



Fachhochschule Darmstadt  
*University of Applied Sciences*

**Diplomarbeit**

**Ontologisches Portal der FHD –  
Konzeption und Prototypentwicklung**

**Joachim Müller**

**05.Juni 2003**

**Fachbereich Informations- und  
Wissensmanagement**

**Referenten:**

**Prof. Dr.-Ing. Gerhard Knorz**

**Prof. Dr. rer. nat Elke Lang**

# Erklärung

Hiermit versichere ich, die vorliegende Diplomarbeit selbständig erarbeitet und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet zu haben.

Mit der Ausleihe dieser Arbeit erkläre ich mich einverstanden.

Darmstadt, 05. Juni 2003

---

Joachim Müller

# Inhaltsverzeichnis

<b>1.</b>	<b>EINLEITUNG</b>	<b>6</b>
<b>2.</b>	<b>WISSENSCHAFTLICHER RAHMEN</b>	<b>10</b>
2.1	Definition und Abgrenzung von Wissensmanagement	10
2.1.1	Allgemeiner Überblick über das Wissensmanagement	10
2.1.2	Die Bausteine des Wissensmanagements	15
2.1.2.1	Wissensziele	15
2.1.2.2	Wissensidentifikation	17
2.1.2.3	Wissenserwerb	18
2.1.2.4	Wissensentwicklung	19
2.1.2.5	Wissensverteilung	20
2.1.2.6	Wissensnutzung	21
2.1.2.7	Wissensbewahrung	23
2.1.2.8	Wissensbewertung	24
2.2	Anforderungen an ein Wissensmanagementsystem und Schwachpunkte bestehender Systeme	25
2.3	Ontologien	27
2.3.1	Definition	28
2.3.2	Ontologien für das Wissensmanagement	32
2.3.2.1	Der Wissens – Metaprozess	33
2.3.2.2	Der Wissensprozess	35
<b>3.</b>	<b>DAS PORTAL DER FACHHOCHSCHULE DARMSTADT – ANFORDERUNGEN UND BEDARF</b>	<b>38</b>
3.1	Wissensmanagement im Hinblick auf die Fachhochschule	38
3.2	Definitionsbereich von Intranetportalen	39
3.3	Bedarfserhebung an ein Intranet-Portal der Fachhochschule Darmstadt	42

<b>4.</b>	<b>DIE ONTOLOGIE DES FHD-PORTALS – MODELLIERUNG UND ANBINDUNG</b>	<b>45</b>
4.1	Die Software-Suite „K-Infinity“	45
4.1.1	Systemkomponenten der „K-Infinity“ Suite	46
4.2	Status Quo der Ontologie	48
4.3	Methodologie zur Evaluierung der Ontologie	50
4.4	Anbindung der Ontologie an ein Webfrontend – Das „Struts Application Framework“	55
4.4.1	Involvierte Technologien	56
4.4.2	Die „Model-View-Controller“-Architektur	59
4.4.3	Ablauf einer Suchanfrage	62
<b>5.</b>	<b>KONZEPTE FÜR DAS PORTAL DER FACHHOCHSCHULE DARMSTADT</b>	<b>64</b>
5.1	Konzeption innerhalb des Wissens-Metaprozesses	64
5.2	Konzeption innerhalb des Wissensprozesses	68
5.2.1	Konzepte für die Wissensgenerierung	69
5.2.2	Konzepte für den Wissensimport und die Integration vorhandener Systeme	71
5.2.3	Konzepte für die Wissenserfassung	77
5.2.4	Konzepte für den Wissenszugriff / die Wissensnutzung	78
5.3	Konzepte für das Kommunikationsmodell	89
5.4	Profilgestaltung	96
5.5	Visualisierung und Ergonomie	99
<b>6.</b>	<b>ENTWICKLUNG EINES PROTOTYPEN DES ONTOLOGISCHEN FHD-PORTALS</b>	<b>104</b>
6.1	Rahmenbedingungen	104
6.2	Das Nutzerszenario	105
6.3	Ein Blick auf den Prototypen	107
6.3.1	Gestalterische Grundsätze	108
6.4	Das Rechtesystem / Profilgestaltung	110

6.5	Die Systemkomponenten	113
6.5.1	Die Push-Dienste	113
6.5.2	Die Menüdienste	116
6.5.3	Die Suchfunktionalitäten	120
6.5.4	Der Verlauf	121
6.5.5	Der Kalender	121
6.6	Use Case: Beantragung von Fördergeldern für ein Forschungsprojekt	122
6.6.1	Prozessbeschreibung	122
6.6.2	Realisierung der Antragsstellung mithilfe des FHD-Portals	124
6.5.3	Verfügbarmachen der neuen Anträge	128
<b>7.</b>	<b>DISKUSSION DER ERGEBNISSE</b>	<b>129</b>
7.1	Konzeption	129
7.2	Prototyp	130
7.3	Modellierung mit der „K-Infinity“-Software Suite	131
<b>8.</b>	<b>ZUSAMMENFASSUNG UND AUSBLICK</b>	<b>133</b>
8.1	Zusammenfassung	133
8.2	Ausblick	134
	<b>LITERATURVERZEICHNIS</b>	<b>135</b>
	<b>ABBILDUNGS- UND TABELLENVERZEICHNIS</b>	<b>140</b>
	<b>ANHANG</b>	<b>142</b>
	Fragebogen zur Erhebung des studentischen Bedarfs	142
	Ablauf einer Suchanfrage	143
	Kompetenzfragen hinsichtlich der Yellow Pages	148
	Funktionsbeschreibung der Prototypdateien	149
	Hinweise bezüglich des Lesens der Prototypdateien	154
	Hardware-Empfehlung für die Anschaffung eines Portalservers	155
	Digitaler Anhang	156

# 1. Einleitung

## Danksagungen

Diese Diplomarbeit wurde in Zusammenarbeit mit dem Präsidium der Fachhochschule Darmstadt und der „intelligent views GmbH“ , Julius-Reiber-Straße 17 in 64293 Darmstadt durchgeführt.

Mein Dank geht zunächst an den Vizepräsidenten der Fachhochschule Darmstadt, Herrn Prof. Dr.-Ing. Gerhard Knorz, für die Betreuung dieser Arbeit und die geduldige Beantwortung der zahlreichen Fragen.

Frau Prof. Dr. rer. nat. Elke Lang gilt mein Dank für die Übernahme des Korreferats.

Herrn Dipl. Inf. Jonny Newald, Mitarbeiter der Firma „intelligent views“ und Herrn Dipl. Inf. Peter Baumgart danke ich für die Unterstützung bei der technischen Umsetzung des Prototypen.

Herzlichen Dank an Frau Dr. Ute Jochem vom „Zentrum für Forschung und Entwicklung“ für die freundliche Hilfe bei inhaltlichen Fragen im Bereich „Forschung und Entwicklung“.

Für die Korrekturarbeit danke ich Herrn Dipl. Ing. Wendelin Müller, Herrn Oliver Baltz und Frau Maria-Inti Metzendorf.

Schließlich geht mein Dank an meine Familie und meine Freunde für die außerfachliche Unterstützung während dieser intensiven Zeit.

## **Motivation und Problemstellung**

„People can't share knowledge if they don't speak a common language“

Thomas H. Davenport, Lawrence Prusak [Davenport et al.]

Wissensmanagement ist heutzutage kein Fremdwort mehr. Als Folge einer intensiven Abhandlung des Themas innerhalb einschlägiger Literatur gibt es nur wenige mittelständische Unternehmen, die nicht schon zumindest über die Einführung einer Wissensmanagement-Lösung nachgedacht hätten. Probleme aktueller Ansätze sind häufig die Fokussierung technischer Lösungen ohne begleitende Beratung der Mitarbeiter und das Vernachlässigen von Anpassungen in der Firmenpolitik, die Wissensmanagement-Maßnahmen im Sande verlaufen lassen. Zusätzlich handelt es sich bei den gängigen Lösungen zumeist um Maßnahmen auf der Ebene des Daten- und Informationsmanagements, die als isolierte Lösungen den Kontext des Unternehmens nicht berücksichtigen.

Die Fachliteratur betrachtet seit geraumer Zeit Ontologien als „state-of-the-art“ Ansatz, der dieser Misere bestehender Systeme entgegensteht. Das Schaffen einer gemeinsamen Sprache zur Beschreibung von Wissensinhalten, sowie die Bereitstellung von Werkzeugen, welche in der Lage sind, die Inhalte zu nutzen, werden dabei oftmals als „key-issues“ bezüglich eines erfolgreichen und integrieren Wissensmanagements betrachtet. Mit dem Einsatz eines ontologiebasierten Systems können diese hinreichend ausgefüllt werden. Vor allem die Möglichkeit der Integration bestehender Systeme und die Schaffung einer gemeinsamen Kommunikationsbasis zwischen diesen bringt ontologiebasierte Systeme in Vorfahrtsstellung gegenüber klassischen Wissensmanagementsystemen. Die Möglichkeit, dem Mitarbeiter einen „single point of access“ hinsichtlich allen innerhalb der Organisation verfügbaren Wissens zu bieten, ist dabei als eines der Hauptziele zu erachten. Das Erreichen eines gemeinsamen Verständnisses – auch in Bezug auf Prozesse und Vorgänge innerhalb eines Unternehmens – und der daraus resultierende effiziente Austausch von Daten sowohl zwischen Menschen, als auch zwischen Maschinen<sup>1</sup>, kommt dabei der Vernetzung der Geschäftswelt entgegen und hilft, ineffiziente Vorgänge und Fehler, resultierend aus Unwissen über bestimmte Prozesse, zu vermeiden. Doch von der Literatur als „Killer-Technologie“ bejubelt, fehlt es bislang in der Praxis oft noch an veranschaulichenden Anwendungen ontologischer Technologien. Ontologieneditoren übersteigen in ihrer Anzahl mittlerweile das Dutzend, die Integration einer Ontologie in ein effizient nutzbares und ergonomisches System kommt jedoch meist zu kurz.

---

<sup>1</sup> Aus diesem Grunde müsste das Davenport'sche Eingangszitat neben dem Aspekt „People“ um den Aspekt „machines“ erweitert werden.

Auch die Fachhochschule Darmstadt (FHD) hat die Probleme ineffizienten Arbeitens, hoher Fehlerquoten und Mehrfacharbeit erkannt, die aus mangelnder Transparenz und mangelndem Wissen über die entsprechenden Prozesse resultieren. Im Frühjahr 2002 wurde daher die Senatskommission „Wissensmanagement“ an der FHD ins Leben gerufen, deren Aufgabe die Einführung eines umfassenden, hochschulweiten Wissensmanagements ist. Aufgabe dieser Kommission, unter Vorsitz des Vizepräsidenten Prof. Dr.-Ing. Gerhard Knorz, ist es, Bedingungen zu schaffen, die in der Fachhochschule vorhandenes Wissen über Ressourcen und ablaufende Prozesse verfügbar und leicht zugänglich machen. Als Ziele der Kommission definierte man dabei unter anderem die Einführung eines Portals zur Erhöhung der Transparenz innerhalb der Institution „Fachhochschule Darmstadt“. Dieses, zunächst als Intranetportal geplante System<sup>2</sup>, soll als zentrale Anlaufstelle für jeden Mitarbeiter der Fachhochschule dienen und jegliche in der FH vorhandenen Informationen in dem dahinterliegenden Wissensmodell vereinen. Vor allem die standortbedingte lokale Streuung der einzelnen Abteilungen, Organisationseinheiten und Fachbereiche, sowie die heterogene Struktur bestehender Informationssysteme sind Schlüsselaspekte, die es dabei zu betrachten gilt.

Im Zuge der Arbeit der Kommission wurde mit der Firma „intelligent views“ ein Vertrag ausgehandelt, der die Lieferung einer technologischen Basis (die sogenannte „K-Infinity“ Software Suite) zur Erstellung eines auf einer Ontologie basierenden Portals als Lösung für das Wissensmanagement beinhaltet. Die Entwicklung einer Ontologie als gemeinsames Verständnis des Weltausschnittes „Hochschulwelt“ begann dabei bereits im Frühjahr 2002 und wurde im Wintersemester 2002/2003 im Zuge eines Hochschulprojektes des damaligen 7. Semesters des Fachbereiches „Informations- und Wissensmanagements“ forciert.

### **Aufgabenstellung**

In der vorliegenden Diplomarbeit soll eine Konzeption eines Portals für die FHD entstehen, welche die Vorteile eines ontologiebasierten Ansatzes demonstriert und Lösungsansätze für die an der FH bestehenden Problematiken liefert.

Im Zuge dessen gilt es, das Stadium des bereits existierenden, aber noch nicht umfassenden Wissensmodells zu hinterfragen, über ein Anbindung des Wissensmodells nachzudenken, Möglichkeiten einer Integration bestehender Systeme zu erörtern, fundierte

---

<sup>2</sup> Eine spätere Erweiterung auf die Internetebene ist geplant und wird in dieser Arbeit bereits ansatzweise behandelt.



Konzepte für das Wissensmanagement herauszuarbeiten und schließlich die gewonnenen Einsichten beispielhaft innerhalb eines Prototypen umzusetzen.

Die vorgestellten Ergebnisse sind dabei aufgrund ihrer kurzen Bearbeitungszeit als Ansatz im Sinne eines „proof-of-concept“ zu werten. Eine umfassende Realisierung des Portals wird im Anschluss an diese Arbeit stattfinden.

### **Aufbau der Arbeit**

Die Diplomarbeit umfasst insgesamt acht Kapitel. Diese gliedern sich dabei wie folgt:

In dem der Einleitung folgenden Kapitel 2 wird der wissenschaftliche Rahmen dieser Arbeit aufgezeigt. Einem allgemeinen Überblick über Konzepte des Wissensmanagements und Anforderungen an ein Wissensmanagementsystem folgt ein kurzer Überblick über den Ansatz der Ontologien. Die Synthese dieser Abschnitte erfolgt am Ende des 2. Kapitels mit der Herausarbeitung eines methodischen Rahmens zur Erstellung eines ontologiebasierten Wissensmanagementsystems.

Kapitel 3 beschreibt die Anforderungen, die eine Institution wie die Fachhochschule an ein Wissensmanagementsystem stellt und schildert den konkreten Bedarf der Mitarbeiter, der in einzelnen Workshops erarbeitet wurde.

In Kapitel 4 wird auf die Ontologie als Wissensmodell hinter dem geplanten Portal eingegangen. Nach einem kurzen Einblick in die zur Erstellung verwendete Software wird eine Methodologie erarbeitet, welche den Hintergrund zur Überarbeitung der Ontologie liefert. Ferner wird die Anbindung des Wissensmodells in der zweiten Hälfte des vierten Kapitels exemplarisch aufgezeigt.

Die eigentlichen Konzepte für das Portal werden in Kapitel 5 herausgearbeitet und exemplarisch in Kapitel 6 im Zuge einer Prototypisierung umgesetzt.

Die Diskussion der Ergebnisse findet sich in Kapitel 7, Vorschläge über die weitere Vorgehensweise bei der Einführung des Portals werden in Kapitel 8 gegeben.

# 1. Wissenschaftlicher Rahmen

Im zweiten Kapitel dieser Diplomarbeit wird die Arbeit thematisch in ein wissenschaftliches Umfeld eingeordnet. Zunächst wird ein Überblick über allgemeine Anforderungen an ein modernes Wissensmanagement gegeben und im zweiten Abschnitt Schwachpunkte existierender Systeme für das Wissensmanagement aufgezeigt. Der dritte Abschnitt dieses Kapitels gibt einen Überblick über Ontologien, zunächst auf Definitionsebene und schließlich im Kontext von Wissensmanagementsystemen. Im Zuge dessen wird eine Methodologie zur Entwicklung eines ontologiebasierten Wissensmanagementsystem ausgearbeitet, welche in Kapitel 5 dieser Arbeit den Rahmen für die Konzeption des Portals liefert.

## 2.1 *Definition und Abgrenzung von Wissensmanagement*

Der folgende Abschnitt soll einen Überblick über das betriebliche Wissensmanagement geben. Dabei werden zunächst Konzepte vorgestellt, die in der einschlägigen Literatur gemeinhin als gültig betrachtet werden. Anhand einzelner Definitionen soll so schrittweise der Kontext des Wissensmanagements aufgebaut werden. Darüber hinaus findet im zweiten Abschnitt dieses Unterkapitels eine Einordnung der Fachhochschule als Institution in die unternehmerische Welt statt, um den Bogen zu den speziellen Anforderungen an ein Wissensmanagement an Hochschulen zu spannen.

### 2.1.1 **Allgemeiner Überblick über das Wissensmanagement**

“Where is the wisdom we have lost in knowledge?  
Where is the knowledge we have lost in information?”  
Thomas Stearns Elliot [Elliot]

Die Zukunft von Unternehmen wird davon abhängen, wie schnell sie das für sie relevante interne und externe Wissen managen können. Zunehmende Marktdynamik, kürzere Produktlebenszyklen, eine zunehmende Komplexität von Produkten und die Zunahme der Bedeutung von Informationstechnologie, bedingt durch die wachsende Informationsflut, haben Wissen zu einem Gut werden lassen, das systematisch und aktiv gesteuert und entwickelt werden muss. Analog zum Finanzkapital hat dabei eine Mehrung des Wissenskapitals einen Wertezuwachs zur Folge, den das Unternehmen vorteilhaft nutzen kann. Wichtig ist dabei, dass Wissensmanagement nicht an den Pforten des Unternehmens endet, sondern vielmehr darüber hinaus externe Mitarbeiter, Lieferanten, Kunden

etc. mit einbezieht. Eine Globalisierung von Information und Wissen hat stattgefunden, die selbst kleineren Marktteilnehmern ein hohes Maß an Flexibilität und Geschwindigkeit auf den Informations- und Wissensmärkten abverlangt [Zürich]. Wissen ist dabei nicht mit Information gleichzusetzen. Ziel von Wissensmanagement ist es vielmehr, aus Informationen Wissen zu gewinnen und dieses gewinnbringend und als Wettbewerbsvorteil einzusetzen. Anhand der **Wissenstreppe** können die Grundbegriffe für eine wissensorientierte Unternehmensführung erklärt werden:

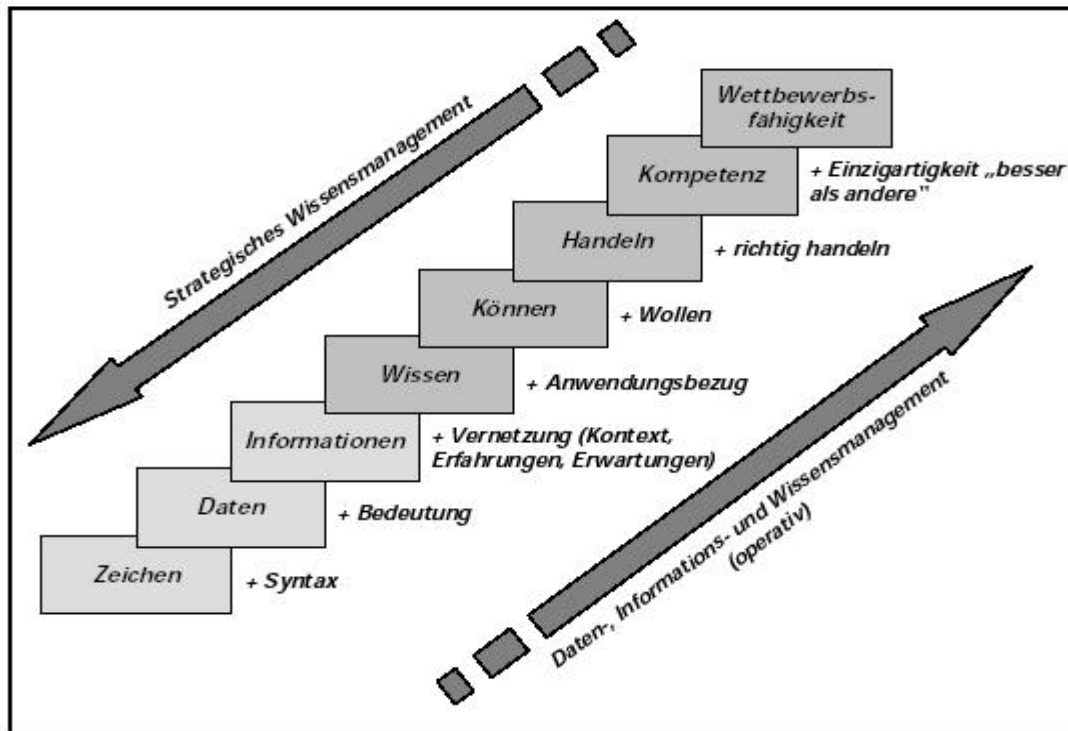


Abb. 1: Die Wissenstreppe (aus [kpmg2002])

Zeichen werden durch Ordnungsregeln (=Syntax) zu Daten, die für sich noch nicht interpretierbar sind („k25jdoimn“ wäre ein Datenbeispiel). Durch Hinzufügen von Bedeutung werden Daten zu Information (Bspw.: Die Tastatur hat die Produktnummer „k25jdoimn“). Diese Informationen sind für den Betrachter wertlos, wenn er sie nicht mit anderer Information in Verbindung setzen kann. Hieraus ergibt sich die Definition von Wissen: Bei Wissen handelt es sich um zweckdienlich vernetzte Information. Es entsteht durch Verarbeitung durch das Bewusstsein. Freier ausgedrückt bedeutet dies, dass Information das „Rohmaterial für Wissen“ ([Probst et al.], S. 38) ist und gleichzeitig auch der Träger. Die Interpretation von Information kann sehr unterschiedlich ausfallen.

Ein Kopfnicken wird hierzulande als Zustimmung aufgefasst, in Griechenland hingegen als Ablehnung<sup>3</sup>. Der Kontext ist also entscheidend und Wissen demnach geprägt von persönlichen Erfahrungen immer kontextspezifisch:

„Wissen ist die Gesamtheit der Kenntnisse, Fähigkeiten und Fertigkeiten, die Personen zur Lösung von Problemen einsetzen.“ [Probst et. al.], Seite 46.

Dieses Zitat verdeutlicht darüber hinaus, dass Wissen im Gegensatz zu Daten und Information immer an Personen gebunden ist.

[Englert] beschreibt daher den Misserfolg vieler Wissensmanagementlösungen als Folge des falschen Fixpunktes beim Ansatz. So ist der fehlende Fokus auf die Mitarbeiter und den kommunikativen Aspekt häufig für ein Scheitern verantwortlich.

Wissen wird für das Unternehmen nur dann nützlich, wenn sich aus dem Wissen ein Können entwickelt, d.h. aus „wissen was“ wird „wissen wie“. Es genügt nicht Wissen zu erlangen (etwa in Fortbildungsmaßnahmen), sondern auch die Anwendung von Wissen muss erlernt werden. Um Wissen in Handlungen umzusetzen, kommt zu dem Können des Mitarbeiters noch zusätzlich ein Wollen hinzu. Nur wenn der Wissende kann **und** will, wird sein Wissen handlungsorientiert eingesetzt werden. Erst das Handeln liefert messbare Ergebnisse darüber, wie der Handelnde aus Informationen Wissen generiert und dieses Wissen zur Lösung gewisser Aufgaben einsetzt. Dieses Handeln wird auch als Kompetenz bezeichnet. Als besonders wichtig werden die sogenannten Kernkompetenzen erachtet. Diese Kernkompetenzen sind ein Verbund von Fähigkeiten, die auf explizitem<sup>4</sup> und verborgenem (implizitem) Wissen beruhen, unabhängig von einzelnen Produkten sind und sich durch zeitliche Stabilität auszeichnen. Zusätzlich sind Kernkompetenzen nicht leicht imitierbar, eröffnen dem Unternehmen neue Märkte und verschaffen ihm so einen Wettbewerbsvorteil.

Wissensmanagement bedeutet, alle Stufen der Wissenstreppe zu gestalten. Ist eine Stufe der Treppe nicht ausgefüllt, so entstehen Hindernisse im Geschäftsgeschehen.

In einem Unternehmen existieren verschiedene Arten von Wissen. Dabei unterscheidet man zunächst, wie bereits erwähnt, explizites von implizitem Wissen (letzteres in [Probst et al.], S. 195 auch als „Handlungswissen“ bezeichnet). Ferner wird externes von inter-

---

<sup>3</sup> Anmerkung: Bei der Betrachtungsweise der Geste des Kopfnickens handelt es sich nicht um eine Störung auf Datenebene. Die Dateneinheit an sich („das Nicken mit dem Kopf“) ist eindeutig für jeden sichtbar (das Ergebnis also: jemand hat mit dem Kopf genickt). Vielmehr die Einordnung der Information, was das Kopfnicken im kulturellen Zusammenhang bedeutet, ruft das Missverständnis hervor. Demzufolge ist die Wissensebene die entscheidende für die Interpretation

<sup>4</sup> Explizites Wissen: Fest in der Firma, auf Trägern wie Dokumenten, Datenbanken, Notizen, Intranetseiten etc. vorhandenes Wissen.

nem Wissen und altes von vorhandenem Wissen unterschieden. In [Knorz2000] bildet sich daraus der Wissenswürfel zur Klassifizierung von unterschiedlichen Wissensarten (siehe Abbildung 2). Das Wissensmanagement muss in seiner Implementierung alle Arten von Wissen berücksichtigen.

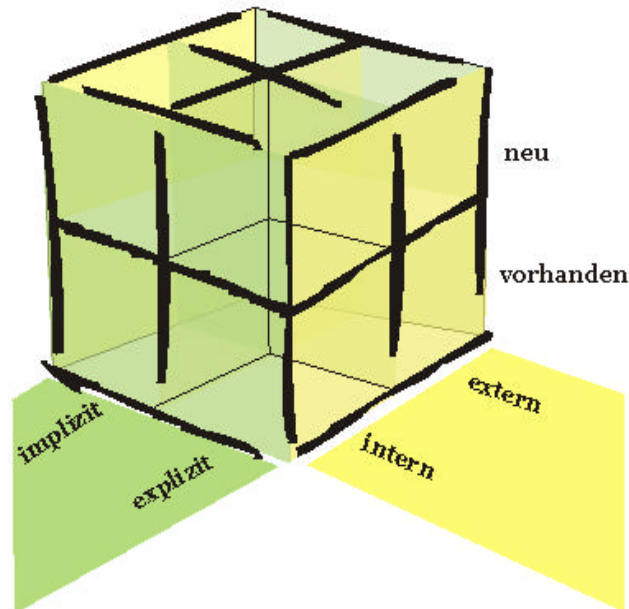


Abb. 2: Der Wissenswürfel

Aus der Wissenstreppe lassen sich folgende Handlungsfelder des Informations- und Wissensmanagements ableiten:

- Das **strategische Wissensmanagement** durchläuft die Treppe von oben nach unten, um die Frage zu beantworten, welche Kompetenzen (und abgeleitet: welches Wissen und Können) benötigt wird, um wettbewerbsfähig zu sein. Wissensziele leiten sich aus Unternehmenszielen ab und geben die Stoßrichtung des Wissensmanagements an.
- Das **operative Wissensmanagement** hat die Vernetzung von Information zu Wissen, Können und Handeln zum Hauptziel. Wichtig ist dabei das Transferieren von kollektivem Wissen in individuelles Wissen und anders herum. Eine große Bedeutung erhält hierbei der Transfer von implizitem Wissen in explizites Wissen. Rahmenbedingungen, die Anreize für den Wissensaufbau schaffen, müssen entwickelt werden.
- **Informations- und Datenmanagement:** Grundlage des Wissensmanagements ist die Identifizierung, Speicherung und Verteilung von Daten und Informationen.

Wissensmanagement ist stets ein umfassender Ansatz, der sowohl die Einführung neuer Informations- und Kommunikationssysteme beinhaltet, als auch eine Anpassung der Unternehmensstrategie zur Folge hat und den einzelnen Mitarbeiter als konkreten Faktor berücksichtigt. Dabei müssen sowohl strategische als auch kulturelle, organisatorische sowie technologische Aspekte vereint werden.

Nach [Probst et al.] hat es sich bewährt, Wissensmanagement in Bausteine zu zerlegen. Diese zeigen sich in Abbildung 3:

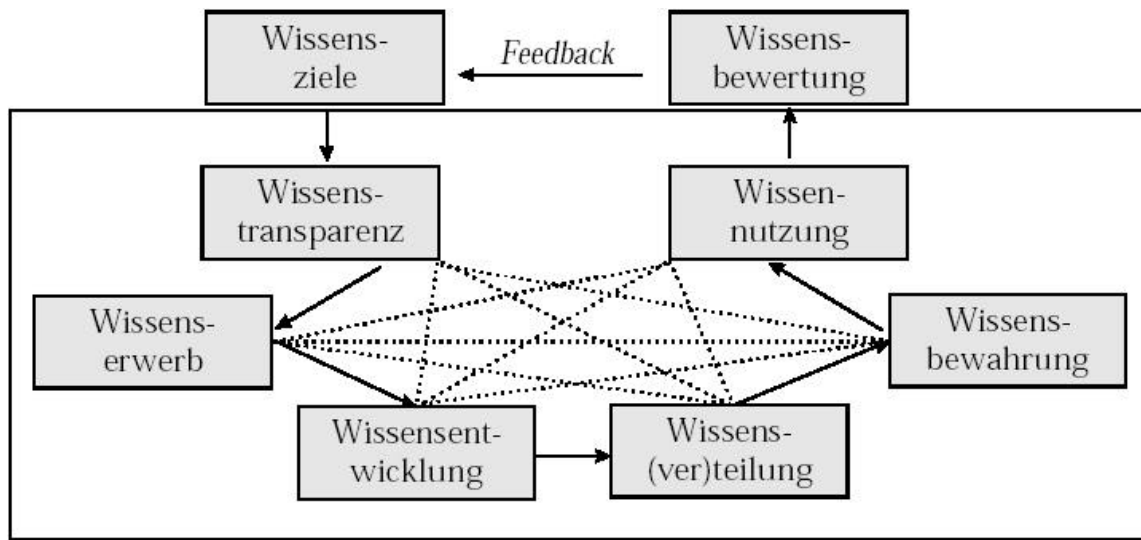


Abb.3: Bausteine des ganzheitlichen Wissensmanagements nach Probst (aus [Zeeb])

Diese Bausteine dienen als Suchraster für die Behebung von Problemen beim Wissensmanagement. Sie strukturieren den Prozess des Wissensmanagements und zeigen Interventionsmöglichkeiten sinnvoll auf.

Dabei folgen sie zwei logischen Kreisläufen: Der äußere Kreislauf bildet, vergleichbar dem betrieblichen Controlling, einen traditionellen Managementprozess mit den Komponenten Zielsetzung (Wissensziele) sowie Messung (Wissensbewertung) ab und geht vor allem auf die strategischen Aspekte des Wissensmanagements ein.

Zusätzlich bildet sich ein innerer Kreislauf, der den Prozess „Umsetzung“ in die Komponenten Wissensidentifikation, Wissenserwerb, Wissensentwicklung, Wissensverteilung, Wissensnutzung und Wissensbewahrung aufteilt. Die einzelnen Komponenten stehen dabei mehr oder weniger in Bezug zueinander (siehe Abschnitt 2.1.2).

## 2.1.2 Die Bausteine des Wissensmanagements

### 2.1.2.1 Wissensziele

Wissensziele unterteilen sich laut [KPMG2001] und [Zürich] in drei Arten von Zielen:

- **Normative Ziele:** Schaffung einer wissensbasierten Unternehmenskultur und kultureller Rahmenbedingungen. Normative Ziele schaffen die Voraussetzungen für die Ziele im strategischen und operativen Bereich.
- **Strategische Ziele:** sichern die Kohärenz und konsequente Weiterentwicklung von Kompetenzen und bestimmen den aktuellen und zukünftigen Kompetenzbedarf des Unternehmens.
- **Operative Ziele:** sorgen für die Umsetzung des Wissensmanagements und für eine Konkretisierung der Zielvorhaben. Wissensziele des normativen und strategischen Bereichs werden in konkrete operationalisierbare Teilziele übersetzt, welche die Wissensinfrastruktur des Unternehmens optimieren und aktiv eingreifen.

Ähnlich dem klassischen Controlling, müssen die strategischen und die operativen Ziele aufeinander abgestimmt sein. Erst durch die Konkretisierung der Ziele können geeignete Instrumente zu deren Erfüllung eingesetzt werden.

In vielen Unternehmen spielen die operativen Ziele oftmals die größte Rolle bei der Einführung von Wissensmanagement. Dies hat zur Folge, dass IT-Systeme nicht selten die oberste Priorität besitzen und kulturelle und strategische Aspekte vernachlässigt werden. Die Folge davon sind häufig Datenfriedhöfe und ungenutzte Systeme, wie eine Studie der KPMG [KPMG1999] herausgefunden hat. Heiko Roehl spricht in diesem Zusammenhang in [Roehl et al.] gar vom Wissensmanagement als Vehikel gen Utopia. Demzufolge liegt das Scheitern von Wissensmanagement selten an den IT-Systemen, viel häufiger ist eine Vernachlässigung der anderen, nicht technischen (insbesondere der sozialen) Faktoren als Ursache auszumachen. Microsoft Gründer Bill Gates wird hinsichtlich dieser Problematik in [Nohr] wie folgt zitiert:

"Der Begriff Wissensmanagement meint [...] kein Software-Paket oder eine Software-Klasse. Wissensmanagement beginnt noch nicht einmal mit der Technologie. Es beginnt mit Unternehmenszielen und Arbeitsabläufen und der Erkenntnis über die Notwendigkeit, Informationen auszutauschen."

Eine Vernachlässigung der strategischen Ziele ist oftmals durch fehlende notwendige Anreize, sein Wissen zu teilen und Zusammenarbeit zu fördern zu begründen. Fehlt diese Bereitschaft, wird ein Nutzungserfolg ausbleiben, ebenso wie bei einer nicht ausreichenden Nutzung der Kernkompetenzen und einer fehlenden Anpassung an die Unternehmensstrategie. Da aus den oben genannten Zeitgründen, die aus der oftmals ständigen Personalknappheit resultieren, die strategische Seite zu kurz kommt, lassen sich keine Wissensziele ableiten, woraus wiederum resultiert, dass keine operativen Maßnahmen getroffen werden können. Als Folge davon ist eine systematische Planung und Organisation von Prozessen nur noch schwer möglich. Daher ist es für Unternehmen jeder Größe essentiell, Unternehmensziele zu analysieren, diese an das jeweilige Umfeld (Marktfaktoren, Kunden, Lieferanten, Kunden, Wettbewerber) anzupassen und darauf aufbauend eine Wissensmanagementstrategie zu entwickeln. Wissensmanagement implementieren heißt zunächst Wissensaufbau und –nutzen zu organisieren. Hierbei ergeben sich folgende Leitfragen:

- Wie wichtig ist das Wissen für den Geschäftserfolg?
- Welches Wissen haben wir heute, welches benötigen wir zur Festigung unserer Wirtschaftsposition?
- Wie gehen wir mit der Ressource Wissen um?
- Wie sollte unser Unternehmen umgestaltet werden, damit es dem aktuellen und dem zukünftigen Wissenswettbewerb gewachsen ist?

Dabei gilt es zu fragen, welche strategischen Ziele durch die Mobilisierung von Wissen vornehmlich unterstützt werden sollen. Die wichtigsten Ziele sind zumeist eine hohe Produktqualität, Kundennähe und Produktführerschaft.



### 2.1.2.2 Wissensidentifikation



Abb.4: Wissensidentifikation

Der Baustein Wissensidentifikation beschäftigt sich nach [Probst et al.] S. 105 f. mit der Erlangung von Transparenz über vorhandenes Wissen. Dafür ist es wichtig, das entsprechende Wissen systematisch zu identifizieren und einen Überblick über interne und externe Daten, Informationen und Fähigkeiten zu erlangen. Die Transparenz spielt insbesondere auf der informellen Ebene eine gewichtige Rolle. Erfahrungen, Vorwissen und Spezialwissen gilt es kollektiv sichtbar zu machen, ebenso wie Protokolle, Kundeberichte, Projektberichte etc., um Mehrfacharbeit möglichst gering zu halten. Auf der Transparenzebene ist sowohl die interne, als auch die externe Transparenz als Ziel der Wissensidentifikation zu nennen. Es gilt dabei nicht nur, das interne Wissen zu strukturieren, sondern auch externe Wissensbestände (die bspw. durch Kooperation entstehen) sichtbar zu machen. Zur Identifikation von Wissen bieten sich unterschiedliche Hilfsmittel an, wie beispielsweise:

- **Gelbe Seiten:** die gelben Seiten sind Branchenbücher des Wissens. Sie beinhalten Qualifikationen, Erfahrungen, Kompetenzen, Kenntnisse und Fähigkeiten sowohl von internen als auch externen Mitarbeitern.
- **Knowledge Maps:** helfen dem Unternehmen einen Überblick über die Wissensbestände zu geben und Lücken ausfindig zu machen. In Wissenslandkarten werden beispielsweise auch Verbindungen zu Kunden dargestellt, was eine Analyse der Kundenbetreuung ermöglicht.

Ein hohes Maß an Wissenstransparenz garantiert einen besseren Zugriff auf interne und externe Wissensbestände, ebenso wie eine effektivere Nutzung der vorhandenen Ressourcen. Darüber hinaus spielt laut [Zürich] die Identifikation eigener Lücken und Missstände als Voraussetzung für eine Effizienzsteigerung eine gewichtige Rolle.

### 2.1.2.3 Wissenserwerb

Beim Wissenserwerb geht es um den Einkauf von Wissen in das Unternehmen. Bedingt durch Wissensexplosion und Wissensfragmentierung sind Unternehmen oftmals nicht in der Lage, das für sie benötigte Wissen aus eigener Kraft zu entwickeln. Nach [Probst et al.] gibt es dabei vier unterschiedliche Arten des Wissenserwerbs:

- Den Erwerb von Wissen anderer Wissensträger (Hochschulen, Forschungseinrichtungen etc.),
- den Erwerb von Wissen anderer Firmen,
- den Erwerb von Wissensprodukten (Software, Patente etc.) und
- den Erwerb von Stakeholderwissen (bspw. Kundenwissen).

Der Wissenserwerb ist in der Praxis mit mehreren Problemen verbunden. So weisen [Rohmhardt et al.] insbesondere auf die Wichtigkeit der Integrations- und Anschlussfähigkeit des angekauften Wissens hin. [Zürich] spricht gar von einem „not-invented-here-Syndrom“ in Bezug auf Abwehrreaktionen gegenüber extern erworbenem Know-How. Daher ist es wichtig, dass externe Quellen derart genutzt werden können, dass ein Netzwerk entsteht, mit dessen Hilfe neues Wissen akquiriert und direkt in das Unternehmen integriert und genutzt werden kann, ohne dass ein merkbarer Unterschied zwischen intern und extern erworbenem Wissen sichtbar wird.

Ferner muss dem einzelnen Mitarbeiter vermittelt werden, dass der Kontakt zu externen Quellen keine Zeitverschwendung ist. In diesem Kontext spielt auch die Vermittlung von Wissen darüber, wie man externes Wissen ausfindig macht und nutzt, eine Rolle<sup>5</sup>. Vor allem die Kooperation mit Partnern kann zu Synergieeffekten führen, die weit über die Produktionsebene hinaus gehen.

---

<sup>5</sup> Diese Fähigkeit wird in Quellen wie [Knorz2002] als „Informationskompetenz“ bezeichnet

### 2.1.2.4 Wissensentwicklung

Bei der Wissensentwicklung steht die Produktion von Fähigkeiten, neuen Produkten und Ideen im Vordergrund. Dabei spielt die Kreativität der Mitarbeiter und deren Fähigkeit, Probleme zu lösen, eine gewichtige Rolle und muss gefördert werden. Eine Beispielanwendung wäre das betriebliche Vorschlagswesen als Prozess der kollektiven Wissensentwicklung. Auch die sog. „job-rotation“ ist ein Ansatzpunkt, der den Mitarbeitern die Möglichkeit gibt, die gesamte Wertschöpfungskette genauer kennen zu lernen.

Eine wichtige Bedeutung kommt in der praktischen Anwendung von Wissensmanagement der Integration in die tägliche Arbeit zu. Dabei ist es essentiell, die Integration der verbessernden Maßnahmen in die alltägliche Arbeit zu gewährleisten, so dass kein Mehraufwand empfunden wird. Auch hier ist die job-rotation ein guter Ansatz, da Wissensakquise durch „Alltagsarbeit“ stattfindet. Ähnliches gilt für Gruppenarbeit und Projektaktivitäten.

Wissensentwicklung umfasst alle Managementanstrengungen, mit denen die Organisation sich bewusst um die Produktion bisher intern noch nicht bestehender oder gar um die Kreierung intern und extern noch nicht existierender Fähigkeiten bemüht. Wissensentwicklung kann dabei auf der individuellen und auf der kollektiven Ebene konzeptionalisiert werden.

Prozesse der individuellen Wissensentwicklung beruhen vornehmlich auf Kreativität und systematischer Problemlösungsfähigkeit. Da nach [Probst et al.] S.185 ff. Kreativität eher als einmaliger Schöpfungsakt betrachtet werden muss, folgt die Lösung von Problemen eher einem Prozess, der durch mehrere Phasen beschrieben werden kann. Kreativität könnte als chaotische Komponente und Problemlösungskompetenz als systematische Komponente des Wissensentwicklungsprozesses bezeichnet werden. Beide Komponenten müssen durch Steuerungsmaßnahmen unterstützt werden, welche das Individuum in seiner Wissensproduktion fördern. Zur Unterstützung können teilweise althergebrachte Instrumente (wie z.B. das Vorschlagswesen) revitalisiert und neu interpretiert werden.

Kollektive Prozesse der Wissensentwicklung hingegen folgen häufig einer anderen Logik als individuelle Prozesse: Nimmt man das Team als Keimzelle kollektiven Lernens in der Unternehmung, so muss auf die Schaffung komplementärer Fähigkeiten in der Gruppe und die Definition sinnvoller und realistischer Gruppenziele geachtet werden. Nur in einer Atmosphäre von Offenheit und Vertrauen, welche durch eine hinreichende Kommunikati-

onsintensität unterstützt und erzeugt werden muss, sind kollektive Prozesse der Wissensentwicklung individuellen Bemühungen überlegen. Durch die Einrichtung interner Think Tanks und den Aufbau innerbetrieblicher Kompetenzzentren oder Produktkliniken können diese Prozesse unterstützt werden.

### **2.1.2.5 Wissensverteilung**

Nachdem das Wissen erfolgreich identifiziert und angereichert wurde, muss es nun dorthin verteilt werden, wo es benötigt wird. Oftmals erfolgt dieser Informationsaustausch auf informeller Ebene. Dies ist durch den Geschwindigkeits- und Aufwandsvorteil im Vergleich zu technologischen Ansätzen wie Intranet, e-Mail etc. zu begründen. Aufgabe des Wissensmanagements wäre hierbei mitunter, den Informationsaustausch zu verbreitern. Denn auch in kleineren Unternehmen ist der informelle, direkte Informationsaustausch oft beschränkt. Netzwerke, Diskussionsboards und Expertentools können hier zum Vorteil gereichen, ermöglichen neue Dimensionen der Wissensansammlung und Weitergabe und öffnen Quellen, die bislang nicht verfügbar waren. Ein Nachteil bisheriger IT Lösungen für das Wissensmanagement ist die Dezentralität, die den meisten von ihnen zugrunde liegt. So ist schon der Einsatz von PCs häufig dezentral, auf der Basis von Abteilungen. Dies führt zu einer Art „Informationsparadoxon“: Je mehr Information durch verbesserte IT-Möglichkeiten verfügbar ist, desto schwieriger wird es, die relevante zu finden. Diese Situation verlangt nach einer Strukturierung und Organisation von in Unternehmen vorhandenen Informationen.

Oftmals werden in Unternehmen vorhandene IT-Systeme dabei nicht richtig genutzt. Häufig verfügen noch nicht einmal die Führungskräfte über genügend Know-How, um das vorhandene Kapital richtig einzuschätzen und wissensorientiert zu nutzen. Dies impliziert, dass eine umfassende Analyse vorhandener Systeme oftmals die Anschaffung kostspieliger Systeme vermeiden kann.

Ein sehr hohes Gewicht bei der Einführung von Wissensmanagementtechnologien kommt ferner der Einrichtung personalisierter Dienste zu. Diese Systeme sollten vor allem den Bedarf und die Gewohnheiten des Nutzers berücksichtigen und ihn (mit möglichst hohem Automatisierungsgrad) mit den Informationen versorgen, die er benötigt.

Außerdem ist die Integration externer Teilnehmer in interne Systeme nicht zu vernachlässigen. Ein Aufbau eines Extranets für Partner ist ebenso denkbar, wie die Aufbereitung von Internet- und Newsfeed-Quellen durch eine Art Wissensredaktion.

Ohne eine effizientes Informationsmanagement (Bereitstellung, Verteilung und Speicherung von Informationen) ist Wissensmanagement nicht denkbar. Laut [KPMG2001] sollte ein professionelles Informationsmanagement folgende Punkte umfassen:

- **Füllen von Informationen:** Es sollte geregelt sein, wer was in das System stellen kann. Eine zentrale Stelle zur Überprüfung und Einordnung der Informationen wäre vorteilhaft.
- **Filtern:** Durch Selektion muss die (nicht zuletzt durch das Internet hervorgerufene) Datenflut vermindert werden und die Dokumente gespeichert werden, die für den Geschäftsablauf wichtig sind. Diese Informationen sollten darüber hinaus einen Lebenszyklus durchlaufen und abhängig vom Typ u.U. nach einer gewissen Zeitspanne wieder gelöscht werden.
- **Verteilen:** Nicht ausreichend ist eine Informationsversorgung nach dem Pull-System. Vielmehr ist es vonnöten, dass der Mitarbeiter in bestimmten Abständen die für ihn wichtigen Informationen automatisch erhält. Wichtig hierbei ist das Berücksichtigen der persönlichen Präferenzen bezüglich der Intervalle. Informationsmanagement hat somit die Aufgabe, für unterschiedliche Interessen-cluster Informationsprofile zusammenzustellen.
- **Vernetzen:** Ansonsten getrennt existierende Informationen müssen vernetzt werden. Bspw.: Marketinginformationen mit Kunden- und Produktinformationen.
- **Monitoring/Feedback:** Die Weiterentwicklung von Systemen aufgrund von Nutzerdaten, Zugriffshäufigkeiten, Hitlisten etc.
- **Sichern:** Je mehr sich ein Wissensmanagement zum Unternehmensgedächtnis entwickelt, desto wichtiger ist eine adäquate Absicherung der Daten nach außen.

Hinsichtlich der Verteilung von Informationen gilt es ebenfalls, zu keinem Zeitpunkt die sozialen und kulturellen Rahmenbedingungen zu vernachlässigen. Auch hierbei spielt IT eine wichtige Rolle, wird jedoch als rein operative Anwendung nicht ausreichen. Deshalb ist auch hier ein ganzheitlicher Ansatz vonnöten.

### 2.1.2.6 Wissensnutzung

Unternehmen müssen die Nutzung von geschäftsrelevantem Wissen sicherstellen. [Romhardt et. al] sehen die Wissensnutzung als „Ziel und Zweck des Wissensmanagements“. Um den Mitarbeiter zur Nutzung des angehäuften Wissens zu motivieren, dürfen Nutzersicht und Nutzerbedürfnisse zu keinem Zeitpunkt aus den Augen verloren werden. Eine ebenfalls gewichtige Rolle bei der Motivation spielt ein einfacher Zugang zu den Wissensquellen.

Bei IT-Systemen sind diese Aspekte eher einfach zu lösen, vor allem bei der Entwicklung einer Wissenskultur jedoch gestaltet sich dies schwerer. Vor allem Führungskräfte müs-

sen den Weg in Richtung Informationsoffenheit betreten. Dieser führt oftmals weg von einer hierarchischen Struktur, hin zu einer kollegialen, gleichberechtigten Arbeitsweise. Das Eingestehen von Wissenslücken gehört bei diesem Prozess ebenso dazu, wie die unbedingte Weitergabe eigenen Wissens und der positiv-konstruktive Umgang mit Fehlern. Dabei geht es in erster Linie darum, den Mitarbeiter von der Angst zu befreien, ihn seines Wissens berauben und nach „Abgabe“ austauschbar machen zu wollen (in [Schneider] wird dieser Faktor als „FUZ-Faktor“ bezeichnet, stehend für Furcht, Unsicherheit, Zweifel). Wichtig bei diesem Aspekt ist die Einführung von Anreizsystemen, welche die Mitarbeiter von einer Art Wissensegoismus weglösen können. Monetäre Anreizsysteme haben sich dabei in der Praxis nur bedingt bewährt. Immaterielle Anreize, wie eine Verbesserung des Arbeitsumfeldes zeigen sich da weitaus erfolgsversprechender. Merkt der Mitarbeiter, dass ihm die Arbeit durch Wissensmanagement spürbar erleichtert wird, wird er bereit sein, sich darauf einzulassen.

Auch sogenannte, in den neunziger Jahre von Xerox erstmals eingeführte, Communities of Practice bieten eine Möglichkeit, Wissen anzureichern und zu nutzen. Dabei bilden sich (abseits der Organisationsstruktur) Gruppen, die sich selbständig mit einem Themenkomplex auseinandersetzen. Aufgabe der IT ist es hierbei, Rahmenbedingungen, vor allem auf dem kommunikativen Bereich zu setzen und eine dauerhafte Speicherung der dort erhaltenen Daten, Informationen und schließlich des daraus gebildeten Wissens zu garantieren (siehe 2.1.2.7).

Stellt die Unternehmensführung nicht sicher, dass neues Wissen genutzt wird, bewegt sich ein neu eingeführtes Wissenssystem schnell in die in Abbildung 5 skizzierte „Todesspirale“ [Romhardt et al.].

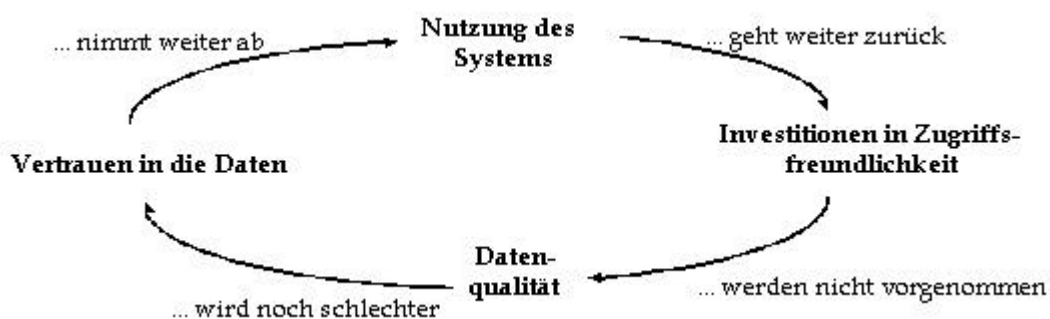


Abb. 5: Die Todesspirale von Wissenssystemen

### 2.1.2.7 Wissensbewahrung

Einmal erworbene Fähigkeiten und vorhandenes Wissen sollten dauerhaft bewahrt werden. Vor allem Kündigungen, Pensionierungen, aber auch Alltagssituationen wie Urlaub oder Krankheit sollten soweit abgefangen werden, dass es keine Unterbrechung der Arbeit aufgrund solcher Vorfälle gibt. Eine detaillierte Übersicht über das Vergessen von unternehmerischem Wissen gibt Tabelle 1:

<i>Modus</i> \ <i>Form</i>		<i>individuell</i>	<i>kollektiv</i>	<i>elektronisch</i>
		<i>Gedächtnisinhalt wird gelöscht</i>	<ul style="list-style-type: none"> <li>■ Kündigung</li> <li>■ Pensionierung</li> <li>■ Tod</li> </ul>	<ul style="list-style-type: none"> <li>■ Auflösung eingespielter Teams</li> <li>■ Reengineering</li> <li>■ Outsourcing von Funktionsbereichen</li> </ul>
<i>Zugriff nicht möglich</i>	<i>befristet</i>	<ul style="list-style-type: none"> <li>■ Überlastungen</li> <li>■ Versetzungen</li> <li>■ Krankheit/Urlaub</li> <li>■ Mangelndes Training</li> <li>■ Dienst nach Vorschrift</li> </ul>	<ul style="list-style-type: none"> <li>■ Tabuisierung von alten Routinen</li> <li>■ Kollektive Sabotage</li> </ul>	<ul style="list-style-type: none"> <li>■ Reversible Datenverluste</li> <li>■ Überlastung (befristet)</li> <li>■ Schnittstellenproblem</li> </ul>
	<i>auf Dauer</i>	<ul style="list-style-type: none"> <li>■ Überlastung</li> <li>■ Kein Bewusstsein für Wichtigkeit des eigenen Wissens</li> <li>■ Innere Kündigung</li> </ul>	<ul style="list-style-type: none"> <li>■ Verkauf von Unternehmensteilen</li> <li>■ Abwanderung von Teams</li> </ul>	<ul style="list-style-type: none"> <li>■ Dauerhafte Inkompatibilität von Systemen</li> <li>■ Überlastung (permanent)</li> <li>■ Falsche Kodifizierung</li> </ul>

Tabelle 1: Formen des organisatorischen Vergessens [KPMG2001]

Um dem organisatorischen Vergessen entgegenzuwirken, sind verschiedene Maßnahmen zu treffen. Man unterscheidet bei der Wissensbewahrung drei unterschiedliche Prozesse:

- Zunächst muss das sicherungswürdige Wissen identifiziert werden und dann
- Maßnahmen entwickelt werden, die eine fortlaufende selektive Speicherung ermöglichen und schließlich muss
- das vorhandene Wissen regelmäßig aktualisiert werden.

Ferner sollte durch Verbesserung von sozialen Kriterien, wie bspw. des Arbeitsklimas, eine hohe Bindung der Fachkräfte an die Firma garantiert werden.

Mithilfe von Dokumentation einzelner Prozesse können ferner wichtige Abläufe protokolliert und formalisiert werden. Dadurch ist gewährleistet, dass Wissen an andere Mitarbeiter weitergegeben wird und bei Ausscheiden eines Mitarbeiters nicht verloren geht. Schließlich sollte es Aufgabe des Unternehmens sein, dem Mitarbeiter zu vermitteln,

dass sein Wissen innerhalb der Firma mindestens genauso geschätzt wird, wie außerhalb der Firma.

Allgemein bieten sich folgende Möglichkeiten zur Formalisierung und Bewahrung:

- Einsatz von Wissensmanagern zur Motivation, das Wissen zu teilen
- Interne Beratungen: zur Klärung von Wissenslücken und ungelösten Fragen
- Patenschaftsmodelle: Bestimmte Wissensgebiete bekommen feste Ansprechpartner. Diese stehen außerhalb einer Hierarchie und dienen als Anlaufstelle für Fragen.
- Dokumentation von wichtigen Prozessen: Durch eine Prozessdokumentation lassen sich Strukturen entwickeln, mit deren Hilfe Wissenssicherung betrieben werden kann

### 2.1.2.8 Wissensbewertung

Wissenscontrolling ist ein bisher ungelöstes Problem des Wissensmanagements. Wissen kann nur über den Preis der Verdinglichung quantifizierbar werden, also der Entnahme aus zeitlichen, situativen und persönlichen Kontexten. Es kann deshalb, ebenso wie es nur kontextuell gesteuert werden kann, nur mittelbar und unscharf erfasst werden; bspw. anhand von vermittelten Messgrößen wie Prozesszeiten, Ausbildungserfolg oder Ähnlichem. Der Anspruch, Wissen exakt messen zu können, gaukelt dort Objektivität vor, wo nur Unschärfe sein kann [Roehl et al.].

Obwohl Wissen als gewichtiger Faktor zur Wertschöpfung eines Unternehmens in nicht unwesentlichen Ausmaßen auftritt, wird es in den Berechnungen der Managementkontrollsysteme und ökonomischen Modelle nicht berücksichtigt. Das Controlling von Maßnahmen des Wissensmanagements muss daher neben den finanziellen auch Faktoren der Wissensentwicklung und –nutzung mit einbeziehen. Indikatoren zur Wissensbewertung können dabei personalbezogen, lernbezogen oder finanzbezogen sein.

[Romhardt et al.] schlagen für die Wissensbewertung die in Tabelle 2 dargestellten Maßnahmen vor.



	Wissensziele	Bewertungsmethoden
N O R M A T I V	<ul style="list-style-type: none"> <li>→ schaffen Voraussetzungen für wissensorientierte Ziele im strategischen und operativen Bereich</li> <li>→ zielen auf „wissensbewusste“ Unternehmenskultur</li> <li>→ erfordern Commitment des Top-Managements</li> </ul>	<ul style="list-style-type: none"> <li>◆ Kulturanalysen</li> <li>◆ Beobachtung des Top-Management Verhaltens (z.B: Agenda-Analysen)</li> <li>◆ Glaubwürdigkeitsanalysen (Lücke zwischen Ideal und Ist)</li> </ul>
S T R A T E G	<ul style="list-style-type: none"> <li>→ inhaltliche Bestimmung organisationalen „Kernwissens“</li> <li>→ Definition des angestrebten Kompetenzportfolio</li> <li>→ legen Hauptrichtung des Kompetenzaufbaus fest</li> </ul>	<ul style="list-style-type: none"> <li>◆ Wissensbilanz erstellen</li> <li>◆ Analyse des Kompetenzportfolios</li> <li>◆ Controlling der bedeutendsten „Wissensprojekte“</li> </ul>
O P E R A T I V	<ul style="list-style-type: none"> <li>→ übersetzen normative und strategische Wissensziele ins Konkrete</li> <li>→ sichern die Angemessenheit der Interventionen in Bezug auf die jeweilige Interventionsebene</li> </ul>	<ul style="list-style-type: none"> <li>◆ Ausbildungscontrolling mit klaren Lerntransferzielen</li> <li>◆ Messung von Systemnutzung (→ Intranet)</li> <li>◆ Erstellung individueller Fähigkeitsprofile</li> </ul>

Tabelle 2: Ansätze zur Wissensbewertung

## 2.2 Anforderungen an ein Wissensmanagementsystem und Schwachpunkte bestehender Systeme

In Kapitel 2.1 wurde auf die Bedeutung von Wissensmanagement für Unternehmen und Organisationen und dessen Gründe eingegangen. Vor allem eine zunehmende Produktkomplexität, die Zunahme virtueller Unternehmen und die Notwendigkeit einer stärkeren Kundenorientierung erfordern den systematischen Umgang mit Wissen. [Studer et al.] sehen die Informationstechnik hierbei als „Enabler“ für Wissensmanagementlösungen. Informationstechnische Wissensmanagementsysteme bezwecken dabei eine bedarfsgerechte Bereitstellung von Wissen und sollen den Wissensaustausch unterstützen. Diesen Zweck zu erfüllen benötigt ein solches System die folgenden Funktionalitäten:

- Eine (unabhängige) Infrastruktur zur Kommunikation
- Komponenten, die organisationale Anforderungen erfüllen
- Inhaltsorientierte Methoden für den Austausch von Wissen
- Effizienten Zugriff auf Wissen

Konventionelle, dokumentenbasierte Wissensmanagementsysteme stellen zumeist eine Infrastruktur (Intranet) und Rahmenbedingungen für organisationale Funktionalitäten zur Verfügung (bspw. Content Management Systeme). Technische Systeme für die Unterstützung inhaltsorientierter Funktionalitäten waren bisher nicht sehr verbreitet.

Vor allem ein unterschiedlicher Kontext kommunizierender Personen bringt häufig Missverständnisse und unterschiedliche Interpretationsansätze einzelner Aussagen mit sich. Ein verbreiteter Ansatz zur Verdeutlichung dieser Zusammenhänge ist das sogenannte „Semiotische Dreieck“ (oder auch Bedeutungs-dreieck) nach [Ogden et al.], welches den Zusammenhang zwischen Wörtern (=Zeichen), Gedanken und realer Welt wiedergibt. Dieses semiotische Dreieck ist vereinfacht wie folgt darzustellen:

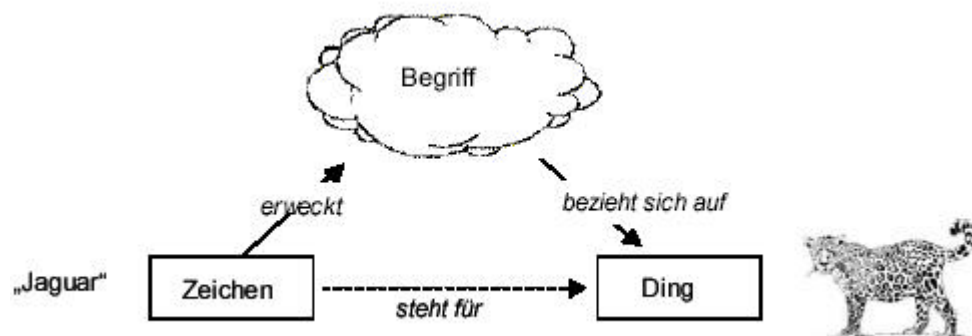


Abb.6: Das Semiotische Dreieck (aus [Maedche])

Das Semiotische Dreieck besagt, dass es keinen direkten Bezug zwischen Zeichen (dem Wort) und dem Referenzobjekt (im Abbildung 6: das Tier Jaguar) gibt. Dieser Bezug wird indirekt, „durch einen Gedanken“ eines Individuums hergestellt. Das Ergebnis dieser gedanklichen Leistung wird gemeinhin als „Begriff“ bezeichnet [Straub]. Zu beachten ist hier ebenfalls, dass die indirekte Beziehung zwischen Wort und Referenz immer eine subjektive ist, d.h. vom Erfassungskontext des Akteurs stark abhängig. Verdeutlichen lässt sich diese Problematik anhand von Begriffen, die unterschiedliche Bedeutungen besitzen.

So lässt das Wort „Birne“ den interpretierenden Obsthändler wohl auf das Obst schließen, während ein Elektroinstallateur eher das Leuchtmittel als Begrifflichkeit in Betracht ziehen würde. Hierbei wird deutlich, dass der Kontext des Denkenden eine gewichtige Rolle spielt. Das Fehlen eines gemeinsamen Verständnisses führt zu Missverständnissen zweier Kommunikationspartner. Dabei spielt es keine Rolle, ob es sich um Mensch/Mensch – Kommunikation oder Mensch/Maschine – Kommunikation handelt.

Ziel des Wissensmanagements muss es folglich sein, zu einem gemeinsamen Verständnis zu gelangen und konzeptuelle und terminologische Unklarheiten zu beseitigen. [Uschold et al.] sehen die primäre Aufgabe der IT daher darin, ein „geteiltes Verständnis“ (im Sinne von „gemeinsames Verständnis“) zu erreichen. Für das Bedeutungs-dreieck hätte dies die Auswirkung, dass die Begriffsbildung in einem fest vorgegebenen Kontext vonstatten geht. Die Folge davon ist eine erhöhte Interoperabilität zwischen Mensch und Maschine und eine höhere Mehrfachnutzung bestehender Systeme, Daten, Informationen und schließlich Wissen.

In diesem Zusammenhang wird auch oftmals ein weiteres Defizit bestehender, dokumentenbasierter Systeme genannt. So ist immer noch ein nicht unerheblicher intellektueller Aufwand vonnöten, die relevanten Informationen zu extrahieren, selbst wenn der Nutzer, nach einer Suche, das passende Dokument gefunden hat. Dies hängt vor allem mit der fehlenden Semantik in den Dokumenten selbst zusammen.

Auch die häufig ungenügende Implementierung von Methoden zur Erschließung neuen Wissens in Wissensmanagementsystemen ist als Mangel zu nennen. Bestehende Systeme sind in der Nutzung der Wissensressourcen vornehmlich starr, d.h. explizit vorhandenes Wissen wird weitestgehend durch das System erfasst, implizite, nicht direkt abgebildete (aber dennoch vorhandene) Zusammenhänge aber nicht als relevant erfasst.

Lösungen für all diese Problematiken bieten Ontologien.

## **2.3      *Ontologien***

Das folgende Kapitel gibt einen Einstieg in das Konzept der Ontologien. Zunächst wird in 2.3.1 kurz definiert, was eine Ontologie im computerlinguistischen Sinne ist, auf welche Komponenten sie zurückgreift und nach welchen Kriterien man Ontologien unterscheiden kann. Im zweiten Teil dieses Kapitels werden Ontologien aus dem Blickwinkel des Wissensmanagements betrachtet. Hierbei wird zunächst der Zusammenhang zu Kapitel 2.1 dieser Arbeit und den in Kapitel 2.2 herausgearbeiteten Missstände bestehender Wissensmanagementsysteme hergestellt. Ferner wird eine Methodologie zur Entwicklung eines ontologiebasierten Wissensmanagementsystems herausgearbeitet, die, in angepasster Form, später in dieser Arbeit noch zur Anwendung kommen wird.

### 2.3.1 Definition

Ursprünglich stammt die Begrifflichkeit der Ontologien aus der Philosophie. Dort bezeichnet „Ontologie“ den Zweig, der sich mit der Natur und der Organisation der Realität beschäftigt, also den Bereich der Seinsforschung.

Im computerwissenschaftlichen Zusammenhang sollen Ontologien das Wissen einer bestimmten Domäne generisch darstellen und ein allgemein gültiges Verständnis dieser Domäne zur Verfügung stellen, welches dann universal eingesetzt werden kann. Ontologien stellen so das Vokabular eines bestimmten Bereichs zu Verfügung und definieren (auf unterschiedlichen formalen Levels) die Bedeutung von Begriffen (=Terms) und deren Beziehungen untereinander.

Im Laufe der Zeit haben sich mehrere Definitionen gebildet, deren Synthese sich in der folgenden wiederfindet:

„Eine Ontologie ist eine formale, explizite Spezifikation einer gemeinsamen Konzeptualisierung“ [Gruber]

Diese Definition erklärt sich wie folgt: „Formal“ bedeutet, dass Ontologien maschinenlesbar sein sollen, während „explizit“ meint, „dass Art und Bedingung eines Begriffes explizit angegeben werden müssen“ [Studer et al.]. Konzeptualisierung bezeichnet die Erstellung eines abstrakten Modells, wobei das „gemeinsam“ signalisiert, dass es sich bei diesem Modell nicht um ein singuläres, sondern vielmehr um ein von einer gewissen Gruppe anerkanntes Modell handelt. Diese Gruppe kann sowohl aus Menschen, als auch Maschinen bestehen, die auf Grundlage dieser Modellierung kommunizieren können. Anhand des oben angesprochenen „Semiotischen Dreiecks“ bedeutet dies, dass die Begriffsbildung als Relation zwischen Zeichen und Referenzobjekt in einem fest vorgeschriebenen Kontext abläuft.

Dabei greifen Ontologien auf die folgenden Komponenten zurück <sup>6</sup> (vgl. [Corcho et al.] und [Brachmann]):

**Klassen:** Klassen sind normalerweise in Taxonomien<sup>7</sup> geordnet. Teilweise wird der Begriff der Ontologie verwässert, indem diese Taxonomien bereits als Ontologie bezeichnet werden. Diese Ontologien nennt man auch „lightweight Ontologies“, im Gegensatz dazu

---

<sup>6</sup> an dieser Stelle sei angemerkt, dass sich die Bezeichnungen der Komponenten quellenabhängig unterscheiden können, die Konzepte dahinter sind jedoch stets dieselben.

<sup>7</sup> Taxonomie: [...]Aufbau des Sprachsystems auf dem Wege der Segmentierung u. der Klassifikation sprachlicher Einheiten [...] [Duden]

stehen die „heavyweight Ontologies“. Klassen (oder auch **Konzepte**) sind definiert als etwas, was beschrieben werden kann, also auch die Beschreibung einer Aufgabe, einer Funktion, einer Tätigkeit, Strategie usw.

**Instanzen:** repräsentieren Individuen

**Attribute:** Attribute beschreiben Eigenschaften von Konzepten

**Rollen:** Rollen beschreiben Erweiterungen einzelner Klassen für bestimmte Situationen

**Beziehungen:** stehen für Interaktionen zwischen den einzelnen Konzepten einer Domäne. Beziehungen werden formal als Teilmenge eines Produktes aus  $n$  Teilen definiert:  $R: C_1 * C_2 * \dots * C_n$ . Beispiele binärer Beziehungen sind die part-of- (Meronomie) und is-a- (Hyponymie) Beziehungen

**Axiome:** bilden Sätze, die immer richtig sind. Axiome können für unterschiedliche Zwecke in Ontologien integriert werden: Um die Bedeutung einzelner Ontologiekomponenten zu definieren, komplexe Beschränkungen der Attributswerte zu realisieren, Beziehungsargumente usw. zu definieren, die Richtigkeit der in der Ontologie spezifizierten Information zu beweisen oder neue Information abzuleiten.

Eine Ontologie enthält also die Begriffe eines Bereichs, die taxonomisch geordnet die relevanten Begriffe dieses Ausschnittes beschreiben. Mittels Erweiterung durch Attribute, Rollen und Beziehungen entsteht ein Modell, welches für einzelne Worte immer eindeutige Verbindungen zu den Referenzobjekten der realen Welt gewährleistet. Wird dieses Modell verinnerlicht, so verschiebt sich die syntaktische Kommunikation hin zu einer semantischen, da die Bedeutung eindeutig durch das Modell festgelegt wird und sich die Beteiligten auf dasselbe Referenzobjekt beziehen.

Ontologien unterscheiden sich dabei einerseits durch ihren Allgemeingrad [Mueller]. Top-Level-Ontologien, die allgemeinste Form, beschreiben hierbei äußerst allgemeingültige Konzepte wie Raum, Zeit etc unabhängig von einer bestimmten Domäne. Domain und Task Ontologien hingegen beschreiben das Vokabular einer allgemeinen Domäne (Medizin, Elektronik etc.) bzw. allgemeinen Aufgabe (Controlling, Verkauf etc.), indem sie die Begriffe, die in der Top-Level-Ontologie eingeführt worden sind, spezialisieren. Anwendungsontologien beschreiben zumeist Konzepte, die sowohl von einer speziellen Domäne, als auch von einer bestimmten Aufgabe abhängen.

Im Beispiel der Ontologie für das Portal der Fachhochschule Darmstadt handelt es sich um eine Domain-Ontologie zur Beschreibung der Hochschulwelt. Allerdings lassen sich kleinere Teile der Ontologie gleichfalls als Task-Ontologien identifizieren, ebenso wie die allgemeineren Konzepte als Top-Level-Ontologie bezeichnet werden können.

Ferner unterscheidet man sogenannte konzeptionelle Ontologien von den linguistischen. Laut [Fensel] stellt sich der Unterschied zwischen beiden Ansätzen wie folgt dar: Konzeptionelle Ontologien benötigen demzufolge eine bestimmte Strukturierung, um bessere Kontrolle, bessere Performanz oder eine kompaktere, schlüssigere Struktur zu erreichen. Daher ist es nötig, künstliche Level für Konzepte einzuführen, die in einer Sprache nicht lexikalisch erfasst sind. Ebenso ist es möglich, dass Level, die lexikalisch existieren, vernachlässigt werden können, weil sie für den Zweck der Ontologie nicht relevant sind. Linguistische Ontologien hingegen geben exakt die Verbindungen zwischen lexikalischen Wörtern und Ausdrücken in einer Sprache wieder. Um das zu gewährleisten, fassen sie nützliche Informationen über die Ausdrücke und die Menge an Worten und Ausdrücken in einer Sprache zusammen (vgl. auch [Eurowordnet]).

Zuletzt lassen sich Ontologien durch ihren Formalisierungsgrad unterscheiden. Mit zunehmendem Formalismus wird das Datenmodell ausdrucksstärker. Die unterschiedlichen Ontologiemodelle sind in Abbildung 7 zu sehen. Das Ontologiemodell des Portals der FHD ist im Bereich „Frame Systeme“ anzusiedeln.

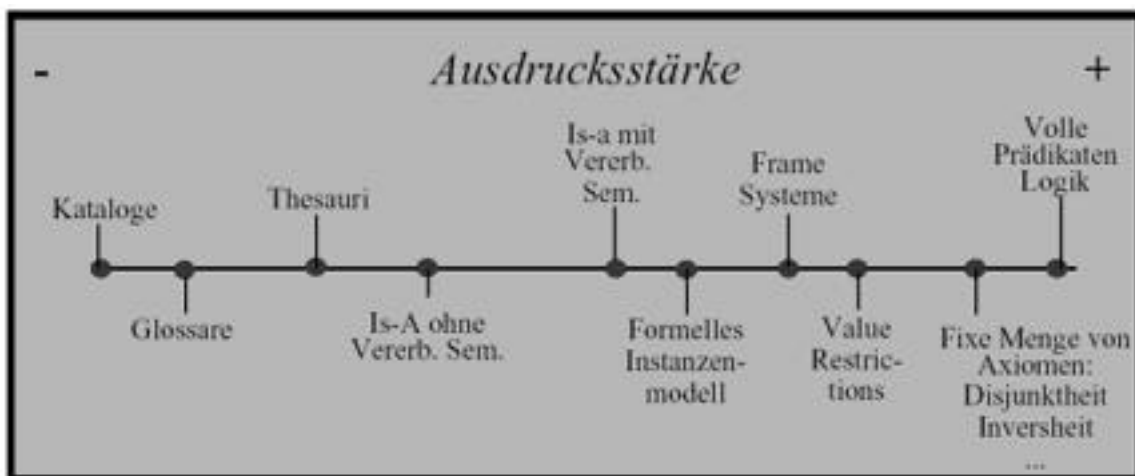


Abb.7: Ontologiemodelle (aus [Volz])

Ontologien bieten die Möglichkeit der Implementierung sogenannter intelligenter Suchen über ein Realitätsmodell. Charakteristisch dabei ist die Berücksichtigung der semantischen Beziehungen zwischen einzelnen Wissensobjekten (=Konzepten). Während die Volltextsuche lediglich Wortnähe im Zieltext bewerten kann und somit auf statistischer Häufigkeitsanalyse basiert (d.h., wenn zwei Wortformen nah beieinander stehen, stehen sie „irgendwie“ inhaltlich beieinander), wird bei der semantischen Suche die Bedeutungsnahe einzelner Worte berücksichtigt. So liefert die Volltextsuche immer nur eine Relation zwischen zwei Textworten zurück, welche spezifiziert, dass ein Wort in räumlicher Nähe

zu einem Anderen steht. Die semantische Suche hingegen kann die Verbindung zweier Wörter benennen (bspw.: „Wort A ist Oberbegriff zu Wort B“).

Ein weiteres Schlüsselement beim Einsatz von Ontologien ist die Möglichkeit, neues Wissen zu inferieren. Durch Ableitung bestehender Regeln wird dabei implizites Wissen transparent gemacht und neues Wissen abgeleitet. Unter Inferenz versteht man dabei „Vorgänge des Schlussfolgerns auf eine explizit nicht gegebene Information aus bestimmten gegebenen Informationen innerhalb von Texten. Dazu gehören u. a. Schlüsse nach den logischen Regeln, wie z. B. >Alle Menschen haben zwei Beine. Max ist ein Mensch.<, woraus geschlussfolgert wird >Max hat zwei Beine.<, sowie die Folgerung auf weitere Informationen, wie z. B. aus der Frage >Wo arbeitet der Bruder von Anton?<, auf >Anton hat einen Bruder“ [Brockhaus].

Das folgende Beispiel aus [Oppermann et al.] verdeutlicht die Möglichkeit der Inferenzen im Hinblick auf ein Wissensmanagementsystem:

Es seien die beiden folgenden logischen Aussagen (Relationen) gegeben:

besucht\_veranstaltung (Mitarbeiter, Veranstaltung)

hat\_Thema (Veranstaltung, Thema)

Die über diese Relationen allgemein definierbaren Ableitungen sind die Folgenden: Wenn Mitarbeiter X eine Veranstaltung Y besucht und diese Veranstaltung das Thema Z hat, dann hat X Kenntnisse über das Thema Z. Eine Anfrage an ein System mit integrierter Inferenz gibt in diesem Fall bei der Frage „Wer hat Kenntnisse zu Thema Z“ als Antwort „Mitarbeiter X“ zurück, obwohl diese Information in dem System nicht explizit vorhanden ist.

Die Einsatzgebiete von Ontologien sind dabei vielfältig. In [Uschold et al.] werden die Gebiete, die durch den Einsatz von Ontologien verbessert werden können, in die folgenden drei Kategorien geteilt:

a.) Kommunikation:

Um die Kommunikation in organisatorischen Einheiten zu verbessern, bedienen Ontologien verschiedene Aspekte. So helfen Ontologien, ein „genormtes“ Bild von Themenbereichen zu vermitteln und tragen durch feste Begriffsbildung zu einer Verringerung von Missverständnissen bei. Gleichzeitig integrieren sie mehrere unter-

schiedliche Sichten unterschiedlicher Nutzer und Systeme. Ferner helfen Ontologien dabei, ein Beziehungsnetzwerk innerhalb einer Organisation aufzubauen.

b.) Interoperabilität

Viele Unternehmen arbeiten mit unterschiedlichen Teilsystemen, die Daten untereinander austauschen sollen. Ontologien bieten dabei nicht nur ein gemeinsames Vokabular, sondern können auch als Übersetzungstools zwischen einzelnen Anwendungen dienen.

c.) Spezifikation

Ontologien können zur Standardisierung und Spezifikation bestimmter Vorgänge genutzt werden. So gibt es Ontologien, die Spezifikationen für Softwareentwicklung definieren (Bsp: IBM Business System Development Model) oder Rahmenbedingungen für die Entwicklung von Domänenontologien bereitstellen (Bsp.: CommonKADS Conceptual Modelling Language, siehe [Ollson]).

d.) Verlässlichkeit

Ontologien können durch Unterstützung halbautomatischer Konsistenzchecks die Verlässlichkeit von Systemen verbessern

e.) Mehrfachnutzung

Um vorhandene Systeme effektiver zu gestalten, müssen einzelne Module zwischen verschiedenen Systemen hin- und her importiert werden können. Ontologien werden derart konzipiert, dass zumindest Teile (nämlich die domainunabhängigen) wiederverwertbar sind. So ist ein ontologiebasiertes Softwaresystem im besten Fall zu weiten Teilen nicht abhängig von einer bestimmten Anwendung. Ontologien sollten zu diesem Zweck zwei Anforderungen erfüllen: Sie sollten erweiterbar und veränderbar sein und zwar in der Art, dass Änderungen/Erweiterungen den bestehenden Weltausschnitt nicht komplett ungültig machen.

### **2.3.2 *Ontologien für das Wissensmanagement***

Ontologien bieten Ansätze für alle oben angesprochenen Bausteine des Wissensmanagements. Gleichzeitig unterstützen sie im Einsatzgebiet Wissensmanagement Lösungen für alle unter 2.3.1 aufgeführten Kategorien.

Besonders das unter 2.2 angesprochene Schlüsseldefizit existierender Systeme im Hinblick auf die mangelnde Semantik wird durch die Bereitstellung einer eindeutigen Termini-



nologie gelöst. Bei der Einführung eines ontologiebasierten Wissensmanagementsystems werden nach [Studer et al.] und [Staab et al.], analog zu den in 2.1.2 behandelten Bausteinen des Wissensmanagements die Aktivitäten in zwei verschiedene Prozesse aufgeteilt, aus denen dann eine methodische Vorgehensweise abgeleitet wird. Diese Vorgehensweise, im folgenden näher erläutert, wird, in angepasster Form, als methodischer Rahmen bei der Konzeption und Prototypisierung des ontologiebasierten FHD-Portals im Rahmen dieser Arbeit eingesetzt werden.

- Wissens – Metaprozess: Der Wissens – Metaprozess bezeichnet den Prozess der Einführung und Instandhaltung von Wissensmanagementlösungen und ist korrespondierend zu dem äußeren Kreislauf aus 2.1.1 / 2.1.2
- Wissensprozess: Unter dem Wissensprozess versteht man die Generierung, Anhäufung, Erfassung und Nutzung von Wissen. Dieser Prozess entspricht dem inneren Kreislauf aus 2.1.1 / 2.1.2

### 2.3.2.1 Der Wissens – Metaprozess

Der Wissens – Metaprozess definiert nach [Sure et al.] folgende Entwicklungsphasen bei der Entwicklung eines ontologiebasierten Wissensmanagementsystems:

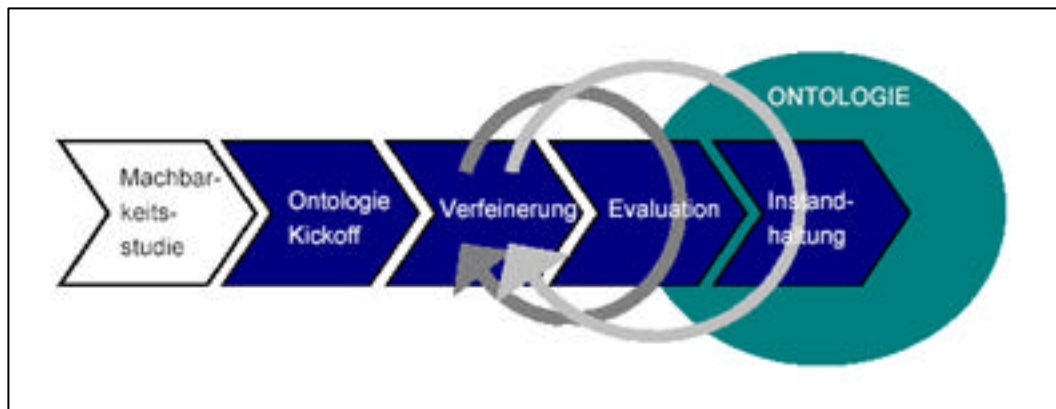


Abb. 8: Der Wissens – Metaprozess

Die einzelnen Phasen werden dabei sequenziell durchlaufen, wobei aus einer bestimmten Phase bei Bedarf zurück in eine vorherige gesprungen wird (bspw. von der Evaluierungsphase in die Verfeinerungsphase).

Die einzelnen Phasen erklären sich wie folgt:

**a.) Machbarkeitsstudie**

Die Machbarkeitsstudie dient der Integration des Wissensmanagementsystems in den organisatorischen Zusammenhang und bildet die Grundlage für die Garantie einer ökonomisch sinnvollen Implementierung des Systems. Dabei gilt es laut [Benjamins et al.] sowohl technologische, als auch soziale und organisationale Risiken abzuschätzen und einzuordnen.

**b.) Ontologie-Kickoff**

In der Ontologie-Kickoff-Phase sollen Anforderungen an die Ontologie definiert werden. Diese Spezifikation sollte nach [Studer et. al] und [Davies et al.] folgende Punkte enthalten: Ziele der Ontologie, Domäne und Anwendung, Wissensressourcen, Nutzer- und Nutzerszenarien, Designleitlinien (zur Unterstützung der Modellierung), Kompetenzfragen (nach [Uschold et al.], zur späteren Überprüfung der Zielorientierung), verwendbare vorhandene Ontologien.

**c.) Verfeinerungsphase**

Kernpunkt der Verfeinerungsphase ist die Entwicklung einer anwendungsbezogenen Ziel-Ontologie. Dazu muss zunächst eine Taxonomie der relevanten Begrifflichkeiten erstellt werden, die dann schrittweise in Zusammenarbeit mit Domänenexperten verfeinert wird und zunächst in epistemologischer Form als Kernontologie und später formalisiert als erste Zielontologie die Zieldomäne abdeckt. [Uschold et al.] schlagen dabei eine relativ informale Herangehensweise vor. Durch Brainstorming sollen die wichtigsten Begriffe herausgefunden werden, die dann durch sinnvolle Gruppierung eine erste Taxonomie bilden. Erste Modellierungsentscheidungen sollten hierbei dokumentiert werden. Der Einsatz von (unter b.) identifizierten) bereits existierenden Ontologien kann diese Phase stark abkürzen und wertvolle Modellierungshinweise geben.

**d.) Bewertungsphase**

In der Bewertungsphase wird die Ontologie auf Nutzbarkeit überprüft. Dazu wird untersucht, ob die Kompetenzfragen aus b.) mithilfe der Ontologie beantwortet werden können und ob die Handhabung verständlich ist. An dieser Stelle beginnt ein wiederholender Prozess, der solange durch Verfeinerungs- und Bewertungsphase schleift, bis der gewünschte Detaillierungsgrad erreicht ist. Dabei sollte nach dem in [Uschold et al.] vorgeschlagenen „Middle-Out“ Ansatz vorgegangen werden. Im Gegensatz zu Top-Down (vom Abstrakten zum Konkreten) und Bottom-Up (vom Konkreten zum Abstrakten, siehe auch [Corcho et al.]) bedeutet dies, eine schrittweise Verfeinerung, angefangen bei den relevantesten Konzepten hin zu den weniger relevanten.

Mit dem endgültigen Abschluss der Bewertungsphase beginnt die Überführung der Ontologie in das Zielsystem.

### 2.3.2.2 Der Wissensprozess

Durch die Einhaltung des Wissens – Metaprozesses wird eine klar definierte Struktur für den Wissensprozess garantiert, die den Nutzer bei jedem Schritt unterstützt. Aktuelle Systeme basieren zumeist entweder auf Dokumenten (Dokumenten-Management-Systeme) oder auf der Identifikation von Wissensträgern auf Basis von Skill-Managementssystem wie etwa „Gelbe Seiten“ oder „Skill-Datenbanken“. Der Wissens – Metaprozess der zur Einführung solcher Lösungen verwendet wird, ist in der Regel recht einfach: Nach der Identifizierung des Wissensbedarfs werden zunächst Geschäftsdokumente und Datenbanken für den Wissensbedarf identifiziert, eine Infrastruktur konstruiert und schließlich Wissensprozesse für die Erstellung, Verteilung und Nutzung von Wissen reorganisiert. Deutlich wird hierbei, dass bei dieser Prozesssicht der Fokus auf die Handhabung von Dokumenten und Datenbanken gerichtet wird, während dem eigentlichen Wissensprozess (der Erstellung, Verteilung und dem Nutzen von Wissen) wenig Beachtung geschenkt wird. Der in [Schnurr et al.] vorgestellte Ansatz vertritt im Gegensatz dazu eine auf Wissens Elemente konzentrierte Sichtweise. Dieser Ansatz zeigt, wie „durch Verwendung von Domänen-Ontologien die einzelnen Wissensprozessschritte, nämlich Generierung, Import, Zugriff und Nutzung von Wissen nahtlos integriert werden können“ [Schnurr et al.]. Ferner wird bei dieser Methode der Fokus auf die Schritte des Wissensprozess hin und die Orientierung weg von den Dokumenten hin zu Wissens Elementen verschoben. Grund hierfür ist vor allem die erhöhte Dokumentenzahl, bedingt durch den Einzug der neuen Medien. Dokumente sind laut [Jurisica et al.] heutzutage nicht mehr zwingendermaßen eindeutig als Einheit zu erkennen und müssen daher inhaltlich strukturiert werden. Der Unterschied der Sichtweisen ist in Tabelle 3 dargestellt.

	Fokus auf Dokumente	Fokus auf Wissens Elemente
1.	Identifiziere den Wissensbedarf	
2.	Identifiziere <i>Geschäftsdokumente und Datenbanken</i> für diesen Wissensbedarf	Identifiziere <i>Wissenselemente</i> für diesen Wissensbedarf
3.	<i>Konstruiere Infrastruktur</i> für das Unternehmensgedächtnis	<i>Organisiere Wissensprozesse</i> um einfache Generierung und Handhabung von Wissens Elementen im Prozess zu erreichen
4.	<i>Reorganisiere Wissensprozesse</i> um Erstellung und Verteilung von Wissen zu erlauben	<i>Konstruiere Infrastruktur</i> für das Unternehmensgedächtnis

Tabelle 3: Der Wissens – Metaprozess (aus [Studer et al.]

Um die praktische Unterstützung bezüglich des Wissensprozesses zu verdeutlichen, wird im Folgenden auf die einzelnen Schritte des Wissensprozesses (orientierend an den „Bausteinen des Wissensmanagements“ aus 2.1.2) eingegangen, die sich in einem Kreislauf mit folgenden Komponenten bewegen:

#### **a.) Wissensgenerierung und Wissensimport**

Wissen muss erzeugt oder konvertiert werden, so dass es der bestehenden Struktur und den vorhandenen Konventionen entspricht. Dabei zeichnet sich die Struktur des Wissens durch einen ungleichmäßigen Formalisierungsgrad aus. So unterscheiden sich Datenbankinhalte in ihrem Formalisierungsgrad erheblich von Textdokumenten.

Laut [Studer et al.] werde allerdings oftmals übersehen, dass Textdokumente nicht selten eine implizite Formalisierung mit sich bringen. So unterliege eine Großzahl an Geschäftsdokumenten gewissen formalen Aspekten, die sich entweder durch gewisse Standards, Vorschriften oder einfache Gewohnheit ergeben haben. Aufgabe für den Teilprozess der Wissensgenerierung sei es daher, Dokumentenschemata für Abläufe zu schaffen und direkt in ein Wissensmanagementsystem zu integrieren. Im Beispiel der Fachhochschule könnte man sich vorstellen, in Bereichen des Prozessmanagements Standardisierungen für Dokumente (bspw. Sitzungsprotokolle) zu entwickeln, welche bspw. XML – codiert eine semantische Strukturierung solcher eigentlich unformalen Dokumente erreichen könnten. In Verbindung mit der Ontologie würden so strukturierte Dokumente durch entsprechende Annotation direkt durch diese definiert werden und nicht auf Dokumentenebene über lediglich eine Begrifflichkeit verfügbar gemacht werden. Ein weiterer Ansatz, der allerdings eine nicht ganz so weitreichende Formalisierung mit sich bringt, ist das Hinzufügen von Metadaten zu unstrukturierten Dokumenten. So ist es denkbar (wiederum im Beispiel der Fachhochschule), Dokumente, wie Forschungsberichte, am Ende mit Metadaten bspw. des Dublin Core [Dublincore] auszuzeichnen. Über eine integrierte Metadatensuchmaschine könnte dann ebenfalls nach der Semantik der Dokumente gesucht werden und in Zusammenarbeit mit der Ontologie verwandte Themenbereiche zu Begriffen auch auf der Dokumentenebene aufgezeigt werden (siehe Kapitel 5.2.1 und 5.2.2).

#### **b.) Wissenserfassung**

Wissen muss inhaltlich erfasst werden, um es geeignet abzulegen und wieder auffinden zu können. Auf ontologischer Ebene bedeutet dies die terminologische Integration in die Ontologie und die sinnvolle Verlinkung mit anderen Inhalten. Vor allem in Freitexten enthaltene und somit für den Rechner unverständliches Wissen muss derart aufbereitet

werden, dass es in das System integriert werden kann und auffindbar wird (siehe Abschnitt 5.2.3).

### **c.) Wissenszugriff**

Der Wissenszugriff auf ein ontologiebasiertes Wissensmanagementsystem geschieht zunächst mit klassischen Retrieval-Methoden. Die Stichwortsuche über ein Fragefeld ist dabei wohl die gängigste Variante. Darüber hinaus bieten ontologische Systeme allerdings Möglichkeiten, die herkömmliche Systeme nicht leisten können. So ist die Möglichkeit gegeben, neben einer zielgerichteten Suche, die vor allem unerfahrenen Systemnutzern anfangs Probleme bereitet (resultierend aus der mangelnden Erfahrung und dem fehlenden Wissen, was eigentlich gesucht werden soll), den Nutzer durch die Taxonomien der Ontologie navigieren zu lassen. Darüber hinaus muss es Ziel sein, durch selektive Präsentationsmechanismen, dem Nutzer einen Teil seiner Sucharbeit abzunehmen und ihn automatisch und profilabhängig mit den für ihn wichtigsten Informationen zu versorgen (siehe Kapitel 5.2.4).

### **d.) Wissensnutzung**

Die Nutzung von Wissen ist einer der wichtigsten und gleichzeitig einer der am schwersten zu fassenden Aspekte bei der Einführung eines Wissensmanagementsystems. Ontologiebasierte Systeme bieten bei der Wissensnutzung wiederum Vorteile gegenüber klassischen, dokumentenbasierten Systemen. So können mit dem Einsatz einer Ontologie dem Wissensnutzer Überblicke über gewisse Themengebiete geboten werden. Oftmals ist nicht nur ein spezielles Zieldokument für den Suchenden interessant, sondern auch der Kontext des Dokumentes. [Davis et al.] sind ferner der Meinung, dass sich aus Nutzerdaten wiederum wertvolle Informationen ableiten lassen, die direkt in das System mit einfließen könnten. So wäre es denkbar, Zugriffsdaten auszuwerten und anhand dieser bestimmte Mitarbeiter eines Unternehmens bestimmten Fachgebieten zuzuordnen. Diese Information könnte dann bei einer Suche nach Experten aktiv genutzt werden (siehe Abschnitt 5.2.4).

### **3. Das Portal der Fachhochschule Darmstadt – Anforderungen und Bedarf**

Dieses Kapitel beschäftigt sich mit dem Bedarf nach und den Anforderungen an ein Wissensmanagementsystem der Fachhochschule Darmstadt. Dafür wird zunächst die Fachhochschule und die daraus resultierenden Anforderungen in den Kontext der Unternehmen eingeordnet. Darauf folgend wird der Definitionsbereich von Intranetportalen in 3.2 beschrieben. In 3.3 werden schließlich Anforderungen der Mitarbeiter an eine Wissensmanagementlösung, die durch Abhalten von Workshops im Vorfeld erhoben wurden, evaluiert.

#### **3.1 Wissensmanagement im Hinblick auf die Fachhochschule**

Die Fachhochschule nimmt als Institution einen besonderen Platz in der Unternehmenswelt ein. Zur genaueren Einordnung muss man sich die organisatorischen Eigenschaften der Fachhochschule im Besonderen klarmachen. Die Fachhochschule Darmstadt ist den Studentenzahlen zufolge die größte Fachhochschule Hessens. Allein im Wintersemester 2002 begannen rund 2500 Studenten ihr Studium an dieser Hochschuleinrichtung. Diese Masse an „Kunden“ und die heterogene Mitarbeiterwelt an Hochschulen bedingen eine besondere Blickweise auf ein Wissensmanagementsystem. Laut [Bielefeld] ergeben sich im Rahmen der oben genannten Bausteine des Wissensmanagements unter anderem die folgenden Anforderungen an ein Wissensmanagement für Hochschulen:

- Aufbereitung der multimedial vorhandenen Daten
- Sicherung der Wissensressourcen und der Zugriff auf sie über ein Wissensportal
- Entwicklung von Instrumenten zur Wissensgenerierung
- Elektronisches Verfügbarmachen von Wissensobjekten
- Verbesserung der Informationskompetenz von Mitarbeitern
- Verbesserung der Kommunikationskompetenz von Mitarbeitern
- Entwicklung von elektronischen synchronen und asynchronen Kommunikationsformen innerhalb der Hochschule und in Wahrnehmung der Außenkontakte (Foren, Chats, Videoconferencing, „Virtual Communities“ etc).

- Integration der präsentationsgebundenen Funktionalitäten (wie bspw. des Web-auftritts)

Der Anforderungskatalog bestätigt die unter 2.1.1 festgestellten Teilaufgaben des Wissensmanagements. Vor allem der kommunikative Teil spielt dabei an einer Institution wie der Fachhochschule Darmstadt eine gewichtige Rolle. Der lokalen Streuung der Personalressourcen entgegenwirkend ist es besonders wichtig, räumliche Distanzen zwischen Wissensträgern zu verkleinern und Lösungen für eine verbesserte Kommunikationskultur zu finden.

Bei dem Wissensmanagementportal der Fachhochschule Darmstadt wird es sich dabei zunächst um ein Intranetportal handeln, welches erst zu einem späteren Zeitpunkt auch den Internetauftritt der Fachhochschule bedienen wird. Im nun folgenden Unterkapitel dieser Arbeit wird daher zunächst ein Überblick über Intranetportale (auch Mitarbeiterportale oder B2E (für Business to Employee) – Portale genannt) gegeben. Die eben genannten Punkte werden bei den Anforderungen an das FHD-Portal in den Kapitel 5 genauer zur Sprache kommen und mit den angesprochenen Bausteinen des Wissensmanagement und dem in Unterkapitel 3.3 noch festzustellenden Bedarf in Verbindung gesetzt werden.

### **3.2      *Definitionsbereich von Intranetportalen***

In den meisten Unternehmen gibt es bereits ein Intranet, welches, intensiv genutzt, die Informationsversorgung und Kommunikationsmöglichkeiten der Mitarbeiter verbessert und als zentrale Sammelstelle des internen Wissens dient. Allerdings ist dieses Wissen oft noch dezentral abgelegt und auf einer Vielzahl von Webseiten und Datensystemen verstreut. Um die gewaltigen Informationsbestände in den Intranets effektiv nutzen zu können, muss der Zugriff auf dieses Wissen individuell für jeden einzelnen Mitarbeiter optimiert werden. Außerdem sollen die Mitarbeiter von jedem Ort aus und zu jeder Zeit schnell und sicher Zugriff auf wichtige Informationen erhalten. Diese Verbesserung des Informationszugriffs kann die Arbeitseffizienz im Unternehmen wesentlich steigern. Ein individuell optimierter Zugriff auf die Informationen des Intranets lässt sich durch Mitarbeiterportale realisieren. Im Falle der Fachhochschule geht es dabei nicht nur darum, ein Portal für die Mitarbeiter (also Professoren, Laboringenieure, Verwaltungspersonal etc.) zu entwerfen, sondern auch die Bedürfnisse studentischer Mitglieder der Fachhochschule zu berücksichtigen. Dabei ist zunächst eine Einbindung der studentischen Gremien-

mitglieder geplant, die als Mittelsmann zu ihren Kommilitonen effektiven Zugriff auf die relevanten Informationen bekommen sollen.

Allgemein betrachtet ist ein Portal eine Umgebung, die kontextsensitiv relevante Informationen und Serviceangebote auf individuelle Bedürfnisse abgestimmt bereitstellt. Dabei kann sich generell der persönliche Bezug an Kunden, Partnern oder auch den eigenen Mitarbeitern oder, wie im Beispiel der Fachhochschule, an Studenten orientieren. Je nach Zielrichtung werden dann Portale für den Kontakt mit Lieferanten (B2B – Business-to-Business), mit den Kunden (B2C – Business-to-Customer) und schließlich für den Kontakt mit den eigenen Mitarbeitern (B2E – Business-to-Employee) unterschieden. Weiterhin unterscheidet man zwischen horizontalen und vertikalen Portalen. Vertikale unterstützen eine bestimmte Funktion im Unternehmen, horizontale Portale sind auf Anwender zugeschnitten. Bei dem geplanten Portal der FHD handelt es sich folglich um ein horizontales.

Durch den stark wachsenden Informationsbestand im Intranet sehen sich viele Institutionen heute dabei mit redundanten Datenbanken, unterschiedlichen Datentypen und vielfältigen Prozessen für Erzeugung und Bearbeitung dieser Daten konfrontiert. Historisch gewachsen existieren auch verschiedene Plattformen und Betriebssysteme nebeneinander. Daraus resultiert die Anforderung, das Mitarbeiterportal auf einer stabilen und zuverlässigen Architektur aufzusetzen, welche die nötige Interoperabilität gewährleistet und zugleich eine Möglichkeit zur Konsolidierung der bestehenden heterogenen IT-Landschaft bietet. Idealerweise soll das Mitarbeiterportal allen Mitarbeitern als zentrale Informationsdrehscheibe und Kommunikationsplattform dienen, Inhalte schnell und unkompliziert verfügbar machen, sowie einen Bereich für das Management von Aufgaben und zum Ideen- und Meinungsaustausch enthalten.

In der gegenwärtig von starkem Wettbewerb geprägten ökonomischen Situation muss das Mitarbeiterportal auch an einer Institution wie der Fachhochschule Service nach Bedarf liefern können. Für den Erfolg der Portal-Strategie ist bei jedem einzelnen Schritt eine rasche Amortisierung entscheidend. Eine bessere Handhabbarkeit, reduzierte Kosten pro Transaktion und ein schnellerer Zugriff auf die gesuchten Informationen erhöhen den Wert der verfügbaren Inhalte und Services. Dieser Zusammenhang ist der Kerngedanke, welcher den Trend hin zu Mitarbeiterportalen begründet. Ein erfolgreiches Portal sorgt für die Rationalisierung der Informationsbestände und IT-Ressourcen im Unternehmen, sowie für eine effizientere Kommunikation.



Dabei ist es wichtig dem Nutzer des Portals einen sogenannten „Single-Point-of-Access“ zu gewähren. Dies bedeutet, dass der Mitarbeiter über das Portal Zugang zu allen vorhandenen Informationsquellen der Institution erhält und auf die für ihn wesentlichsten, ohne danach suchen zu müssen. Eine einheitliche Benutzerführung (d.h. Navigation) sollte dabei ebenso gewährleistet sein, wie gleichbleibende Darstellungselemente. Dies bedeutet, dass Submenüs sich dem Nutzer immer in gleicher Optik präsentieren sollten, egal auf welche Quelle er gerade zugreift. Zusammenfassend bedeutet dies einen einheitlichen Zugriff auf Daten und Information, externe Programme sind bestenfalls nicht sichtbar, sondern werden in ihrem Erscheinungsbild dem Aussehen des Portals angepasst.

Der allgemeine Aufbau von Mitarbeiterportalen lässt sich dabei wie in Abbildung 9 dargestellt definieren. Demzufolge gliedert sich die Struktur eines Intranet Portals in mehrere Bereiche. Für den Prototypen der Fachhochschule Darmstadt wird diese Struktur exemplarisch zu Hilfe genommen werden, Details werden jedoch aus Gründen der spezifischen Situation der Fachhochschule und des besonderen Informationsbedarfs von dieser Darstellung abweichen.

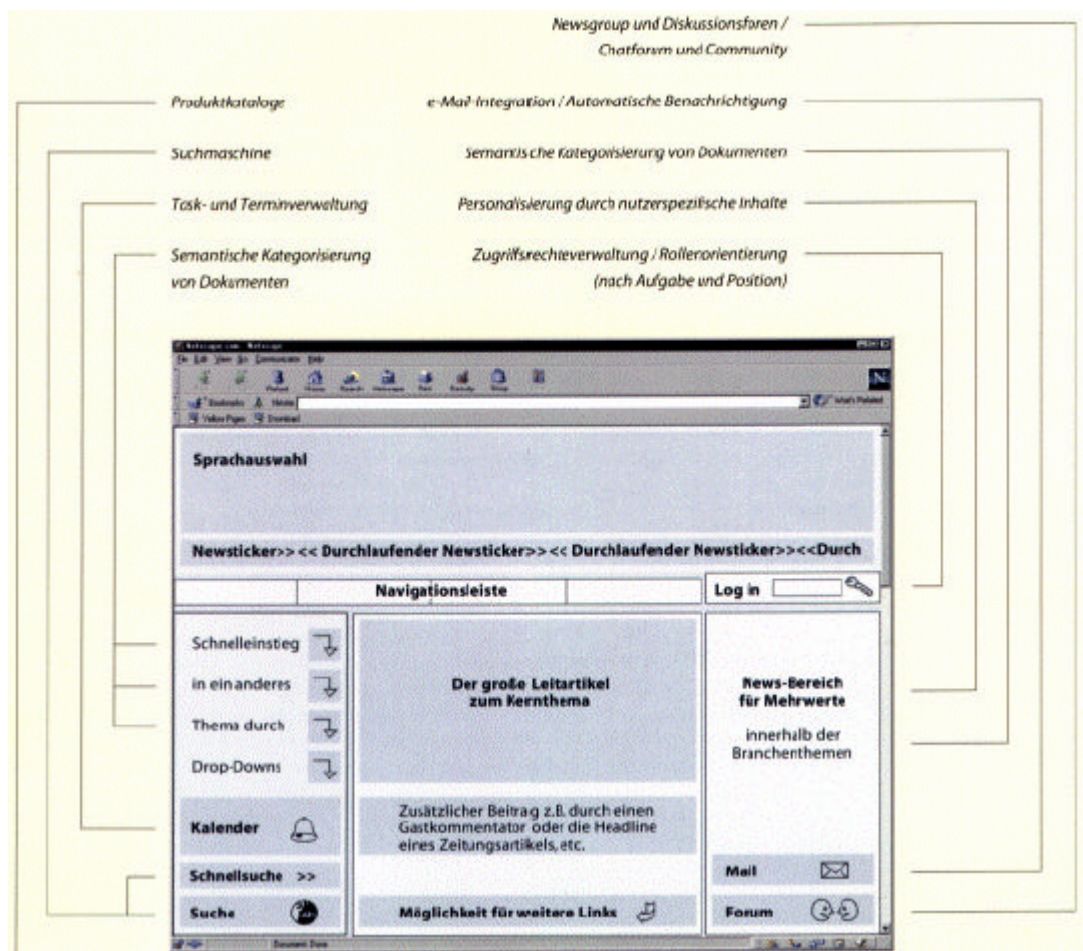


Abb.9: Aufbau von Portalen ([Schöpe])

Bei der Entwicklung von Unternehmensportalen sind dabei laut [Managers] mehrere Sichten auf den Mitarbeiter im Zuge der Entwicklung eines Intranetportals vonnöten. So sei die Sicht des Mitarbeiters als Kunde mit den daraus resultierenden Bedürfnissen zunächst erforderlich. Darüber hinaus gälte es, den Mitarbeiter als Wissensträger zu betrachten und im Zuge von Knowledge Managementmaßnahmen ihn zum „Wissensweitergeber“ werden zu lassen. Die entgegengesetzte Sicht, nämlich die des Wissensempfänger sei ebenso zentral und gleichzeitig Voraussetzung für die Betrachtungsweise des Leistungserbringers. Die Sichtweise auf den Mitarbeiter als wichtigste Ressource des Unternehmens fasst schließlich die übrigen Blickwinkel zusammen und verdeutlicht die Rolle des Mitarbeiters bei der Einführung eines Intranetportals.

### **3.3      *Bedarfserhebung an ein Intranet-Portal der Fachhochschule Darmstadt***

Die Senatskommission „Wissensmanagement“ hat im Zuge der Vorarbeiten für die Einführung des Intranet-Portals der Fachhochschule Darmstadt verschiedene Workshops zur Erhebung des Informationsbedarfs der unterschiedlichen Parteien der Fachhochschule durchgeführt. Dabei gelten zunächst alle Mitarbeiter der Fachhochschule (Professoren, Verwaltung, Referate etc.) als Zielgruppe des Portals. Teilnehmer der Workshops waren Vertreter der wissenschaftlichen Mitarbeiter und Laboringenieure, der zentralen technischen Einheit Dieburg (ZTE), der Professoren und der Verwaltung. Zusätzlich zu den Workshops wurde in (unterschiedlich formalen) Gesprächen mit Studierenden verschiedener Fachbereiche (vor allem Informations- und Wissensmanagement, Informatik und Wirtschaft) der Fachhochschule Darmstadt ein Bedarf der studentischen Gremienmitglieder als weitere Nutzergruppe des Portals erhoben. Dabei spielte vor allem die Berücksichtigung der unterschiedlichen Standorte eine gewichtige Rolle. Die Ergebnisse der Workshops und dieser Erhebungen finden sich in Kapitel 5 dieser Arbeit in den Konzepten wieder.

Die Vorgehensweise bei den Workshops war dabei stets dieselbe: Zunächst wurde danach gefragt, wie gut sich die Mitarbeiter mit den nötigen Informationen versorgt und in wie weit sie sich in ihrer Arbeit geschätzt sehen. Danach wurde konkret nach Verbesserungswünschen gefragt, die dann in der Gruppe diskutiert wurden. Die Gespräche mit den Vertretern der Studierenden verlief zumeist informeller. Ein Fragebogen (zu finden im Anhang) wurde entworfen, um den Vertretern einen Denkanstoß zu geben; notwendig war dies bei den meisten Gesprächen jedoch nicht, da die Vertreter oft konkrete Mängel bzw. Verbesserungsvorschläge nennen konnten. Da die Protokolle selbst aus da-

tenschutztechnischen Gründen nicht in den Anhang aufgenommen werden können, wird der Bedarf als Ergebnis der Workshops im Folgenden in aller Kürze wiedergegeben.

- **Verfügbarkeit und Qualität der Informationen**  
In den Workshops wurde deutlich, dass alle teilnehmenden Parteien der Fachhochschule den Informationszugang und die Informationsqualität als mangelhaft einstufen. Als Resultat davon fühlen sich viele in ihren Bedürfnissen unverstanden und in ihren Fähigkeiten nicht ausgelastet. Auf studentischer Seite bringt dieser Punkt ein erhöhtes Frustpotential mit sich. Die befragten Gremienmitglieder betonten bei den Gesprächen, dass insbesondere Informationen über andere Fachbereiche und über die Verwaltung sehr schwer zugänglich seien und somit Ansprechpartner für entsprechende Vorhaben nur mit einem gewissen Aufwand zu finden seien.
- **Verbesserung der Kommunikation**  
Mangelhafte Kommunikation wurde als einer der Hauptpunkte für Ineffizienz und Mehrarbeit genannt. Vor allem die Forderung nach einem einheitlichen Kommunikationsrahmen war diesbezüglich einer der Hauptanforderungspunkte. Von studentischer Seite wurde zu bedenken gegeben, dass für eine erfolgreiche Kommunikation eine Transparenz bezüglich Ansprechpartner und deren Funktionen unerlässlich sei .
- **Vereinheitlichung von Begriffen**  
Um „eine Sprache“ an der Hochschule zu sprechen, wurde gefordert, ein einheitliches Begriffssystem an der FH einzuführen, um Missverständnisse bei der Kommunikation zu vermeiden.
- **Redundanzvermeidung bei der Datenhaltung:**  
Redundante Datenhaltung wird unter den Mitarbeitern der Fachhochschule als Hauptursache für ineffizienten Informationszugriff gesehen. So sollte Ziel sein, einen universellen Zugriff auf alle vorhandenen Daten und Quellen zu bieten, um redundante Mehrfachhaltung von Daten zu vermeiden.
- **Raumplanung / Ressourcenmanagement**  
Im Zuge eines Intranetportals für die Fachhochschule Darmstadt sehen es die Befragten als wünschenswert, in die Portalfunktionalität die Möglichkeit einer

Ressourcenverwaltung bezüglich Räume, technischer Gerätschaften usw. zu bieten.

- **Abstimmung bei abteilungsübergreifenden Prozessen**  
Vor allem bei Prozessen und Projekten, die sich über mehr als eine Abteilung (respektive: mehr als einen Fachbereich) ziehen, seien Abstimmungsdefizite zu erkennen. Diese Defizite resultierten einerseits aus mangelnder Kommunikation, andererseits aus fehlendem Wissen über die andere Partei.
- **Transparenz bei den Zuständigkeiten**  
Fehlende Transparenz bei den Zuständigkeiten wurde von allen befragten Parteien als Manko erkannt. So gälte es, mittels Zuständigkeitslisten den Mitarbeitern / Gremienmitgliedern Informationen über Ansprechpartner für gewisse Probleme oder Aufgaben zur Verfügung zu stellen.
- **Mitarbeiterorganisation**  
Mitarbeiterorganisation wurde vor allem auf die Erreichbarkeit der einzelnen Hochschulmitglieder bezogen. So sei es wünschenswert Abwesenheitslisten zu entwerfen, die Aufschluss darüber geben, wie und wann ein Mitarbeiter der Fachhochschule erreichbar ist (bspw. bezogen auf Halbtagskräfte).
- **Qualifikation und Weiterbildung**  
Innerhalb der Workshops wurde auch die Verbesserung von Informationen und Zuordnungen zu weiterbildenden Maßnahmen als ein Aspekt genannt. So sei es derzeit schwer möglich, Weiterbildungsmaßnahmen entsprechend zu verbreiten bzw. als Interessent zentral an solche Informationen zu gelangen.
- **Ablauf und Prozessbeschreibungen**  
Wiederholt auftretende Prozesse und der Ablauf fester Vorgänge sollte in der Art standardisiert und festgehalten werden, dass ein Einarbeiten in diese stark verkürzt werden kann.
- **Ombudsmann**  
Der Bedarf nach einem Vermittler zwischen einzelnen Parteien (insbesondere für Studentenprobleme) wurde innerhalb der Workshops ebenfalls deutlich. Dieser solle als unabhängiger Ansprechpartner bei Problemen, Unstimmigkeiten und Beschwerden dienen.

## 4. Die Ontologie des FHD-Portals – Modellierung und Anbindung

### 4.1 Die Software-Suite „K-Infinity“

„K-Infinity“ ist eine von der Darmstädter Firma „intelligent views“ entwickelte Software-Suite, die den Anwender bei der Entwicklung ontologiebasierter Systeme unterstützt. Mit den unterschiedlichen Komponenten ist es so möglich, eine Wissensbasis aufzubauen, zu pflegen und zu nutzen. Die Kernfunktionalität von „K-Infinity“ ist dabei die Vernetzung von Informationen aus unterschiedlichsten Quellen mit Hilfe eines Wissensnetzes (=Ontologie) und damit die Basis für intelligentes Wissensmanagement. Schwerpunkt bei der Entwicklung eines „K-Infinity“-basierten Systems ist dabei die Strukturierung von Information. Die unterschiedlichen Komponenten interagieren dabei nach dem in Abbildung 10 dargestellten Schema.

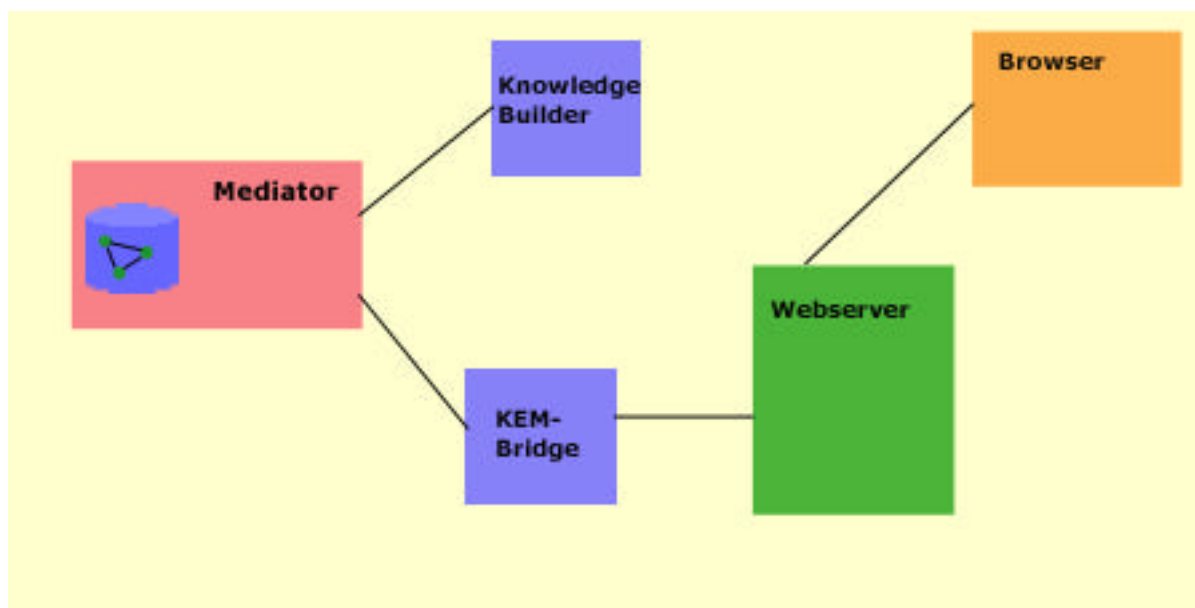


Abb. 10: Komponenten der „K-Infinity“-Suite

Der sogenannte Mediator ist dabei die Serversoftware, welche die Ontologie und sämtliche Zugriffe auf sie verwaltet. Als unterschiedliche Clients fungieren einerseits der sogenannte Knowledge Builder und die KEM-Bridge<sup>8</sup>. Der Knowledge Builder ist dabei die Software zum Editieren der Begriffstaxonomie, während die Bridge als Schnittstelle zum Webserver dient. Per Browser greift der Endnutzer schließlich über den Webserver und

<sup>8</sup> KEM: „K-Infinity“ Editor Model

die Bridge auf das Wissensmodell zu. Im folgenden Unterkapitel werden die Komponenten im Einzelnen erklärt.

#### **4.1.1 Systemkomponenten der „K-Infinity“ Suite**

##### **a.) Der Mediator**

Der Serverteil der „K-Infinity“ Software sorgt für die Persistenz und Konsistenz der Datenhaltung und für die Aktualität der Daten auf den angeschlossenen Clients. Die Serverschnittstellen ermöglichen dabei den verteilten Zugriff auf das Wissensnetz und realisieren Funktionalitäten wie semantische Suche, Volltextsuche und Navigationssuche (Browsing). Im Sinne einer kollektiven Modellentwicklung propagiert der Server Änderungen am Wissensmodell an alle Clients. Die Implementierung des Mediator erfolgte in Smalltalk, einer streng objektorientierten Programmiersprache mit reinen Objekttypen. (vgl. [Holzer]).

##### **b.) Der Knowledge Builder**

Der Knowledge Builder (K-Builder) bildet die Kernkomponente der Software Suite und regelt den Aufbau und die Pflege der Ontologie. Konsistenzprüfungen sind bereits integriert, bspw. in Form von kaskadierendem Löschen oder bei der Vergabe von Attributen und Relationen.

Zwei Programmkomponenten ermöglichen dabei dem Nutzer die ergonomische Entwicklung: Einerseits der K-Organizer und andererseits der Graph-Editor. Beide Sichten ermöglichen die Erstellung, Editierung und das Löschen von Wissensobjekten, die sich in Begriffe, Individuen, Relationen und Erweiterungen (Rollen) aufteilen. Die Modellierung ist dabei objektorientiert, d.h. Relationen und Attribute können weitervererbt werden.

Der K-Organizer (Abbildung 11) ist ein textorientiertes Verwaltungstool, welches in Taxonomien geordnet die einzelnen Wissensobjekte anzeigt (Getrennte Taxonomien für Begriffe, Relationen, Individuen und Dokumentenbegriffe).

Dem Modellierer werden zahlreiche Möglichkeiten geboten die Ontologie mittels Suchanfragen auf Konsistenz und Umfang zu überprüfen. So ist im K-Organizer neben einer Volltextsuche eine semantische Suche implementiert, die automatisch Inferenzen über die gezogenen Relationen zieht. Auch hat der Modellierer die Möglichkeit, komplexe Suchanfragen anzufertigen, die später direkt über das Webfrontend aufgerufen werden können („Expertensuche“).

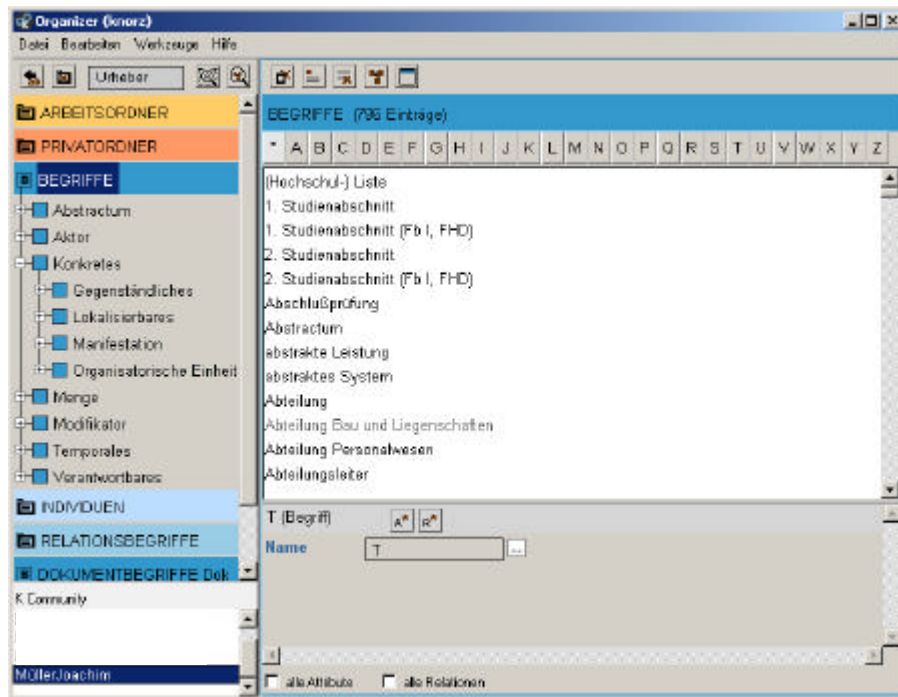


Abb. 11: K-Organizer

Im Graph-Editor (Abbildung 12) werden die Wissensnetze und ihre Beziehungen zueinander dargestellt. Der Nutzer kann dabei auf visueller Ebene assoziativ interagieren und zudem durch Navigation durch die vorhandene Ontologie einen schnellen Überblick über einzelne Wissensobjekte und deren Kontext erhalten. Ein Algorithmus ordnet dabei automatisch die Wissensobjekte an, durch „Ziehen“ mit der Maus ist die individuelle Layoutgestaltung möglich.

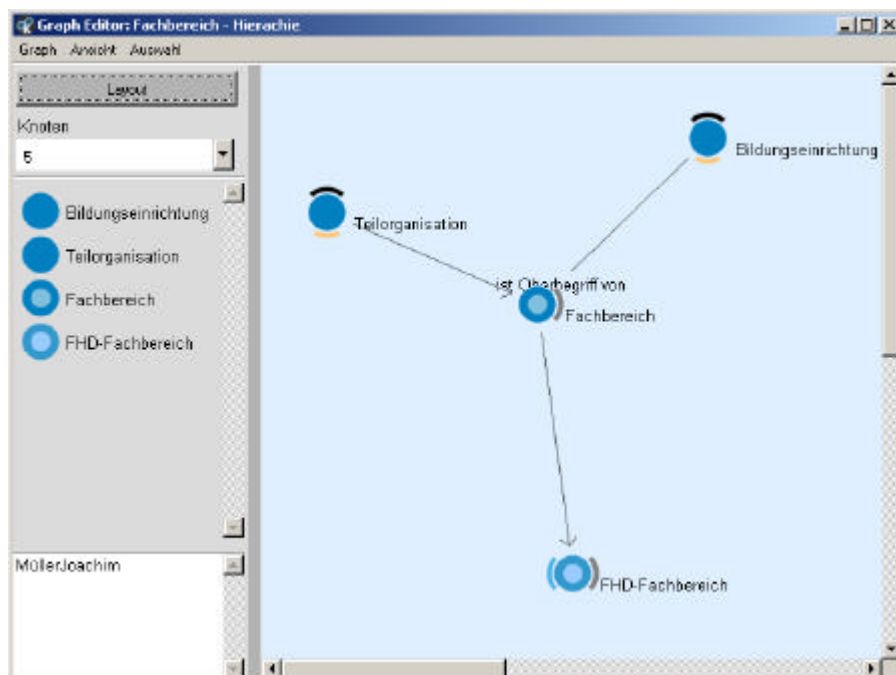


Abb. 12: Graph-Editor

Zur Unterstützung des kollektiven Arbeitens ist in allen Bereichen des K-Builders ein Chatsystem implementiert, welches die Kommunikation der einzelnen Modellierer unterstützt. Zur Überprüfung des Modells sind verschiedene Suchmöglichkeiten integriert. Ferner gibt es ein umfassendes Rechtesystem, welches den Zugriff auf das Datenmodell regelt (siehe Kapitel 5.4) .

### **c.) Die KEM-Bridge**

Die KEM-Bridge ist die Bridge zwischen dem Mediator und dem Webserver. Sämtliche Webzugriffe auf die Ontologie finden über die Bridge statt. Ferner wird, orientierend an implementierten Zugriffsrechten, die Ontologie gefiltert. Zusätzlich zur KEM-Bridge gibt es faktisch noch die sogenannte Multibrige. Dieser Teil der Bridge ist nur für den Zugriff des Net-Navigators zuständig und soll künftig ebenfalls über die KEM-Bridge realisiert werden

### **d.) Webserver**

Als Webserver (und Servlet Container, siehe 4.3) für die Anbindung der modellierten Ontologie dient derzeit der Tomcat Server des Apache Jakarta Projektes. Die Anbindung der Ontologie an ein Webfrontend geschieht unter Zuhilfenahme des Struts-Frameworks und wird in Kapitel 4.4 ausführlich beschrieben. Der Zugriff durch den Nutzer geschieht über einen Standardwebbrowser, zusätzliche Software ist, außer einer „Java Runtime Environment“, nicht vonnöten.

## **4.2 Status Quo der Ontologie**

Die Kern-Ontologie des Portals der Fachhochschule Darmstadt wurde im Rahmen eines Studienprojektes im Wintersemester 2002/2003 des damaligen siebten Semesters am Fachbereich Informations- und Wissensmanagement mithilfe des Software-Toolkits „K-Infinity“ entwickelt. In Gruppen zu je zwei bis drei Studierenden wurde dabei der Weltausschnitt der Hochschule und deren Umfeld modelliert. Das zugrundeliegende Datenmodell ist framebasiert (vgl. 2.3.1) und somit von erhöhter Ausdrucksstärke. Aufgrund der beschränkten Importfunktion der „K-Infinity“-Software wurde auf die Integration vorhandener Top-Level-Ontologien verzichtet<sup>9</sup>. Dies stellt aber im Rahmen der Fachhochschule Darmstadt und vor allem im Hinblick auf das Einsatzgebiet der Ontologie, nämlich das Intranet, kein ernstzunehmendes Problem dar, da ein durch Integration bestehender On-

---

<sup>9</sup> Die Möglichkeit des Imports bestehender Ontologien besteht erst seit der „Knowledge Builder“-Version 1.3, die im dritten Drittel dieser Arbeit geliefert wurde und deren Importfunktionalität daher nicht berücksichtigt wurde.



tologien mögliches erhöhtes „Ontology commitment“ (d.h. der Bezug auf ein gemeinsames Modell über der Fachhochschule hinaus) für eine betriebsinterne Anwendung nicht unbedingt vonnöten ist. Auch eine in der Ontologie selbst modellierte Definition des Sprachmodells ist aufgrund des proprietären, in „K-Infinity“ fest vorgeschriebenen Datenmodells nicht nötig. Bei der Ontologie des FHD-Portals handelt es sich vielmehr um ein isoliertes, nur für den Gebrauch an der Fachhochschule vorgesehenes und daraufhin optimiertes Realitätsmodell.

Zum Zeitpunkt dieser Arbeit (April 2003) besteht die Ontologie aus knapp 800 Begriffen, 700 Individuen und knapp 250 Relationen (die inversen Relationen mitgerechnet). Zusätzlich existieren ca. 100 Dokumentenbegriffe, deren Instanzen die Dokumente der Ontologie sind. Die Dokumentenbegriffe bilden dabei eine Taxonomie unabhängig vom eigentlichen Begriffsnetz, die am Beispiel der FHD Ontologie ausschnittsweise in Abbildung 13 zu sehen ist.

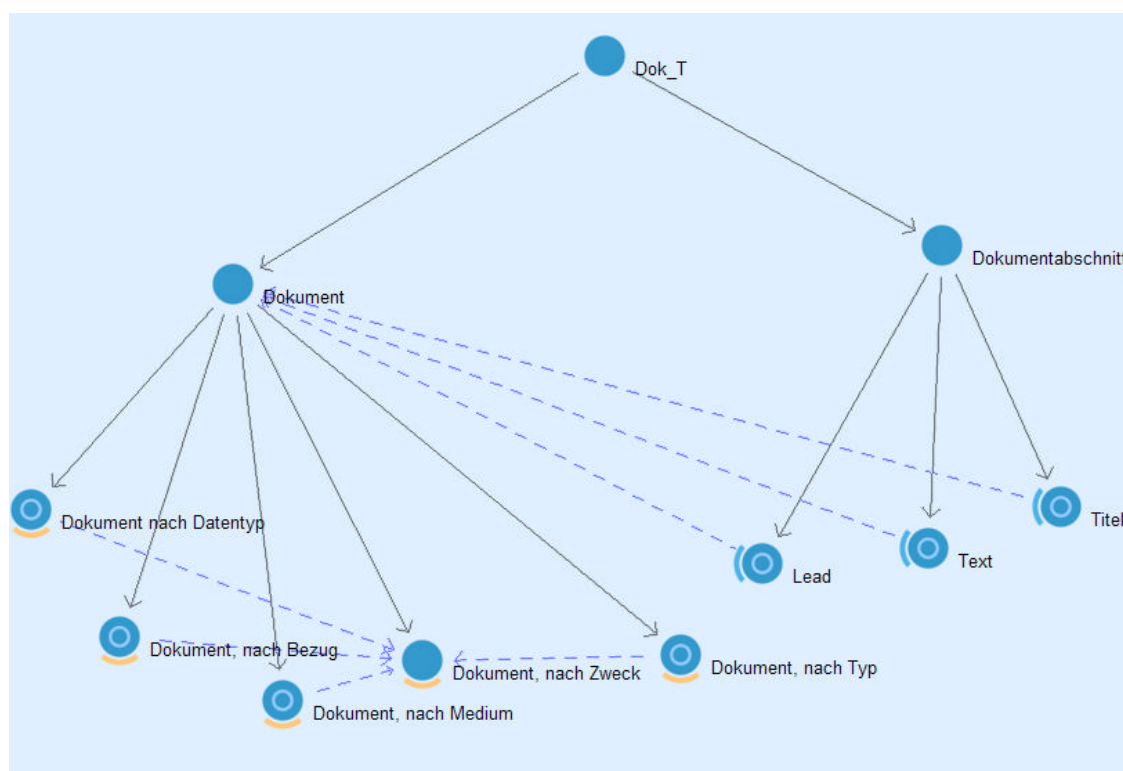


Abb. 13: Dokumentenbegriffshierarchie der FHD Ontologie

Die gestrichelt blauen Relationen bedeuten dabei, dass die einzelnen Dokumentenbegriffe Instanzen des Zielbegriffes erweitern.

113 Dokumente sind derzeit in das Begriffsnetz eingepflegt. An dieser Stelle sei nochmals betont, dass im Netz selbst nicht dokumentenbasiert, sondern begriffsbasiert gesucht wird. Dies impliziert, dass ein exaktes Einpflegen der Dokumente und die genaue Zuweisung zu den Begriffen unerlässlich ist.

### 4.3 *Methodologie zur Evaluierung der Ontologie*

Die Ontologie des FHD-Portals befindet sich, wie im vorangegangenen Kapitel beschrieben, bereits in einem fortgeschrittenen Entwicklungszustand. An diesem Punkt ist es wichtig, eine Methodologie zu entwickeln, die es möglich macht, anhand fester Kriterien die formelle und inhaltliche Richtigkeit zu überprüfen. Ziel ist es dabei, den Vorgang der Ontologienentwicklung weg von individueller Einzelleistung, hin zu einem allgemein nachvollziehbaren Vorgang werden zu lassen. [Jones et al.] sprechen dabei vom Ziel, Ontologiedesign als Wissenschaft im Gegensatz zu intellektueller Kunst zu betreiben. Im Zuge dessen ist die Überprüfung folgender Kriterien hilfreich:

- **Deutlichkeit:** Ein Subkonzept muss sich vom Superkonzept unterscheiden. Die Unterscheidungen, die dabei getroffen werden, müssen sowohl notwendig, als auch hinreichend sein. Notwendig bedeutet dabei, die Unterscheidung der Konzepte aus intensionaler Sicht. Dies bedeutet, dass basierend auf Annahmen des Modellierers verschiedene Konzepte, Beziehungen und Individuen modelliert werden.

Unter Intension versteht man laut [Brockhaus] die Assoziation des Benutzers mit dem Begriff. Als Beispiel hierfür wird in der Literatur zumeist das Beispiel des „Morgensterns“ und des „Abendsterns“, nach Gottlieb Frege zitiert. Beide Bezeichnungen haben dieselbe extensionale Bedeutung, nämlich den Planeten Venus, unterscheiden sich aber in ihrer Intension. Extensional bedeutet dabei das, was der Begriff zusammenfasst, also der Umfang des Begriffes. An diesem Beispiel wird auch deutlich, was es heißt, Unterscheidungen sowohl notwendig, als auch hinreichend zu definieren. Eine notwendig getroffene Unterscheidung wäre die nach der Intension, also im Venus-Beispiel die Unterscheidung zwischen Abend- und Morgenstern in zwei verschiedenen Konzepten (vgl. [Bouaud et al.]). Das sogenannte Extensionalitätsaxiom hingegen besagt, dass zwei Begriffe gleich sind, wenn ihre Extension übereinstimmt. In einer Ontologie ist es laut Definition Ziel, gleiche Begriffe in einem Konzept zu subsumieren. Eine notwendige und hinreichende Unterscheidung bedeutet also im Beispiel die Modellierung des Begriffsumfanges des Planeten Venus. D.h. ein einziges Konzept mit vorstellbaren intensionalen Unterscheidungen „Morgenstern“ und „Abendstern“ in der Synonymie. Ziel bei Unterscheidungen nach notwendigen und hinreichenden Aspekten ist es dabei, die Abbildung der Wirklichkeit so homomorph wie möglich zu gestalten, ohne den Anwendungsblickwinkel der Ontologie zu verlieren. Dies be-

deutet, dass sehr genau überlegt werden muss, wann die Extension als Modellierungsziel zu verstehen ist und wann intensionale Unterscheidungen in der Konzeptdefinition eine Rolle spielen.

- **Klarheit:** Definitionen einer Ontologie sollen objektiv die Bedeutung des entsprechenden Konzeptes modellieren. Die Entscheidung, ein Konzept zu definieren, entspringt oftmals sozialen Zusammenhängen, die Definition hingegen sollte frei von sozialem Kontext sein.
- **Kohärenz:** Eine Ontologie sollte kohärent sein, d.h. einen logischen Zusammenhang bilden. Konzepte dürfen sich dabei in ihrer Definition nicht widersprechen. Inferenzen müssen anhand der Konzeptdefinitionen gebildet werden können und dürfen dabei in keinerlei Widerspruch zu vorhandenen Definitionen stehen. Gilt es als erwiesen, dass ein Konzept Subkonzept eines anderen ist, dies aber der Definition des Superkonzeptes widerspricht, so muss dessen Definition angepasst werden.
- **Unabhängigkeit:** Die Subklassen eines Konzeptes sind unvereinbar. Um eine konsistente Taxonomie aufzubauen ist es unerlässlich, sicherzustellen, dass Subkonzepte eines Konzeptes gegenseitig inkompatibel sind. Das bedeutet, zwischen den direkten Geschwistern innerhalb einer Taxonomie gibt es keine Gemeinsamkeit außer der Bedeutungsgemeinsamkeit mit dem Superkonzept.
- **Ähnlichkeit:** Das Ähnlichkeitsprinzip besagt, dass ein Subkonzept eine Bedeutungsgemeinsamkeit mit dem Superkonzept besitzen muss. Diese, zunächst recht vage Definition soll anhand eines Beispiels verdeutlicht werden: Man stelle sich ein Konzept „Zustand“ vor. Ein Zustand sagt etwas über Objekte der Welt aus. „Geistiger Zustand“ beschreibt hingegen mentale Objekte, welche wiederum Objekte der Welt sind. Weil „geistiger Zustand“ eine Gemeinsamkeit in der Bedeutung mit „Zustand“ hat, ist „geistiger Zustand“ Subkonzept von Zustand. Anders ausgedrückt bedeutet dies, dass die Aussage, ein Zustand zu sein, Voraussetzung dafür ist, ein geistiger Zustand zu sein. „Geistiger Zustand“ ist also eine Spezialisierung von „Zustand“.
- **Spezifität:** Das Spezifitätsprinzip dient ebenfalls der Rechtfertigung eines Konzeptes, allerdings aus dem Blickwinkel des entsprechenden Oberbegriffs. Dabei werden die notwendigen und hinreichenden Bedingungen zur Unterscheidung der Konzepte deutlich, die zu der vollständigen Definition des Subkonzeptes führt. Im

gerade eingeführten Beispiel würde das Spezifitätsprinzip angewandt die folgende Definition ergeben: „Geistiger Zustand“ ist, relativ betrachtet zu „Zustand“, ein Zustand, der mentale Objekte involviert, die spezifische Objekte der Welt darstellen. Ein Zustand zu sein, der mentale Objekte betrifft, ist damit die komplette Definition von „Geistigem Zustand“.

- **Einheitliche semantische Achse (Monodimensionalität):** Die Subklassen eines Konzeptes sollten sich von der Superklasse anhand einheitlicher Eigenschaften auf gleichem Abstraktionsniveau unterscheiden. Dies bedeutet, dass der Kontext der Unterscheidungen der einzelnen Subkonzepte derselbe ist [Jones et al.]. Dieser Aspekt soll im Folgenden im Zusammenhang von Forschung verdeutlicht werden. Derzeit zeigt sich die Begriffstaxonomie im Bereich Forschung in der Ontologie der Fachhochschule Darmstadt wie in Abbildung 14 dargestellt:

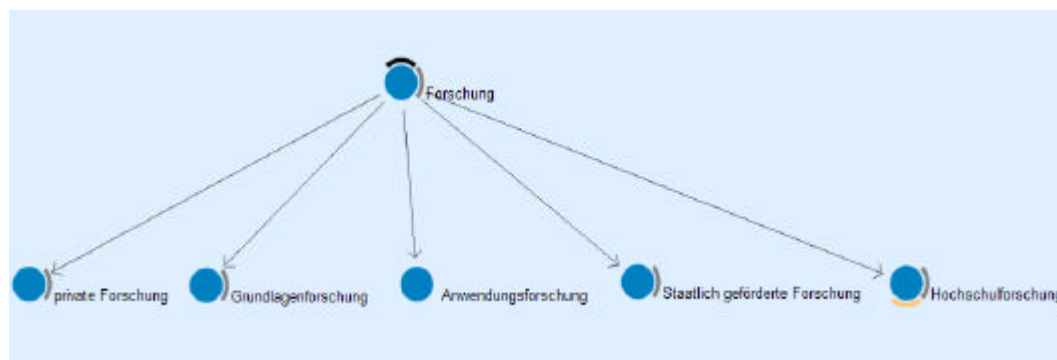


Abb. 14: Teiltaxonomie im Bereich Forschung

Wie zu sehen, hat das Konzept „Forschung“ dabei die direkten Subkonzepte „private Forschung“, „Grundlagenforschung“, „Anwendungsforschung“, „Staatlich geförderte Forschung“ und „Hochschulforschung“. Die Verletzung des Grundsatzes der einheitlichen semantischen Achse zeigt sich hierbei wie folgt. Ein genauerer Blick auf den Begriff „Grundlagenforschung“ ergibt folgende Definition: Grundlagenforschung ist „zweckfreie, nicht auf unmittelbare praktische Anwendung hin betriebene Forschung, die sich mit den Grundlagen einer Wissenschaft [...] beschäftigt“ [Duden]. Dies bedeutet auf das oben eingeführte Ähnlichkeitsprinzip angewandt, Grundlagenforschung ist Forschung, die sich mit den Grundlagen einer Wissenschaft beschäftigt. Im Gegensatz dazu steht der Begriff der Anwendungsforschung, welche per Definition Forschung ist, die sich mit der Anwendung einer Wissenschaft beschäftigt. Hier wird deutlich, dass die einheitliche semantische Achse vorhanden ist, der Unterscheidungsaspekt der Subkonzepte ist ein-

heitlich (woraus ferner folgt, dass die Unabhängigkeit der Subkonzepte gewährleistet ist, siehe oben).

Betrachtet man hingegen das Konzept „Hochschulforschung“, so ergibt sich folgende Definition: Hochschulforschung ist Forschung mit der zusätzlichen Eigenschaft, dass sie an Hochschulen durchgeführt wird. Das Kriterium, welches Hochschulforschung von Forschung unterscheidet, ist also die Lokalität, an welcher sie stattfindet. Demzufolge ist die semantische Achse eine andere als die der erstgenannten Subkonzepte. Selbiges gilt für die Konzepte „Private Forschung“ und „Staatlich geförderte Forschung“. Als Folge davon ist auch die Inkompatibilität der Subkonzepte nicht mehr gewährleistet (Anwendungsforschung kann durchaus im Rahmen von Hochschulforschung erfolgen) und die Taxonomie nicht mehr wohlgeformt (siehe auch Kapitel 2.3.1 bezüglich „konzeptionelle[n] Ontologien im Vergleich zu linguistischen“ und [Fensel]).

Eine Lösung dieses Problems ist allerdings schwerer, als es auf den ersten Blick scheint. Mit der Einführung von Zwischenebenen wird nämlich zumeist der unter „Deutlichkeit“ beschriebene Aspekt berührt. So ist es hier möglich eine semantische Zwischenebene einzuführen, die Forschung unterteilt in „Forschung nach Lokalität“, „Forschung nach Mittelherkunft“ und „Forschung nach Bereich“. Die bisher vorhandenen Konzepte würden dann dem jeweiligen Konzept als Subkonzept zugeordnet. Starr nach dem Prinzip der extensionalen Entscheidung handelnd, wäre dies allerdings ein Verstoß gegen diese Regel: So haben die nun neu eingeführten Konzepte zwar eine unterschiedliche intensionale Bedeutung, aber ihre Extension ist jedesmal dieselbe, nämlich „Forschungsprojekt“. An dieser Stelle ist es besonders wichtig abzuwägen, welches Prinzip höher gewichtet werden sollte. In dem konkreten Anwendungsfall ist eine Unterscheidung nach intensionalen Aspekten eher zu empfehlen, als ein Verstoß gegen das Prinzip der einheitlich semantischen Achse. Diese Entscheidung beruht vor allem darauf, dass in der Praxis der Forschung an der Fachhochschule (insbesondere im Zentrum für Forschung und Entwicklung) nach eben genau diesen intensionalen Aspekten unterschieden wird und auch bspw. Formulare intern nach diesen Kriterien geordnet sind. Verallgemeinernd sollte der Fokus auf die reale Welt das ausschlaggebende Kriterium für eine Entscheidung sein. Ziel der Ontologie sollte es sein, der realen Welt (und dem Einsatzgebiet) so nahe wie möglich zu kommen.

- **Objektidentität:** Die Objektidentität ist einer der Faktoren, die für den Zugriff auf die Ontologie entscheidend sind. Das Prinzip wird in [K-Infinity2001] anhand folgendem Beispiel verdeutlicht: Ein Ort wie London kommt in einem Realitätsmodell in verschiedenen Kontexten vor. So kann er einmal als Geburtsort einer Person auftreten, ein anderes Mal als Ort des Arbeitsplatzes und ein drittes Mal als Hauptstadt Englands. Im Sinne der Objektidentität werden solche Informationen nicht einfach als Attribute der Wissensobjekte eingetragen, sondern vielmehr sollte ein eindeutiges Informationsobjekt vom Typ Ort angelegt werden, auf welches diese verweisen. Dieses eindeutige Wissensobjekt akkumuliert alle Informationen, die in der Wissensbasis zu London eingetragen werden und macht sie an einer zentralen Stelle verfügbar.
- **Umfang der Relationen:** Dieses Kriterium zur Evaluierung einer Ontologie ist eines der am schwersten zu definierenden. Wie oben bereits im Zuge der einheitlich semantischen Achse, sollte bei der Anzahl der Relationen die Anwendung ausschlaggebend für eine mögliche Implementierung einer Relation sein. Zu viele und zu spezifische Relationen machen die Anwendung unübersichtlich, zu wenige hingegen repräsentieren den dargestellten Weltausschnitt nicht adäquat. Im Gegensatz zu Begriffen sollten Relationen **immer** in ihren Anwendungsbereichen klar voneinander abgetrennt sein, um eine uneindeutige Anwendung zu vermeiden. Als Anhaltspunkt kann bei der Frage, ob eine Relation definiert werden soll oder nicht, der oben bereits angesprochene Aspekt der Entscheidung nach intensionalen oder extensionalen Gesichtspunkten hinzugezogen werden. Hat eine neue Relation dieselbe intensionale Bedeutung wie eine bereits existierende, so ist in der Regel der Fälle von einer Modellierung abzuraten. So drückt eine Relation „hat Systembestandteil“ nichts anderes aus, als eine Relation „hat Teil“, extensional finden sich keine Unterschiede. Der Aspekt „hat Systembestandteil“ ergibt sich vielmehr aus dem Zusammenhang. Beziehungen, die sich in ihrer Bedeutung überschneiden, bedingen eine erschwerte Auswertung der Relationen innerhalb einer Ontologie.
- **Axiomsprinzip:** Im Zusammenhang des Umfangs der Relationen spielen Axiome eine wichtige Rolle. Laut Kapitel 2.3 handelt es sich bei Axiomen um Aussagen, die generell als richtig betrachtet werden. Bei der Modellierung von Ontologien können Axiome helfen, die Anzahl der Relationen zu dezimieren. Konzepte, die als allgemein gültig erkannt werden, müssen nicht aus der Ontologie heraus und mithilfe von Relationen definiert werden, sondern können als „gegeben“ in die On-

tologie eingebaut werden. Da zum Zeitpunkt dieser Arbeit die Anzahl der Relationen der Ontologie auf bereits über 250<sup>10</sup> angestiegen ist, sollte aus Gründen der Übersichtlichkeit und der späteren Nutzung in dem Portal über eine Erhöhung der Axiome und eine gleichzeitigen Verminderung der Relationen nachgedacht werden (Als Beispiel sei hier die Beziehung zwischen „Person“ und „Menge von Personen“ angeführt. Der Modellierung zufolge ist eine Person „das Spezifische an“ einer Menge von Personen, während die Menge von Personen „spezifisch hinsichtlich“ Personen ist. Eine solch explizite Modellierung ist nicht vonnöten, da der Fakt, dass eine „Menge von Personen“ aus Personen besteht und mehrere Personen eine „Menge von Personen“ bilden als allgemein gültig und bekannt betrachtet werden kann; Diese Herangehensweise unterstützt auch die Unterscheidung nach extensionalen Gesichtspunkten).

- **Rollenprinzip:** Bei der Frage, ob ein Aspekt der zu modellierenden Relation als Rolle (Unter „K-Infinity“: Erweiterung) modelliert werden soll, muss die Beständigkeit des zu modellierenden Aspektes berücksichtigt werden. Rollen sollten immer dann verwendet werden, wenn Aspekte der Ontologie nur temporär bestehen. Bei dauerhaften Fakten sollte die Ober-/Unterbegriffsrelation bzw. die Individuenbildung gewählt werden.

#### **4.4 Anbindung der Ontologie an ein Webfrontend – Das „Struts Application Framework“**

In diesem Kapitel wird die Anbindung der Ontologie an ein Nutzer-Frontend auf Webbasis erläutert. Die Webapplikation wird dabei mithilfe des Struts-Frameworks realisiert, welches stetig vom Apache Jakarta Projekt weiterentwickelt wird. Um einen Einblick in dieses Framework zu geben, folgt zunächst ein kurzer Überblick über involvierte Technologien, darauffolgend wird deren Zusammenarbeit im Kontext der zugrundeliegenden „Model-View-Controller“-Architektur vorgestellt, und schließlich exemplarisch die Ausführungen der ersten Teile dieses Kapitels durch Skizzierung des technischen Ablaufs einer Suchanfrage am Beispiel des Portals belegt.

---

<sup>10</sup> Anmerkung: Gesprächen mit Modellierern der Firma „intelligent views“ zufolge ist eine Anzahl von über 50 unterschiedlichen Relationen aus Übersichts- und Handhabbarkeitsgründen nicht zu empfehlen. Allerdings beinhaltet die Zahl „250“ bereits die Umkehrrelationen.

#### 4.4.1 Involvierte Technologien

Das Struts-Framework basiert auf einer Vielzahl unterschiedlicher Technologien, die als Hintergrund für die späteren Ausführungen im Folgenden kurz vorgestellt werden.

##### - XML

Die Eigenschaften von Struts stützen sich auf Objekte, welche Konfigurationsdateien nutzen, die in XML<sup>11</sup> geschrieben werden. Auch Java-Applikationen werden gewöhnlich mithilfe einer solchen Datei konfiguriert.

##### - Java

Struts ist in der von Sun Microsystems entwickelten Programmiersprache Java programmiert. Dabei nutzt Struts viele der objektorientierten Eigenschaften von Java ebenso wie die Möglichkeit des Threadings<sup>12</sup>.

##### - JavaBeans

JavaBeans sind in Java geschriebene Softwarekomponenten. Diese Komponenten sind in sich geschlossene, wiederverwertbare Softwareeinheiten, die in weitere Komponenten, Applets, Anwendungen und Servlets integriert werden können. Java Beans unterliegen dabei bestimmten Entwurfsmustern (im Folgenden: Design Patterns), welche die Benennung einzelner Bean-Eigenschaften regeln. Diese Patterns ermöglichen einen standardisierten Zugriff auf die Beans. Dadurch ist es möglich, Beans ohne jegliche Programmierung mithilfe visueller Tools (bspw der „BeanBox“) zu editieren [Voss].

##### - Java Servlets

Java Servlets haben das Ziel, HTTP<sup>13</sup>-Operationen in ein objektorientiertes Modell umzuwandeln. Der Vorteil hiervon ist, dass bei der Entwicklung der Fokus ganz auf Java verlagert werden kann, HTTP muss nicht separat behandelt werden. Für die Funktionalität der Java Servlets muss eine CGI<sup>14</sup>-Erweiterung auf dem Server installiert werden, welche die Kommunikation des Servers mit Servlet-Containern ermöglicht. Der Server leitet eine HTTP-Anfrage an den Servlet Container, dieser erfüllt die Anfrage oder leitet sie an den Server zurück, abhän-

---

<sup>11</sup> XML: Extensible Markup Language

<sup>12</sup> Unter Threading versteht man das gleichzeitige Ausführen mehrerer Tasks.

<sup>13</sup> HTTP: Hypertext Transfer Protocol

<sup>14</sup> CGI: Common Gateway Interface



gig davon, ob er ein auf die Anfrage passendes Servlet bereit hält oder nicht. Die Funktionsweise von Servlets soll anhand des folgenden Beispiels verdeutlicht werden:

- Der Client baut mit einem Browser die Verbindung zum Webserver auf und schickt dabei einen HTTP-Anfrage (im Folgenden: Request).
- Dieser Request wird vom Webserver empfangen und an den Servlet Container weitergereicht.
- Der Servlet Container entscheidet nun auf Grund der Konfiguration seiner Servlets, an welches Servlet der Request weitergeleitet wird. Dabei wird jeweils ein Objekt für den Request und die Antwort (im Folgenden: Response), die später gesendet werden soll, erzeugt. Diese beiden Objekte werden nun dem Servlet übergeben.
- Das Servlet wertet nun die Daten aus dem Request aus. Dabei kann es feststellen, von wem die Anfrage kommt, wie eventuell HTTP POST Parameter gesetzt sind, oder andere relevante Daten. Aus den gewonnenen Daten (in Verbindung mit der Servletfunktionalität) modifiziert das Servlet nun das Response-Objekt.
- Nach der vollständigen Erzeugung des Response-Objekts sorgt der Servlet Container dafür, dass der Response auch vollständig gesendet wird. Anschließend wird die Kontrolle wieder dem Webserver übergeben

#### - **Java Server Pages (JSPs)**

Java Server Pages sind „umgedrehte“<sup>15</sup> Servlets, welche die Entwicklung und Pflege dynamischer Webseiten erleichtern. Anstatt das, was an die HTTP-Response geschickt werden soll, innerhalb einer Java „print“-Anweisung zu schreiben, wird mit JSP alles innerhalb einer Seite direkt an die HTTP-Response übergeben, außer den Teilen, die innerhalb eines Java-Befehls stehen. Unter Zuhilfenahme von JSPs werden normale HTML-Seiten erstellt, welche mithilfe von Java-Befehlen und speziellen JSP-Tags um dynamische Elemente erweitert werden. Die Struts-Distribution enthält verschiedene Tags, die den Zugang zu dem Framework via Java Server Pages vereinfachen.

Haupteigenschaften von JSPs sind dabei die Folgenden:

---

<sup>15</sup> umgedreht insofern, dass JSPs es erlauben, Java-Code in eine HTML-Seite einzubauen, während Java-Servlets die Integration von HTML-Tags in Javacode ermöglichen.

- JSPs sind textbasierte Dokumente, die beschreiben, wie eine Anfrage bearbeitet werden soll und eine Antwort konstruieren.
- JSPs bieten Konstrukte für den Zugriff serverseitiger Objekte und
- Mechanismen für die Definition von Erweiterungen der JSP-Sprache

Innerhalb des Struts-Frameworks arbeiten JSPs oft mit JavaBeans zusammen. Die Anwendung schickt ein JavaBean an die JSP-Seite, welche dieses dazu verwendet, die Ausgabe an die Nutzereingaben anzupassen. Eine exemplarische Darstellung hierzu wird in Kapitel 4.4.3 und im Zuge der Prototypgestaltung in Kapitel 6 angeführt. Ergänzend dazu existieren sogenannte Tag-Libraries (sogenannte „Taglibs“), welche, in JSP eingebunden, die Standardfunktionalitäten der Java Server Pages erweitern. Die Definition von Struts selbst ist über Taglibs realisiert, die den Zugriff auf die Applikation und die Mehrfachnutzung von Code erleichtern. Die Firma „intelligent views“ hat darüber hinaus die Strutsfunktionalität um eine eigene Taglib, „SKAT“ genannt, erweitert, die speziell die Anbindung der Ontologie erleichtert.

Die Struts-Architektur unterstützt die modulare Entwicklung von Webanwendungen. Mit Hilfe sogenannter „Tiles“<sup>16</sup>, die ähnlich einem „include“ in traditionellen Programmiersprachen funktionieren, wird der Code modular gehalten und Redundanz vermieden. Im Gegensatz zu den includes werden Tiles aber nicht nach Herkunft (dem Skript) definiert, sondern über logische Namen angesprochen. Unter Nutzung der sogenannten Tiles-Definition, einer zentralen XML-Konfigurationsdatei, werden die Funktionalitäten gekapselt den Zielpfaden (welche schließlich auf JSP-Dateien verweisen) zugewiesen. Der Vorteil einer solchen logischen Benennung liegt darin, dass JSPs immer den logischen Namen aufrufen können, ohne den Namen der entsprechenden Ziel-JSP zu kennen. Eine leichtere Systemwartung ist der Hauptvorteil einer solchen Realisierung. Abbildung 13 zeigt eine beispielhafte Tile-Definition.

```
<definition name="sl.textQueryForm" path="/textQueryForm.jsp" />
```

Abb. 15: Tile Definition

Das hier definierte Tile trägt den Namen „sl.textQueryForm“. Das sogenannte Mapping, also das Zuordnen der Funktionalität zu der ausführenden JSP-Datei erfolgt mittels dem „path“-Attribut (d.h. die Tile-Funktionalität wird in „text.QueryForm.jsp“ definiert). Tiles un-

---

<sup>16</sup> Tiles: Englisch für „Kacheln“

terliegen dabei Vererbungsregeln. Ist ein Tile ein Subtile eines anderen, wird dies mit dem „extend“-Attribut definiert. Das Subtile erbt dann alle Attribute und Methoden des Supertiles. Ferner ist es möglich, bereits in der Tile-Definition Parameter zu definieren, die das Tile bei Aufruf übergeben bekommt.

#### 4.4.2 Die „Model-View-Controller“-Architektur

Das Struts Framework basiert auf der „Model-View-Controller“-Architektur (MVC), einer allgemeinen Rahmenstruktur für dialogorientierte, ereignisgesteuerte Applikationen, zu denen praktisch alle Anwendungen in modernen graphischen Oberflächen zählen. Dabei werden Funktionalitäten zwischen Objekten, die in der Verarbeitung und Präsentation von Daten involviert sind, aufgeteilt, um die Abhängigkeiten zwischen diesen zu verringern. Die Architektur bildet herkömmliche Anwendungsaufgaben (Eingabe, Verarbeitung und Ausgabe) auf das ereignisgesteuerte Modell ab. MVC teilt dabei Anwendungen strikt in drei Schichten auf: „Model“, „View“ und „Controller“. Jede Schicht behandelt spezifische Aufgaben und hat bestimmte Verantwortlichkeiten im Hinblick auf die jeweils anderen Schichten. Die „Controller“-Schicht bildet dabei den Vermittler, der Anfragen (in der Anwendung: HTTP-Requests), an eine geeignete Steuerkomponente übergibt. Die Steuerkomponenten sind mit einem Modell verbunden und dienen als Zwischenstück zwischen Request und Modell. Das Modell hingegen repräsentiert (oder kapselt) die Anwendungslogik bzw. den Zustand der Anwendung. Abfragen an die „Model“-Komponente werden dann wieder über den „Controller“ zurückgeleitet und zur Darstellung an einen passenden „View“ weitergeleitet. Einen Überblick über die Interaktion zwischen den verschiedenen Komponenten und über deren Aufgabe gibt Abbildung 16.

Das Aufteilen der Zuständigkeiten auf „Model“, „View“ und „Controller“ hilft dabei, Code-redundanz zu vermeiden und vereinfacht gleichzeitig die Anwendungspflege. Darüber hinaus wird die Handhabung der Daten einfacher, da die Verarbeitungslogik von den Daten separiert abläuft. Dadurch wird die Unterstützung neuer Clients stark vereinfacht, weil die Zugriffslogik bei Hinzufügen eines neuen Clienttyps nicht verändert werden muss. Auch die Nutzung externer Datenquellen spielt dabei eine Rolle. Im Zusammenhang des Portals der Fachhochschule wird das insbesondere bei den Überlegungen hinsichtlich einer Integration externer Anwendungen (Intrexx, Pica, HIS) zu beachten sein. Hierbei könnte ein Zugriff auf diese innerhalb der „Controller“-Komponente realisiert und „Views“ speziell für das Betrachten dieser Inhalte definiert werden (siehe hierzu Kapitel 5.2.2: Import von Wissensquellen).

Im Folgenden wird nun die Umsetzung der einzelnen MVC-Komponenten innerhalb des Struts-Frameworks und anhand von Abbildung 16 beschrieben.

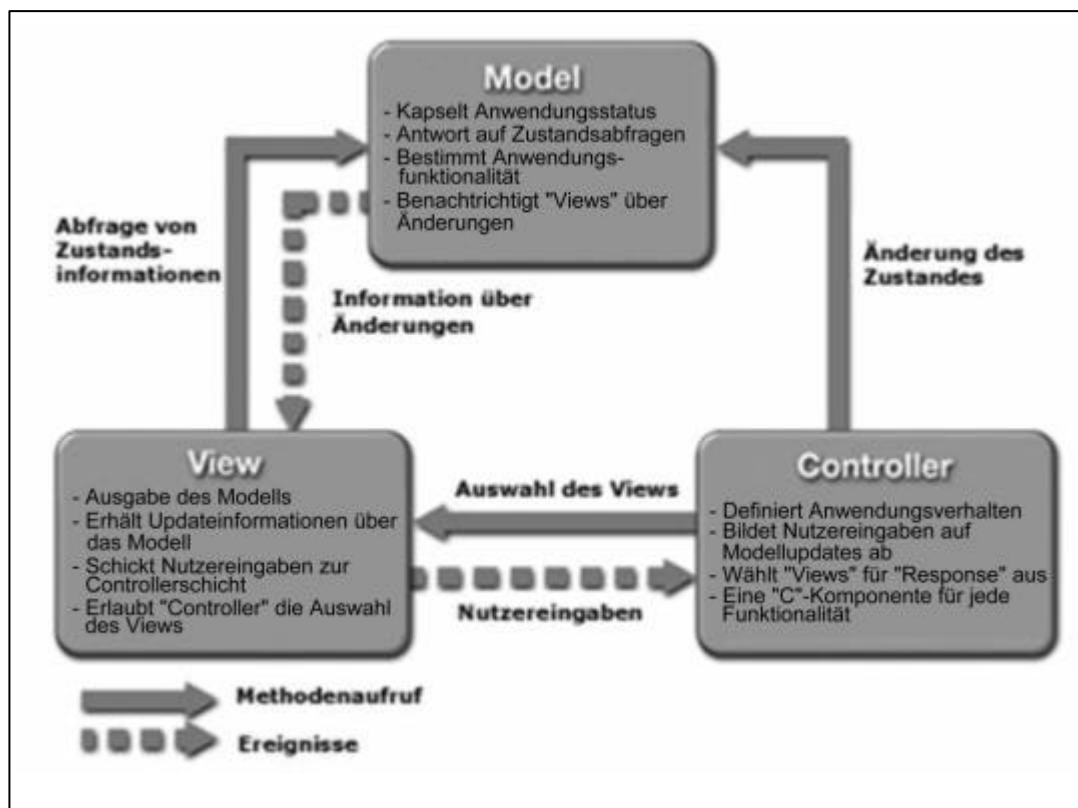


Abb. 16: Die Model-View-Control-Architektur (analog zu [Sunmvc])

### a.) Model: Systemzustand und Anwendungslogik

Die „Model“-Schicht der MVC-Architektur repräsentiert Daten und deren zugehörige Anwendungslogik, die den Zugang und die Modifizierung dieser Daten ermöglicht. Die „Model“-Komponente benachrichtigt die „View“-Komponente, wenn Änderungen am Modell vorgenommen werden und bietet der „View“-Komponente die Möglichkeit, das Modell über seinen Zustand zu befragen. Ferner ermöglicht die „Model“-Schicht der „Controller“-Schicht den Zugriff auf gekapselte Anwendungsfunktionalitäten [Sunmvc].

Dabei kann der „Model“-Teil einer Anwendung generell in zwei große Subsysteme aufgeteilt werden – der interne Zustand des Systems und die Aktionen, die vorgenommen werden können, um diesen Zustand zu verändern. Ausgedrückt in grammatischen Begriffen würden der Zustand als Substantive und die Aktionen als Verben bezeichnet werden.

Im „K-Infinity“ Kontext wird der interne Zustand des Systems in einer Sammlung von Java Beans repräsentiert. Die Bean Eigenschaften speichern dabei Details des Systemzustandes.

Große Anwendungen repräsentieren im Struts-Kontext zumeist die Menge der möglichen Anwendungsoperationen als Bean-Methoden, die aufgerufen werden können. Bei kleineren Anwendungen hingegen sind diese oft in den Action-Klassen integriert, die Teil der Controller-Komponente sind. Nachteil hiervon ist eine eingeschränkte Mehrfachnutzung der Anwendungslogik. Daher ist es üblich, die Anwendungslogik („wie wird etwas gemacht?“) von den Actions („was wird gemacht?“) zu trennen.

### **b.) View**

Unter „View“ versteht man die Ausgabekomponente zur Darstellung der Daten der „Model“-Komponente [Landshut]. Die Datenausgabe wird immer dann geändert, wenn das Modell sich ändert. Ferner gibt „View“ die Nutzereingaben an die „Controller“-Schicht weiter. Im MVC Kontext kann es mehrere Views geben, während die Modellkomponente nur einmal existieren sollte. Im Struts-Framework stützt sich das „View“ auf die oben vorgestellte JSP-Technologie. JSP-Seiten können statischen HTML-(oder XML-) Text (sog. „template text“) enthalten und bieten zusätzlich die Möglichkeit, dynamische Inhalte, basierend auf speziellen Action-Tags einzufügen (in der Anbindung der FHD-Ontologie: SKAT-Actiontags).

### **c.) Controller**

Die „Controller“-Komponente definiert das Verhalten einer Anwendung. Die Behandlung der Nutzereingaben ist ebenso Aufgabe vom „Controller“, wie die Auswahl der „View“-Komponente für die Präsentation. Die „Controller“-Schicht wählt den jeweils nächsten „View“ basierend auf Nutzereingaben und auf dem Ergebnis der Modelloperationen aus. In einer Webanwendung bestehen diese Nutzereingaben aus HTTP GET- und POST-Requests. Eine Anwendung hat üblicherweise eine „Controller“-Komponente für jedes Bündel verwandter Funktionalitäten.

Der „Controller“-Teil einer Anwendung konzentriert sich auf den Empfang von Anfragen der Clients (bspw. eines Webbrowsers), entscheidet, welche Anwendungslogik ausgeführt werden soll und leitet die Ergebnisse an eine „View“-Komponente weiter, welche die nächste Seite des Interfaces abhängig von diesen aufbaut. Innerhalb von Struts besteht die primäre „Controller“-Komponente aus einem Servlet der Klasse „ActionServlet“. Dieses Servlet wird mithilfe eines XML-Konfigurationsfiles konfiguriert, welches sogenannte „Action Mappings“ vornimmt. Ein „Action Mapping“ definiert einen Pfad, der mit der Anfrage-URI<sup>17</sup> verglichen wird und bestimmt den kompletten Klassennamen einer Actionklasse. Alle Actions liegen dabei in einer Hierarchie vor, deren Mutterelement `org.apache.struts.action` ist. Actions kapseln die Anwendungs-

---

<sup>17</sup> URI: Uniform Resource Identifier

logik, interpretieren deren Ergebnis und geben die Kontrolle des Anwendungsflusses schließlich an die geeignete „View“-Komponente weiter, die dann die Antwort generiert. Wie schon im Falle der Tiles, werden mithilfe von Struts logische Namen definiert, die später aufgerufen werden. So kann eine Action-Methode nach der Hauptmenüführung fragen, ohne den richtigen Namen der entsprechenden Action zu kennen. Diese Eigenschaft unterstützt den Entwickler in der Trennung der Kontrolllogik (was ist zu tun?) von der „View“-Logik (Wie soll es aussehen?).

#### 4.4.3 Ablauf einer Suchanfrage

In diesem Unterkapitel wird, um ein besseres Verständnis der Arbeitsweise des Struts-Frameworks zu gewährleisten, der Ablauf einer Suchanfrage skizziert. Die Beschreibung dieses Ablaufes soll an dieser Stelle nur sehr beispielhaft geschehen, eine umfassende Beschreibung findet sich im Anhang. Die Interaktion der MVC Komponenten lässt sich zunächst wie in Abbildung 17 gezeigt zusammenfassen.

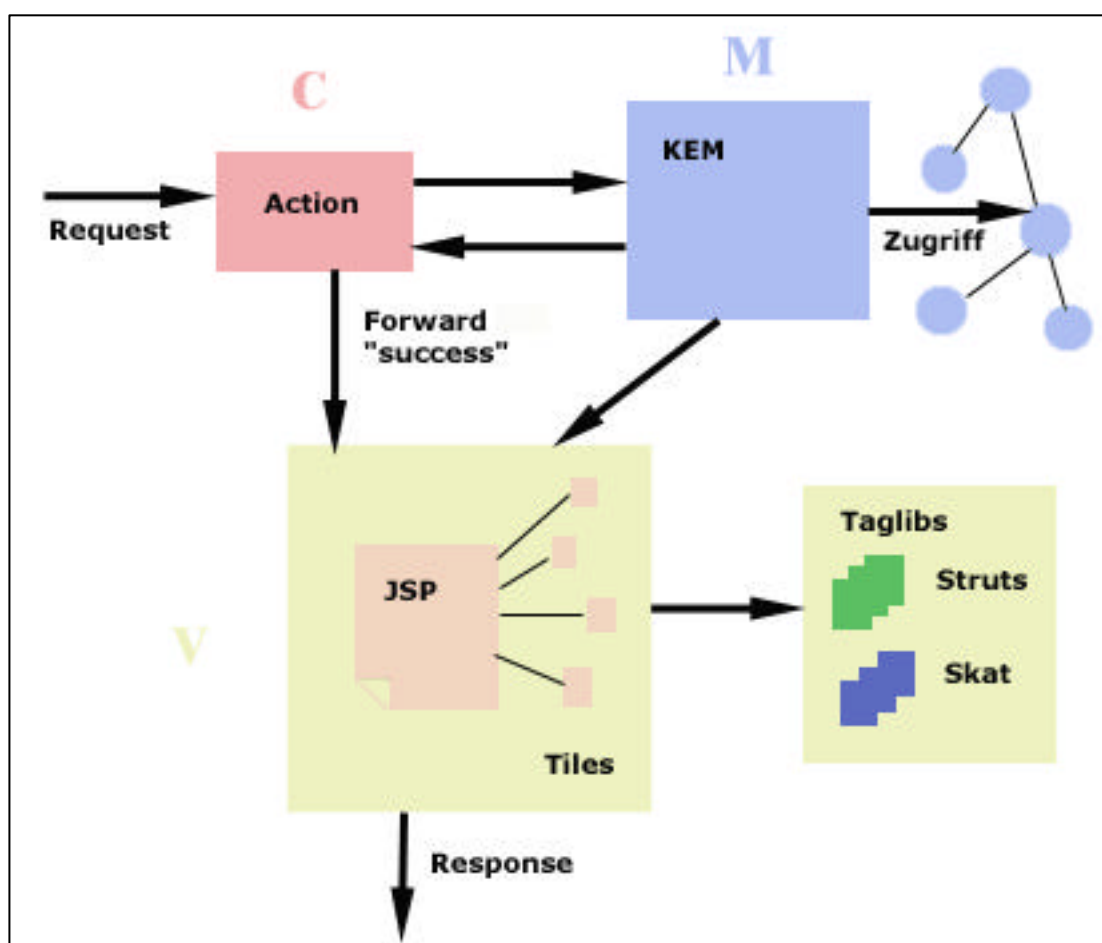


Abb. 17: Ablauf einer Suchanfrage auf dem Portal der FHD

Die HTTP-Anfrage wird direkt an eine Action im „Controller“-Fragment der Anwendung weitergeleitet. Diese Action wird zuvor in einer JSP-Seite der „View“-Komponente definiert. Die Action greift auf die „Model“-Komponente zu, wertet die Ergebnisse aus und gibt abhängig von Ergebnis an einen „View“ zurück, der diese dann, orientierend an der Art des Ergebnisses für den Anwender layoutet. Zur Visualisierung greift die „View“-Komponente dabei auf sogenannte Taglibs zu, welche die Definition der entsprechenden genutzten Funktionalitäten enthalten.

## **5. Konzepte für das Portal der Fachhochschule Darmstadt**

Im folgenden Kapitel werden die Anforderungen und der in Kapitel 3 festgestellte Bedarf der einzelnen Parteien der Fachhochschule Darmstadt in Zusammenhang mit dem methodischen Ansatz aus 2.3.2 gebracht. Zunächst wird die Vorgehensweise im Rahmen des Wissens-Metaprozesses erläutert, der sich im Schwerpunkt mit der Entwicklung und Modellierung der Ontologie beschäftigt. Darauf folgend werden die erhobenen Anforderungen im Zusammenhang mit dem Wissensprozess betrachtet, wobei ein Teil des Fokus auf die Integration bestehender Systeme gelegt wird. Ergebnis der Implementierung des Wissensprozesses werden die konkreten Konzepte für das Portal der FHD sein.

Darüber hinaus wird das Modell um einen weiteren Metaprozess erweitert, der auf allen Ebenen eine gewichtige Rolle bei der Realisation eines Mitarbeiterportals einnimmt, nämlich dem Kommunikationsprozess. Dies wird auf diese Weise gehandhabt, da die Kommunikationskonzepte übergreifend auf die einzelnen Teilprozesse des Wissensprozesses Einfluss nehmen und daher die Definition als Metakonzept am ehesten zutrifft.

In den hinteren Abschnitten dieses Kapitels wird schließlich auf die Profilgestaltung und das zu implementierende Rechtesystem eingegangen und im Hinblick auf Software-Ergonomie werden Grundsätze für die Visualisierung angeschnitten.

### **5.1 *Konzeption innerhalb des Wissens-Metaprozesses***

Der Wissens-Metaprozess spielt in dem aktuellen Stadium der Portalentwicklung keine allzu gewichtige Rolle mehr, da sich die Ontologienentwicklung (die durch den Wissens-Metaprozess gelenkt wird) schon in einem späten Stadium befindet. Doch obwohl Machbarkeitsstudie und Ontologie-Kickoff-Phase bereits durchlaufen sind, sollen im Folgenden die wichtigsten Punkte der Kick-Off-Phase nachgeliefert werden, um ein methodisches Fundament für das System zu liefern. Der Inhalt der Machbarkeitsstudie hingegen kann im Nachhinein nicht mehr behandelt werden, da die entsprechenden strategischen Entscheidungen bereits mit dem Entschluss, das Portal einzuführen, gefallen sind.



### a.) **Ontologie-Kickoff-Phase**

In der Ontologie-Kickoff-Phase sollen laut [Sure et al.] die folgenden Aspekte im Hinblick auf die Ontologienentwicklung berücksichtigt werden:

- **Domäne und Ziel der Ontologie:**

Die Domäne der Ontologie des Portals der FHD ist die Hochschulwelt im Allgemeinen. Dazu gehört zunächst die interne Welt der Fachhochschule Darmstadt, aber auch ein um Indikatoren, die Einfluss auf die Hochschule nehmen, erweiterter Welt-ausschnitt. Letzten Endes muss jeder einzelne Vorgang, der in der Fachhochschule stattfindet, durch die Ontologie darstellbar und mithilfe der Ontologie semantisch auffindbar sein.

- **Design-Richtlinien**

Design-Richtlinien dienen der Unterstützung der Modellierer. In Kapitel 4.3 wurde bereits eine Methodologie zur Evaluierung der Ontologie vorgestellt. Die dort behandelten Kriterien können auch als Richtlinien zum Ontologiedesign hinzugezogen werden. Darüber hinaus gibt es einige Notationen, welche die Benennung von Wissensobjekten in der Ontologie regeln. Im Fall der Modellierung der FH-Ontologie gelten dabei folgende Richtlinien:

- **Relationen:**

hat\_Relation\_zu:

Definiert eine Beziehung zwischen zwei Begriffen

hat Relation zu:

Definiert eine Beziehung zwischen zwei Instanzen

hat Relation zu (IB):

Definiert Relation zwischen Instanz und Begriff

hat\_Relation\_zu (BI):

Definiert Relation zwischen Begriff und Instanz

Im Allgemeinen ist es ratsam, aus Unterscheidungsgründen gegenüber Konzepten die Relationsbezeichnungen klein zu schreiben.

- **Konzepte**

Konzepte werden groß geschrieben. Insgesamt gilt es, in der Hierarchie keine Synonyme mit Oberbegriffen zu belegen. Namen sollten sprechend gewählt, kurze, aber eindeutige Bezeichner vorgezogen werden.

- **Personen** (Konzepte)

Objekte vom Typ Person werden in der Form „Nachname, Vorname“ benannt.

- Wissensquellen zur Ontologienentwicklung

Der Umfang der Zielontologie wurde durch das Einsatzgebiet (FH Portal) und das Ziel (Abbildung der gesamten Hochschulwelt) vorgegeben. Als Wissensquellen dienten beim Ontologiedesign vornehmlich das Hessische Hochschulgesetz (HHG), Internetquellen wie wissen.de, xipolis.de sowie Nachschlagewerke wie „Duden“ und „Brockhaus“. Ferner wurde domänenspezifisches Wissen und Informationen über Personen aus einschlägigen Webseiten (die der Fachbereiche und Teilinstitutionen) und in Einzelfällen durch persönliche Gespräche von den Modellierern extrahiert.

- Nutzungsszenarien / Use Cases

Innerhalb der Prototypgestaltung gilt es einerseits, die verschiedenen Arten wie auf die Ontologie zugegriffen wird, zu unterscheiden und andererseits diese anhand eines konkreten Use Cases zu verdeutlichen.

Bei der Bestimmung der Nutzungsszenarien gilt es zunächst folgende Gruppen zu unterscheiden<sup>18</sup>:

- a.) Professoren
- b.) Mitarbeiter
- c.) Studentische Gremienmitglieder

Innerhalb der einzelnen Gruppen gilt es ferner, feinere Unterscheidungen zu treffen. So ist es nötig, im Beispiel der Professoren, die Fachbereichszugehörigkeit und die Gremienfunktionen zu berücksichtigen, da sich daraus ein unterschiedlicher Informationsbedarf ergibt.

Bei der grundsätzlichen Art der Zugriffe auf das Systems ergeben sich allerdings kaum Unterschiede zwischen den einzelnen Gruppen. Die Nutzungsszenarien (also die Frage: Auf welche Art und Weise greifen die Nutzer auf die Ontologie zu) gliedern sich dabei wie folgt:

---

<sup>18</sup> Die Nutzergruppen bestimmen dabei nicht die **Nutzungsszenarien**, vielmehr ergeben diese sich aus der Art und Weise des Zugriffs. Daher wurde der Begriff „**Nutzerszenarien**“ aufgrund möglicher Irreführungen in diesem Zusammenhang bewusst vermieden.

- i.) Passive Informationsversorgung durch sogenannte Push-Dienste:  
Der Nutzer bekommt anfangs ein bestimmtes Interessenprofil zugewiesen und wird anhand dieses Profils automatisch mit den relevantesten Informationen versorgt.
- ii.) Aktive Informationsversorgung:  
Der Nutzer hat einen weiteren Informationsbedarf, den er durch Suche im System befriedigen soll. Unterschiedliche Art und Weisen der Suche spielen dabei eine gewichtige Rolle. So muss bspw. unterschieden werden, unter welcher Zielsetzung der Nutzer auf das System und die dahinter liegende Ontologie zugreift. Es besteht ein Unterschied, ob der Nutzer das System zur Befriedigung seiner eigenen Informationsbedürfnisse nutzt oder im Auftrag eines Kollegen / Vorgesetzten recherchiert. Um die unterschiedlichen Perspektiven und unterschiedliche Erfahrungslevel dabei zu berücksichtigen, müssen vielfältige Aspekte des Zugriffs beachtet werden (siehe Kapitel 5.4).
- iii.) Informationsanreicherung:  
Der Nutzer pflegt selbständig Informationen in das System ein. Hierzu müssen ihm die notwendigen Rahmenbedingungen zur Verfügung gestellt werden.
- iv.) Pflege des Begriffssystems  
Auch hier wird pflegend auf das Netz zugegriffen. Allerdings ergibt sich dabei ein Unterschied bezüglich des Akteurs. Unter c.) sind Personen gemeint, die als „normale“ Nutzer auch Rechte zum Einpflegen von Informationen haben. Bei der Pflege des Begriffssystems handelt es sich um einen Zugriff eines Administrators (oder Knowledge Engineers).

Um für den Prototyp des Portals einen reichhaltigen Anwendungsfall ausarbeiten zu können, wurde als repräsentative Nutzerin eine Mitarbeiterin des Zentrums für Forschung und Entwicklung ausgewählt. Innerhalb dieses Nutzerbereichs wird es Aufgabe sein, einen Use-Case zu entwickeln, der es ermöglicht, alle oben genannten Nutzungsszenarien prototypisch aufzuzeigen. Dies bedeutet, dass die passive Informationsversorgung, die aktive Informationsversorgung und die Informationsanreicherung innerhalb dieses Use-Cases durchgeführt werden sollen.

- **Kompetenzfragen**

Zur späteren Evaluierung der Ontologie ist es nach [Uschold et al.] ratsam, Kompetenzfragen auszuarbeiten, die später bei der Überprüfung der Ontologie Aufschluss darüber geben, ob die Ontologie der Aufgabenstellung gerecht wird. Das Bilden von Kompetenzfragen wird innerhalb spezieller Anwendungsszenarien vonstatten gehen und wird exemplarisch innerhalb der Prototypentwicklung demonstriert werden. Im Gegensatz zu der unter 4.3 entwickelten Methodologie, welche den Fixpunkt auf formelle Korrektheit der Ontologie legt, wird mit dem Einsatz von Kompetenzfragen die inhaltliche Zielorientierung geprüft. Im Zuge der weiteren Portalentwicklung jenseits dieser Arbeit wird es häufiger nötig sein, Kompetenzfragen für die verschiedensten Bereiche, die durch die Ontologie abgedeckt werden sollen, zu entwickeln, um die Nutzbarkeit der Ontologie auf dem Zielsystem zu maximieren.

**b.) Verfeinerungsphase**

In der Verfeinerungsphase wird die Taxonomie schrittweise erweitert. Die Verfeinerungsphase wechselt sich dabei solange mit der Bewertungsphase ab, bis der gewünschte Detaillierungsgrad und die nötige Präzision der Ontologie erreicht ist. Die Ontologie des FHD Intranet-Portals befindet sich derzeit zwischen der Verfeinerungs- und Bewertungsphase.

**c.) Bewertungsphase**

In der Bewertungsphase wird die Ontologie auf ihre Tauglichkeit hin überprüft. Dabei wird zunächst mithilfe eines studentischen Mitarbeiters die Ontologie schrittweise auf Gültigkeit und Detaillierungsgrad unter Anderem anhand der unter 4.3 erarbeiteten Methodologie überprüft. Zu einem späteren Zeitpunkt muss das Stadium der Ontologie ferner mithilfe von festzulegenden Kompetenzfragen auf Anwendungen hin getestet werden.

## **5.2 *Konzeption innerhalb des Wissensprozesses***

Die eigentlichen Konzepte des ontologiebasierten Portals werden im Kontext des Wissensprozesses erstellt. Dabei orientiert sich die Konzeption nicht strikt an der unter 2.3.2 vorgestellten Struktur. Vielmehr muss im Zusammenhang des FHD-Portals der Prozess angepasst und um einen weiteren Teilprozess erweitert werden. So ergeben sich, orientierend an den Bausteinen des Wissensmanagements nach [Probst et al.] und an der in 2.3.2 vorgestellten Methodik für ontologiebasiertes Wissensmanagement, folgende Teilprozesse des Wissensprozesses:

- Wissensgenerierung
- Wissensimport
- Wissenserfassung
- Wissenszugriff / Wissensnutzung

Wie zu sehen, werden die Prozesse des Wissenszugriffs und der Wissensnutzung aufgrund ihrer Nähe nicht getrennt, sondern werden vielmehr im gegenseitigen Kontext betrachtet.

Die einzelnen Konzepte ergeben sich aus den durchgeführten Bedarfserhebungen, angereichert durch Überlegungen und Vorschläge bereits existierender Intranetportale. Die Konzepte wurden ferner im Rahmen eines Portal-Kickoff-Meetings mit Teilnehmern aus allen relevanten und beteiligten Teilinstitutionen diskutiert und validiert.

Um eine präzise Betrachtungsweise der Ansätze zu gewährleisten, soll nochmals kurz auf die terminologischen Definitionen aus Kapitel 2.1.1 (→ die Wissenstreppe, siehe Abbildung 1) eingegangen werden. Demzufolge handelt es sich bei Wissen um Informationen, die zweckdienlich miteinander vernetzt werden. Bei Vergegenwärtigung dieser Definition wird klar, dass es sich bei dem Portalsystem der Fachhochschule Darmstadt, im Gegensatz zu vielen anderen gängigen Systemen, um ein tatsächliches Wissensmanagementsystem handelt. Wissen wird durch Vernetzung vorhandener Informationen und den Kontext der Ontologie generiert und verfügbar gemacht. Von Informationen wird in den folgenden Unterkapiteln hingegen immer dann die Rede sein, wenn es sich um isolierte Inseln handelt (also bspw. spricht man von Informationsbedarf des Nutzers, da der Transfer zu Wissen, also das Einordnen in den eigenen Kontext, auf der Nutzerseite erst nach erfolgter Extraktion aus dem System vonstatten geht). Diese Betrachtungsweise bestätigt auch die kontextabhängige Definition von Wissen aus Kapitel 2.1.1.

### **5.2.1 Konzepte für die Wissensgenerierung**

Eine der Hauptaufgaben des FHD Portals ist die Unterstützung der Generierung von Wissen. Hierbei gilt es, unterschiedliche Arten von Wissen zu betrachten: Einerseits das Explizite und andererseits das Implizite. Unter explizitem Wissen versteht man dabei das fest in der Firma (mit Dokumenten, Datenbanken, Notizen, Intranetseiten etc. als Träger) vorhandene, während implizites Wissen flüchtig und zumeist durch den Mitarbeiter als Träger existiert [Lindvall et al.] oder erst durch Auswertung sichtbar wird. Bei der Erzeugung von explizitem Wissen geht es vornehmlich darum, explizit vorhandene Informatio-

nen zu strukturieren und einheitlich in das System zu integrieren. Im Anwendungsbeispiel der Fachhochschule ist es erforderlich, Masken zu entwerfen, welche die Erzeugung expliziten Wissens durch Integration expliziter Information (bspw. in Form von Anträgen, Protokollen, Prozessbeschreibungen) in das Wissensnetz ermöglichen. Im Zuge der Prototypisierung wird die Erzeugung expliziten Wissens anhand eines Use Cases des Zentrums für Forschung und Entwicklung aufgezeigt werden. Anhand des Einpflegens eines neuen Forschungsprojektes, sowie der Beantragung von Fördergeldern für dieses Projekt, sollen so die Möglichkeiten der dezentralen Dokumentenpflege im Zuge eines Knowledge Sharings auf Mitarbeiterbasis aufgezeigt werden. Einhergehend mit dieser dezentralen Dokumentenpflege ist ein Optimierungsvorgang auf der Ebene des Qualitätsmanagements. Durch angemessene und durchdachte Standardisierung der Formulare wird die Wissensgenerierung nicht nur ermöglicht, sondern Wissen auf einem qualitativ hochwertigen Level generiert.

Im Hinblick auf implizites Wissen ist der Fokus ein anderer. Wichtig hierbei ist die **Verfügbarmachung** impliziten Wissens, was wiederum die Generierung expliziten Wissens zur Folge hat. Mithilfe der Unterstützung der zugrundeliegenden Ontologie geschieht die Generierung dabei teilweise komplett automatisch. Im Beispiel einer Expertensuche soll dieses Phänomen verdeutlicht werden:

So kann bei einer Suche nach einer Person mit Kenntnissen über einen bestimmten Sachverhalt implizites Wissen direkt genutzt werden. Man gehe von einer Suche nach einer Person aus, die über Kenntnisse des Gebietes „Informationswissenschaft“ verfügt. Da es aus Gründen der Formalität kein Attribut „kennt sich aus mit“ gibt (selbst wenn es eines gäbe, wäre es fast unmöglich, dieses so zu pflegen, dass sämtliche Wissensträger erfasst werden), müssen dazu Querverbindungen genutzt werden. In diesem Beispiel wird die Verbindung zwischen einem Dokument und dessen Urheber benutzt. Hat ein Dokument das Thema „Informationswissenschaft“, ist davon auszugehen, dass der Verfasser des Dokumentes über Kenntnisse auf diesem Gebiet verfügt. Diese Inferenz (vgl. Kapitel 2.3) wird von „K-Infinity“ unterstützt. So ist es möglich, mithilfe der Suchfunktionalität im K-Builder, feste Suchanfragen zu formulieren, welche solches implizit im Netz vorhandenes Wissen sichtbar machen. Im Zuge der Prototypisierung wird ein Beispiel für solch eine Inferenz realisiert werden, im Unterkapitel 5.2.4 die Formulierung einer solchen Suche im Knowledge Builder aufgeführt.

Neben dem im Unternehmensmodell vorhandenen, aber nicht sichtbaren Wissen, gibt es darüber hinaus noch den häufigen Fall, dass Wissen, welches implizit vorhanden ist,

nicht durch Inferenz sichtbar gemacht werden kann, da jegliche Anhaltspunkte für das Vorhandensein fehlen. Dieses in Kapitel 2.1.1 als „Handlungswissen“ bezeichnete Wissen existiert nur in Verbindung mit den Mitarbeitern als Träger. Es entsteht durch die tägliche Arbeit oder durch Ausüben von Nebenbeschäftigungen und Freizeitaktivitäten des Mitarbeiters und wird in der Regel nicht für die Organisation sichtbar. Lösungsansätze für dieses Dilemma finden sich vornehmlich im kommunikativen Bereich und werden daher in Unterkapitel 5.3 beschrieben.

Der letzte Aspekt der Unterstützung der Wissensgenerierung durch das Intranetportal der FHD ist die Pflege des Begriffsnetzes. Durch zentrale Pflege der Terminologien und der Begriffstaxonomie wird stetig neues Wissen generiert und in den Organisationskontext eingebracht. Dieser Aspekt spielt ferner im Kontext der Wissenserfassung im Unterkapitel 5.2.3 eine Rolle. Auch wird durch die Pflege externer Quellen, auf welche die Ontologie (vor allem durch Hyperlinks) verweist, stetig neues Wissen generiert und in die Organisation eingebracht.

## 5.2.2 Konzepte für den Wissensimport und die Integration vorhandener Systeme

Bei der Konzeption des Portals der Fachhochschule Darmstadt gilt es, verschiedene vorhandene, systembasierte Datenquellen in das Portal einzubinden und in den Kontext der Ontologie zu setzen. Die für das Portal zu berücksichtigenden Systeme sind dabei die Folgenden:

- **Intranet:** Das Intranet der Fachhochschule Darmstadt befindet sich derzeit noch in einem sehr frühen Stadium. Basierend auf der Softwareumgebung „Intrex“, wird es Funktionalitäten wie Terminverwaltung, Dokumentenmanagement, Kalenderfunktionalitäten, Newsletterfunktionen, To-Do-Listen, Schwarzes Brett, Marktplätze und Foren bieten. Rückgrad der Anwendung ist dabei eine MS-SQL-Server-Datenbank. Teile dieser Funktionalitäten werden durch die Entwicklung des FHD-Portals abgelöst werden (bspw. das Dokumentenmanagement), andere auf verschiedene Art und Weise (siehe unten!) integriert werden. Zum Zeitpunkt dieser Arbeit kann das Intranet aufgrund seines frühen Entwicklungsstadiums nur exemplarisch in den Prototypen mit einfließen.
- **Hochschulinformationssystem:** Das Hochschulinformationssystem (HIS) ist eine modulare Standardanwendung, die speziell für Hochschulen entwickelt wurde. An der Fachhochschule Darmstadt wurden die Teilmodule HIS-POS, HIS-SOS

und HIS-ZUL eingeführt. HIS-POS dient dabei der Prüfungs-, HIS-SOS der Studenten- und HIS-ZUL der Zulassungsverwaltung. In ihrer Funktionalität umfassen die Module unter Anderem die Erfassung von Prüfungsordnungen, die Durchführung von Prüfungsanmeldungen, das Verbuchen von Noten, die Überprüfung von Zulassungsvoraussetzungen, die Berechnung von Noten, Frist- und Terminüberwachung und den Abruf von Notenlisten. Teile der Funktionalitäten werden dabei dem Studenten browserbasiert zur Verfügung gestellt.

- **Blackboard Learning System:** Das Blackboard Learning System ist ein webbasiertes System zur Verwaltung von Lehrveranstaltungen und dient der zentralen Organisation von Vorlesungen an der Fachhochschule. Dozenten haben die Möglichkeit, Vorlesungsmaterialien zentral für die Studenten zugänglich zu machen und Nachrichten automatisch an Kursteilnehmer zu versenden. Studenten bietet das Blackboard Learning System ein zentrales Verwaltungstool für Termine, Kurse, Noten und Nachrichten.
- **PICA-Webopac:** Der Opac der Bibliothek der Fachhochschule Darmstadt dient der Bestellung und Ausleihe von Büchern an den einzelnen Bibliotheksstandorten der Fachhochschule.
- **Helpdesk:** Im Zuge einer parallel laufenden Diplomarbeit entsteht derzeit ein Helpdesk, der für verschiedene Bereiche der Fachhochschule genutzt werden kann. Die Umsetzung dieses Helpdesks geschieht dabei innerhalb des Intranets. Dieses System kann im Zuge dieser Arbeit nicht berücksichtigt werden, sollte aber zu einem späteren Zeitpunkt mit einfließen.
- **SAP/R3:** Die betriebliche Standardsoftware SAP, die seit einiger Zeit an der Fachhochschule Darmstadt in der Version R3 zum Einsatz kommt, beinhaltet u.a. viele wertvolle Informationen über die Organisationsstruktur der Fachhochschule. Diese Daten gilt es, bestmöglich in die Portalanwendung zu integrieren.
- **Fachbereichshomepages:** Auch wenn die Fachbereichshomepages keine Anwendungen im herkömmlichen Sinne sind, müssen sie als Behälter wichtiger Informationen der Fachhochschule im Portal der FHD berücksichtigt werden. Das Wissen über die einzelnen Fachbereiche ist in geballter Form über deren Internetauftritt zu finden und muss im System sichtbar gemacht werden.

Im Folgenden soll nun ein Überblick über die verschiedenen Ansätze zur Wissensintegration gegeben werden. Dabei werden die eben aufgeführten Systeme konzeptionell den jeweiligen Ansätzen zugeordnet.



Unabhängig von den konkreten Anwendungen lassen sich verallgemeinernd vier unterschiedliche Arten der Wissensintegration unterscheiden:

#### **a.) Integration durch Extraktion hochschulinterner Quellen**

Beim Aufbau der Ontologie unter „K-Infinity“ wurden diverse Datenquellen der FHD direkt genutzt und Wissensobjekte anhand dieser Quellen modelliert. Auch bei der Weiterentwicklung der Ontologie werden diese Quellen eine wesentliche Rolle spielen, da Informationen und Wissen über hochschulinterne Belange in diesen schon vorhanden sind. Zu solchen internen Quellen gehören vor allem Abteilungs- und Fachbereichshomepages. Durch manuelle Extraktion verfügbar gemacht, bilden die dort enthaltenen Informationen eine breite Basis für die inhaltliche Implementierung des Portals. Da es sich bei diesem Vorgang um eine Verbesserung der Wissensqualität handelt (weitgehend semantisch unstrukturierte Information wird in den Gesamtkontext des Modells eingeordnet und so zu modelliertem Wissen), ist eine weitere Integration der Homepages in das Portalsystem nicht vonnöten. Verweise auf die Webseiten der Fachbereiche und organisatorischen Einheiten (Abteilungen, Teilinstitutionen wie das „Zentrum für Forschung und Entwicklung“ etc.) werden in den Attributswerten der Fachbereichsinstanzen definiert sein.

Eine weitere gewichtige Rolle bei der Betrachtung der Integration von Wissen spielt der Import von Daten aus der betrieblichen Standardsoftware SAP. Vornehmlich personenbezogene Daten werden im Zuge dessen in die Ontologie des FHD-Portals integriert. Die technische Realisation des Imports bzw. der Synchronisation wird dabei von der Firma „intelligent views“ übernommen.

#### **b.) Verlinkung zu hochschulinternen Quellen**

Eine weitere Möglichkeit der Integration bietet die Verlinkung der bereits vorhandenen Quellen. Diese Verlinkung geschieht auf der untersten Ebene zunächst nicht innerhalb des Datenmodells, sondern auf den eigentlichen Portalseiten. Nachteil einer solchen Integration ist eine gewisse Statik. Daher ist diese nur bei stetigen Quellen sinnvoll, die keiner allzu häufigen Änderung unterliegen. Darüber hinaus muss das Verlinkungsziel sich inhaltlich für eine feste Integration in die Portalmenüführung eignen.

Im Prototyp wird anhand der Kalenderfunktion, die im Intranetsystem Intrexx sehr detailliert implementiert ist, diese Art der Wissensintegration aufgezeigt werden. Es ist geplant, einen Kalender, der das aktuelle Datum anzeigt, in das Portal zu integrieren, der als Hyperlink bei Aktivierung einen Link auf das Intrexx-System öffnet und die dort vorhandene Terminfunktion aufruft. Auf diese Art könnte ein Zugriff auf alle hochschulinternen Termine gegeben werden.

Das Blackboard-System sollte zu Anfang auf dieselbe Art und Weise integriert werden.

Grundsätzlich gilt: Je mehr unterschiedliche Funktionalitäten eines Systems genutzt werden sollen, desto schlechter ist eine derartige Integration in das Portal.

Da es sich bei dieser Art der Verlinkung um keine Integration der Informationen **in** das System handelt, sondern nur um einen Aufruf (was zur Folge hat, dass die Funktionalität im Wissensmodell nicht auftaucht), sollte diese Art der Anbindung längerfristig immer in Verbindung mit Punkt c.) geschehen.

### **c.) Modellierung der vorhandenen Quellen als Wissensobjekte**

Bei Systemen mit vielseitigen Anwendungen, die inhaltlich das Portal erweitern, ist die Gestaltung als Wissensobjekt die zu wählende Variante. Dabei wird ein Objekt im Wissensnetz integriert, welches das System repräsentiert und dessen Relationen die Funktionalitäten des Systems ableitbar machen. Die Ontologie soll immer dann auf dieses Teilmodul verweisen, wenn nach einer dort implementierten Funktionalität gesucht wird.

### **d.) Systemgesteuerter Zugriff auf externe Systeme**

Der systemgesteuerte Zugriff auf externe Systeme ist die qualitativ hochwertigste Variante, die mithilfe des Portals unter „K-Infinity“ realisiert werden kann. Unter dieser Variante der Wissensintegration ist der on-demand Zugriff des Portalsystems auf ein externes System zu verstehen.

Dabei gibt es zwei Varianten des Zugriffs, die im Zusammenhang des Struts-Frameworks (siehe Kapitel 4.4) und anhand eines Beispielzugriffs auf Intrexx erläutert werden sollen. Wie in Unterkapitel 4.4 dargestellt, arbeitet das Struts-Framework nach dem Model-View-Controller-Prinzip. Zur Erinnerung sei nochmals auf Abbildung 17 verwiesen, welche den typischen Ablauf einer Suchanfrage darstellt.

Die „einfachere“ Variante dieser Art des Wissenszugriffs erklärt sich dabei wie folgt: Ge setzt sei der Fall, dass innerhalb der Portaloberfläche ein Menüpunkt „Ressourcenverwaltung“ implementiert ist. Bei Anwählen dieses Menüpunktes wäre es wünschenswert, nicht auf das Datenmodell von „K-Infinity“ (=die Ontologie) zuzugreifen, sondern die Funktionalität direkt aus Intrexx heraus aufzurufen. Hierzu müsste die HTTP-Request an eine Action weitergeleitet werden, die nicht auf das Datenmodell, sondern auf Intrexx zugreift. Dabei gibt es zwei Möglichkeiten: Einerseits könnte über Speicherung der Intrexx-Nutzerdaten in den Nutzerprofilen und Durchschleifen dieser Daten zum Intrexx-Login-Bereich direkt auf die Anwendung zugegriffen werden. Die zweite Möglichkeit nutzt das dem Intrexxsystem zugrundeliegende Datenmodell: Eine MS-SQL-Server-Datenbank.

Durch direkten Zugriff der Action mittels JDBC<sup>19</sup> könnte die Anwendungsschnittstelle umgangen und die notwendigen Daten direkt aus der Datenbank extrahiert werden. Die Daten würden zurück an die Action gegeben, der korrespondierende View ausgewählt und schließlich die Daten an die HTTP-Response übergeben und dem Nutzer dargestellt.

Die „schwerere“, aber flexiblere Variante verknüpft die eben genannten Aspekte mit der Wissensintegration aus Punkt c.). Das bedeutet, dass, wie unter c.) beschrieben, ein Wissensobjekt in der Ontologie modelliert wird, das Portal aber automatisch bei einer Anfrage an dieses Wissensobjekt auf Selbiges zugreift und die notwendigen Informationen extrahiert. Abbildung 18 zeigt nun den Vorgang aus Abbildung 17 in etwas abgeänderter Form. Um ein besseres Verständnis zu gewährleisten, wurden nun die einzelnen Vorgänge durchnummeriert.

---

<sup>19</sup> JDBC: Java DataBase Connectivity

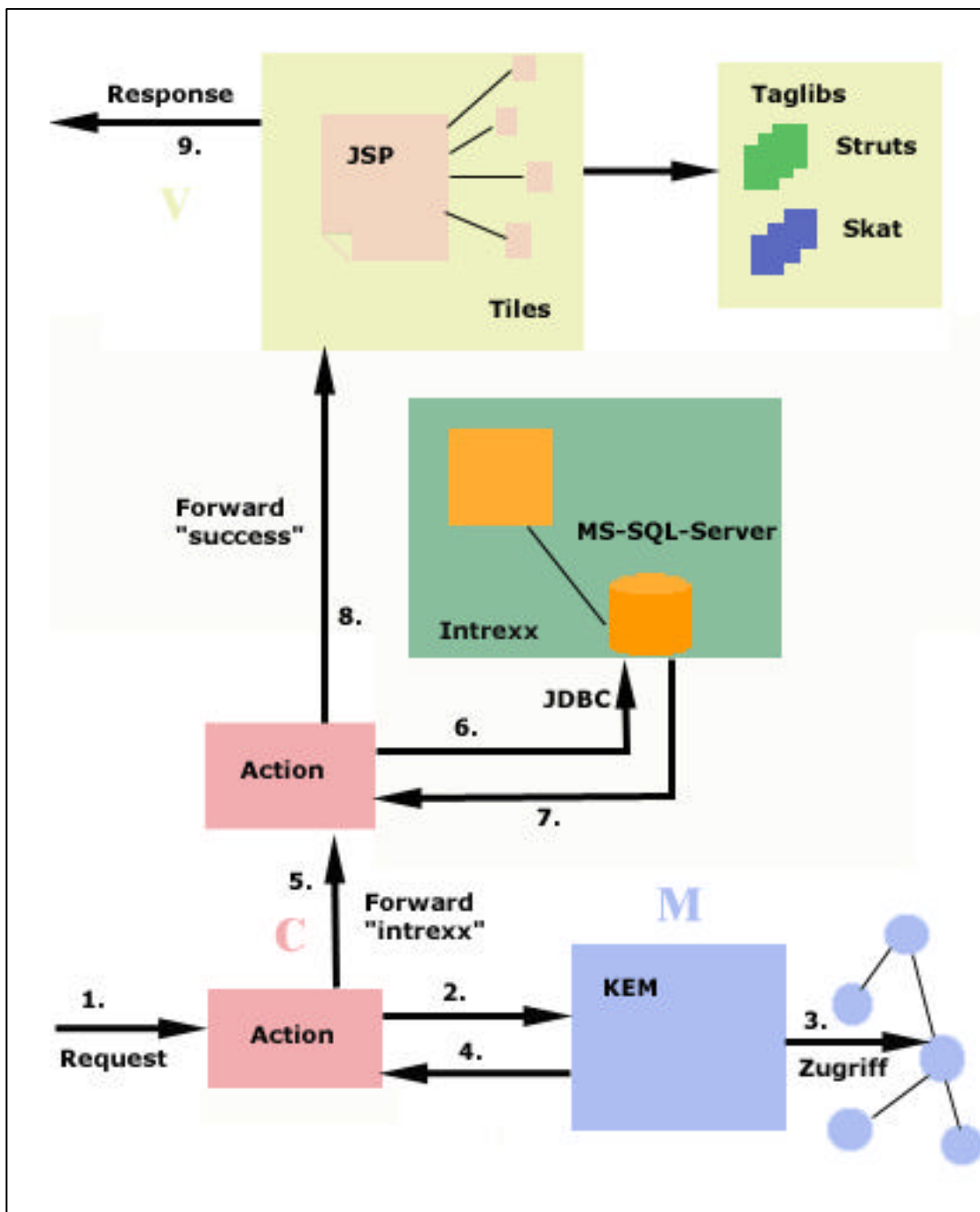


Abb. 18: Die Integration von Intrexx

Wie zu sehen, erreicht eine Nutzeranfrage per HTTP die zugehörige Action (Vorgang 1). Diese Action ist derart implementiert, dass sie auf das Datenmodell zugreift, um der Anfrage entsprechende Informationen aus der Ontologie zu ziehen (Vorgänge 2 bis 4). Nun sei der Fall gesetzt, dass die Suchanfrage „Ressourcenverwaltung“ lautet. Die Action, die einer Nutzeranfrage zugeschrieben („gemappt“) wird, greift auf das Modell zu und findet ein Wissensobjekt namens „Intrexx“, welches unter anderem eine Relation zu einem Wissensobjekt „Ressourcenverwaltung“ besitzt. Diese Information wird in ein JavaBean geschrieben und an die Action zurückgegeben. Diese liest das Bean aus. Knackpunkt

des Vorgangs ist nun, dass die Action nicht derart implementiert ist, dass sie direkt an einen view weitergibt, sondern mithilfe eines sog. „forwards“, das auf das Intrexxsystem verweist, auf dieses zugreifen kann (Vorgang 5)<sup>20</sup>. Die hinter dem Intrexxsystem liegenden Daten sind, wie oben bereits erwähnt in einer MS-SQL-Server-Datenbank gespeichert. Ab diesem Punkt gibt es, wie bei der ersten Zugriffsvariante, zwei Möglichkeiten: Einerseits könnte die Action über die Loginschnittstelle mittels im Portalprofil des Nutzers gespeicherter Logindaten auf die Funktionalität zugreifen. Diese Variante ist die längerfristig zu empfehlende, da die unter Intrexx vorhandene Infrastruktur direkt in das Portal übernommen werden kann. In Abbildung 18 ist eine vereinfachte Variante dargestellt, die Action greift in diesem Falle nicht auf die eigentliche Anwendung zu, sondern extrahiert die Daten mittels JDBC direkt aus der MS-SQL-Server Datenbank (Vorgang 6). Durch den standardisierten JDBC-Zugriff ist diese Möglichkeit leichter zu implementieren, unter anwendungsmodellierenden Gesichtspunkten aber nicht ganz unkritisch (da die Daten nochmals komplett neu verarbeitet werden müssten).

Bei beiden Varianten wird das Ergebnis an die Action zurückgegeben. Diese wählt den entsprechenden View aus und gibt die erhaltenen Informationen in Form eines Java-Beans an diesen (Vorgang 8), der sie den JSPs entsprechend visualisiert und in ein Response schreibt (Vorgang 9).

Eine solche Realisierung der Wissensintegration wird innerhalb des Prototyps nicht umgesetzt werden, da die eigentliche Action in Java neu programmiert werden müsste und dies mehr Ressourcen in Anspruch nehmen würde, als im Rahmen dieser Arbeit zur Verfügung stehen. Für eine spätere Weiterentwicklung des Prototyps empfiehlt es sich, diesen Aspekt ausgegliedert (bspw. im Fachbereich Informatik) behandeln zu lassen.

### 5.2.3 Konzepte für die Wissenserfassung

Für gewöhnlich ist es nicht ausreichend, Wissen im System verfügbar zu haben, vielmehr muss das Wissen inhaltlich erfasst werden, um es geeignet abzulegen, wieder auffinden und im Zusammenhang verschiedener Kontexte betrachten zu können. Auf ontologischer Ebene bedeutet dies die terminologische Integration in die Ontologie und die sinnvolle Verlinkung mit anderen Inhalten. Vor allem unvollständige Informationen, wie in Dokumenten als Freitext vorhandene und für den Rechner unverständliche, müssen derart aufbereitet werden, dass sie in das System integriert werden können und auffindbar sind. Mit Hilfe von entsprechenden Annotationstools können Freitexte auf Basis der Ontologie in Wissensobjekte umgewandelt werden, indem Textpassagen als Attribute von

---

<sup>20</sup> Die genaue Funktionsweise ist dem Abschnitt „Ablauf einer Suchanfrage“ im Anhang zu entnehmen.

Konzepten bzw. als Instanzen beschrieben und Textaussagen als Relationen dieser Wissensobjekte ausgedrückt werden. Bei der Einführung des FHD-Portals spielen solche Überlegungen zunächst noch keine Rolle, daher wird die Zuordnung des Wissens zu den einzelnen Ontologiekonzepten die einzige Art der Wissenserfassung im Sinne der semantischen Annotation sein. Der Blickwinkel auf komplexere Annotationssysteme sollten jedoch nicht aus den Augen verloren werden. Ein in „K-Infinity“ integriertes Markup-Tool unterstützt zumindest ansatzweise eine solche Auszeichnung der Dokumente<sup>21</sup>.

#### **5.2.4 Konzepte für den Wissenszugriff / die Wissensnutzung**

Wie in Kapitel 2.3.2 bereits erwähnt, geschieht der Wissenszugriff auf ein ontologiebasiertes Wissensmanagementsystem, welches mithilfe einer Software wie „K-Infinity“ entwickelt wurde, zunächst mit klassischen Retrieval-Methoden. Die Stichwortsuche über ein Fragefeld ist dabei die gängigste Variante. Darüber hinaus bietet das „K-Infinity“ System allerdings Zugriffsmöglichkeiten, die herkömmliche Systeme nicht leisten können, wie die noch zu erläuternde „semantische Suche“.

Auch können beim Wissenszugriff implizite Querverbindungen zwischen einzelnen Wissensobjekten hergestellt werden, die nicht direkt im System ausmodelliert sind.

Hauptaufgabe eines Mitarbeiterportals ist es, den Mitarbeiter als wichtigste Unternehmensressource immer bestmöglich mit Informationen zu versorgen, um eine Effizienzsteigerung bei der täglichen Arbeit zu erreichen. Bezüglich des FHD-Portals lassen sich die verschiedenen Arten des Wissenszugriffs und der Wissensnutzung in zwei verschiedene Kategorien einteilen:

##### **a.) Push-Dienste**

Aufgrund eines Nutzerprofils, welches anfangs erstellt wird (und grundsätzlich und bestenfalls durch Auswertung von Logfiles ständig angepasst werden könnte), werden dem Mitarbeiter automatisch nach Login in das Intranet-Portal, die wichtigsten Informationen für seine tägliche Arbeit präsentiert. So ist es bspw. denkbar, einen wissenschaftlichen Mitarbeiter des Fachbereichs Informatik über Vorfälle im Fachbereich automatisch zu unterrichten, ohne das er selbständig danach recherchieren muss. Mithilfe von „K-Infinity“ sollen Profile generiert und durch Abfragen immer die aktuellen, für den Nutzer interessanten Nachrichten zur Verfügung gestellt werden.

---

<sup>21</sup> Nur ansatzweise deshalb, da lediglich Dokumente ausgezeichnet werden können, die direkt in die Ontologie eingetragen werden, also **nicht** als Datei hochgeladen werden.

Unter Push-Diensten versteht man dabei eine passive Form der Informationsversorgung, passiv in dem Sinne, dass der Nutzer nicht selbständig nach den im Netz vorhandenen Informationen suchen muss, sondern diese von Systemseite automatisch zur Verfügung gestellt bekommt. Innerhalb des FHD-Portal bedeutet dies einerseits die Bereitstellung von Newsdiensten, die auf der Profilstartseite eines einzelnen Nutzers automatisch und abhängig vom Nutzerprofil dargestellt werden. Im Zuge der Lieferung des technischen Basissystems durch die Firma intelligent-views werden Teile der Pushfunktionalität direkt implementiert sein. Da dieser Systemteil allerdings erst nach Fertigstellung dieser Arbeit geliefert werden wird, werden Grundfunktionalitäten im Prototypen exemplarisch selbständig und unabhängig davon umgesetzt.

Darüber hinaus ist eine Implementierung von sogenannten „Trigger“-Mechanismen geplant, die ereignisgesteuert wiederum Ereignisse auslösen können. So wird es möglich sein, abhängig von Änderungen an der Ontologie, automatisch Nachrichten über die Änderungen an interessierte Nutzer zu versenden. Wie schon bei den Pushdiensten, wird diese Funktionalität Teil des technischen Basissystems sein, die Entwicklung der Trigger ist allerdings ebenfalls zum Zeitpunkt der Arbeit noch nicht abgeschlossen und wird erst später in das Portal einfließen. Daher werden sich die Push-Funktionalitäten des Prototypen auf die Newsdienste beschränken:

Newsdienste liefern dem Nutzer auf einer Startseite die wichtigsten Informationen auf einen Blick und bilden mit den Triggerdiensten zusammen die Art der Informationsversorgung mit dem höchsten Automatisierungsgrad. Hierbei ist es nötig, eine exemplarische Startseite zu entwerfen, welche die aktuellsten Inhalte der für den Nutzer interessanten Gebiete automatisch extrahiert und visualisiert. Dabei werden zwei Arten der Selektion unterschieden: Einerseits die Art von Information, die für ihn als modelliertes Objekt des Netzes aufgrund der dort bestehenden Verknüpfungen für ihn interessant sind, andererseits Informationen, die er in einem selbst gestalteten Profil auswählt (siehe auch Kapitel 5.4 zur Profilgestaltung) . Erstere sind unter Anderem folgende Informationen:

- Nachrichten aus der Abteilung / dem Fachbereich des Nutzers
- Informationen über Projekte, an denen der Nutzer beteiligt ist oder für die er sich interessiert
- Informationen aus den Gremien, in denen der Nutzer Mitglied ist
- Informationen über Beschlüsse, von denen der Nutzer berührt wird
- Informationen aus der Verwaltung über Vorgänge, die den Nutzer unmittelbar und mittelbar betreffen.

- Hochschulinterne Mitteilungen (Stellenausschreibungen, Fortbildungsmaßnahmen etc.)
- Informationen über hochschulinterne Termine
- Pressemitteilungen
- Informationen über hochschulrelevante Themen (Politik, Wirtschaft etc.)

Bei der späteren Miteinbindung der Studenten als Nutzer des Portals muss darüber hinaus eine Berücksichtigung des studentischen Blickwinkels gewährleistet werden. So sollte dem studentischen Nutzer ein Überblick über sämtliche Fachbereiche, Studiengänge und organisatorischen Einheiten gewährt, Ansprechpartner transparent gemacht und allgemeine Informationen über Rahmenbedingungen des Studiums (Rückmeldung, Beurlaubung, Gebühren, Fristen etc.) auf den ersten Blick sichtbar gemacht werden. Eine weitgehende Integration des Student Service Centers wäre für diesen Zweck als Hauptziel zu erachten. Darüber hinaus sind für den Studenten auch Push-Dienste aus der oben aufgeführten Liste interessant, vor allem im Hinblick auf Geschehnisse innerhalb der Hochschulverwaltung, innerhalb der Gremien und bezüglich interner Termine. Ein Veranstaltungskalender speziell für Studenten würde vor allem über Kontaktmessen, Jobbörsen und fachbereichsübergreifende Veranstaltungen und Vorträge informieren.

Allerdings wird der Anteil der Studenten, der nicht in die Gremienarbeit direkt involviert ist, im Hinblick auf die erste Entwicklungsstufe des Portals noch keine allzu große Rolle spielen. Dies ist dadurch zu begründen, dass das Portal in dieser Stufe als Intranetportal realisiert wird und demzufolge nur die kleine Gruppe der Studierenden, die Mitarbeiter der Gremien sind, bedienen wird.

Um die in der Ontologie enthaltenen Informationen zu extrahieren und derart selektiv dem Nutzer zu präsentieren, wird sich der Technik der Expertensuche bedient. Diese bereits in Kapitel 4.1.1 angesprochene Funktionalität ermöglicht es, vorgefertigte Suchanfragen zu definieren, die dann als Basis für die Newsdienste unter Zuhilfenahme der Struts-Technologie angesprochen werden können. Die Expertensuche soll an dieser Stelle exemplarisch vorgestellt werden (eine ausführliche Beschreibung folgt im Zuge der Prototypisierung, siehe Kapitel 6).

Einer der späteren Newsdienste sollte bspw. über Neuigkeiten aus den hochschulweiten Gremien informieren; Senat und Studentenparlament sind solche Gremien. Mithilfe des Knowledge Builders werden verschiedene Expertensuchen modelliert, die zu einer einzigen Suche vereinigt werden, die dann für das Webfrontend ausgelesen wird. Wichtig ist hier im Vorfeld eine eindeutige Modellierung der Beziehungen und Attribute in der Art,



dass ähnliche Dokumente durch ähnliche Attribute und Relationen zu Konzepten beschrieben werden. Im Beispiel der Gremien sind das die Relationen „ist Dokument zu“, „hat Hauptthema“, „hat Thema“ und „wird veranstaltet“<sup>22</sup>. Die inhaltliche Gestaltung dieser drei Abfragen unterscheidet sich dabei grundlegend. Dies soll anhand der Expertensuche nach Neuigkeiten aus dem Senat der FHD dargestellt werden. Während „ist Dokument zu“, „hat Hauptthema“ und „hat Thema“ sich direkt auf Beziehungen zwischen dem Dokument und Senat beziehen (d.h.: ein Dokument hat eine direkte Relation zum Senat der FHD, bzw. zu einer Instanz des Konzeptes „Senat FHD“), nutzt die Relation „wird veranstaltet“ indirekt vorhandene Beziehungen mittels Inferenz. Dies bedeutet, dass automatisch geschlussfolgert wird, dass ein Dokument, welches eine Relation zu einem beliebigen Begriff der Ontologie hat, welcher wiederum über eine „wird veranstaltet von“-Relation mit dem Konzept „Senat FHD“ verbunden ist, für die Themensuche nach Neuigkeiten aus dem Senat relevant ist

Die zugehörige Expertensuche im Knowledge Builder stellt sich wie in Abbildung 19 gezeigt dar:



Abb. 19: Expertensuche „wird veranstaltet von“

Wie verdeutlicht, ist die Beziehung zwischen dem Senat der FHD und dem Dokument dabei indirekt über ein beliebiges Konzept aus der Ontologie, welches eine „wird veranstaltet von“ – Relation zum Konzept „Senat FHD“ hat. Das Ergebnis dieser Abfrage ist in Abbildung 20 zu sehen<sup>23</sup>.

---

<sup>22</sup> es sei hier angemerkt, dass aufgrund des Entwicklungsstadiums der Ontologie diese Angaben nicht als absolut zu sehen sind und sich im Laufe der Ontologieentwicklung noch erweitern bzw. ändern können.

<sup>23</sup> Dabei handelt es sich um die Ergebnisliste innerhalb des „Knowledge Builders“.

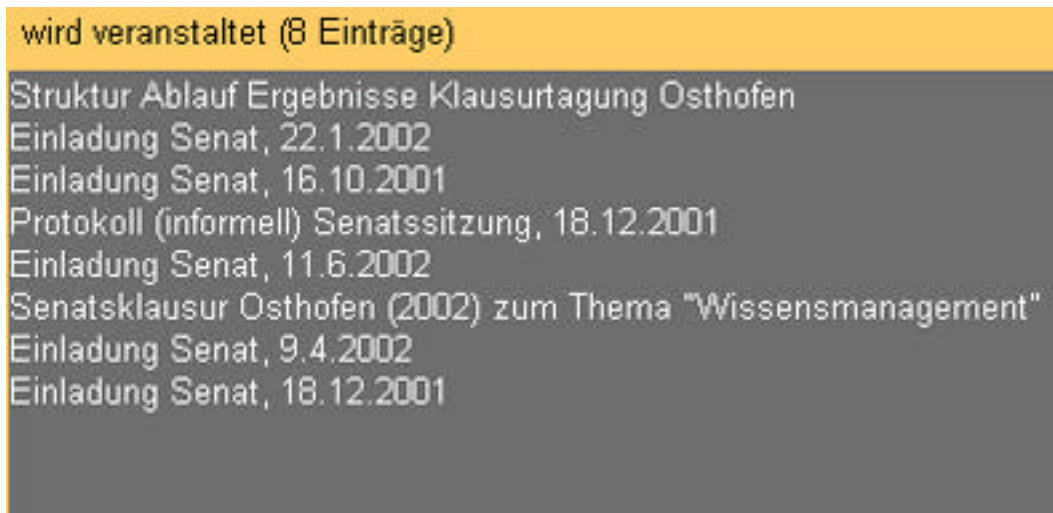


Abb. 20: Suchergebnis der Expertensuche aus Abb.21

Die datumsabhängige Auswertung müsste dabei zunächst durch Auswertung der Ergebnisse innerhalb der Webanbindung geschehen. So würde das Feld „Eintragsdatum“ als Schlüsselattribut für die Aktualitätsbestimmung verwendet werden. Die Firma „intelligent views“ liefert innerhalb eines Zeitraums, der den dieser Arbeit überschreitet, einen Mechanismus, der die Auswertung des Datums innerhalb der Ontologie ermöglicht, ohne die einzelnen Felder auf Programmiererebene auswerten zu müssen.

Im Zuge der Prototypisierung wird es ferner Aufgabe sein, möglichst variable Suchanfragen zu formulieren, die zu einem späteren Zeitpunkt, abhängig vom Nutzerprofil, die Ontologie abfragen. So soll bspw. ein Newsdienst über aktuelle Geschehnisse der eigenen Abteilung informieren. Um nicht für jede Abteilung eine eigene Suchanfrage generieren zu müssen, ist es Ziel eine parametrisierbare Expertensuche zu definieren, die über eine bestimmte Action (siehe Kapitel 4.4) einen Variablenwert übergeben bekommt, der die Abteilung des Nutzers beinhaltet. Die Suche wird dann abhängig von diesem Wert ausgeführt. Diese Art der parametrisierbaren Expertensuche ist seit der „Knowledge Builder“-Version 1.3 (April 2003) möglich und soll in der Prototypgestaltung exemplarisch verwendet werden.

Zusätzlich zu Newsquellen aus der Ontologie ist über die Integration eines allgemeinen Nachrichtentickers eines externen Anbieters nachzudenken, um den Mitarbeiter auch tagespolitisch auf Augenhöhe des Geschehens zu halten. Hierzu gibt es zahlreiche Anbieter, die, wie bspw. der Service „shortnews“ vom Magazin Stern<sup>24</sup>, teilweise kostenlos Nachrichten zur Erstellung von Tickern zur Verfügung stellen.

<sup>24</sup> [www.sternshortnews.de](http://www.sternshortnews.de) (Stand April 2003)

## **b.) Pull-Dienste**

Auf der anderen Seite muss über die Realisierung der Informationsversorgung auf Pull-Basis nachgedacht werden. Pull-Dienste sind Informationsdienste, die aktiv durch den Mitarbeiter genutzt werden können und die der Nutzer aktuell und gezielt einsetzen kann. Hierbei ist es von Systemseite besonders wichtig, dem Mitarbeiter einfache, nachvollziehbare Möglichkeiten zu bieten, die auch Einsteiger zum gewünschten Erfolg gereichen.

Im Falle des FHD Portals lassen sich die Pull-Dienste in zwei Kategorien aufteilen:

- i.) Mechanismen zur freien Suche innerhalb der Ontologie, die den Nutzer auf unterschiedliche Weise Zugang zur Ontologie gewähren
- ii.) Menüdienste, welche einen ausgewählten und zugeschnittenen Blick auf die Ontologie, korrespondierend dem Informationsbedarf des dem System bekannten Nutzers bieten.

## **i.) Suchmechanismen**

Im Zuge der Bereitstellung von Suchmechanismen muss dem Nutzer die Möglichkeit gegeben werden, alle für ihn zugänglichen Themenbereiche des Portals auf unterschiedliche Weise zu durchsuchen. Im Anwendungsfall wird das zunächst, wie oben bereits angesprochen, die Implementierung einer Stichwortsuche sein. Diese Stichwortsuche berücksichtigt allerdings im Gegensatz zu den sonst gängigen Suchmaschinen semantische Zusammenhänge zwischen den einzelnen Suchbegriffen.

Dabei bedient sich die Suchfunktionalität (zusätzlich zur syntaktischen Suche, d.h. der Suche nach Zeichenfolgen) der sogenannten semantischen Kegelsuche. Diese Art der semantischen Suche berücksichtigt auch Konzepte der Ontologie, die nicht genau der Suchanfrage entsprechen, die sich aber semantisch in der Nähe der Anfrage befinden. Vereinfacht dargestellt funktioniert diese Art der Suche wie folgt: Die Relevanz eines Konzeptes, welches nicht direkt in das Raster der Suchanfrage passt, definiert sich einerseits über eine Relation zu dem der Anfrage entsprechenden Konzept. Zusätzlich muss dieses Konzept mindestens über eine weitere Relation zu einem zweiten der Suchanfrage entsprechenden Wissensobjekt verfügen, um als relevant eingestuft zu werden.

Abbildung 26 zeigt das Suchergebnis der Suchanfrage „Gremium“ „FHD“. Eine rein syntaxorientierte Suche fände den Begriff „Gremium“ und den Begriff „FHD“ unabhängig der

Betrachtung des Kontexts der beiden Begrifflichkeiten. Die semantische Suche hingegen betrachtet den Zusammenhang (=die Relationen) der beiden Begriffe bei der Ergebnissuche und auch Relationen, die von den Begrifflichkeiten weggehen. Ein weiterer Vorteil der semantischen Suche zeigt sich bei der genaueren Betrachtung der Suchergebnisse: So werden keine Begriffe wie Senat, Studentenparlament etc. zurückgegeben, sondern die auf die Fachhochschule Darmstadt bezogenen spezifischeren Konzepte „Senat der FHD“ und „Studentenparlament der FHD“. Der Zusammenhang von Begrifflichkeiten entsteht hierbei nicht durch gemeinsames Auftreten der Begriffe in einem Dokument, sondern durch Beziehungen zwischen den einzelnen Begriffen innerhalb der Ontologie. Daraus resultiert, dass eine Suchanfrage mit inkompatiblen Begriffen streng genommen kein Ergebnis auf der Ebene der semantischen Suche zurückliefert<sup>25</sup>.

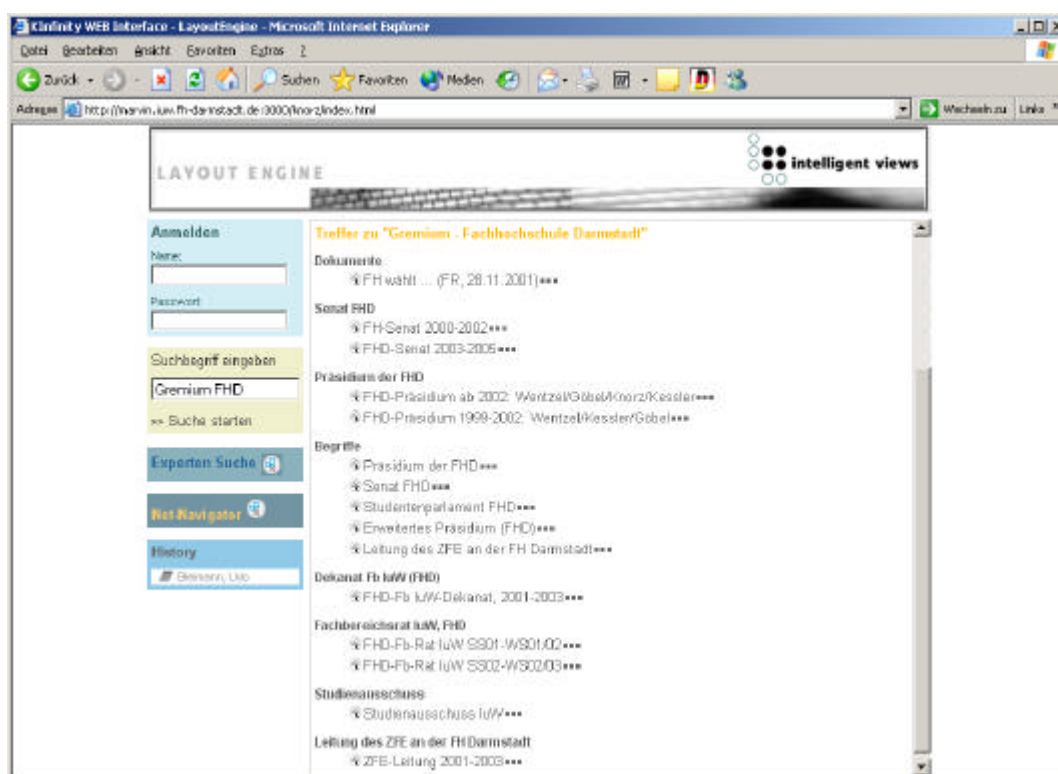



Abb. 21: Ergebnis einer semantischen Suche

Die zweite Möglichkeit, die den Nutzer bei der Suche unterstützt, ist die sogenannte Themensuche. Diese Suche wird im Knowledge Builder konfiguriert. Anhand verschiedener statistischer Nähekriterien werden automatisch verwandte Themengebiete zum aktuell ausgewählten Wissensobjekt angezeigt. Die Auswahl der Themengebiete geschieht dabei auf zwei Ebenen:

<sup>25</sup> Es sei allerdings darauf hingewiesen, dass die „K-Infinity“ Suche die Suchbegriffe auch isoliert betrachtet und als „direkte Treffer“ ausgibt, d.h. eine Suchanfrage „FHD Bauklötze“ würde alle Treffer zu „FHD“ ausgeben.

- a.) Zum einen werden Relationen innerhalb des Knowledge Builders gewichtet. Abhängig vom Wert dieser Gewichtung werden Wissensobjekte, die über diese Relation mit dem aktuellen Wissensobjekt verbunden sind, in die Themensuche aufgenommen.
- b.) Über die Bestimmung eines Grenzwertes in den Skripten der Webanbindung, ebenso wie über eine Angabe der maximal anzuzeigenden Ergebnisse, wird diese Auswertung zugeschnitten. Direkt verwandte Wissensobjekte können von der Themensuche ausgegrenzt werden.

Führt auch diese Suche den Nutzer nicht zum Ziel, existiert schließlich die Möglichkeit des Browsings, die dem Nutzer ein selbstständiges Auffinden von Informationen ermöglicht. Mithilfe des Net-Navigators (siehe Abbildung 22) ermöglicht diese Variante vor allem dem Nutzer, der die Suchbegriffe nicht in ihrer notwendigen Präzision definieren kann, ein Browsen durch die Taxonomien der Ontologie. Durch ein „Ablaufen“ der semantischen Relationen, startend an einem beliebigen Einstiegspunkt, kann er so sein Interessengebiet eingrenzen und sich nach und nach den für ihn relevanten Konzepten nähern. Der Net-Navigator wird durch Anklicken des -Symbols neben den gefundenen Begriffen aktiviert und soll zusätzlich über einen eigenen Menüpunkt („Navigieren“) aufrufbar sein.

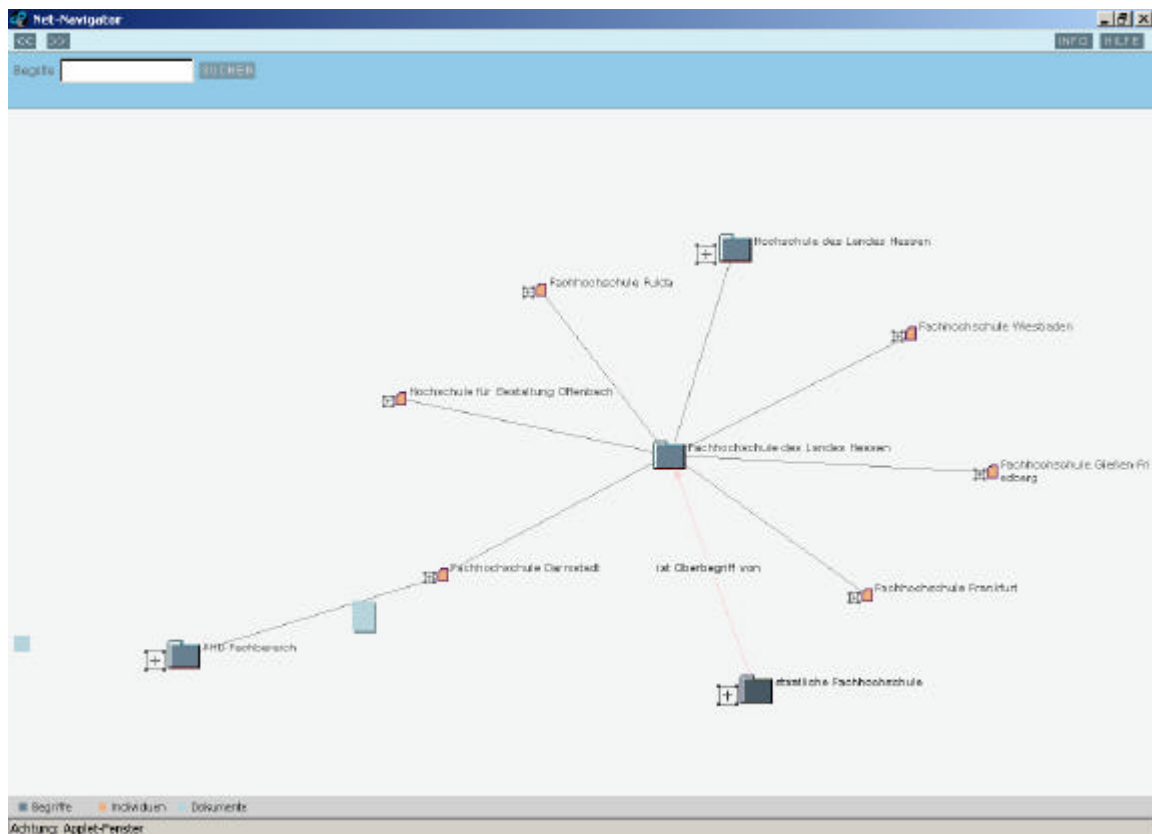


Abb. 22: Der Net-Navigator

Inhaltlich werden bei den Suchmöglichkeiten dem Nutzer scheinbar verschiedene Suchmaschinen präsentiert. Tatsächlich handelt es sich bei diesen Suchmöglichkeiten lediglich um unterschiedliche Einschränkungen der Zieldaten, die eine bessere Übersichtlichkeit gewährleisten. So wird eine Suche nach Experten oder Zuständigkeiten zu bestimmten Sachgebieten implementiert werden, welche derart angepasst ist, dass sie als Ergebnis nur Personen rückliefert: Dem Nutzer wird ein Interface geboten, das ihm die Eingabe einer Tätigkeit bzw. Zuständigkeit liefert, in der Ontologie wird dann über vordefinierte Relationen nach Personen gesucht, die eine dieser Relationen zu der eingegebenen Tätigkeit bzw. Zuständigkeit hat. Details der Implementierung solcher Pull-Dienste werden unter ii.) und im Zuge der Prototypisierung dargestellt werden.

## **ii.) Menüdienste**

Menüdienste unterscheiden sich in Ihrer Implementierung kaum von den Suchmechanismen, allerdings wird die „Sucharbeit“ dem Nutzer bereits in der Entwicklung abgenommen. Durch eine durchdachte Menüführung soll dem Nutzer ein vereinfachter Zugang zu den Informationen gewährt werden, die er besonders hochfrequent nutzt. Wie auch schon bei den Push-Diensten ist dabei die Hauptaufgabe, möglichst allgemein gültige Expertensuchen als Rückgrad der Menüpunkte zu definieren, die durch Parametrisierung an die jeweiligen Nutzerprofile angepasst werden. Mögliche Menüpunkte sind dabei im Allgemeinen:

- Ressourcenverwaltung

Die Ressourcenverwaltung ist ein Menüpunkt, der abhängig vom jeweiligen Nutzerprofil in der Menüführung direkt berücksichtigt werden soll. Die Implementierung der Ressourcenverwaltung soll dabei über Intrexx realisiert werden. Da die Funktionalität allerdings von Intrexxseite aus noch nicht bereitgestellt ist, wird dieser Aspekt im Rahmen dieser Arbeit nicht näher implementiert werden.

- Yellow Pages

Der Aspekt der Yellow Pages war einer der häufigst genannten Punkte innerhalb der Bedarfserhebung hinsichtlich des FHD-Portals und sollte unabhängig von Nutzerprofilen ein fester Bestandteil innerhalb der Menüführung werden. Die Realisierung dieser Funktionalität geschieht, wie bereits angesprochen, über eine Parametrisierung einer Expertensuche innerhalb des Knowledge-Builders. Der Nutzer hat dann die Möglichkeit, gezielt nach Personen zu suchen, die eine bestimmte Tätigkeit ausüben bzw. einen bestimmten Verantwortungsbereich haben. Hieraus lassen sich auch Informationen über Vertreter bestimmter Personen ableiten, die sich wiederum in Hilfestellun-

gen bei Abwesenheit einzelner Mitarbeiter (Wer ist Ansprechpartner bzw. wo finde ich das Know-How, wenn der Hauptverantwortliche nicht anwesend ist?) umwandeln lassen. Verbunden mit der Suchfunktionalität nach Personen, ergibt sich daraus eine der Schlüsselfunktionalitäten des Wissensmanagements auf der Ebene der Wissensträger. Ferner lässt sich durch Ableitung der Zuständigkeiten auf Abteilungsebene die Form eines Organigramms entwickeln, welche dynamisch dem Nutzer zugänglich gemacht wird.

- Helpdesk

Der Helpdesk soll eine Hilfestellung betreffs unterschiedlicher Prozessabläufe an der FHD werden und wird derzeit im Zuge einer parallel verlaufenden Arbeit entwickelt (siehe 5.2.2).

- Interner Marketplace:

Der interne Marketplace soll verschiedenen Abteilungen und Fachbereichen ermöglichen, freistehende Ressourcen intern zu veräußern. Der Nutzer hat dann die Möglichkeit, direkt nach diesen Angeboten zu suchen.

So wäre denkbar, dass der Fachbereich Informations- und Wissensmanagement alte, nicht mehr benötigte PCs intern an die Bibliothek weiterverkaufen könnte, die diese noch als Opac-Rechner einsetzen könnten. Der Vorteil ist eine zusätzliche Einnahmequelle für den Verkäufer und eine Einsparung für den Käufer gegenüber eines Neuerwerbs. Eine beispielhafte Umsetzung innerhalb der Ontologie wäre dabei die folgende:

Zunächst muss eine Erweiterung (Rolle) angelegt werden, welche ein (nicht zwingenderweise direktes) Individuum von „Gegenständliches“ zur Ware erweitert. Diese Erweiterung besitzt die Attribute „Anzahl“ und „Preis“ (zusätzliche Attribute wie „Zustand“ etc. wären denkbar). Ferner wird eine Relation definiert, welche die Verbindung zwischen dem Verkaufsobjekt und dem verkaufenden Organ herstellt („bietet an“).

Denkbar wäre auch die zusätzliche Definition einer Erweiterung „Verkäufer“, die Akteuren erweitern könnte und der die Relation „bietet an“ zugeordnet würde. Nachteil einer solchen Erweiterung wäre allerdings eine erschwerte Pflege der Angebote. So müsste, um das System integer zu halten, zur Löschung eines Verkaufsartikels nicht nur dieser Artikel selbst gelöscht werden, sondern darüber hinaus auch die Erweiterung des Verkäufers. Daher wurde der Relation allgemeiner die Quelle „Aktor“ und

das Ziel „Gegenständliches“ gegeben. Zusätzlich zur „bietet an“ Relation, die zumeist die Ware mit dem verkaufenden Fachbereich (in Abbildung 23: der Fachbereich IuW) verbindet, gibt es eine Relation, die zum Ausdruck bringt, welche Person als Ansprechpartner für das Angebot dient.

Die Ware (im Beispiel: Das PC-System) kann dabei aus mehreren Teilen bestehen. In Abbildung 23 sind ein Monitor (Iiyama 1536T) und ein Drucker (Epson C60) über die „ist Teil von“-Relation mit dem erweiterten Individuum verbunden.

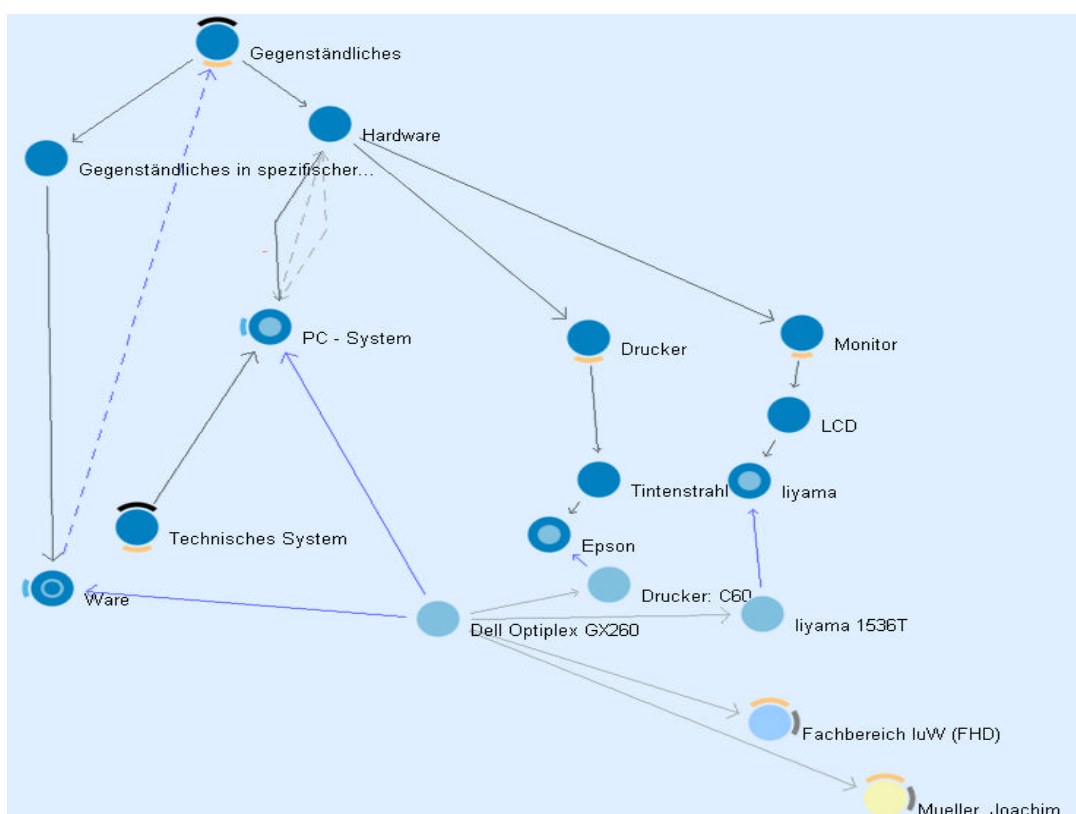


Abb. 23: Integration eines Marketplace-Angebotes

Da es sich bei den Anbietern vornehmlich um Institutionen handeln wird, wird zwischen dem Verkaufsobjekt (welches um die Rolle „Ware“ erweitert wurde) und einem Ansprechpartner eine „ist zuständig“-Relation gezogen, die den Ansprechpartner für den jeweiligen Artikel definiert.

Das in Abbildung 23 gezeigte Beispiel ist formal nicht ganz korrekt. Um vollkommene formale Korrektheit zu erlangen, müsste es einen Begriff „Dell Optiplex GX260“ geben und eine Instanz dazu, welche das konkrete, zum Verkauf anstehende PC-System beschreibt. Bei einer endgültigen Implementierung des Marketplaces müsste die Modellierung im Bereich „Gegenständliches“ stark verfeinert werden. Zum derzei-



tigen Entwicklungsstand der Ontologie und zur exemplarischen Demonstration der Möglichkeiten ist die in Abbildung 23 dargestellte Modellierung jedoch ausreichend. Verfügbar gemacht werden die einzelnen Angebote wiederum mithilfe einer Expertensuche, welche Konzepte mit der Erweiterung Ware auswertet (siehe Kapitel 6).

- Interner Arbeitsmarkt

Eine interne Jobsuchmaschine könnte mit dem eben vorgestellten Marketplace-Konzept realisiert werden. Sind in einer Abteilung / in einem Fachbereich Stellen vakant, wird dies mithilfe einer Erweiterung zum Ausdruck gebracht. Der Mitarbeiter bekommt so als erster die Möglichkeit, sich auf eine für ihn interessante Stelle zu bewerben. Im Zuge der Einführung eines umfassenden Wissensmanagements an der Fachhochschule Darmstadt kam auch die Verwirklichung interner „Job Rotation“ zur Sprache [Knorz2002]. Unter „Job Rotation“ versteht man dabei die Möglichkeit, durch unterschiedliche Abteilungen zu rotieren und sich an verschiedenen Stellen der Organisation einerseits Know-How anzueignen, andererseits das eigene dem Unternehmen und anderen Abteilungen zur Verfügung zu stellen. Auch dieser daraus entstehende „Markt“ für Arbeitskraft kann mithilfe der Ontologie im Stile der in Abbildung 28 vorgestellten Art und Weise realisiert werden.

Aufgrund der nicht ausreichenden Modellierung der Ontologie auf diesem Bereich, wird der interne Arbeitsmarkt erst zu einem späteren Zeitpunkt realisiert werden<sup>26</sup>.

Wichtig auf der Ebene der Wissensnutzung ist der Fokus auf den Mitarbeiter als Nutzer des Portals. Um nicht in die in 2.1.2.6 vorgestellte „Todesspirale von Wissenssystemen“ (siehe Abbildung 3) zu gelangen ist eine ergonomische, gut nutzbare Oberfläche ebenso wichtig wie die Aktualität des Nutzerprofils und die Pflege der Ontologie. Die Ergonomie der Benutzerschnittstelle wird in 5.5 unter dem Stichwort „Visualisierung“ angeschnitten, der Aspekt der Profilgestaltung in Unterkapitel 5.4 behandelt.

### **5.3 Konzepte für das Kommunikationsmodell**

Verdecktes Wissen ausfindig und nutzbar zu machen erfordert an erster Stelle eine Verbesserung der internen Kommunikation. Alle durchgeführten Bedarfsanalysen ergaben, dass mangelnde Kommunikation zwischen den einzelnen Parteien die meisten Defizite

---

<sup>26</sup> Bei diesem Punkt handelt es sich ferner um einen hochschulpolitisch sehr brisanten Punkt, dessen Umsetzung eine Genehmigung auf verschiedenen Ebenen erfordert.

zu verantworten hat. Dieses Unterkapitel beschäftigt sich daher im Folgenden mit Kommunikation einerseits formellen und andererseits informellen Charakters, die mithilfe des Portals externalisiert werden soll. Aufgabe der Kommunikation ist dabei:

- a.) eine Verfügbarmachung impliziten Wissens,
- b.) die Sicherstellung dessen Nutzung und
- c.) die Entwicklung neuen Wissens aus vorhandenem.

Diese Zielstellung begründet die (Quellen wie [Zeeb] widersprechende) Definition der Kommunikation als Metaprozess; Eine eindeutige Zuordnung zu den einzelnen Teilprozessen des Wissensprozesses wäre nur mit Einschränkungen möglich. Eine verbesserte Kommunikation innerhalb eines Organisationsverbundes wie der Fachhochschule sorgt dabei für eine erhöhte Integration der Mitarbeiter und Abteilungen und führt durch eine gefühlte Akzeptanz und Wahrnehmung der Mitarbeiter zu einer verbesserten Motivation jedes Einzelnen [Bernard].

Um eine Verbesserung der Kommunikationskultur zu erreichen, müssen dabei zwei unterschiedliche Arten von Kommunikationsprozessen unterschieden werden:

- 1.) Durch eine Integration formeller Kommunikationsmöglichkeiten wird Kommunikation vorgelebt, was zur Nutzung von Technologien informeller Kommunikation anreizen soll. Die Konzeption hinsichtlich der formellen Kommunikation ist dabei durch die in den vorangehenden Unterkapiteln beschriebenen Konzepte bereits weitestgehend definiert. Betrachtet man die Definition der formellen Kommunikation, wird deutlich, dass darunter zumeist die unilaterale Kommunikation von Strategien und Zielen seitens der Organisationsleitung verstanden wird, ebenso wie Informationen zu aktuellen Projekten, Arbeitsgruppen etc. Diese Art der Kommunikation, traditionell durch den Versand von Newslettern oder der Herausgabe einer Firmenzeitung realisiert, wird durch die in 5.2.4 vorgestellte Implementierung des Mitarbeiterportals bereits weitestgehend abgedeckt.

Allerdings gibt es Fälle, in denen diese Aspekte nicht weitreichend genug erscheinen. Die Informationsversorgung durch Push- und Pull Dienste ist zwar präzise auf die Anforderungen der Nutzer zugeschnitten, doch nach ihrer Implementierung zunächst starr, d.h. die Profile befinden sich nicht im stetigen Wandel, sondern werden von Zeit zu Zeit durch den Nutzer selbst, oder den Systemadministrator angepasst. Die Reaktionszeit des Systems ist daher in einigen Fällen zu langsam. Abhilfe soll hier ein

Newsverteiler schaffen, der es einzelnen Nutzern des Systems ermöglicht, zielgerichtet die notwendigen Informationen an eine Auswahl von Mitarbeitern zu schicken (als Beispiel wäre hier der Dekan eines Fachbereichs zu nennen, der an alle Fachbereichsmitarbeiter eine Nachricht schickt). Innerhalb des FHD-Portals können sich für diesen Zweck Eigenschaften der Ontologie als Basis des Portalsystems zunutze gemacht werden. So soll es möglich sein, aus der Ontologie heraus eine Mail an eine bestimmte Nutzergruppe zu schicken. Diese Benutzergruppe kann dabei entweder explizit ausgewählt oder aus der gegenwärtigen Netzstruktur zusammengesetzt werden. Zeitnahe und interessengerichtete Kommunikation wird so möglich. Wichtig dabei ist nicht nur hierarchieorientierte Verteiler zu ermöglichen. Vielmehr sollte dem Mitarbeiter die Möglichkeit gegeben werden, Nachrichten auch an höhere Hierarchieebenen zu verfassen, um die Kommunikation und die daraus resultierende Meinungsbildung und Wissensgenerierung zu „enthierarchisieren“. Dem Nutzer soll dabei ermöglicht werden, über zuvor definierte Relationen von einem Objekt weg Nachrichten an direkt mit dem Konzept in Verbindung stehende Objekte zu versenden. Mailziele können dabei sowohl organisatorische Einheiten (Fachbereiche, Abteilungen etc.) als auch Personen sein.

Der Ablauf beschreibt sich dabei wie folgt: Nach Aufruf des Mailversands (bspw. vom Konzept „Senat FHD 2000-2002“ aus) bietet sich dem Anwender eine Liste der möglichen Relationen, die Mailziele definieren. Dabei wird ein Relationsattribut ausgewertet, welches eine „bildhafte“ Beschreibung der Relation enthält (die Relation „hat Mitglied“ würde so mit „Mitglieder“ beschriftet werden). Nach Auswahl der Relationen zur Zielentscheidung, erscheint eine weitere Auswahl der über die Relationen bestimmten Empfänger (bei einer Relation „hat Mitglied“ vom Individuum „Senat FHD 2000-2002“ weg, würden die Mitglieder des vergangenen Senats als Ziel ausgewählt). Nach dem Schreiben und Absenden der Nachricht erhält der Nutzer die eigene Mail an sich selbst geschickt und kann so die aus dem Netz gesendeten Nachrichten in einem eMail-Client übersichtlich verwalten. Bei der Realisierung der Funktionalität soll ebenfalls Wert darauf gelegt werden, dass eine Person in unterschiedlichen Rollen unterschiedliche e-Mail-Adressen haben kann. D.h. eine Mail an Herrn Prof. Dr. Knorz als Vizepräsident der Fachhochschule Darmstadt ginge an eine andere e-Mail-Adresse als eine Mail an Prof. Dr. Knorz als Professor des Fachbereichs Informations- und Wissensmanagement.

Doch auch der zielgruppenorientierte Versand von Verteilernachrichten erfüllt noch nicht gänzlich die Anforderungen an ein Konzept zur Ausgestaltung der formellen

Kommunikation. Dies zeigt sich bei der Betrachtung der Kommunikation bezüglich des alltäglichen Geschäftsablaufs. So ist die tägliche Arbeit geprägt von Gesprächsrunden, Meetings und Abstimmungsgesprächen, die sowohl abteilungsintern, als auch abteilungsübergreifend stattfinden. Diesbezüglich kann sich herausstellen, dass auch der elektronische Mailversand als Kommunikationsmedium nicht zeitnah genug ist. Ferner sind „Gespräche“, die über die bilaterale Ebene hinaus gehen, mittels e-Mail nur schwer durchführbar. So ist es im Zuge der Einführung des FHD-Portals Aufgabe, Technologien für die multilaterale Kommunikation zur Verfügung zu stellen. Eine gängige Praxis zur Implementierung einer solchen Technologie ist eine Art Chatfunktionalität, welche die Kommunikation auch mehrerer Personen zeitgleich ermöglicht. Zur Realisierung einer solchen Funktionalität kann sich wiederum der ontologischen Technologie des Portals bedient werden. Chatteilnehmer könnten hier, ebenso wie beim Newsletterversand, aus der Ontologie heraus definiert werden. Die technische Entwicklung der Messengerfunktion könnte dabei über eine Integration von Standardchatsoftware („IRC Daemon“, „Telnet Chatserver“ etc.) oder u.U. sogar als „Voice over IP“-Lösung umgesetzt werden. Die Integration einer solchen Softwarelösung in das System könnte ähnlich der Realisierung der Mailverteilerlisten gelöst werden.

- 2.) Bei der Ausgestaltung der informellen Kommunikation mithilfe eines Portals wie dem der Fachhochschule ist es Ziel, den Mitarbeiter zu ermutigen, das eigene Wissen zu kommunizieren und ihm den Nutzen davon zu verdeutlichen. Der Wissensaustausch und die damit verbundene Verfügbarmachung verdeckten Wissens sollte dabei möglichst den informellen Charakter beibehalten. Traditionell wird implizites Wissen in einer Organisation wie der Fachhochschule Darmstadt durch informelle Kommunikation ausgetauscht und externalisiert. Dies geschieht zumeist auf der Ebene von Gesprächen mit unmittelbaren Kollegen, auf abteilungsübergreifenden bzw. hochschulweiten Veranstaltungen oder bei einem mittäglichen Treffen von Kollegen in der Mensa. Diese Art der Kommunikation muss mithilfe des Portalsystems verbreitert werden, räumliche Entfernungen sollen als einer der Haupthinderungsgründe für informelle Kommunikation verschwinden. Dabei muss dem Mitarbeiter das „unverbindliche“ Gefühl eines Gesprächs erhalten bleiben. In Unternehmen hat in den letzten Jahren der elektronische Mailversand auch hierbei eine zentrale Rolle als Kommunikationsmedium auf informeller Ebene eingenommen. Grenzen dieses Mediums der Mitarbeiterkommunikation zeigen sich allerdings vor allem im Hinblick auf Starrheit und auf Verzögerungen der Kommunikation. Auch löst die Kommunikation per e-Mail ein weiteres Hauptproblem nicht: die Identifizierung der Wissensträger. Kontakt findet auch

hierbei lediglich im Bekanntenkreis statt, Interessengleichgesinnte lassen sich nur bedingt (bspw. durch „Networking“ im Sinne von „Ich kenne jemanden, der jemand kennt“) ausfindig machen. Zur Lösung dieses Dilemmas sollen verschiedene Funktionalitäten im Zuge der Portaleinführung ausdefiniert werden:

Die Möglichkeit zur Diskussion von Themen, die nicht unmittelbar mit der täglichen Arbeit verbunden sind, sehen [Probst et al.] als Kernaspekt für die Generierung neuen Wissens. Dieser Teilpunkt des Wissensprozesses ist allerdings nicht der einzige, der von einer derartigen Kommunikationskultur profitieren kann. Unter 2.1.2. bereits vorgestellte „Communities of Practice“, also Interessengruppen, die sich abseits der Organisationsstruktur bilden, können mithilfe des Portals Rahmenbedingungen zur Diskussion bestimmter Themen bereitgestellt bekommen. Durch ein in die Ontologie eingebundenes Forensystem kann nicht nur das Wissen der Mitarbeiter über diese Themen zugänglich gemacht und geteilt, sondern direkt in den Unternehmenskontext eingegliedert werden. Dies hat Einwirkungen auf alle Subprozesse des Wissensprozesses. Wissen wird über die Träger identifiziert, automatisch in den Unternehmenskontext (die Ontologie) integriert, verfügbar gemacht und direkt von anderen Teilnehmern genutzt.

Abbildung 24 zeigt eine exemplarische Modellierung eines solchen Forensystems. Zunächst wird eine neue Begriffsebene namens „Forum“ eingeführt. Diese Begriffsebene steht in der Hierarchie außerhalb der bisherigen Ontologie und findet ihren Einhängpunkt im obersten Punkt der Top-Level-Ontologie („Wurzelbegriff für normale Begriffe“), dessen Subkonzept auch das Mutterkonzept der bisherigen Ontologie ist („T“). Dies wird derart gehandhabt, um die Forumsaktivitäten von der restlichen Taxonomie zu trennen und so unabhängig voneinander beide Aspekte pflegen zu können. Diese Art der Modellierung ermöglicht ferner eine leichtere und gezieltere Suche direkt nach den Forumsbeiträgen. Im Beispiel hat das Konzept „Forum“ nur einen Unterbegriff, „Topic“, der als Individuenerzeuger für die einzelnen Forenthemen fungiert. Die Topics definieren dabei die Attribute „Startdatum“ und „Enddatum“ (wobei auch nach Beendigung eines Forenthemas dieses nicht aus der Ontologie gelöscht, sondern das dort produzierte Wissen bewahrt werden sollte). Weitere Attribute wären hier denkbar. Exemplarisch wurden hier zwei Foren erzeugt: „Wissensmanagement“ und „Hochschulpolitik“. Über eine Teilnehmerrelation mit Instanzen von Person als Ziel können Mitglieder der Diskussionsforen festgelegt werden. Eine explizite Definition dieser bringt folgende Vorteile mit sich: So kann bei der späteren Realisierung einer Suche nach Experten zu bestimmten Fachgebieten, diese Relation mit ausgewertet werden (Im Sinne von: Wenn sich jemand für ein Thema interessiert und in ei-

nem Forum aktiv selbiges diskutiert, ist er als Experte zu werten). Die thematische Integration in den Kontext der FHD findet mittels Relationen der Topics zu Subkonzepten von T statt. So ist in Abbildung 24 das Forum „Wissensmanagement“ mit der Senatskommission Wissensmanagement über eine „hat Thema“-Relation verbunden. Die einzelnen Forumsbeiträge sind als Dokumente innerhalb der Dokumentenhierarchie zu finden und über eine Relation „hat Beitrag“ bzw. „ist Beitrag von“ mit den Topicinstanzen verbunden. Dabei wesentlich ist für eine Bestimmung der Threadreihenfolge eine Relation „hat Vorgänger“ bzw. „ist Vorgänger von“. Die Trennung von Forum und Foruminhalt hat den Vorteil, dass selbst bei einer Löschung der Forentopics (welche generell im Sinne der Wissensbewahrung nicht zu empfehlen ist!) die Inhalte zur Suche erhalten bleiben. Das in Abbildung 24 dargestellte Modell soll als exemplarisch verstanden werden. Im Zuge der Prototypisierung des Portals wird es noch keine Realisierung einer Webschnittstelle geben. Dies ist einerseits durch die zeitlichen Grenzen dieser Arbeit zu begründen, andererseits hat die Fachhochschule derzeit nur ein sehr beschränktes Kontingent an Schreiblizenzen auf das „K-Infinity“-System. Da ein Forensystem allerdings Schreibrechte für alle Beteiligten benötigt, macht eine Implementierung zu diesem Zeitpunkt wenig Sinn. In einer weiteren Entwicklungsphase kann das Modell dennoch als Input verwendet werden.

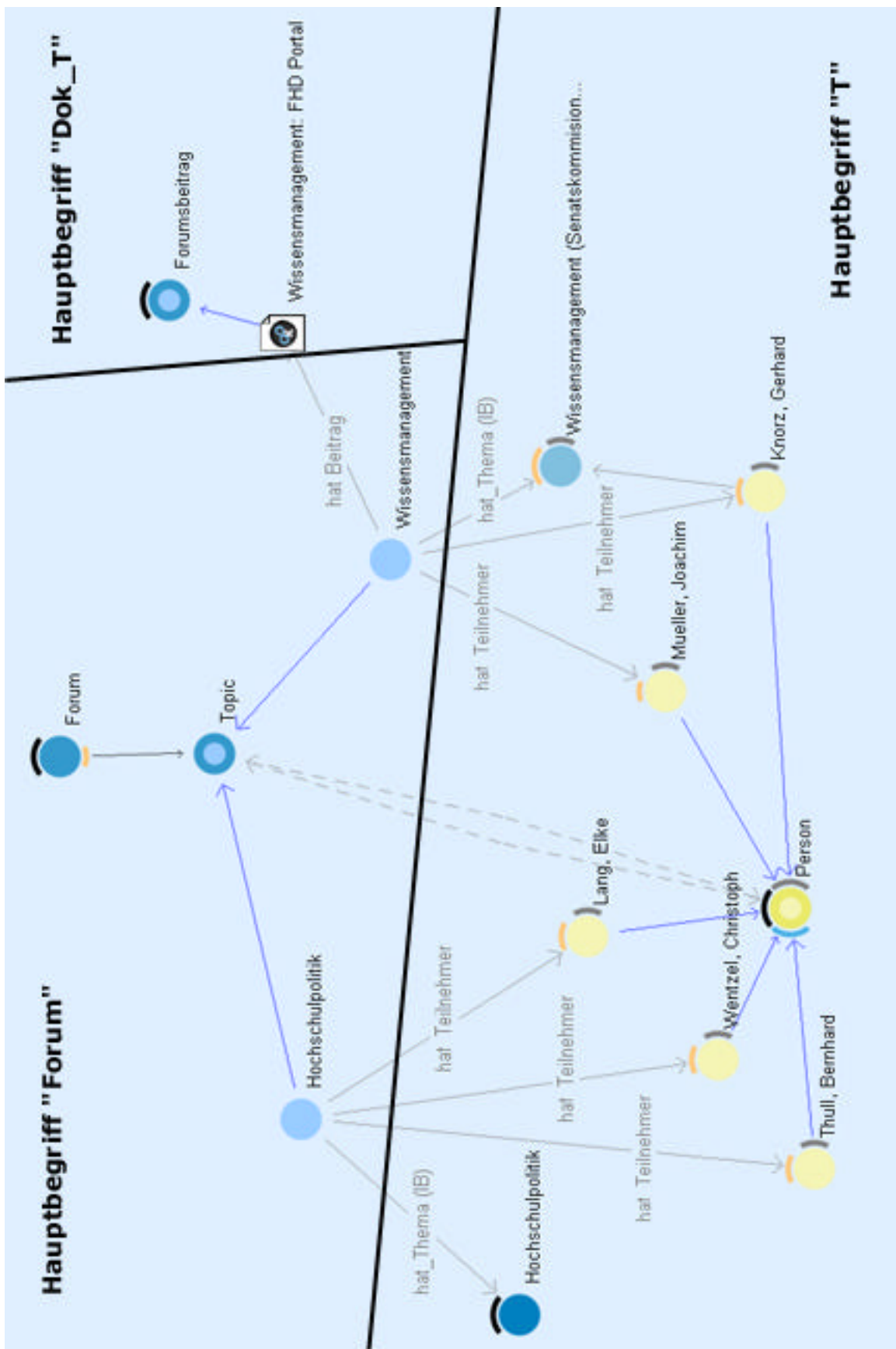


Abb. 24: Beispielhafte Modellierung eines Forensystems

Die Art der Unterstützung informeller Kommunikation mithilfe eines Forensystems ist allerdings nur ein Teilaspekt, den es zu berücksichtigen gilt. Ein Großteil der Kommunikation auf informeller Ebene findet, wie oben bereits erwähnt als Gespräch unter Kollegen statt. Diese Art der Kommunikation erweitert den Wissensbereich der Mitarbeiter in oft nicht unerheblichen Maße. Allerdings findet traditionell ein Informations- und Wissenstransfer beschränkt auf Personen im direkten lokalen Umfeld des Mitarbeiters statt. Ziel muss es folglich sein, diese Art der Wissensbildung, elektronisch unterstützt, auf eine breitere Ebene zu heben. Hierbei sollte sich, wie schon im Hinblick auf formelle Kommunikation, der Chat-Technologie bedient werden. Eine direkte Integration eines Messengermoduls in die Portaloberfläche mit der Möglichkeit, alle Mitglieder des Portals direkt anzusprechen, fördert die Kommunikationskultur und verkürzt Kommunikationswege. Der Vorteil einer direkten Integration sollte auch hier ersichtlich sein: Eine Art „Chat-Verzeichnis“ ähnlich dem Telefonverzeichnis kann in Verbindung mit dem Datenmodell Aufschluss darüber geben, wer sich auf einem bestimmten Themengebiet auskennt, oder bestimmten Interessen nachgeht. Dieses Chat-Verzeichnis könnte mit Informationen über den Nutzer aus der Ontologie angereichert werden, auch persönliche Darstellungen der Mitarbeiter wären möglich, ebenso wie das Hinzufügen eines Bildes, um die persönliche Komponente der Funktion hervorzuheben. Vor allem die Möglichkeit multilateraler Kommunikation zeigt die Stärken einer solchen Technologie im Vergleich zu bspw. Telefon.

Zu guter Letzt sollte dem Mitarbeiter im Rahmen eines Feedbacksystems auf allen Bereichen die Möglichkeit der Kritik gegeben werden. Ehemals als „Kummerkasten“ bekannt, soll dies vor allem den konstruktiven Umgang mit Fehlern (den der Anderen und resultierend daraus den eigenen) fördern.

## **5.4 Profilgestaltung**

Schlüsselkomponente beim Entwurf des Portals der FHD ist die Entwicklung einer Nutzerverwaltung mit einhergehender Profilgestaltung des in der Ontologie enthaltenen Wissens. So ist es einerseits Aufgabe, dem Nutzer abhängig seiner Interessen und seines Aufgabengebietes (welche sich beide zum Teil aus der Ontologie selbst ergeben), einen möglichst einfachen Zugang zu den Informationen zu gewährleisten, die er benötigt. Darüber hinaus soll durch die Implementierung eines Rechtesystems mithilfe von „K-Infinity“, die Ontologie derart zugeschnitten werden, dass für ihn einerseits uninteressante und andererseits nicht zugriffsberechtigte Konzepte ausgeblendet werden.



Mit Hilfe des Rechtesystems können positive und negative Rechte formuliert werden. Bei einem Recht handelt es sich dabei um die Angabe einer Operation, die ein bestimmter Benutzer an einem Objekt des Wissensnetzes ausüben darf (oder – im Falle eines negativen Rechts – auch nicht ausüben darf). Die Benutzer sind, ebenso wie die für sie formulierten Rechte, selbst Individuen der Ontologie. Wird eine Frage an das System gestellt (also beispielsweise eine Suche), dann sucht es nach einem passenden Recht. Findet es eines, gibt das System die entsprechende Antwort. Wird kein Recht der Anfrage entsprechend gefunden, so gibt das Rechtesystem eine negative Antwort.

Ein Recht besteht unter „K-Infinity“ dabei aus drei Teilen: Einem Benutzer, einem Ziel und einer Operation und hat die folgende Aussage: Besitzer des Rechts (=Benutzer) darf Operation auf Ziel ausüben. Ein Beispiel hierfür wäre: „Ein Individuum von Person mit der Erweiterung „Pflegebeauftragter“ darf Instanzen vom Objekt „Fachbereich“ erzeugen. Bei der Rechtedefinition ist es auch möglich, eine der drei Komponenten nicht zu definieren. Ließe man das Ziel beispielsweise weg, so könnte ein Individuum von Person mit der Erweiterung „Pflegebeauftragter“ Instanzen aller Objekte erzeugen.

Rechte können in ihren Komponenten sowohl explizit definiert werden (also „Mueller, Joachim) oder Gruppen von Benutzern umfassen (Person mit Erweiterung „Student“). Operationsziele können dabei auch durch eine oder mehrere Expertensuchen bestimmt werden. Mögliche Operationen umfassen die Bereiche Lesen, Modifizieren, Erzeugen und Löschen.

Wie oben bereits erwähnt, unterscheiden sich die definierten Rechte dabei zwischen positiven und negativen Rechten (Verbote). Findet das System bei einer Anfrage eines bestimmten Nutzers weder ein positives noch ein negatives Recht, so wird dies als negatives Recht interpretiert. Die Folge davon ist, dass im Zuge der Profilgestaltung alle entsprechenden Rechte des Nutzers definiert werden müssen.

Ein Recht kann dabei keine Relationen als Zielobjekt haben. Dies bedeutet, dass eine Angabe über Attribute von Relationen ebenso wenig möglich ist, wie die Einschränkung von möglichen Zielen einer Relation (Person A darf Relation r nur zwischen Zielobjekten B und C ziehen).

Abbildung 25 zeigt die Definition eines Rechts im Knowledge Builder, welches einem Individuum mit der Erweiterung „Pflegebeauftragter“ gestattet, Individuen von allen in der Ontologie enthaltenen Konzepten zu erzeugen.

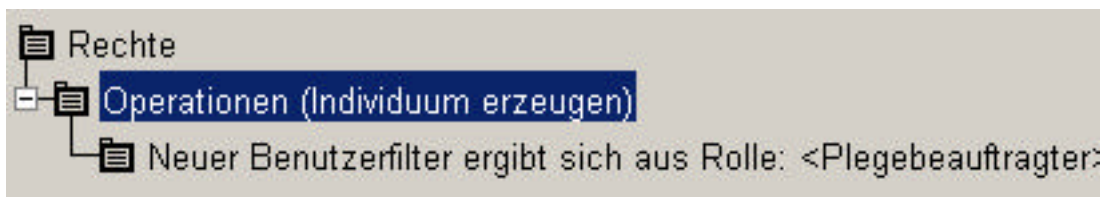


Abb. 25: Rechtedefinition

Im Zuge der Profilgestaltung des FHD-Portals wird das Rechtesystem nicht nur dazu verwendet, dem Nutzer die Ontologie seinen Zugriffsrechten und Interessen entsprechend zu filtern. Auch die Gestaltung der Push und Pull-Dienste soll teilweise mithilfe von Rechten definiert werden, um den Anwender, abhängig von den ihm zugeschriebenen Rechten mit den notwendigen Informationen zu versorgen. Kritisch gilt an dieser Stelle zu bemerken, dass die den Diensten zugrundeliegenden Expertensuchen **keine** Wissensobjekte darstellen und daher nicht direkt angesprochen werden können. Daher werden bei der Gestaltung des Prototyps die Profile zunächst über die Anbindungsseite definiert werden (siehe Kapitel 6).

Für die Profilgestaltung der einzelnen Nutzer ist das Rechtesystem jedoch alleine nicht ausreichend. Ferner muss dem Nutzer die Möglichkeit gegeben werden, abhängig von seinen Interessen (die unter Umständen in der Ontologie nicht erfasst sind), Zielkonzepte bzw. Teiltaxonomien auszuwählen, zu denen er automatisch Informationen aus der Ontologie erhalten will.

Die Personalisierung der Startseiten des FHD-Portals soll dabei zwei Subaspekte berücksichtigen: Zum einen soll automatisch ein Interessenprofil generiert werden, welches, orientierend an der Ontologie, dem Nutzer Interessensgebiete gemäß seines Kontextes zuteilt. So interessiert sich ein Mitglied eines Projektes „von Amts wegen“ für die Vorgänge innerhalb dieses Projektes.

Zusätzlich zu den daraus entstehenden Informationen der Startseite, die man unter einem Push-Dienst namens „Meine Aktivitäten“ zusammen fassen könnte, soll der Nutzer die Möglichkeit bekommen, Informationen aus weiteren, für ihn interessanten Gebieten zu erhalten. Dazu wird eine Pflegeseite generiert werden („My Account“) unter deren Zuhilfenahme der Anwender die Möglichkeit bekommen soll, Konzepte aus der Ontologie als für ihn relevant zu deklarieren. Dazu wird eine zusätzliche Relation kreiert, die den Fakt, dass sich der Anwender für ein bestimmtes Thema interessiert, darstellt („interessiert sich für“). Um auch den Kontext der so ausgewählten Interessensgebiete auszuwählen, wird ferner die Möglichkeit gegeben, ein Entfernungsparameter anzugeben, der den Abstand der mitauszuwertenden Knoten angibt. So könnte der Nutzer als Interes-

sensgebiet „Hochschulpolitik“ und zusätzlich eine Knotenweite von zwei angeben. Dies hätte zur Folge, dass für die Startseite des Nutzers der Knoten „Hochschulpolitik“ und alle umliegenden Knoten, die maximal zwei Knotenlängen (=Relationen) entfernt liegen ausgewertet würden. Eine Nutzung der Abkürzungsrelationen könnte bei beiden Aspekten eine Rolle spielen.

## **5.5 Visualisierung und Ergonomie**

Das nun folgende Unterkapitel soll einen Rahmen für die spätere Visualisierung der Portalinhalte bereit stellen. Das spätere Portallayout wird dabei im Anschluss an diese Arbeit vom Fachbereich Gestaltung erstellt. Daher soll hier nicht auf Visualisierungsdetails (im Sinne von Layoutgestaltung) eingegangen werden, sondern vielmehr ein Überblick über den „state of the art“ von Visualisierungsrichtlinien im Hinblick auf Ergonomie von Webanwendungen gegeben werden. Grundlage dieser Ergonomieüberlegungen ist dabei die europäische Norm EN ISO 9241, in der Kriterien für das Design ergonomischer Benutzerschnittstellen definiert sind. Die Norm besteht aus 17 Teilen, von denen vor allem Teil 10 eine zentrale Rolle bei der Gestaltung von Webanwendungen spielt. Dieser Teil der Norm beschreibt dabei „Grundsätze der Dialoggestaltung [ISO9241]“. Laut [Becker] ist dabei im Hinblick auf Webanwendungen unter Dialog die Benutzerschnittstelle im allgemeinen zu verstehen. Die Richtlinien für die Entwicklung von Webanwendungen können dabei wie folgt zusammengefasst werden:

- Aufgabenangemessenheit

Unter Aufgabenangemessenheit ist die Unterstützung des Nutzers durch die Anwendung bei der effizienten und effektiven Erledigung einer bestimmten Aufgabe zu verstehen. Hierzu zählt man das sichtbare Auffinden von Ansprechpartnern, die effiziente Gestaltung von Eingabefeldern und Menüs und die Unterstützung durch sinnvolle Suchmöglichkeiten.

Ziel ist es zu gewährleisten, dass der Benutzer sich auf seine Aufgabenlösung konzentrieren kann. Die Software soll ihn bei der Arbeit unterstützen und darf ihn dabei keinesfalls zusätzlich belasten. Da die Anzahl der Sinneinheiten, die das Kurzzeitgedächtnis auf einmal wahrnehmen kann, auf  $7 \pm 2$  limitiert ist, kann die Wahrnehmung durch eine Gliederung der angebotenen Informationen in sinnvolle Gruppen erhöht werden. Die Anwendung sollte zudem so gestaltet sein, dass sie die realen Arbeitsabläufe eines Benutzers nachempfunden. Sie muss sich entsprechend den Kenntnissen und Gewohnheiten des Benutzers verhalten. Das

heißt, dass bspw. die Positionsmarke automatisch dort positioniert wird, wo es vom Arbeitsablauf zweckmäßig erscheint.

- Selbstbeschreibungsfähigkeit

Elemente eines dialogorientierten Systems sollten den Nutzer über ihre Tätigkeit aufklären. Dies beinhaltet eine für sich sprechende Benennung von Links, Menüpunkten und Formularen. Dem Nutzer sollte jederzeit klar sein, welche Schritte er gerade ausführt. Die Norm selbst beschreibt „das unmittelbare Anzeigen eingegebener Daten und das Anzeigen des Änderungszustandes der Daten [als notwendig, um] [...] dem Benutzer beim Verstehen dessen zu helfen, was in der Anwendung geschieht und was er beeinflussen kann“ [ISO9241].

Selbstbeschreibend bedeutet zum Beispiel, dass ein situationsabhängiges Hilfesystem mit Bezug zur Arbeitsaufgabe existiert. Systemmeldungen (Anweisungen, Fehlermeldungen etc.) müssen präzise, einfach und unmissverständlich sein. Der Benutzer muss den Funktionsumfang einer Anwendung schnell und einfach erfassen können. Der Benutzer sollte aus visuell angebotenen Eingabemöglichkeiten auswählen können, und nicht aus dem Gedächtnis eingeben müssen. Leitaspekt zur Klärung der Selbstbeschreibungsfähigkeit sind dabei die sogenannten „5 Ws“: - woher komme ich, was ist bis jetzt gemacht worden, wo bin ich, was kann ich als nächstes tun und was kann das System. Nur bei einer Beantwortung dieser 5 Ws wird die Software dem Grundsatz der Selbstbeschreibungsfähigkeit gerecht.

- Steuerbarkeit

Die Vorgänge einer Webanwendungen sollen aktiv durch den Nutzer gesteuert werden können. Der Nutzer soll jederzeit in der Lage sein, Einfluss darauf nehmen zu können, was ihm wann und in welcher Art angezeigt wird.

Plakativ könnte das Ziel der Steuerbarkeit derart formuliert werden, dass der Benutzer das Gefühl vermittelt bekommen sollte, die Software zu kontrollieren und nicht den Eindruck haben sollte, selbst durch die Software kontrolliert zu werden. Die Initiative für einen weiteren Arbeitsschritt sollte immer beim Benutzer verbleiben [Kreativland]. So sollte der Nutzer die Möglichkeit haben, Reihenfolgen von Dialogschritten zu beeinflussen, ohne dass seine Arbeitsgeschwindigkeit darunter leidet.

- Erwartungskonformität

Anwendungselemente sind dann erwartungskonform, wenn sie einerseits über eine konsistente Benutzungsschnittstelle (Konsistenz in Bezeichnung, Layout, Sprache und Stil) verfügen und andererseits allgemeine Konventionen berücksichtigen. So ist es konform, den „Homebutton“ einer Homepage auf dem Logo der Webseite zu platzieren oder die Tabulatortaste für den Wechsel zwischen einzelnen Formularfeldern zu definieren (vgl. [Komdesign]). Zusammenfassend verhält sich eine Anwendung immer dann erwartungskonform, wenn der Benutzer sich durch den Ablauf von Dialogen in seiner Erwartung bestätigt fühlt

- Fehlertoleranz

Eine Webanwendung „ist fehlertolerant, wenn das beabsichtigte Arbeitsergebnis trotz erkennbar fehlerhafter Eingaben entweder mit keinem oder mit minimalem Korrekturaufwand durch den Benutzer erreicht werden kann“ [ISO9241].

Ein Dialog sollte sich dadurch auszeichnen, dass Eingabefehler zum einen weitgehend verhindert, zum anderen fehlerhafte Eingaben bzw. zu einem falschen Zeitpunkt ausgeführte Funktionen leicht zu korrigieren sind.

Die Anwendung sollte dabei Nutzereingaben automatisch auf die Korrektheit der Daten überprüfen. Sollten Funktionen zur automatischen Korrektur implementiert sein, so muss der Nutzer von der Anwendung über die Korrektur informiert werden und die Möglichkeit bekommen, die Korrektur zu revidieren.

Das heißt zum Beispiel, dass die Software Eingaben auf das korrekte Format überprüft. Wenn das System Fehler automatisch korrigieren kann, sollte es dem Benutzer über die Ausführung der Korrektur informieren und ihm Gelegenheit geben, die Korrektur zu überschreiben. Fehlermeldungen sollten sachlich und verständlich sein, technische Verkläuserung („Servlet Exception in index.jsp in line 24.03“) sollte ebenso vermieden werden wie Werturteile in den Fehlermeldungen („Unsinnige Eingabe“). Generell sollten Eingaben **vor** dem Absenden eines Formulars überprüft werden.

- Individualisierbarkeit

Die Individualisierbarkeit spielt eine der am meisten zu gewichtenden Rollen bei Portalen. Darunter ist einerseits die Konfigurierbarkeit der Inhalte (vgl. auch Kapitel 4.4) zu verstehen, wie auch individualisierbare Anordnungen im Layout.

Beständige Informationen des Nutzers sollten dauerhaft gespeichert werden und nicht jedes Mal durch diesen neu eingegeben werden müssen. Im Rahmen der weiteren Portalentwicklung nach dieser Arbeit sollte auch über Möglichkeiten

nachgedacht werden, unterschiedliche Nutzerlevel (Anfänger und Fortgeschrittener) zu implementieren, um jeden einen sachgerechten Zugang zu dem Portal zu gewähren und so die Nutzerakzeptanz zu erhöhen. Mithilfe der Struts-Technologie und dem Anbieten von Alternativtexten innerhalb einer Seite, kann diese Möglichkeit berücksichtigt werden.

- Lernförderlichkeit

Das System sollte den Nutzer beim Erlernen des Umgangs unterstützen. Dieser Punkt nimmt bei der Entwicklung eines Portals wie dem der FHD einen zentralen Aspekt ein. Die neuartige Technologie und deren optimale Nutzung muss dem Nutzer durch ständige Bereitstellung von anwendungsbegleitenden Ausführungen näher gebracht und Besonderheiten erläutert werden. Dabei ist die sprachliche und konzeptionelle Ausgestaltung der Oberfläche maßgeblich für die Zeit, die der Nutzer zur Erlernung des Systems benötigt. Im Gegensatz zur Selbsterklärungsfähigkeit wird mit Lernförderlichkeit das Langzeitgedächtnis des Benutzers angesprochen. Um eine möglichst kurze Einarbeitungszeit zu gewährleisten, ist es förderlich, folgende Kriterien zu berücksichtigen:

Abkürzungen sollten sinnvoll gewählt werden, Dinge, die gleich aussehen, sollten auch das Gleiche tun und ähnliche Objekte sollten auch ähnlich gestaltet sein. Eine Unterstützung der „Learning by Doing“ Methode ist laut [Wirth] ebenfalls vorteilhaft.

Bei der gegenseitigen Gewichtung der Grundsätze müssen mehrere Kriterien berücksichtigt werden. So spielen die Fähigkeiten des Anwenders, sowie die Merkmale der zu lösenden Aufgaben, wie auch die der Arbeitsumgebung, in der er gewohnt ist zu arbeiten, eine Rolle. Ein maximierter Bedienungskomfort ist dabei als Zielgröße zu sehen. Die durch das Portal bereit gestellten Funktionalitäten müssen in ihrer Anwendung einfacher und reibungsloser funktionieren, als die herkömmliche Variante der Aufgabenerledigung. Auf funktionslose Gestaltungselemente sollte aus Effizienzgründen verzichtet werden, eine nachvollziehbare Seitenlogik einen „lost in hyperspace“ [Wirth], also ein Verlorengehen in der Seitenstruktur, verhindern. Das FHD-Portal fängt viele dieser Aspekte durch die zugrundeliegende Technik bereits ab. Richtig genutzt, kann die Ontologie dem Nutzer zu jedem Zeitpunkt Aufschluss darüber geben, an welcher Stelle er sich aktuell befindet.

Bei der Frage nach der Gewichtung der Faktoren Nutzen und Benutzbarkeit lässt sich feststellen, dass der Nutzer eines Systems bereit ist, ergonomische Defizite zu erdul-

den, wenn der Nutzenzuwachs durch die Anwendung ausreichend ist. Dies bedeutet, dass Mängel in der Benutzbarkeit bis zu einem gewissen Grade ausgleichbar sind. Daraus lässt sich folgern, dass der Gesichtspunkt der Nutzbarkeit („Usability“) immer nur in Verbindung mit dem Nutzen („Utility“) der Anwendung betrachtet werden kann.

## 6. Entwicklung eines Prototypen des ontologischen FHD-Portals

In diesem Abschnitt der Arbeit werden die Erkenntnisse der vorangehenden Kapitel prototypisch in einen Portalentwurf umgesetzt. Dazu wird zunächst das für den Prototypen ausgewählte Nutzerszenario beschrieben. Innerhalb dieses Nutzerszenarios wird eine exemplarische Portalsicht vorgestellt und beispielhaft die Umsetzung der unter 5. beschriebenen Konzepte aufgezeigt. Das Nutzerszenario dient dabei nur der inhaltlichen Abrundung und Verdeutlichung der generellen Überlegungen. Im letzten Teil dieses Kapitels wird schließlich innerhalb dieses Nutzerszenarios ein konkreter Use-Case bestimmt, der die Möglichkeiten der Prozessunterstützung durch das ontologische Portal belegen soll.

In den folgenden Unterkapiteln wird weitestgehend auf die Erklärung von Quellcode verzichtet, der gesamte Quellcode findet sich in kommentierter Form im digitalen Anhang. Im Anhang befindet sich auch eine Liste der Prototypdateien mit einer jeweils kurzen Funktionsbeschreibung, sowie allgemeine Hinweise zum Lesen des Struts-Quellcodes.

### 6.1 Rahmenbedingungen

Der Prototyp entstand auf der Vorlage eines von der Firma „intelligent views“ entwickelten „Basislayouts“, welches Grundfunktionalitäten des Ontologienzugriffs (Textsuche, Themensuche und Navigation) definierte. Da erst im letzten Drittel der gesamten Arbeitszeit mit der Umsetzung der Konzepte begonnen werden konnte und die Entwicklungszeit aufgrund des fortgeschrittenen Zeitpunktes auf einen knappen Monat beschränkt war, bleibt die Implementierung der in Kapitel 5 vorgestellten Konzepte in einem beispielhaften Rahmen. Der späte Startzeitpunkt der Entwicklung resultierte dabei aus dem festen Liefertermin (18.04.03) der für den Prototypen notwendigen technischen Basis. Einige der in den folgenden Unterkapiteln vorgestellten Funktionalitäten sind darüber hinaus aufgrund noch fehlender technischer Funktionalitäten in der „K-Infinity“ Software Suite und Funktionalitäten, die erst zu einem späteren Zeitpunkt geliefert werden, provisorisch implementiert und sollten mit verbessertem technischen Rahmen angepasst werden. Da sich das Wissensnetz zum Zeitpunkt der Prototypisierung ebenfalls noch in der Evaluierungs- bzw. Verfeinerungsphase befand, sind einige inhaltliche Aspekte ebenfalls nur beispielhaft und wurden zum Teil eigens zur Veranschaulichung der verschiedenen Funktionalitäten in das Wissensmodell eingepflegt. Ziel der gesamten Prototypenentwick-



lung ist es daher, die generellen Möglichkeiten eines ontologiebasierten Portalsystems aufzuzeigen. Die detaillierte Umsetzung wird im Anschluss an diese Arbeit stattfinden.

Bei der Implementierung der Prototyp-Funktionalitäten mithilfe des Struts-Frameworks stand ein Absolvent des Fachbereichs Informatik unterstützend zur Seite.

## 6.2 Das Nutzerszenario

Um die in Kapitel 5 herausgearbeiteten Konzepte für ein ontologiebasiertes Portal der Fachhochschule Darmstadt prototypisch zu belegen, wurde als Nutzerszenario<sup>27</sup> eine Mitarbeiterin des „Zentrums für Forschung und Entwicklung“ (ZFE) der FHD gewählt. Anhand des daraus resultierenden Informationsbedarfs sollen dabei die Konzepte des Wissensprozesses dargestellt werden. Der aus dem Nutzerszenario resultierende Informationsbedarf wurde dabei in persönlichen Gesprächen mit Frau Dr. Ute Jochem, Mitarbeiterin des ZFEs erhoben. Einen Ausschnitt des Modells im Bereich „Zentrum für Forschung und Entwicklung“ ist in Abbildung 26 zu sehen.

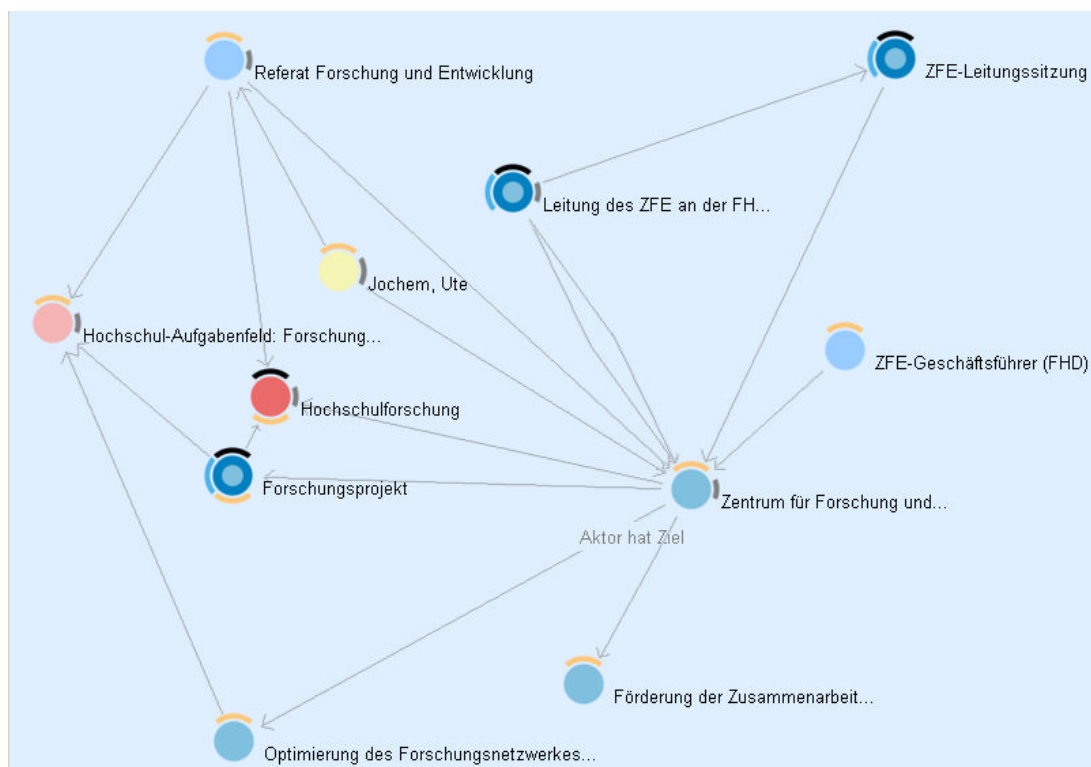


Abb.26: Das Modell im Bereich „Zentrum für Forschung und Entwicklung“

<sup>27</sup> Im Gegensatz zu den Beschreibungen aus 5.1 ist hier von Nutzerszenarien die Rede. Dies ist darin zu begründen, dass es tatsächlich um den Zugriff eines Beispielnutzers geht und nicht, wie in 5.1 um die verschiedenen Nutzungsszenarien, also die unterschiedlichen Zugriffsarten auf das System.

Das „Zentrum für Forschung und Entwicklung“ ist eine zentrale wissenschaftliche Einrichtung der FHD. Aufgaben des ZFEs umfassen die Pflege und Förderung der Forschung und Entwicklung an der Fachhochschule Darmstadt. Dazu gehören u.a. die Finanzierung von Projektassistenten an der FHD, die Antragstellung zu den einzelnen Förderprogrammen, die Koordination interdisziplinärer Projekte, die Unterstützung bei der Projektverwaltung, die Recherche zu speziellen Forschungsförderungen und die Unterstützung bei der Vermarktung von Forschungsergebnissen [ZFEWWW].

Innerhalb dieses Nutzerszenarios wird in Unterkapitel 6.5 ein Use-Case bestimmt, mit dessen Hilfe die Funktionalität des Portals hinsichtlich einer exemplarischen Prozessgestaltung aufgezeigt werden soll.

### 6.3 Ein Blick auf den Prototypen

In Abbildung 27 ist die Startseite der Beispielnutzerin Ute Jochem zu sehen. In den folgenden Unterkapiteln wird jeweils auf diese Abbildung verwiesen.

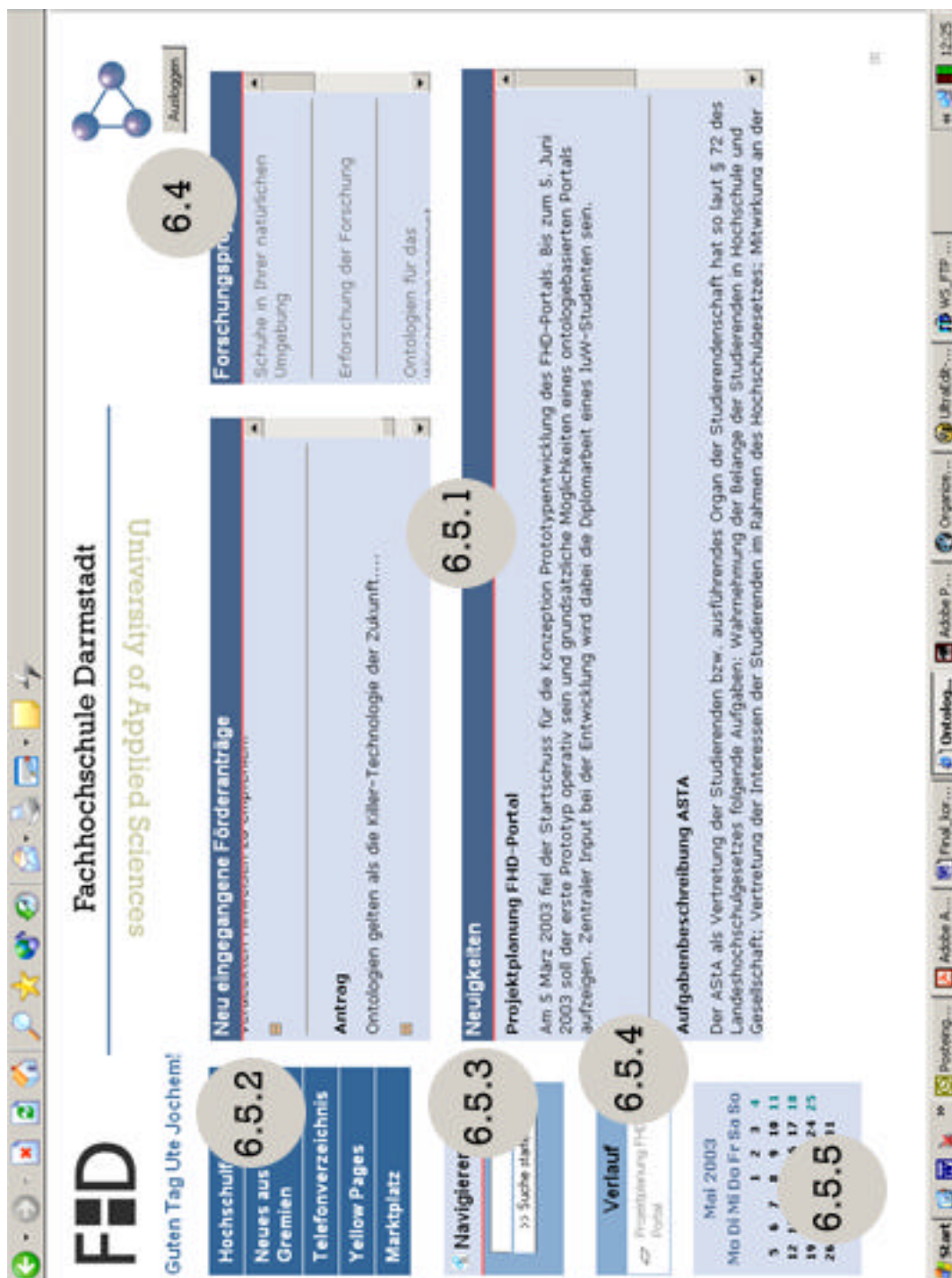


Abb. 27: Blick auf den Prototypen

### 6.3.1 Gestalterische Grundsätze

Im Zuge der Prototypgestaltung wurden einige gestalterische Grundsätze entwickelt, die über den gesamten Prototypen hinweg einheitlich gehalten wurden. An dieser Stelle sei angemerkt, dass die endgültige Layoutgestaltung durch den Fachbereich „Gestaltung“ der FHD realisiert wird, auf eine umfassende Layoutgestaltung wurde daher (und nicht zuletzt auch aufgrund des ohnehin engen Zeitrahmens) verzichtet.

#### a.) Anordnung

Wie bereits in Abbildung 27 zu sehen, orientiert sich die Anordnung der einzelnen Elemente an der in Kapitel 3.2 vorgestellten Gestaltungsrichtlinie von Portalen. Im Unterschied zu sonst gängigen Portalsystemen handelt es sich bei dem Prototypen allerdings nicht um ein Framesystem<sup>28</sup>; die gesamte Anordnung der Elemente geschieht vielmehr auf Tabellenbasis. Vorteil hiervon ist eine erhöhte Flexibilität im Gegensatz zur Frame-Anordnung, da prinzipiell jede einzelne Ausgabe auf andere Weise gelayoutet werden kann.

Den Prinzipien der Erwartungskonformität entsprechend (vgl. Kapitel 5.5), befindet sich der „Home“-Button des Prototypen auf dem Logo und dem Schriftzug der FHD. Hier sei betont, dass dieser den Nutzer auf seine Profilstartseite zurückbringt. Ein Ausloggen aus dem personalisierten Bereich findet entweder durch Anklicken des „Ausloggen“-Buttons statt, oder durch Klick auf das Ontologienlogo am oberen rechten Bildschirmrand.

Auch die Anordnung der weiteren Portalelemente wird den Erwartungen an ein Portal gerecht. So befindet sich die Menüführung links, ebenso wie die Suchfunktionalität und ein Kalender mit dem aktuellen Datum. Innerhalb eines Profils bleiben diese Komponenten statisch und dienen dem Nutzer so als Orientierungshilfe. Die personalisierte Anrede steht über dem gesamten Inhalt, der Logout-button unterhalb des Ontologienlogos. Die Inhaltstabellen sind zentral angeordnet, bei Aufruf enthalten diese die Inhalte der Push-Dienste (siehe Abschnitt 6.4.1). Eine extra Gestaltung der rechten Bildschirmseite findet erst mit Nutzung der Portalfunktionen statt. Die sogenannte Themensuche (siehe Kapitel 6.4.3) schlägt an dieser Stelle dem Nutzer bei getätigten Ontologiezugriffen verwandte Wissensobjekte innerhalb der Ontologie vor.

---

<sup>28</sup> Genau genommen gibt es neben dem Hauptframe zwei weitere Frames, die allerdings jeweils nur einen Pixel Breite haben. Einer davon lädt den Net-Navigator im Hintergrund, um Wartezeiten beim Aufruf zu verkürzen, der zweite greift auf das Intranet unter Intrex zu, um Cookies für die Kalenderfunktion setzen zu lassen (siehe Abschnitt 6.4.4).

## **b.) Farben**

Die Auswahl der Farben wurde in Anlehnung an die in [DesignManual] vorgeschlagenen Farben getroffen. Die Basisfarbe (Angabe im HTML-Format: #336699) wurde der „Farbpalette für FH[D]direkt“ entnommen. Alle weiteren Farben orientieren sich an dieser als Ausgangspunkt. Die Zuordnung der Farben ist dabei die folgende:

### **- Menüleiste**

Die Menüleiste hat als Hintergrundfarbe die Basisfarbe (HTML-Format: #336699). Die Menüpunkte sind in der Schriftfarbe weiß gehalten. Bei Mausbewegungen über die Menüpunkte wird die Schriftfarbe schwarz, die Hintergrundfarbe wechselt in ein helleres Blau (HTML-Format: #88ADD2).

### **- Navigieren und Kalender**

Der von der Leiste leicht abgesetzte Menüpunkt „Navigieren“ und der Kalender haben eine hellere Hintergrundfarbe (HTML-Format: #D6DEEF) als Gegenpol zu dem dunklen, leicht grünlichen Blauton der Grundfarbe.

### **- Suchfeld**

Die Farbe des Suchfeldes liegt zwischen der Menüleiste und der des Navigierenpunktes (HTML-Format: #88ADD2)

### **- Push-Dienste**

Push-Dienste werden in Newsboxen visualisiert. Der Kopf der Newsbox ist in einem kräftigen Blauton (HTML-Format: #46648C) gehalten, Schriftfarbe ist weiß. Unterhalb des Kopfes verläuft eine zweipixelbreite Linie in rot (HTML-Format: #FF6666) als Kontrast zu den sonst blauen Farbtönen. Die Newsbox-Inhalte sind in der gleichen Farbe gehalten, wie der Hintergrund des Punktes „Navigieren“ und der Kalender (HTML-Format: #D6DEEF).

### **- Wissensobjekte (lesen)**

Wissensobjekte werden grundsätzlich gleich dargestellt: Der Kopf ist dabei farblich genauso gehalten, wie der der Push-Dienste. Attribute werden ohne Hintergrundfarbe angezeigt, Relationen in dem hellen Blauton, der auch für den Kalender und den „Navigieren“-Menüpunkt verwendet wird (HTML-Format: #D6DEEF). Dokumente und Erweiterungen werden mit einem hellen Gelb gekennzeichnet (HTML-Format: #F6F6EC).

### - Wissensobjekte (schreiben)

Beim Editieren von Wissensobjekten (siehe Unterkapitel 6.5) werden Wissensobjekte anders dargestellt. Um dem Nutzer den Unterschied zwischen den beiden Modi „auf den ersten Blick“ zu verdeutlichen, wurden dafür die Farben im Kopf des Objektes vertauscht. Hintergrund des Kopfes ist nun das Rot, welches, die dünne Linie unterhalb des Kopfes im Lesemodus bildete. Die zweipixelbreite Linie hingegen ist im Blau des Lesemodus-Kopfes gehalten.

### c.) Schriftarten

Für das Portal wurden unterschiedliche Schriftarten verwendet. Als Standardschriftart dient die Schriftart „Rockwell“, die auch als „Hausschrift“ [DesignManual] der Fachhochschule Darmstadt verwendet wird. Auch der Schriftzug der Titelleiste ist in dieser Schriftart verfasst. Da es sich allerdings bei Rockwell um keine Windows-Standardschriftart handelt, wurden ferner Verdana und Arial als zu verwendende Schriftarten mit angegeben. Die Menübeschriftungen und Überschriften sind dabei in Arial, feinere Texturen (Volltexte, Attribute, Relationen) hingegen in Verdana gehalten.

## 6.4 Das Rechtesystem / Profilgestaltung

Das Rechtesystem des Prototypen wurde auf die in Kapitel 5.4 vorgestellte Weise implementiert. Die Logindaten der einzelnen Nutzer werden, ebenso wie deren Zugriffsrechte innerhalb des Knowledge Builders definiert. Zu diesem Zweck muss zunächst das Rechtesystem eingerichtet werden. Abbildung 28 zeigt den entsprechenden Dialog im Knowledge Builder.

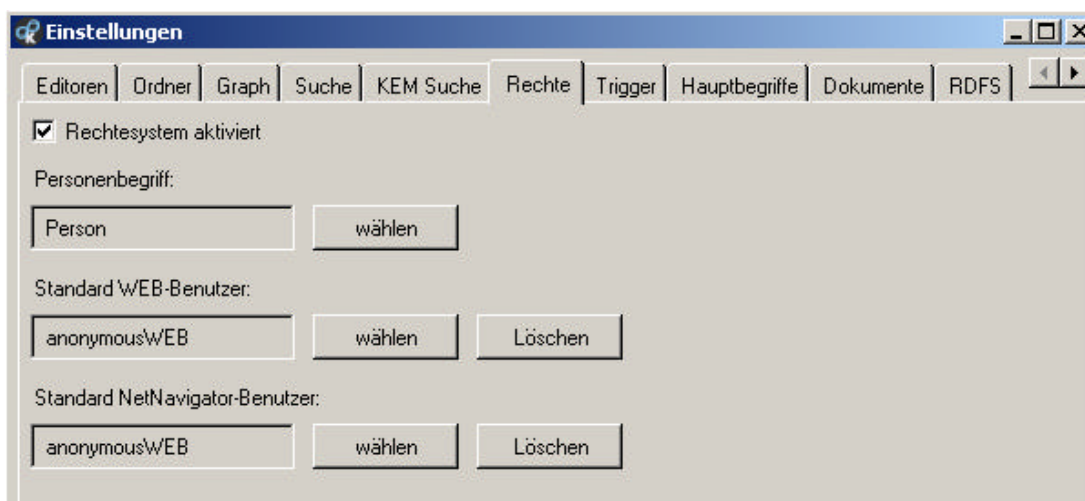


Abb.28: Aktivierung und Konfiguration des Rechtesystems

Innerhalb dieses Dialogs wird zunächst festgelegt, welcher Begriff die Individuen zur Profilbestimmung liefert (im Beispiel der Personenbegriff „Person“). Anschließend werden die Standardnutzer für den Webzugriff und den Net-Navigator bestimmt. Die differenzierte Definition zwischen Standard Web-Nutzer und Standard Net-Navigator-Nutzer liegt daran, dass die beiden Komponenten derzeit noch faktisch zwei verschiedene Bridges nutzen (Die „Multibridge“ für den Net-Navigator und die „KEM-Bridge“ für die Webanbindung). Planungen von „intelligent views“ zufolge soll in naher Zukunft eine Bridge (die KEM-Bridge) für beide Zugriffe zuständig sein.

Die Definition der Zugangspasswörter wird nach Aktivierung des Rechtesystems in den einzelnen Personenindividuen definiert. Hierzu existiert das Attribut „password“, welches nach Aktivierung des Rechtesystems automatisch angelegt wird.

Bislang sind im Prototypen zwei verschiedene Nutzertypen implementiert.

### **1.) Standardnutzer („anonymousWEB“)**

Der Standardnutzer ist der Nutzer, der auf den Prototypen zugreift, ohne eingeloggt zu sein. Derzeit hat der Standardnutzer keinerlei Schreibrechte und kann zudem auch nur eingeschränkt auf die Ontologie zugreifen. Aufgrund des Ontologienstadiums werden die Leserechte allerdings nur innerhalb der Top-Level-Konzepte der Ontologie beschränkt. Dazu wird ein boole'sches Attribut „ausblenden“ angelegt, welches angibt, ob ein Konzept für den Standardnutzer sichtbar ist oder nicht. Die Einschränkung mittels einfacher Expertensuche nach Konzepten ist innerhalb des Rechtesystems ebenfalls möglich, hat aber die Einschränkung von ganzen Taxonomien zur Folge. So würde eine Expertensuche nach Individuen von „T“ innerhalb eines einschränkenden Leserechts alle Individuen der Ontologie für den Nutzer unsichtbar machen. Da im Anwendungsbeispiel lediglich einzelne Konzepte ausgeblendet werden sollen, wird die Attributvariante hier vorgezogen.

Zusätzlich zu den eingeschränkten Leserechten werden dem Standardnutzer grundsätzlich alle Schreibrechte untersagt. So ist es erst nach erfolgter Registrierung und anschließendem Login möglich, Änderungen in das System einzupflegen. Die Startseite, die sich dem Nutzer bei Aufruf des Prototypen zeigt, ist in Abbildung 29 zu sehen.

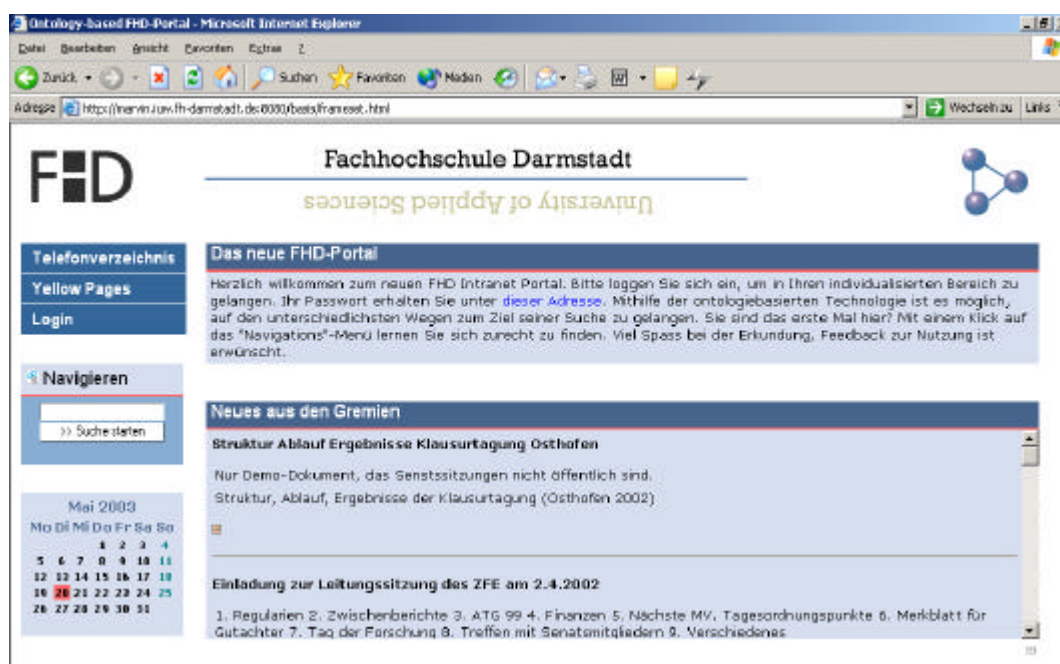


Abb.29: Portalsicht des Standardnutzers

Wie zu sehen, hat der Standardnutzer Zugriff auf das Telefonverzeichnis und die Yellow Pages der Menüführung (eine Erklärung der Punkte folgt in Abschnitt 6.5.2). Zusätzlich existiert ein Menüpunkt „Login“, der ihn auf die Loginseiten des Prototypen weiterleitet. Als zentrale Newsboxen erhält er einerseits eine allgemeine Informationsbox bezüglich des neuen Portals, zum anderen die neuesten Dokumente, die aus der Arbeit der Gremien heraus entstanden sind.

## 2.) Nutzerin Ute Jochem

Mit Anwählen des Menüpunktes „Login“ öffnet sich die Loginbox des Prototypen. Nach Eingabe der Nutzerdaten (Nutzername ist standardmäßig der Name des Individuums in der Ontologie, Passwort wird zunächst innerhalb des Knowledge Builders zugewiesen), gelangt der Nutzer zu seiner personalisierten Portalsicht. Im Anwendungsbeispiel von Frau Ute Jochem ist das die in Abbildung 27 gezeigte Sicht. Die Definition der zu zeigenden Menüs und Push-Dienste geschieht derzeit über die Webanbindung mithilfe des Struts-Frameworks. Dabei wird auf den Nutzer geprüft und abhängig davon verschiedene Dienste angezeigt. Im Rahmen des Projektauftrages an „intelligent views“ wird eine durch den Nutzer konfigurierbare Sicht entstehen. Im Gegensatz zum Standardnutzer hat Frau Jochem alle Schreibrechte im Bereich Forschung und Entwicklung der Ontologie. Die einzelnen Dienste des Portals werden im folgende Unterkapitel beschrieben.



## **6.5 Die Systemkomponenten**

Innerhalb des Loginbereichs werden in diesem Kapitel nun die einzelnen Komponenten des Prototypen beschrieben. Die Ausführungen beziehen sich dabei auf das Profil der Beispielnutzerin Ute Jochem und finden sich in Abbildung 27.

### **6.5.1 Die Push-Dienste**

Nach erfolgreichem Login gelangt der Nutzer in seinen personalisierten Bereich. Bereits auf der Startseite des Accounts sollten dabei die wichtigsten Informationen, die der Nutzer für seine tägliche Arbeit benötigt, verfügbar sein. Mithilfe von Push-Funktionalitäten (vgl. 5.2.4) werden dabei automatisch und profilabhängig die relevanten Informationen aus der Ontologie extrahiert und für den Nutzer visualisiert.

Dabei werden beim Portalprototyp der FHD zwei verschiedene Arten von Push-Diensten unterschieden:

#### **1.) Push-Dienste aus dem Ontologienkontext heraus**

Der einzelne Portalnutzer ist jeweils selbst in der Ontologie als Individuum des Konzepts „Person“ modelliert. Aus der Modellierung ergibt sich dabei ein bestimmter Informationsbedarf, abhängig vom Kontext der modellierten Person. So ergibt sich bei Frau Dr. Jochem ein Bedarf an Information primär bedingt durch ihre Arbeit beim Zentrum für Forschung und Entwicklung.

#### **2.) Konfigurierbare, interessenorientierte Push-Dienste**

Neben dem Informationsbedarf, der sich aus dem Kontext des modellierten Nutzers heraus ergibt, existiert darüber hinaus ein Informationsbedarf, der nicht zwangsweise in der Ontologie abgebildet sein muss. So könnten gewisse Interessen aus Hobbies, Freizeitaktivitäten oder ehrenamtlichen Tätigkeiten resultieren, die nirgendwo im semantischen Modell auftauchen. Die zweite Art der Push-Dienste soll daher durch den Nutzer frei konfigurierbar sein. Dafür wurde eigens eine Relation „interessiert sich für“ angelegt, die für die Auswertung der Ontologie genutzt wird. Dabei soll es zu einem späteren Zeitpunkt möglich sein, nicht nur das Zielkonzept anzugeben, sondern auch einen Entfernungsparameter, der die Anzahl der Knotenentfernungen weiterer Konzepte vom Zielkonzept angibt, die vom Nutzer als ebenfalls interessant erachtet werden. Ein Entfernungsparameter von zwei würde besagen, dass alle Konzepte, die bis zu zwei Relationen vom Zielkonzept entfernt sind, noch in die Auswertung der Pushfunktionalität mit einfließen. Die Maßnahme, eine solche konfigurierbare Pushfunktionalität anzubieten, liefert auch Infor-

mationen, die zu einem späteren Zeitpunkt genutzt werden können, etwa bei der Suche nach Experten<sup>29</sup>. Die freie Konfiguration der Push-Dienste ist in diesem ersten Prototypen noch nicht realisiert, da die Technologie erst zu einem späteren Zeitpunkt seitens der Firma „intelligent views“ geliefert wird. Daher wurden exemplarisch Relationen vom Individuum „Jochem, Ute“ zu anderen Konzepten der Ontologie gezogen (siehe Abbildung 30)

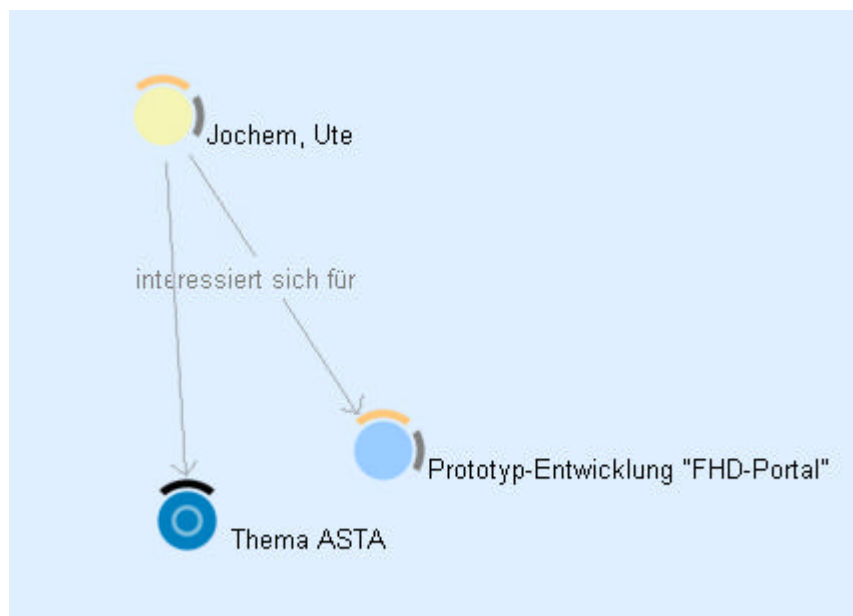


Abb.30: Interessensrelationen vom Individuum „Jochem, Ute“

Die Inhalte der jeweiligen Push-Dienste stammen aus im Knowledge Builder vordefinierten Expertensuchen (siehe Kapitel 5.2.4). Bei der Layoutgestaltung der Inhalte lassen sich drei unterschiedliche Sucharten unterscheiden: Nach Dokumenten, nach Dokumentteilen und schließlich nach Titeln von Wissensobjekten im Allgemeinen. Die Unterscheidung geschieht dabei allerdings nicht im Knowledge Builder, sondern beim Auslesen der Suchergebnisse mithilfe des Struts-Frameworks. So wird einmal das Dokument als Treffer direkt angezeigt, bei der zweiten Möglichkeit Titel und Lead des Dokumentes und bei der dritten Möglichkeit nur der Titel des Wissensobjektes. Wie in Abbildung 27 zu sehen, gibt es im Anwendungsfall „Mitarbeiterin des ZFE“ drei unterschiedliche Push-Dienste, die sich wie folgt einteilen lassen:

- Die Newsboxen in der oberen Bildschirmhälfte enthalten die für die tägliche Arbeit wichtigen Informationen, die sich aus dem Kontext des modellierten Nutzers erge-

---

<sup>29</sup> Interessiert sich ein Mitarbeiter für ein bestimmtes Thema, ist die Wahrscheinlichkeit, dass er sich mit diesem Thema auskennt, als höher zu erachten.

ben. Dabei ist es wichtig, Pushdienstfunktionalitäten bereits so präzise zu formulieren, dass keine intellektuelle Interaktion durch den Nutzer mehr vonnöten ist<sup>30</sup>. Im konkreten Nutzerfall ergeben sich darausfolgend die in der oberen Bildschirmhälfte zu sehenden Newsboxen „Neu eingegangene Förderanträge“ und „Forschungsprojekte“. Die erstere bezieht sich dabei auf den unter 6.9 noch zu behandelnden Use-Case. Formal werden hierbei Titel und Lead von enthaltenen Antragsdokumenten ausgelesen. Die zweite Newsbox beinhaltet eine Übersicht aller aktiver Forschungsprojekte an der FHD. Die formale Ausgabe umfasst hier lediglich die Bezeichner der Projekte (= Name der Projektinstanz).

- In der unteren Bildschirmhälfte hingegen findet sich eine Newsbox, die mit der Überschrift „Neuigkeiten“ betitelt ist. Bei genauem Blick auf die Inhalte dieser Newsbox stellt man fest, dass es sich hierbei um die Informationen aus der Ontologie handelt, die der Nutzer mithilfe der „interessiert sich für“-Relation frei bestimmt (vgl. Abbildung 28). Die Ausgabe der modellierten Ontologieteile geschieht hierbei im Volltext.

Es sei hier ausdrücklich darauf hingewiesen, dass es sich bei den in den Newsboxen ausgegebenen Elementen um gefilterte Wissensobjekte handelt. Durch Anklicken des orangenen Erweiterungspfeils bzw. des Titels (bei der Forschungsprojekt-Newsbox) gelangt man zur Ansichtseite des entsprechenden Wissensobjektes (siehe Abbildung 31).

Aufgabenbeschreibung ASTA	
Thema	Organisationsinformation
ist Dokument zu	Prototyp-Entwicklung "FHD-Portal"
<p>Der ASTa als Vertretung der Studierenden bzw. ausführendes Organ der Studierendenschaft hat so laut § 72 des Landeshochschulgesetzes folgende Aufgaben: Wahrnehmung der Belange der Studierenden in Hochschule und Gesellschaft; Vertretung der Interessen der Studierenden im Rahmen des Hochschulgesetzes; Mitwirkung an der Erfüllung der Aufgaben der Hochschulen, insbesondere durch Stellungnahmen zu hochschul- oder wissenschaftspolitischen Fragen; Förderung der politischen Bildung, des staatsbürgerlichen Verantwortungsbewusstseins und der Bereitschaft zur aktiven Toleranz der Studierenden auf der Grundlage der verfassungsmäßigen Ordnung; Wahrnehmung fachlicher, wirtschaftlicher und soziale Belange der Studierenden; Wahrnehmung kultureller Belange der Studierenden; Förderung des Studierendensports; Pflege überörtlicher und internationaler Studierendenbeziehungen. Für die genannten Aufgaben können Medien aller Art genutzt werden und in diesen Medien kann auch die Diskussion und Veröffentlichung zu allgemeinen gesellschaftspolitischen Fragen ermöglicht werden.</p>	

Abb.31: Vollansicht des Wissensobjektes „Aufgabenbeschreibung ASTA“

<sup>30</sup> Weniger präzise formulierbarer Informationsbedarf wird vorzugsweise über die Menüführung bedient, siehe Unterkapitel 6.5 .

Längerfristig soll ergänzend zur Nutzung der Push-Dienste die Möglichkeit der Verwendung von Trigger-Mechanismen berücksichtigt werden, die ereignisabhängig Newsdienste mit Inhalt füllen. Diese Möglichkeit ist allerdings in der aktuellen Knowledge-Builder Version in einem noch nicht nutzbaren Stadium.

## 6.5.2 Die Menüdienste

Die Menüdienste bilden den zweiten Teil des individualisierten und nutzerabhängigen Zugriffs auf das Wissensmodell. Während die Push-Dienste die relevanten Informationen „auf den ersten Blick“ zur Verfügung stellen, erfordern die Menüdienste einen aktiven Aufruf durch den Nutzer. Im Zuge der Gestaltung der Menüdienste fiel auf, dass „K-Infinity“ ein Ansprechen der vorgefertigten Suchen, welche die Basis der Menüdienste bilden, als Wissensobjekt nicht unterstützt<sup>31</sup>. Dies hat zur Folge, dass das profilabhängige Auslesen der Menüdienste **nicht** über das Rechtesystem in „K-Infinity“ gesteuert werden kann, sondern (auf unflexiblere Weise) über die Webanbindung definiert werden muss.

Die in Abbildung 27 zu sehenden Menüpunkte innerhalb des Nutzerszenarios lassen sich in zwei unterschiedliche Varianten unterteilen:

### a.) Zuschnitte der Ontologie ohne Nutzerinteraktion

Unter Zuschnitte der Ontologie ohne Nutzerinteraktion lassen sich zunächst die beiden Menüpunkte „Forschung an der FHD“ und „Neues aus den Gremien“ einordnen. Diese beiden Punkte sind wiederum im Knowledge Builder vorformulierte Suchanfragen, die in das Portal eingebunden werden. Der Zuschnitt der Ergebnisse, die beim Anwählen des jeweiligen Menüpunktes angezeigt werden, geschieht dabei nach wie vor ohne Einflussnahme des Nutzers. Im Vergleich zu der Realisierung der Push-Dienste werden diese allerdings nicht automatisch, sondern erst mit Anwählen des entsprechenden Menüpunktes ausgeführt<sup>32</sup>.

Der Menüpunkt „Forschung an der FHD“ filtert die Ontologie auf Wissensobjekte, die in Relation zu den Konzepten „Forschung an der FHD“ und „Hochschulforschung“ stehen. Zusätzlich werden alle Individuen zum Konzept „Forschungsprojekt“ in die Suchanfrage

---

<sup>31</sup> D.h. Es können keine Rechte auf Expertensuchen als Objekte, sondern nur auf deren Ergebnisse definiert werden.

<sup>32</sup> Insofern erfordert die Anwahl des Menüs eine Interaktion des Nutzers, der Zuschnitt der Ontologie geschieht jedoch ohne Einflussnahme des Anwenders bereits bei der Definition des Menüpunktes.

miteinbezogen. Der Menüpunkt „Neues aus den Gremien“ hingegen sucht nach Dokumenten zum Konzept „Gremiensitzung“, zum Konzept „Senat FHD“ und zum Konzept „Studentenparlament FHD“. Dabei werden die Relationen „ist Dokument zu“, „hat Thema“ und „hat Hauptthema“ (inklusive der Begriffsrelationen) ausgewertet. Bei beiden Menüpunkten wird die Ontologie demzufolge nach mehreren Gesichtspunkten zugeschnitten. Die Funktionalität der einfachen Expertensuche reicht in solchen Fällen nicht aus<sup>33</sup>. Daher wurden die einzelnen Suchanfragen mithilfe eines sogenannten „Union-Ordners“ miteinander verknüpft. Der Union-Ordner bildet die Vereinigungsmenge aller in ihm enthaltenen Suchanfragen. Abbildung 32 zeigt eine solche Unionsuche im Knowledge Builder.



Abb.32: Anlegen einer Vereinigungssuche zum Thema „Gremien“

Problematisch ist hierbei allerdings, dass die zum Entwicklungszeitpunkt aktuelle Version der „K-Infinity“-Software die Anbindung der Vereinigungsordner an das Prototyplayout noch nicht ermöglicht<sup>34</sup>. Daher muss sich eines Tricks bedient werden, welcher die Einbindung der Vereinigungssuchen in das Portallayout ermöglicht. Mithilfe einer weiteren Suchanfrage, die als Suchziel alle Objekte im Union-Ordner der Hauptsuche definiert, ist so eine Abfrage der Vereinigungssuche durchführbar. Die Prototypseiten lesen daher nicht die eigentliche Suche aus, sondern nur die aufrufende. Das Ergebnis der Suche ergibt sich bei Aktivierung des Menüpunktes und ist im Beispiel „Neues aus den Gremien“ in Abbildung 33 zu sehen. Eine zeitabhängige Filterung der Ontologie ist in der aktuellen Version dabei noch nicht möglich, wird aber im Zeitraum nach dieser Arbeit von „intelligent views“ nachgeliefert<sup>35</sup>.

<sup>33</sup> Expertensuchen erlauben nur „UND“-Verknüpfungen, während die hier beschriebenen Suchanfragen auf „ODER“-Verknüpfungen basieren.

<sup>34</sup> Dies ist dadurch zu begründen, dass das Java-Objekt „Union-Ordner“ noch nicht modelliert und daher mittels Struts nicht zugreifbar ist.

<sup>35</sup> Die Lieferung erfolgt im Zuge einer Definition von Lebenszyklen von Wissensobjekten.

Suchergebnisse	gefunden: 11 Objekte
zu: "Neues aus den Gremien"	
Struktur Ablauf Ergebnisse Klausurtagung Osthofen	
<a href="#">Einladung zur Leitungssitzung des ZFE am 2.4.2002</a>	
Einladung Senat, 22.1.2002	
Einladung Senat, 16.10.2001	
Protokoll (informell) Senatssitzung, 18.12.2001	
Senatsklausur Osthofen (2002) zum Thema "Wissensmanagement"	
Arbeiten und Diskutieren auf der Senatsklausur im Januar 2002	
Einladung Senat, 18.12.2001	
Einladung 1. Sitzung, Senatskom. Wissensmanagement (FHD), 26.2.2002	
Einladung 2. Sitzung Senatskommission Wissensmanagement	
Vor-Einladung zur 1. Sitzung, Senatskommission WM (FHD), Feb.2002	

Abb. 33: Inhalt des Menüpunktes „Neues aus den Gremien“

Auch der Menüpunkt „Marktplatz“ ist zu den Menüpunkten zu zählen, welche die Ontologie ohne Nutzerinteraktion filtern. Das dahinterliegende Modell wurde bereits in Kapitel 5.2.4 beschrieben und soll hier nicht näher erläutert, ein Blick auf die Anbindung im Prototypen jedoch gewährt werden. Die Realisierung des Marktplatzes ist dabei aufgrund des engen Zeitrahmens noch sehr provisorisch. Bei Aktivierung des Menüpunktes „Marktplatz“ wird eine vorgefertigte Suche ausgeführt, welche die Ontologie nach Konzepten mit der Erweiterung „Ware“ (vgl. 5.2.4) durchsucht. Als Ergebnis erhält der Nutzer eine Liste analog der in Abbildung 33 dargestellten. Mit Auswahl einer konkreten Ware (als erweitertes Individuum von Gegenständliches, vgl. 5.2.4), öffnet sich die Objektsicht des Angebots (Abbildung 34).

Dell Optiplex GX260	
<b>Thema</b>	<b>PC - System</b>
Arbeitsspeicher	512 MB
Festplatte	40 GB 20 GB
Grafikkarte	Nvidia G-Force IV 64 MB
Prozessor	P4 2 GHZ
<b>hat Peripherie</b>	Drucker: C60 Monitor: Iiyama 1536T
<b>Ware</b>	
Anzahl	15
Preis	27.000 EUR
<b>hat Ansprechpartner</b>	Mueller, Joachim
<b>wird angeboten von</b>	Fachbereich IuW (FHD)

Abb.34: Objektansicht eines Marktplatzangebotes

Dabei wird deutlich, dass nicht nur die eigentlichen Attribute und Relationen des PCs angezeigt werden, sondern auch diejenigen, die durch die Erweiterung „Ware“ hinzugefügt wurden. Die einzelnen Relationswerte sind wiederum als Hyperlinks zu den entsprechenden Wissensobjekten der Ontologie dargestellt. D.h. ein Klick auf den „Iiyama“-Monitor „1536T“ öffnet das zugehörige komplette Wissensobjekt und würde im Beispiel so Auskunft über die technischen Merkmale des Monitors geben.

Pflegeseiten für den Marktplatz werden aufgrund des fortgeschrittenen Zeitpunktes dieser Arbeit erst zu einem späteren Zeitpunkt realisiert werden.

### **b.) Zuschnitte der Ontologie mit Nutzerinteraktion**

Während bei den Menüpunkten unter a.) die Filterung der Ontologie ohne Einflussnahme des Nutzers vonstatten geht, hat der Anwender bei der zweiten Art der Menüfunktionen die Möglichkeit, aktiv auf die ihm präsentierten Ontologieinhalte Einfluss zu nehmen. Ein Zuschnitt der Ontologie mit Nutzerinteraktion findet im Prototypen bei den Menüpunkten „Telefonverzeichnis“ und „Yellow Pages“ statt. Die dem Telefonverzeichnis zugrundeliegende Expertensuche filtert die Ontologie nach Aktoren mit Attribut „Telefon-Buero“, die Yellow Pages nach Individuen von Aktor, die für etwas zuständig sind. Im Gegensatz zu den bisher gesehenen Expertensuchen wird hierbei dem Nutzer allerdings die Möglichkeit gegeben, das Namensattribut der Ziele über einen Benutzerdialog einzugeben. Der Vorteil der parametrisierbaren Expertensuche liegt darin, den Nutzer das Suchergebnis aktiv mitbestimmen zu lassen. Gleichzeitig wird die Suche allerdings vom Systementwickler bereits so weit eingeschränkt, dass das Suchergebnis bereits auf alle in Frage kommenden Objekte zugeschnitten ist. Bei der Suche in den Yellow Pages nach der Zuständigkeit „\*Technik\*<sup>36</sup>, erhält der Nutzer bspw. den „Leiter der Betriebseinheit Technik“ als Ergebnis, der für die Technik an der FHD zuständig ist.

Die Software „K-Infinity“ unterstützt allerdings die Integration von Union-Suchen bei parametrisierten Expertensuchen nicht. Auch die Umgehung in der in a.) vorgestellten Art und Weise führt leider nicht zum gewünschten Erfolg. Dies hat zur Folge, dass sowohl beim Telefonverzeichnis als auch (mit gravierenderem Effekt) bei den Yellow Pages, nur **eine** vorgefertigte Expertensuche berücksichtigt werden kann. Mit Fertigstellung der technischen Komponenten seitens der Firma „intelligent views“ sollten daher beide Menüpunkte um weitere Suchanfragen erweitert werden<sup>37</sup>.

---

<sup>36</sup> Die „\*“-Zeichen dienen bei der Suche als Trunkierungssymbole

<sup>37</sup> Bei den Yellow Pages wären Suchen denkbar, die nicht nur explizite Zuständigkeiten ausfindig machen, sondern auch implizite Hinweise für Zuständigkeiten berücksichtigen, wie bspw. „besetzt Stelle“ etc.

Eine Überprüfung der Zielgenauigkeit der dahinter liegenden Expertensuchen und der Modellierung der Ontologie sollte dann mithilfe von Kompetenzfragen (vgl. Kapitel 5.1) erfolgen. Eine Vorschlagsliste für Kompetenzfragen bezüglich der Yellow Pages befindet sich im Anhang.

### 6.5.3 Die Suchfunktionalitäten

In Kapitel 5.2.4 wurde als einer der großen Vorteile eines ontologischen Portalsystems die vielfältigen Zugänge zum Wissensmodell genannt. Ein Teilaspekt dieses Vorteils bilden die unterschiedlichen Suchmöglichkeiten, welche verschiedene Ansätze und unterschiedliche Erfahrungslevel der verschiedenen Nutzer mit berücksichtigen. In Abbildung 27 ist zunächst ein Menüpunkt „Navigieren“ zu sehen. Mit Aktivierung dieses Punktes öffnet sich der Net-Navigator, der bereits in 5.2.4 beschrieben wurde. Eine sinnvolle Sicht auf die Ontologie, abhängig von den Kenntnissen und auf den Bedarfs des Nutzers (im Beispiel wurden alle abstrakten Top-Level-Konzepte ausgeblendet) zugeschnitten, hilft dem Nutzer, mittels Navigation durch die Teiltaxonomien vor allem bei unpräzise formuliertem Informationsbedarf zum Zielobjekt vorzustoßen. Die Einschränkung der Sicht geschieht dabei mithilfe des in 6.4 vorgestellten Rechtesystems.

Unterhalb des Navigationsaufrufs befindet sich ein Eingabefeld zur Textsuche. Der dahinterliegende Suchmechanismus basiert auf der semantischen Kegelsuche (vgl. 5.2.4). Einstieg in den Prototypen wird schließlich der Nutzer stetig von dem Portal bei seiner Suche nach für ihn relevanten Objekten unterstützt. Die Themensuche, die abhängig von semantischen Relationen und deren Gewichtung verwandte Wissensobjekte vorschlägt (vgl. Kapitel 4.2.4), wird dabei im Knowledge Builder konfiguriert. Dazu bekommt jede Relation eine eigene Gewichtung („Basiswert für die Ähnlichkeit“), die ausschlaggebend für die Berücksichtigung bei der Vorschlagsvergabe ist. Abbildung 35 zeigt das Ergebnis der Themensuche zum Wissensobjekt „Präsidium FHD“.

Weitere Themen
Gremium (89.0 %)
Hochschule (84.0 %)
Präsidentenwahl (82.0 %)
Leitungsgremium (81.0 %)
Hochschulpräsident (77.0 %)

Abb. 35: Vorschläge der Themensuche



Die Prozentangaben hinter den vorgeschlagenen Wissensobjekten beschreiben dabei die Ähnlichkeitswerte der Konzepte, abhängig von der Relationskonfiguration im Knowledge Builder.

#### **6.5.4 Der Verlauf**

Der Verlauf zeigt die innerhalb der Nutzersession zuletzt betrachteten Wissensobjekte. Demzufolge wird diese Funktionalität erst angezeigt, nachdem mindestens ein Wissensobjekt aufgerufen wurde. Ziel des Verlaufs ist dabei, den Nutzer hinsichtlich der Orientierung innerhalb des Portals zu unterstützen (siehe hierzu auch „die 5 Ws der Selbstbeschreibungsfähigkeit“ unter Kapitel 5.5). Vor allem innerhalb einer Suche nach einem bestimmten Konzept hilft diese Funktionalität dem Nutzer beim Nachvollziehen seiner Suchstrategie. Dies hat sich in der Praxis bezüglich der Erlernung des Umgangs mit neuen Systemen als nützliche Komponente im Rahmen des „Learning by doing“-Ansatzes (siehe Kapitel 5.5) bewiesen, da der Anwender durch Reflektion seiner Suchanfragen seine Zugriffsstrategien auf das Portal stetig verbessern kann.

#### **6.5.5 Der Kalender**

Der Kalender im Prototypen des FHD-Portals ist eine Javascript-Anwendung, die den aktuellen Monat und den aktuellen Tag, entsprechend den Datumseinstellungen des Clients<sup>38</sup> ausgibt. Die eigentliche Funktionalität steckt allerdings dahinter: Die einzelnen Kalendertage sind als Links realisiert, die auf die Funktion „Hochschulweiter Kalender“ des Intranets unter Intrexx verweisen. Mit Auswahl eines beliebigen Tags öffnet sich ein Fenster welches den angewählten Tag im Hochschulkalender mit den jeweiligen Terminen anzeigt. Die Funktionalität ist nur intern aus dem IP-Bereich der Fachhochschule aufrufbar. Das Intrexxsystem verlangt dabei, abhängig von der aufgerufenen Anwendung, ein teilweise mehrfaches Setzen von Cookies. Ein direkter Aufruf der Kalenderfunktion ist daher nicht möglich, da das notwendige Cookie dafür fehlen würde. Um dennoch direkt auf die Kalenderfunktion von Intrexx zugreifen zu können ohne die notwendigen Cookies simulieren zu müssen, wird beim Aufbau der Portalseiten ein einpixelbreiter Frame erzeugt, der die notwendige Intrexxseite aufruft, die Voraussetzung für den späteren Kalenderaufruf ist. Abbildung 36 zeigt den Blick auf den hochschulweiten Kalender.

---

<sup>38</sup> Längerfristig wäre es empfehlenswert, auf eine serverseitige Lösung umzusteigen, um die Datumsanzeige konsistent zu halten. Der Einfachheit halber wurde hier mit Javascript die clientseitige Variante gewählt.



Abb.36: Tagesansicht des hochschulweiten Kalenders unter Intrex

## 6.6 Use Case: Beantragung von Fördergeldern für ein Forschungsprojekt

In diesem Kapitel wird ein Use-Case herausgearbeitet, der die Möglichkeit der Unterstützung und Optimierung von Prozessen durch ein ontologisches Portal belegen soll. Die Zusammenhänge aus dem Bereich Forschung wurden dabei in Gesprächen mit Frau Ute Jochem vom Zentrum für Forschung und Entwicklung ausgearbeitet.

### 6.6.1 Prozessbeschreibung

Aufgabe des Zentrums für Forschung und Entwicklung ist u.a. die Annahme und Bearbeitung von Anträgen für Fördergelder interner Forschungsprojekte. Bei der Vergabe von Fördergeldern an der Fachhochschule Darmstadt gibt es dabei verschiedene Geldtöpfe zur Finanzierung unterschiedlicher Belange. So gibt es bspw. Mittel aus den sogenannten „HWP-Töpfen“<sup>39</sup>, die ausschließlich für Personalkosten genutzt werden und oftmals zur Finanzierung „frischer“ Diplomabsolventen in Forschungsprojekten der Fachhochschule genutzt werden.

<sup>39</sup> HWP: Hochschul- und Wissenschaftsprogramm

Darüber hinaus unterscheidet man interne Fördergelder von Geldern externer Institutionen (Beispiel: „Deutsche Forschungsgesellschaft“ oder „VW-Stiftung“). Im Zuge des Use-Cases wird sich auf die Fördertöpfe beschränkt, deren Gelder direkt beim ZFE beantragt werden können. Im speziellen sind das:

- Gelder aus dem Wirtschaftsfond der Fachhochschule
- HWP-Mittel
- Gelder vom Hessischen Ministerium für Wissenschaft und Kunst (HMWK)<sup>40</sup>
- Gelder vom Hessischen Ministerium für Bildung und Forschung (HMBF)

Die Beantragung der Fördergelder ist dabei abhängig von der Projektgröße. So sind Kleinprojekte mit einem Budget von bis zu 13.000 Euro von Anschubprojekten (über 13.000 und bis 25.000 Euro) und Großprojekten mit einem Budget von über 25.000 Euro zu unterscheiden. Auch die Zielsetzung der einzelnen Projekte unterscheiden sich. So sollen Kleinprojekte einer größeren Zahl von Professorinnen und Professoren „die Möglichkeit zum wissenschaftlichen Arbeiten in einem überschaubaren Rahmen“ [Kleinprojekt] geben. Anschubprojekte hingegen sollen die Möglichkeit geben, „eine Diplomarbeit wissenschaftlich abzuschließen“ [Anschubprojekt] oder als Anschubfinanzierung für Forschungsansätze dienen. Auch die Organisation und Abhaltung eines wissenschaftlichen Kongresses kann als Anschubprojekt finanziert werden.

Großprojekte schließlich sind Projekte mit einer Dauer von bis zu 24 Monaten und dienen der Verwirklichung größerer Forschungsmaßnahmen.

Abhängig vom Förderumfang gelten unterschiedliche Richtlinien, die bei der Beantragung von Fördergeldern berücksichtigt werden müssen. Dabei muss laut ZFE jeder Projektantrag unabhängig vom beantragten Fördervolumen die folgenden Angaben enthalten:

- 1.) Eine kurze und allgemeinverständliche Zusammenfassung von einer halben bis ganzen Seite, welche Auskunft über Bedeutung des Projektes, das Projektziel und die Projektteilnehmer gibt.
- 2.) Eine detaillierte Beschreibung des Projektes hinsichtlich Stand der Forschung und Arbeitsprogramm.
- 3.) Eine Bibliographie als Ergebnis einer projektbezogenen Recherche
- 4.) Auskunft über Kooperationsbeziehungen mit Unternehmen, anderen Fachbereichen und sonstigen Partnern

---

<sup>40</sup> Über die Bewilligung von HMWK-Geldern entscheidet das HMWK selbst. Der Antrags- und Gutachterprozess wird allerdings vom ZFE durchgeführt.

- 5.) Informationen zu den beantragten Mitteln (Verwendungszweck, Finanzierungsplan, zugesagte Drittmittel)
- 6.) Aussagen über Folgeperspektiven
- 7.) Information über die Förderung von Mitarbeiterinnen und Mitarbeitern

Ein Antrag für Fördergelder für ein Anschubprojekt sollte zusätzlich einen Gutachter benennen, der das Projekt bewertet, während für den Antrag für Fördergelder bezüglich eines Großprojektes mindestens zwei Gutachter benannt werden müssen, von denen einer extern sein muss, d.h. kein Mitglied der Fachhochschule sein darf. Zusätzlich dazu muss der Antrag für ein Großprojekt Auskunft über die fachliche Qualifikation des Antragstellers geben.

Die Vorgehensweise bei der Beantragung von Fördergeldern ist dabei die folgende:

Die Beantragung durch den Projektleiter geschieht, wie eben bereits angesprochen, abhängig vom Volumen des Projektes zumeist zielgerichtet auf Gelder eines bestimmten Fördertopfes. Nach der Beantragung werden die entsprechenden Gutachten gestellt. Kleinere Projekte erhalten dabei nur einen FH-internen Gutachter, während für Großprojekte mindestens zwei Gutachten erstellt werden. Über die Förderungsanträge kleiner Projekte entscheidet die Leitung des ZFE selbständig (wenn vorhanden) auf der Grundlage der erstellten Gutachten, Großprojekte werden, abhängig von der Mittelherkunft entweder vom HMWK oder einer externen Organisation, wie der Deutschen Gesellschaft für Forschung beschieden.

Größere Projekte erfordern nach Gewährung der Mittel einen jährlichen Zwischenbericht, auf dessen Grundlage über Mittelbewilligung für ein weiteres Jahr entschieden wird. Zusätzlich dazu gibt es für jedes Forschungsprojekt, unabhängig von der Größe, einen Abschlussbericht am Ende der Projektlaufzeit.

### **6.6.2 Realisierung der Antragsstellung mithilfe des FHD-Portals**

Mithilfe des Prototypen ist es möglich, Anträge als Dokumentdateien direkt in die Ontologie zu laden und für die bearbeitende Person (im Anwendungsbeispiel Frau Ute Jochem) verfügbar zu machen. Die einzelnen Schritte sollen im folgenden erklärt werden. Um den Use Case inhaltlich logisch zu gestalten, wird ein dritter Nutzer für den Prototypen angelegt (Prof. Dr. Gerhard Knorz), innerhalb dessen Profil die Beantragung der Fördergelder demonstriert werden soll. Eine detailliertere Ausgestaltung des Profils erfolgt dabei zunächst nicht.

### 1.) Anlegen eines neuen Forschungsprojektes

Über die Suche nach „Großprojekt“ findet der Einstieg statt. Nach Aufruf des zugehörigen Wissensobjektes in Abbildung 37 und Anklicken des „Neu“-Buttons kommt der Anwender zu der Eingabemaske eines neuen Forschungsprojektes, die in Abbildung 38 gezeigt wird.

Thema	Forschungsprojekt
<input type="button" value="neu"/>	Forschungsprojekt
Begriffsdefinition	Großprojekte sind Forschungsvorhaben mit einem Etat über 50.000 DM und einer Dauer von maximal 24 Monaten.
Synonym	Forschungsprojekt mit großem Volumen

Wie beantrage ich ein Großprojekt?

Wie beantrage ich ein Großprojekt?

Abb. 37: Wissensobjekt „Großprojekt“

**Großprojekt**

Name:

**Mögliche Attribute**

Adressergänzung (Adressierb)

Ansichtsbild (--)

Auffindunterstützung

Begin (Zeit)

Beginn (Tag)

Definition

**Mögliche Relationen**

Aktivität bedient sich an

hat Mitglied

Abb. 38: Anlegen eines neuen Forschungsprojekts

Neben der Eingabe eines Projektnamens hat der Nutzer die Möglichkeit, verschiedene Attribute anzugeben bzw. Relationen zu anderen Ontologiekonzepten hin zu ziehen. Die Eingabe der Relationen geschieht dabei unter Zuhilfenahme des Net-Navigators. Durch Anwählen des „neu“-Buttons öffnet sich dieser und bietet die visuelle Auswahl der Ziel-

konzepte der Relation zusätzlich zu einem Eingabefeld. Dabei wird überprüft, ob das ausgewählte Konzept ein für die Relation gültiges darstellt.

## 2.) Förderantrag für das Projekt stellen

Nach Ausfüllen der Attribute und Relationen hat der Anwender die Möglichkeit einen Antrag auf Fördergelder zu stellen. Durch Aktivierung des „Antrag stellen“-Buttons gelangt er zum Upload-Formular (Abbildung 39).



Abb. 39: Formular zum Upload eines Projektantrags

Wie zu sehen, variieren hierbei die Formulare abhängig von der Art des zu fördernden Projektes (im Beispiel handelt es sich um ein Upload-Formular für Förderanträge hinsichtlich von Großprojekten). Die Überlegung, den Antrag direkt über ein Formular mit den Angaben aus 6.5.1 entsprechenden Eingabefeldern in die Ontologie einzutragen (im Gegensatz zur aktuellen Variante, den Antrag hoch zu laden und mit Metainformationen zu versehen) wurde aus zweierlei Gründen verworfen: Zum einen sind die Anträge so umfangreich, dass eine formularbasierte Eingabe eher unpraktisch wäre, zum anderen wäre die Weiterleitung an dritte Personen, die keinen Zugriff auf das Portal haben in der Art erschwert, dass aus den Wissensobjekten der Ontologie (in die sich ein solcher Antrag dann aufteilen würde) ein Dokument erzeugt werden müsste.

Neben der Eingabe der notwendigen Informationen (derzeit sind exemplarisch die Attribute „Erstellungsdatum“ und „Zusammenfassung“ realisiert) und Auswahl der Datei hat der Nutzer darüber hinaus die Möglichkeit, den Antrag thematisch mit Wissensobjekten der Ontologie zu verknüpfen. So könnte ein Antrag zum Projekt „Ontologien für das Wissensmanagement“ thematisch mit dem Konzept „Wissensmanagement“ verknüpft werden und so unabhängig von seinem Dokumenttyp (= „Antrag“) zur Wissensquelle für Su-

chende werden. Zusätzlich kann er über eine Relation namens „beantragte Mittel“ den Fördertopf, auf den sich der Antrag bezieht, auswählen. Dazu wurden exemplarisch Individuen von „Fördertopf“ angelegt. Dieses Konzept ist Subkonzept von „Forschungsförderung an der FHD“.

Mit dem Hochladen des Dokumentes wird dieses als Individuum zum entsprechenden Dokumentenbegriff („Antrag für ein Kleinprojekt“, „Antrag für ein Anschubprojekt“ oder „Antrag für ein Großprojekt“) in der Ontologie gespeichert. Dieses Dokument bekommt unter Anderem ein Attribut namens „Status“ eingetragen. Dieser Wert wird mit Absenden des Formulars automatisch auf „gestellt“ gesetzt. Der Status der Fördermittelbeantragung wird hierbei bewusst nicht im Forschungsprojekt festgehalten. Die Begründung hierfür ist darin zu finden, dass bei einer Ablehnung seitens des ZFE der Antragsteller immer noch die Möglichkeit hat, sich bei externen Förderinstitutionen um Fördermittel zu bewerben<sup>41</sup>.

Nachdem alle Eingaben getätigt und übernommen sind, gelangt der Nutzer über den „zurück“-Button zurück zur Eingabemaske des neuen Projektes. Als Unterschied zu Abbildung 39 wird nun der soeben hoch geladene Antrag angezeigt. Zusätzlich gibt es den Knopf „Antrag stellen“ nicht mehr, stattdessen zwei andere Buttons. Einen mit der Beschriftung „weiteren Antrag stellen“ und einen mit der Beschriftung „weiteres Dokument hochladen“. Der erstere dient der Beantragung weiterer Fördergelder, unabhängig vom Ausgang des ersten Antrages auf Fördermittel. Über den zweiten Button gelangt der Nutzer zu einem weiteren Formular, welches allerdings nicht den Upload von Anträgen, sondern das Hochladen von Dokumenten bezüglich der Projektdokumentation ermöglicht. Mithilfe dieser zweiten Möglichkeit, Dokumente in die Ontologie zu laden, wird der Projektleiter in seiner Dokumentationspflicht (Zwischenberichte und Abschlussbericht) unterstützt. Vor allem bei Projekten mit mehreren Jahren Laufzeit muss pro Jahr ein Zwischenbericht abgeliefert werden, auf dessen Basis über die Weiterführung der Projektförderung nachgedacht wird. Der Upload der Dateien geschieht analog zum eben gesehenen Vorgang, nur wird das Dokument als Instanz des Dokumentenbegriffs „Dokumentation“ eingebunden.

---

<sup>41</sup> Streng genommen hat er auch die Möglichkeit, sich nochmals für ein internes Projekt zu bewerben, die Chance bei unverändertem Projekt auf eine Gewährung von Fördermitteln bei einem zweiten Antrag stehen allerdings eher schlecht.

### 6.5.3 Verfügbarmachen der neuen Anträge

Die neu gestellten Anträge werden auf unterschiedliche Weise verfügbar gemacht. Über die in 6.4.3 behandelten Suchmöglichkeiten sind die Anträge als Objekte der Ontologie zugreifbar. Zusätzlich dazu sind innerhalb des Nutzerszenarios „Ute Jochem“ die Anträge auf Fördergelder als Push-Dienst realisiert. Abbildung 40 zeigt die zugehörige Newsbox. Wie zu sehen, wird dabei der Name des Antrags und die Kurzzusammenfassung ausgegeben.

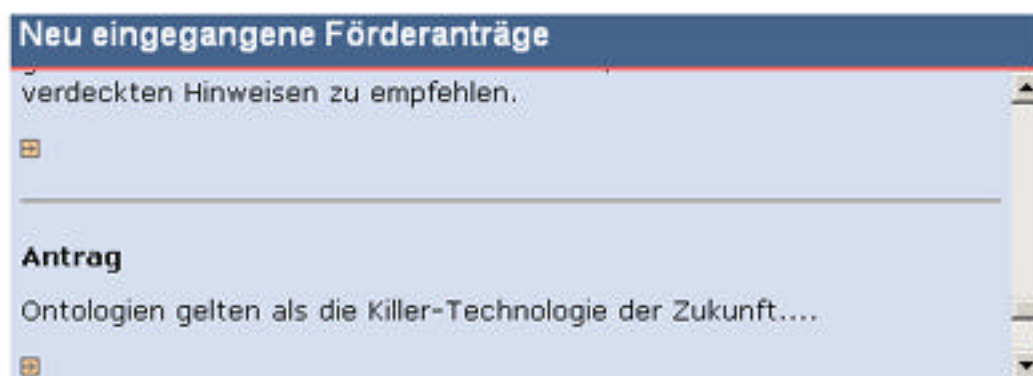


Abb. 40: Newsbox „Neu eingegangene Forschungsanträge“

Frau Jochem, zuständig für die Koordination der Fördermittelanträge, hat nun die Möglichkeit, durch Anwählen des orangenen Pfeils auf die Komplettansicht des entsprechenden Antrags zu gelangen, um so auf das Dokument zuzugreifen. Über Wechsel in den Editiermodus des Forschungsprojektes gelangt sie dann zu dem in Abbildung 39 gezeigten Formular. Im Unterschied zur Sicht des Antragstellers existiert nun allerdings ein Auswahlfeld „Status“, dessen Wert entsprechend des Entscheidungsstandes über Bewilligung von Fördermitteln auf „genehmigt“ oder „abgelehnt“ gestellt werden kann.

Der kommunikativen Aspekt der Informationsversorgung des Antragstellers über eine Entscheidung hinsichtlich seines Antrages, könnte nun über das Nutzen eines Trigger-Mechanismus bedient werden. So könnte der Antragsteller automatisch eine e-Mail mit den entsprechenden Informationen erhalten, sobald das Leitungsgremium des ZFEs über seinen Antrag entschieden hat. Da diese Funktionalität in der aktuellen „K-Infinity“- Version noch nicht ausreichend realisiert ist, muss eine Implementierung eines solchen Dienstes auf einen Zeitpunkt nach dieser Arbeit verschoben werden<sup>42</sup>.

---

<sup>42</sup> Als Alternative könnte zunächst über die Verfügbarmachung der Antragsinformationen innerhalb eines Push-Dienstes nachgedacht werden.



## 7. Diskussion der Ergebnisse

### 7.1 *Konzeption*

In Kapitel 1 dieser Arbeit wurde festgestellt, dass ein umfassendes Wissensmanagement folgende Bausteine des inneren und äußeren Wissenskreislaufes berücksichtigen muss:

- Wissensziele
- Wissensidentifikation
- Wissenserwerb
- Wissensentwicklung
- Wissensverteilung
- Wissensnutzung
- Wissensbewahrung
- Wissensbewertung

Während die Bausteine des äußeren Kreislaufs (Wissensziele und Wissensbewertung) als Ergebnisse von Managementprozessen und –strategien ausgefüllt werden müssen, finden sich die Bausteine des inneren Kreislaufes in der Konzeption des FHD-Portals in Kapitel 5 wieder. Die vorgestellte Methodik zur Entwicklung eines ontologiebasierten Wissensmanagementsystems berücksichtigt dabei alle Aspekte, um Prozesse und Abläufe zu erfassen und an der Fachhochschule vorhandene Systeme in ein einheitliches Datenmodell zu integrieren. Auch die in Kapitel 3 herausgearbeiteten Anforderungen an ein Wissensmanagement für die Fachhochschule als spezieller Organisationstyp, sowie der festgestellte Bedarf der Mitarbeiter der FHD, wurden in der Konzeption berücksichtigt. Hinsichtlich Nutzerakzeptanz wurden verschiedene Zugriffsmodelle entwickelt, die den Nutzer abhängig seiner Vorkenntnisse mit den für ihn relevanten Informationen versorgen. Darüber hinaus wurden Ansätze für ein Kommunikationsmodell erarbeitet, welche auf sämtlichen Bausteinsebenen die Unterstützung der Aufgaben durch effiziente Kommunikation ermöglichen.

Mithilfe der vorgestellten Methodologie zur Evaluierung und Weiterentwicklung der vorhandenen Ontologie ist es ferner möglich, das vorhandene Wissensmodell kriterienabhängig zu evaluieren und zu straffen und um noch fehlende Komponenten (vor allem auf der Instanzbasis) auf Basis von formal eindeutigen Merkmalen zu erweitern.

Es sei darauf hingewiesen, dass bis zur endgültigen Einführung des Portals noch viele hochschulinterne Prozesse und Abläufe erfasst und ausgewertet werden müssen. Hierbei wird es besonders wichtig sein, den Mitarbeiter frühest möglich in die nun folgende nächste Entwicklungsphase einzubinden, um so frühzeitig die Notwendigkeit und den Nutzen des Portals zu vermitteln. Das Portal kann innerhalb einer Einführung von Wissensmanagement nur technischer Wegbereiter sein, die Motivation des einzelnen Hochschulmitgliedes muss durch ein Umdenken der Hochschulleitung und das Vorleben einer neuen Informations- und Kommunikationskultur erreicht werden.

## **7.2 Prototyp**

Im Gegensatz zu vielen sonst in der Literatur gegebenen Ansätzen für ein ontologiebasiertes Wissensmanagementsystem, liefert diese Arbeit ein direktes Umsetzungsbeispiel des erarbeiteten Konzeptes. Anhand eines Nutzerszenarios werden die in den vorangehenden Kapiteln erarbeiteten Vorteile eines ontologiebasierten Systems demonstriert und es werden Möglichkeiten der Gestaltung aller operativen und organisatorischen Ebenen des Wissensmanagements aufgezeigt. Aufgrund der knappen Entwicklungszeit von weniger als einem Monat sind die Funktionalitäten des Prototypen zwar noch sehr exemplarisch realisiert, die Mächtigkeit des Ansatzes wird dennoch sichtbar.

Anhand der Realisierung des Use-Cases wird darüber hinaus ersichtlich, inwiefern das spätere Portal sowohl zur Unterstützung interner Prozesse (im Use-Case: Die Antragstellung auf Fördergelder hinsichtlich eines Forschungsprojektes), als auch zur dezentralen Wissensakquise mithilfe der Vergabe von Rechten und der Bereitstellung von Pflegeseiten genutzt werden kann.

Der Prototyp entspricht weitestgehend den aktuellen Ansätzen ergonomischer Webapplikationen. Intuitive Benutzerführung und erwartungskonforme Dialoggestaltung sind bereits in dieser frühen Phase der Entwicklung erkenntlich, wenn auch gerade im Bereich von Fehlerbehandlung und Ausgabe von Fehlermeldungen sicherlich noch ausbaufähig. Das endgültige Layout des späteren Portals wird ohnehin im Anschluss an diese Arbeit vom Fachbereich Gestaltung entwickelt.

Insgesamt sollte eine Nutzung des Systems im „Learning by doing“-Verfahren dem Anwender jedoch ohne einschlägige Vorkenntnisse möglich sein.

Problematisch bei der Prototypgestaltung erwies sich, dass sich alle drei Komponenten, die zur Umsetzung des Prototypen verwendet wurden („K-Infinity“-Software, Struts/Skat-Anbindung und Ontologie), selbst noch in mehr oder weniger fortgeschrittenen Entwick-

lungsstadien befanden. Abstürze der Software-Suite hielten ebenso auf, wie noch nicht realisierte Anbindungsmöglichkeiten oder fehlende Modellierungen. Vor allem erstere Punkte spielten eine gewichtige Rolle, während Modellierungslücken in Einzelfällen exemplarisch nachgezogen werden konnten.

Aufgabe muss es nun sein, durch ausgiebige Evaluierung des Vorhandenen und durch Testen des Prototypen durch einzelne Mitarbeiter, Verbesserungsmöglichkeiten zu entdecken und gleichzeitig durch Erweiterung der Nutzerszenarien, einhergehend mit einer stetigen inhaltlichen Erweiterung der Ontologie, das Modell und den Prototypen zu ergänzen (siehe Kapitel 8).

### **7.3 Modellierung mit der „K-Infinity“-Software Suite**

Die „K-Infinity“-Software ist im Gesamtblick als benutzerfreundlich und mächtig zu bewerten und daher für die Entwicklung von Ontologien zu empfehlen. Ein umfassender Formalismus und der effiziente Umgang auch mit großen Datenmengen und umfangreichen Ontologien ist innerhalb der Anwendung vorbildlich und besser als bei Konkurrenzprodukten gelöst.

Allerdings existierten in allen während dieser Arbeit verwendeten Versionen Fehler, die zeitweise die Arbeit mit der Software nicht unerheblich erschwerten. Systemabstürze waren dabei oftmals an der Tagesordnung. Hinsichtlich einer endgültigen Marktreife sollte daher von Seiten der Firma „intelligent views“ die Zuverlässigkeit der Software verbessert werden. Wenigere aber dafür umfassende Updates wären vielen kleineren ebenfalls vorzuziehen.

Ein weiteres Problem bei der Nutzung der „K-Infinity“-Software-Suite während der gesamten Arbeit war die oft mangelhafte Dokumentation der Funktionalitäten. Ein der Version entsprechendes Handbuch des „Knowledge Builders“ existierte bis zuletzt nicht, einige der Funktionalitäten mussten entweder durch „Exploration“ oder ständiges Rückfragen geklärt werden.

Auch in den Möglichkeiten der Anbindung der Ontologie an ein Webfrontend bestanden von Seiten der Firma „intelligent views“ Mängel auf den Bereichen der Dokumentation, vor allem hinsichtlich der Beschreibung der proprietären SKAT-Tags. Ferner zeigte sich während der Anbindung, dass einige der im „Knowledge Builder“ vorhandenen Funktio-

nalitäten nur umständlich oder in Einzelfällen noch gar nicht im Webfrontend genutzt werden konnten. Als Beispiel wäre hier die Möglichkeit der Bildung von Vereinigungssuchen zu nennen (als „UND“-Verknüpfungen einzelner Suchanfragen auf das ontologische Modell), die im Knowledge Builder prinzipiell möglich, in der Anbindung allerdings nur durch Umwege oder, im Falle einer Parameterübergabe, gar nicht realisierbar war.

Der in der Knowledge-Builder Version 1.3, die im Zuge dieser Arbeit eingeführt wurde, erstmalig angebotene RDF/RDF(S)-Import bzw. Export ist der erste Schritt in Richtung einer späteren Plattformunabhängigkeit und Mehrfachnutzung auf externen Systemen. Unter Zuhilfenahme von Ontologieeditoren wie Protégé oder Ontoedit wäre man so ebenfalls in der Lage, das Wissensmodell zu bearbeiten. Auch der Import bestehender Ontologien<sup>43</sup>, vor allem auf dem Top-Level-Bereich (siehe 2.3.1), ist hiermit ebenfalls erstmalig möglich. Eine Weiterentwicklung dieser Möglichkeiten in Richtung des „DAML+OIL“-Formats<sup>44</sup>, würde eine komplette Abbildung des „K-Infinity“-internen, framebasierten Modells in einem systemneutralen Format ermöglichen und die Möglichkeiten der Integration bestehender Ontologien deutlich verbessern.

---

<sup>43</sup> Im Internet existieren Bibliotheken mit RDF/RDF(S) -Ontologien, siehe bspw. <http://www.semanticweb.org/library/> mit der Wordnet-Ontologie im RDF-Format

<sup>44</sup> Bei dem DAML+OIL Format handelt es sich um eine Erweiterung von RDF/RDF(S), die reicher an Möglichkeiten bezüglich der Modellierung ist (siehe [Daml]).

## 8. Zusammenfassung und Ausblick

### 8.1 Zusammenfassung

Das ontologische Portal der Fachhochschule Darmstadt soll als zentrales Wissensmanagementsystem helfen, Mitarbeitern einen „single point of access“ zu allen in der Fachhochschule vorhandenen Informationen zu gewähren. Die Optimierung hochschulinterner Prozesse und das Erlangen von Transparenz über vorhandenes Wissen und verfügbare Informationen, sowie eine Integration derselben gelten dabei als Schlüsselaspekte des Systems.

In dieser Arbeit wurde zunächst eine Konzeption für ein ontologiebasiertes Portal der Fachhochschule Darmstadt ausgearbeitet. Basierend auf Betrachtungen gängiger Wissensmanagementpraktiken wurde dazu zunächst eine wissenschaftliche Basis formuliert, die den Rahmen für die spätere Konzeption der Portalapplikation bildet. Zusätzlich wurde der Bedarf der Mitarbeiter der Fachhochschule ermittelt, der als Ansatzpunkt für die inhaltliche Gestaltung der Konzepte dient. Die Konzepte umfassen dabei alle relevanten Aspekte eines modernen Wissensmanagements, die, strategisch durch das Management erweitert, die einzelnen Wissensprozesse ausfüllen.

Darüber hinaus wurde eine Methodologie entwickelt, welche eine Überprüfung des zu Beginn dieser Arbeit bereits existierenden Wissensmodell ermöglicht. Zur Modellierung der Ontologie wurde dabei die Software-Suite „K-Infinity“ der Firma „intelligent views“ verwendet, welche einen framebasierten Repräsentationsformalismus nutzt.

Im Zuge einer Prototypisierung wurden anschließend die erarbeitenden Konzepte exemplarisch umgesetzt und durch Definition eines beispielhaften Nutzerprofils mit daraus resultierendem Informationsbedarf die Machbarkeit praktisch belegt. Anhand eines Use-Cases aus dem Bereich Forschung- und Entwicklung wurden die Ansätze hinsichtlich der Möglichkeiten einer Prozessunterstützung durch das ontologiebasierten Portal dargestellt, die unterschiedlichen Zugangsweisen zur Wissensbasis bestätigten dabei die Flexibilität des ontologischen Ansatzes. Die Anbindung der Ontologie wurde unter Zuhilfenahme des Struts-Frameworks realisiert, einem Projekt der Apache Jakarta Gruppe.

## 8.2 *Ausblick*

„Knowledge Management is expensive! [...] But so is stupidity“

Thomas H. Davenport [Davenport]

Bei der Weiterentwicklung des Prototypen und hinsichtlich der endgültigen Portaleinführung kann die Fachhochschule auf ein umfassendes Know-How zurückgreifen, das es richtig zu nutzen gilt. In verschiedensten Fachbereichen sollte über eine Einbindung der Portalentwicklung in studentische Projekte nachgedacht werden. Am Fachbereich „Informations- und Wissensmanagement“ könnte so die Ontologie weiterentwickelt und umfassend überarbeitet werden, der Fachbereich Informatik hätte die Ressourcen, die Realisierung der Anbindung der Ontologie innerhalb des Struts-Frameworks zu übernehmen und der Fachbereich Gestaltung (oder auch Media System Design) schließlich könnte sich über die (Weiter-)Entwicklung eines Portallayouts an dem Projekt beteiligen. Wichtig wird hierbei sein, den Nutzen des späteren Portals für alle Fachbereiche der FHD zu verdeutlichen und daraus resultierend die einzelnen Fachbereiche „in die Pflicht“ zu nehmen. Eine interne Marketingkampagne (unter Umständen als Studienprojekt des Fachbereichs Wirtschaft durchgeführt) könnte bei dieser Zielsetzung unterstützend mitwirken. Im Zuge der Weiterentwicklung wird es ferner eine der nächsten Aufgaben sein, den Prototypen, der derzeit noch auf einem Server des Fachbereichs Informations- und Wissensmanagement gehostet wird, auf einen eigenen Server zu transferieren. Eine Empfehlung bezüglich der Hardware-Anforderungen des „K-Infinity“-Systems befindet sich im Anhang dieser Arbeit.

Zusätzlich zu der Weiterentwicklung des Systems in den Fachbereichen muss eine zentrale Instanz geschaffen werden, welche die Entwicklung und die spätere Anwendung des Portals überwacht. Gerade im Hinblick auf dezentrale Pflege kann ein derartiges System ohne zentrale Kontrolle seinen eigentlichen Sinn nicht erfüllen. Um daher nicht den Weg in Richtung „Datenfriedhof“ zu beschreiten, sollte das Davenport´sche Zitat, das dieses letzte Kapitel überschreibt, nicht vergessen werden.

## Literaturverzeichnis

- [Anschubprojekt]: o.V.: "Richtlinien - III die Beantragung von Anschubprojekten", keine Jahresangabe, Zentrum für Forschung und Entwicklung, Fachhochschule Darmstadt
- [Becker]: Rainer Becker: "EG-Richtlinie 90/270/EWG"; keine Jahresangabe, [http://www.dfpug.de/loseblattsammlung/loseblatt/auflage/lose2/09\\_soft/eg\\_richtlinien.htm](http://www.dfpug.de/loseblattsammlung/loseblatt/auflage/lose2/09_soft/eg_richtlinien.htm) (Mai 2003)
- [Benjamins et al.]: Richard Benjamins, Dieter Fensel, Asuncion Gomez Perez: "Knowledge Management through Ontologies", keine Jahresangabe
- [Bernard]: Frederic Bernad: "Interne Kommunikation und Mitarbeiterportal: Mit-tendrin statt nur dabei", Düsseldorf, keine Jahresangabe
- [Bielefeld]: Exposé des Workshops „Die Zukunft der wissenschaftlichen Informationsversorgung an den Hochschulen“, Bielefeld, Mai 2002
- [Bouaud et al.]: "Methodological Principles for Structuring an Ontology", Department Intelligence Artificielle et Médecine, August 1995, <http://citeseer.nj.nec.com/bouaud95methodological.html> (Stand April 2003)
- [Brachmann]: Ronald J. Brachmann, James G. Schmolze: An Overview of the KL-One Knowledge Representation System. In: Cognitive Science vol. 9 1985, pp. 171-216
- [Brockhaus]: Brockhaus Die Enzyklopädie: in 24 Bänden. 20., neu bearbeitete Auflage. Leipzig, Mannheim: F.A. Brockhaus 1996-99.
- [Corcho et al.]: Oscar Corcho, Mariano Fernández-López, Asunción Gómez Pérez: "Technological Roadmap V.1.1", Universidad Politécnica de Madrid, <http://ontoweb.aifb.uni-karlsruhe.de/About/Deliverables/Deliverable111.pdf>
- [Daml] o.V.: "The Darpa Agent Markup Language Homepage", keine Jahresangabe, <http://ww.daml.org> (Stand Mai 2003)
- [Davenport]: Thomas H. Davenport: "Some Principles of Knowledge Management", <http://www.bus.utexas.edu/kman/kmprin.htm> (Stand: März 2003)
- [Davenport et al.] Thomas H. Davenport; Lawrence Prusak: Working Knowledge : "How Organizations Manage What They Know", 1997.

- [Davies et al.] John Davies;Dieter Fensel ; Frank Van Harmelen : "Towards the Semantic Web – Ontology Driven Knowledge Management“, Wiley Verlag Chichester, England, 2003
- [DesignManual]: o.V.: "Corporate Design Manual", keine Jahresangabe, Fachhochschule Darmstadt
- [Dublincore]: Dublin Core Meta Data Element Set:  
<http://dublincore.org/documents/dces/> (Stand März 2003)
- [Duden]: "Duden: Das große Wörterbuch der deutschen Sprache in 10 Bänden", 3., völlig neu bearbeitete und erweiterte Auflage. Mannheim, Leipzig, Wien, Zürich: Dudenverlag 1999.
- [Elliot]: T.S. Elliot: "Choruses from the Rock", 1936
- [Englert]: Englert, S.: „Der Kampf um Ressource Vier“, 2001,  
[http://www.changex.de/pdf/d\\_a00378.pdf](http://www.changex.de/pdf/d_a00378.pdf) (Stand März 2003)
- [Eurowordnet]: N.N: "EuroWordNet", 1998, <http://www-ksl.stanford.edu/ontology/euowordnet.pdf> (Stand: März 2003)
- [Fensel] Dieter Fensel: "Ontologies – A Silver Bullet for Knowledge Management and e-commerce“, Springer Verlag Heidelberg, 2001
- [Großprojekt]: o.V.: "Richtlinien - III die Beantragung von Großprojekten", keine Jahresangabe, Zentrum für Forschung und Entwicklung, Fachhochschule Darmstadt
- [Gruber]: T. R. Gruber : A Translation Approach to Portable Ontology Specifications. in: Knowledge Acquisition. vol. 6, no. 2, 1993, Seite 199-221.
- [ISO9241]: o.V."Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten Teil 10: Grundsätze der Dialoggestaltung", Brüssel, 1995,  
<http://www.informatik.uni-stuttgart.de/ifi/ds/Lehre/Softerg/iso9241.pdf>
- [Jones et al.]: Dean Jones, Trevor Bench-Capon, Pepijn Visser: "Methodologies for Ontology Development", ohne Jahresangabe
- [Jurisica et al]: Igor Jurisica, John Mylopoulos, Eric Yu: "Using Ontologies for Knowledge Management: An Information Systems Perspective", 1999,  
[www.cs.toronto.edu/pub/eric/asis99.pdf](http://www.cs.toronto.edu/pub/eric/asis99.pdf) (Stand März 2003)
- [„K-Infinity“HB]: o.V. „K-Infinity“ - Knowledge Builder: Grundlagen und Anwendungen], Darmstadt, 2001
- [„K-Infinity“2001]: o.V., "White Paper „K-Infinity“", Darmstadt, 2001, [www.i-views.de](http://www.i-views.de) (Stand April 2003)
- [Kleinprojekt]: o.V.: "Richtlinien - III die Beantragung von Kleinprojekten", keine Jahresangabe, Zentrum für Forschung und Entwicklung, Fachhochschule Darmstadt



- [Knorz2000]: Gerhard Knorz: "Wissensmanagement im Betrieb", Beitrag zur Tagung "Die lernende Organisation - Ein Modell für die öffentliche Verwaltung", April 2000
- [Knorz2002] Gerhard Knorz: „Wissensmanagement? Wissensmanagement“, in FHDdirekt, Sonderbeilage Wissensmanagement, Darmstadt, 2002
- [KPMG1999]: o.V., "KPMG Knowledge Research Report", 2000
- [KPMG2001]: o.V., "Bedeutung und Entwicklung des multimediebasierten Wissensmanagements in der mittelständischen Wirtschaft“, Studie der KPMG im Auftrag des Bundesministerium für Wirtschaft und Technologie, 2001
- [Kreativland]: o.V.: "Teil 10 - Grundsätze der Dialoggestaltung" 2002, <http://www.kreativland.de/ergonomie/din/>
- [Landshut]: o.V.: "MVC-Tutorial", keine Jahresangabe, [http://www.fh-lands-hut.de/~gschied/projektess2001/mvc\\_tutorial/MVC1\\_\(model\).html](http://www.fh-lands-hut.de/~gschied/projektess2001/mvc_tutorial/MVC1_(model).html) (Stand April 2002)
- [Lindvall et al.] Mikael Lindvall, Ioana Rus, Sachin Suman Sinha: "Technology Support for Knowledge Management", University of Maryland, 2002
- [López et al.]: Mariano Fernández-López; Asunción Gómez-Pérez: "Methontology", Madrid, <http://www.ontoweb.org/workshop/ontoweb2/slides/methontosig3.pdf> (Stand März 2003)
- [Maedche]: Alexander Maedche: "Semantikbasiertes Wissensmanagement – Eine Anwendung im Human Resource Bereich", [www.ontoprise.de/](http://www.ontoprise.de/)
- [Managers]: o.V.: "Damit aus Strategien Handeln wird", keine Jahresangabe, [www.4managers.de/10-inhalte/Dateien/b2e.pdf](http://www.4managers.de/10-inhalte/Dateien/b2e.pdf) (Stand April 2003)
- [Mueller]: Joachim Müller: „Ontologien – eine Einführung“, Darmstadt, 2002; im digitalen Anhang
- [Nohr]: Holger Nohr: "Einführung in das Wissensmanagement", Hochschule der Medien, Studiengang Informationswirtschaft, September 2002
- [Ogden et al.]: Charles K Ogden, Ivor A. Richards: "The meaning of meaning", 1923, mehrfach aufgelegt, zuletzt 1999 bei Thoemmes Press, Ausschnitte auf <http://www.colorado.edu/communication/meta-discourses/Theory/richards/sld003.htm> und <http://www.missouri.edu/~engjnc/burke/ogden.html> (Stand März 2003)
- [Ollson]: Olle Ollson: "CommonKADS and the KADS-II Project", 1996, <http://www.sics.se/ktm/projects/kads.html> (Stand März 2003)

- [Oppermann et al.]: Henrik Oppermann, Rudi Studer, Hans-Peter Schnurr: "Die Bedeutung von Ontologien für das Wissensmanagement", Karlsruhe, keine Jahresangabe
- [Probst et al.] Gilbert Probst; Steffen Raub; Kai Romhardt: „Wissen managen. Wie Unternehmen ihre wertvollste Ressource nutzen“; Wiesbaden: Gabler, 3.Auflage, 1999
- [Quinn]: Andy Quinn, "The Java Tutorial", keine Jahresangabe, <http://java.sun.com/docs/books/tutorial/javabeans/> (Stand April 2003)
- [Roehl et al.]: Heiko Roehl, Kai Romhardt: "Wissensmanagement -Ein Dialog über Totes und Lebendiges", 2000, [www.mediaworx.com/owx\\_medien/heiko\\_roehl\\_dialog.pdf](http://www.mediaworx.com/owx_medien/heiko_roehl_dialog.pdf) (Stand März 2003)
- [Rohmhardt et al.]: Kai Romhardt; Gilbert Probst: „Bausteine des Wissensmanagements - ein praxisorientierter Ansatz“, <http://www.cck.uni-kl.de/wmk/papers/public/Bausteine/> (Stand März 2003)
- [Schneider]: Ulrich H. Schneider: "Nun schärt mal schön", Februar 2001, wissensmanagement online, [http://www.wm.4media-verlag.de/online/archiv/2001/02\\_0301/knowledge\\_sharing.htm](http://www.wm.4media-verlag.de/online/archiv/2001/02_0301/knowledge_sharing.htm) (Stand März 2003)
- [Schnurr et al.] Hans-Peter Schnurr, Steffen Staab, Rudi Studer, York Sure: "Ontologiebasiertes Wissensmanagement— Ein umfassender Ansatz zur Gestaltung des Knowledge Life Cycle", Karlsruhe, 2001
- [Schöpe]: Lothar Schöpe: „Erfahrungsbericht Portalentwicklung“, Universität Dortmund, 2001
- [Holzer]: Thomas Holzer, "Was ist Smalltalk?", keine Jahresangabe, <http://www.ges-training.de/Expertentips/holz1.html> (Stand April 2003)
- [Staab]: Steffen Staab, Hans-Peter Schnurr, Rudi Studer, York Sure: "Knowledge Processes and Ontologies", 2001
- [Straub]: Hans Rudolf Straub: "Das interpretierende System", Kapitel 10.1, Z/IM-Verlag, 2001, ISBN 3-9521232-6-9
- [Strutsdoc]: o.V. "The Struts User Guide", 2003, <http://jakarta.apache.org/struts/userGuide/> (Stand April 2003)
- [Studer et al.]: Rudi Studer, Michael Erdmann, Alexander Mädche, Henrik Oppermann, Hans-Peter Schnurr, Steffen Staab, York Sure, Christoph Tempich: "Arbeitsgerechte Bereitstellung von Wissen –Ontologien für das Wissensmanagement“, <http://www.aifb.uni-karlsruhe.de/WBS>

- [Sunmvc]: o.V.: "Designing Enterprise Applications with the J2EE-Plattform", 2. Auflage, Kapitel 4.4, Keine Jahresangabe
- [Sure et al.] York Sure, Rudi Studer: "On-To-Knowledge Methodology — Employed and Evaluated Version", Karlsruhe, 2001  
[www.ontoknowledge.org/download/del16.pdf](http://www.ontoknowledge.org/download/del16.pdf) (Stand März 2003)
- [Uschold et al.] Mike Uschold, Michael Gruninger: "Ontologies: Principles, Methods and Applications", 1996,  
<http://citeseer.nj.nec.com/rd/435280%2C40296%2C1%2C0.25%2CDownload/http://citeseer.nj.nec.com/cache/papers/cs/3214/http:zSzzSzwww.cm.cf.ac.ukzSzUserzSzJ-C.PazzagliazSzReferenceszSzart:Uschold-96.pdf/uschold96ontologie.pdf> (Stand März 2003)
- [Volz]: Raphael Volz: "Einführung in Ontologien", Beitrag zum Workshop "Begriffliche Formalisierung von Prozessen und Systemen", TU Dresden, 2001
- [Voss]: Greg Voss: "Introducing JavaBeans", 1996,  
<http://developer.java.sun.com/developer/onlineTraining/Beans/Beans1/> (Stand Mai 2003)
- [Wirth]: Thomas Wirth: "Die EN ISO 9241", keine Jahresangabe,  
<http://www.kommdesign.de/texte/din.htm> (Stand Mai 2003)
- [Zeeb]: Uli Zeeb: „Kommunikation als Innovationsmotor“, Seminararbeit an der Universität Mannheim, 2002
- [Zelewski et al.] Stephan Zelewski, Reinhard Schütte, Jukka Siedentopf: "Ontologien zur Strukturierung von Domänenwissen - Ein Annäherungsversuch aus betriebswirtschaftlicher Perspektive", Berlin, Juni 1999
- [Zürich]: o.V.: "Wissensmanagement", Institut für Informatik, Zürich,  
[www.ifi.unizh.ch/ikm/Vorlesungen/IM2/SS01/IM2\\_files/Vorlesung/knowledgemgmt\\_12.pdf](http://www.ifi.unizh.ch/ikm/Vorlesungen/IM2/SS01/IM2_files/Vorlesung/knowledgemgmt_12.pdf) (Stand März 2003)

# Abbildungs- und Tabellenverzeichnis

## Abbildungen

Nummer	Abbildungstitel	Seite
1	<i>Die Wissenstreppe</i>	9
2	<i>Der Wissenswürfel</i>	11
3	<i>Bausteine des ganzheitlichen Wissens- managements nach Probst</i>	12
4	<i>Wissensidentifikation</i>	15
5	<i>Die Todesspirale von Wissenssystemen</i>	20
6	<i>Das Semiotische Dreieck</i>	24
7	<i>Ontologiemodelle</i>	28
8	<i>Der Wissens – Metaprozess</i>	31
9	<i>Aufbau von Portalen</i>	39
10	<i>Komponenten der „K-Infinity“-Suite</i>	43
11	<i>K-Organizer</i>	45
12	<i>Graph-Editor</i>	45
13	<i>Dokumentenbegriffshierarchie der FHD Ontologie</i>	47
14	<i>Teiltaxonomie im Bereich Forschung</i>	50
15	<i>Tile Definition</i>	56
16	<i>Die Model-View-Control-Architektur</i>	58
17	<i>Ablauf einer Suchanfrage auf dem Por- tal der FHD</i>	60
18	<i>Die Integration von Intrexx</i>	74
19	<i>Expertensuche „wird veranstaltet von“</i>	79
20	<i>Suchergebnis der Expertensuche aus Abb.19</i>	80
21	<i>Ergebnis einer semantischen Suche</i>	82
22	<i>Der Net-Navigator</i>	83
23	<i>Integration eines Marketplace- Angebotes</i>	86
24	<i>Beispielhafte Modellierung eines Foren- systems</i>	93

25	<i>Rechtedefinition</i>	96
26	<i>Das Modell im Bereich „Zentrum für Forschung und Entwicklung“</i>	103
27	<i>Blick auf den Prototypen</i>	105
28	<i>Aktivierung und Konfiguration des Rech- tesystems</i>	108
29	<i>Portalsicht des Standardnutzers</i>	110
30	<i>Interessensrelationen des Individuums „Jochem, Ute“</i>	112
31	<i>Vollansicht des Wissensobjektes „Auf- gabenbeschreibung ASTA“</i>	113
32	<i>Anlegen einer Vereinigungssuche zum Thema „Gremien“</i>	115
33	<i>Inhalt des Menüpunktes „Neues aus den Gremien“</i>	116
34	<i>Objektansicht eines Marktplatzangebo- tes</i>	116
35	<i>Vorschläge der Themensuche</i>	118
36	<i>Tagesansicht des hochschulweiten Ka- lenders unter Intrex</i>	120
37	<i>Wissensobjekt „Großprojekt“</i>	123
38	<i>Anlegen eines neuen Forschungspro- jekts</i>	123
39	<i>Formular zum Upload eines Förderan- trags</i>	124
40	<i>Newsbox „Neu eingegangene Förderan- träge“</i>	126

## **Tabellen**

<b>Nummer</b>	<b>Tabellentitel</b>	<b>Seite</b>
1	<i>Formen des organisatorischen Vergessens</i>	21
2	<i>Ansätze zur Wissensbewertung</i>	22
3	<i>Der Wissens – Metaprozess</i>	33

## Anhang

### Fragebogen zur Erhebung des studentischen Bedarfs

- 1.) Bekommst Du die Informationen, die Du für dein Studium benötigst?
- 2.) Wie ist der Zugang zu Informationen zu beurteilen?
- 3.) Beurteile den Kontakt zwischen Studenten, Fachbereich und Verwaltung!
- 4.) Inwieweit weißt du über bürokratische Vorgänge bescheid?
- 5.) Findest Du Ansprechpartner bei Problemen? Wie?
- 6.) Wie ist die Unterstützung der studentischen Studienberater durch die zentrale Verwaltung? Kommunikation?
- 7.) Kommst du an Informationen über HiWi-jobs?
- 8.) Wie gut zugänglich sind allgemeine Studieninfos?
- 9.) Wie gut zugänglich sind spezifische (→ fachbereichsbezogene) Infos?
- 10.) Wirst Du über für dich interessante Veranstaltungen gut informiert?

## Ablauf einer Suchanfrage

Die Interaktion der MVC Komponenten lässt sich zunächst wie in Abbildung I gezeigt zusammenfassen.

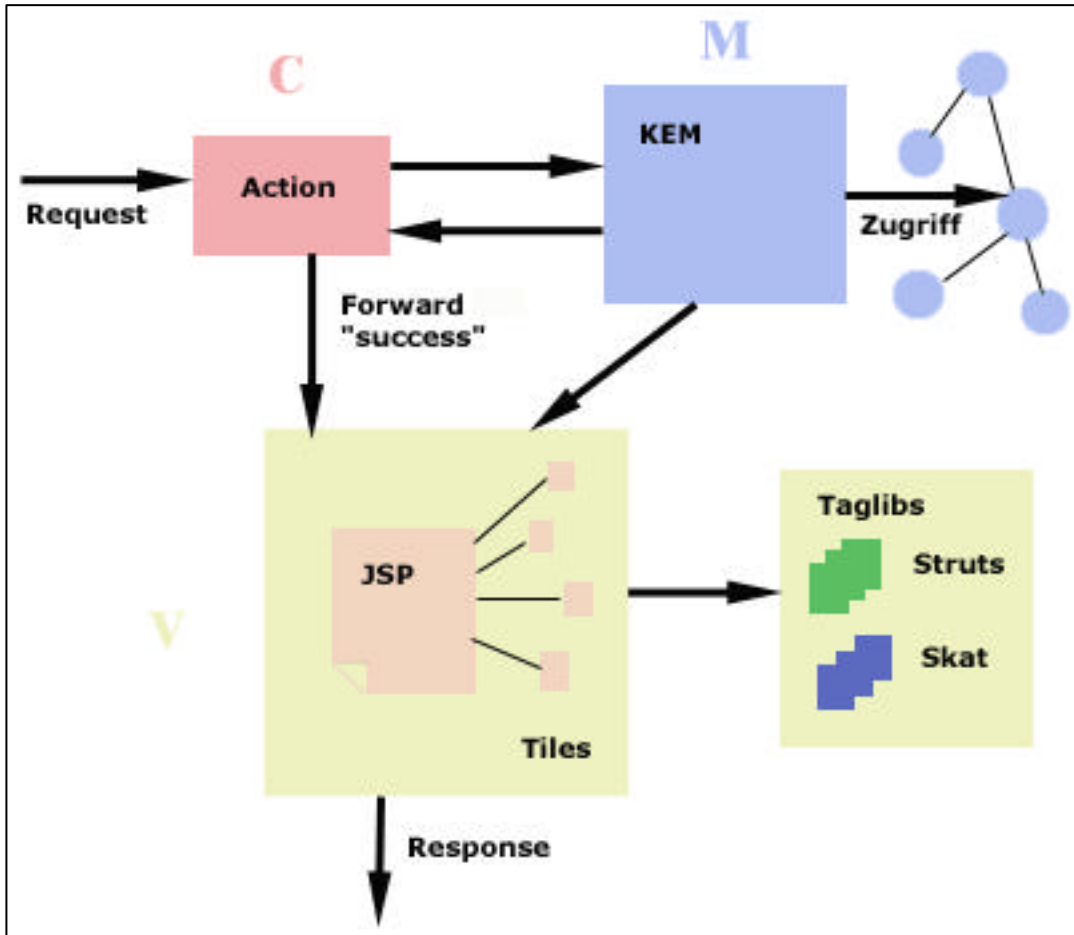


Abb. I: Ablauf einer Suchanfrage auf dem Portal der FHD

Die HTTP-Anfrage wird direkt an eine Action im „Controller“-Fragment der Anwendung weitergeleitet. Diese Action wird zuvor in einer JSP-Seite der „View“-Komponente definiert. Abbildung II zeigt ein Codefragment einer solchen JSP-Seite, welches die Darstellung einer Liste mit im Knowledge Builder vordefinierten Expertensuchen generiert.

```
<table width="100%">
  <bean:define id="searchLabels" name="sessionbean" property="expertSearches"/>
  <logic:iterate id="searchLabel" name="searchLabels" property="itemLabels"
indexId="index">
  <tr><td>
    <skat:action-url id="href" name="searchLabels" action="showExpertSearchResults">
      index=<%=index%>
    </skat:action-url>
    <tiles:insert definition="sl.link" flush="false">
      <tiles:put name="href" beanName="href"/>
      <tiles:put name="text">
        <bean:write name="searchLabel"/>
      </tiles:put>
      <tiles:put name="class" value="expertSearchList"/>
    </tiles:insert>
  </td></tr>
</logic:iterate>
</table>
```

Abb. II: Code Fragment für Aufruf von Expertensuchen (View)

Das Code-Fragment erklärt sich dabei wie folgt: In der ersten Zeile wird zunächst eine gewöhnliche HTML-Tabelle generiert. Hier wird bereits deutlich, dass es die JSP-Technologie ermöglicht, HTML und Java- bzw. XML-Fragmente problemlos in einer Datei zu vereinen. Die nächste Zeile definiert ein JavaBean. Die Syntax ist dabei folgendermaßen zu erklären: Das Bean erhält zunächst eine eindeutige id, diese ist, analog zu einem Variablennamen, das eindeutige Identifizierungsmerkmal des Beans. Das „name“- Attribut gibt das Ursprungs-Bean an, dessen Werte dem neu definierten Bean zugewiesen werden. Das „property“- Attribut definiert schließlich die Eigenschaft (=das Attribut) des Quellbeans, welches als Wertelieferant für das neu definierte Bean dient. Im Beispiel bedeutet dies: Es wird ein neues Bean definiert, welches die id (den Namen) „searchLabels“ und die Attributwerte „expertSearches“ des Beans „sessionbean“ erhält.

Das „<logic:iterate...>“ Tag in der Folgezeile ist funktional äquivalent zu einem Schleifenkonstrukt konventioneller Programmiersprachen. Dabei wird über das soeben definierte Bean „searchLabels“ iteriert. An dieser Stelle wird deutlich, dass die aus „sessionbean“ zugewiesene Property „expertSearches“ selbst ein Objekt mit eigenen Attributen ist. Innerhalb der Iteration wird über das Bean „searchLabel“, genauer gesagt über die Property „itemLabels“ des Beans „searchLabels“ iteriert. Die einzelnen „itemLabels“-Attributwerte, welche die



Bezeichner der vordefinierten Expertensuchanfragen enthalten, werden der id „searchLabel“ zugewiesen.

Nach öffnen einer Tabellenreihe und eines Tabellendatenfeldes wird erstmals die SKAT-Taglib genutzt: Das sogenannte „action-url“ Tag definiert hierbei einen Link und die Action, die aufgerufen wird, sobald der Link aktiviert wird. Es wird wiederum eine eindeutige id angegeben, ebenso wie das Inputbean erneut definiert wird. Das Action Attribut definiert die aufzurufende Action der „Controller“-Komponente. Zusätzlich wird ein Parameter (index) als Bezeichner-Werte-Paar übergeben.

Nach der inhaltlichen Definition des Links, muss dessen Visualisierung erfolgen. Hierbei stößt man erstmals auf das Tiles-Konzept, welches den Struts-Anwendungen zugrunde liegt. Im Beispiel wird ein Tile namens „sl.link“ aufgerufen. Im Körper (dem „Body“) des Tile-Inserts stehen die Parameter (in der Terminologie „Attribute“), die mit dem Aufruf des Tiles übergeben werden („href“, also der soeben definierte link, „text“, welchem die Iterationswerte von „searchLabel“ zugewiesen werden und „class“, welche die bereits definierte style-sheet-Klasse „expertSearchList“ zugewiesen bekommt). „Flush=false“ bedeutet, dass der Speicher nicht nach jedem Schleifendurchlauf geleert („geflusht“) wird. Dies hat zur Folge, dass die Ausgabe erst erfolgt, nachdem die Ausführung des Tags beendet ist. Flush muss immer dann auf „false“ gesetzt sein, wenn es sich bei dem Tag um ein sogenanntes „Body-Buffer-Tag“ handelt. Diese Art von Tags puffern die Werte, die in ihrem Tag-Body entstehen und schreiben diese erst nach Schließen des Tags. Wäre das Flush-Attribut hier auf „True“ gesetzt, wären die Werte der Iterationen verloren.

Um ein Verständnis des Tiles-Konzepts zu vermitteln, wird im folgenden aufgezeigt, wie die Anwendung mit einem solchen Tiles-Aufruf verfährt. Dazu muss zunächst die XML-Datei, welche die Tiles-Definition enthält, betrachtet werden. Ein Ausschnitt der Tiles-Definition ist in Abbildung III zu sehen.

```
<tiles-definitions>
    <definition name="sl.link" path="/link.jsp"/>
</tiles-definitions>
```

Abb. III: Ausschnitt der Tiles-Definition

Wie zu sehen, geschieht hier die Zuordnung (das „Mapping“) zwischen logischem Namen und dem Pfad, unter dem die JSP-Datei, welche die Funktionalität bereitstellt, zu finden ist

(im Beispiel die Datei link.jsp im Rootverzeichnis). Dieses JSP enthält den Code zur optischen Generierung des Links (Abbildung IV):

```
<a href="<tiles:getAsString name="href"/>"  
    class="<tiles:getAsString name="class" ignore="true"/>">  
    <tiles:getAsString name="text"/>  
</a>
```

Abb. IV: Codefragment link.jsp

Ein Link wird innerhalb des HTML-Anchor-Tags definiert. Das Ziel wird definiert durch den übergeben Parameter „href“ innerhalb des HTML-href-Attributs, der auf die Action „showSearchResults“ verweist (siehe oben). Das „class“-Attribut verweist auf ein CSS<sup>45</sup>, welches die Formatierung des Links spezifiziert. Als Linktext wird der Text definiert, der in Abbildung II dem Parameter „text“ zugewiesen wurde, also der Bezeichner der jeweiligen Expertensuche.

Mit Aktivierung des Links beginnt die Übergabe des Programmflusses an die „Controller“-Komponente der Anwendung. Ein Blick auf die sogenannten „Action Mapping“ - Definitionen der Struts-Konfigurationsdatei zeigt die in Abbildung V gezeigten Codezeilen.

```
<action name="skatForm" path="/showExpertSearchList"  
    scope="request" type="com.iviews.skat.action.GenericAction"  
    unknown="false" validate="false">  
    <forward name="success" path="sl.expertSearch.list"/>  
</action>
```

Abb. V: Codefragment der Struts-Konfigurationsdatei

Die erste drei Zeilen dieses Ausschnittes öffnen das Mapping-Tag. Unter dem Attribut „name“ ist hierbei nicht der Name der Action gespeichert, sondern vielmehr der Name des Javakonstrukts, welches die Action verwendet (des sogenannten actionForm Beans), also ein Bean namens „skatForm“. Das „path“-Attribut beschreibt den Pfadnamen der Action, mittels dem sie aufgerufen wird. Dieser Pfadname steht im Request und wird, wie oben beschrieben, in der action-url des aufrufenden Tags definiert. Das „scope“-Attribut das action-tags legt den Gültigkeitsbereich des erzeugten actionForm Beans fest. Hierbei gibt es die beiden Möglichkeiten „request“ (Gültigkeit innerhalb des Requests) oder „session“ (Gültigkeit innerhalb der Usersession). „type“ beinhaltet den vollständigen Java Klassennamen der imple-

---

<sup>45</sup> CSS: Cascading Stylesheet

mentierten Actionklasse, die das Mapping nutzt. Das Attribut „unknown“ mit boole'schem Datentyp wird dann „true“ gesetzt, wenn die entsprechende Action immer dann genutzt werden soll, wenn keine Andere den Request behandelt („Defaultmapping“). Demzufolge kann lediglich eine Action den „unknown“-Wert „true“ haben. Unter dem Attribut „validate“ wird spezifiziert ob die „validate“-Funktion der zugehörigen Action aufgerufen werden soll.

Im Body des action tags steht das „forward“ Tag. Dieses Forward-Tag, hier mit Namen „success“ beschreibt das Ziel der Action (= Viewkomponente), an welches, in diesem Fall bei erfolgreicher Abfrage, weitergegeben wird. Prinzipiell besteht die Möglichkeit, einer Action unterschiedliche forwards zuzuordnen, abhängig vom Ergebnis der Modellabfrage. Standard-Forward ist „success“.

Der Vorgang der Action, nämlich das Abfragen der Model-Komponente ist hier nicht ersichtlich, da es sich lediglich um ein Mapping handelt. Die eigentliche Action (der Teil, der die Abfrage realisiert) ist in Java programmiert und ergibt sich aus der Angabe des „type“-Attributs. Aus Übersichtsgründen wird allerdings hier eine Zäsur gesetzt und nicht weiter in die Materie eingedrungen. Nach erfolgter Abfrage des Modells über die KEM-Bridge (siehe oben) wird die Programmkontrolle wieder an die View-Komponente übergeben, die im „forward“-Tag des Actionmappings angegeben wurde (sl.expertSearch.list). Wie oben bereits angesprochen, handelt es sich bei der View-Komponente wieder um ein Tile, welches wiederum innerhalb der Tiles-Definition auf eine JSP-Seite verweist. Vorteil dieser Art des Aufrufes ist die Unabhängigkeit der Dateinamen von den Aufrufen. Ändert sich ein Dateiname, muss nur der Name in der Tile-Definition geändert werden und nicht in allen involvierten Dateien. Die dem Beispiel korrespondierende Tile-Definition wird nun nicht weiter erläutert, das Prinzip ist das in Abbildung III dargestellte. Das in der Tile-Definition angegebene JSP-Skript gibt schließlich eine Ausgabe, abhängig vom Ergebnis und der Eingabe des Nutzers an ein HTTP-Response Objekt. Die oben genannten Prinzipien werden hierfür erneut genutzt. Das Ergebnis des Suchvorgangs ist das Anzeigen einer im Knowledge Builder vorgefertigten Expertensuche. Abbildung VI zeigt eine Expertensuche, die alle Personen, die eine Dekan-Stelle besetzen als Ergebnis liefert (zum jetzigen Zeitpunkt ist allerdings nur der Dekan des Fachbereichs Informations- und Wissensmanagement in der zugrundeliegenden Ontologie).



Abb. V: Expertensuche nach Dekane

## Kompetenzfragen hinsichtlich der Yellow Pages

Die hier aufgelisteten Kompetenzfragen sind Beispiele und sollen bei der Überarbeitung sowohl des inhaltlichen Umfangs der Ontologie, als auch der Treffsicherheit der Yellow Pages-Funktionalität helfen.

- 1.) Wer ist an der FHD zuständig für die Beantragung von Fördergeldern bezüglich Forschungsprojekten?
- 2.) Wer ist für den Ablauf der unterschiedlichen Wahlen zuständig?
- 3.) Wer ist für das Sekretariat des Fachbereiches „Informations- und Wissensmanagement“ zuständig?
- 4.) Wer ist für die Pflege der Hochschulehomepage zuständig?
- 5.) Wer ist Leiter des Student Service Centers?
- 6.) Wer ist / sind die Ansprechpartner des Astas?
- 7.) Wer ist SAP-Beauftragter an der FHD?
- 8.) Wer ist für die Technik zuständig?
- 9.) Wer betreut das Intranet?
- 10.) Wer ist zuständig, bzw. kennt sich mit dem Bereich Wissensmanagement aus?
- 11.) Wer ist Ansprechpartner bezüglich Auslandsbeziehungen?
- 12.) Wer ist für die Öffentlichkeitsarbeit zuständig?

## Funktionsbeschreibung der Prototypdateien

Hier werden die Funktionen der wichtigsten Dateien beschrieben, um einen Einstieg in den zugehörigen Quellcode zu ermöglichen. Die zugehörigen Dateien finden sich komplett kommentiert im digitalen Anhang.

### a.) Hauptverzeichnis

Im Hauptverzeichnis wird der generelle Seitenaufbau generiert. Als Einstiegspunkt eignet sich hierbei defaultLayout.jsp

- \_template.jsp  
Vorlage zur Einbindung der verschiedenen Tag-Libraries
  
- bild.jsp  
Erzeugt dynamisch ein HTML-image-Tag und bekommt den Pfad als Parameter
  
- button.jsp  
Erzeugt dynamisch einen Button und bekommt die Parameter "href" (= Buttonziel), "class" (=css-Klasse), "text" (=Buttonbeschriftung) als String.
  
- defaultHTMLHeader.jsp  
erzeugt den HTML-Head des Portals
  
- defaultLayout.jsp  
Zentrale Datei, die das Layout der Seite dynamisch bei jedem Aufruf zusammenstellt. Dabei gibt es folgende Seitenkomponenten: Kopf, left, center und right, die abhängig vom Aufruf zusammengefügt werden.
  
- defaultLeft.jsp  
Aufbau des gesamten linken Bereichs der Seite, also Menü, Such- und Navigierenfeld und Kalender. Vor der Anzeige wird überprüft, ob der Nutzer eingeloggt ist oder nicht und abhängig davon die Ausgabe erzeugt.
  
- expParams.jsp  
Wird bei Anwahl eines Menüpunktes aufgerufen. Falls der Menüpunkt eine Parameterabfrage verlangt, wird ein Dialog zur Eingabe ausgegeben, ansonsten direkt die Informationen aus der Ontologie angezeigt. Handelt es sich bei dem aufgerufenen

Menüpunkt um das Telefonverzeichnis oder die Yellow Pages, wird eine spezielle Ausgabe erzeugt. Wenn nicht, wird falls vorhanden, der Parameter durch eine Standardaufforderung abgefragt. Falls kein Parameter vorhanden, wird direkt die Ergebnisliste der hinter dem Menüpunkt liegenden Expertensuche angezeigt.

- fh\_link.jsp  
Erzeugung eines speziellen Links, der den "More"-Pfeil als Linktext hat. Ansonsten wie in link.jsp: Dynamisches Zusammenbauen des Links anhand übergebener Attribute.
- frameset.html  
Hier wird das Frameset auf der Obersten Ebene aufgebaut. Diese Datei wird beim Portaleinstieg aufgerufen. Das Frameset gliedert sich dabei in 3 Frames: Einen Hauptframe, einen einpixelbreiten Frame, der den Netnavigator im Hintergrund lädt und einen einpixelbreiten Frame, der auf Intrexx zugreift.
- home.jsp  
Der Seitenmittelteil, der bei Erstaufwurf des Portals aufgerufen wird. Bindet das Tile „fh.push“ mit ein.
- labeledHorizontalList.jsp  
Wird bei zur Darstellung von Listen aufgerufen (Darstellung von Attributen, Relationen, Erweiterungen etc.).
- link.jsp  
Dynamische Linkerzeugung. Belegen der Attribute des <a>-Tags durch übergebene Attributswerte („href“, „class“, „text“, „target“) aus den aufrufenden Tiles.
- login.jsp  
Beinhaltet das Login-Formular des Portals
- menu.jsp  
Ausgabe der Menüpunkte.
- push.jsp  
Auslesen der Pushdienste. Dabei wird zwischen eingeloggten und nicht eingeloggten Nutzern unterschieden.

- startNN.jsp  
Aufruf des Net-Navigators
- tableFrame.jsp  
Wird immer dann aufgerufen, wenn bestimmter Inhalt mit einem Rahmen ausgegeben werden soll.
- telefon.jsp  
Spezielle Ausgabeseite für das Telefonverzeichnis
- yellowP.jsp  
Spezielle Ausgabeseite für die Yellow Pages

**b.) Verzeichnis „viewTopic“**

Im Verzeichnis „viewTopic“ liegen die Dateien, die zum Anzeigen von Wissensobjekten („Topics“) benötigt werden. Zentrale Ausgangsdatei ist dabei viewTopic.jsp

- attList.jsp  
Zentrale Datei für die Ausgabe von Attributen.
- attValue.jsp  
Abhängig vom Datentyp des anzuzeigenden Attributes wird die Ausgabe erzeugt (Bilder werden direkt angezeigt, URLs als Links, etc.)
- body.jsp  
Zentrale Anordnung von Attributen, Relationen und Erweiterungen
- docShort.jsp  
Erstellen von Kurzformen von Dokumenten, die größer 200 Zeichen sind. Diese Datei wird verwendet, wenn Konzepte angezeigt werden sollen, die eine Themenrelation zu einem Dokument haben.
- extension.jsp  
Gestaltung der Erweiterungen der Ergebnis-Topics
- head.jsp  
Gestaltung des Kopfes von Wissensobjekten

- matchings.jsp  
Gestaltung der Themensuche
- relList.jsp  
Zentrale Datei für die Ausgabe von Relationen. Pendant zu attList.jsp
- viewTopic.jsp  
Zentrale Layoutgestaltung von Wissensobjekten (Topics)

**c.) Verzeichnis „WEB-INF“**

- Dateien mit Endung .tld  
Beinhalten die Definition der Tag-Libraries
- struts-config.xml  
Enthält das sogenannte Mapping der Actions und die Konfiguration der Anwendungskomponenten mediator und bridge
- tiles-def.xml  
Das Zuordnen der logischen Codebezeichnungen (Tiles) zu den physischen Dateien.
- web.xml  
Zentrale Konfigurationsdatei der Webanwendung

**d.) Verzeichnis „results“**

Im Verzeichnis „results“ liegen die Dateien, die für die Ausgabe von Suchanfragen sind, die über die Textsuche an das Datenmodell gestellt wurden. Einstiegsdatei ist hierbei „textSearchResults.jsp“.

- folder.jsp  
Ausgabe der Suchergebnisse
- head.jsp  
Aufbau des Kopfes der Suchergebnisanzeige
- textSearchResults.jsp  
zentrale Datei für den Aufbau der Ausgabe von Suchergebnissen



- topLevelFolder.jsp  
Formatierung der Ausgabe der Ergebnisse

**e.) Verzeichnis „fp“**

Im Verzeichnis „fp“ liegen die Dateien zur Realisierung des Use-Cases. Einstiegspunkt ist in diesem Falle die Datei view.jsp

- body.jsp  
Aufbau des Körpers (Relationen und Attribute)
- chooseTarget.jsp  
Auswahl von Relationszielen via Net-Navigator und Suchfeld bei der Antragsstellung.
- edit.jsp  
Einstiegspunkt des Use-Cases. In dieser Datei wird eine spezielle Ansicht eines Wissensobjekts erstellt. Dabei kann es sich entweder um ein bereits existierendes oder um eines, was mithilfe dieses Formulars erzeugt wird handeln. Derzeit (im Zuge der Prototypisierung und des Use-Cases) handelt es sich dabei um Forschungsprojekte. Es können Attribute und Relationen neu gesetzt oder editiert werden und Anträge bzw. Dokumentationsdokumente hochgeladen werden.
- editAntrag.jsp  
Hochladen und editieren von Anträgen.
- editAtts.jsp  
Formatierung der Attributsausgabe im Editiermodus
- editRels.jsp  
Formatierung der Relationenausgabe im Editiermodus
- editValue.jsp  
Erzeugung von Eingabefeldern für Attributswerte abhängig vom Datentyp des Attributes (ein Boole'sches Attribut benötigt bspw. eine andere Darstellungsform als ein Textattribut).
- head.jsp  
Erzeugung des Kopfes der zu editierenden Wissensobjekte

- view.jsp

Darstellung der Wissensobjekte im Editiermodus, analog zu viewTopic.jsp im Unterverzeichnis viewTopic.

## Hinweise bezüglich des Lesens der Prototypdateien

Um ein Verständnis des Quellcodes zu garantieren, empfiehlt es sich, den im Anhang vorhandenen exemplarischen Ablauf einer Suchanfrage zu lesen, ebenso wie den vorangehenden Abschnitt über die Funktionsbeschreibungen der wichtigsten Dateien. Beim Lesen des Quellcodes gibt es ferner einige generelle Aspekte zu beachten:

- 1.) Struts arbeitet tile-basiert. Das bedeutet, dass einzelnen Dateien nur über Tilesaufrufe angesprochen werden. Das Zuweisen der Tiles als logische Abschnitte zu den physischen Dateien geschieht dabei über die Datei „tiles-def.xml“. Ruft eine Datei ein Tile auf, so können Attributswerte für den Aufruf mitgegeben werden<sup>46</sup>. Diese Attribute müssen zu Beginn des Tiles zunächst importiert werden, um im Tiles-Kontext verfügbar zu sein.
- 2.) Am Anfang jeder Datei werden die notwendigen Tag-Libraries per Namespace eingebunden. Um diese Tag-Libraries zu verwenden, wird zusätzlich ein Präfix vergeben, über dieses auf die entsprechende Tag-Library zugegriffen werden kann. Beispiel:  
  
`<%@ taglib uri="/WEB-INF/struts-logic.tld" prefix="logic" %>` verweist auf die logic-library von struts und kann über `<logic:.....>` angesprochen werden.
- 3.) Vorgänge (Zugriffe, Formularbearbeitung etc.) innerhalb des Portals werden über „actions“ realisiert. Die „actions“ sind in Java geschrieben und werden in der Datei „struts-config.xml“ zugewiesen. Ein solches, sog. „action-mapping“ definiert die Art der action und unterschiedliche Weiterleitungen an Dateien, abhängig vom Ergebnis ihrer Ausführung.
- 4.) Sämtliche Links werden im Portal dynamisch erzeugt. Dazu gibt es das „action-url“-Tag aus der skat-Tag-library, das abhängig einer zugehörigen Action das Linkziel dynamisch erzeugt. Zusätzlich dazu muss die Layoutgestaltung des Links mithilfe von HTML erfolgen.

---

<sup>46</sup> Als Attribute werden diese Werte deshalb bezeichnet, da alle Beans und Parameter Attribute des Request-Objektes sind.

5.) Mithilfe von Struts kann man Standardausgaben abhängig von der Sprache des zugreifenden Browsers ausgeben lassen. Dazu gibt es das `<bean:message.../>`-Tag, welches eine Datei namens `msg.properties` im Unterverzeichnis `/WEB-INF/classes/com/iviews/sl` ausliest. Prinzipiell ist es so möglich vollkommen automatisch an die Sprache des Nutzers angepasste Ausgaben zu produzieren. Hier wurde nur exemplarisch der Zugriff auf deutsche Sprachelemente realisiert.

### **Hardware-Empfehlung für die Anschaffung eines Portalservers**

Bezüglich eines internen Portalhostings muss ein Server mit den folgenden Minimalkomponenten angeschafft werden<sup>47</sup>. Dabei lassen sich die einzelnen Anforderungen nicht immer genau spezifizieren; dies ist mit der mangelnden Erfahrung des Einsatzes eines solchen Systems, wie des FHD-Portals, zu begründen:

#### **CPU:**

Prinzipiell ist ein „Single-Processor“ System auf dem aktuellen technischen Stand ausreichend. Bezüglich zunehmender Komplexität des Rechtessystems und zunehmend komplexerer Suchanfragen sollte jedoch ein Multiprozessorensystem in Betracht gezogen werden. Erfahrungswerte existieren dabei, wie bereits erwähnt, nicht.

#### **Arbeitsspeicher:**

Die einzelnen Komponenten der „K-Infinity“-Software-Suite benötigen die jeweils angegebene Größe an Arbeitsspeicher. Bei der Bridge sei erwähnt, dass der Arbeitsspeicherbedarf abhängig der Netzgröße variiert. Daher ist die folgende Angabe als Richtlinie zu sehen:

Web-Frontend (Java-Server): 128 Megabyte

Bridge: 128 Megabyte

Mediator: 64 Megabyte

Um Kapazitäten nach oben hin zu garantieren, sollten 512 Megabyte Arbeitsspeicher in dem System enthalten sein. Festplattenspeicher spielt bei der Wahl der Hardware zunächst keine Rolle, da gängige Festplatten den Ansprüchen vollends genügen. Als Betriebssystem eignet sich prinzipiell jede erhältliche Linux-Distribution. Je schlanker das Paket dabei ist, desto geeigneter ist es für eine derartige Serverlösung. Als Empfehlung seien hier „Debian“ „Red Hat“ oder „SuSe“ genannt.

---

<sup>47</sup> Die Angaben ergaben sich dabei aus Rücksprachen mit der Firma „intelligent views“.

## **Digitaler Anhang**

Auf der beigefügten CD-ROM befinden sich folgende Verzeichnisse:

- a.) Diplomarbeit: Enthält diese Arbeit um PDF-Format
- b.) Prototyp: Enthält die Dateien des Prototypen komplett kommentiert
- c.) Einfuehrung\_Ontologien: Enthält den Artikel „Ontologien – eine Einführung“, der im Vorfeld dieser Arbeit entstanden ist.