiki Verb Article

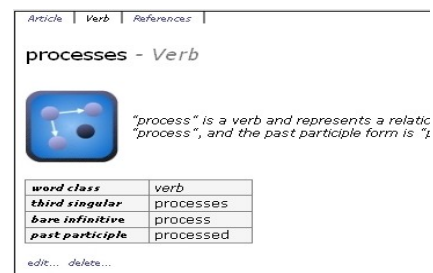


Abbildung 52: AceWiki Verb Verb

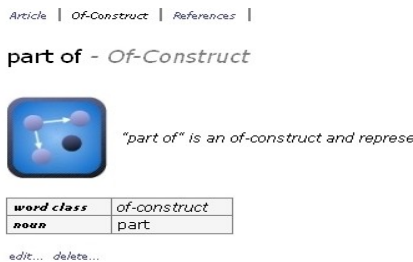


Abbildung 53: AceWiki Of-Construct

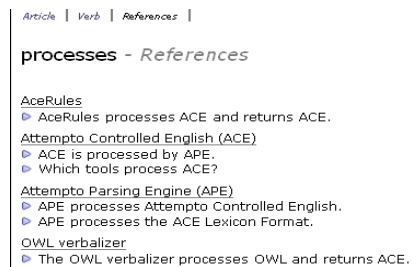


Abbildung 54: AceWiki Verb References

2.4.2.4 Funktionswörter

Zu den vordefinierten Funktionswörter in ACE zählen unter anderem *every*, *if*, *then*, *something*, *and*, *or* und *not*. Diese Worte zusammen mit den oben genannten Wortarten und Phrasen erlauben es intuitiv verständliche Sätze zu bilden, die gleichzeitig ontologische Ausdrücke sind. [Ku08a S.3] Zu Funktionswörtern existieren keine Wiki-Seiten. Sie lassen sich nur direkt im *Sentence Editor* auswählen. Abbildung 55 zeigt alle im Moment verfügbaren Funktionswörter.

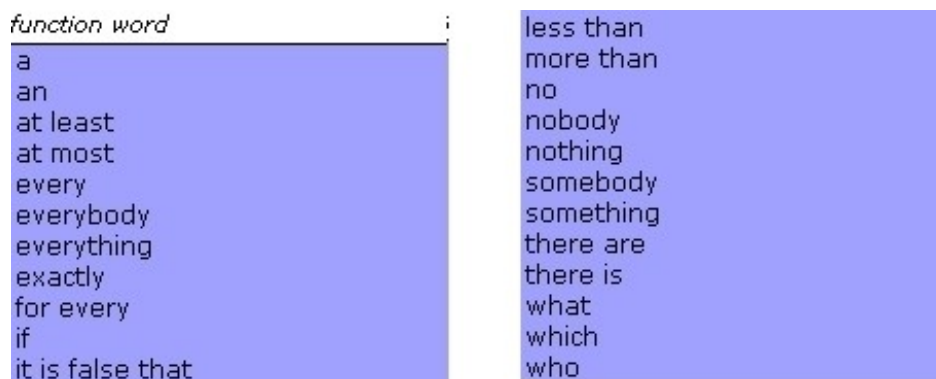


Abbildung 55: AceWiki Funktionswörter

2.4.2.5 Sätze

Ein einfacher Satz hat grundsätzlich die Struktur *Nominalphrase* + *Verb* + *Komplement* + *Ergänzung*. Komplemente sind Objekte (zum Beispiel Nominalphrasen) und Ergänzungen können optionale Adverbien oder Präpositionalkonstrukte sein. [At08b]

Zur Einführung neuer Objekte in den Kontext dient *there is/are* + *Nominalphrase*. Da hier neue Objekte definiert werden, dürfen in solchen Sätzen keine schon definierten Objekte, Eigennamen,

Zahlen, arithmetische Ausdrücke, Zeichenketten, Mengen oder Listen vorkommen. Ebenso ist es nicht möglich Komplemente und Ergänzungen zu verwenden. [At08b]

Boolsche Ausdrücke zählen ebenfalls als Sätze. Vergleichsoperatoren sind =, !=, >, >=, < und <=. [At08b]

Sätze bzw. Satzteile lassen sich mittels *and* und *or* optional mit vorangestelltem Komma zusammenfügen. Zur Quantifizierung von Objekten gibt es drei Möglichkeiten:

- lokal, beschrieben durch das oben genannte *there is/are* Konstrukt
- existentiell, beschrieben durch *existential quantification: there is/are ... that ..., there is/are ... such that ...* und
- universell, beschrieben durch *universal quantification of countable nouns: for every/each, for each of* und *universal quantification of mass nouns: for all*. [At08b]

Zur Erzeugung von Hierarchien bzw. Unterordnungen kennt ACE Bedingungen (*Conditional Sentences*), Verneinungen (*Logical Negation* und *Negation as Failure*), Möglichkeiten und Notwendigkeiten (*Modality*) sowie eine sogenannte *that* - Unterordnung (*that-Subordination*). Für Bedingungen wird das *if ... then ...* Konstrukt verwendet (*If John enters a card then the automated-teller accepts it.*). Negationen werden zum einen mit den Wendungen *it is false that ...* oder *it is not true that ...* (*It is false that a screen blinks.*) und zum anderen mit *it is not provable that ...* (*It is not provable that a screen blinks.*) ausgedrückt. Zur Darstellung von Möglichkeiten und Notwendigkeiten dienen die Ausdrücke *it is (not) possible/necessary that ...* (*It is (not) possible/necessary that John waits.*). Mit Hilfe der *that* – Unterordnung wird von einem Verb ein ganzer (Teil)Satz als Objekt verstanden (*A clerk believes that a customer inserts a card.*). [At08b]

Im AceWiki werden einen Sachverhalt beschreibende Sätze in drei Kategorien eingeteilt:

- Aussagen, die sich in die Web Ontology Language (OWL) [Wi08g] übersetzen lassen
- Aussagen, die sich in die Semantic Web Rule Language (SWRL) [Wi08h] übersetzen lassen und
- Aussagen, die sich weder in OWL noch in SWRL übersetzen lassen. [Ku08a S.3]

In OWL oder SWRL übersetzbare Aussagen werden mit einem blauen Dreieck, andere Aussagen mit einem roten Dreieck markiert (Abbildung 56) [Ku08a S.3]. Dies ist relevant für den Export der Daten, da Aussagen, die sich weder in OWL noch in SWRL übersetzen lassen, auch nicht exportiert werden können. Zur Erhaltung eines konsistenten Zustandes der Ontologie verwendet AceWiki die Open Source Software Pellet (<http://clarkparsia.com/pellet>) [Ku08b S.1].

Article	Proper Name	References	Assignments
MI HS2			
▶ MI HS2 is a room. ▶ Every room is a location that has everything. add...			

Abbildung 56: AceWiki exportierbarer und nicht exportierbarer Satz

AceWiki bietet bisher nicht den kompletten Funktionsumfang von ACE bezüglich der Sprachkonstrukte an. Tabelle 1 zeigt einen Vergleich der Funktionalitäten von ACE und AceWiki.

Von ACE unterstützte Funktionalität	Von AceWiki unterstützt?
Nominalphrasen	
Eigennamen	Ja
Substantive in Einzahl und Mehrzahl	Ja
Substantive in unbestimmter Zahl	Nein
ganze und reelle Zahlen, arithmetische Ausdrücke	Nein
Zeichenketten, Mengen und Listen, Maßeinheiten	Nein
Pronomen, Reflexivpronomen	Nein
Indefinitpronomen	ja
Variablen	ja
Genitivkonstrukte	Nur mittels Of-Constructs
Adjektive	Nein
Transitive Adjektive	Ja
Verben	Ja
Of-Constructs	Ja
Adverbien	Nein
Funktionswörter	
begrenzende Angaben	Ja
Sätze	
Bedingungen	Ja
Verneinungen	Ja
Möglichkeiten und Notwendigkeiten	Nein
Vergleiche und Vergleichsoperatoren	Nein
Fragen	Ja

Tabelle 1: AceWiki Funktionalitätsvergleich ACE und AceWiki

Beim Erstellen von Sätzen ist dem Benutzer der *Sentence Editor* des AceWiki behilflich. Er kennt alle bisher erstellten Wortkonstrukte und bietet diese zum Bau eines Satzes an. Die Möglichkeit der Erstellung neuer Worte ist im *Sentence Editor* ebenfalls gegeben. [Ku08a S.5] Die Abbildungen 57 bis 60 zeigen die Erstellung eines Satzes mit Hilfe dieses Editors sowie den vollständigen Satz in der Wiki-Seite.

Zu erst wird die Nominalphrase bestehend aus einem Eigennamen oder einem Funktionswort und einem Substantiv erstellt. Anschließend folgt ein Verb.

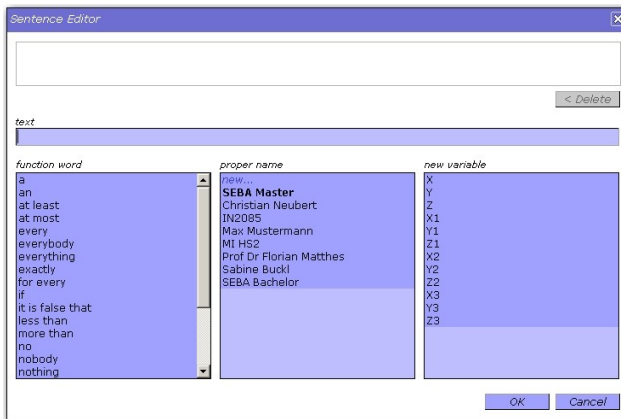


Abbildung 57: AceWiki Sentence Editor I

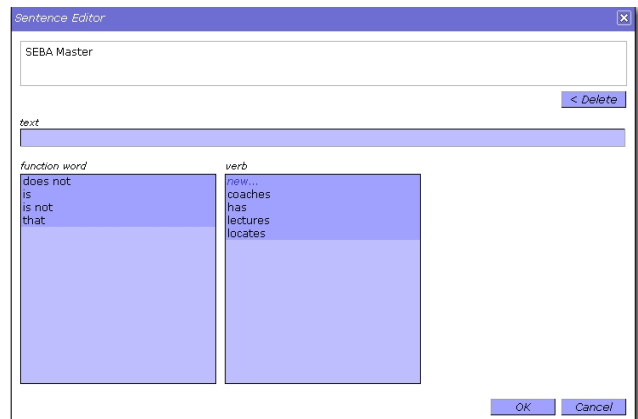


Abbildung 58: AceWiki Sentence Editor II

Für einen vollständigen Satz fehlt noch ein Komplement.

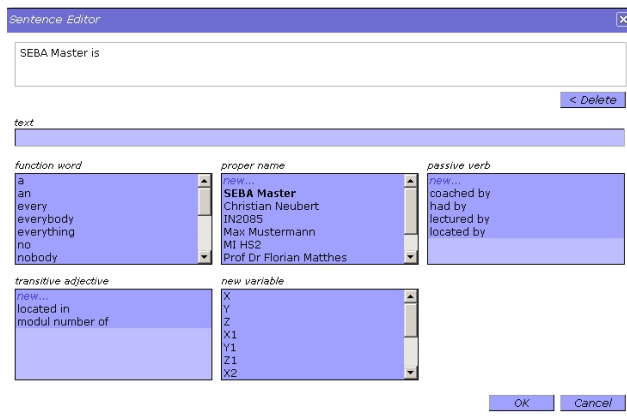


Abbildung 59: AceWiki Sentence Editor III



Abbildung 60: AceWiki Sentence Editor IV

Der Satz lässt sich nun beliebig durch *and* oder *or* erweitern (Abbildung 61 bis 63).

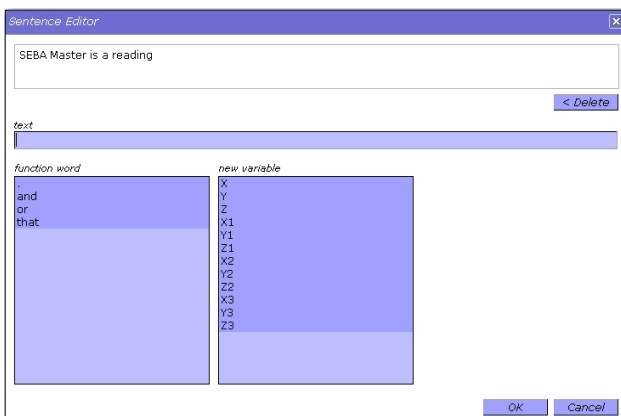


Abbildung 61: AceWiki Sentence Editor V

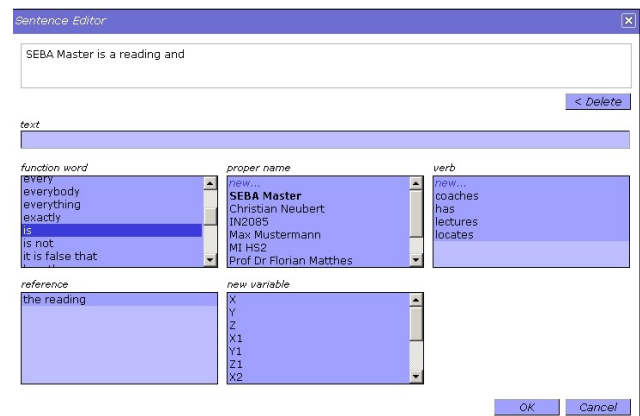


Abbildung 62: AceWiki Sentence Editor VI

Zum Schluss wird der Satz auf Korrektheit geprüft und entsprechend markiert auf der Wiki-Seite eingefügt (Abbildung 64). Alternativ ließen sich auch der Satz oder Satzteile in der Eingabezeile unter *text* direkt eingeben.

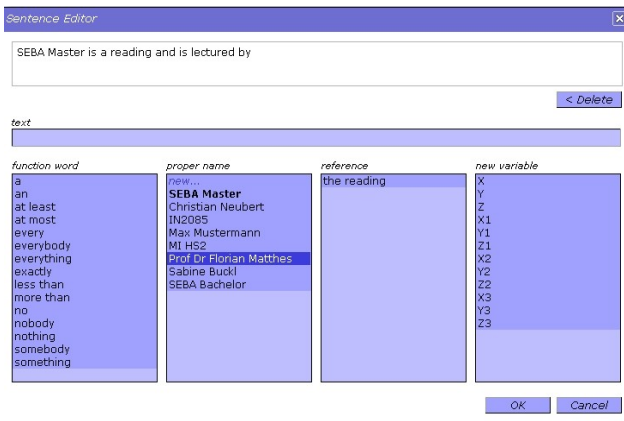


Abbildung 63: AceWiki Sentence Editor VII

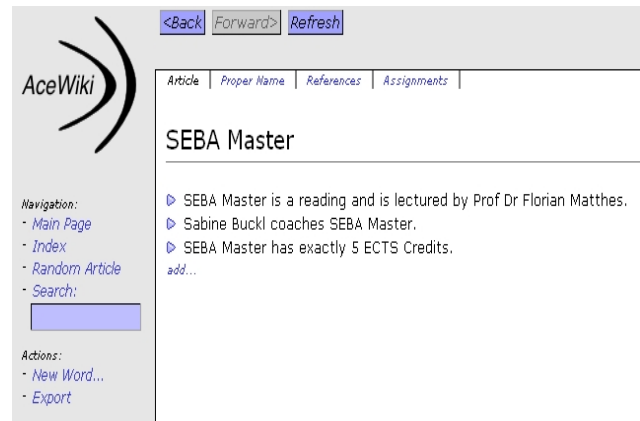


Abbildung 64: AceWiki erstellter Satz

AceWiki kann sowohl die Struktur als auch die Ontologie eines Satzes grafisch aufbereiten und darstellen. So lässt sich überprüfen, ob die Interpretation des Satzes durch das Wiki und die Intention des Benutzers übereinstimmen. Abbildung 65 und 66 zeigen die entsprechenden Ausschnitte.

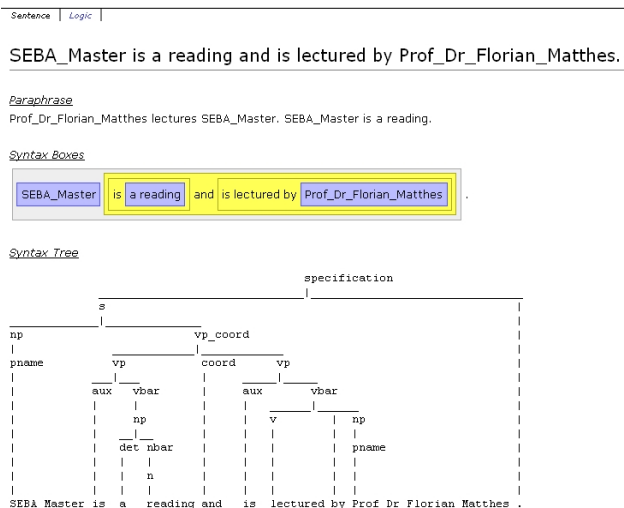


Abbildung 65: AceWiki Satzstruktur

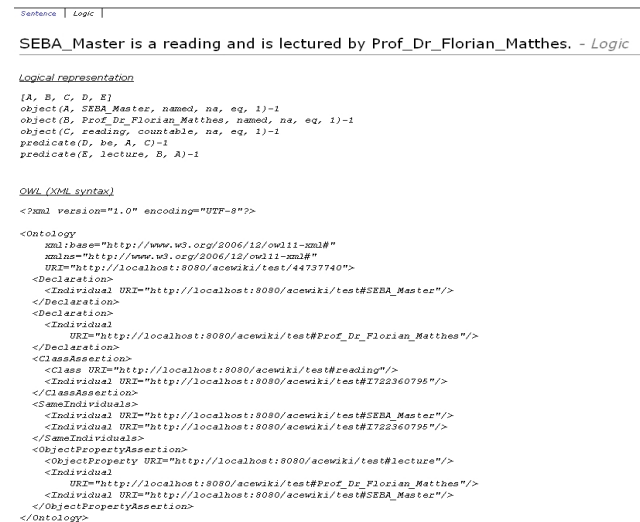


Abbildung 66: AceWiki Satz-Ontologie

2.4.3 Abfragesprache

Ein vollständig entwickelte Abfragesprache ist in AceWiki bisher nicht vorhanden. Dies wird in [Ku08a S.3] als zukünftige Arbeit betrachtet. Dennoch gibt es mit den Fragen bereits Ansätze Abfragen stellen zu können.

2.4.3.1 Fragen

ACE kennt sowohl Ja/Nein Fragen(Yes/No queries) als auch sogenannten wh-Fragen (wh-queries). Beispiele für Ja/Nein Fragen sind: *Does John enter a card?*, *Does John not wait?* oder *Is the card valid?*. Die wh-Fragen beinhalten immer ein Fragewort. Wobei Fragen nach dem Subjekt oder dem Objekt des Satzes *who*, *whose*, *what* oder *which* und Fragen nach der näheren Erläuterung eines Verbs *how*, *where* oder *when* enthalten. [At08b] Fragen werden genauso wie alle anderen Sätze mit dem *Sentence Editor* (Abbildung 67) erstellt. Abbildung 68 zeigt eine Wiki Seite mit unterschiedlichen Fragen und den von AceWiki gefundenen Antworten.

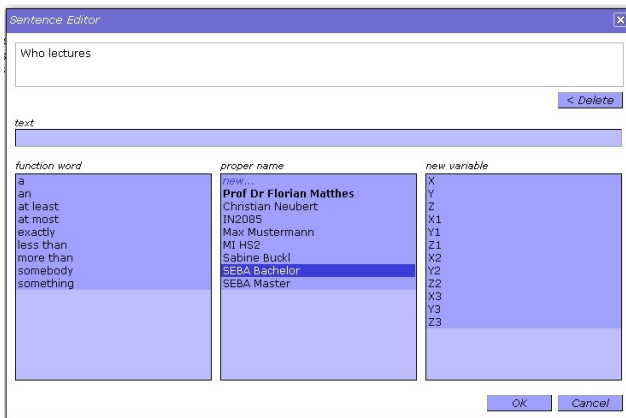


Abbildung 67: AceWiki Sentence Editor Frage

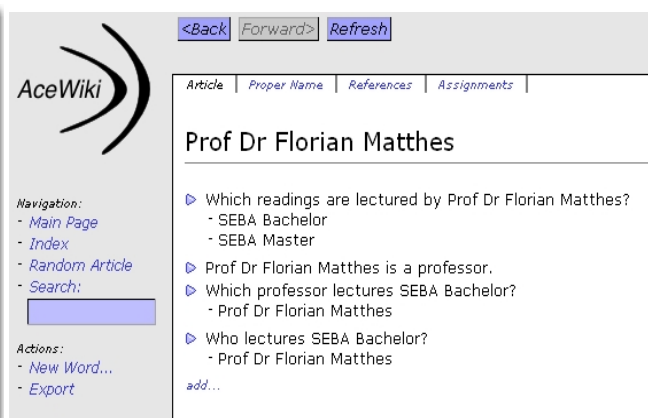


Abbildung 68: AceWiki Wiki-Seite mit Fragen

2.4.4 Export

Daten und Schema lassen sich in ein XML – Dokument im OWL/RDF Format exportieren. Den exportierten XML-Datei ist zu entnehmen, dass AceWiki für diese Funktionalität die OWL API in der Version 2.2.1.941 (<http://owlapi.sourceforge.net>) verwendet. Dabei wird immer der ganze Inhalt des Wikis ausgegeben. Eine selektive Auswahl einzelner Wiki-Seiten ist nicht möglich. Wie oben beschrieben und in Abbildung 66 zu sehen, lässt sich aber die Ontologie einzelner Sätze anzeigen und kann somit auch einzeln entnommen werden.

2.4.5 Import

Eine Importfunktion ist nicht vorhanden.

2.4.6 Berechtigungskonzept

Ein Berechtigungskonzept ist nicht vorhanden. Jeder Benutzer, der zu dem Wiki per Webbrowser Zugang hat, kann alle Inhalte ansehen, ändern und neue Inhalte erstellen.

2.4.7 Useability

Ein Anspruch von AceWiki ist die einfache Handhabung insbesondere für Benutzer, die kein Hintergrundwissen bezüglich Logik, formalen Sprachen oder Ontologien aufweisen. In [Ku08a S.4-7] wird ein Experiment zur Verwendung von AceWiki, durch auf dem Gebiet Ontologien unbedarften Benutzern, aufgeführt. Das Ergebnis zeigt, dass 80% der erstellten Sätze korrekt waren. Korrekt bedeutet hier, dass es sich bei dem erstellten Satz um eine wahre Aussage in der realen Welt handelt und AceWiki die Aussage der Intention des Erstellers entsprechend interpretiert. [Ku08a S.5].

AceWiki prüft bei der Erstellung zwar die Korrektheit eines Satzes aber bisher nicht, ob der Sachverhalt schon einmal in einem Satz dargestellt wurde. So ist es möglich komplett identische Sätze auf ein und derselben Wiki-Seite zu erstellen.

Eine Möglichkeit einfache Zeichenketten ohne wohldefinierte Bedeutung einzufügen, ist nicht gegeben. Alle Satzbestandteile müssen vor der Verwendung definiert werden.

2.5 DITF-MR-Recherchesystem

Das Zentrum für Management Research der Deutschen Institute für Textil- und Faserforschung Denkendorf (DITF MR Denkendorf <http://www.ditf-denkendorf.de/mr/>) hat mit dem Recherchesystem ein Experteninformationssystem entwickelt, welches Informationen strukturiert speichert. Diese Daten können zu einer fallbasierten Recherche und Argumentation verwendet werden. „The retrieval system for structured information is a ontology-based expert system with case-based reasoning functionality especially designed for small and medium enterprises.“ [Re08a] Das Recherchesystem ist für den Einsatz in der mittelständischen Textilindustrie erstellt worden und soll zum Beispiel durch die Speicherung und die damit mögliche kontinuierliche Verbesserung der Webmaschineneinstellung zur Produktionsoptimierung und Qualitätssteigerung beitragen. [Re08b S.29ff]

2.5.1 Steckbrief

Status:	unbekannt
Version:	0.4
Veröffentlichungsdatum:	unbekannt
Lizenzmodell:	GNU GPL V3
Homepage:	http://rs-demos.ditf-denkendorf.de/tum/
Bearbeitungsmodell:	Formularfelder
Programmiersprache:	PHP
Annotierbare Elemente:	Informationsobjekttypen
Unterscheidung zwischen Wiki-Seiten und Konzepten:	Ja
Annotationstyp:	Zeichen
Kontextinformationen über Annotationen:	Nein
Instanzenimport:	Nein
Ontologieimport:	Nein
Instanzenexport:	Nein
Ontologieexport:	Nein

2.5.2 Strukturierung

Zur Strukturierung der Daten im DITF-MR-Recherchesystem dienen *Informationsobjekttypen* mit dazugehörigen *Attributen*, *Informationsobjekte* als Instanzen der Informationsobjekttypen sowie *Relationenstypen* zwischen den Objekttypen und somit auch zwischen den Objekten.

2.5.2.1 Informationsobjekttypen

In [Re08b S.86] werden Informationsobjekttypen mit Formatvorlagen verglichen. Sie beschreiben die Struktur eines, diesem Typ zugeordneten, Informationsobjekts mit verschiedenartigen Attributen. Zudem lassen sich zwischen Informationsobjekttypen Beziehungstypen definieren, die dann durch Beziehungen konkreter Informationsobjekte verwendet werden. Das Anlegen neuer Informationsobjekttypen (Abbildung 69) gehört zum Bereich der Systemverwaltung und ist somit nicht für jeden Benutzer zugänglich. Die Typen *Personen*, *Rollen*, *Fragen und Antworten*, *Suchanfragen* und *Test* sind vordefiniert.



Abbildung 69: DITF-MR-RS Informationsobjekttyp anlegen

2.5.2.2 Beziehungstypen

Zwischen Informationsobjekttypen können Beziehungstypen definiert werden, die immer einen sogenannten *Start-Typ* und einem sogenannten *End-Typ* besitzen. Die Beziehung selbst erhält für beide Navigationsrichtungen jeweils eine Bezeichnung. Multiplizitäten lassen sich nicht angeben. Das entsprechende Eingabefenster zeigt Abbildung 70.

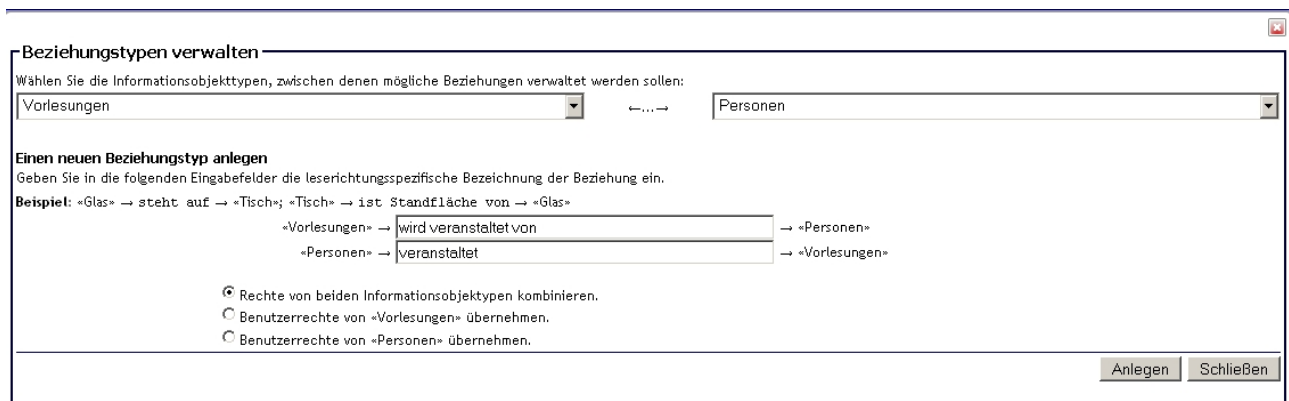


Abbildung 70: DITF-MR-RS Beziehungstyp Bezeichnungen

Zwischen zwei Informationsobjekttypen lassen sich mehrere Beziehungstypen erstellen. Zudem können Bezeichnungen bestehender Beziehungstypen geändert und Beziehungstypen gänzlich gelöscht werden. In Abbildung 71 sind das Anlegen eines weiteren sowie die Buttons zur Änderung eines bestehenden Beziehungstypen abgebildet.

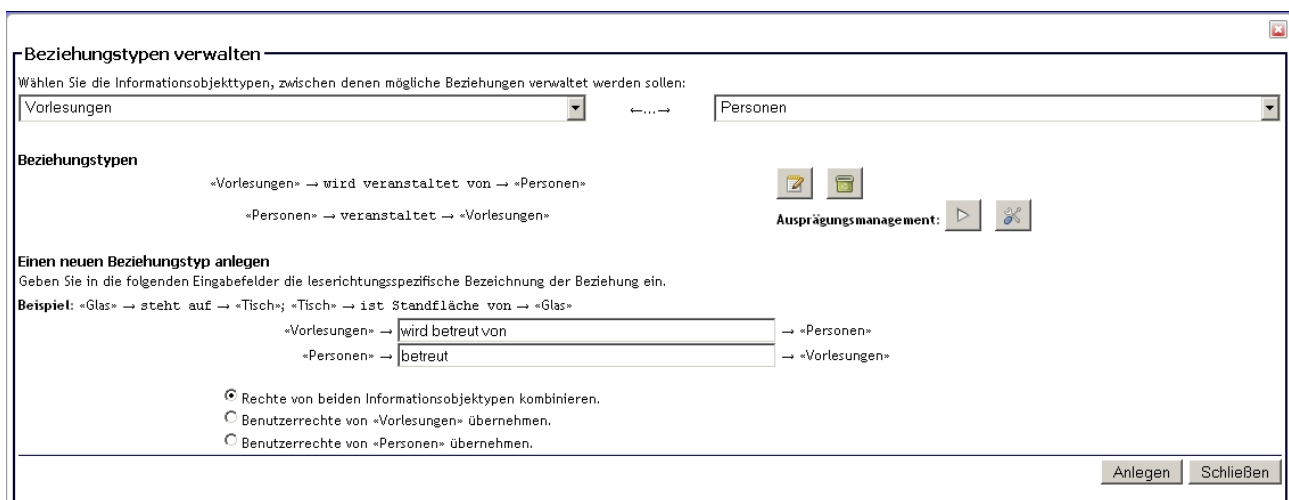


Abbildung 71: DITF-MR-RS weiterer Beziehungstyp

2.5.2.3 Attribute

Attribute werden einem Informationsobjekttyp zugeordnet. Sie erhalten eine Bezeichnung sowie einen vordefinierten Attributtyp. Diese Attributtypen sind keine Datentypen im eigentlichen Sinne. Sie stellen vielmehr eine Mischung aus, zum Teil zusammengesetzten, Datentypen und Präsentationsinformationen dar. Der Typ Personenbezeichnung besteht zum Beispiel aus drei Zeichenketten, jeweils eine für den akademischen Titel, den Vornamen und den Nachnamen. Jeder Typ bietet verschiedene Parameter, mit denen das jeweilige Attribut genauer definiert werden kann. Als Beispiel können hierzu bei Textfeldern die Zeichenanzahl oder beim Typ Zahl die Tausender- und Dezimaltrennzeichen, die Anzahl der Dezimalstellen und die Einheit der Zahl genannt werden. Abbildung 72 zeigt die Auswahl der verfügbaren Attributtypen.

The screenshot shows the 'Attribute-Definitionen' window in the DITF-MR-RS system. The window has a title bar with 'TU München' and a user name 'Martin Stange'. Below the title bar is a navigation bar with 'Vorlesungen' and a search bar. The main content area is titled 'Attribute-Definitionen' and contains a list of attribute types. A green box highlights the text 'Hier abgelegte Attribute kommen an die letzte Stelle der Liste.' Below this is a section titled 'Attribute hinzufügen' with a dropdown menu and a 'Hinzufügen!' button. The dropdown menu is open, showing a list of attribute types: Einzeiliges Textfeld, Mehrzeiliges Textfeld, Datei, Bild, Binärdaten (intern), Passwort, Personenbezeichnung, Zahl, Wertebereich, Auswahlbox (DropDown-Liste; Auswahl aus einer vorgegebenen Wertemenge), Zeitpunkt (Datum & Uhrzeit), URL, Letzte Änderungen (Person & Zeitpunkt), Relation, Klassifikationsnummer, and Status des Informationsobjekts. A small message on the right says 'Sie also im Zweifelsfall vorher ein Backup an.'

Abbildung 72: DITF-MR-RS Attributauswahl

Nach der Auswahl und dem Drücken des *Hinzufügen!* Buttons wird der Name des Attributes abgefragt (Abbildung 73).

The screenshot shows the 'Attribute-Definitionen' window with a form for creating a new attribute. The form has three main sections: 'Ein neues Attribut anlegen', 'Wie soll das neue Attribut vom Typ "Datei, Bild" heißen?', and 'Tipp: Wählen Sie einen kurzen, aber dennoch eindeutigen Begriff, den alle Nutzer verstehen. Beispiele: Nutzeffekt, Reißpunkt, ...'. The 'icon' button is highlighted. The bottom bar contains 'Attribut anlegen' and 'Verwerfen -> zurück zur Attributliste' buttons.

Abbildung 73: DITF-MR-RS Attributbezeichnung

Für ausgewählte Attributtypen zeigen die Abbildungen 74 bis 77 die entsprechenden Definitionsparameter. Attributname, Variablenname, Tabellenname in der Datenbank, das Benutzerrecht zur Attributänderung sowie die Angabe ob dem Attribut zwingend ein Wert zuzuweisen ist, lassen sich bei nahezu allen Attributtypen bearbeiten.

Attribut-Definitionen

Attribut "Vorlesungstitel" ändern

Name:

Variablenname für Name & Beschreibung:

Speziell formatierte Wertanzeige? ☐ Ja ☒ Nein

HTML-Code vor allen Werten:

HTML-Code zwischen Werten:

HTML-Code nach den Werten:

Attribut darf durch Benutzer geändert werden: ☒ Ja ☐ Nein

Attribut-Wert muss gesetzt sein: ☐ Ja ☒ Nein

Anzahl der Wertesets (1..20, n (beliebig)):

Maximale Zeichenzahl in Eingabefeld:

Minimalwert: 1 Zeichen, Maximalwert: 255 Zeichen

HTML-Tags erlaubt: ☐ Ja ☒ Nein

Attribut-Tabellenname in Datenbank:

Erlaubte Zeichen: a-zA-Z0-9_ . Maximal 16 Zeichen.

Abbildung 74: DITF-MR-RS Attributtyp einzeliges Textfeld

Attribut-Definitionen

Attribut "Zeit" ändern

Name:

Variablenname für Name & Beschreibung:

Speziell formatierte Wertanzeige? ☐ Ja ☒ Nein

HTML-Code vor allen Werten:

HTML-Code zwischen Werten:

HTML-Code nach den Werten:

Attribut darf durch Benutzer geändert werden: ☒ Ja ☐ Nein

Attribut-Wert muss gesetzt sein: ☐ Ja ☒ Nein

Anzahl der Wertesets (1..20, n (beliebig)):

Ein- und Ausgabeformat:

Syntax-Beispiele: DD.MM.YYYY hh:mm:ss, YYYY-MM-DD, mm min ss sec ...

DD → Tag(01...31) hh → Stunde (00...23)

MM → Monat (01...12) mm → Minute (00...59)

YYYY → Jahr (0000...9999) ss → Sekunde (00...59)

Einheit:

Attribut-Tabellenname in Datenbank:

Erlaubte Zeichen: a-zA-Z0-9_ . Maximal 16 Zeichen.

Abbildung 75: DITF-MR-RS Attributtyp Zeitpunkt

Attribut-Definitionen

Attribut "Sprache" ändern

Name:

Variablenname für Name & Beschreibung:

Speziell formatierte Wertanzeige? ☐ Ja ☒ Nein

HTML-Code vor allen Werten:

HTML-Code zwischen Werten:

HTML-Code nach den Werten:

Attribut darf durch Benutzer geändert werden: ☒ Ja ☐ Nein

Attribut-Wert muss gesetzt sein: ☐ Ja ☒ Nein

Anzahl der Wertesets (1..20, n (beliebig)):

Zur Verfügung stehende Werte:

de deutsch
en englisch

Syntax: eindeutigerSchlüssel|Wert.
Ein Eintrag pro Zeile, die Zeilen mit Return trennen, maximale Schlüsselänge: 10 Zeichen. Das Trennzeichen «|» können Sie durch gemeinsames Drücken der Tasten «AltGr» + «>» eingeben.

Attribut-Tabellenname in Datenbank:

Erlaubte Zeichen: a-zA-Z0-9_. Maximal 16 Zeichen.

Abbildung 76: DITF-MR-RS Attributtyp Auswahlbox

Allein durch die Anlage eines Beziehungstypen lassen sich noch keine Informationsobjekte miteinander verknüpfen. Hierzu bedarf es der in Abbildung 77 dargestellten Zuweisung eines Beziehungstypen zu einem Attribut vom Typ Relation. Erst dadurch wird ein vorher definierter Beziehungstyp zum festen Bestandteil eines Informationsobjektypen.

Attribut-Definitionen

Attribut "Veranstalter" ändern

Name:

Variablenname für Name & Beschreibung:

Speziell formatierte Wertanzeige? ☐ Ja ☒ Nein

HTML-Code vor allen Werten:

HTML-Code zwischen Werten:

HTML-Code nach den Werten:

Attribut darf durch Benutzer geändert werden: ☒ Ja ☐ Nein

Attribut-Wert muss gesetzt sein: ☐ Ja ☒ Nein

Anzuzeigender Relationstyp:

- ☐ Vorlesungen → wird veranstaltet von → Personen
- ☒ Vorlesungen → wird betreut von → Personen
- ☐ Vorlesungen → findet statt in → Orte

Personen, die diese Relationen pflegen, sollten für die gewählten Relationstypen zumindest Leserechte haben.

Verknüpftes Informationsobjekt anzeigen als: ☒ Name ☐ Beschreibung ☐ Name und Beschreibung ☐ Eigenschaften

Attribut-Tabellenname in Datenbank:

Erlaubte Zeichen: a-zA-Z0-9_. Maximal 16 Zeichen.

Abbildung 77: DITF-MR-RS Attributtyp Relation

Ein Beispiel für einen definierten Informationsobjektyp zeigt Abbildung 78.



Abbildung 78: DITF-MR-RS Informationsobjekttyp

2.5.2.4 Informationsobjekte

Die eigentlichen Daten werden in Informationsobjekten hinterlegt. Sie sind Instanzen immer genau eines Informationsobjekttyps. Dieser Typ legt, wie oben beschrieben, die Struktur der Objekte fest. Nach der Auswahl des Informationsobjekttyps lässt sich mit Hilfe des Karteireiters *Neues Objekt erstellen* ein neues Informationsobjekt entsprechenden Typs anlegen (Abbildung 79).



Abbildung 79: DITF-MR-RS Informationsobjekt anlegen

Nach dem Anlegen eines neuen Objekts lassen sich die Attribute mit Werten versehen. Die Eingabe bei einzeilige Textfelder, Zahlen oder Datumsangabe gestaltet sich intuitiv. Ein besonderes Augenmerk gilt den Attributen vom Typ Relation. In Abbildung 80 ist die Instantiierung eines Beziehungstypen abgebildet. Das erste Feld dient der Eingabe eines Suchbegriffes, das zweite Feld zeigt die gefundenen Informationsobjekte und das dritte Feld zeigt die bestehenden, die hinzugefügten und die zu löschenden Verknüpfungen zu Informationsobjekten dieser Beziehung. [Re08a S.32]

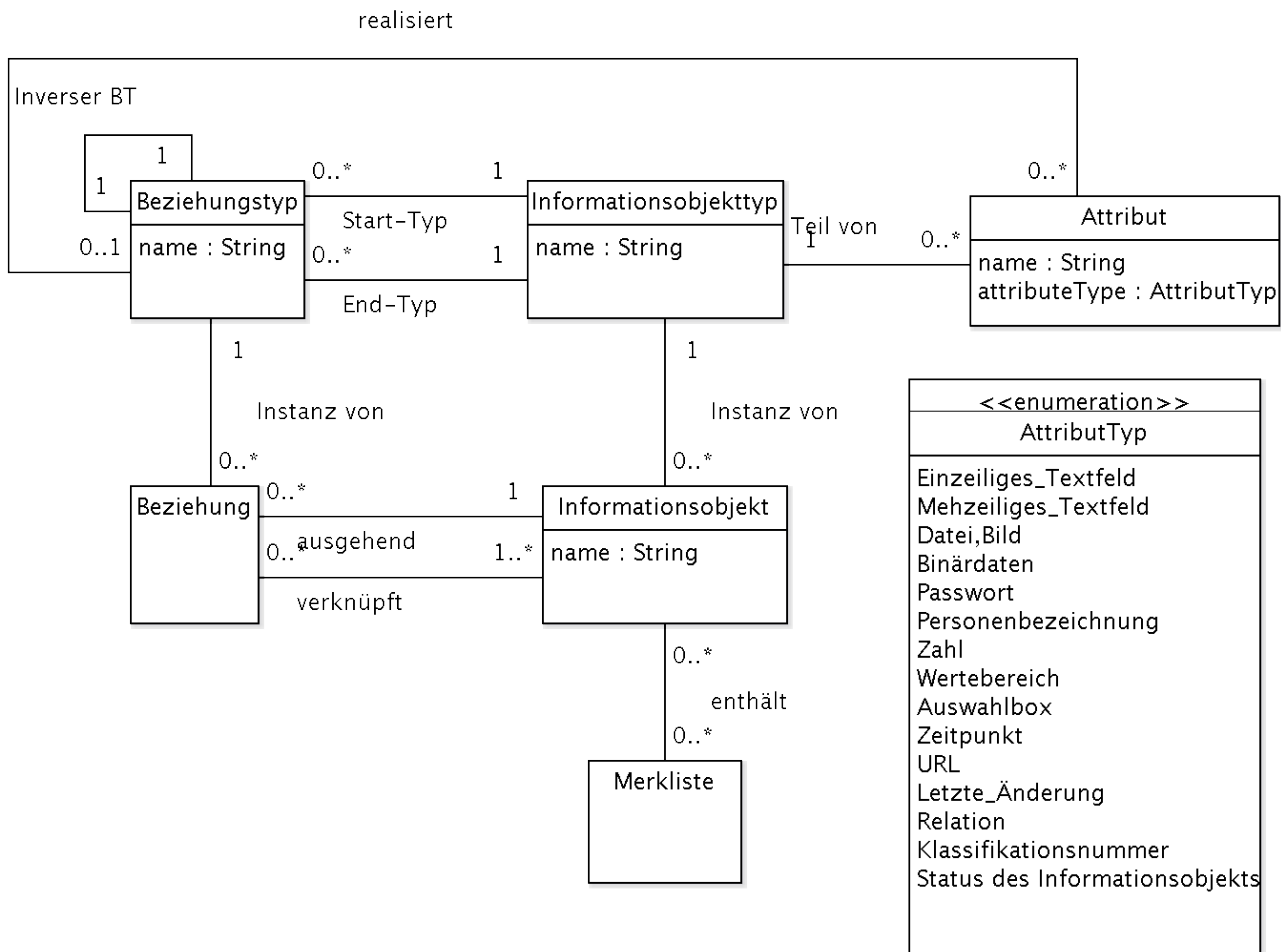


Abbildung 82: DITF-MR-RS UML Klassendiagramm

2.5.3 Abfragesprache

Das DITF-MR-Recherchesystem bietet unterschiedliche Möglichkeiten der Informationsaufbereitung und Suche. Hierzu zählen die Definition verschiedener Attribut-Ansichten pro Informationsobjekttyp, die *Volltextsuche* sowie die *Recherchefunktion*.

2.5.3.1 Attribut-Ansichten

Für jeden Informationsobjekttypen lassen sich sogenannte Attribut-Ansichten erstellen. Sie dienen zur anwendergerechten Aufbereitung und Darstellung der Attribute. Abbildung 83 zeigt die Maske zur Verwaltung der Attribut-Ansichten. Hier lassen sich neue Ansichten definieren, bestehende bearbeiten und löschen. Zudem ist es möglich die Sichtbarkeit der Standard-Attribut-Ansicht, welche alle definierten Attribute enthält, auf Administratoren zu begrenzen.

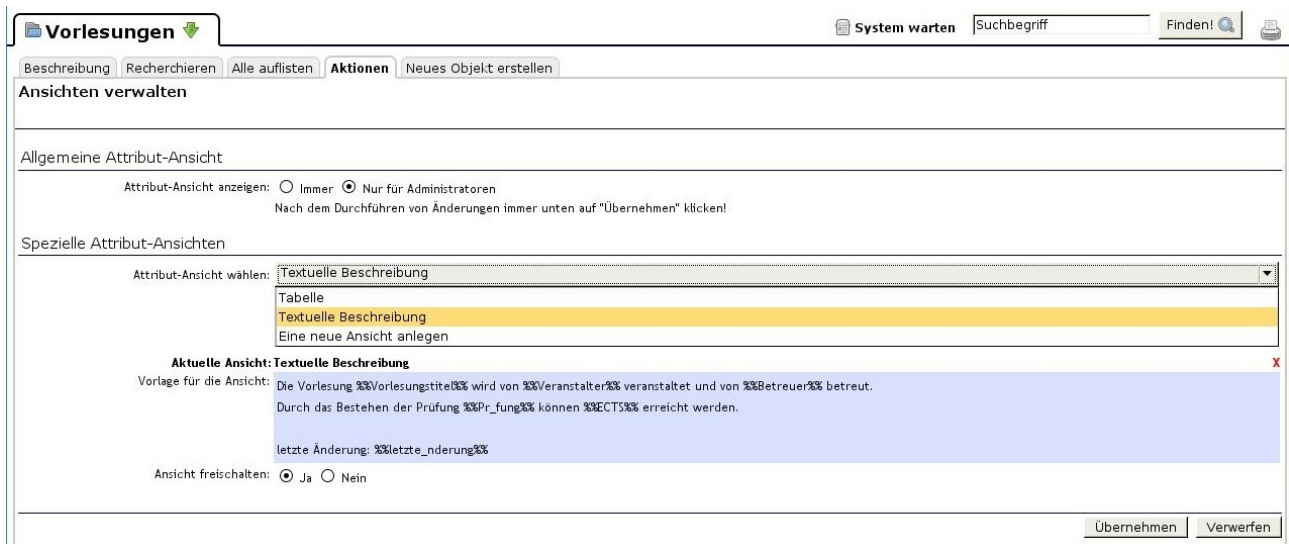


Abbildung 83: DITF-MR-RS Attribut-Ansichten

Zum Erstellen der Ansicht dient wieder der schon bekannte Rich-Text-Editor. Die Attribut lassen sich durch die Auswahl aus einer Drop-Down-Liste einfügen (Abbildung 84). Der Platzhalter für den Attributwert besteht aus dem bei der Attributdefinition angegebenen Variablennamen sowie zwei vorangestellten und zwei angehängten Prozentzeichen (z.B. `%%Vorlesungstitel%%`).

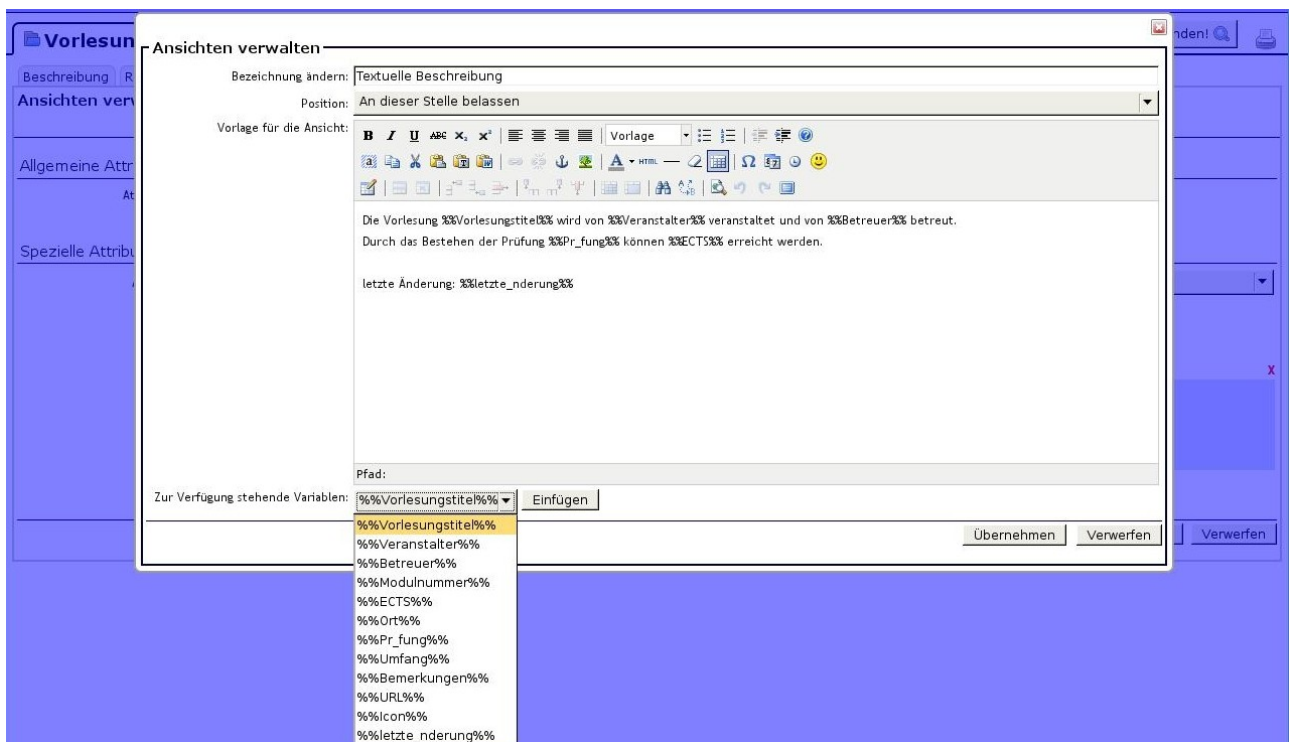


Abbildung 84: DITF-MR-RS Attribut-Ansicht erstellen

Abbildung 85 zeigt die Standard-Attribut-Ansicht, die in diesem Falle nur noch für Administratoren sichtbar ist. Weitere Ansichtsmöglichkeiten stehen in der Karteireiterleiste zur Verfügung.



Abbildung 85: DITF-MR-RS Standard-Attribut-Ansicht

Die beiden erstellten Attribut-Ansichten zeigen Abbildung 86 und Abbildung 87. Die Standard-Attribut-Ansicht ist hier nicht auswählbar, da der Benutzer kein Administrator ist.

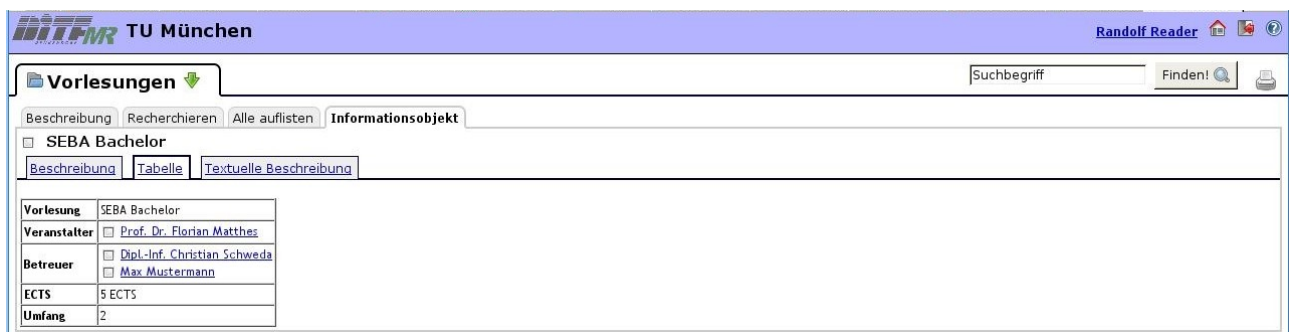


Abbildung 86: DITF-MR-RS Attribut-Ansicht Tabelle

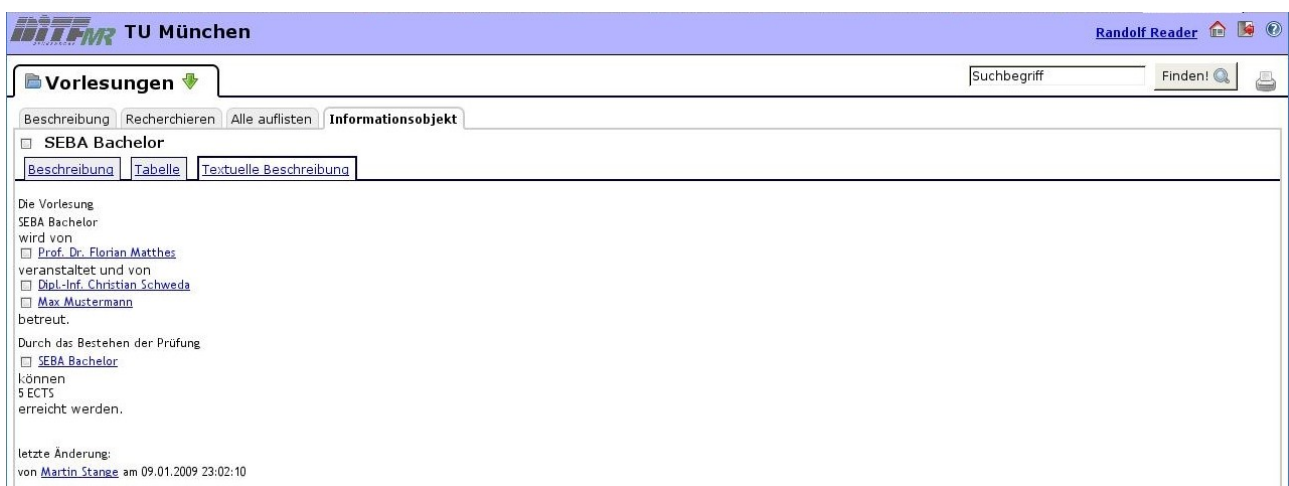


Abbildung 87: DITF-MR-RS Attribut-Ansicht Text

2.5.3.2 Volltextsuche

Die Volltextsuche (Abbildung 88) „findet Informationsobjekte der ausgewählten Typen, die den Suchbegriff in Name oder Beschreibung enthalten“ [Re08a S.14]. Eine Suche in anderen Textattributen, wie zum Beispiel den Textfeldern, erfolgt hierbei nicht. Die Bewertung der gefundenen Informationsobjekte beruht auf dem Vorkommen des Suchbegriffes in Name und/oder Beschreibung. Ein Informationsobjekt wird mit 10 Punkten für das Vorhandensein im Namen, mit 5 Punkten für das Vorkommen in der Beschreibung und mit 15 Punkten für das Vorkommen sowohl im Namen als auch in der Beschreibung bewertet. [Re08a S.17]



Abbildung 88: DITF-MR-RS Volltextsuche

2.5.3.3 Recherche

Recherche bedeutet im DITF-MR-Recherchesystems die „Suche nach Informationsobjekten mit speziellen Eigenschaften innerhalb des ausgewählten Informationsobjekttyps“ [Re08a S.14]. Zur Definition einer Recherche können Bedingungen mit *Muss*, *Kann*, *Darf nicht* und *Nicht spezifiziert* angegeben werden (Abbildung 89).



Abbildung 89: DITF-MR-RS Recherche

Hierzu werden Attribute eines Informationsobjekttyps ausgewählt (Abbildung 90) und mit Werten versehen, die die gewünschte Auswahl beschreiben. Bei eine Beziehungen darstellenden Attributen werden, wie in Abbildung 91 zu sehen, entsprechende Informationsobjekte vorgeschlagen.

Diese Eigenschaften *muss* «Personen» erfüllen

Muss-Kriterien
Nur Informationsobjekte mit diesen Eigenschaften kommen ins Suchergebnis.

Attribute hinzufügen
Wählen Sie ein Attribut aus der unten stehenden Liste aus und klicken Sie auf "Hinzufügen".

Hat Rollen Hinzufügen!

– Wählen Sie ein Attribut aus! –

Personenbezeichnung
Benutzername
Hat Rollen
Startseite
Kurzbeschreibung Kann-Kriterien hinzufügen

Abbildung 90: DITF-MR-RS Recherche Attributauswahl

Diese Eigenschaften *muss* «Personen» erfüllen

Muss-Attribut «Hat Rollen» der Recherche ändern

Hat Rollen entspricht ...

Wenn Sie nach dem Eintippen etwas warten, erhalten Sie automatisch Recherchevorschläge.
Beachten Sie bitte, dass die Vorschläge auch auf Treffern in der Beschreibung der entsprechenden Informationsobjekte beruhen können. Recherchiert wird aber ausschließlich nach den Namen.

7 Vorschläge (zum Übernehmen anklicken):

- [Administrator](#)
- [Informationspfleger](#)
- [Informationspfleger mit Löschberechtigung](#)
- [Leser](#)
- [Professor](#)
- [wissenschaftlicher Mitarbeiter](#)
- [Zugriffsberechtigter](#)

Tipps:

- Haus* findet Haus, Hausmeister, Hausverwalter, ...
- kann +unbedingt -ohne findet Einträge, die *kann* enthalten können, auf jeden Fall *unbedingt* enthalten und nicht *ohne* enthalten.
- "genau diese Reihenfolge" findet Einträge, wo die Wörter *Reihenfolge*, *diese* und *genau* in "*genau dieser Reihenfolge*" vorkommen.

Übernehmen Verwerfen

Abbildung 91: DITF-MR-RS Recherche Attributwert

In den Abbildungen 92 und 93 sind die Definitionen der Bedingungen *Muss* und der *Nicht spezifiziert* zu sehen.

Diese Eigenschaften *muss* «Personen» erfüllen

Muss-Kriterien
Nur Informationsobjekte mit diesen Eigenschaften kommen ins Suchergebnis.

▪ ... Hat Rollen «["Zugriffsberechtigter"](#)» X

Attribute hinzufügen
Wählen Sie ein Attribut aus der unten stehenden Liste aus und klicken Sie auf "Hinzufügen".

– Wählen Sie ein Attribut aus! – Hinzufügen!

Attribute löschen
Wenn Sie ein Attribut aus der Suche löschen möchten, klicken Sie auf das rote "X" neben dem Attributnamen.

Filter anwenden Kann-Kriterien hinzufügen

Abbildung 92: DITF-MR-RS Recherche Muss-Kriterien

Diese Eigenschaften von «Personen» sind nicht definiert

Leere Felder
Nur Informationsobjekte, für die diese Eigenschaften **nicht spezifiziert** sind, kommen ins Suchergebnis.

■ **Kurzbeschreibung** X

Attribute hinzufügen
Wählen Sie ein Attribut aus der unten stehenden Liste aus und klicken Sie auf "Hinzufügen".
- Wählen Sie ein Attribut aus! - Hinzufügen!

Attribute löschen
Wenn Sie ein Attribut aus der Suche löschen möchten, klicken Sie auf das rote "X" neben dem Attributnamen.
Darf nicht-Kriterien hinzufügen Filter anwenden

Abbildung 93: DITF-MR-RS Recherche Nicht spezifiziert-Kriterien

Die Gesamtheit der erstellten Kriterien wird als *Filter* bezeichnet. Alle Filterkriterien lassen sich, wie Abbildung 94 zeigt, in einer Übersicht auflisten. Filter können für eine spätere Wiederverwendung gespeichert werden.

Aktuell definierter Filter für «Personen»

Das IObjekt <i>muss</i> folgende Kriterien erfüllen:	... Hat Rollen «Zugriffsberechtigter» Weitere Muss-Kriterien hinzufügen
Das IObjekt <i>sollte</i> folgende Kriterien erfüllen:	- Personenbezeichnung entspricht («Herr») ... Hat Rollen «Professor» Weitere Kann-Kriterien hinzufügen
Das IObjekt <i>darf</i> folgende Kriterien <i>nicht</i> erfüllen:	Benutzername entspricht «muster» Weitere "darf nicht"-Kriterien hinzufügen
Die folgenden Werte sind <i>nicht</i> gesetzt:	Kurzbeschreibung Nach weiteren leeren Feldern suchen

Filter speichern aktuellen Filter löschen Filter anwenden Schließen

Abbildung 94: DITF-MR-RS Recherche Übersicht Filterkriterien

In Abbildung 95 ist das Rechercheergebnis zu sehen, welches die oben genannten Kriterien erfüllt. Die vor den Namen der Informationsobjekte stehenden Bezeichner *ok* bedeutet, dass alle *Muss*, *Kann nicht* und *Nicht spezifiziert* Kriterien erfüllt wurden. Werden darüber hinaus *Kann* Kriterien erfüllt, gibt eine Bezeichnung 1* respektive 2** die Anzahl der erfüllten *Kann* Kriterien an. [Re08a S.22]

Personen System warten Suchbegriff Finden!

Beschreibung Recherchieren Alle auflisten Aktionen Neues Objekt erstellen

Filtern Sie die Informationsobjekte nach folgenden Kriterien:
-Muss- -Kann- -Darf nicht- -keine Werte vorhanden-
-Filter anzeigen- -Filter laden/löschen-

Ergebnisse merken Filter speichern Filter löschen

Name und Beschreibung
2** Prof. Dr. Florian Matthes Kurzbeschreibung: -- Rollen: Professor , Zugriffsberechtigter
1* Dipl.-Inf. Christian Neubert Kurzbeschreibung: -- Rollen: wissenschaftlicher Mitarbeiter , Zugriffsberechtigter
1* Dipl.-Inf. Christian Schweda Kurzbeschreibung: -- Rollen: wissenschaftlicher Mitarbeiter , Zugriffsberechtigter
ok Dipl.-Inf. Sabine Buckl Kurzbeschreibung: -- Rollen: wissenschaftlicher Mitarbeiter , Zugriffsberechtigter

Personen Alle ...
■ [Prof. Dr. Florian Matthes](#)
■ [Dipl.-Inf. Christian Schweda](#)

Abbildung 95: DITF-MR-RS Rechercheergebnis

Mit Hilfe der Auswahl des kleinen Kästchens vor dem Namen des Informationsobjekts lässt sich dieses in die am rechten Rand erscheinende Merklste aufnehmen. Es können auch Informationsobjekte eines anderen Typs in der Merklste abgelegt werden. Somit verliert der Benutzer bei einer Recherche oder Bearbeitung alle gemerkten und für ihn relevanten Objekte nicht aus den Augen. Es lassen sich zudem allen Objekte eines Typs auf einmal ändern, löschen oder duplizieren. Der Vergleich von Objekten gleichen Typs ist ebenso möglich und in Abbildung 96 dargestellt. [Re08a S.24f]

Abbildung 96: DITF-MR-RS Vergleich Personen

2.5.4 Export

Es lassen sich im DITF-MR-Recherchesystem Backups von Informationsobjekt- und Beziehungstypen sowie von Informationsobjekten und Beziehungen erstellen. Dabei ist es möglich von Objekten sowie von Typen sowohl getrennte als auch gemeinsame Backups zu erstellen. Zudem können die zu exportierenden Informationsobjekttypen eingeschränkt werden. Im Rahmen eines Backups ist auch das Löschen von Informationsobjekttypen samt aller Informationsobjekte möglich. Abbildung 97 zeigt den entsprechende Auswahldialog.

Abbildung 97: DITF-MR-RS Backup

Informationsobjekte eines Informationsobjekttyps lassen sich in eine *.csv-Datei exportieren. Diese ist mit der unten genannten Vorlagen-Datei des Imports deckungsgleich.

2.5.5 Import

Vom DITF-MR-Recherchesystem wird der Import vorher mit dem System erstellter Backups von Informationsobjekt- und Beziehungstypen sowie von Informationsobjekten und Beziehungen unterstützt. Das Format der Backups ist textbasiert und proprietär.

Informationsobjekte eines einzelnen Typs lassen sich aus einer *.csv – Datei importieren. Hierzu stellt das DITF-MR-Recherchesystem für jeden Informationsobjekttyp eine Vorlage-Datei zur Verfügung, welche durch den Benutzer mit den zu importierenden Daten gefüllt werden kann. Diese Vorlage enthält nicht nur die Struktur des Informationsobjekttyps sondern auch alle bereits im System vorhandenen Informationsobjekte diesen Typs.

2.5.6 Berechtigungskonzept

Das Berechtigungskonzept ist rollenbasiert. Jedem Benutzer können beliebig viele Rollen zugeordnet werden. Das DITF-MR-Recherchesystem bringt vordefinierte Rollen für den Systemzugriff von Haus aus mit. Die Rolle *Zugriffsberechtigter* erlaubt einem Benutzer die Anmeldung am System. Ohne diese Rolle sind die Informationen zu der Person im System zwar sichtbar, ein Zugriff auf das System ist für diese Person aber nicht möglich. Zur Systemkonfiguration dient die Rolle *Administrator*. Personen mit dieser Rolle ist es möglich Informationsobjekttypen und Beziehungstypen anzulegen, Backups zu erstellen und einzulesen, Systemeinstellungen vorzunehmen sowie ein Aktionsprotokoll einzusehen. Für den lesenden Zugriff sowie für Rechercheaktivitäten steht die Rolle *Leser* zur Verfügung. Mit der Rolle *Informationspfleger* können Personen Informationsobjekte anlegen und ändern. Das Löschen von Objekten ist dieser Rolle nicht gestattet. Hierzu ist die Rolle *Informationspfleger mit Löschberechtigung* erforderlich. Die drei letztgenannten sind vordefinierte Rollen des Systems für die beschriebenen Berechtigungen. Da aber, wie Abbildung 98 zeigt, für jeden Informationsobjekttyp die Zugriffsrechte einzeln vergeben werden, ist es durchaus möglich für das Lesen, Ändern und Anlegen sowie das Löschen auch anderen Rollen entsprechende Rechte zu erteilen. Hierzu können selbstverständlich auch neue Rollen angelegt werden. Alle Zugriffsrechte eines neu angelegten Informationsobjekttyps sind zu Beginn auf *Administrator* gesetzt und sollten deshalb nach der Fertigstellung des Typs entsprechend geändert werden.

Zugriffsrechte verwalten

Zugriffsrechte von Rollen auf Objekte dieses Typs ändern

Rolle mit Lese- und Recherche-Rechten: **Leser**

Rolle mit Anlege- und Änderrechten: **Informationspfleger**

Rolle mit Anlege-, Änder- und Löschrchten: **Informationspfleger mit Löschberechtigung**

Gelegentlich kann es erforderlich sein, die Schreib- und Löschrchte einiger Informationsobjekte auf eine spezielle Rolle einzuschränken. Beispielsweise, wenn Informationsobjekte von verschiedenen Abteilungen gepflegt werden sollen.

Wenn Sie dies ermöglichen wollen, gehen Sie bitte folgendermaßen vor:

1. Legen Sie bitte zwischen diesem Informationsobjekttypen und dem Typen «Rollen» einen neuen Beziehungstyp – beispielsweise "darf ändern und löschen" – an.
2. Fügen Sie ein Attribut vom Typ «Relationen» zu diesem Informationsobjekttyp hinzu und stellen Sie dort den angelegten Beziehungstyp ein.
3. Kehren Sie zur Rechteverwaltung zurück. Wählen Sie den angelegten Beziehungstyp in der untenstehenden Liste aus.
4. Wenn Sie nun die Schreib- und Löschrchte eines Informationsobjekts auf eine oder mehrere spezielle Rollen beschränken möchten, so legen Sie einfach eine Beziehung zwischen dem Informationsobjekt und der Rolle an.

Beziehungstyp für eingeschränkte Schreib- und Löschrchten: **— keine Einschränkung der Schreib- und Löschrchten —**

Übernehmen **Verwerfen**

Abbildung 98: DITF-MR-RS Informationsobjekttyp - Zugriffsrechte

Beim Anlegen eines Beziehungstypen lassen sich auch Benutzerrechte vergeben. Der Anwender hat die Auswahl zwischen den Rechten des Start-Typs, den Rechten des End-Typs und einer Kombination aus beidem. Diese Rechtevergabe dient zur Festlegung der Rollen, die eine Beziehung diesen Typs Einsehen, Anlegen, Ändern oder Löschen dürfen.

2.5.7 Useability

Das DITF-MR-Recherchesystem bietet vor allem bei der Suche nach Informationen eine einfache und leicht verständliche Handhabung. Die Definition der Filterregeln erfolgt intuitiv und die Aufnahme von Informationsobjekten in eine Merkliste ist eine durchdachte und eine bisher so nicht vorgefundene Funktionalität bei semantischen Wikis.

Das Anlegen von Informationsobjekttypen gestaltet sich dagegen ein wenig aufwändiger und bedarf genauer Kenntnisse der Abläufe und Strukturen des DITF-MR-Recherchesystems. So hat es sich zum Beispiel als vorteilhaft erwiesen nach dem Erstellen eines Typs erst die Beziehungen zu definieren bevor mit der Einrichtung der Attribute begonnen wird, da sonst bei der Definition von Relations-Attributen ein häufiges Wechseln zwischen verschiedenen Eingabemasken die Folge ist. Die Möglichkeit direkt bei der Erstellung eines solchen Attributs für berechnete Personen auch die Anlage eines neuen Beziehungstyps zuzulassen, könnte hier Abhilfe schaffen.

Weiterhin scheint die Skalierung des Systems fraglich und bisher nicht getestet worden zu sein. In [Re08b S.97] wird ein älterer Rechner (512MB RAM) mit Linux Betriebssystem für ca. 50 Benutzer als „locker ausreichend“ bezeichnet. Zur Evaluierung, ob sich das DITF-MR-Recherchesystem auch für Benutzerzahlen größer 100 eignet, ist die Leistungsfähigkeit des Systems auf aktueller Hardware mit einem vorher zu definierendem Benutzerverhalten durchaus von Interesse.

3 Diskussion der Konzepte / Empfehlung

In diesem Kapitel werden die unterschiedlichen Konzepte und Eigenschaften wie Datenstrukturen und Benutzeroberflächen der betrachteten Wikis miteinander verglichen und diskutiert. Vor- und Nachteile einzelner Ansätze werden dabei herausgearbeitet. Abschließend werden Empfehlungen für die Erstellung eines semantischen Wikis gegeben.

3.1 Modell / Struktur

Im Folgenden werden einzelne Modell- bzw. Strukturmerkmale der Wikis betrachtet. Hierzu zählen die *Typisierung*, die Verwendung von *Vorlagen*, *Kategorien* und *Tags* sowie das *Beziehungskonzept*. In den weiteren Ausführungen dient der Ansatz, dass Metainformationen grundsätzlich nur zu Wiki-Seiten (also Dokumenten) hinzugefügt werden können, als Betrachtungsgrundlage. Andere Ansätze beschäftigen sich auch mit der Annotation von Abschnitten, Sätzen oder gar Worten, welches allerdings die Komplexität sowohl der Anwendung als auch der Modelle und Konzepte wesentlich erhöht. Auch bei allen betrachteten Wiki-Engines stellen Wiki-Seiten die attributierbaren Informationsobjekte dar, was bei der Evaluierung als ausreichend erachtet wurde.

3.1.1 Vorlagen und Typensystem

Die Frage der Typisierung von Informationsobjekten ist in den betrachteten Wikis auf unterschiedliche Art und Weise gelöst. Zum Einen gibt es eine strenge Typisierung im DITF-MR-Recherchesystem. Das Anlegen einer Wiki-Seite ist ohne die vorherige Erstellung eines entsprechenden Typs nicht möglich. Somit lässt sich dieses System dem in [Roh06 S.4] genannten, auf technischen Architekturentscheidungen beruhenden Designansatz zuordnen. Auch andere semantische Wikis verfolgen eine strenge Typisierung: „Mit OntoWiki haben wir [...] klassisches Knowledge Engineering um Wiki- und Community-Ansätze [...] [erweitert], anstatt Wikisysteme um Semantik.“ [DAR06]

Einen umgekehrten Ansatz verfolgt die Hypertext Knowledge Workbench. Es gibt außer den vom System vorgegebenen Typen *NameItem*, *ContentItem*, *Relation* und *Statement* keine Möglichkeit weitere Typen zu erstellen. Somit hat der Benutzer völlig freie Hand in der Gestaltung und Ausprägung seiner hinterlegten Informationen. Dies bietet auf der einen Seite die Möglichkeit alle Elemente exakt nach den Anwenderbedürfnissen anzulegen auf der anderen Seite die Gefahr Objekte gleichen Typs völlig unterschiedlich zu gestalten. Heißen Beziehungen oder Annotationen bei gleichartigen Objekten aber völlig unterschiedlich, wirkt sich dies zum Beispiel bei Suchen oder Abfragen negativ aus. Eigentlich typgleiche Elemente lassen sich nun nicht mehr finden oder vergleichen und die Idee der semantischen Informationsspeicherung ist ad absurdum geführt.

Einen Mittelweg zwischen den beiden genannten Möglichkeiten zeigt das Semantic MediaWiki auf. Das Konzept der *Vorlagen* bietet zum einen die Möglichkeit eine Wiki-Seite mit einer vordefinierten Struktur semantischer Informationen zu versehen, zwingt zum anderen aber keinen Nutzer Wiki-Seiten mit einer Vorlage versehen zu müssen. Verwendet der Anwender keine Vorlage, kann er frei von Vorgaben seine Wiki-Seite mit zusätzlichen semantischen Informationen anreichern. Da dieses Konzept dem Benutzer sowohl Freiraum gestattet, ihn aber auch, sofern er es wünscht, bei der Annotation unterstützt, lässt es sich einem aus Anwendersicht getriebenen Designansatz zuordnen, welcher ebenfalls in [Roh06 S.4] aufgeführt ist.

Der letztgenannte Ansatz eignet sich am besten, um von unstrukturierten über teilweise strukturierte zu vollständig strukturierten Inhalten zu gelangen. Der Benutzer kann ein Wiki mit diesem Ansatz zum Anfang wie eine „normales“ Wiki ohne semantische Informationen verwenden ohne sich

vorher im Klaren über die Struktur der Inhalte sein zu müssen. In der weiteren Entwicklung der Inhalte ist es dann möglich Annotationen hinzuzufügen und diese später zu einer Vorlage auszubauen. Somit ist auch ein nicht erfahrener Anwender in der Lage das Wiki, abgesehen von der Einarbeitung in die Wikisyntax, sofort zu verwenden ohne sich vorher mit der Strukturierung des Inhalts beschäftigen zu müssen, wie es bei streng typisierten Systemen, zum Beispiel dem DITF-MR-Recherchesystem oder OntoWiki, notwendig ist.

3.1.2 Kategorien und Tags

Zur Klassifizierung von Wiki-Seiten gibt es in den vorgestellten Systemen unterschiedliche Ansätze. Im DITF-MR-Recherchesystem ist zusätzlich zum Typensystem keine dieser Möglichkeiten gegeben. Das Semantic MediaWik verwendet ein Kategoriesystem, welches erlaubt verwandte Inhalte der selben Kategorien zuzuordnen und mit Hilfe der Kategorien-Seiten einen wesentlichen Beitrag zur einfacheren Nutzung bietet. „Categories [...] are mainly used to assist browsing“ [Vö06 S.3]. Sowohl die Hypertext Knowledge Workbench als auch NEPOMUK lassen das explizite Hinzufügen von Schlüsselwörtern, sogenannten *Tags*, zu Informationsobjekten zu. NEPOMUK bietet zusätzlich eine Bewertungsfunktionalität mit einer Skala von eins bis fünf Sternen.

Die Idee hinter Kategorien und Tags besteht in der Zuordnung einer Wiki-Seite zu objektiven Themengebieten und Sachverhalten oder subjektiven Meinungen und Bewertungen. In [GH06] beschreiben Golder und Hubermann am Beispiel des Internetdienstes del.icio.us (delicious.com) das gemeinschaftliche Hinzufügen von Metadaten in Form von Schlüsselwörtern zu gemeinsam genutzten Inhalten in Form von Bookmarks. In dieser Arbeit identifizieren sie unterschiedliche Funktionen der Tags bezüglich des entsprechend annotierten Inhalts [GH06 S.5]:

- Beschreibungen der Themenfelder; welcher Sachverhalt, Person oder Organisation wird beschrieben
- Beschreibung der Art (Wiki-Seite, Artikel, Buch, Blog, ...)
- Beschreibung des Besitzers oder Autors
- verfeinernde oder einschränkende Kategorien zu bestehenden Beschreibung
- Charakterisierung oder Qualitätsbeschreibung (lustig, aufschlussreich, interessant, ...)
- Referenzierung des Tag-Erstellers (mein Artikel, mein Kommentar)
- Organisationsaspekte (zu lesen, bearbeiten, kommentieren, ...)

Alle diese Funktionen sind zwar in [GH06] auf Bookmarks bezogen, lassen sich aber auch auf Wiki-Systeme und andere Inhalte in elektronischer Form anwenden. So bilden ID3-Tags [Wi09a] zur Beschreibung von MP3 Audiodateien und die IPTC-Daten [Wi09b] zur Speicherung von Textinhalten zu Bilddateien zumindest einen Teil der oben genannten Information ab.

Die sieben genannten Funktionen lassen sich offensichtlich in zwei Gruppen einteilen. Die ersten vier bezeichnen Information, die nicht nur für den Ersteller sondern für alle Benutzer von Interesse sein können. Im Gegensatz hierzu sind die anderen drei Funktionen auf den Ersteller dieses Tags bezogen und für ihn aussagekräftig und relevant. Charakterisierung und Qualitätsbeschreibung können unter Umständen eine Ausnahme bilden und auch für andere Benutzer als Hinweis für eine eigene Bewertung dienen „[...] if many users find something funny, there is a reasonable likelihood some else would also find it to be so [...]“ [GH06 S.7]. Der entscheidende Aspekt für die Relevanz von Tags die der Sinnhaftigkeit (*Sensemaking*) für den jeweiligen Anwender [GH06 S.3]. Festzustellen bleibt, dass es sowohl öffentliche als auch private Tags gibt. Im Wiki-Umfeld sollten öffentliche Tags als nur einmal pro Informationsobjekt existent und für allen Benutzern sicht- und bearbeitbar definiert werden. Private Tags hingegen sollten einmal pro Benutzer existieren können, wobei es dem Anwender überlassen bleibt, ob alle, nur bestimmte Benutzer oder nur er allein diese Tags betrachten kann.

Ist in einem Wiki das Tagging, wie im DITF-MR-Recherchesystem, nicht vorhanden, ist es dem Benutzer nicht möglich für ihn relevante Informationen an einer Wiki-Seite anzumerken. Sind aber diese Metainformationen für den Anwender in seiner täglichen Arbeit mit dem Wiki von großer Bedeutung, kann dies zu Akzeptanzproblemen führen. Die als öffentlich bezeichneten Tags können jedoch durch die Typisierung und automatisch gespeicherte Metadaten, zum Beispiel „zuletzt editiert am“, abgedeckt werden.

Auf der anderen Seite besteht bei Wiki-Systemen mit Taggingmechanismus die Gefahr, dass die Typisierung einer Wiki-Seite mittels Tags und nicht durch die eventuell dafür vorgesehene Vorlage respektive den Typ erfolgt. Um dieses Risiko zu minimieren, bedarf es einer sinnvollen Benutzerführung. Eine Lösung wäre die ausschließliche Zulassung von vorab definierten und aus einer Liste auswählbaren Tags. Mit einem Vergleich zwischen den verwendeten Tags und den definierten Attributen von Vorlagen könnten dem Benutzer passende Vorlagen zur Strukturierung der Inhalte vorgeschlagen werden. Ebenso ließe sich der Anwender auf eine größere Anzahl Wiki-Seiten mit gleicher Vorlage und gleichen Tags hinweisen, um zu überprüfen, ob es sich bei dem Tag nicht doch um ein Attribut handeln könnte. Eine weitere unwesentlich schwerer realisierbare Idee stellt die systemseitige Unterstützung zum Übergang Tag – Vorlage dar. Hier bedarf es einer intelligenten Mustererkennung um aus mit gleichen oder ähnlichen Tags versehenen Informationsobjekten eine Vorlage für eben jene zu erstellen. Die Ermittlung und Änderbarkeit der Ähnlichkeitsdefinition ist dabei das entscheidende Kriterium.

Tagging ist in vielen Internetdiensten ein zentraler Bestandteil, birgt aber die Gefahr, dass Attribute und Tags konkurrierend eingesetzt werden und somit die Strukturierung der Inhalte darunter leidet. Dennoch sollte eine geeignet eingeschränkte Verschlagwortung von Wiki-Inhalten möglich sein, um den Benutzern das Hinzufügen vor allem persönlich relevanter Metainformationen zu ermöglichen, denn auch Golder und Hubermann stellen fest, dass Tags mehr für den persönlichen als den gemeinschaftlichen Nutzen eingesetzt werden: „a significant amount of tagging, if not all, is done for personal use rather than public benefit“ [GH06 S.7].

3.1.3 Beziehungskonzept

Die Beziehungen zwischen den einzelnen Objekten in einem Wiki können unidirektional oder bidirektional abgebildet werden. In den meisten der betrachteten Systeme werden Beziehungen beim Erstellen sofort in beide Richtungen definiert. Ein Beispiel für eine Beziehung mit der dazu gehörigen Inversen lautet:

- Prof. Dr. Florian Matthes **veranstaltet** SEBA Bachelor.
- SEBA Bachelor **wird veranstaltet von** Prof. Dr. Florian Matthes.

Somit ist die beidseitige Navigierbarkeit von Anfang an gegeben und der Informationsgehalt des Wikis erweitert sich per gepflegter Beziehung nicht nur um eine sondern um zwei Aussagen. Die Definition getrennter entgegengesetzter Beziehungen, sofern diese sich überhaupt als Inverse kennzeichnen lassen, entfällt. Für die Suche nach Objekten und Zusammenhängen stellt die beidseitige Navigierbarkeit ebenfalls einen Vorteil dar, da der Ausgangspunkt der Suche nicht auf eine Beziehungsseite beschränkt ist. „This trivial mechanism has nevertheless the capability to build automatically symmetric cross-referencing and navigation from one card / page to the other. It is also extremely useful to express queries and rules in a symmetric way.“ [Roh06 S.6]

Die beschriebenen Vorteile lassen für neu zu erstellende Wiki-Systeme den Schluss zu, dass diese über inverse Beziehungen verfügen sollten. „I am always very surprised to notice that the notion of inverse relation is nearly absent in semantic tools. It was one of our first decisions in the design of Ideliance.“ [Roh06 S.6]

3.2 Benutzeroberfläche

Das Design der Benutzeroberfläche ist für die Akzeptanz und somit die Verwendung eines Wiki-Systems ein sehr wichtiges Kriterium. Betrachtet werden in diesem Zusammenhang die Eingabemöglichkeiten der Inhalte.

3.2.1 Eingabearten

Zur Eingabe von Inhalten und Metadaten stehen in den untersuchten Wiki-Systemen drei grundsätzlich verschiedene Möglichkeiten zur Verfügung. Hierzu zählen die natürliche Sprache, die klassische Wiki-Syntax und die Eingabe mittels Formularfelder.

3.2.1.1 Natürliche Sprache

AceWiki verwendet zur Eingabe das Attempto Controlled English (ACE), eine kontrollierte natürliche Sprache. Jeder Englisch sprechende Anwender kann demzufolge zum einen dieses Wiki lesen und zum anderen auch ohne Kenntnis formaler Sprachen semantische Inhalte einfügen. Hierbei entsteht idealer Weise eine Wiki-Seite in vereinfachter natürlicher Sprache, welche in eine Ontologie überführt werden und zur weiteren automatisierten Verarbeitung verwendet werden kann.

Der oben genannte Vorteil der natürlichen Sprache ist aber auch gleichzeitig der größte Nachteil. Dieses Wiki-System ist auf eine spezielle Sprache festgelegt und kann nur mit dieser verwendet werden. Um solch ein Wiki zum Beispiel in deutscher Sprache zu verwirklichen, bedarf es der Ausarbeitung einer kontrollierten, im Umfang reduzierten Untermenge des deutschen Wortschatzes und einer vereinfachten deutschen Grammatik. Diese Aufgabe ist jedoch wohl auf Grund der Komplexität der deutschen Sprache den Linguistikern zu überlassen.

3.2.1.2 Formularfelder

Vor allem bei stark typisierten Wikis, wie dem DITF-MR-Recherchesystem, ist die Eingabe mit Hilfe von Formularfeldern beliebt und sinnvoll. Die Eingabefelder erwarten einen festgelegten Typ an Information und können so zum einen die Eingabe sofort auf Korrektheit prüfen und zum anderen dem Benutzer auch Vorschläge für mögliche Inhalte unterbreiten. So werden zum Beispiel bei Assoziationen auf Personen auch nur Personen zur Auswahl angeboten. Für Datumseingaben ist eine Kalenderdarstellung wie bei NEPOMUK denkbar, aus welcher der Anwender direkt das gesuchte Datum auswählen kann. Typprüfung und Eingabeunterstützung sind die großen Vorteile der Verwendung von Formularfeldern bei der Inhaltserstellung. Für frei editierbare Textinhalte verwenden sowohl das DITF-MR-Recherchesystem als auch die Hypertext Knowledge Workbench spezielle Typen, die mit einem Rich Text Editor bearbeitet werden können. Die Einbettung der anderen definierten Attribute ist aber nur in der Sichtenerstellung des DITF-MR-Recherchesystems möglich. Die freie Positionierbarkeit der Attribute im Wiki-Text ist, wie bei NEPOMUK, nicht gegeben und stellt einen deutlichen Nachteil vieler Formularfelder verwendender Wikis dar. Meist werden die Attribute alphabetisch oder nach der Reihenfolge der Eingabe sortiert.

3.2.1.3 Wiki-Syntax

Laut [Sw08g] verwenden zehn der 18 dort aufgeführten Wiki-Engines eine Wiki-Syntax zur Hinterlegung von Metadaten zu einer Wiki-Seite. Semantic MediaWiki, als Beispiel hierfür, verdeutlicht sowohl die Vor- als auch die Nachteile dieser Eingabeform. Die direkte Eingabe ermöglicht dem Benutzer, die Kenntnis der benötigten Annotationen vorausgesetzt, ein zügiges und effektives Arbeiten ohne Umwege über Eingabemasken und Formularfelder. Je nach Speicherung

der Wiki-Seiten und vorhandener Versionsverwaltung sind auch die Änderungen der Metadaten dokumentiert, ohne hierfür dem Wiki-System neue Funktionalität hinzufügen zu müssen. Wie bereits angedeutet, existiert jedoch keinerlei Unterstützung beim Einfügen von Attributen oder Vorlagen. Der Anwender muss daher die Attribute und Vorlagen sehr genau kennen, um nicht versehentlich ein weiteres Attribut anzulegen dessen Bedeutung und Verwendung mit einem schon bestehenden Attribut identisch ist. Eine Typprüfung ist bei der Verwendung der Wiki-Syntax bisher ebenso nicht vorhanden wie eine Eingabeunterstützung bei der Wertzuweisung zu Attributen.

3.2.1.4 Hybride Eingabeform

Die Zusammenführung der Ansätze mit Wiki-Syntax und Formularfeldern könnte die Vorteile beider Konzepte vereinen. Denkbar wäre eine Eingabe mittels Wiki-Syntax allerdings mit einer Eingabeunterstützung ähnlich der Eingabehilfe bei Entwicklungsumgebungen, wie zum Beispiel Eclipse. Die semantischen Informationen wären in der Wiki-Syntax extra ausgezeichnet, vergleichbar zum SMW, zum Beispiel durch `[[...]]`. Gibt der Anwender eine Zeichenkette beginnend mit `[[` ein, erkennt der Editor diese und bietet die verfügbaren Attribute zur Auswahl an. Entsprechend der weiteren Eingabe wird die Auswahl weiter eingeschränkt. Darüber hinaus würde, ähnlich dem SMW, zu einer Wiki-Seite eine Faktenliste angezeigt, die auch zur Modifizierung der Attribute verwendet werden könnte. Es entfielen bei Änderung über die Faktenliste die, vor allem bei umfangreichen und unübersichtlichen Inhalten notwendige, Suche nach dem Vorkommen des Attributs im Wiki-Text. Zudem könnte hier eine Typprüfung durchgeführt werden.

Um die Verwendung von Formularfelder schon bei der Eingabe und nicht erst bei Änderungen zu ermöglichen, bedarf es eines weiterführenden Ansatzes. Neue Attribute werden per Formularfeld erstellt und anschließend in einem speziell gekennzeichneten Bereich des Wiki-Seiten-Quelltextes gespeichert. Somit bleibt auch die direkte Eingabe im Quelltext möglich. Dieser Bereich wird in der späteren Ansicht der Wiki-Seite nicht angezeigt. Die Einbindung der Attributwerte in den Wiki-Text kann, ähnlich der Sichtenerstellung im DITF-MR-Recherchesystem, mittels Platzhaltern erfolgen. Diese sind den Attributen zum Beispiel in dieser Form `[[property::value::placeholder]]` zuzuweisen.

Zu diskutieren bliebe, ob das Anlegen und Bearbeiten von Attributen ausschließlich in den Formularfeldern oder parallel auch mit der direkten Eingabe im Wiki-Seiten Code erfolgen soll. Existieren beide Möglichkeiten, ist zudem zu entscheiden, ob Attribute nur an einer festgelegten Stelle oder überall im Wiki-Text spezifiziert werden dürfen. Hieraus ergibt sich ob für die Verwendung der Werte immer Platzhalter zu verwenden sind oder ob der vom SMW bisher verfolgte Ansatz gleichberechtigt verwendet werden kann.

Die oben beschriebene Idee bietet, bei Verwendung der Formularfelder, eine Typprüfung schon bei der ersten Wertzuweisung. Es bleibt dennoch dem Anwender überlassen mit Hilfe welchen Mechanismus er Wiki-Seiten erstellt. Die Versionierung der Attribute ist ohne zusätzlichen Aufwand allein durch die vorhandene Versionierung der Inhalte der Wiki-Seite möglich, sofern die Attribute als Bestandteil des Wiki-Textes gespeichert werden. Für die Verwendung von Platzhaltern ist eine Eingabeunterstützung, ähnlich der für die Attribute vorgeschlagenen, denkbar. In einer von der Eingabeunterstützung bereitgestellten Auswahlliste könnten ebenfalls automatisch erstellte Metadaten wie Änderungszeitpunkt, Autor, etc. angeboten werden. Auch die Vorlagenauswahl sollte durch eine geeignete Auswahlliste möglich sein, so dass der Benutzer keine separate Wiki-Seite nach der richtigen Vorlage durchsuchen muss.

3.3 Suchen / Abfragen

Suchen und Abfragen sind wichtige Eigenschaften für Wikis im allgemeinen und semantische Wikis im Besonderen. In diesem Abschnitt werden Such- und Abfragefunktionen erörtert, die ein semantisches Wiki neben der Suche nach Artikelname sowie der Volltextsuche aufweisen sollte.

3.3.1 Semantische Suche

Ein fortgeschrittenes Beispiel für eine semantische Suche stellt SMW bereit. Hier ist es möglich Wiki-Seiten mit festgelegten Attributen zu suchen. Der Wert der Attribute kann mittels Operatoren und Wildcards genauer beschrieben werden. Zudem ist es möglich weitere Eigenschaften der Wiki-Seite in der Trefferliste anzuzeigen. Dieser Ansatz ist leicht verständlich und schnell erlernbar. Durch eine Eingabeunterstützung in Form von kontextabhängigen Auswahllisten, zum Beispiel für verfügbare Attribute oder Kategorien, könnte die Handhabbarkeit zusätzlich verbessert werden.

Der Einsatz einer standardisierten Abfragesprache wie SPARQL (vgl. NEPOMUK) eignet sich für fortgeschrittene Anwender, die sowohl Erfahrung aus dem Bereich der Ontologien als auch im Bereich von Abfragesprachen wie SQL mitbringen. Auf Grund des Aufbaus und der Funktionalität lassen sich mit einer solchen strukturierten Abfragesprache komplexere Sachverhalte zur Suche verwenden.

3.3.2 Suchergebnisse in Wiki-Seiten einbetten

Zur Erstellung dynamischer Inhalte in Wiki-Seiten ist die Einbettung von Suchergebnissen in die Darstellung einer Seite und somit die Einbettung von Abfragen in den Quelltext unerlässlich. Auch hier bietet SMW ein Beispiel, welches nicht nur die Abfrage anhand der Attribute sondern auch die Bestimmung weiterer anzuzeigender Informationen sowie die Sortierung und Limitierung der Trefferzahl zulässt. Für häufiger verwendete Anfragen bietet SMW den Ansatz der *Concepts*, eine Art vordefinierter Anfrage, die wiederum durch Namensreferenz in eine Wiki-Seite eingebunden werden können.

Die dynamische Erstellung von Inhalten ist einer der entscheidenden Vorteile semantischer Wikis. So können zum Beispiel die im Wiki beschriebenen Vorlesungen eines Universitätsmitarbeiters durch eine einfache Abfrage auf dessen persönlicher Wiki-Seite eingebunden werden. Ein manuelles und damit doppeltes Pflegen dieser Daten entfällt. Nach dem Eintrag einer neuen Vorlesung in das Wiki ist auch die Vorlesungsauflistung der Wiki-Seite des Mitarbeiters auf dem neuesten Stand.

3.3.3 Merkliste

Die im DITF-MR-Recherchesystem vorgestellte Merkliste besitzt einige interessante Aspekte für die Suche in aber auch für die Bearbeitung von Wiki-Seiten.

Die Speicherung von Wiki-Seiten in einer Merkliste bedeutet für den Anwender einen höheren Komfort bei der Suche. Relevante Inhalte müssen nicht erst als Lesezeichen abgelegt oder in einer externen Anwendung gespeichert werden um für eine spätere Bearbeitung zur Verfügung zu stehen. Die Vergleichsfunktion der vorgestellten Merkliste erlaubt dabei eine Abfrage mit der Ausgabe entsprechender Attribute. Durch die einfacheren Erstellung des Vergleiches ist sie jedoch benutzerfreundlicher als die Erstellung der entsprechenden Abfrage. Werden allerdings kaum typisierte Inhalte verglichen, stellt sich die Frage nach der Anzeige der verwendeten Attribute. Hier sollte es dem Anwender überlassen bleiben, ob alle verwendeten, nur gemeinsam verwendete oder nur Attribute mit einer Mindestanzahl an Verwendungen angezeigt werden.

Bei der gleichzeitigen Änderung mehrerer Wiki-Seiten ließen sich nur Attribute bearbeiten, die auch in allen Seiten bereits vorhanden sind. Beim DITF-MR-Recherchesystem stellt dies kein Problem dar, da nur Objekte gleichen Typs gemeinsam bearbeitet werden können. Für weniger stark typisierten Wikis bleibt zu klären, ob nur bei allen Objekten vorhandene Attribute bearbeitet werden können. Andererseits ist auch das Hinzufügen des gemeinsam bearbeiteten Attributes zu Objekten, welche dieses bisher nicht besitzen, eine denkbare Lösung. Die Entscheidung über die Verfahrensweise kann und sollte dem Anwender überlassen werden.

3.3.4 Synonyme

Wie im Abschnitt über Kategorien und Tags schon angesprochen, ist das Hinzufügen von Schlagwörtern immer abhängig vom Benutzerkontext und den persönlichen Vor- und Einstellungen des Anwenders. Hinzu kommt der Umstand, dass unterschiedliche sprachliche Ausdrücke die gleiche oder eine ähnliche Bedeutung aufweisen können [Wi09c]. Für die Suche bedeutet dies, dass das Suchergebnis nur deshalb nicht die Ausprägung hat, die es haben könnte, weil bei Suche und Verschlagwortung jeweils unterschiedliche Begriffe mit der selben Bedeutung verwendet wurden. In [Vö06 S.9] wird diese Problemstellung noch weiter gefasst, auf Internationalität ausgeweitet und für unterschiedliche Sprachen betrachtet. Ein Lösungsvorschlag ist die Verwendung des Online-Wörterbuches Wiktionary [Wn09] zum einen zur Suche nach Synonymen und zum anderen zur Übersetzung.

3.4 Weitere Funktionen

Der folgende Abschnitt beschäftigt sich mit weiteren Funktionen, die ein Wiki aufweisen könnte.

3.4.1 Maßeinheiten mit Umrechnung

Die Funktionalität des SMW selbst definierten Datentypen ein oder mehrere Maßeinheiten zuzuordnen, ist unabhängig vom Verwendungszweck des Wikis ein nützliches und anwenderfreundliches Merkmal. Die automatisch Umrechnung zwischen verschiedenen Einheiten eines Typs ermöglicht Autor und Leser die Verwendung der jeweiligen bevorzugten Bezeichnungen ohne konkrete Umrechnungsfaktoren kennen zu müssen.

3.4.2 Sichten

Zur Darstellung spezieller Sachverhalte gibt es in einigen Wikis besondere Ansichten. Beliebte Beispiele hierfür sind zum einen Kalender und zum anderen geografische Koordinaten. NEPOMUK und OntoWiki [DAR06 S.3] bieten jeweils eine eigene Sicht für Kalenderdaten und Koordinaten als festen Bestandteil der Anwendung an. SMW beschreitet hierfür den Weg der definierbaren Plugins zur Verwendung von Internetdiensten. Während der NEPOMUK oder OntoWiki Anwender nur die vorgegebenen Ansicht verwenden kann, ist es SMW Nutzern möglich jeden beliebigen, die Anforderungen erfüllenden Internetdienst zu konfigurieren und zu verwenden. Sogar das Einbinden mehrerer Dienste für die selbe Aufgabe ist möglich. Durch die Erweiterbarkeit ist der Anwendungsbereich auch nicht auf Kalender und Orte beschränkt.

3.4.3 Bewertung von Inhalten

Bewertungskriterien stellen, wie bereits erwähnt, eine Sonderform des Taggings dar. Wie die Bewertung einer Wiki-Seite erfolgen kann, ist dabei verschieden. Grundsätzlich gilt zum Beispiel zu entscheiden welche Benutzer bewerten dürfen, ob es verschiedene Bewertungskriterien gibt und woran diese gebunden sind (allgemein gültig, Vorlage, Kategorie, etc.). Bewertungen jedoch stellen

kein originäres Problem semantischer Wikis dar. Für eine weitergehende Behandlung des Themas sei auf das Beispiel OntoWiki [DAR06 S.5f] verwiesen.

3.4.4 Export / Import

Auch Export und Import stellen wesentliche Wiki-Funktionen dar. Dabei kann im Bereich der Ontologien auf bereits seit längerem existierende Standards wie RDF/XML, OWL oder N3 [W3C08] zurückgegriffen werden.

4 Zusammenfassung und Ausblick

Abschließend werden die wesentlichen Erkenntnisse aus der vorliegenden Arbeit kurz zusammengefasst sowie weitere interessante Forschungsrichtungen, welche sich in Folge ergeben, vorgestellt.

4.1 Kritische Würdigung

Im Rahmen der Arbeit erfolgte die Betrachtung von fünf sehr unterschiedlichen Ansätzen ein semantisches Wiki zu realisieren. Ausgewählt wurden das *Semantic MediaWiki*, *NEPOMUK*, die *Hypertext Knowledge Workbench*, *AceWiki* sowie das *DITF-MR Recherchesystem*. SMW ist eines der bekanntesten semantischen Wiki-Systeme und stellt eine Erweiterung zum MediaWiki dar, welches durch die Verwendung beim den Wikipedia-Projekten einen hohen Bekanntheitsgrad erlangt hat. Bei NEPOMUK und HKW handelt es sich vorrangig um sogenannte *Social Semantic Desktop* Systeme, welche dem Benutzer die Strukturierung von Information auf verschiedensten Abstraktionsniveaus erlauben. Beide Systeme basieren auf der *Conceptual Data Structures* als Modell zur Datenstrukturierung. AceWiki verwendet zur zur Modellierung und Darstellung der Wiki-Inhalte die strukturierte natürliche Sprache *Attempto Controlled English*. Das von den betrachteten am strengsten strukturierte Wiki ist das DITF-MR-Recherchesystem. Dieses Wiki wurde vorrangig zur Suche und Recherche in gespeicherten Webmaschineneinstellung entwickelt, wird aber mittlerweile auch anderweitig verwendet.

Diese unterschiedlichen Ansätze beschreiben sehr gut das Spektrum der Varianten und Möglichkeiten ein semantisches Wiki-System zu erstellen. Aufbauend auf den Erkenntnissen und Erfahrungen aus der Evaluation der Wiki-Systeme konnten diese anhand einzelner Merkmale verglichen und bewertet werden. Hieraus ergeben sich die Empfehlungen für die Entwicklung eines am Lehrstuhl sebis eingesetzten Wiki-Systems. Hierzu zählen die Verwendung einer vornehmlich auf Vorlagen basierenden Strukturierung der Inhalte, eines eingeschränkten Tagging-Systems sowie eines Beziehungskonzeptes mit inversen Beziehungen. Für die Oberflächengestaltung wird eine hybride Eingabeform mit einer Mischung aus Formularfeldern und Wiki-Texteingaben favorisiert. Hierbei soll zur Erstellung von dynamischen Wiki-Seiten auch das direkte Einbetten von Abfragen in den Wiki-Seiten-Quelltext möglich sein. Zur weiteren Verbesserung der Benutzerfreundlichkeit sind auch eine Merklistenfunktionalität zum Vergleich und gemeinsamen Bearbeiten von Objekten sowie die Einbindung eines Dienstes zur Synonymfindung und Übersetzung bei Suchanfragen wünschenswert. Zusätzlich denkbare Funktionen sind die automatische Umrechnung von Maßeinheiten, Anbindung von Internetdiensten zum Beispiel zur Visualisierung von geografischen Koordinaten sowie die Bewertungsmöglichkeit von Inhalten.

Auf Grund des gesetzten zeitlichen Rahmens wurde von einer noch umfangreicheren Evaluierung bestehender Wiki-Systeme sowie der Validierung der Empfehlungen anhand einer prototypischen Implementierung Abstand genommen. Zur Realisierung eines konkreten Wiki-Systems bedarf es zudem der exakten Festlegung des gewünschten Funktionsumfangs, der Eingabemöglichkeiten und, damit verbunden, der Definition der Datenstrukturen. Auch Fragen bezüglich der Einbindung multimedialer Inhalte, wie Bilder, Videos und Audiodateien sowie anderer Dokumente wie pdf-Dateien bleiben in diesem Zusammenhang zu klären.

4.2 Künftige Forschungsfragen

Im Rahmen dieser Arbeit konnte ein Überblick über die Vielfalt und das Spektrum bestehender semantischer Wikis gewonnen werden. Die Aufgabe weiterführender Arbeiten ist es nun zum einen weitere, potentiell relevante Wiki-Systeme zu betrachten sowie zum anderen die gegebenen

Empfehlungen in einem prototypischen System umzusetzen.

Zur Auswahl weiterer Wikis bietet [Sw08f] als Einstieg eine Funktionsübersicht, deren Eigenschaften in [Sw09a] beschrieben werden. Grundlegende Merkmale semantischer Wikis werden in [Sw09b] diskutiert, wenngleich diese Quelle zuletzt 2006 geändert worden ist und damit einen veralteten Stand aufweisen dürfte. Ein weiterer Vergleich allgemeiner Eigenschaften ist in [SBB07 S.4] zu finden. Um einen Einblick in alle, auch nicht semantische Wiki-Engines, zu erhalten und diese auch auf einfache Art und Weise vergleichen zu können, eignet sich die Website des WikiMatrix Projektes (www.wikimatrix.org). Aus der immer größer werdenden Anzahl semantischer Wikis zur Betrachtung die richtigen herauszusuchen, ist keine einfache Aufgabe, die jedoch mit Hilfe der oben genannten Webseiten adressiert werden kann. Folgende System könnten für weitere Arbeiten von Interesse sein:

- IkeWiki und Nachfolger KiWi als ein globalerer Ansatz „More than a Wiki, this project is a general purpose, users-oriented platform for semantic information processing.“ [Roh06 S.5]
- OntoWiki als ontologiebasierter Ansatz [DAR06] und
- SweetWiki als ein an den Benutzerbedürfnissen orientiertes Wiki.

Auch in verschiedenen wissenschaftlichen Beiträgen, so zum Beispiel [SBB07] und [AL07], werden semantische Wikis verglichen und vorgestellt. Im Rahmen der Arbeit fiel darüber hinaus auf, dass ein Vergleich der Wiki-Systeme anhand der „Papierlage“ nicht ausreichend war um speziell die Vor- und Nachteile in der Benutzung zu evaluieren. Ein durchgängiges Beispiel, welches auf den Livesystemen durchgespielt wird, kann hier einen wesentlichen Beitrag zur Evaluation leisten.

Um Verbreitung und Akzeptanz von strukturierenden Wiki-Systemen auszuweiten und deren Nutzung zu fördern sind vorallem einfach und intuitiv zu bedienende Systeme mit einer simplen aber wirkungsvollen Benutzerführung notwendig. Nur wenn es gelingt eine große Anzahl technisch nicht versierter Internetnutzer zum Mitgestalten der Inhalte zu bewegen und von dessen Nutzen zu überzeugen, können semantische Systeme ihre Vorteile wie bessere Strukturierung, automatische Verarbeitbarkeit, effizientere und genauere Suche sowie die Entdeckung neuer Zusammenhänge vollends entfalten.

Die Semantic Wiki Community beschreibt auf ihrer Website die Zukunft semantischer Wikis folgendermaßen: „In the future, Semantic Wiki systems might play an important role for 'knowledge acquisition', enabling non-technical users to contribute to the Semantic Web.“ [Se09]

Quellen:

- [Ac08a] AceWiki <http://attempto.ifi.uzh.ch/acewiki> zugegriffen: 2008-12-13
- [AL07] Auer, S.; Lehmann, J. „What have Innsbruck and Leipzig in common? Extracting Semantics from Wiki Content.“ In Franconi et al. (eds), *Proceedings of European Semantic Web Conference (ESWC'07)*, LNCS 4519, pp. 503-517, Springer, 2007.
- [At08a] Attempto <http://attempto.ifi.uzh.ch> zugegriffen: 2008-12-13
- [At08b] Attempto http://attempto.ifi.uzh.ch/site/docs/ace_constructionrules.html zugegriffen: 2008-12-14
- [Be04a] Tim Bernes-Lee „Semantic Web Tutorial Using N3“ <http://www.w3.org/2000/10/swap/doc/Overview.html> zugegriffen: 2009-01-18
- [Be06a] Tim Bernes-Lee „Notation 3“ <http://www.w3.org/DesignIssues/Notation3.html> zugegriffen: 2009-01-18
- [Be08a] Dave Beckett <http://www.dajobe.org/2005/04-sparql> zugegriffen: 2008-12-13
- [BHL01] Tim Berners-Lee, James Hendler, Ora Lassila: „The Semantic Web: a new form of Web content that is meaningful to computers will unleash a revolution of new possibilities“. In: *Scientific American*, 284 (5), S. 34–43, May 2001
<http://www.sciam.com/article.cfm?id=the-semantic-web>
- [DAR06] Sebastian Dietholz, Sören Auer, Thomas Riechert „Kolloborative Wissensarbeit mit OntoWiki“ <http://www.informatik.uni-leipzig.de/~auer/publication/WissensarbeitMitOntoWiki.pdf> zugegriffen: 2008-10-30
- [GH06] Scott A. Golder, Bernardo A. Hubermann „Usage Patterns of Collaborative Tagging Systems.“ *Journal of Information Science*, 32(2). 198-208, 2006,
<http://www.hpl.hp.com/research/idl/papers/tags/tags.pdf> zugegriffen: 2009-01-23
- [Ku08a] Kuhn, Tobias „AceWiki: A Natural and Expressive Semantic Wiki“ in *Proceedings of Semantic Web User Interaction at CHI 2008: Exploring HCI Challenges, CEUR Workshop Proceedings*, 2008
<http://attempto.ifi.uzh.ch/site/pubs/papers/kuhn08acewiki.pdf> zugegriffen: 2008-12-13
- [Ku08b] Kuhn, Tobias „AceWiki: Collaborative Ontology Management in Controlled Natural Language“ in *Proceedings of the 3rd Semantic Wiki Workshop, CEUR Workshop Proceedings*, 2008
http://attempto.ifi.uzh.ch/site/pubs/papers/kuhn08acewiki_onto.pdf zugegriffen: 2008-12-13
- [Ku08c] Kuhn, Tobias „Combining Semantic Wikis and Controlled Natural Language“ in *Proceedings of the Poster and Demonstration Session at the 7th International Semantic Web Conference (ISWC2008), CEUR Workshop Proceedings*, 2008
http://attempto.ifi.uzh.ch/site/pubs/papers/iswc2008pd_kuhn.pdf zugegriffen: 2008-12-13

- [Le08] Francesco Lelli, Leo Sauermann, Gunnar Aastrand Grimnes, Martin Klinkigt, Rodolfo Stecher, Robert Jaeschke, Karim Hajy, Malte Kiesel, Michael Loerscher, Gerald Reif, Cristian Bogdan, Enrico Minac „NEPOMUK users guide“ 24.10.2008
<http://nepomuk.semanticdesktop.org/xwiki/bin/download/Main1/D6-7A/D6.7.A-NEPOMUK-user-guide-v1.0.pdf> zugegriffen: 2008-12-07
- [Me08a] MediaWiki <http://www.mediawiki.org> zugegriffen: 2008-11-25
- [Me08b] MediaWiki <http://www.mediawiki.org/wiki/Help:Categories> zugegriffen: 2008-11-22
- [Me08c] MediaWiki http://meta.wikimedia.org/wiki/Main_Page zugegriffen: 2008-11-22
- [Me08d] MediaWiki http://www.mediawiki.org/wiki/Help:Sysops_and_permissions
 zugegriffen: 2008-12-04
- [Me08e] MediaWiki http://www.mediawiki.org/wiki/Help:Protecting_and_unprotecting_pages
 zugegriffen: 2008-12-04
- [Me08f] MediaWiki http://www.mediawiki.org/wiki/Manual:User_rights_management
 zugegriffen: 2008-12-04
- [Ne08a] Nepomuk <http://nepomuk.semanticdesktop.org> zugegriffen: 2008-12-06
- [Ne08b] Nepomuk
<http://nepomuk.semanticdesktop.org/xwiki/bin/view/Main1/Project+Objectives>
 zugegriffen: 2008-12-06
- [Ne08c] Nepomuk <http://nepomuk.semanticdesktop.org/xwiki/bin/view/Main1/Project+Summary>
 zugegriffen: 2008-12-06
- [Ne08d] Nepomuk <http://nepomuk.semanticdesktop.org/xwiki/bin/view/Main1/Participants>
 zugegriffen: 2008-12-06
- [Ne08e] Nepomuk <http://dev.nepomuk.semanticdesktop.org> zugegriffen: 2008-12-06
- [Ne08f] Nepomuk <http://nepomuk.kde.org> zugegriffen: 2008-12-07
- [Ne08g] Nepomuk <http://nepomuk-eclipse.semanticdesktop.org> zugegriffen: 2008-12-07
- [Ne08h] Nepomuk <http://nepomuk-mozilla.semanticdesktop.org> zugegriffen: 2008-12-07
- [Ne09] Netcraft January 2009 Web Server Survey
http://news.netcraft.com/archives/2009/01/16/january_2009_web_server_survey.html
 zugegriffen: 2009-02-07
- [[Re08a] “DITF Recherchesystem Benutzerhandbuch” Stand: 2008-05-13
http://rs-demos.ditf-denkendorf.de/tum/manuals/de_DE/Benutzerhandbuch.pdf
 zugegriffen: 2009-01-03
- [Re08b] „Workshop Webmaschineneinstellungen mit dem DITF-MR-Recherchesystem kontinuierlich und adaptiv verbessern“ Stand: 2007-11-27
<http://www.ditf-denkendorf.de/mr/images/stories/downloads/workshop%20recherchesystem,%202007-11-27,%20zusammengefasst.pdf>
 zugegriffen: 2009-01-03
- [Roh06] Jean Rohmer “Wiki and Semantics” in *Proceedings of the First Workshop on Semantic Wikis - From Wiki to Semantics* Juni 2006
<http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-206/paper13.pdf>
 zugegriffen: 2008-11-25
- [SBB07] Sebastian Schaffert, François Bry, Joachim Baumeister „Semantic Wiki“
<http://www.pms.ifl.lmu.de/publikationen/PMS-FB/PMS-FB-2007-19/PMS-FB-2007-19-SemanticWiki-Schlagwort.pdf> zugegriffen: 2009-01-28

- [Sc01] Dagmar Schipanski „Antrittsrede der Ministerin für Wissenschaft, Forschung und Kunst des Freistaates Thüringen, Frau Professor Dr.-Ing. habil. Dagmar Schipanski, anlässlich der Übernahme der Präsidentschaft der Kultusministerkonferenz am 16. Januar 2001 im Sekretariat der Kultusministerkonferenz in Bonn“
<http://www.kmk.org/fileadmin/pdf/PresseUndAktuelles/2002/redeschipanski-2.pdf>
zugegriffen: 2009-01-30
- [Sc08a] Hilmar Schmudt „Vater der Zettelsuchmaschine“ in Der Spiegel Ausgabe 26/2008 S. 140, 23.06.2008
- [Sc09] Sebastian Schaffert, François Bry, Joachim Baumeister, Malte Kiesel „Semantische Wikis“ in Andreas Blumauer, Tassilo Pellegrini *Social Semantic Web Web 2.0 -- Was nun?*, Springer Berlin, 2009
- [Se08a] SMW <http://www.semantic-mediawiki.org> zugegriffen: 2008-11-25
- [Se08b] SMW http://semantic-mediawiki.org/wiki/Help:Properties_and_types
zugegriffen: 2008-11-22
- [Se08c] SMW http://semantic-mediawiki.org/wiki/Help:Custom_units zugegriffen: 2008-11-22
- [Se08d] SMW http://semantic-mediawiki.org/wiki/Help:Many-valued_properties
zugegriffen 2008-11-22
- [Se08e] SMW http://semantic-mediawiki.org/wiki/Help:Semantic_templates
zugegriffen 2008-11-23
- [Se08f] SMW http://semantic-mediawiki.org/wiki/Help:Selecting_pages
zugegriffen: 2008-11-26
- [Se08g] SMW http://semantic-mediawiki.org/wiki/Help:Inline_queries zugegriffen: 2008-11-26
- [Se08h] SMW http://semantic-mediawiki.org/wiki/Help:Displaying_information
zugegriffen: 2008-11-26
- [Se08i] SMW <http://semantic-mediawiki.org/wiki/Help:Inferencing> zugegriffen: 2008-11-26
- [Se08j] SMW <http://semantic-mediawiki.org/wiki/Help:Concepts> zugegriffen: 2008-11-26
- [Se08k] SMW http://semantic-mediawiki.org/wiki/Help:RDF_export zugegriffen: 2008-11-23
- [Se08l] SMW <http://semantic-mediawiki.org/wiki/Help:Reuse> zugegriffen: 2008-11-23
- [Se08m] SMW http://semantic-mediawiki.org/wiki/Help:Import_vocabulary
zugegriffen: 2008-11-23
- [Se08n] SMW http://semantic-mediawiki.org/wiki/Help:Ontology_import
zugegriffen: 2008-11-23
- [Se08o] SMW http://semantic-mediawiki.org/wiki/Help:Service_links zugegriffen: 2008-12-06
- [Se09] SemWiki.org - The Semantic Wiki Community <http://www.semwiki.org>
zugegriffen: 2009-01-30
- [Sw08a] Semantic Web http://semanticweb.org/wiki/Semantic_MediaWiki
zugegriffen: 2008-11-25
- [Sw08b] Semantic Web http://semanticweb.org/wiki/Conceptual_Data_Structures
zugegriffen: 2008-12-07
- [Sw08c] Semantic Web <http://semanticweb.org/wiki/AceWiki> zugegriffen: 2008-12-23
- [Sw08d] Semantic Web http://semanticweb.org/wiki/Hypertext_Knowledge_Workbench
zugegriffen: 2008-11-25

- [Sw08e] Semantic Web http://semanticweb.org/wiki/Hypertext_Knowledge_Workbench/German_Description zugegriffen: 2008-11-25
- [Sw08f] Semantic Web http://semanticweb.org/wiki/Semantic_Wiki_State_Of_The_Art#Wiki_Feature_Matrix zugegriffen: 2008-11-25
- [Sw08g] Semantic Web http://semanticweb.org/wiki/Property:Has_editing_paradigm zugegriffen: 2008-11-23
- [Sw08h] Semantic Web <http://semanticweb.org/wiki/Property:SWikiFeat:SubjectGranularity> zugegriffen: 2008-11-23
- [Sw08i] Semantic Web <http://semanticweb.org/wiki/Property:SWikiFeat:ObjectType> zugegriffen: 2008-11-23
- [Sw08j] Semantic Web <http://semanticweb.org/wiki/Property:SWikiFeat:RepresentationDistinction> zugegriffen: 2008-11-23
- [Sw08k] Semantic Web <http://semanticweb.org/wiki/Property:SWikiFeat:Context> zugegriffen: 2008-11-23
- [Sw09a] Semantic Web http://semanticweb.org/wiki/Semantic_wiki_features zugegriffen: 2009-01-29
- [Sw09b] Semantic Web http://semanticweb.org/wiki/Semantic_Wiki_State_of_The_Art_Paper zugegriffen: 2009-01-29
- [Sw09c] Semantic Web http://semanticweb.org/wiki/Semantic_wiki zugegriffen: 2009-01-29
- [VKV07] Denny Vrandečić, Markus Krötzsch, Max Völkel „Die Zukunft der Wikis: Semantic Web“ In Christoph Lange, *Wikis und Blogs: Planen, Einrichten, Verwalten*, Kapitel 15, S. 789--810. C&L Computer und Literaturverlag, January 2007.
- [Vö06] Max Völkel, Markus Krötzsch, Denny Vrandečić, Heiko Haller, Rudi Studer „Semantic Wikipedia“ in *Proceedings of the 15th international conference on World Wide Web, WWW 2006*, Edinburgh, Scotland, May 23-26, 2006. Mai 2006.
<http://www.aifb.uni-karlsruhe.de/WBS/hha/papers/SemanticWikipedia.pdf> zugegriffen: 2008-10-29
- [Vö08a] Max Völkel, Heiko Haller, William Bolinder, Brian Davis, Henrik Edlund, Kristina Groth, Rosa Gudjonsdottir, Mikhail Kotelnikov, Pär Lannerö, Sinna Lindquist, Mikhail Sogrin, Yngve Sundblad, Bosse Westerlund „Conceptual Data Structures (CDS) Tools“ 15.01.2008
http://mavenrepo.fzi.de/xam.de/2008/D1.2_v10_CDS-Tools.pdf zugegriffen: 2008-12-07
- [Vö08b] Max Völkel „A Semantic Web Content Model and Repository“
<http://mavenrepo.fzi.de/xam.de/2007/2007-05-voelkel-ISEMANTICS-swcmm-CR.pdf> zugegriffen: 2008-12-07
- [Vö08c] Max Völkel „Hypertext Knowledge Workbench“
<http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-360/paper-18.pdf> zugegriffen: 2008-12-07
- [W3C08] W3C Semantic Web <http://www.w3.org/2001/sw> zugegriffen 2008-10-29
- [Wi08a] Wikipedia http://de.wikipedia.org/wiki/Semantischer_Desktop zugegriffen: 2008-12-07
- [Wi08b] Wikipedia <http://de.wikipedia.org/wiki/SPARQL> zugegriffen: 2008-12-13
- [Wi08c] Wikipedia <http://de.wikipedia.org/wiki/Nominalphrase> zugegriffen: 2008-12-14

- [Wi08d] Wikipedia [http://de.wikipedia.org/wiki/Transitivität_\(Grammatik\)](http://de.wikipedia.org/wiki/Transitivität_(Grammatik))
zugegriffen: 2008-12-14
- [Wi08e] Wikipedia [http://de.wikipedia.org/wiki/Intransitivität_\(Grammatik\)](http://de.wikipedia.org/wiki/Intransitivität_(Grammatik))
zugegriffen: 2008-12-14
- [Wi08f] Wikipedia <http://de.wikipedia.org/wiki/Ditransitiv> zugegriffen: 2008-12-14
- [Wi08g] Wikipedia http://de.wikipedia.org/wiki/Web_Ontology_Language
zugegriffen: 2008-12-17
- [Wi08h] Wikipedia http://en.wikipedia.org/wiki/Semantic_Web_Rule_Language
zugegriffen: 2008-12-17
- [Wi08i] Wikipedia [http://de.wikipedia.org/wiki/Notation_3_\(Informatik\)](http://de.wikipedia.org/wiki/Notation_3_(Informatik))
zugegriffen: 2009-01-18
- [Wi08j] Wikipedia http://de.wikipedia.org/wiki/Uniform_Resource_Identifier
zugegriffen: 2008-12-07

- [Wi09a] Wikipedia <http://de.wikipedia.org/wiki/ID3-Tag> zugegriffen 2009-01-23
- [Wi09b] Wikipedia <http://de.wikipedia.org/wiki/IPTC-NAA-Standard> zugegriffen 2009-01-23
- [Wi09c] Wikipedia <http://de.wikipedia.org/wiki/Synonym> zugegriffen 2009-01-26
- [Wi09d] Wikipedia http://de.wikipedia.org/wiki/World_Wide_Web zugegriffen 2009-01-26
- [Wi09e] Wikipedia http://de.wikipedia.org/wiki/Semantisches_Web zugegriffen 2009-01-26
- [Wi09f] Wikipedia http://de.wikipedia.org/wiki/Semantisches_Wiki zugegriffen 2009-01-26
- [Wi09g] Wikipedia http://en.wikipedia.org/wiki/Semantic_wiki zugegriffen 2009-01-26

- [Wn09] Wikitionary <http://wiktioary.org> zugegriffen 2009-01-26

- [Xm08a] xml.com <http://www.xml.com/pub/a/2005/11/16/introducing-sparql-querying-semantic-web-tutorial.html> zugegriffen: 2008-12-13