

Freie Universität Berlin – Institut für Informatik  
Fachbereich Mathematik und Informatik  
AG Netzbasierte Informationssysteme – Prof. Dr.-Ing. Robert Tolksdorf  
Sommersemester 2003

## Seminar 19565 PJ: Projekt Semantic Web

Dokumentation

Roman Schmidt  
Renatenweg 17  
12249 Berlin

Tel. +49-30-773 58 69  
szmidt@zedat.fu-berlin.de  
<http://userpage.fu-berlin.de/~szmidt/index.html>

## Abstract

Das Projekt ergänzte das Seminar "Grundlagen des Semantic Web": Ziel war es, im Seminar behandelte Technologien des Semantic Web anzuwenden. Dazu sollten die Studenten ein Online-Informationssystem zum Institut für Informatik erstellen.

Als Auftraggeber des Projektes fungierte die AG Netzbasierte Informationssysteme des Instituts für Informatik. Auftragnehmer waren die Studierenden des Seminars.

Nach einer Recherchephase zu Eigenschaften verwendbarer Codes und Tools sowie einer Theoriediskussion zu Prinzipien der abfrage-tauglichen Modellierung von semantischen Informationen aus der "Real World" erstellten die Studierenden ein DAML-OIL-Modell des Instituts für Informatik und des Diplomstudienganges Informatik, recherchierten und beseitigten Fehler der Implementierung verwendeter Softwarekomponenten, implementierten dann das Online-Abfragesystem.

---

Infosystem: <http://projects.mi.fu-berlin.de/semweb/>

Namespaces: [http://projects.mi.fu-berlin.de/semweb/ns/ schema](http://projects.mi.fu-berlin.de/semweb/ns/schema), [../ns/uni](http://projects.mi.fu-berlin.de/semweb/ns/uni), [../ns/inf/institut](http://projects.mi.fu-berlin.de/semweb/ns/inf/institut), [../ns/inf/personen](http://projects.mi.fu-berlin.de/semweb/ns/inf/personen), [../ns/inf/2003/SS](http://projects.mi.fu-berlin.de/semweb/ns/inf/2003/SS), [../ns/inf/1993/diplom](http://projects.mi.fu-berlin.de/semweb/ns/inf/1993/diplom), [../ns/display](http://projects.mi.fu-berlin.de/semweb/ns/display)

Seminar: [www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/S\\_SW/default.htm](http://www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/S_SW/default.htm).

Projekt: [www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/P\\_SW/default.htm](http://www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/P_SW/default.htm)

Fuwiki: <http://fu.freezope.org/fuwiki/ProjektSemanticWebSS03>

# 1 Arbeitsziele

Im Sommersemester 2003 fand an der Freien Universität Berlin das Projektseminar "Semantic Web" statt,<sup>1</sup> als Ergänzung zu einem anderen Seminar "Grundlagen des Semantic Web".<sup>2</sup> Das Ziel war es, im Seminar behandelte Technologien des Semantic Web anzuwenden. Dazu sollten die Studenten ein Online-Informationssystem erstellen, dessen Inhalt "semantisch erschlossene Angaben zum Studium der Informatik an der FU Berlin und des Instituts für Informatik" sein sollten, wie die Projektbeschreibung angab:

"Ausgehend von den Studien- und Prüfungsordnungen zum Diplom-, Bachelor- und Master-Studiengang Informatik sollen entsprechende Modelle auf RDF oder DAML+OIL, OWL Basis erstellt werden, die die in den Ordnungen benannten Entitäten wie Studierende, Lehrveranstaltungen etc. in die ebenfalls dort benannten Beziehungen setzen. Diese Modelle sollen geeignet durch entsprechende Instanzen (konkrete Lehrveranstaltungen, Personen, etc.) gefüllt werden. Das so entstandene Semantic Web soll online zugänglich sein und geeignet Navigation und Suchen ermöglichen."<sup>3</sup>

Als Auftraggeber des Projekts fungierte die AG Netzbasierte Informationssysteme, vertreten durch Prof. Dr.-Ing. Robert Tolksdorf. Auftragnehmer waren die Studierenden des Seminars.

Zur Realisierung waren folgende Teilschritte vorgesehen:

- die inhaltliche Erarbeitung der genannten Modelle
- die Repräsentation der Modelle
- die Auswahl und Implementierung von Software zu ihrer Nutzung
- die Realisierung des Semantic Web als Online-Informationssystem.<sup>4</sup>

## Revisionen der Ziele

Die Realisierung dieses anspruchsvollen Ziels gelang im Prinzip, wenngleich es nötig war, die Ziele aus folgenden Gründen zu revidieren:

- Am Seminar nahmen nur vier Studenten teil, was sich als gering erwies.
- Die zur Verfügung stehende Software erwies sich als überwiegend un-ausgereift, absturzanfällig und inkompatibel.

---

<sup>1</sup> Robert TOLKSDORF (Download 25.6.2003), *Projekt Semantic Web (SoSe 2003) 19565 Pj*, Berlin: Freie Universität Berlin, Fachbereich Mathematik und Informatik, Institut für Informatik, AG Netzbasierte Informationssysteme, [www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/P\\_SW/default.htm](http://www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/P_SW/default.htm).

<sup>2</sup> Robert TOLKSDORF (Download 25.6.2003), *Seminar Grundlagen des Semantic Web (SoSe 2003) 19579 P*, Berlin: Freie Universität Berlin, Fachbereich Mathematik und Informatik, Institut für Informatik, AG Netzbasierte Informationssysteme, [www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/S\\_SW/default.htm](http://www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/S_SW/default.htm).

<sup>3</sup> TOLKSDORF (Download 25.6.2003), *Projekt Semantic Web*.

<sup>4</sup> ebd.

- Daher dauerte nicht nur die Recherche geeigneter Tools länger als geplant, sondern es wurde auch eine Entscheidung für ein bestimmtes System notwendig, was den Beginn der Implementierung verzögerte.
- Die Notwendigkeit und die Schwierigkeiten, theoriegeleitete Regeln zur Ontologiebildung eindeutig zu formulieren und bei der Modellbildung anzuwenden, waren größer als erwartet.
- Die Komplexität der zu modellierenden Studienordnungen ist wesentlich höher als die der meisten bereits verfügbaren Ontologien.
- Abfragetools stehen bisher nur rudimentär zur Verfügung

Die Revision der Projektziele betraf in erster Linie den vorläufigen Verzicht auf die Modellierung des Bachelor- und Master-Studiengangs Informatik.<sup>5</sup> Das Informationssystem wird bis auf weiteres nur den Diplom-Studiengang behandeln.

## 2 Durchführung

### 2.1 Modellbildung (A1)

#### 2.1.1 Modellinhalt

Den Modellinhalt spezifizierte die Seminargruppe wie folgt:

Der Inhalt der Studien- und Prüfungsordnungen, das Kommentierte Vorlesungsverzeichnis, die Arbeitsgruppen, die Mitarbeiter und die thematische Einordnung der Lehrveranstaltungen sollten in Modell eingehen. Alle diesbezüglichen Informationen, die nicht aus der Website und den Studien- und Prüfungsordnungen hervorgingen, schloss sie hingegen von der Modellierung aus.

Zur Anbindung des Modell an bestehende Ontologien einigte sich die Gruppe auf folgende Ontologien, die zu berücksichtigen waren:

- Dublin Core Metadata Initiative als allgemeine RDF-Standard-Ontologie
- ACM-Klassifikation zur thematischen Einordnung der Lehrveranstaltungen
- für Kalender: vcalendar der RDF Calendar taskforce
- für Personen: FOAF: the "friend of a friend" vocabulary

---

<sup>5</sup> Einstimmig beschlossen auf der Projektsitzung vom 27.6.2003 Roman SCHMIDT (8. Juni 2003), "Protokoll der Projektsitzungen vom 22. und 27. Mai 2003", *FuWiki*, Berlin: FU, <http://fu.freezope.org/fuwiki/ProjektSemanticWebProtokoll4>

### 2.1.2 Modelloutput

Vor der eigentlichen Modellierung (des Inputs ins Modell) erachtete die Gruppe es für sinnvoll, die Art des zu erzeugenden Modell-Outputs zu spezifizieren. Das letztlich zu produzierende Online-Abfragesystem sollte folgendes leisten:

- Eine dokumentierte Ontologie für fremde Nutzer
- Eine Vorauswahl von Items zur Suche (etwa in einer HTML-Schnittstelle)
- Ermöglichung von Browsen durch die Instanzen; dazu Rendering des semantischen Web nach HTML
- Generierung von Links aus der Liste der Beziehungen

Das Rendern nach HTML oder andere Darstellungsformen sollten universell anwendbar sein, und nicht speziell auf die Bedürfnisse des Instituts angepasst werden. Daher waren Informationen über die Darstellung ins Modell zu integrieren. Für den Nutzer uninteressante Informationen sollten dabei unsichtbar sein. Dazu war es notwendig, die Meta-Information, ob eine Information anzuzeigen sei, im Modell zu erfassen (als Property "Nicht anzuzeigen"). Nach Möglichkeit sollten die Abfragen keine inhaltlichen Informationen mehr enthalten.

### 2.1.3 Modellierungsrichtlinien und Repräsentation

Schon zu Beginn der Modellierung zeigte sich, dass zur gemeinsamen Arbeit an einem Modell genauere Modellierungsrichtlinien erforderlich sein würden. Die aufkommenden Fragen waren einerseits erkenntnistheoretischer, andererseits praktischer Natur, und betrafen sowohl die Problematik, Abfragen zu programmieren, als auch die Semantik von Ontologien. Insbesondere die Möglichkeit, in DAML/OIL zwischen Klassen und Individuen zu unterscheiden, eröffnete verschiedene Interpretations- und Zuordnungsmöglichkeiten, die eine Inkonsistenz des Modells verursacht hätten.

Die aufkommenden Fragen konnte die Gruppe mit Hilfe eines Probemodells klären. Dieses beschrieb einen exemplarischen "Mini-Studiengang" mit zwei Vorlesungen, einer Prüfung und fünf Studenten.<sup>6</sup>

Die Abfragen der Probemodellierungen des Mini-Studiengangs ergaben, dass keines der beiden Modelle besser für Abfragen geeignet ist. Die Anforderung, nicht nur Restrictions, sondern auch simple Aussagen über Subjekte zu treffen, erfordert die Verwendung von Individuals. Die Aussagenlogik erfordert hingegen die Verwendung von Klassen.

Um nachprüfbar und reliabel modellieren zu können und Unklarheiten bei der Zuordnung von Klassen, Individuen und Eigenschaften auszu-

---

<sup>6</sup> Richard CYGANIAK (5. Juni 2003), " SWMiniModellAnforderungen ", *FuWiki*, Berlin: FU, <http://fu.freezope.org/fuwiki/SWMiniModellAnforderungen>

räumen, definierte die Gruppe in mehreren Sitzungen Modellierungsregeln. Demnach sind zu starke Hierarchisierungen im Prinzip zu vermeiden. Das Festhalten an flachen Hierarchien ist jedoch nicht dadurch zu erreichen, dass man daml-Individuals anstelle von daml-Klassen verwendet und diesen, wenn tiefere Klassifizierungspfade notwendig sind, Properties wie "gehört zu Klasse" bzw. "hat Typ" oder "ist ein" zuzuordnen, denn dies würde eine scholastische Entscheidung für das Arbeiten mit dem Modell bedeuten, was nach der "richtigen Lehre" wie zu klassifizieren wäre. Die Frage, ob ein zu modellierendes Subjekt eine Klasse oder ein Individuum ist, wird nicht logisch, sondern pragmatisch entschieden, nach folgenden Regeln:

- Typbestimmung erfolgt durch Subklassifizierung. Klassifikationen und Typisierungen erfolgen prinzipiell durch Klassen und Klassenhierarchien von Super- und Subklassen, es sei denn, die unterste Subklassenebene soll nicht weiter modelliert werden.
- Sämtliche Klassifikationshierarchien sind unten durch Individuals abgeschlossen - auch dann, wenn das modellierte Subjekt logisch gar kein echtes "Real-World"-Individuum ist.
- Wenn keine "Real-World"-Individuals modelliert werden, sondern nur deren Real-World-Klasse, so geschieht dies durch DAML-OIL-Individuals.
- DAML-OIL-Individuals, die für Real-World-Klassen stehen, betrachtet die Seminargruppe als "Alias" und kennzeichnen sie durch den unbestimmten Artikel.<sup>7</sup> Dies führt nebenbei dazu, dass die Aussagen besser lesbar werden.
- Diese Vorgehensweise ist nur zulässig, wenn keine echten "Real-World"-Individuals modelliert werden. Wenn es Ressourcen gibt, die für echte Individuals stehen, dann darf kein Individual-Alias für diese stehen, weil sonst unerwünschte Modellierungen notwendig werden.<sup>8</sup>
- Die Seminargruppe arbeitet verstärkt mit der Funktion "Same As", die es ermöglicht, Klassen aus Property-Beziehungen zu generieren. Für Abfragen sollte daher zu jeder relevanten Property eine Klasse vorliegen, die mit "Same As" kodiert ist und abfragbar wird.

---

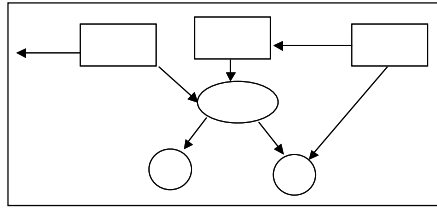
<sup>7</sup> Ein Beispiel:           <DAML-Individual-Subjekt="Ein Student" >  
                                   <Property-Prädikat="hatErworben">  
                                   <DAML-Individual-Objekt="Einen Schein">

<sup>8</sup> Ein Negativbeispiel: <DAML-Individual-Subjekt="Max Müller" >  
                                   <Property-Prädikat="ist">  
                                   <DAML-Individual-Objekt="Ein Student">

Das wäre Unsinn, weil man "ist" bzw. "hat Typ" in DAML mit Klassifizierungen ausdrückt.

#### 2.1.4 Inhaltliche Erarbeitung des Modells

Bei der Erarbeitung schälten sich im Seminar drei Ansätze heraus, die sich in den verschiedenen Phasen der Modellbildung abzeichnen.



Der erste Ansatz geht von der Makro- zur Mikroperspektive. Er identifiziert zuerst zentrale Bestandteile des Modells und stellt deren Beziehungen graphisch und global dar.

Danach erfolgt die Modellierung in DAML/OIL. Seine Vorteile sind, dass von vorneherein ein spontaner Überblick über Wesentliches besteht. Nebensächliches stört nicht bei der Modellierung. Auf abstrakte Modellierungsregeln kann man evtl. verzichten, und das Modell lässt sich an die gewünschte Abfrage anpassen.

Der zweite Ansatz geht von der Mikro- zur Makroperspektive. Er beginnt mit einer Inhaltsanalyse der zu modellierenden Ressourcen. Singulärer Statements der

Ressource werden satzweise nach Modellierungsregeln in DAML/OIL überführt. Dabei entsteht Durch wiederholte Verwendung derselben Properties und Ressourcen ein Netz, dessen Form anfangs noch unbekannt ist. Er ermöglicht eine detailreiche Modellierung nah an der „Real-World“. Klassifikationsentscheidungen sind weniger folgenreich, und die Terminologie bleibt mit den Ressourcen konsistent.

„§ 2 Das Studium gliedert sich in Grundstudium und Hauptstudium.“

→ *Subject* : Studium,

→ *Property* : consistsOf,

→ *Bag* : {Grundstudium, Hauptstudium}

Der dritte Ansatz übernimmt maschinenlesbare Daten mittels eines Parsers in das Modell. Die Struktur der erzeugten Klassen und Instanzen hängen dabei erst in zweiter Linie von der Real-World oder von den Ideen eines menschlichen Modellierers ab. Entscheidend ist zuerst die Struktur der geparsten Daten. Diese sind selbst ein Modell der Welt, das aber bisher in der Regel nicht mit dem Ziel erstellt wurde, ein semantisches Netz zu bilden.

Die Seminargruppe verwendete bei der Modellierung alle drei Vorgehensweisen in der aufgeführten Reihenfolge.

#### 2.1.5 Ergebnis: Meilenstein W7

Bis zum Meilenstein W7 lag ein DAML-OIL-Modell vor, das die in der Studienordnung für den Diplomstudiengang Informatik beschriebenen Entitäten und ihre Beziehungen enthielt. Abfragbare Instanzen sind bereits enthalten und werden bis zum Meilenstein W13 noch eingearbeitet.

Weiterhin lag ein abfragbares Modell des Mini-Studiengangs vor.

Die noch ausstehende Modellierung des Bachelor-, Master- und Lehramtsstudiengangs Informatik bietet Chancen und Aufgabenstellungen für künftige Modellierungsprojekte.

## 2.2 A2: Infrastruktur

### 2.2.1 Standard / Code: DAML-OIL

Auf welchen Standard das semantische Modell aufbauen sollte, war nicht vorgegeben. Vor dem Beginn der praktischen Umsetzung jeglicher Modellierung und auch vor der Auswahl von Editoren und anderen Tools war daher zu entscheiden, welcher der vorhandenen Standards für das Projekt geeignet ist.

Zur Auswahl standen RDF, RDF-Schema, DAML/OIL, OWL und RuleML. Topic Maps waren wegen des gleichzeitig stattfindenden Seminars der Technischen Universität Berlin zu diesem Thema ausgeschlossen. Die Entscheidung fiel auf DAML/OIL, weil

- das Seminar die Möglichkeiten von RDF für zu beschränkt erachtete,
- Tools für DAML/OIL zur Verfügung standen ([www.daml.org/tools](http://www.daml.org/tools))
- DAML/OIL auf RDF basiert,
- der Lernaufwand für Rule M L für zu groß erachtet wurde,
- Rule M L noch zu wenig etabliert war, und
- die Regeln, deren Implementierung Rule M L bietet, nicht benötigt wurden.<sup>9</sup>

Die Entscheidung für DAML/OIL war wesentlich vom Ergebnis der Toolrecherche beeinflusst.

### 2.2.2 Auswahl eines Editors

Es gab überhaupt nur eine begrenzte Anzahl geeigneter Editoren, von denen eine Vielzahl absturzanfällig war. Als hinreichend installationsfreundlich erwiesen sich die RDF-Tools *Fenfire Loom*, *RDF-Editor* (*rdfedt*), *Rdf-VRP 2.5* und *Isaviz*, sowie die DAML-Editoren *OiEd* und *Smore* sowie der OWL-Editor *OntoEdit*. All diese waren untereinander weitgehend inkompatibel.<sup>10</sup>

Vom einem Editor erzeugte Codes konnten andere Programme nicht oder nur teilweise laden, nicht sinnvoll visualisieren und vor allem nicht austauschen. Einzig der RDF-Editor *rdfedt* konnte die meisten Dateien problemlos öffnen und als RDF darstellen, aber oft in einer nicht sinnvollen Darstellung und einer reinen Klassenhierarchie. Daher war die Festlegung auf nur ein Tool zum Editieren, Visualisieren und Ausgeben unumgänglich.

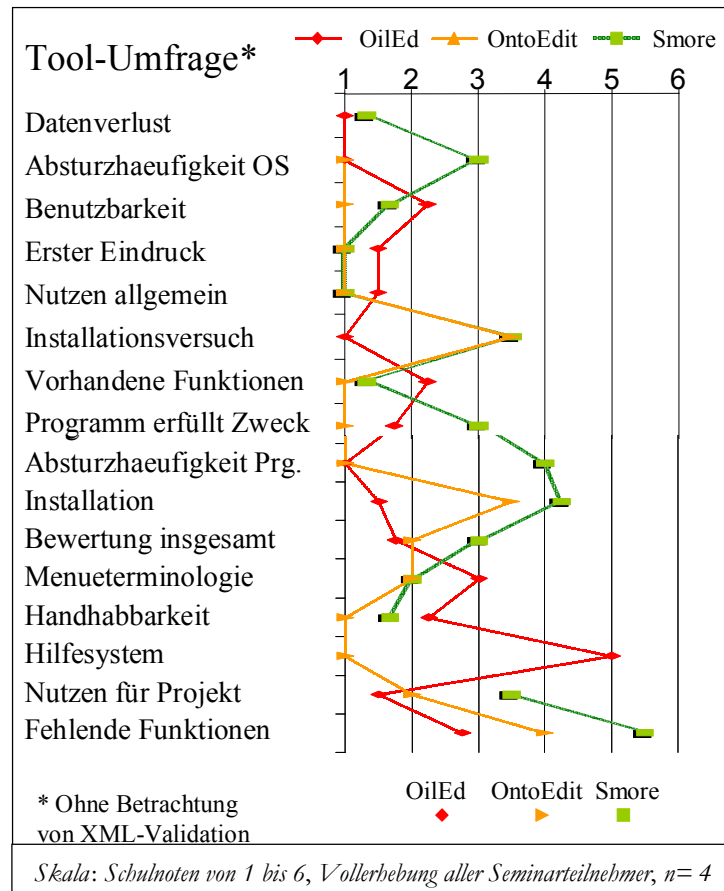
---

<sup>9</sup> Roman SCHMIDT (9.5.2003), "Protokoll der Projektsitzung vom 8. Mai 2003", *FuWiki* Berlin: FU, <http://fu.freezope.org/fuwiki/ProjektSemanticWebProtokoll2>

<sup>10</sup> Dennis HECKERT, Roman SCHMIDT et al. (20.5.2003), "Erfahrungsberichte mit Tools aus dem Seminar", *FuWiki*, Berlin: FU, <http://fu.freezope.org/fuwiki/ErfahrungsberichteDamlOilTools>



Den geeigneten Editor ermittelte eine Umfrage unter den Seminarteilnehmern, die folgende Ergebnisse brachte:<sup>11</sup>



Bereits während der Toolevaluation fiel – auch weil mit *OilEd* ein geeigneter Editor vorlag – die Entscheidung zugunsten von DAML/Oil, sodass nicht alle Seminarteilnehmer die Evaluation für die RDF-Tools *Fenfire*, *Loom*, *RDF-Editor (rdfedt)*, *Rdf-VRP 2.5* und *Isaviz* beendeten. Daher sind die Ergebnisse für diese Programme hier ausgespart.

*OilEd* erwies sich als der geeignetste Editor für die Ziele des Projekts.

### 2.2.3 Reparatur des *OilEd*-Outputs

Beim Prüfen der von *OilEd* generierten DAML-Files mit dem DAML-  
 • *alidator*<sup>12</sup> entdeckte Richard Cyganiak Fehler im Output.<sup>13</sup> Es handelte sich um Code-Schreibweisen, die nicht der DAML-Norm entsprachen. Das Script zur Fehlerbeseitigung programmierte Dennis Heckert.<sup>14</sup>

<sup>11</sup> Dennis HECKERT, Roman SCHMIDT et al. (20.5.2003), "Erfahrungsberichte mit Tools aus dem Seminar", *FuWiki*, Berlin: FU, <http://fu.freezope.org/fuwiki/ErfahrungsberichteDamlOilTools>

<sup>12</sup> <http://www.daml.org/validator/>

<sup>13</sup> Richard CYGANIAK (7.6.2003 19:35:38), [*Nbi\_p\_sw*] *Zwischenstand*, Mail an [nbi\\_p\\_sw@lists.spline.inf.fu-berlin.de](mailto:nbi_p_sw@lists.spline.inf.fu-berlin.de).

<sup>14</sup> Dennis HECKERT (Download 8.7.2003) "Index of /~heckert/PSemWeb/SED", *Userpages/~heckert*, Berlin: Freie Universität,

#### 2.2.4 Implementierung der Hardware: Webserver

Die Arbeitsgruppe Rechnerbetrieb des Instituts für Informatik richtete einen Webserver für das Seminar ein, sowie eine Unix-Usergruppe "Semweb". Zugriff auf den Projektserver erfolgt über:

```
ssh troll.inf.fu-berlin.de mit Informatik-Account  
cd /import/projects/semweb
```

Der Server ist per HTTP zu erreichen

Weiterhin wurde ein CVS-Repository eingerichtet.

```
Host: lin211.inf.fu-berlin.de  
Pfad: /home/bude/cyganiak/cvsroot  
Methode: extssh  
User und Passwort: Informatik-Account  
Modul: semweb15
```

Die Aktivitäten während der Servlet-Programmierung überwachte eine Logfunktion, die für jeden Commit automatisch eine Mail an die Seminar-Mailliste schickte.<sup>16</sup>

Die Arbeitsgruppe Rechnerbetrieb richtete der Seminargruppe nur eingeschränkte Zugriffsrechte auf den Webserver ein, sodass ein Start des Reasoners auf dem Webserver bis in die letzte Seminarwoche nicht möglich war.

#### 2.2.5 Implementierung der Abfrage-Software

Als API verwendete die Seminargruppe die bekannte, in vielen anderen Programmen ebenfalls verwendete Jena-API von Hewlett-Packard-Entwicklern. Diese sollte vollständiges DAML+OIL-Reasoning ermöglichen.

Als Reasoner kam, wie Richard Cyganiak berichtete,<sup>17</sup> zunächst der FaCT-Reasoner zur Anwendung. Dieser setzt auf CORBA auf und wies sich als zu langsam, fehleranfällig und umständlich. Stattdessen verwendete er den RACER-Reasoner, den man als Standalone-HTTP-Server betreiben kann.<sup>18</sup>

Die Ausgabe des Reasoners hat das DIG-XML-Format,<sup>19</sup> was zur Integration ins Jena-Repository ein Übersetzen in RDF oder Jena-Aufrufe

---

<http://page.inf.fu-berlin.de/~heckert/PSemWeb/SED/>.

<sup>15</sup> Richard Cyganiak (2.7. 2003 18:52:51) [*Nbi\_p\_sw*] *CVS / Projektserver*, Mail an [nbi\\_p\\_sw@lists.spline.inf.fu-berlin.de](mailto:nbi_p_sw@lists.spline.inf.fu-berlin.de)

<sup>16</sup> (Seminarer Teilnehmer) (Juni-Juli 2003), [*cvs-semweb*] ..., [nbi\\_p\\_sw@lists.spline.inf.fu-berlin.de](mailto:nbi_p_sw@lists.spline.inf.fu-berlin.de).

<sup>17</sup> Richard CYGANIAK (1.7.2003 04:25:41), [*Nbi\_p\_sw*] *Kurzer Zwischenstand von der Reasoner-Front*, Mail an [nbi\\_p\\_sw@lists.spline.inf.fu-berlin.de](mailto:nbi_p_sw@lists.spline.inf.fu-berlin.de).

<sup>18</sup> <http://www.cs.concordia.ca/~faculty/haarslev/racer/download.html>

<sup>19</sup> <http://dl-web.man.ac.uk/dig/2002/10/interface.pdf>

erforderte. Das DAML+OIL-Reasoning erlaubte auch die korrekte Auswertung von DAML-Features wie Transitivität oder inversen Properties.

### 2.2.6 Online-Informationssystem

Die Implementierung des Online-Informationssystems begann mit Verzögerung. Gründe waren hierfür in erster Linie, dass die technischen und theoretischen Schwierigkeiten unterschätzt worden waren:

- Langwierige Definitionsfindungen während der Modellbildung
- geringe, inkompatible und fehleranfällige Tools
- inkompatible Standards
- wenig mächtige Abfragesprachen

Die nur langsam voranschreitende Modellbildung verzögerte die Implementierung der Abfragesoftware ebenso wie der spät und mit eingeschränkten Zugriffsrechten zur Verfügung stehende Webserver.

Die Verspätungen konnten eingeholt werden. Bei Seminarende stand ein funktionsfähiges Online-Informationssystem zur Verfügung, das unter <http://projects.mi.fu-berlin.de/semweb/> nutzbar ist.<sup>20</sup> Dort steht eine DAML+OIL-Version des Modells mit allen Instanzen,<sup>21</sup> die HTML-Version, die dieses Modell dokumentiert,<sup>22</sup> sowie die jeweiligen Namespaces mit DAML+OIL-Datei und HTML-Dokumentationen.

## 3 Ergebnisse

(Stand: 18.7.2003)

- Online-Informationssystem<sup>23</sup>
- Online-Informationssystem, in Echtzeit nach HTML gerendert
- Modellierung eines exemplarischen "Mini.-Studiengangs"
- Definition von Modellierungsregeln
- Gesamtes DAML-OIL-Modell der Studienordnung des Diplomstudiengangs Informatik und des Instituts für Informatik mit abfragbaren Instanzen
- HTML-Version des Modells
- Evaluation von RDF- und DAML/OIL-Editoren

---

<sup>20</sup> PROJEKTSEMINAR SEMANTIC WEB (Richard CYGANIAK, Dennis HECKERT, Anja JENTZSCH, Roman SCHMIDT) (SS2003/18.7.2003), *Semantische Modellierung des Instituts*, Berlin: Freie Universität Berlin, <http://projects.mi.fu-berlin.de/semweb/>

<sup>21</sup> Richard CYGANIAK (=PROJEKTSEMINAR SEMANTIC WEB, d.i. Richard CYGANIAK, Dennis HECKERT, Anja JENTZSCH, Roman SCHMIDT) (SS2003/18.7.2003), *Das gesamte Modell des Instituts für Informatik, FU Berlin*, Berlin: Freie Universität Berlin, Version: 1.6, <http://projects.mi.fu-berlin.de/semweb/model/all.daml>

<sup>22</sup> ebd., <http://projects.mi.fu-berlin.de/semweb/model/documentation.html>

<sup>23</sup> CYGANIAK (=PROJEKTSEMINAR SEMANTIC WEB) (SS2003/18.7.2003), <http://projects.mi.fu-berlin.de/semweb/>

## 3.1 Meilensteine

### *Erreichte Meilensteine*

W3: Projektorganisation und Arbeitsverteilung  
(8.5.2003, 3. Sitzung)

W5: Komponenten der Infrastruktur  
Entscheidung für und Nutzung von DAML/OIL mit Editor OilEd  
(22.5. 2003, 5. Sitzung)

W7: Modell  
Diplom-Studiengangs Informatik (5.6.2003, 6. Sitzung)

W10: "Meilenstein Modellierung 1" (26.6.2003, 9.Sitzung)

W10-a: "Meilenstein Modellierung 2" (30.6.2003)

W10-b: "Meilenstein Dokumentation" (30.6.2003)

### *Verspätete, bis zur Präsentationsitzung eingeholte Meilensteine*

W9-a "Meilenstein Webserver" (25.6.2003)

W9: Navigationsdienste

W12: "Meilenstein Abnahme" (10.7.2003, 11. Sitzung)

W13: Projektabschluss (17.7.2003, 12.Sitzung)

W13: Überarbeitetes Modell mit Instanzen (17.7.2003, 12.Sitzung)

W13: Online-Version des Informationssystems (17.7.2003, 12.Sitzung)

### *Vorläufig angegebene Ziele:*

Modell des Master-, Bachelor- Magister- und Lehramtsstudiengangs Informatik

## 3.2 Das Online-Modell und seine Namespaces

### 3.2.1 Gesamtmodell

- Für OilEd optimierte Version des gesamten Modells: DAML+OIL:  
<http://projects.mi.fu-berlin.de/semweb/model/all.daml>
- HTML-Dokumentation des gesamten Modells:  
<http://projects.mi.fu-berlin.de/semweb/model/schema.html>

### 3.2.2 Teile des Modells

#### *Allgemeines Schema*

Grundlegende Konzepte wie Person, Organisation, Veranstaltung, Adresse, Wochentag.

- Namespace und DAML+OIL:  
<http://projects.mi.fu-berlin.de/semweb/ns/schema#>
- HTML-Dokumentation:  
<http://projects.mi.fu-berlin.de/semweb/model/schema.html>

### *Universitäts-Vokabular*

Dieser Teil des Modells definiert Konzepte zur Beschreibung einer Universität, wie zum Beispiel "Fachbereich" und "Institut" als Unterklassen von "Organisation", "Vorlesung" und "Seminar" als Unterklassen von "Veranstaltung", "Dozent" und "Tutor" als Unterklassen von Person.

- Namespace und DAML+OIL:  
<http://projects.mi.fu-berlin.de/semweb/ns/uni#>
- HTML-Dokumentation:  
<http://projects.mi.fu-berlin.de/semweb/model/schema.html>

### *Beschreibung des Instituts für Informatik*

Instanzen zur Beschreibung des Instituts, der Mitarbeiter, der Arbeitsgruppen und Räume.

- Namespace und RDF:  
<http://projects.mi.fu-berlin.de/semweb/ns/inf/institut#>
- Namespace und RDF:  
<http://projects.mi.fu-berlin.de/semweb/ns/inf/personen#>

### *Beschreibung der Veranstaltungen im Sommersemester 2003*

Instanzen zur Beschreibung der Veranstaltungen am Institut für Informatik im Sommersemester 2003

- Namespace und RDF:  
<http://projects.mi.fu-berlin.de/semweb/ns/inf/2003/SS#>

### *Modellierung der Studien- und Prüfungsordnung Informatik*

und Vokabular für Rendering-Tool

### *Anzeige-Informationen für klassenspezifisches HTML-Rendering*

- Namespace und DAML+OIL:  
<http://projects.mi.fu-berlin.de/semweb/ns/display#>

## 3.3 Das Modell in der Entwicklung

### 3.3.1 Modell Eins

Der erste Modellansatz vom 6.5.2003 entstand nach gründlicher Durchsicht der Studienordnungen und des KVV auf Klassenebene. Das Modell nutzte nur die Möglichkeiten, die RDFS bietet, d.h. es waren z.B. noch keine Kardinalitäten einbezogen.

Diejenigen Klassen, die ab Modell 5 wieder entfernt wurden, sind unten separat aufgeführt.

### 3.3.2 Modell Zwei

Unser zweites Modell vom 2003/05/14 entstand aufbauend auf Modell 1 unter Berücksichtigung der folgenden Erkenntnisse und Kritikpunkte:

- Als Ontologiesprache wird nun auf jeden Fall DAML+OIL verwendet.
- Durch Rollen spezialisierte Personen sollten auf jeden Fall auch so modelliert werden. Eine Person sollte nicht aufhören zu existieren, nur weil sie nicht mehr Student oder Dozent ist. Auch wenn diese Personen dann vielleicht aufhören, in unserem Datenbestand zu existieren, sollte dennoch diese wirklichkeitsnähere Modellierung gewählt werden.

### 3.3.3 Modell Acht

Das Modell entstand auf Basis der ersten DAML-Implementierung von Modell 2 als inhaltanalytisches Pendant zur Studienordnung des Diplomstudiengangs Informatik. Es enthält zahlreiche "Real-World"-Modellierungen, bei deren Kodierung die Semantik der Studienordnung den Vorrang vor konkreten, später zu realisierenden Abfragemethoden hatte.

### 3.3.4 Modell 14

Das Modell enthält zusätzlich zur Studienordnung zahlreiche Daten aus dem Webangebot des Instituts für Informatik, darunter sämtliche Arbeitsgruppen, Sekretariate, Mitarbeiter, eine Reihe Räume und Telefone.

### 3.3.5 Modell 24

Das Modell enthält die wesentlichen Aspekte der Prüfungsordnung, also die Abhängigkeiten der Vordiplomprüfung und Diplomprüfung wie Zulassungsvoraussetzungen etc.. Weiterhin wurden in diesem Modell die ACM-Topics aus Kompatibilitätsgründen gemäß der originalen ACM-Topic-Datei überarbeitet. Diese stimmt mit den Kodierungsregeln des Seminars insofern nicht überein, als dass sämtliche Topics Individuen sind und dass die Klassenhierarchie durch den Namen der Instanzen definiert ist, anstelle durch die DAML-OIL-Klassen. Auch übergeordnete Topics, die als Oberklasse von Subtopics agieren, sind als Individuals kodiert. Folgerichtig finden die *properties* "Subtopic" und "Supertopic", die logische *subproperties* der in unserem Modell ausgeschlossenen *property* "hat Typ" sind, wieder Verwendung.

### 3.3.6 Modell 25 ff.

Das Modell enthält die aus der KVV-Datenbank geparsten Instanzen der Veranstaltungen und Dozenten des Instituts für Informatik.

### 3.3.7 Modell, getrennt nach Namespaces

Das Modell wurde für den Webserver in mehrere DAML-Dateien getrennt, nach folgenden Namespace-Präfixen und Namespaces:

dipl = [http://projects.mi.fu-berlin.de/semweb/ns/dipl\\_inf\\_1993#](http://projects.mi.fu-berlin.de/semweb/ns/dipl_inf_1993#)  
Diplom-Studienordnung und Prüfungsordnung

display = [http://projects.mi.fu-berlin.de/semweb/ns/display#  
Templates](http://projects.mi.fu-berlin.de/semweb/ns/display#Templates)  
s = [http://projects.mi.fu-berlin.de/semweb/ns/schema#  
allgemeingültige Klassen, unabhängig vom Fachbereich \(Bsp.:  
Raum oder Person\)](http://projects.mi.fu-berlin.de/semweb/ns/schema#allgemeingültige_Klassen,_unabhängig_vom_Fachbereich_(Bsp.:_Raum_oder_Person))  
inst = [http://projects.mi.fu-berlin.de/semweb/ns/institut#  
konkrete Instanzen](http://projects.mi.fu-berlin.de/semweb/ns/institut#konkrete_Instance)

### 3.4 Klassen des Modells

Die Dokumentation der Klassen erfolgte ab Modell 5 in DAML, im "Documentation"- Feld von OilEd. Das vollständige aktuelle Modell ist im Anhang, S. **Fehler! Textmarke nicht definiert.** ff, enthalten.

Die im Folgenden angegebenen Klassen stammen aus der ersten Modellierung (Modell Eins).

Soweit die Klassen des ersten Modells Verwendung fanden, sind sie hier angegeben. Bei Änderungen im Konzept sind die Klassen des ersten Modells als *entfernt* markiert; auch diejenigen, die noch unter gleichem Namen, aber mit anderer Semantik im Modell vorkommen.

#### 3.4.1 Klassen aus Modell 1 im Modell 15

##### **Abschluss**

Der Abschluss, mit dem ein Studium beendet wird. Superklasse aller Abschlüsse von Bildungsgängen, Studien, etc. Bsp.: *Vordiplom* für das Studium *Diplom-Grundstudium*

##### **Arbeitsgruppen\_Inst\_Inf\_FUB**

Gemäß unserer Absprache wird diese Organisationseinheit des Instituts als Konzept mit ins Modell aufgenommen.

##### **Dozent**

Spezielle Person, die eine konkrete *VERANSTALTUNG* durchführt (aber keine Übungen oder Praktika! Das machen "Akademische Mitarbeiter" und "Tutoren".

STO Dipl\_Inf §8-2:

Die Übungen, Praktika bzw. Projekte, die pro Gruppe nicht mehr als 15 Teilnehmer umfassen sollten, werden unter der verantwortlichen Leitung des Dozenten, der die Vorlesung hält, von akademischen Mitarbeitern durchgeführt.

##### **Leistungsnachweis**

Klasse aller Leistungsnachweise. Konzept eines konkreten Leistungsnachweises, d.h. des Nachweises eines erfolgreichen Besuches einer konkreten Veranstaltung durch einen konkreten Studenten. #Ggf. als *EIN\_LEISTUNGSNACHWEIS* oder *EIN\_ALP2\_LEISTUNGSNACHWEIS* modellieren.

##### **LV\_\_nach\_Bereich\_Grundstudium**

Jede *LV\_\_NACH\_FACHGEBIET\_HAUPTSTUDIUM* und jede *LV\_\_NACH\_BEREICH\_GRUNDSTUDIUM* sollte für ein oder mehrere Studienbereiche als Pflicht- oder Wahlveranstaltung zugeordnet sein.  
Bsp.: Bereich *Algorithmen und Programmierung* ist für das Diplom-Grundstudium eine *LV\_\_NACH\_BEREICH\_GRUNDSTUDIUM*.

### **LV\_\_nach\_Fachgebiet\_Hauptstudium**

Jede *LV\_\_NACH\_FACHGEBIET\_HAUPTSTUDIUM* sollte für ein oder mehrere Studienbereiche als Pflicht- oder Wahlveranstaltung zugeordnet sein..

### **Ort**

Allgemeines Konzept eines Ortes. Ist z.B. für die genaue Bestimmung eines Termins von Nöten; wird durch die Komponenten einer Adresse definiert.

### **Person**

Konzept einer natürlichen Person.

### **Raum**

Subklasse von *ORT*. Stellt einen Raum am Institut dar. Die Unterscheidung zwischen dem allgemeinen Konzept *Ort* und dem spezielleren Konzept *Raum* ist sinnvoll, da auch Veranstaltung außerhalb des Instituts modelliert werden können sollen, z.B. Exkursionen oder gewisse Seminare.

### **Student**

Spezielle Person, die im Rahmen eines *STUDIUM* konkrete *VERANSTALTUNGEN* besucht bzw. zudem Leistungsnachweise sammelt.

Ob die Matrikelnummer eines Studenten ins Modell einbezogen werden soll, bleibt zu klären. Im Moment scheint das nicht sinnvoll zu sein.

### **Studienbereich**

Gliederungskonzept für ein Studium.

Jedes Studium gliedert sich in verschiedene thematische Bereiche mit eigenen Anforderungen und Veranstaltungen, die diesen Bereichen anrechenbar sind. Die Studienordnung gliedert diese in

*LV\_\_NACH\_BEREICH\_GRUNDSTUDIUM* und

*LV\_\_NACH\_FACHGEBIET\_HAUPTSTUDIUM*

### **Studientyp**

Der Typ eines Studiums, d.i. *DIPLOM*, *MASTER*, *BACHELOR*, *LEHRAMT*, *MAGISTER*.

### **Termin**

Eine *VERANSTALTUNG* findet zu ein oder mehreren Terminen statt.

Zu einem Termin gehört die Angabe eines Ortes und einer Zeitspanne.

### **Topic**

Ein Topic ist ein wissenschaftlicher Teilbereich der Informatik, klassifiziert nach der ACM-Klassifikation.

### **Veranstaltung**



Die Superklasse aller Veranstaltungen, einschließlich Lehrveranstaltungen und anderen. Eine *VERANSTALTUNG* findet zu einem oder mehreren Terminen statt.

**Veranstaltungsnummer:**

Eine Veranstaltungsnummer ist durch das KVV einer *LEHRVERANSTALTUNG\_INSTINF* zugeordnet, und zwar nach Möglichkeit eineindeutig jeweils einer *LV\_\_NACH\_BEREICH\_GRUNDSTUDIUM* oder *LV\_\_NACH\_FACHGEBIET\_HAUPTSTUDIUM*. Ob diese Nummern sich als eineindeutige Schlüssel eignen, ist mehr als fraglich.

**Veranstaltungstyp**

Der Typ einer *LEHRVERANSTALTUNG\_INSTINF*

Bsp.: *Vorlesung* oder *Seminar*.

**Zeit**

Stellt eineregelmäßige zeitliche Spanne dar, z.B. *Montags, 10:15h - 11:45h, Regelstudienzeit, Semester, Semesterwochenstunden, Termin*.

---

*\_\_***Noch nicht entfernt: Thema**

Ein Thema ist ein wissenschaftlicher Teilbereich der Informatik. Die Klasse sollte mit der ACM-Klassifikation zusammengeführt und dann überflüssig werden.

---

3.4.2 Entfernte Klassen

*\_\_***Noch nicht implementiert: Note**

Konzept einer Benotung.

In unserem Modell können Leistungsnachweise und Abschlüsse benotet werden. Was genau unter einer Note subsummiert wird ist noch zu klären, wahrscheinlich sind aber ein Literal (int) für die Note als Zahl und ein Literal für die Note als String (z.B. "Sehr gut").

*\_\_***Entfernt: Abstrakte Veranstaltung**

Das abstrakte Konzept einer Veranstaltung (im Gegensatz zur konkret stattfindenden *VERANSTALTUNG*, s.u.). Eine Abstrakte Veranstaltung ist das, was i.d.R. in den Studienordnungen gemeint ist, wenn von *Veranstaltungen* die Rede ist.

Eine Abstrakte Veranstaltung hat:

- eine Anzahl an SWS (Literal: int)
- evtl. eine Anzahl an Leistungspunkten (LP) gemäß ECTS (Literal: int)
- einen Titel
- ein oder mehrere Themen
- eine Veranstaltungsnummer

Diese Klasse wird b Modell 5 differenziert in

*LV\_\_NACH\_FACHGEBIET\_HAUPTSTUDIUM*,

*LV\_\_NACH\_BEREICH\_GRUNDSTUDIUM* und *LEHRVERANSTALTUNG\_INSTINF*, da in diesem Konzept inkompatibel mit der Studienordnung.

**Entfernt: Instituts-AG**

Die Klasse ist enthalten, hat aber ab Modell 14 die Bezeichnung **Arbeitsgruppen\_Inst\_Inf\_FUB Instituts-AG**

**Entfernt: Studientyp** nach der folgenden Definition: "Der Typ eines Studiums. Bsp.: *Diplom-Grundstudium* oder *Bachelor*." Ab Modell 5 wird zwischen Studientyp und Studienabschnitt unterschieden.

**Entfernt: Studium**

Das Konzept eines Studienganges unabhängig von dessen Typ.

Die Klasse ist bis auf weiteres entfernt, da unklar ist, ob und wofür sie gebraucht wird.

**Entfernt: Veranstaltung** in dieser Definition:

"Eine räumlich und zeitlich konkret stattfindende Umsetzung einer *ABSTRAKTEN VERANSTALTUNG*.

Bsp.: *ALP III im WS02/03 bei Prof. Schweppe* im Gegensatz zum Abstraktum *ALP III*.

Zusätzlich zu den Themen der abstrakten Veranstaltung sollte eine konkrete Veranstaltung eine Beschreibung (wahrscheinlich) als Literal haben."

Die Klasse wird ab Modell 14 ersetzt durch

*LV\_\_NACH\_BEREICH\_GRUNDSTUDIUM* und  
*LV\_\_NACH\_FACHGEBIET\_HAUPTSTUDIUM*

**Entfernt: Zeit** in dieser Definition:

Stellt eine wiederkehrende, regelmäßige zeitliche Spanne dar, z.B. *Montags, 10:15h - 11:45h*.

---

## 3.5 Relationen zwischen Klassen (Properties)

### 3.5.1 Properties aus Modell 1 in Modell 15

**hatRaum:** *INSTITUTS-AG* --- *RAUM*

Eine AG, ein Dozent, eine Veranstaltung hat einen oder mehrere Räume (in denen typischerweise die Mitarbeiter der AG anzufinden sein sollten).

**hatZeit:** *TERMIN* --- *ZEIT*

Modelliert die zeitliche Komponente eines Termins.

**hatZeit:** *VERANSTALTUNG* --- *TERMIN*

Setzt eine konkret stattfindende Veranstaltung in Beziehung zu den Terminen, an denen sie stattfindet.

**istAnrechenbarFür:** *ABSTRAKTE VERANSTALTUNG* --- *STUDIENBEREICH*

Modelliert das Faktum, dass eine Veranstaltung im Sinne von erbrach-

ten SWS oder LP für einen Studienbereich anrechenbar ist.

**leitet:** *DOZENT --- VERANSTALTUNG*

Jede konkrete Veranstaltung wird von (mind.?) einem Dozenten angeboten.

**requires:** *ABSTRAKTE VERANSTALTUNG --- ABSTRAKTE VERANSTALTUNG*

Nicht alle Veranstaltungen können bedingungslos existieren. Einige Veranstaltungen sind direkt von der Existenz anderer Veranstaltungen abhängig.

Übungen sind z.B. direkt abhängig von der Existenz einer entspr. Vorlesung.

---

### 3.5.2 Entfernte Properties

*noch nicht implementiert:* **abschlussNote:** *ABSCHLUSS --- NOTE*

Ein Abschluss eines Studiums ist benotet.

*noch nicht implementiert:* **behandelt:** *ABSTRAKTE VERANSTALTUNG --- THEMA*

Eine Lehrveranstaltung behandelt ein oder mehrere Themen.

Die Property wäre, falls wir sie implementieren, eine Subproperty von *HATTHEMA*

*noch nicht implementiert:* **benotet:** *NOTE --- LEISTUNGSNACHWEIS*

Ein Leistungsnachweis kann benotet sein, d.h. er kann keine oder genau eine Note haben.

*noch nicht implementiert:* **beschreibt:** *VERANSTALTUNG --- Beschreibung (Literal)*

Eine Veranstaltung kann durch eine textuelle Beschreibung erläutert sein.

*noch nicht implementiert:* **bestanden:** *KONKRETE VERANSTALTUNG --- LEISTUNGSNACHWEIS*

Hat ein Student einen Leistungsnachweis für eine konkrete Veranstaltung erworben, besteht diese Verbindung zwischen der konkreten Veranstaltung und dem Leistungsnachweis des Studenten für diese Veranstaltung.

*noch nicht implementiert:* **besucht:** *STUDENT --- KONKRETE VERANSTALTUNG*

Wenn ein Student eine konkrete Veranstaltung besucht hat, besteht diese Verbindung zwischen ihm und der besuchten Veranstaltung.

*noch nicht implementiert:* **bietetAn:** *INSTITUTS-AG --- VERANSTALTUNG*

Modelliert die Zugehörigkeit einer konkreten Veranstaltung zu einer AG am Institut.

*noch nicht implementiert:* **dauert:** *ABSTRAKTE VERANSTALTUNG --- SWS (Literal: int)*

Eine abstrakte Veranstaltung hat eine gewisse Anzahl an Semesterwochenstunden, die sich ein Student durch den Besuch einer Umsetzung der abstrakten Veranstaltung für sein Studium anrechnen kann.

Die Property wäre eine Subproperty von *DURATION*.

*noch nicht implementiert: erbracht: STUDENT --- LEISTUNGSNACHWEIS*

Hat ein Student durch den erfolgreichen Besuch einer konkreten Veranstaltung einen Leistungsnachweis erbracht, wird dies durch diese Verbindung zwischen dem Studenten und dem Leistungsnachweis modelliert.

*noch nicht implementiert: gliedertSich: STUDIUM --- STUDIENBEREICH*

Modelliert das Faktum, dass sich ein Studiengang in ein oder mehrere thematische Bereiche gliedert.

*noch nicht implementiert: hatAbschluss: STUDIUM --- ABSCHLUSS*

Jedes Studiengang hat einen Abschluss.

*noch nicht implementiert: hatBuero: DOZENT --- RAUM*

Ein Dozent hat kein (in Ausnahmefällen) oder ein (in aller Regel) Büro.

*noch nicht implementiert: hatNummer: ABSTRAKTE VERANSTALTUNG --- VERANSTALTUNGSNUMMER*

Jede abstrakte Veranstaltung hat genau eine eindeutige Veranstaltungsnummer.

*noch nicht implementiert: hatOrt: TERMIN --- ORT*

Ein Termin hat neben einer zeitlichen Komponente eine örtliche Komponente, was durch diese Property modelliert wird.

*noch nicht implementiert: hatSekretariat: DOZENT --- RAUM*

Ein Dozent hat kein (in Ausnahmefällen) oder ein (in aller Regel) zuständiges Sekretariat.

*noch nicht implementiert: istPflicht: ABSTRAKTE VERANSTALTUNG --- STUDIENBEREICH*

Sub-Property von *ANRECHENBAR*.

Modelliert das Faktum, dass eine Veranstaltung für einen Studienbereich verpflichtend sein kann.

*noch nicht implementiert: leistet: ABSTRAKTE VERANSTALTUNG --- LP (Literal: int)*

Einer abstrakten Veranstaltung kann eine Anzahl an Leistungspunkten gemäß ECTS zugeordnet sein, in dem Sinne, dass der erfolgreiche Besuch einer Umsetzung dieser abstrakten Veranstaltung in Höhe eben dieser Leistungspunkte anrechenbar ist.

*noch nicht implementiert: minLP: STUDIENBEREICH --- LP (Literal: int)*

Ein Studienbereich kann eine minimal zu erbringende Anzahl an Leistungspunkten haben, in dem Sinne, dass durch den Besuch an konkreten Veranstaltungen dieses Bereichs in Summe mindestens diese Anzahl an LP gesammelt werden muss.

*noch nicht implementiert: minStunden: STUDIENBEREICH --- SWS (Literal: int)*

Ein Studienbereich kann eine minimal zu erbringende Anzahl an Semesterwochenstunden haben, in dem Sinne, dass durch den Besuch an konkreten Veranstaltungen dieses Bereichs in Summe mindestens diese Anzahl an SWS gesammelt werden muss.

*noch nicht implementiert:* **studiert:** *STUDENT --- STUDIUM*

Stellt die Verbindung zwischen einem Studenten und dem Studium, das er in unserem Modell absolviert, dar.

*Entfernt:* **findetStattAn.** Ersetzt durch *HATZEIT*.

*Entfernt:* **gehörtAn:** *DOZENT --- INSTITUTS-AG*

Ein Dozent kann einer AG am Institut angehören.

Property ersetzt durch *WORKSFOR* und *ISTZUSTAENDIGFUER*

*Entfernt:* **hatStudientyp:** *STUDIUM --- STUDIENTYP*

Realisiert das Faktum, dass jedes Studium einen bestimmten Studientyp hat.

Alle Properties "hat Typ" werden durch Subklassen ausgedrückt.

*Entfernt:* **hatVTyp:** *ABSTRAKTE VERANSTALTUNG ---*

*VERANSTALTUNGSTYP*

Jeder Veranstaltung ist mindestens ein, u.U. aber auch mehrere Veranstaltungstypen in dem Sinne zugeordnet, dass eine Student, der an einer konkreten *VERANSTALTUNG* teilgenommen hat, diese als Teilnahme an eben diesem Typ für sein Studium anrechnen kann.

→ Alle Properties "hat Typ" werden durch Subklassen ausgedrückt.

*entfernt:* **setztUm:** *VERANSTALTUNG --- ABSTRAKTE VERANSTALTUNG*

"Eine konkrete Veranstaltung setzt durch ihr Stattfinden eine abstrakte Veranstaltung um."

→ Alle Properties "hat Typ" werden durch Subklassen ausgedrückt!

*entfernt:* **setztVoraus** ersetzt durch **requires**

## 4 Beiträge der Teilnehmer

Richard CYGANIAK

- Recherche im Open Directory Project
- Recherche von Zeitontologien
- Spezifikation des Mini-Studiengangs
- Evaluation des Mini-Studiengangs
- Vorbereitung von Abfragen
- Installation der Webserver-Software (Tomcat)
- Auswahl und Installation eines Reasoners
- Trennung der DAML-Datei in mehrere nach logisch organisierten Namespaces

Dennis HECKERT

- Recherche nach bestehenden Klassifikationen (ACM)
- Korrektur von Fehlern des OilEd-Editors (Speicherprobleme und defekter RDF-Output)
- Modellierung: Erstellung des ersten Modells (Brainstorming-Methode): Modell 1 und 2
- Begleitung der Modellierung von Modell 5 bis 14
- Abfrage- Logik
- Programmierung von Servlets und Rendering nach HTML

Anja JENTZSCH

- Kommunikation: Einrichtung des FuWiki und der Mailingliste
- Seminar-Laptop
- Hardware: Organisation und Administration des Webserver. Kommunikation mit Arbeitsgruppe Rechnerbetrieb
- Parsing der KVV-Datenbank des Instituts für Informatik in eine DAML-Datei. Einbau ins Modell (ab Modell 25).
- Modellierung der Instanzen der Veranstaltungen und Dozenten
- Programmierung von Servlets und Rendering nach HTML

Roman SCHMIDT

- Projektleitung
- Recherche von Visualisierungs- und Modellierungs-Tools
- Evaluation der Tools
- Evaluations-Umfrage unter Seminarteilnehmern
- Klassifikationstheorie
- Inhaltsanalytische Modellierung und Ausbau des Modells (Modelle 3 bis 14)
- Modellierung der Diplom-Studienordnung, der Mitarbeiter, der Arbeitsgruppen und Projekte des Instituts (Modell 3 bis Modell 24).
- Sitzungsprotokolle
- Dokumentation

## 5 Quellen

- CYGANIAK, Richard (=PROJEKTSEMINAR SEMANTIC WEB, d.i. CYGANIAK, HECKERT, JENTZSCH, SCHMIDT) (SS2003/18.7.2003), *Das gesamte Modell des Instituts für Informatik, FU Berlin*, Berlin: Freie Universität Berlin, <http://projects.mi.fu-berlin.de/semweb/model/all.daml>, <http://projects.mi.fu-berlin.de/semweb/model/documentation.html>
- CYGANIAK, Richard (1.7.2003 04:25:41), [*Nbi\_p\_sw*] *Kurzer Zwischenstand von der Reasoner-Front*, Mail an [nbi\\_p\\_sw@lists.spline.inf.fu-berlin.de](mailto:nbi_p_sw@lists.spline.inf.fu-berlin.de).
- Cyganiak, Richard (2.7. 2003) [*Nbi\_p\_sw*] *CVS / Projektserver*, Mail an [nbi\\_p\\_sw@lists.spline.inf.fu-berlin.de](mailto:nbi_p_sw@lists.spline.inf.fu-berlin.de) (Seminarteilnehmer) (Juni-Juli 2003), [*cvs-semweb*] ..., [nbi\\_p\\_sw@lists.spline.inf.fu-berlin.de](mailto:nbi_p_sw@lists.spline.inf.fu-berlin.de).
- CYGANIAK, Richard (5. Juni 2003), "SWMiniModellAnforderungen", *FuWiki*, Berlin: FU, <http://fu.freezope.org/fuwiki/SWMiniModellAnforderungen>
- CYGANIAK, Richard (7.6.2003 19:35:38), [*Nbi\_p\_sw*] *Zwischenstand*, Mail an [nbi\\_p\\_sw@lists.spline.inf.fu-berlin.de](mailto:nbi_p_sw@lists.spline.inf.fu-berlin.de).
- HECKERT, Dennis (Download 8.7.2003) "Index of /~heckert/PsemWeb/SED", *Userpages/~heckert*, Berlin: Freie Universität, <http://page.inf.fu-berlin.de/~heckert/PsemWeb/SED/>.
- HECKERT, Dennis; Roman SCHMIDT et al. (20.5.2003), "Erfahrungsberichte mit Tools aus dem Seminar", *FuWiki*, Berlin: FU, <http://fu.freezope.org/fuwiki/ErfahrungsberichteDamlOilTools>
- HECKERT, Dennis; Roman SCHMIDT et al. (20.5.2003), "Erfahrungsberichte mit Tools aus dem Seminar", *FuWiki*, Berlin: FU, <http://fu.freezope.org/fuwiki/ErfahrungsberichteDamlOilTools>
- PROJEKTSEMINAR SEMANTIC WEB (Richard CYGANIAK, Dennis HECKERT, Anja JENTZSCH, Roman SCHMIDT) (SS2003/18.7.2003), *Semantische Modellierung des Instituts*, Berlin: Freie Universität Berlin, <http://projects.mi.fu-berlin.de/semweb/>
- SCHMIDT, Roman (8. Juni 2003), "Protokoll der Projektsitzungen vom 22. und 27. Mai 2003", *FuWiki*, Berlin: FU, <http://fu.freezope.org/fuwiki/ProjektSemanticWebProtokoll4>
- SCHMIDT, Roman (9.5.2003), "Protokoll der Projektsitzung vom 8. Mai 2003", *FuWiki* Berlin: FU, <http://fu.freezope.org/fuwiki/ProjektSemanticWebProtokoll2>
- TOLKSDORF, Robert (Download 25.6.2003), *Projekt Semantic Web (SoSe 2003) 19565 PJ*, Berlin: Freie Universität, FB Mathematik und Informatik, Institut für Informatik, AG Netzbasierende Informationssysteme, [www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/P\\_SW/default.htm](http://www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/P_SW/default.htm).
- TOLKSDORF, Robert (Download 25.6.2003), *Seminar Grundlagen des Semantic Web (SoSe 2003) 19579 P*, Berlin: Freie Universität Berlin, Fachbereich Mathematik und Informatik, Institut für Informatik, AG Netzbasierende Informationssysteme, [www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/S\\_SW/default.htm](http://www.inf.fu-berlin.de/inst/ag-nbi/lehre/03/S_SW/default.htm).

## 6 Gliederung

<u>Abstract</u>	<u>1</u>
<u>1 Arbeitsziele</u>	<u>3</u>
<u>2 Durchführung</u>	<u>4</u>
2.1 Modellbildung (A1)	4
2.1.1 Modellinhalt	4
2.1.2 Modelloutput	5
2.1.3 Modellierungsrichtlinien und Repräsentation	5
2.1.4 Inhaltliche Erarbeitung des Modells	7
2.1.5 Ergebnis: Meilenstein W7	7
2.2 A2: Infrastruktur	8
2.2.1 Standard / Code: DAML-OIL	8
2.2.2 Auswahl eines Editors	8
2.2.3 Reparatur des OilEd-Outputs	9
2.2.4 Implementierung der Hardware: Webserver	10
2.2.5 Implementierung der Abfrage-Software	10
2.2.6 Online-Informationssystem	11
<u>3 Ergebnisse</u>	<u>11</u>
3.1 Meilensteine	12
3.2 Das Online-Modell und seine Namespaces	12
3.2.1 Gesamtmodell	12
3.2.2 Teile des Modells	12
3.3 Das Modell in der Entwicklung	13
3.3.1 Modell Eins	13
3.3.2 Modell Zwei	13
3.3.3 Modell Acht	14
3.3.4 Modell 14	14
3.3.5 Modell 24	14
3.3.6 Modell 25 ff.	14
3.3.7 Modell, getrennt nach Namespaces	14
3.4 Klassen des Modells	15
3.4.1 Klassen aus Modell 1 im Modell 15	15
3.4.2 Entfernte Klassen	17
3.5 Relationen zwischen Klassen (Properties):	18
3.5.1 Properties aus Modell 1 in Modell 15	18
3.5.2 Entfernte Properties	19
<u>4 Beiträge der Teilnehmer</u>	<u>22</u>
<u>5 Quellen</u>	<u>23</u>