

TECHNISCHE UNIVERSITÄT DARMSTADT



INSTITUT FÜR NUMERISCHE METHODEN UND INFORMATIK IM BAUWESEN

DIPLOMARBEIT

**„Analyse von Wissensnetzen unter Berücksichtigung der Anforderungen einer
internetgerechten Bauprojektorganisation“**

VORGELEGT VON:

STEFFEN MOLDANER

MATR. NR. 722058

DARMSTADT, IM OKTOBER 2001

1	EINLEITUNG	4
1.1	Motivation	4
1.2	Zielsetzung	5
2	GRUNDLAGEN DER WISSENSVERARBEITUNG	7
2.1	Was ist Wissen?	7
2.2	Techniken zur Verarbeitung von Wissen.....	7
2.3	Symbolische Verarbeitung	8
2.3.1	Wissensrepräsentation und -verarbeitung	8
2.4	Subsymbolische Verarbeitung mittels neuronaler Netze	16
2.4.1	Wissensrepräsentation und -verarbeitung	17
2.5	Gegenüberstellung symbolische- und subsymbolische Wissensverarbeitung	22
3	SOFTWARETECHNISCHE UMSETZUNGEN	23
3.1	Intelligent views mit K-Infinity	23
3.1.1	Die Technologie	23
3.2	USU AG mit dem KnowledgeMiner	26
3.2.1	Die Technologie	27
3.3	SER Systems AG mit SERware	28
3.3.1	Die Technologie	29
3.4	Autonomy	31
3.4.1	Technologie	32
3.5	Weitere Unternehmen	33
3.6	Gegenüberstellung der verschiedenen Technologien.....	34
4	KONZEPTION EINES GRUNDNETZES FÜR BAUVORHABEN.....	36
4.1	Definition.....	36
4.2	Grundprinzipien bei der Modellierung eines semantischen Netzes	37
4.2.1	Die Ober-/Unterbegriffsrelation (OUR).....	37
4.2.2	Anzahl der Unterbegriffe	38
4.2.3	Hierarchietiefe	38
4.2.4	Gruppierung von Begriffen mit gleichem Merkmal	38
4.2.5	Gleicher Abstraktionsgrad aller Unterbegriffe.....	39
4.2.6	Ein Wissensnetz ist kein Baum	40

4.2.7	Relationen.....	41
4.2.8	Anwendung von Erweiterungen (Rollen)	41
4.2.9	Auswahl der Individuenfähigen Begriffe	42
4.2.10	Erkennen abhängiger Begriffe	42
4.3	Anforderungen an das Baunetz	43
4.4	Grundkonzept.....	43
4.5	Das allgemeingültige Grundnetz.....	44
4.6	Das Baugrundnetz	46
4.6.1	Bauprojekt	47
4.6.2	Bauvorgang	48
4.6.3	Baunormen	51
4.6.4	Projektbeteiligter	56
4.6.5	Bauobjekt	58
4.6.6	Dokumente	60
4.6.7	Version	62
4.6.8	Änderung.....	65
4.7	Modellierungsbeispiel	67
5	MODELLIERUNG DES GRUNDNETZES MIT K-INFINITY	68
6	ANWENDUNGSBEISPIEL	82
6.1	Anforderungen	82
6.2	Verwendete Werkzeuge	82
6.2.1	K-Infinity.....	82
6.2.2	Usage Tools Server	82
6.2.3	Web-Server.....	82
6.2.4	JAVA Servlet Technologie	83
6.2.5	XML-Parser.....	83
6.2.6	XSL und XSLT Stylsheet Processor	83
6.2.7	Das com.oreilly.servlet Paket.....	83
6.2.8	Grundkonzept	84
6.3	Umsetzung des Internetauftritts mit JAVA-Servlets	85
6.3.1	Das Paket <i>com.iviews.bn</i>	85
6.3.2	XSL	92
6.4	Konzeption des Internetauftritts.....	92
6.4.1	Aufbau des Internetportals	92
6.4.2	Funktionalität	95
7	ZUSAMMENFASSUNG UND AUSBLICK.....	109
8	ANHANG	111

8.1	Installation	111
8.1.1	Softwarevoraussetzungen zur Installation	111
8.1.2	Installation Jakarta-Tomcat 3.2.3	111
8.1.3	Installation Baunetz Servlet	111
8.1.4	Installation Wissensnetz	112
8.1.5	Starten der Anwendung	112
8.2	Abbildungsverzeichnis	113
8.3	Literaturverzeichnis	116
9	ERKLÄRUNG ZUR DIPLOMARBEIT GEMÄß § 19 ABS. 6 DPO/AT	117

1 Einleitung

1.1 Motivation

Bei der im Bauwesen vorherrschenden Unikatbauweise unterscheiden sich die einzelnen Bauprojekte in der Planung, in der Ausführung und der Zusammensetzung der Projektbeteiligten. Dies erfordert eine sorgfältige Planung und Abstimmung der einzelnen Prozesse sowie die Kooperation der einzelnen Beteiligten untereinander. Um geforderte Termine und Kosten, auch bei außerplanmäßigen Änderungen, fristgerecht einhalten zu können, ist eine effiziente Information der von den Änderungen, direkt oder indirekt, betroffenen Projektbeteiligten nötig. Dies setzt eine konsequente Dokumentation aller Änderungen, die von der ursprünglichen Planung abweichen, und eine gleichzeitige sofortige Publikation unter allen Projektbeteiligten voraus, damit die Planung der einzelnen Projektbeteiligten gegebenenfalls angepasst werden kann.

Die elektronisch gestützte Datenverarbeitung hat in den letzten Jahren immer stärker Einzug in die Bauunternehmen erhalten. So existieren verschiedene Programme zur Terminverwaltung, Ablaufplanung und Dokumentenverwaltung. Die Software wird jedoch nur firmenintern genutzt, da sie zum einen sehr speziell für den eigenen Unternehmensbereich und die somit ausgeführten Bauphasen ist. Zum anderen ist das Datenaustauschformat der einzelnen Programme teilweise inkompatibel zu Konkurrenzprodukten.

Um den sofortigen Austausch von Dokumenten und Informationen zu gewährleisten, ist ein zentraler Softwareansatz nötig. Das heißt, das Softwaresystem sollte von allen Projektbeteiligten sofort und von überall erreichbar sein, um sofortigen Zugriff auf die jeweils aktuellen Dokumente bzw. Informationen zu haben, oder Änderungen allen Betroffenen sofort zugänglich zu machen. Hierfür bietet sich eine internetbasierte Plattform an, da Informationen zentral an einer Stelle verwaltet werden und der Zugriff von überall, eine bestehende Internetverbindung vorausgesetzt, gewährleistet ist.

Das Forschungsprojekt Baukom-Online verfolgt das Ziel, durch den Einsatz von modernen Informationstechnologien bauorganisatorische Verbesserungen zu verwirklichen und damit die Prozesse des Planens, des Entwerfens, des Bauens und des Betriebs von Bauobjekten zu automatisieren, zu rationalisieren und die Kosten- bzw. Terminalsicherheit zu optimieren. Baukom-Online ist eine internetbasierte Plattform, die es ermöglichen soll, komplette Bauprojekte hierüber abwickeln zu können. So soll zu jeder Zeit ein einfacher und strukturierter Zugriff auf aktuelle Informationen und Dokumente gewährleistet sein. Ebenso soll das Einfügen neuer Dokumente und Informationen über dieses System erfolgen, um sofort allen anderen Projektbeteiligten zur Verfügung zu stehen.

Für die Verwaltung und Speicherung der anfallenden Datenmengen und Informationen muss ein flexibles und leicht auf die unterschiedlichen Bauprojekte anpassbares System gefunden werden. Dieses sollte dem Anwender die Benutzung so einfach wie möglich machen, d.h. die

zu erwartenden Eingaben eines Anwenders sollten so gering wie möglich und die zu gewinnenden, relevanten Information, so groß wie möglich sein.

Bei der Änderung eines Bauteils kann es nicht sofort ersichtlich sein, dass auch andere Bauteile davon betroffen sein können und dies ganze Terminverschiebungen und somit höhere Kosten verursachen kann. Hierfür sollte das System die semantischen Abhängigkeiten der Prozesse bzw. Dokumente untereinander kennen und somit mögliche Auswirkungen aufzeigen können.

Ebenso sollte es dem Anwender möglich sein, sich ohne große Einarbeitungszeit im System zurechtzufinden. Hierfür müssen geeignet Suchmöglichkeiten, die keine bestimmte Terminologie voraussetzen, zur Verfügung stehen.

Denkbar wäre der Einsatz eines klassischen Dokumentenmanagementsystems. Dokumentenmanagementsysteme verlangen bei der Eingabe neuer Dokumente eine Klassifizierung nach einem bestimmten Muster, d.h. der Anwender muss ein Dokument bestimmten Kategorien zuordnen oder zutreffende Schlüsselwörter für das Dokument vergeben. Dies verlangt vom Anwender ein hohes Maß an Wissen über das System. Wissensmanagementsysteme hingegen entlasten den Anwender schon bei der Eingabe. So wird teilweise eine automatische Klassifizierung neuer Dokumente geboten. Hierbei wird versucht, anhand bedeutungsvoller Wörter eines Dokumentes, das Dokument automatisch nach bestimmten Themenbereichen im System zu klassifizieren. Aufgrund dieser internen Klassifizierung erleichtern Wissensmanagementsysteme auch die Suche nach konkreten Informationen, d.h. auch hier wird der Anwender entlastet.

Ein zentrales Projektverwaltungssystem kann sich auf breiter Basis nur dann durchsetzen, wenn der Anwender bei minimaler Einarbeitungszeit einen gleichzeitig gestiegenen Nutzen hat. Daher bietet sich der Einsatz eines Wissensmanagementsystems zur Bauprojekt-Verwaltung an.

1.2 Zielsetzung

Ziel dieser Arbeit ist die Analyse von Wissensnetzen im Hinblick auf den Einsatz in einer gewerkeübergreifenden Internetplattform zur Koordination und Planung von Bauprojekten. Hierfür ist eine genaue Analyse der Grundlagen sowie deren Umsetzung in bestehenden kommerziellen Wissensmanagementsystemen nötig.

Ebenso soll ein Prototyp eines generischen Grundnetzes erstellt werden, der es ermöglicht, das Netz für beliebige Bauprojekte zu verwenden. Dies erfordert ein sehr offenes Grundnetz, in dem auch zur Laufzeit neue Objekte angelegt und Änderungen an bereits bestehenden Objekten vorgenommen werden können. Eine genaue Analyse der einzelnen Bauprojektbestandteile ist daher unerlässlich.

Um die Nutzung des erstellten Netzes zu demonstrieren, sollen entsprechende, internetbasierte Abfragewerkzeuge erstellt werden. Die Umsetzung erfolgt als eine zentrale,

internetbasierten Plattform, die sowohl geeignet Suchanfragemöglichkeiten als auch, nach Möglichkeit, automatisierte Werkzeuge zum Einpflegen neuer Informationen bietet.

2 Grundlagen der Wissensverarbeitung

2.1 Was ist Wissen?

Wissen ist (internationalisierte) Information plus der Fähigkeit, sie zu nutzen. [TIK]

Man kann zunächst verschiedene Arten des Wissens unterscheiden [EMAXPS]:

- Wissen über Objekte. Dies bezeichnet die Beschaffenheit eines Objektes. Das kann z.B. die Beschaffenheit eines Bauteils sein.
- Wissen über Beziehungen. Hierbei handelt es sich um die Beziehung zweier Objekte. Z.B. ein Bauwerk *besteht aus* Wänden.
- Wissen über Vorgehensweisen. Hier ist bekannt, in welcher Reihenfolge vorgegangen werden muss, um ein Problem zu lösen. Z.B muss zuerst das Fundament angelegt werden, um ein Gebäude zu erstellen.
- Wissen über Vorschriften. Zur Lösung eines Problems müssen verschiedene Bedingungen eingehalten werden. Z.B., dass eine Bewehrung eine Mindestabdeckung aufweisen muss.

2.2 Techniken zur Verarbeitung von Wissen

Informationen werden erst dann wirklich nutzbar, wenn sie organisiert, strukturiert, abrufbar und navigierbar sind. Es gibt verschiedene Arten, Informationen maschinell abzulegen und zu strukturieren. Konventionelle Programme haben ihr *Wissen* im Code, d.h. der Programmierer weiß, an welcher Stelle er welchen Algorithmus anwenden muss. Das Wissen, dass ein bestimmter Algorithmus an einer bestimmten Stelle aufgerufen werden muss, ist direkt im Code abgelegt. Dies erschwert es, das Wissen nachträglich abzuändern, denn hier kann der Algorithmus nur vom Programmierer selbst abgeändert werden.

Das Wissenschaftsgebiet der künstlichen Intelligenz (KI) befasst sich mit Problemstellungen, die meist nicht algorithmisch lösbar sind. Etwa Computer so zu programmieren, dass sie Aufgaben ausführen können, welche zum gegenwärtigen Zeitpunkt von Menschen wahrgenommen werden, da sie geistig höhere Prozesse erfordern. Dies sind z.B. Entscheidungen treffen oder Schlüsse ziehen.

Problemlösende Techniken lassen sich nach [HWPD] in vier Verarbeitungsverfahren unterteilen:

- Verfahren zur Informationsverarbeitung
- Verfahren zur numerischen Verarbeitung
- Verfahren zur symbolischen Verarbeitung
- Verfahren zur subsymbolischen Verarbeitung

Die beiden ersten Verfahren werden den klassischen Verfahren zugeordnet. Die symbolischen Verfahren verarbeiten Wissen mittels Symbolen, die auch Schwerpunkt dieser Arbeit sind. Die subsymbolische Techniken wollen ein intelligentes Verhalten erzwingen, wie es der Ansatz bei den neuronalen Netzen ist. Symbole werden hier üblicherweise nur implizit dargestellt (Verteilte Wissensrepräsentation).

2.3 Symbolische Verarbeitung

Der Vorteil symbolischer Wissensverarbeitung liegt im wesentlichen in der Architektur der Systeme begründet. Es findet eine klare Trennung zwischen Wissen und der Wissensverarbeitung statt (s. Abbildung 1). Eine Problemlösungsstrategie kann daher für mehrere Anwendungsfälle verwendet werden. Es muss nur jeweils die Wissensbasis ausgetauscht werden.

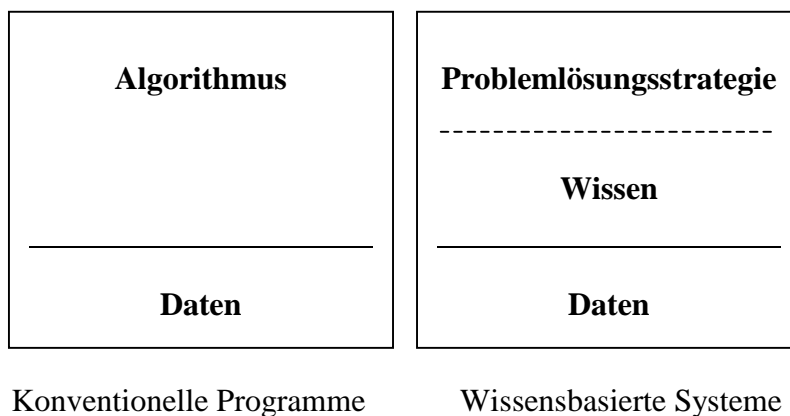


Abbildung 1: Unterschied konventionelle Programme zu wissensbasierten Systemen

2.3.1 Wissensrepräsentation und -verarbeitung

Wissensrepräsentation bezeichnet die Formalisierung von Wissen, um eine maschinelle Verarbeitung zu ermöglichen. Man unterscheidet zwei Arten der Wissensrepräsentation:

- Prozedurales Wissen
- Deklaratives Wissen

Eine Mischform aus beiden wird als hybride Wissensrepräsentation bezeichnet. Die Darstellung der Wissensrepräsentationsarten nach [HH]:

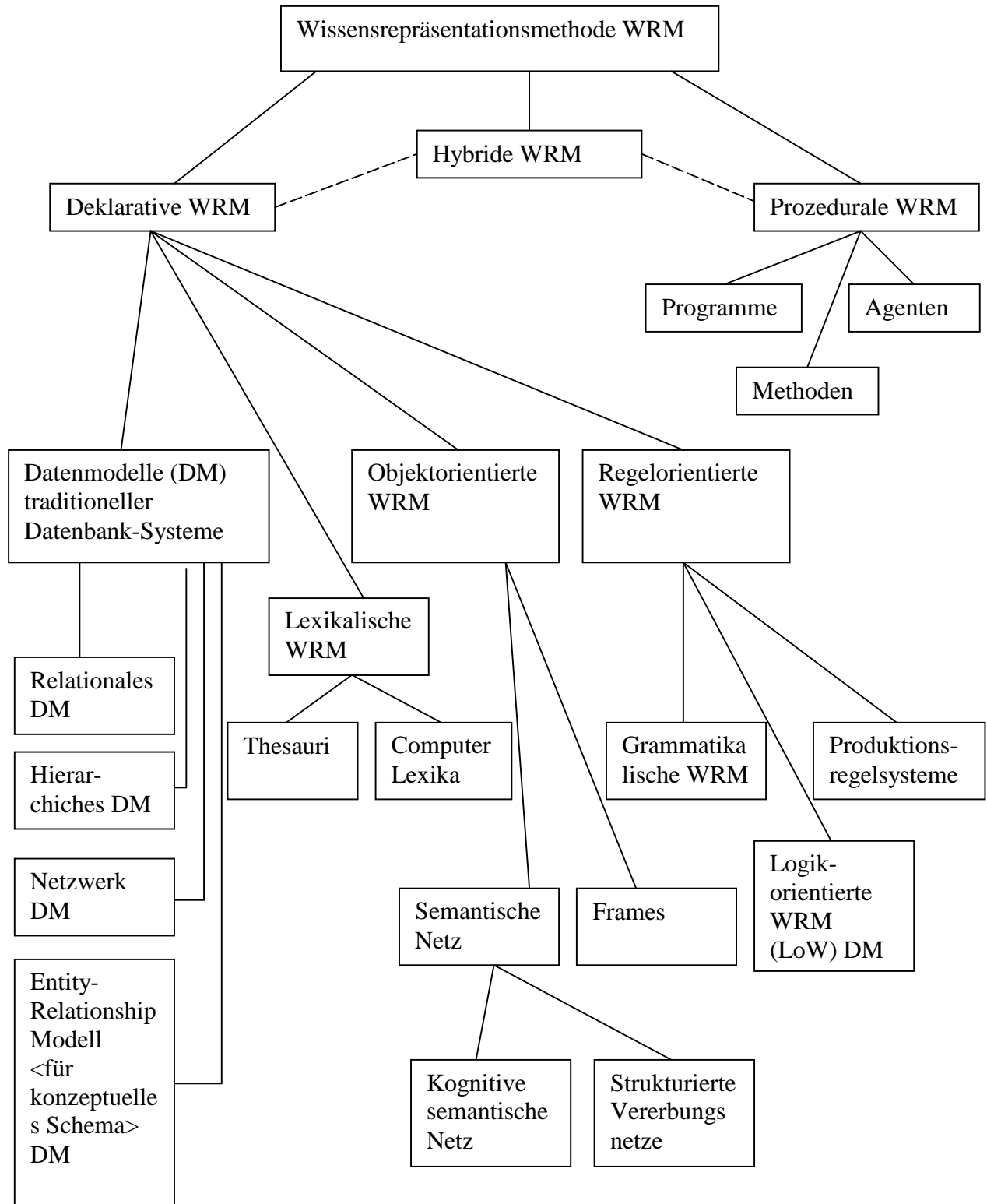


Abbildung 2: Darstellung der Wissensrepräsentationsarten nach [HH]

2.3.1.1 Prozedurale Wissensrepräsentation

Die prozedurale Darstellung beschreibt die Verfahren zur Konstruktion bzw. dem Gebrauch des Wissens. Diese Art der Darstellung ist oft einfacher, gerade bei weniger tiefgründigen Problemen, als die deklarative Methode. Sie lässt sich relativ einfach durch Programmiersprachen darstellen.

Beispiele für prozedurale Darstellungen sind:

- Eine **Produktionsregel** ist eine mit einer Vorbedingung versehene Aktion. Z.B. *Wenn Lagebestand ist leer dann Ware nachbestellen.*
- **Expertensysteme.** Als Expertensysteme werden Computerprogramme bezeichnet, die Wissensrepräsentations- bzw. Wissensverarbeitungsmechanismen kennen und dadurch in der Lage sind, in einem speziellen Wissensgebiet einen menschlichen Experten nachzubilden. Der Kern eines Expertensystems ist die Interferenzmaschine (Problemlösungs- oder Schlussfolgerungskomponente), die in der Lage ist, unterschiedliche Wissensaussagen zu einer Kette von Schlussfolgerungen nach folgendem Schema zu verbinden:
 WENN A gilt, **DANN** muss auch B gelten;
 WENN aber B gilt, **DANN** muss auch C wahr sein;
 Usw.

2.3.1.2 Deklarative Wissensrepräsentation

Die deklarative Darstellung ist die Beschreibung von Sachverhalten, die keine Angaben über Konstruktion und Gebrauch macht

Beispiele für deklarative Darstellungen sind:

- Semantische Netze
- Objekt-Attribut-Wert-Tripel
- Frames
- Topic Maps

Semantische Netze

Ein semantisches Netz stellt das Wissen grafisch dar. In der Literatur finden sich verschiedene Definitionen der Semantischen Netze.

Ein **Semantisches Netz** ist das mathematische Modell einer Begriffsstruktur, die aus einer Menge von Begriffen und den zwischen diesen bestehenden kognitiven Relationen aufgebaut ist. Es wird in Form eines verallgemeinerten Graphen dargestellt, wobei den Knoten des Graphen die Begriffsrepräsentanten und den Kanten Beziehungen zwischen den Begriffen entsprechen. [HH]

Ein **semantisches Netz** ist ein gerichteter Graph aus einer Menge von Knoten, die Objekte (Begriffe oder Konzepte) repräsentieren, sowie einer Menge von gerichteten Kanten (engl. arcs oder links), die Beziehungen (Relationen) zwischen den Objekten darstellen. Normalerweise werden sowohl die Knoten als auch die Kanten (Verbindungen) mit Namen versehen. [DVW]

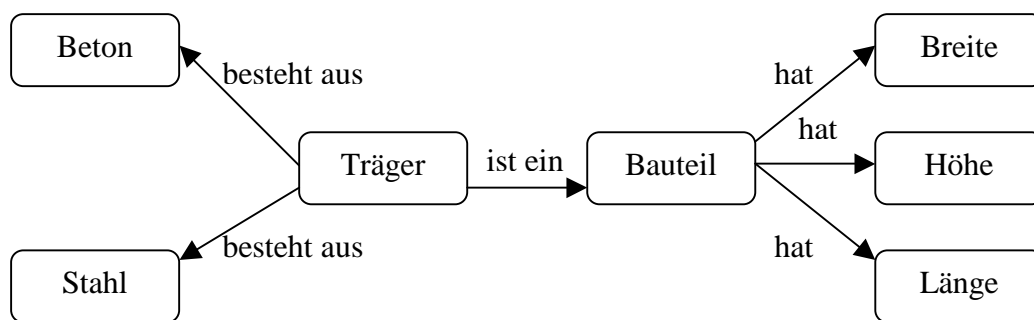


Abbildung 3: Beispiel semantisches Netz

- **Knoten** werden benutzt, um *Objekte* und *Deskriptoren* darzustellen.
- **Objekte** können physische Gegenstände, gedankliche Elemente wie Handlungen und Ereignisse, oder abstrakte Kategorien sein. Reimer [RU] unterscheidet Objekte in *Konzeptklassen*, *Individualkonzepte* und *Gruppenklassen*. *Konzeptklassen* nehmen mehrere zusammengehörige Objekte in sich auf. *Individualkonzepte* stellen nur ein Objekt der realen Welt dar. Die *Gruppenklassen* sind meist abstrakte Zusammenfassungen von *Konzeptklassen*.
- **Deskriptoren** sind zusätzliche Informationen (Attribute, *Eigenschaften*) über *Objekte*.
- **Kanten** stellen die Verbindungen (Relationen) der *Objekte* und *Deskriptoren* untereinander dar. Sie werden in Form eines Graphen dargestellt und beschriftet. Die Beschriftung gibt den Typ der Kante und somit die Beziehung der Objekte zueinander an. Häufige Verbindungen sind:
 - **Ist Instanz von** stellt die Relation zwischen *Konzeptklasse* und *Individualkonzept* dar. Z.B. ist ein Stahlprofil HEB500 *Instanz von* Stahlprofil.
 - **Hat** bezeichnet die Relation zwischen Objekten und Teilobjekte. Ein Bauteil *hat* eine Breite.

Kanten sind immer dann gerichtet, wenn die Beziehung zweier Objekte nicht symmetrisch ist, wie dies z.B. bei der *Ist Instanz von* Beziehung der Fall ist. Hier geht die Kante immer vom Individualkonzept zur Konzeptklasse. Die Richtung der Kante ist frei definierbar, ist aber für alle Kanten dieses Typs einzuhalten.

- **Vererbung** ist ein spezieller Fall von *Kanten*. Die Vererbung wird durch die Relation *ist ein* zu einem Oberobjekt dargestellt. Durch diese Relation erbt ein Objekt alle Charakteristika des Oberobjektes. Beispiel: Ein Träger *ist ein* Bauteil.

Objekt-Attribut-Wert-Tripel (O-A-W-Tripel)

Hierbei handelt es sich um einen Spezialfall der Darstellung semantischen Netze. *Objekte* können physische Entitäten oder Begriffe sein. *Attribute* sind die Eigenschaften eines *Objektes*. Der *Wert* ist der Wert eines *Attributes*. Ein O-A-W-Tripel entspricht einem elementaren semantischen Netz:

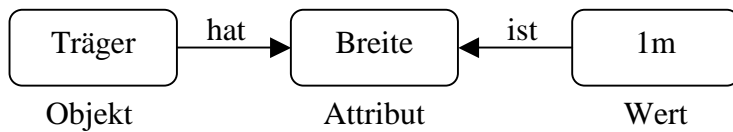


Abbildung 4: O-A-W-Tripel - semantisches Netz

Zu einem Objekt können beliebig viel O-A-W-Tripel existieren.

Objekt	Attribut	Wert
Träger	Breite	1m
Träger	Höhe	0,5m

Abbildung 5: Darstellung Objekt-Attribut-Wert-Tripel

Frames

Frames stellen eine objektorientierte Wissensrepräsentation dar. Objekte der realen Welt werden durch *Frames* dargestellt. Wie bei den semantischen Netzen haben sich auch hier zwei Unterscheidungsmerkmale für *Frames* herausgestellt, generische *Frames* (*Konzeptklassen*) und deren *Instanzen* (*Individualkonzepte*). Ein *Frame* besteht aus mehreren *Slots*, die manchmal auch Rollen genannt werden, die die Eigenschaften des *Frames* repräsentieren. In einer Instanz eines generischen *Frames* wird jedem Slot, entsprechend dem Attribut für das er steht, ein Wert zugeordnet.

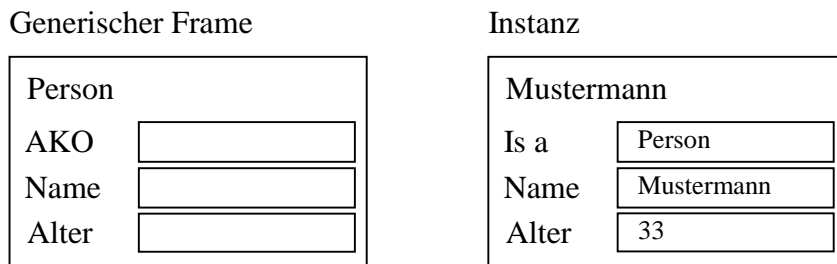


Abbildung 6: generischer Frame – Instanz

Die *Slots* lassen sich unterteilen in strukturelle Beziehungen und nicht-strukturelle Beziehungen. Strukturelle Beziehungen können sein:

- *Assoziative Slots* oder *Superclass Slots* enthalten den Namen einer Konzeptklasse oder deren Unterbegriffe (AKO steht für „a kind of“).
- *Relation Slots* oder *Member of Slots* repräsentieren die semantische Beziehung zwischen Instanz und Klasse (*Ist Instanz von, Is a*).
- *Rollen Slots*. Semantische Beziehungen selbst können als *Konzepte* aufgefasst und daher als Frame dargestellt werden. Sie werden dann Rolle genannt. So kann z. B. eine Instanz eines Konzeptes Person die Rolle Bauingenieur annehmen.

Die Nicht-Strukturellen-Beziehungen sind Slots, die nur einen bestimmten Wert für eine Eigenschaft eines Konzeptes haben

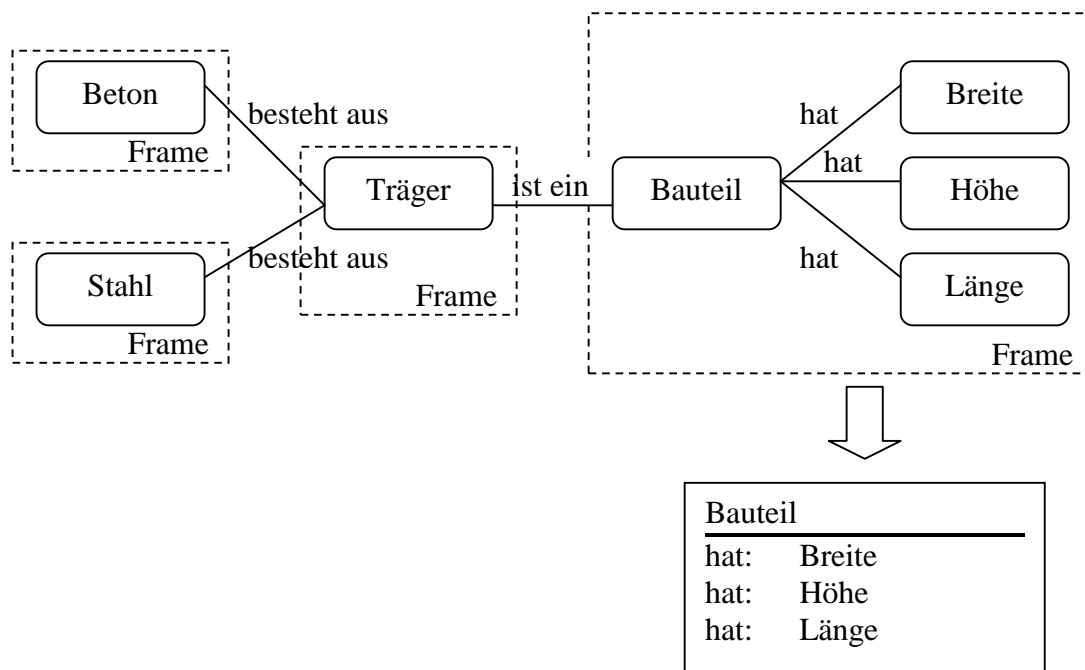


Abbildung 7: Frame - semantisches Netz

Topic Maps

Topic Maps sind SGML- oder XML-Dokumente, die Metadaten über Inhalt und Struktur vorhandener Informationsquellen liefern, indem sie Themen (*Topics*) definieren und diese über Links mit den relevanten Daten verknüpfen. Die Daten befinden sich dabei meist außerhalb der *Topic Map* und können in verschiedenen Datenformaten (ASCII, HTML, XML, PDF, Datenbanken, e-Mail, Tabellenkalkulationen usw.) vorliegen. [TM]. Es existiert ein ISO-Standard für *Topic Maps* (ISO 13250:1999 [ISOTM]), der im Dezember 1999 verabschiedet wurde. *Topic Maps* sind dem Konzept der *semantischen Netze* sehr ähnlich. Der Hauptunterschied liegt darin, dass sich die *Topic Maps* mehr auf die Navigation zwischen den

Topics über die *Associations* konzentrieren. Der Schwerpunkt bei den semantischen Netzen liegt hingegen in der Repräsentation des Wissens durch die verbundenen Knoten.

Die *Topic Map*-Konzepte sind:

- **Topic und Topic Type.** Ein Topic repräsentiert ein beliebiges Objekt, ein Ereignis oder eine Handlung. Jedes Topic ist eine Instanz von einem oder mehreren Topic Types. Die Verbindung zwischen Topic und Topic Type kann als eine *ist Instanz von* Relation interpretiert werden. Jedem *Topic* können bis zu drei Namen zugeordnet werden. Der *base Name* muss immer angegeben werden. Er steht für den Namen, unter dem das Topic eingeordnet werden kann. Der *Display Name* steht für den Namen, den eine Anwendung statt des *Base Names* anzeigt. *Sort Names* werden bei der Sortierung von *Topics* verwendet. Dies ist etwa bei der Sortierung von Namen hilfreich.
- **Occurrences und Occurrences Roles** zeigen auf Informationsobjekte, die für das *Topic* relevant sind. Dies können z.B. Grafiken oder Dokumente sein. Die Informationsobjekte befinden sich normalerweise außerhalb der *Topic Maps*. Auch die *Occurrences* können einem Typ zugeordnet werden. Diese werden *Occurrences Roles* genannt und klassifizieren die einzelnen Vorkommstypen. Die Rollen selbst sind auch wieder Topics.
- **Associations und Association Types** stellen die semantische Beziehung zwischen zwei Objekten dar. *Associations* können transitiv sein. Sie stehen für eine *ist Instanz von-* oder *hat-Beziehung*. *Associations* können auch durch einen Typ klassifiziert werden. Hierfür werden den einzelnen Beziehungen sogenannte *Association Types* zugeordnet. Es lassen sich auch sogenannte *Associations Roles* anlegen, die definieren, welche Rolle die *Topics* in *Associations* spielen.
- Neben den hier vorgestellten Elementen existieren noch weitere Zusatzkonzepte. Diese stellen Metadaten zu einer *Topic Map* zur Verfügung, wodurch es möglich wird, den Inhalt eindeutiger zu identifizieren. Dies sind *Scope, Theme, Public Subject, Identity Attribute und Facets*.

<pre> <topic id="t1" types="Thema> <topname> <basename>Hochbauprojekt <dispname>Hochbauprojekt <sortname>Hochbauprojekt </topname> <occurs> <locator role="MENTION" href="artikel1" /> <locator role="MENTION" href="artikel2" /> <locator role="MENTION" href="artikel3" /> </occurs> </topic> <topic id="t2" types="Begriffe> <topname> <basename>HOAI <dispname>HOAI <sortname>HOAI </topname> <occurs> <locator role="MENTION" href="artikel1" /> </occurs> </topic> <topic id="t3" types="Begriffe> <topname> <basename>Leistungen <dispname>Leistungen <sortname>Leistungen </topname> <occurs> <locator role="MENTION" href="artikel1" /> </occurs> </topic> </pre>	<pre> <occurs> <locator role="MENTION" href="artikel3" /> </occurs> </topic> <topic id="t4" types="Thema> <topname> <basename>Bauprojekt <dispname>Bauprojekt <sortname>Bauprojekt </topname> <occurs> <locator role="MENTION" href="artikel3" /> </occurs> </topic> <assoc types="Spezialisierung"> <assocrl role="Spezialisierung" href="#t1" /> <assocrl role="Generalisierung" href="#t4" /> </assoc> <assoc types="Assoziation"> <assocrl role="ist assoziiert mit" href="#t1" /> <assocrl role="ist assoziiert mit" href="#t2" /> </assoc> <assoc types="Assoziation"> <assocrl role="ist assoziiert mit" href="#t2" /> <assocrl role="ist assoziiert mit" href="#t3" /> </assoc> </pre>
--	--

Abbildung 8: Beispiel Topic Map

2.3.1.3 Suchverfahren

Semantische Netze oder Frames können als Graph angesehen werden. Ebenso bewegen sich regelbasierte Systeme bei der Abarbeitung von Regeln in einem Graph. Daher spielt die Graphentheorie eine wichtige Rolle in der Abarbeitung von Suchanfragen. Da die Graphentheorie zu umfangreich ist, um hier komplett erklärt zu werden, werden die verschiedenen Begrifflichkeiten nur kurz erwähnt (s. auch [GKA]):

Graph

Ein *Graph* ist definiert durch eine Menge *Knoten* und *Kanten*, wobei eine *Kante* ein ungeordnetes Paar aus zwei *Knoten* ist. Haben die *Kanten* eine vorgegebene Richtung, so spricht man von *Gerichteten Graphen*. Für jede Kante eines Graphen kann eine reelle Zahl als Gewichtungsfaktor definiert werden. Diese Graphen werden gewichtete Graphen genannt.

Pfad

Als Pfad wird ein Weg zwischen einem Start- und einem Endknoten bezeichnet. Der Weg läuft dabei jeweils über die Kanten zweier benachbarter Knoten.

Graphensuchstrategien

- **Tiefensuche.** Bei der *Tiefensuche* geht die Suchrichtung von Startknoten zunächst in die Tiefe, d.h. es wird zuerst versucht, so weit wie möglich in den Graphen einzudringen.
- **Breitensuche.** Bei der *Breitensuche* werden zunächst alle benachbarten Knoten eines Knotens untersucht und dann erst wird in die Tiefe gegangen.

2.4 Subsymbolische Verarbeitung mittels neuronaler Netze

Künstliche neuronale Netze sind Abbildungen des menschlichen Gehirns mit den Mitteln der Mathematik und Statistik, unterstützt durch die Computertechnik. Sie bestehen aus einer großen Menge miteinander verbundener Prozessoren (Units). Diese sind stark idealisiert vergleichbar mit den biologischen Nervenzellen.

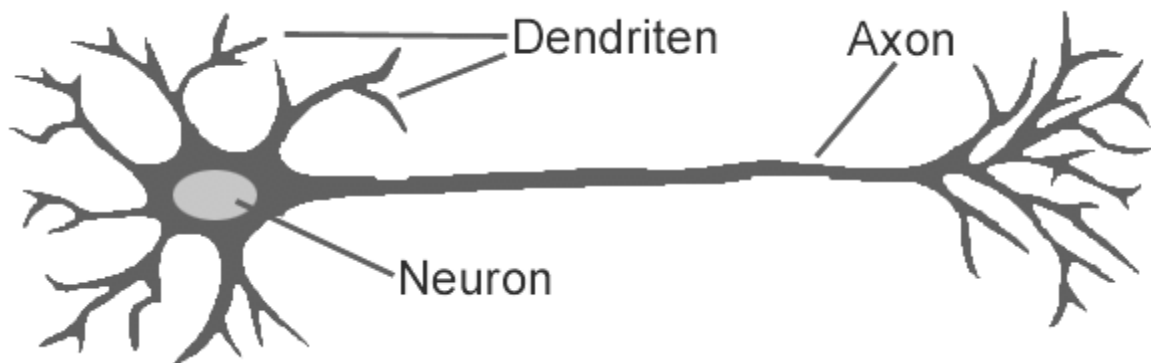


Abbildung 9: Biologisches Neuron

Die Arbeitsweise einer Nervenzelle besteht idealisiert aus drei Schritten:

- Eingangssignale (potentialverändernde Reize) werden über die Dendriten aufgenommen.
- Die Eingangssignale werden verarbeitet und führen zu einer Veränderung des Zellenpotentials.
- Überschreitet das Zellenpotential einen gewissen Schwellwert, so wird über das Axon ein Signal an andere Nervenzellen weitergegeben. Dies wird auch als „feuern“ bezeichnet.

2.4.1 Wissensrepräsentation und -verarbeitung

Nach [EINN] lässt sich ein künstliches Neuron wie folgt definieren:

Ein künstliches Neuron ist ein Tupel $(\vec{x}, \vec{w}, f_a, f_o, o)$ bestehend aus einem Eingabevektor $\vec{x} = (x_1, \dots, x_n)$, einem Gewichtsvektor $\vec{w} = (w_1, \dots, w_n)$, einer Aktivierungsfunktion f_a mit $f_a: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ und einer Ausgabefunktion f_o , für die $f_o: \mathbb{R} \rightarrow \mathbb{R}$ gilt. Dabei wird durch $f_o(f_a(\vec{x}, \vec{w})) = o$ der Ausgabewert des Neurons erzeugt.

In anderen Quellen wird auch die Aktivierungs- und Ausgabefunktion zu einer Funktion zusammengefasst. Diese wird dann als Transfer-, Aktivierungs- oder Ausgabefunktion bezeichnet.

In Analogie zu den Nervenzellen wird den künstlichen Neuronen auch häufig ein „innerer Zustand“ zugeordnet. In diesem Fall wird ein künstliches Neuron nach [EINN] wie folgt definiert:

Ein künstliches Neuron ist ein Tupel $(\vec{x}, \vec{w}, f_a, Z, f_o, o)$ bestehend aus:

1. Eingabevektor \vec{x}
2. Gewichtsvektor \vec{w}
3. Aktivitätsfunktion $f_a: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$
4. Zustand Z
5. Ausgabefunktion $f_o: \mathbb{R} \rightarrow \mathbb{R}$

Hierbei ist die Ausgabefunktion f_o eine Funktion in Abhängigkeit von Z .

Der Zustand Z hängt ab vom „Altzustand“ und der Veränderung der Aktivierungsfunktion, z.B.

$$Z^{new} = Z^{alt} + f_a(\vec{x}, \vec{w}) .$$

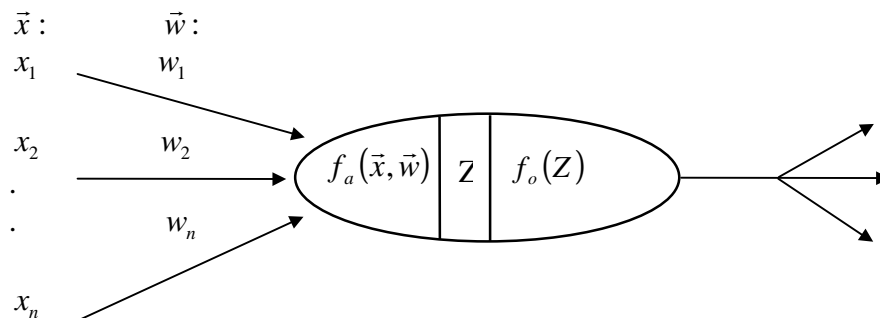


Abbildung 10: Künstliches Neuron [EINN]

In der technischen Realisierung wird der Aktivierungszustand eines Neurons unterschiedlich dargestellt. Man unterscheidet zwischen (quasi-) kontinuierlichen und diskreten Wertebereichen. Der kontinuierliche Wertebereich wird wiederum unterteilt in Modelle, die alle reellen Zahlen als Werte zulassen, und Modelle, die Intervalle als Werte zulassen. Die Implementierung der diskreten Modelle wird mit binären Werten, z.B. $\{0,1\}$ oder $\{-1,1\}$, vorgenommen.

Zum „Feuern“ einer Nachricht an andere Neuronen muss ein gewisser Schwellwert überschritten werden. Dies kann durch eine binäre Schwellwertfunktion realisiert werden, jedoch wird hierbei nicht wie beim biologischem Vorbild die Intensität der aufeinanderfolgenden Aktionspotentiale modelliert. Hierfür bieten sich lineare oder sigmoide (s-förmige) Ausgabefunktionen an.

Der Schwellwert (Bias) kann auf zwei Arten realisiert werden. Zum einen als Parameter in der Aktivierungsfunktion, und zum anderen über einen zusätzlichen gewichteten Eingang.

Das Neuronale Netz entsteht aus der Verbindung mehrerer Neuronen miteinander. Nach [EINN] ist es definiert durch:

Ein Neuronales Netz ist ein Paar (N,V) mit einer Menge N von Neuronen und einer Menge V von Verbindungen. Es besitzt die Struktur eines gerichteten Graphen, für den die folgenden Einschränkungen und Zusätze gelten:

- i. Die Knoten des Graphen heißen Neuronen.*
- ii. Die Kanten heißen Verbindungen.*
- iii. Jedes Neuron kann eine beliebige Menge von Verbindungen empfangen, über die das Neuron seine Eingaben erhält.*
- iv. Jedes Neuron kann genau eine Ausgabe über eine beliebige Menge von Verbindungen aussenden.*
- v. Das Neuronale Netz erhält aus Verbindungen, die der „Außenwelt“ entspringen, Eingaben und gibt seine Ausgaben über in der „Außenwelt“ endende Verbindungen ab.*

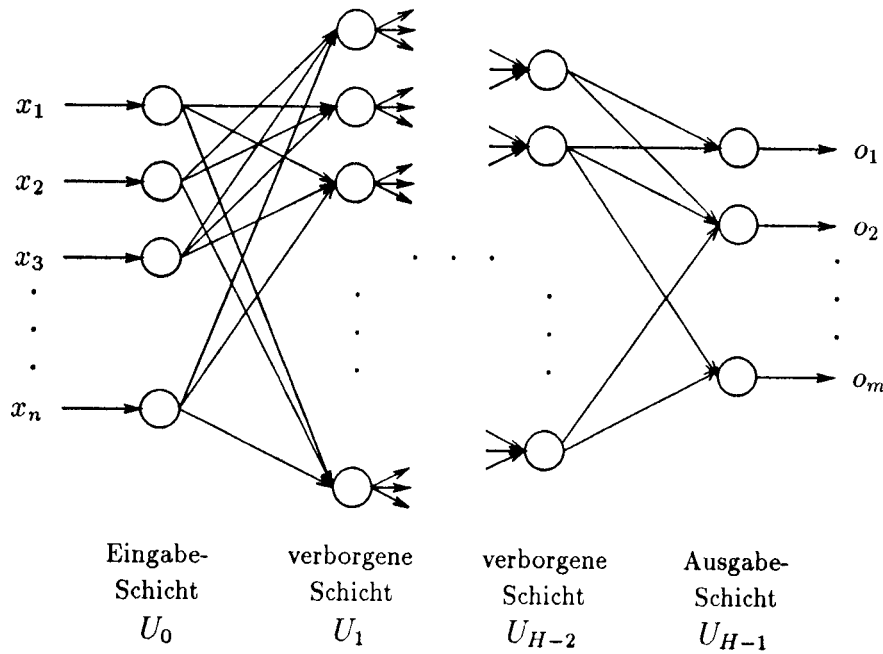


Abbildung 11: Schichten eines neuronalen Netzes [EINNN]

Neuronale Netze lassen sich aufgrund ihrer Anordnung (Topologie) klassifizieren:

Netze ohne Rückkopplung (Feedforward-Netze)

Die Topologie dieser Netze ist mathematische ein azyklischer Graph, d.h. es existiert kein Pfad, der von einem Neuron, sei es direkt oder über andere Neuronen, wieder zum Neuron selbst führt. Ein Netz wird als „ebenenweise verbunden“ (Feedforward-Netze) bezeichnet, wenn es nur Verbindungen von einem Neuron zum nächsten aufweist.

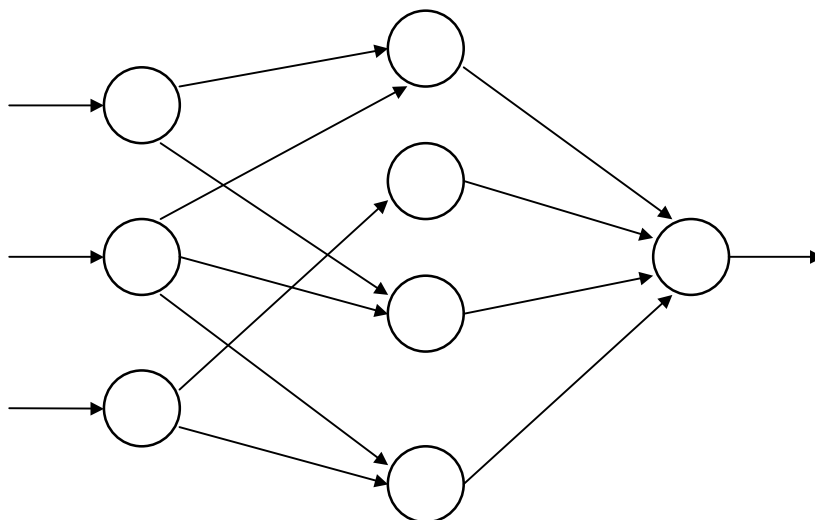


Abbildung 12: Feedforward Netz

Hingegen werden Netze, in denen es Neuronen gibt, die Verbindungen zu aufeinanderfolgende Neuronen und ebenso zu Neuronen in einer höheren Ebene haben, als „Allgemeine Feedforward-Netze (mit shortcut connections)“ bezeichnet.

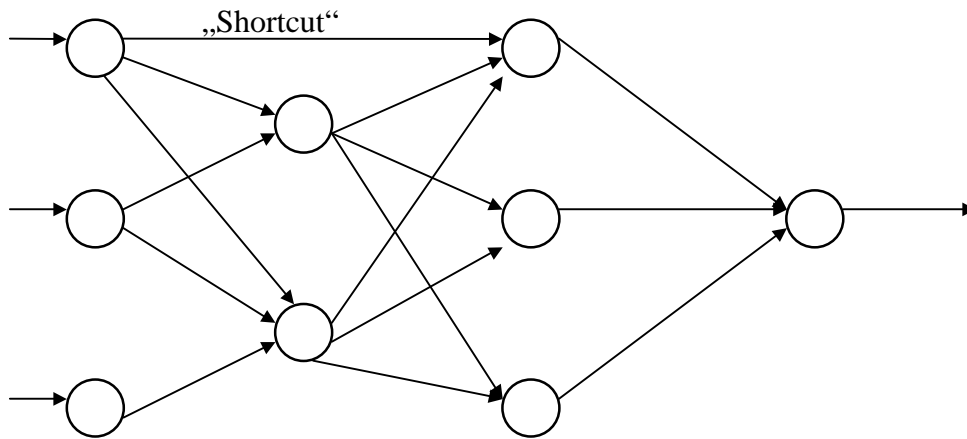


Abbildung 13: Feedforward-Netz mit "Shortcut"

Diese werden auch als vollständig bezeichnet, wenn jedes Neuron einer Schicht k mit jedem Neuron einer Schicht $k+1$ verbunden ist.

Netze mit Rückkopplungen (Recurrente Netze)

Es werden vier Klassen von Netzen mit Rückkopplung unterschieden:

1. Netze mit direkten Rückkopplungen (Direct feedback) erlauben Verstärkung bzw. Abschwächung eines Neurons durch sich selbst.
2. Netze mit indirekten Rückkopplungen (Indirect Feedback) haben eine Verbindung von einem Neuron höherer Ebene zu einem Neuron niedriger Ebene.
3. Netze mit Rückkopplungen innerhalb einer Schicht (Lateral feedback). Diese Topologie wird auch als „winner takes all“-Netzwerk bezeichnet und wird eingesetzt, wenn in einer Gruppe von Neuronen nur ein Neuron aktiv werden soll. Jedes Neuron einer Schicht enthält dann hemmende (inhibitorische) Verbindungen zu anderen Neuronen und aktivierende (exzitatorische) Verbindungen zu sich selbst. Das Neuron mit dem größten Aktivierungspotential hemmt dann alle anderen Neuronen der Schicht.
4. Vollständig verbundene Netze besitzen Verbindungen zwischen allen Neuronen.

Das Netz lernt durch Selbstmodifikation gemäß fest vorgegebener Lernregeln. Der Lernprozess besteht aus:

- Entwicklung neuer Verbindungen
- Löschen existierender Verbindungen

- Modifikation der Stärken der Verbindungen (Gewichtung)
- Veränderung des Schwellenwertes
- Abwandlung der Aktivierungs- bzw.- Ausgabefunktion
- Entwicklung neuer Neuronen
- Löschen bestehender Neuronen

Das am häufigsten verwendete Lernverfahren ist die Modifikation der Stärke der Verbindung.

Das Lernen lässt sich in drei Klassen aufteilen. Diese werden anhand des Lernparadigmas gebildet:

- Für das **Überwachte Lernen** (Supervised learning) wird ein Satz von Trainingsdaten benötigt, der aus Paaren von Ein- und Ausgabedaten besteht. Dem Netz wird nach jeder Eingabe die Differenz der Ausgabe zur korrekten Ausgabe mitgeteilt. Anhand dieser Differenz wird das Netz entsprechend modifiziert.
- Beim **bestärkenden Lernen** (Reinforcement learning) wird dem Netz lediglich mitgeteilt, ob die Ausgabe richtig oder falsch war.
- **Unüberwachtes Lernen** (unsupervised learning oder self-organized learning). Hier versucht das Netz, die Eingaben selbstständig (ohne Hilfe von Außen) in Ähnlichkeitsklassen einzuteilen.

Die Repräsentation und Verarbeitung des Wissens in einem künstlichen neuronalen Netz wird als subsymbolisch bezeichnet, da keine Symbole wie bei der symbolischen Verarbeitung als Informationsträger verwendet werden. Die Aktivierungsmuster des Netzes, repräsentiert durch die Gewichtung der Verbindungen, stellen die Wissensrepräsentation dar.

Vorteile:

Einer der Hauptvorteile künstlicher neuronaler Netze ist, dass Lernalgorithmen existieren, d.h. das das Netz sich lernend an einen Problembereich anpasst. Eine problemspezifische Programmierung entfällt. Jedoch sind für unterschiedliche Problemfälle verschiedene Netz-Topologien günstiger oder ungünstiger, was einen nicht zu vernachlässigenden Aufwand darstellt bzw. eine gute Sachkenntnis voraussetzt.

Ein weiterer Vorteil ist, dass ein neuronales Netz mit unsicheren und unvollständigen Eingaben umgehen kann.

Nachteile:

Ein Nachteil der automatischen Klassifizierung ist die niemals vollständig gesicherte Zuverlässigkeit der Ergebnisse.

Ein weiterer Nachteil eines künstlichen neuronalen Netzes ist die geringe Transparenz. Einzelne Gewichtungen lassen sich nur sehr schwer interpretieren. Es liegt nicht offen, was das Netz weiß bzw. auf welcher Grundlage bestimmte Entscheidungen getroffen werden.

Serielle Computer mit einem Prozessor sind aufgrund der parallelen Arbeitsweise schlecht geeignet. Hierfür bieten sich spezielle Hardwarerealisationen an, die eine parallele Verarbeitung ermöglichen (Mehrprozessor-Systeme, spezielle Neuro-Chips).

2.5 Gegenüberstellung symbolische- und subsymbolische Wissensverarbeitung

Subsymbolische Wissensverarbeitung	Symbolische Wissensverarbeitung
Das zugrunde liegende Paradigma der Intelligenz ist das Lernen.	Das zugrunde liegende Paradigma der Intelligenz ist das Problemlösen.
Wissenserwerb erfolgt durch Neustrukturierung mit Hilfe von Lernalgorithmen.	Wissenserwerb erfolgt durch Strukturanalyse eines Problembereichs und dessen symbolische Abbildung.
Das Wissen ist implizit in der Verbindungsstruktur des Netzes enthalten.	Das Wissen wird explizit durch Symbole repräsentiert.
Die Auswertung des Wissens erfolgt durch parallele Aktivationsausbreitung	Die Auswertung des Wissens wird aufgrund logischer Schlussfolgerungen getroffen.

Abbildung 14: Gegenüberstellung symbolischer und subsymbolischer Wissensverarbeitung

	Subsymbolischer Ansatz	Symbolischer Ansatz
Lernfähigkeit	Leicht	Schwer
Erklärungsfähigkeit	Keine	Gut
Verarbeitungsgeschwindigkeit	Langsam	Schnell
Algorithmus	Parallel	Sequentiell
Umgang mit verrauschten Daten	Gut	Schlecht
Generalisierung	Leicht	Schwierig
Wissensrepräsentation	Schwierig	Leicht

Abbildung 15: Gegenüberstellung symbolischer und subsymbolischer Ansätze nach [HWPD]

3 Softwaretechnische Umsetzungen

3.1 Intelligent views mit K-Infinity

Die intelligent views GmbH (www.i-views.de) entstand 1997 als Ausgründung aus dem Institut für Integrierte Publikations- und Informationssystem (IPSI) der GMD Darmstadt und ist spezialisiert auf die Bereiche internes Wissensmanagement, E-Business in Form von Internet/Intranet Portale, Dokumentenmanagement und Redaktionssysteme. K-Infinity ist die Software für den Aufbau, die Pflege und die Nutzung von semantischen Netzen. K-Infinity ist weitestgehend plattformunabhängig und besitzt einen modularen Aufbau:

- Die **Editing-Tools** bestehen aus dem *Knowledge-Builder* und dem *Markup-Tool*. Mit dem *Knowledge-Builder* können Wissensnetze selbst gestaltet bzw. abgeändert werden. Das *Markup-Tool* dient der semantischen Indexierung von Inhalten wie Texten, Tabellen, Bildern und anderen Medien. So können z.B. Textdokumente mit Begriffen aus dem Wissensnetz verbunden werden. Die Bandbreite der semantischen Indexierung reicht von interaktiv bis vollautomatisch.
- Die **Usage-Tools** sind für Suche, Aufbereitung und Anzeige von Wissen zuständig. Sie untergliedern sich in *Semantic-Finder*, *Net-Navigator*, *Semantic-Search-Language*, *Visulisation-Engine* und *Layout-Engine*. Der *Semantic-Finder* ermöglicht Suchanfragen auf Basis aller Objekte des Wissensnetzes. Suchanfragen lassen sich leicht grafisch per Drag-and-Drop erstellen. Der *Net-Navigator* bietet die Möglichkeit, Wissensnetze thematisch zu durchsuchen. Er bietet einen übersichtlichen Ausschnitt des Wissensnetzes. Es lassen sich einfach zu einem Begriff alle Nachbarbegriffe anzeigen. Die *Semantic-Search-Language* stellt eine Reihe von Sprachkonstrukten für den Zugriff auf Wissensnetze als allgemeine API bereit. Die *Visulisation-Engine* stellt statistisches Zahlenmaterial oder Faktendaten in Form von Diagrammen und Themenkarten grafisch dar. Die *Layout-Engine* fügt mediale Wissensinhalte (Texte, Grafiken, Tabellen, Bilder, etc.), die mithilfe des Wissensnetzes aus unterschiedlichen Wissensquellen zusammengesucht wurden, vollautomatisch zu einem effektiven Seitenlayout zusammen und ermöglicht damit die dynamische Generierung von Webseiten.
- Die **Add-on-Tools** setzen sich aus dem *Resource-Bus* und den *Mapping-Tools* zusammen. Sie übernehmen die Integrationsfunktion von heterogenen Wissensbeständen.
- Die **Production-Tools** erlauben die Extraktion von Teilnetzen sowie der verbundenen Inhalte, um damit marktgerechte Spin-off-Produkte zu erzeugen.

3.1.1 Die Technologie

Die Grundtechnologie der Wissensrepräsentationen sind hybride Frame-Repräsentationen. Es existieren verschiedene Sichten auf das Wissen. Zum einen die frameartige Darstellung mit

Slots, Sloteträgen und Beziehungen, dann eine netzartige Darstellung mit Knoten und Kanten. Die Module von K-Infinity wurden unter Smalltalk entwickelt.

Die Elemente des Wissensnetzes setzen sich zusammen aus:

- Begriffen
- Individuen
- Relationen
- Attributen

Begriffe

Begriffe sind die Grundbausteine des Wissensnetzes. Sie sind eher allgemeiner Natur und stehen in der Regel für eine ganze Klasse konkreter Objekte. Typische Begriffe sind „Person“, „Gegenstand“, „Beruf“, etc. Die Begriffe können untereinander vernetzt werden, d.h. es werden Relationen zwischen den Begriffen gezogen. Die einfachste Form der Vernetzung ist die Zuordnung von Ober-/Unterbegriffen. Es existiert ein Wurzelbegriff, der nur Unterbegriffe hat. Er steht an der Spitze der Hierarchie, und von ihm werden alle anderen Begriffe abgeleitet, ähnlich der Generalisierung/Spezialisierung bei der objektorientierten Programmierung. In der Gliederung muss sich nicht zwangsläufig eine Baumstruktur ausbilden. Ein Begriff kann durchaus mehrere Oberbegriffe haben, analog der Mehrfachvererbung in C++. Ein Begriff ist als eine Art Schema-Definition für die konkreten Instanzen (Individuen) anzusehen, d.h. der Begriff definiert die Art und die Anzahl der Attribute bzw. der Relationen, die ein Individuum annehmen kann. Die konkreten Werte werden dann beim Individuum gesetzt. Zu beachten ist, dass auch Attribute und Relationen für Begriffe vergeben werden können. So hat z.B. jeder Begriff das Attribut *Namen*, das den Namen des Begriffs speichert. Ebenso werden zwischen den Begriffen Systemrelationen gezogen, die z.B. die Vererbung anzeigen.

Ein Begriff kann drei Arten von Ausprägungen haben:

- Der Begriff ist abstrakt. Es können nur Unterbegriffe und keine Individuen von diesem Begriff gebildet werden
- Er ist instanzenfähig, d.h. es können konkrete Individuen von diesem Begriff gebildet werden. Beispiel: Max Mustermann ist ein Individuum von dem Begriff Person.
- Ein Begriff kann Individuen erweitern (Rolle). Eine Rolle ist immer abstrakt und kann somit keine Individuen ausbilden.

Individuen

Konkrete Objekte werden als Instanzen von Begriffen im Wissensnetz abgelegt. So wird z.B. eine Stütze S1 als Individuum des Begriffs Stütze angelegt. Dem Individuum stehen die im dazugehörigen Begriff definierten Attribute und Relationen zur Verfügung.

Relationen

Die Begriffe und Individuen des Wissensnetzes werden über die Relationen miteinander verknüpft. Jeder neu angelegte Begriff bzw. jedes neu angelegte Individuum hat zumindest eine Relation. Ein Begriff hat die Relation *ist Unterbegriff von* zu seinem Oberbegriff. Ein Individuum hat die Relation *ist Instanz von* zu dem Begriff, von dem es abgeleitet wurde.

Es lassen sich beliebig viele benutzerdefinierte Relationen im System anlegen. Die Relationen können zwischen den Begriffen oder den Individuen definiert werden. Eine Relation kann auch abstrakt definiert werden, d.h. nur von nicht abstrakten Unterrelationen dieser Relation können konkrete Beziehungen zwischen zwei Objekten hergestellt werden.

Relationen werden jeweils als Paar angelegt, d.h. eine Relation besteht aus der Relation und der *inversen* Relation. Die Relation beschreibt die Beziehung zwischen Objekt1 und Objekt2, die *inverse* Relation die Beziehung aus sich von Objekt2 nach Objekt1. Beispiel:

Begriff1 *ist Individuum von* Begriff2
Begriff2 *hat Individuum* Begriff1

Einen Sonderfall stellt eine symmetrische Relation dar. Hier sind die Relation und die inverse Relation gleich. Beispiel:

Person *ist verwandt mit* Person

Relationen können auch obligatorisch sein, d.h. alle neu angelegten Individuen eines Begriffs, der eine obligatorische Relation im Individuen-Schema definiert hat, müssen eben diese Relation ausbilden.

Erweiterungen (Rollen)

Eine *Rolle* ist eine spezielle Ausprägung eines Begriffs. Der Begriff ist gleichzeitig abstrakt, d.h. es können keine konkreten Individuen angelegt werden. Individuen haben jedoch die Möglichkeit, eine *Rolle* zu *erweitern*. Dies dient dazu, Individuen, die eindeutig durch ihre „ist Instanz von“-Relation einem Begriff zugeordnet sind, mit einem anderen Begriff zu verknüpfen, um ebenfalls dessen Attribut- und Relationsschema zu übernehmen.

Erweiterungen werden z.B. dazu benutzt, einem Individuum bestimmte Funktionen bzw. eine Rolle zuzuweisen, die es nur temporär annimmt. Max Mustermann ist ein Individuum von Person. Ist sein Beruf z.B. Bauingenieur, so nimmt er die Rolle Bauingenieur an. Hat er ebenfalls noch ein Studium in Architektur abgeschlossen, so kann auch er auch noch die Rolle Architekt annehmen.

Attribute

Attribute dienen dazu, den Begriffen bzw. Individuen und Relationen zusätzliche Werte zuzuordnen. So besitzt z.B. jedes Objekt, egal ob Begriff, Individuum oder Relation, das Attribut *Name*. Dieses Attribut hat den Typ Zeichenkette (String) und bezeichnet den Namen eines Begriffs.

Ein Attribut besteht aus einem Namen und dem Datentyp, den das Attribut annehmen kann. Dies ist analog zu typisierten Programmiersprachen oder Datenbanken. Darüber hinaus kann bestimmt werden, ob ein Begriff nur genau ein Attribut mit diesem Namen hat, oder ob mehrere Attribute mit diesem Namen möglich sind. Ebenfalls kann bei der Erstellung des Attributes angegeben werden, ob der Wert des Attributes als Synonym für den Begriff oder das Individuum stehen soll.

Suchen

Es existieren semantische *Matchingverfahren*, die die Ähnlichkeit beliebiger vom Wissensnetz indexierter Inhalte berechnen. Die Ähnlichkeit wird hier über die *Konnektivität* des semantischen Netzes bestimmt.

3.2 USU AG mit dem KnowledgeMiner

Unter dem Leitgedanken "Wissen ist Markt" entwickelt und vertreibt die USU AG (www.usu.de) Produkte und Lösungen für die ganzheitliche Integration von Unternehmenswissen in die Geschäftsprozesse und Anwendungen von Unternehmen. Die drei Geschäftsbereiche des 1977 gegründeten schwäbischen Softwarehauses bieten Strategien, Produkte und Lösungen für Knowledge Management, für IT-Controlling und e-Business Integration. Die Knowledge Management Lösung der USU AG ist der KnowledgeMiner, eine Metasuchmaschine, die über eine einheitliche Benutzeroberfläche mit Hilfe von Themennetzen (nach ISO 13250) den Zugriff auf mehrere heterogene Datenquellen ermöglicht.

Die Hauptfunktionen des KnowledgeMiner sind die Durchführung von Datenrecherchen in einem externen Datenbestand, die Erstellung von komplexen Suchanweisungen und deren Wiederverwendbarkeit, sowie ein Verfahren zum automatisierten Erstellen von Inhaltsangaben von Textdokumenten. Der USU KnowledgeMiner navigiert den Anwender durch seine Suchthemen, und weist auf über- und untergeordnete Beziehungen hin. Zu den

gefundenen Dokumenten werden automatisch Inhaltsangaben generiert, die anzeigen, warum ein Dokument gefunden wurde und was es beinhaltet.

3.2.1 Die Technologie

Grundlage der Technologie ist die Trennung von Dokument- und Zugangswissen mit Hilfe von Topic Maps (Themennetzen). Der USU KnowledgeMiner selbst enthält keine standardmäßig eingebundene Datenbank, sondern greift über gängige Suchmaschinen auf verschiedene Datenquellen zu. Abhängig von der jeweiligen Suchmaschine sind Dokumente im Intranet, Dateien in Dateisystemen, Notes-Datenbanken und ODBC-fähige relationale Datenbanken einsetzbar. Lediglich die Topic Maps selbst werden in einer relationalen Datenbank abgelegt.

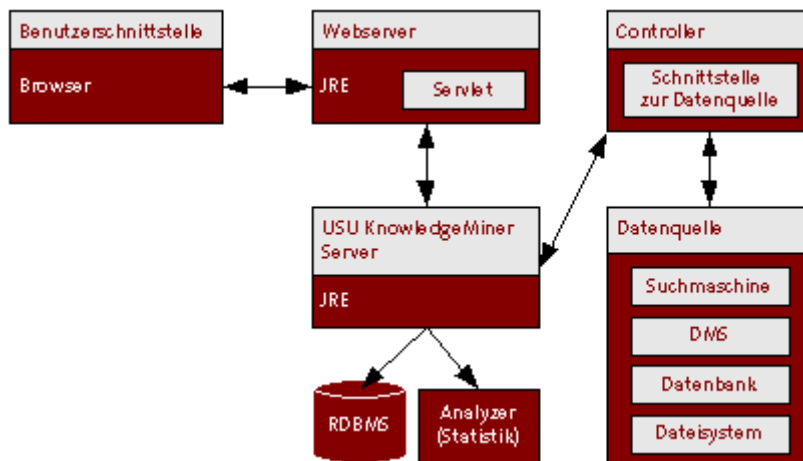


Abbildung 16: USU KnowledgeMiner [USU]

Es werden je nach Schwerpunkt der Kunden verschiedene Volltext-Suchanfragen für die zugrundeliegenden Suchmaschinen in Form semantischer Abfragenetze als *Topic Maps* abgelegt. Für den Anwender bilden die Knoten und Kanten des Netzes die verschiedenen Suchelemente für eine Suchanfrage. Die semantischen Strukturen werden dabei über die Recherchetätigkeit der Nutzer aktiv gepflegt und weiterentwickelt.

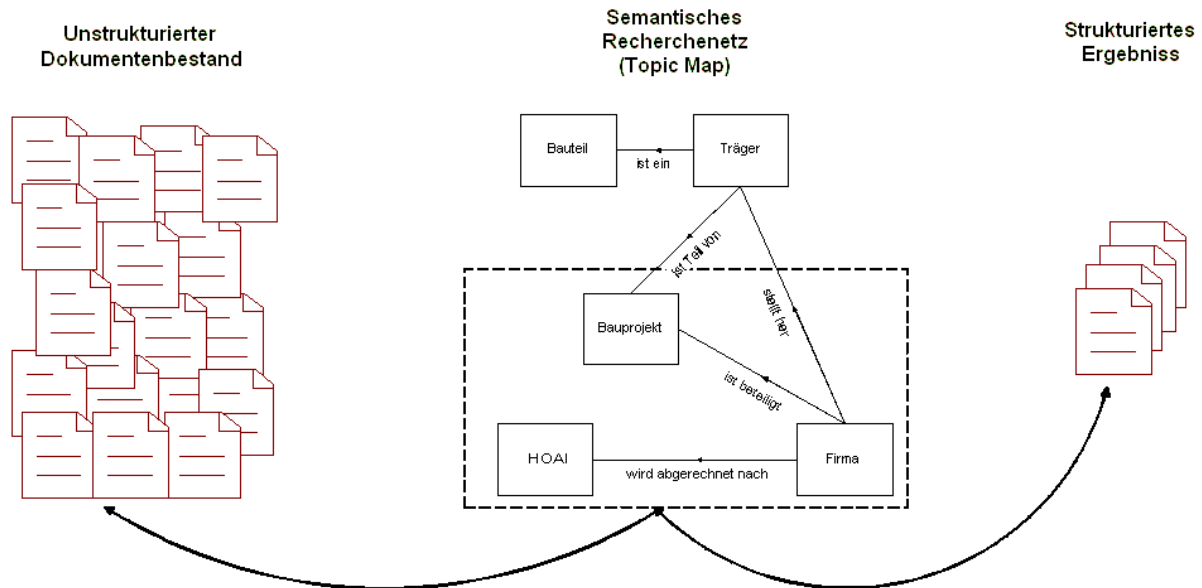


Abbildung 17: KnowledgeMiner - Topic Map

3.3 SER Systems AG mit SERware

Mit SERware (www.ser.de) stellt SER eine Produktfamilie zur Unterstützung des unternehmensweiten Wissensmanagements bereit. Ihren Anforderungen entsprechend sind die einzelnen Produkte wiederum in drei Software-Suiten zusammengefasst:

- **SERdocware** ist ein wissensbasiertes Dokumentenmanagementsystem, das die Funktionen zur Speicherung und Archivierung, sowie den Zugriff auf alle digitalen Informationen beliebigen Formats und beliebiger Struktur innerhalb von SERware bereitstellt. Auch die klassische papiergebundene Korrespondenz und Faxe können damit nach entsprechender Aufbereitung (OCR) automatisch klassifiziert und eingeordnet werden.
- **SERkmware** enthält die Kerntechnologie der *Knowledge Management Suite* von SERware, die Software Engine *SERbrainware*. *SERbrainware* kann beliebige Informationen "lesend verstehen" und inhaltsbezogen klassifizieren. Zusätzlich findet *SERbrainware* im wissensbasierten Zugriff (Such-Modus) alle Informationen, die dem als Anfragemuster gewählten Text ähnlich sind. Ebenfalls enthalten ist ein *Knowledge Broker*, um den Zugang und Zugriff auf Wissensbestände zu ermöglichen, und diese zu verwalten.
- **SERprocess** beinhaltet Funktionen, die für die Modellierung von Informationsverarbeitungs-Prozessen sowie deren Steuerung und Kontrolle während ihrer Ausführung erforderlich sind.

Darüber hinaus wird ein C++-Interface bereitgestellt, dessen verschiedene Layer für eine Anbindung an andere Umgebungen sorgen.

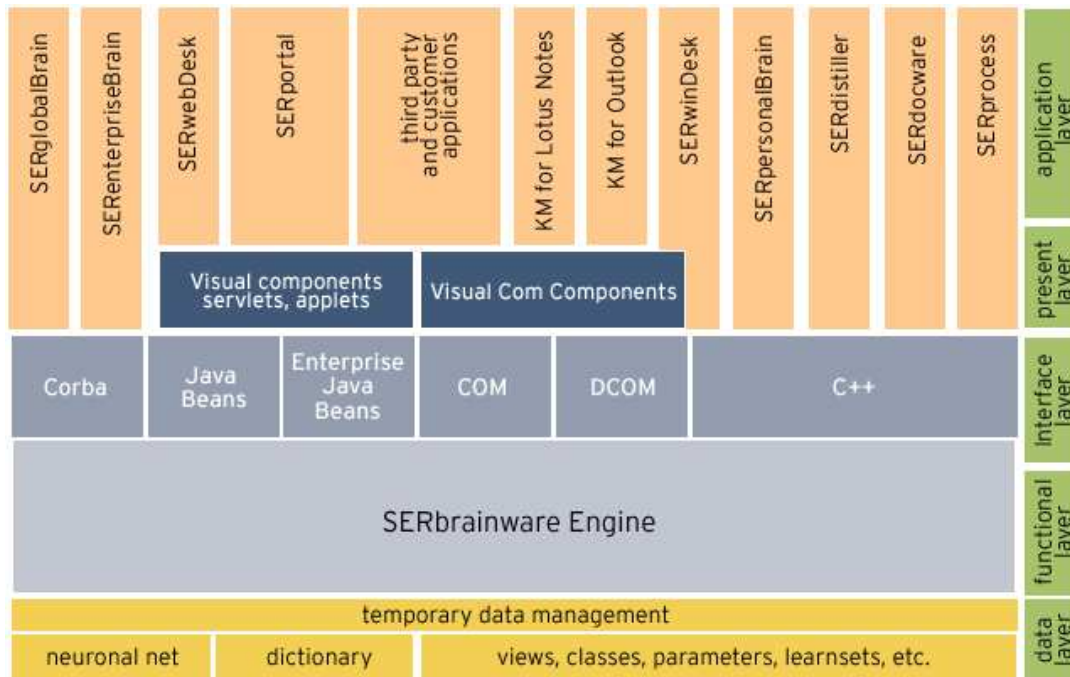
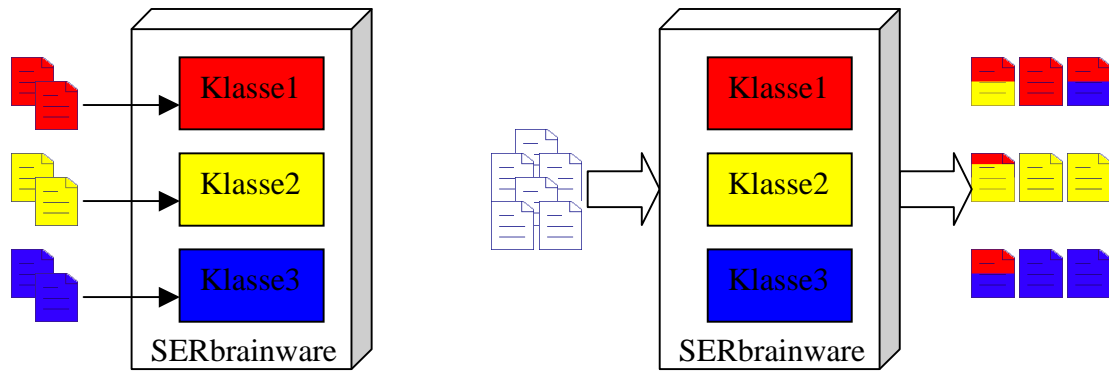


Abbildung 18: SER Brainware – Layer [SER]

3.3.1 Die Technologie

SERbrainware basiert auf der Technologie künstlicher neuronaler Netze. Das Netz muss erst durch *Lernen* aufgebaut werden. Dies erfolgt in zwei Schritten. Zuerst wird dem System ein einfaches Klassenschema als Ordnungsschema vorgegeben. Danach wird jeder Klasse eine Anzahl von Textbeispielen zugeordnet, auf Grund derer *SERbrainware* die wesentlichen Unterscheidungsmerkmale bestimmt, die typisch für die Zuordnung eines Dokumentes zur Klasse sind. Diese Unterscheidungsmerkmale werden auch als „Wissensstrukturen“ oder, in Anlehnung an die neuronalen Netze, einfach als Netze bezeichnet. Nach diesem Lernprozess kann *SERbrainware* neue Texte in die gelernten Kategorien (Klassen) einordnen. Auch ist es möglich, für eine Dokumentenmenge mehrere logisch unabhängige Klassenschemata zu definieren. Diese werden als Sichten (Views) bezeichnet.



Lernen bekannter Dokumente

Klassifizierung unbekannter Dokumente

Abbildung 19: SERbrainware - Lernen und Klassifizieren

Bei der automatischen Klassifizierung wird jedem Dokument pro im System vorhandener Klasse ein Wahrscheinlichkeitswert zwischen 0 und 100 Prozent zugeordnet. Dieser Wert gibt an, mit welcher Wahrscheinlichkeit das System einem Dokument eine Klasse zuteilen würde. Abhängig von der Aufgabenstellung des Klassifizierungsproblems kann der Anwender verschiedene Regeln vorgeben, wann ein Dokument einer bestimmten Klasse zugeordnet werden soll, und wann nicht.

Beispiel:

„Wenn der Wahrscheinlichkeitswert für die beste Klasse K_i größer 80% (dieser Wert wird als „automatic classify threshold“ bezeichnet) ist und der Abstand zu der nächst schlechteren Klasse K_j mindestens 40% (= „minimal next-neighbour gap“) beträgt, dann ordne das Dokument der Klasse K_i zu.“

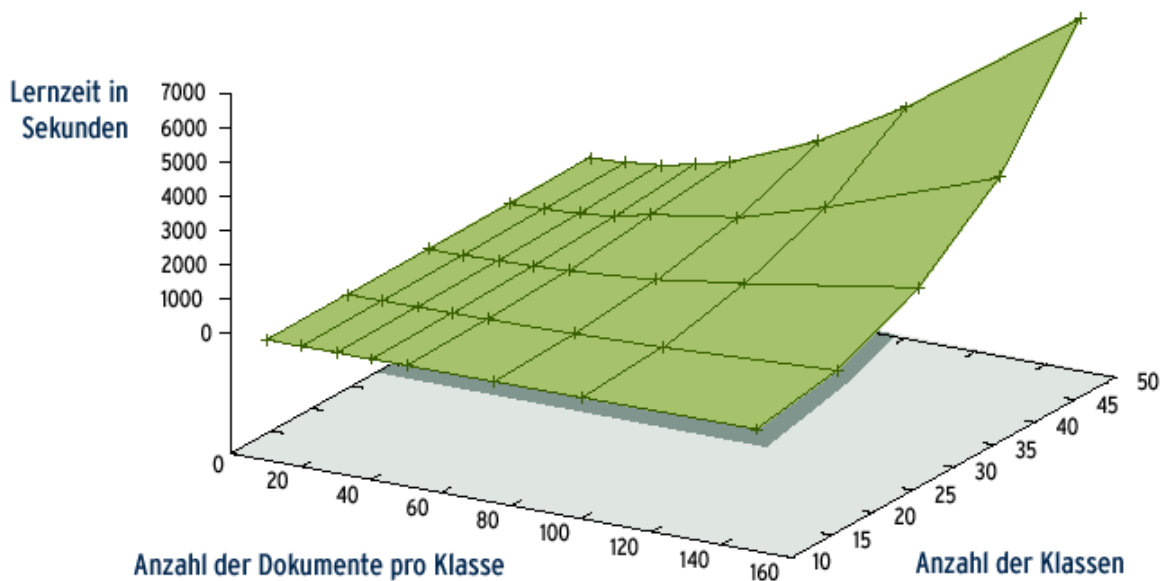


Abbildung 20: SERbrainware – Lernzeit [SER]

3.4 Autonomy

Autonomy (www.autonomy.com) bietet für die drei Geschäftsbereiche *Wissensmanagement*, *New-Media-Content-Publishing* und *E-Commerce* verschiedene Softwaresuiten an:

Autonomy Server

Der *Autonomy Server* ist das Herzstück aller Softwarelösungen von Autonomy. Er hat folgende Funktionen:

- Automatisiertes Erstellen von Kategorien, Querverweisen, Hyperlinks und die Präsentation der abgelegten Informationen.
- Automatische Zusammenfassung von Dokumenten und auf Abruf generierte Verweise auf inhaltlich ähnliche Quellen.
- Bereitstellen einer einheitlichen grafischen Benutzungsschnittstelle zur Ansicht der verschiedenen Datenquellen.

Autonomy Application Builder

Der *Application Builder* bietet eine Reihe von in Funktionseinheiten unterteilten Anwendungsprogrammchnittstellen (API), die es ermöglichen, die Autonomy-Produkte an eigene Anforderungen anzupassen.

Autonomy Portal-in-a-Box

Autonomy Portal-in-a-Box ist eine Portal-Lösung, die es erlaubt, Informationen automatisch zu pflegen und über das sogenannte *Enterprise Information Portal* zu präsentieren. So werden interne und externe Informationen an einer einzigen Stelle zugänglich.

Active Knowledge

Beim Erstellen von Dokumenten überprüft *Active Knowledge* die Eingaben und kann automatisiert *Echtzeit-Links* zu relevanten Informationen erstellen.

Autonomy Daily Briefing

Autonomy Daily Briefing überwacht eine bestimmte Anzahl von Internet- und Intranet-Seiten auf Neuigkeiten oder Veränderungen, die dem Interessensprofil des jeweiligen Nutzers entsprechen.

Dies ist nur ein kurzer Überblick über die verschiedenen Softwarelösungen von Autonomy. Es werden darüber hinaus noch weitere Softwarekomponenten wie Shop-Systeme usw. angeboten.

3.4.1 Technologie

Die Software von Autonomy basiert auf den *Prinzipien der Informationstheorie* von Claude Shannon, dem Theorem von Thomas Bayes aus der Wahrscheinlichkeitstheorie und dem Ansatz der neuronalen Netze.

Bayes befasste sich mit der Berechnung der Wahrscheinlichkeitsbeziehungen zwischen mehreren Variablen, und der Wirkung einer Variablen auf eine andere. Dies besagt, dass bestimmte Ereignisse in aller Regel auf genau definierbare Ursachen zurückzuführen sind. Dieser Ansatz dient bei Autonomy als Grundlage zur Analyse unstrukturierter Daten, da sie die Einschätzung mit bereits bekannten Informationen verknüpft.

Die Informationstheorie von Shannon besagt, dass der Wert von Informationen innerhalb der Kommunikation quantifizierbar ist, d.h. je seltener ein Begriff innerhalb eines Kontextes vorkommt, desto größer ist sein Informationsgehalt. Oft auftauchende Wörter wie Adjektive oder Verben können ignoriert werden. Auf diesen Ansatz stützt sich die Begriffserkennung von Autonomy.

Diese Techniken ermöglichen das Erkennen von Textmustern („Konzepten“) und das automatische Erkennen eben dieser Konzepte in anderen Quellen. Mittels *Pattern matching* (Mustererkennung) versucht die Software aufgrund der Häufigkeit und der Beziehung der Begriffe innerhalb eines Kontextes zueinander, die Kernaussagen eines Textes zu analysieren und auf deren Bedeutung zu schließen. Ist die Kernaussage eines Textes gefunden, werden die zugrundeliegenden Muster kodiert und in die Software übernommen. Die inhaltliche Zusammenfassung („Abstracts“) wird als *Konzept-Agent* bezeichnet, der in der Lage ist, sinnverwandte Inhalte in anderen Quellen zu finden. Da sich das Verfahren nicht auf spezifische linguistische und semantische Strukturen stützt, funktioniert die Zuordnung in jeder Sprache.

Die Grundlage aller Produkte von Autonomy bildet die *Dynamic Reasoning Engine (DRE)*. Sie bietet im wesentlichen folgende vier Funktionen:

- Vergleich von Inhalten. Die DRE analysiert eine Quelle und liefert Verweise zu anderen Dokumenten, die ein hohes Maß an inhaltlicher Übereinstimmung aufweisen.
- Erstellung von Agenten. Als Eingabe erhält die DRE einen Text, einen Übungssatz oder eine Auswahl von Dokumenten und liefert einen Konzept-Agenten zurück.
- Training von Agenten. Bestehende Agenten werden anhand einer Auswahl von Dokumenten trainiert.

- Textsuche. Die DRE nimmt einen Booleschen Ausdruck oder eine in natürlicher Sprache formulierte Abfrage entgegen und liefert die entsprechenden Dokumente, die diese Begriffe beinhalten.

Das Training der Agenten entspricht dem Training der neuronalen Netzen, d.h. die DRE erhält den zu trainierenden Agenten und eine Auswahl zu analysierender Texte als Eingang. Der Agent analysiert dann die Texte und bestimmt die wichtigsten Ideen darin. Der Anwender kann dann zur Feinabstimmung der Beziehungen weitere Dokumente oder Textabschnitte zur Analyse übergeben. Der Agent gewichtet Konzepte, die im ursprünglichen Kontext gefunden wurden, in den Lerntexten aber nicht mehr vorkommen, geringer, oder sie fallen sogar ganz heraus.

Um eine möglichst hohe Effizienz der Agenten zu gewährleisten, sollten sie jeweils auf ein spezifisches Thema hin ausgerichtet sein. Eine Anwendung sollte daher immer aus einer Gruppe von spezifischen Agenten bestehen.

Die Konzept-Agenten bestehen aus einem sehr kleinen Satz von Daten (<1K), die in verschiedenen Anwendungen eingesetzt werden können. So kann ein Agent, der von einer bestimmten Anwendung erstellt wurde, ohne Probleme in eine andere Anwendung portiert werden.

3.5 Weitere Unternehmen

Es existieren zahlreiche weitere Anbieter von Wissensmanagement-Systemen, die aber nicht alle im Rahmen der Diplomarbeit ausführlich behandelt werden können. Daher werden weitere Firmen hier nur namentlich erwähnt.

- Borgs Industries GmbH (www.borgs.com)
- Comma Soft AG (www.comma-soft.com)
- Convera (www.convera.com)
- Ceyoniq (www.ceyoniq.com)
- Hummingbird (www.hummingbird.com)
- Hyperwave (www.hyperwave.de)
- Knowledge AG (www.knowledgepark-ag.de)
- Lotus (www.lotus.de)
- Step (www.step.de)
- Open Text (www.opentext.com)
- Ontoprise (www.ontoprise.de)
- Pironet (www.pironet-ndh.com)
- Verity (www.verity.com)
- Webfair (www.webfair.com)

3.6 Gegenüberstellung der verschiedenen Technologien

Die ausführlich genannten Firmen sind alle in derselben Branche tätig, verfolgen aber zum Teil unterschiedliche Ziele. Daraus folgt auch der unterschiedliche Einsatz von Technologien. Sowohl die Produkte von SER als auch von Autonomy sind stark auf das automatisierte Lesen von Dokumenten spezialisiert, d.h. beide Systeme versuchen anhand von neuronalen Netzen, Dokumente automatisch zu „verstehen“ und zu klassifizieren. Der Schwerpunkt bei beiden Firmen liegt auf dem Automatisierungsprozess, d.h. neue Dokumente ohne Unterstützung des Anwenders zu klassifizieren bzw. aufgrund einer Suchanfrage eben diese Dokumente zu finden. Das abgelegte Wissen ist bei beiden Firmen nur in Form von Dokumenten vorhanden.

Die USU AG geht mit dem Ansatz der Topic Maps einen anderen Weg. Dieser Ansatz ermöglicht eine zentrale Suche in verschiedenen Datenquellen. Es lassen sich beliebige Datenquellen (Dokumente, Datenbanken, Internet, etc.) einbinden und klassifizieren. Das Klassifizieren erfolgt aber auf redaktionelle Weise, d.h. die Pflege der Themennetze erfolgt nicht automatisch, sondern durch einen „Redakteur“. Dieser stellt die verschiedenen Suchanfragen und Themen zusammen. Da der Redakteur allerdings nur Suchanfragen abbildet, hält sich der Pflegeaufwand in Grenzen. Der Vorteil dieses Ansatzes liegt darin, dass das System bestehende Datenquellen verwenden kann. Der Redakteur muss sich allerdings mit den verschiedenen Suchmöglichkeiten der einzelnen Datenquellen auskennen, da er hier sehr spezifische Suchen zusammenstellen muss.

Im intelligent views System wird das Wissen in einem eigenen Netz abgebildet. Hier lässt sich jede Art von Wissen ablegen. Dies erfordert zunächst einen großen Aufwand an redaktioneller Arbeit, denn das Netz muss zunächst mit all seinen Begriffen angelegt und vorhandenes Wissen den Begriffen zugeordnet werden. Neues Wissen muss von einem Redakteur eingepflegt werden. Wobei dieser, je nach gewünschtem Grad der Genauigkeit, von der Software unterstützt wird. Dieser Ansatz hat den Vorteil, dass die Informationen im Netz auch „wirkliche“ Informationen sind.

Generell lässt sich nicht sagen, welches System das „bessere“ ist. Der Einsatz eines bestimmten Systems hängt vom Anwendungsgebiet ab. Werden täglich viele neue Dokumente eingepflegt, so bietet sich ein System mit neuronalen Netzen an, da diese eine sehr gute automatische Indexierung haben. Die Automatisierung wird allerdings durch eine höhere Ungenauigkeit erkaufte.

Wird hingegen eine hohe Genauigkeit des Systems erwartet, so bieten sich redaktionell gepflegte Systeme an.

Für den Einsatz im Bauwesen als gewerkeübergreifende Projektverwaltung wäre eine Mischlösung am besten geeignet. Zum einen sollten neue Dokumente bzw. Informationen von jedem Benutzer eingepflegt werden können. Das Einpflegen sollte daher möglichst automatisiert erfolgen. Da ein großer Teil der Dokumente in einem Bauprojekt aus Plänen

und Zeichnungen besteht, die nicht im Textformat vorliegen, sollte das System auch diese nach Möglichkeit verstehen und einordnen können. Hierfür bietet sich der Einsatz der neuronalen Netze an, da diese unscharfe, d.h. nicht nach einer festen Norm erstellte Dokumente besser verarbeiten können.

Zum anderen sollte das Netz dem Anwender eine große Transparenz bieten. Da der Benutzer oft weiß, was er sucht, sollte eine Navigation durch das Netz möglich sein. Sucht ein Anwender z.B. alle Dokumente für ein bestimmtes Bauteil, so muss er direkt vom Bauteil zu den Dokumenten kommen. Hierfür bietet sich der Einsatz von semantischen Netzen bzw. der Topic Maps an, da diese bessere Visualisierungsmöglichkeiten haben.

Auch sollte ein System administrierbar sein. So sollte z.B. der Projektleiter falsch eingeordnete Informationen erkennen und korrigieren können. Dies lässt sich in einem neuronalen Netz nur durch neues Trainieren bewerkstelligen, in semantischen Netzen bzw. Topic Maps hingegen kann dies einfacher durch Löschen oder Ziehen neuer Relationen erreicht werden.

4 Konzeption eines Grundnetzes für Bauvorhaben

4.1 Definition

Zur Darstellung des Netzes wird eine Darstellungsform gewählt, die an UML angelehnt ist, aber um die wissensnetzspezifischen Elemente erweitert wurde. Grundsätzlich gilt, dass abstrakte Begriffe oder Relationen in *kursiver* Schrift erscheinen.

Begriffe

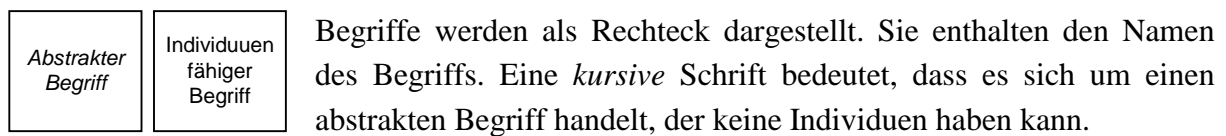


Abbildung 21: Definition - Begriff

Rollen

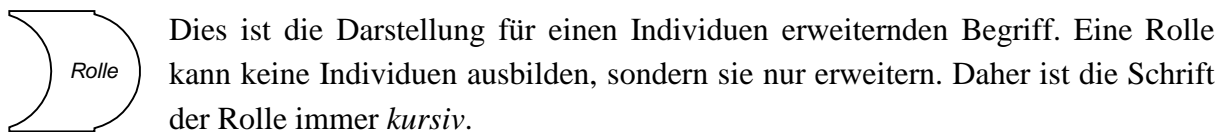


Abbildung 22: Definition - Rolle

Individuen

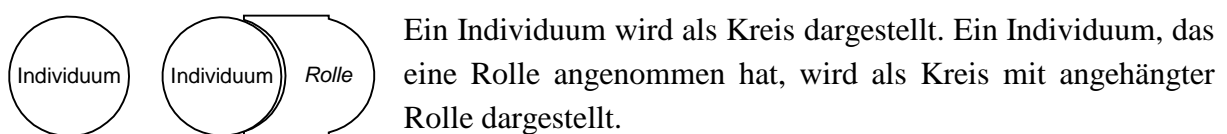


Abbildung 23: Definition - Individuen

Relationen

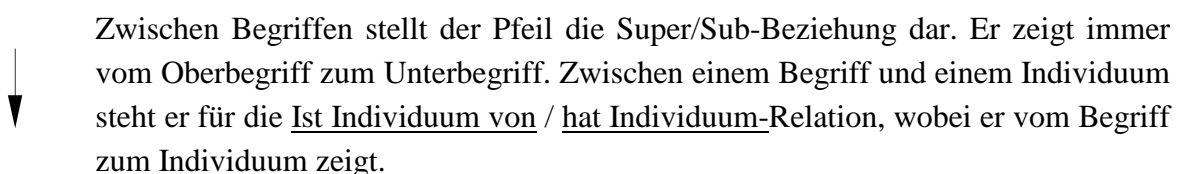
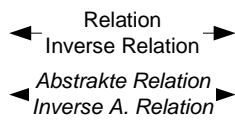


Abbildung 24: Definition - Super/Sub Relation



Relationen werden als Doppelpfeil dargestellt. Auch hier gilt, dass abstrakte Relationen kursiv dargestellt werden. Eine Relation hat immer ein Namenspaar für die Relation und die inverse Relation.

Abbildung 25: Definition - Relationen



Ist auch Relation für die Beziehung zwischen Begriff und Rolle. Dies zeigt an, dass Individuen des Begriffs die Rolle erweitern können.

Abbildung 26: Definition - Ist auch Relation

Erweiternde Erläuterungen zu den einzelnen Begriffen und Relation erfolgen in tabellarischer Form. So werden zu den Begriffen die jeweiligen Attribute, Synonyme, Relationen und Erweiterungen angegeben. In diesem Kontext werden **Konzepte** fett dargestellt und Relationen unterstrichen. Abstrakte Begriffe und Relationen werden *kursiv* darstellt.

Die dargestellten Relationen (bis auf die Systemrelationen) gelten immer zwischen den Individuen der Begriffe. Ebenso sind die Attribute für die Individuen der Begriffe definiert. Bei Ausnahmen wird dies erwähnt.

4.2 Grundprinzipien bei der Modellierung eines semantischen Netzes

Je nach Anwendungsfall gibt es verschiedene Modellierungsarten eines Wissensnetzes. Es haben sich jedoch verschieden Grundprinzipien der Modellierung herausgestellt, die beim Aufbau eines neuen Netzes beachtet werden sollten. Diese gewährleisten, dass das Netz bei Änderung leicht anpassbar ist und neue Objekte ohne Probleme eingefügt werden können.

Diese Grundprinzipien werden hier anhand einiger Beispiel kurz erläutert (aus Handbuch K-Infinity).

4.2.1 Die Ober-/Unterbegriffsrelation (OUR)

Die Relation, die immer zwischen zwei Begriffen, mit Ausnahme des Wurzelbegriffs T, besteht, ist die ist Individuum von/hat Individuum Relation, auch Ober-/Unterbegriffsrelation oder Super-/Subrelation genannt. Diese Relation spezialisiert einen Oberbegriff zu seinem Unterbegriff, d.h. der Unterbegriff kann immer als Teilmenge seines Oberbegriffs angesehen werden. Daher erbt der Unterbegriff auch alle Relationen und Attribute, die für den Oberbegriff definiert wurden.

Die Vererbung macht nur wirklich Sinn, wenn der Unterbegriff auch ein Spezialfall des Oberbegriffs ist.

Beispiel:

Eine Wand ist ein Bauteil. Somit ist Bauteil der Oberbegriff von Wand. Eine Wand kann eine Tür beinhalten, die aber keine Wand ist!

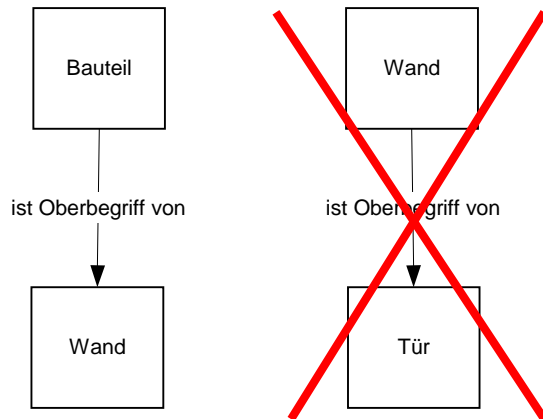


Abbildung 27: Grundprinzipien - Ober-/Unterbegriff

4.2.2 Anzahl der Unterbegriffe

Zu viele Unterbegriffe zu einem Oberbegriff lassen das Netz schnell unübersichtlich und verwirrend werden. Zudem haben alle Unterbegriffe den gleichen Abstand zum Oberbegriff, und sind somit bei der Suche gleichgewichtig. Bei zu wenigen Unterbegriffen macht die Hierarchie natürlich auch keinen Sinn. Als sinnvolle Anzahl an Unterbegriffen hat sich die Zahl drei erwiesen, d.h. ein Begriff sollte mindestens drei Unterbegriffe haben. Dies kann und wird natürlich nicht in jedem Netz eingehalten, sollte aber als Anhaltspunkt gelten.

4.2.3 Hierarchietiefe

Weit oben in der Hierarchie sollten sehr allgemeingültige Begriffe angesiedelt sein, die eher einen Gliederungscharakter haben. Je tiefer ein Begriff in der Hierarchie angesiedelt ist, umso aussagekräftiger und fachspezifischer sollte er sein.

4.2.4 Gruppierung von Begriffen mit gleichem Merkmal

Wenn ein Begriff sehr viele Unterbegriffe aufweist, sollte untersucht werden, ob diese nicht weiter nach einem bestimmten Merkmal gruppiert werden können, um so eine größere Nähe der ähnlichen Objekte untereinander herzustellen.

Im folgendem Beispiel haben alle Bauvorgänge die gleiche Nähe zueinander:

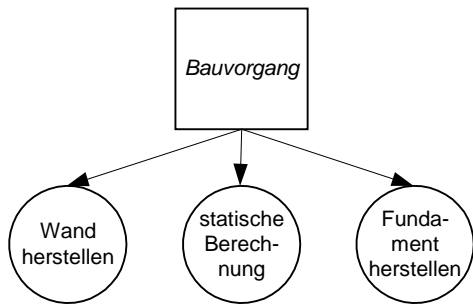


Abbildung 28: Grundprinzipien - Gruppierung von Objekten (1)

Dies lässt sich durch eine weitere Untergruppierung vermeiden:

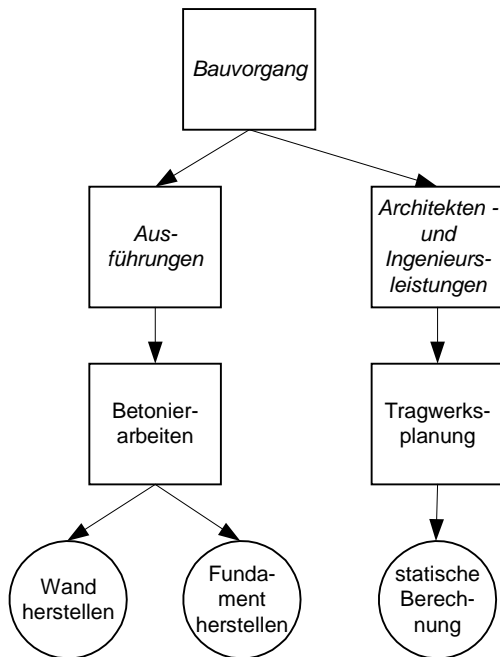


Abbildung 29: Grundprinzipien - Gruppierung von Objekten (2)

Mit dieser Gruppierung liegen die Betonierarbeiten näher beieinander und weisen somit auch eine größere Ähnlichkeit auf.

4.2.5 Gleicher Abstraktionsgrad aller Unterbegriffe

Die Unterbegriffe eines Oberbegriffs sollten alle in etwa den gleichen Abstraktionsgrad besitzen.

Beispiel:

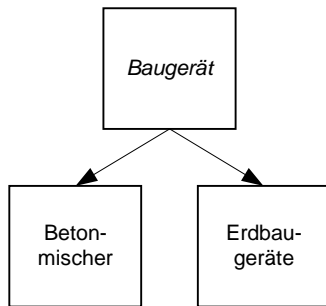


Abbildung 30: Grundprinzipien - Abstraktionsgrad der Begriffe (1)

Erdbaugeräte ist hier als eine Art Oberbegriff zu sehen. So könnten unter Erdbaugeräte noch die Unterbegriffe Bagger, Rammen etc. angehängt sein. Betonmischer ist aber nicht als Begriffsmenge anzusehen, daher fehlt ihm ein konkreter Oberbegriff, der Betonmischer ebenfalls klassifiziert.

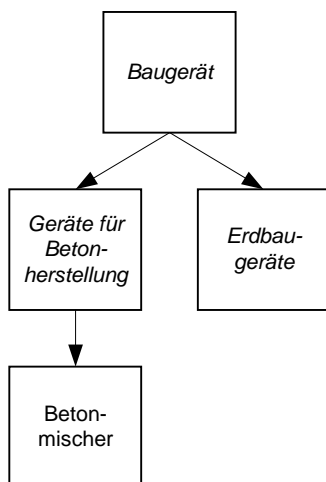


Abbildung 31: Grundprinzipien - Abstraktionsgrad der Begriffe (2)

Durch das Einfügen eines abstrakten Begriffs „Geräte für Betonherstellung“, haben die Unterbegriffe von Baugerät wieder einen ähnlichen Abstraktionsgrad.

4.2.6 Ein Wissensnetz ist kein Baum

In einem Wissensnetz gibt es keine Baumstruktur, d.h. ein Begriff kann mehrere Oberbegriffe haben. Man muss sich nicht entscheiden, ob ein Begriff nun dem einen oder einem anderen Oberbegriff zugeordnet werden kann. Er kann einfach beiden zugeordnet werden.

Beispiel:

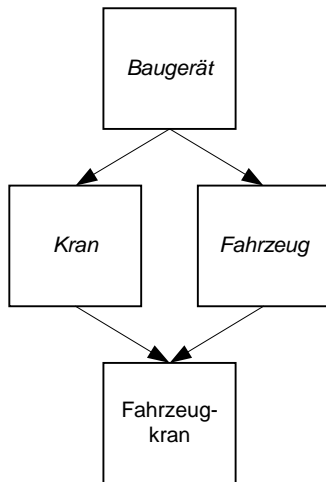


Abbildung 32: Grundprinzipien - Mehrfachvererbung

Somit erbt Fahrzeugkran alle Attribute und Relationen sowohl von Fahrzeug, als auch von Kran.

Auch ist es möglich, dass Unterbegriffe nicht klar voneinander abgegrenzt sind. Das bedeutet, Unterbegriffe können durchaus über eine gemeinsame Schnittmenge verfügen.

4.2.7 Relationen

Generell ist zu sagen, dass es nicht zu viele spezialisierte Relationen geben sollte, die nur auf wenige Begriffe zutreffen. Im Gegenzug sollten Relationen auch nicht zu generell definiert sein, da dies die Suche nach speziellen Begriffen erschwert.

Von Vorteil ist eine abstrakte Relation, die eine grundsätzliche Beziehung zwischen zwei Objekten angibt. Von dieser Relation abgeleitete Relationen können dann die Beziehung zwischen zwei Objekten genauer spezifizieren.

Hier gilt als Anhaltspunkt, dass in Abhängigkeit von der Netzgröße, ca. 10-20 benutzerdefinierte Relationen neben den Systemrelationen bestehen sollten.

Im Unterschied zu den Begriffen sollten Relationen jedoch klar voneinander abgegrenzt sein. Sich überschneidende Relationen wirken sich sowohl negativ auf die Suchmöglichkeiten als auch auf eine konsistente Modellierung des Netzes aus.

4.2.8 Anwendung von Erweiterungen (Rollen)

Erweiterungen verhalten sich ähnlich der Bildung von Individuen eines Begriffs, d.h. das Attribut- und Relationsschema der Rolle geht auf das Individuum über. Erweiterungen

können sehr gut dazu genutzt werden, Begriffen eine temporäre Funktion zuzuweisen, die eventuell wieder weggenommen werden kann. So kann z.B. die Umsetzung verschiedener Dokumentversionen als Rolle erfolgen.

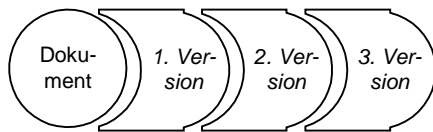


Abbildung 33: Grundprinzipien – Rollenerweiterung (1)

Auch Berufe können z.B. als Rolle realisiert werden.

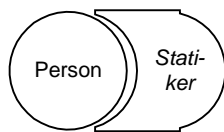


Abbildung 34: Grundprinzipien - Rollenerweiterung (2)

4.2.9 Auswahl der Individuenfähigen Begriffe

In der Regel sollte ein individuenfähiger Begriff eine notwendige Eigenschaft des Individuums sein. Ist das Individuum einmal nicht mehr Instanz dieses Begriffes, so ist es nicht mehr existent.

Beispiel:

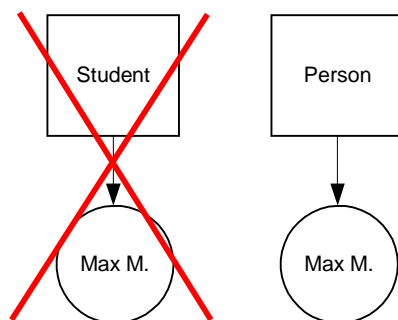


Abbildung 35: Grundprinzipien - Individuenfähige Begriffe

Nur in sehr spezialisierten Netzen kann es vorkommen, dass es Modellierungen gibt, die eben nur in diesem Netz gültig sind.

4.2.10 Erkennen abhängiger Begriffe

In einem Netz kann es vorkommen, dass bestimmte Begriffe nur mit einem anderen Begriff zusammen existieren können. So kann z.B. ein Kind nur existieren, wenn es zwangsläufig

auch eine Mutter hat, d.h. ein Begriff „Kind“ muss eine „ist Kind von“ Relation zu einem Begriff „Mutter“ haben. Dies kann erreicht werden, indem im Schema für die Relationen eines „Kind“-Begriffes angegeben wird, dass alle Individuen eine obligatorische Relation „ist Kind von“ zu einem Begriff „Mutter“ haben müssen.

4.3 Anforderungen an das Baunetz

- Alle Arten von Bauprojekten sollen mit dem Wissensnetz verwaltet werden können. Ebenso sollte es möglich sein, das Netz an neue Aufgabenstellungen anzupassen
- Bauobjekte sollen beliebig fein Aufgeteilt werden können. Es soll dem Anwender überlassen werden, was er schließlich als kleinste Baueinheit wählt, z.B. können zur Darstellung eines Gebäudes als kleinste Einheiten die einzelnen Wände, oder gröber aufgeteilt, die einzelnen Räume gewählt werden.
- Alle Objekte sollten in unbegrenzter Zahl vorkommen dürfen. Ebenso sollte das System jederzeit durch beliebige Objekte erweitert werden können.
- Die verschiedenen Bauprozesse und Dokumente sollten versionierbar sein.
- Nur die Projektbeteiligten sollten über einen eigenen Login und ein Passwort Zugriff auf das System haben.

4.4 Grundkonzept

Die meisten Bauvorhaben werden in der Unikatbauweise realisiert, d.h. die einzelnen Bauprojekte unterscheiden sich in der Planung, in der Ausführung und der Zusammensetzung der Projektbeteiligten. Daher muss ein Wissensnetz für Bauprojekte breit gefächert und leicht erweiterbar sein. Hierzu müssen zuerst alle elementaren Bestandteile eines Bauprojektes analysiert und entsprechend im Wissensnetz abgelegt werden.

Ein Bauprojekt kann zunächst grob eingeteilt werden in:

- **Bauvorgang oder -prozess.** Hierunter fallen alle Tätigkeiten, die während eines Bauprojektes anfallen, z.B. eine Genehmigung durch eine Behörde, die statische Planung eines Bauteils oder die Ausführung selbst. Da ein Bauvorgang immer einen zeitlichen Rahmen hat, ist er durch seinen Start- bzw. Endtermin definiert. Des Weiteren verursacht ein Bauvorgang Kosten. Auch kann sich ein Bauvorgang im Laufe der Zeit verändern, d.h. es kann z.B. Termin- oder Kostenänderungen geben. Daher sollte es möglich sein, verschiedene Versionen eines Bauvorgangs anzulegen.
- **Bauobjekte** sind die Bestandteile eines Bauvorhabens. Diese können beliebig fein untergliedert sein. So kann ein Bauobjekt wiederum aus mehreren Bauobjekten bestehen, z. B. ein Haus aus Etagen, diese aus Räumen, die wiederum aus Wänden bestehen. Es ist

sogar vorstellbar, einzelne Bauobjekte so fein zu untergliedern, dass die einzelnen Baustoffe (Stahlträger, Beton usw.) mitangegeben werden.

- **Projektbeteiligte.** Alle an einem Bauprojekt beteiligten, handlungsfähigen Personen oder Organisationen (Aktoren) werden als Projektbeteiligte bezeichnet. Zum Projektbeteiligten wird automatisch ein Akteur, indem er einen Bauvorgang eines Projektes ausführt, egal ob dies nun eine Genehmigung, eine Planung oder eine Ausführung ist.
- **Dokumente.** Dokumente können in unterschiedlichster Art vorkommen. Sei es als CAD-Pläne, als Word-Dokumente oder spezielle Austauschformate bestimmter Anwendungen. Ein Dokument bezieht sich immer auf einen bzw. mehrere Bauvorgänge. Im Laufe der Zeit kann sich ein Dokument ebenfalls ändern, daher sollte auch hier eine Versionierung möglich sein.
- Das **Bauprojekt** dient als Sammelstelle für alle Projektbestandteile und bildet somit die Wurzel eines Bauvorhabens.
- **Baunormen.** Hierunter fallen alle Normen, die für bestimmte Bauvorgänge gelten. So können z.B. die Planungsprozesse den einzelnen Leistungsphasen der HOAI zugeordnet werden.

Diese Grundbegriffe werden in den folgenden Kapiteln näher erläutert. Zunächst wird jedoch ein allgemeingültiges Grundnetz erstellt, das es ermöglicht, beliebige Objekte der realen Welt im System zu erstellen, damit eine Erweiterbarkeit des Systems gewährleistet ist.

4.5 Das allgemeingültige Grundnetz

Um das Baunetz leicht erweiterbar und offen zu halten, wird ein Grundnetz angelegt, das es ermöglicht, beliebige Begriffe anzulegen und einzuordnen. Das Grundnetz besteht aus einer Menge von abstrakten Konzepten, die es ermöglichen, Objekte der realen Welt im Netz abzubilden und den entsprechenden Oberbegriffen zuzuordnen. Hierfür wird ein Wurzelkonzept angelegt, von dem alle anderen Konzepte abgeleitet werden müssen. Im intelligent views K-Infinity-System hat sich hierfür der Buchstabe **T** (aus der Verbandstheorie für Top) eingebürgert. **T** ist abstrakt und hat folgende System-Attribute:

- **Name.** Der Name bezeichnet jeweils den Namen eines Konzeptes, einer Rolle, oder einer Instanz.
- **EID.** Die ID wird vom System vergeben und ist eine eindeutige Bezeichnung für das jeweilige Konzept, die Rolle oder Instanz. Die EID ist im kompletten Wissensnetz eindeutig.

Diese Attribute vererben sich auf alle Unterobjekte von **T**. Da im K-Infinity System alle Objekte von einem Oberobjekt abgeleitet werden müssen, haben folglich alle Objekte diese Attribute. **T** erhält ebenfalls ein benutzerdefiniertes Attribut:

- **Synonym.** Dieses Attribut wird für den Begriff, nicht für Individuen definiert. Das Attribut kann mehrfach vorkommen und dient dazu, jedem Begriff eine Reihe von Synonymen zuzuordnen. Hierfür wird beim Attribut angegeben, dass es als Synonym für den Begriff dient.

T besitzt folgende System-Relationen:

- Relates to ist die abstrakte Grundrelation. Von ihr werden alle Relationen, auch System-Relationen, abgeleitet.
- Ist Individuum von / hat Individuum sind die Relationen zwischen einem Begriff und einem Individuum. Z.B. Max Mustermann (Individuum) *ist Individuum von* **Person** (Begriff).
- Ist auch / erweitert. Dies ist die Beziehung zu einer Rolle. Z.B. Max Mustermann (Individuum) *ist auch* **Bauingenieur** (Rolle).

T erhält ebenfalls die benutzerdefinierten Relationen

- Hat Beziehung mit ist eine symmetrische Relation, die eine einfache Beziehung zwischen zwei Objekten modellieren soll. Diese ist identisch mit der Systemrelation relates to, die allerdings nicht direkt verwendet werden kann.
 - Besteht aus / ist Bestandteil von. Hiermit kann eine Aggregation realisiert werden. Diese Relation wird abstrakt ausgebildet und kann zwischen Unterbegriffen von *T* entsprechend verfeinert werden. Z.B. Bauprojekt (Individuum) *hat Objektbestandteil* Gewerk (Individuum).
- Hat Dokument ist ebenfalls abstrakt und dient dazu, jedem beliebigen Individuum ein Dokument zuzuordnen. Diese *hat Dokument* Beziehung muss dann ebenfalls zwischen den einzelnen Begriffen näher spezifiziert werden.

Von *T* werden jetzt alle Unterbegriffe abgeleitet. Diese werden möglichst abstrakt und allgemein gewählt.

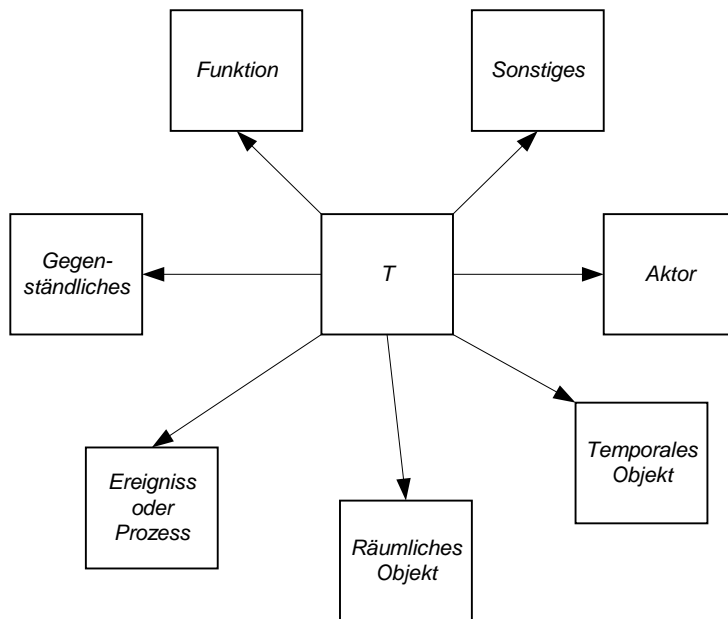


Abbildung 36: Grundnetz - T

Darin bedeuten die einzelnen Begriffe:

- **Aktoren** sind Personen oder Personengruppen (z.B. eine Organisation), die eine Handlung ausüben können.
- **Funktion.** Damit werden z.B. Rollen bezeichnet, die ein Aktor annehmen kann.
- **Ereignis oder Prozess.** Ein Ablauf, eine Entwicklung oder ein Vorgang.
- **Räumliches Objekte** sind Objekte, die über Lageinformationen verfügen. Dies kann ein Raum oder auch ein geographisches Objekt wie eine Stadt sein.
- **Gegenständliches** können Objekte der realen Welt (Gegenstand) oder auch abstrakte Gegenstandsmengen (z.B. Normen) sein.
- **Temporales Objekte** sind Zeitbeschreibungen wie Zeitpunkt, Periode oder Zeitintervall.
- **Sonstiges** wird für alle Konzepte verwendet, die sich nicht in eines der anderen Konzepte einordnen lassen.

Diese Begriffe dienen als Grundkonzept für alle weiteren Begriffe. Sie sind alle abstrakt gehalten und werden durch weitere Unterbegriffe entsprechend spezialisiert. Da sehr viele Unterbegriffe existieren, werden in der Hierarchie der einzelnen Baugrundnetz-Begriffe nur die relevanten angezeigt.

4.6 Das Baugrundnetz

Das Baugrundnetz erweitert das allgemeingültige Grundnetz durch spezielle Begriffe für Bauvorhaben. Diese werden in den folgenden Kapiteln näher erläutert.

4.6.1 Bauprojekt

Das Bauprojekt dient als Sammelstelle für ein Bauvorhaben und ist wiederum in mehrere Bereiche untergliedert. Der Begriff Bauprojekt selbst ist abstrakt, Individuen können nur von den Unterobjekten abgeleitet werden.

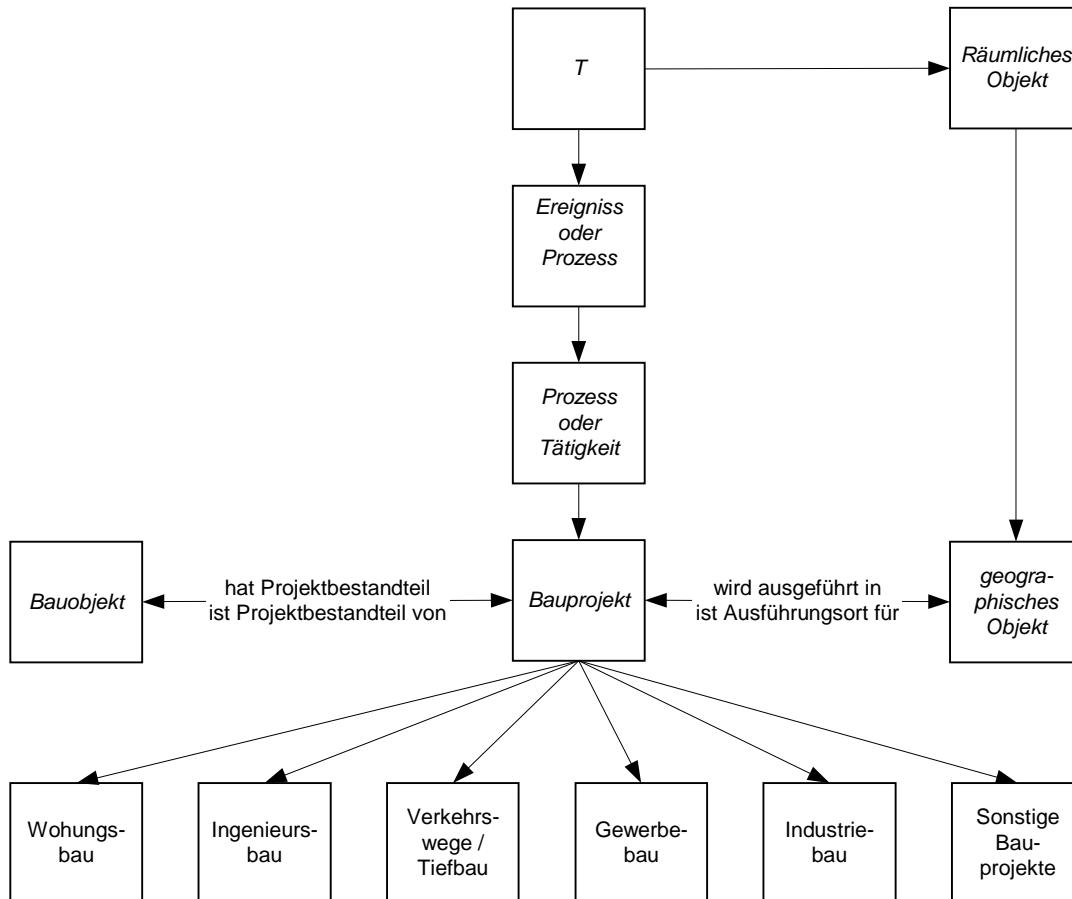


Abbildung 37: Baugrundnetz - Bauprojekt

Attribute

- **Beschreibung** ist eine ausführliche textliche Beschreibung des Bauprojekts, d.h. hier kann eine umgangssprachliche Beschreibung des Projektes eingegeben werden.
- **Kurzbeschreibung** kann für eine kurze, stichpunktartige Beschreibung des Bauprojektes verwendet werden.

Synonyme

- Bauvorhaben
- Baustelle

Relationen

- Hat Projektbestandteil *Bauobjekt*. Diese Beziehung zeigt die zum Projekt gehörenden Bauobjekte an. Diese sollten zuerst allgemeinerer Natur sein, wie Gebäude etc. Hat Projektbestandteil wird von der Oberrelation hat Beziehung mit -> besteht aus abgeleitet.
- Wird ausgeführt in *Geographisches Objekt*. Über diese Relation können dem Bauprojekt ein oder mehrere Ausführungsorte zugeordnet werden. Dies können z.B. eine Stadt, ein Land oder ein Gebiet sein.

Mögliche Rollen

- Keine

4.6.2 Bauvorgang

Als Bauvorgang werden alle Vorgänge (Handlungen) in einem Bauprojekt bezeichnet. Somit lassen sich alle Phasen (Planung, Genehmigungen, Ausführungen, etc.) eines Bauprojektes abbilden.

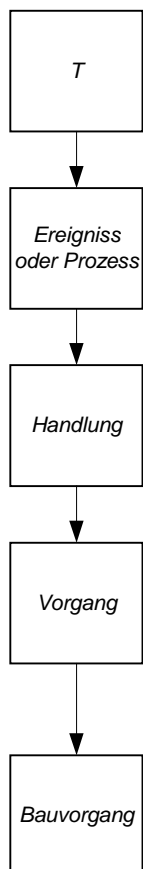


Abbildung 38: Baugrundnetz - Bauvorgang

Attribute

- **Beginn.** Dies ist der Startzeitpunkt für den Bauvorgang
- **Ende** bezeichnet den Endzeitpunkt für den Bauvorgang
- **Kosten.** Die Kosten des Bauvorgangs

Synonyme

- Baurealisierung
- Bauprozess
- Bauverlauf
- Baudurchführung
- Bauablauf

Relationen

- Wird ausgeführt an **Bauobjekt**. Mit dieser Relation wird eine Beziehung zu einem Bauobjekt hergestellt, die angibt, an welchem Bauobjekt der Vorgang ausgeführt wird.
- Wird ausgeführt von **Projektbeteiligter** ist eine abstrakte Relation, die angibt, von welchem Projektbeteiligtem der Vorgang ausgeführt wird. Diese Relation selbst kann nicht verwendet werden, sondern wird durch die Unterkonzepte von Bauvorgang realisiert. Die Unterrelationen sind bei **Projektbeteiligter** näher spezifiziert.
- Bestimmt Zeitpunkt für / Ist Zeitlich abhängig von **Bauvorgang**. Hiermit wird eine zeitliche Abhängigkeit der Bauvorgänge untereinander dargestellt. Damit können z.B. alle von einer Terminverschiebung betroffenen Bauvorgänge ermittelt werden. Die Relation wird von hat Beziehung mit abgeleitet.

Mögliche Rollen

- **Bauvorgangversion**. Über diese Erweiterung wird eine Versionierung gewährleistet. S. *Version*

Das Konzept Bauvorgang ist abstrakt und wird spezialisiert durch die Unterkonzepte, die wiederum spezialisiert sein können:

- **Genehmigungen**. Hierunter fallen alle Behördengänge und gesetzliche Genehmigungen.
- **Architekten- und Ingenieursleistungen** sind alle planerischen Prozesse.
- **Ausführungen** sind alle Vorgänge auf der Baustelle, d.h. die konkreten Umsetzungen der Planungsprozesse.
- **Bauherrenaufgaben**. Hierunter fallen die Aufgaben des Bauherren wie etwa das Projektmanagement.
- **Gutachten und Beratung**.

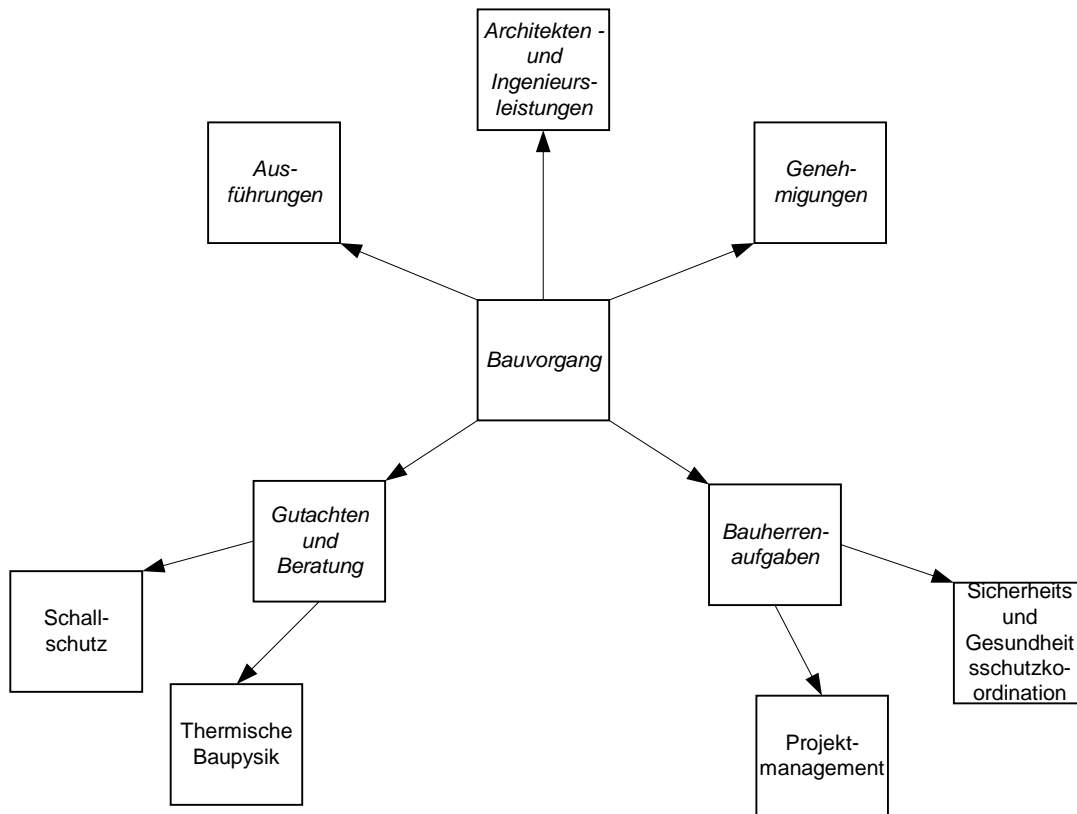


Abbildung 39: Baugrundnetz – Unterbegriffe von Bauvorgang (1)

Dieser Begriffsaufbau ist natürlich nicht vollständig, kann aber jederzeit durch neue Begriffe erweitert werden. Die einzelnen Begriffe können noch beliebig durch Unterbegriffe verfeinert werden, die auch sehr projektspezifisch angelegt sein können.

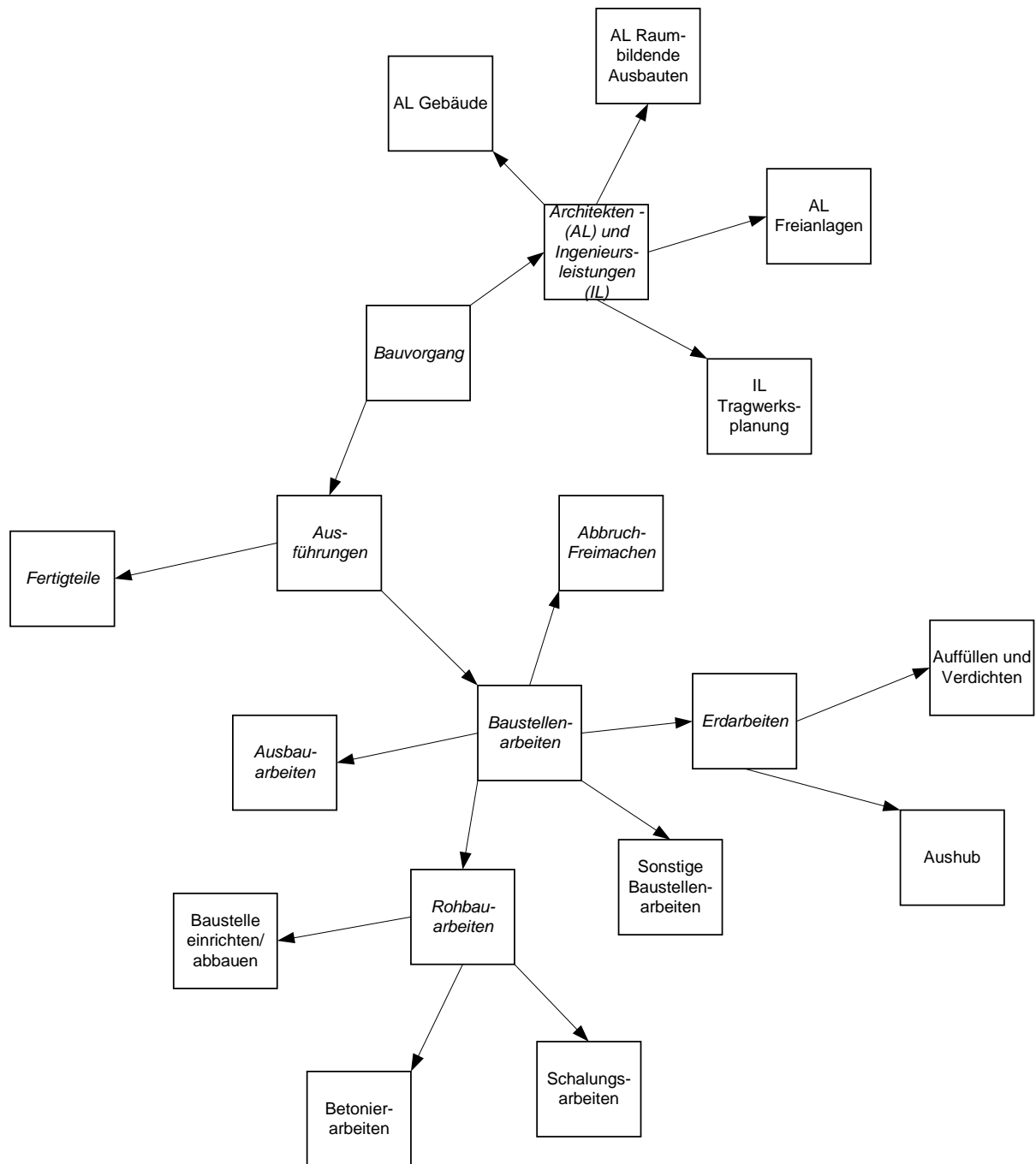


Abbildung 40: Baugrundnetz – Unterbegriffe von Bauvorgang (2)

4.6.3 Baunormen

Unter den Baunormen können alle Normen, die für ein Bauprojekt relevant sind, abgelegt werden.

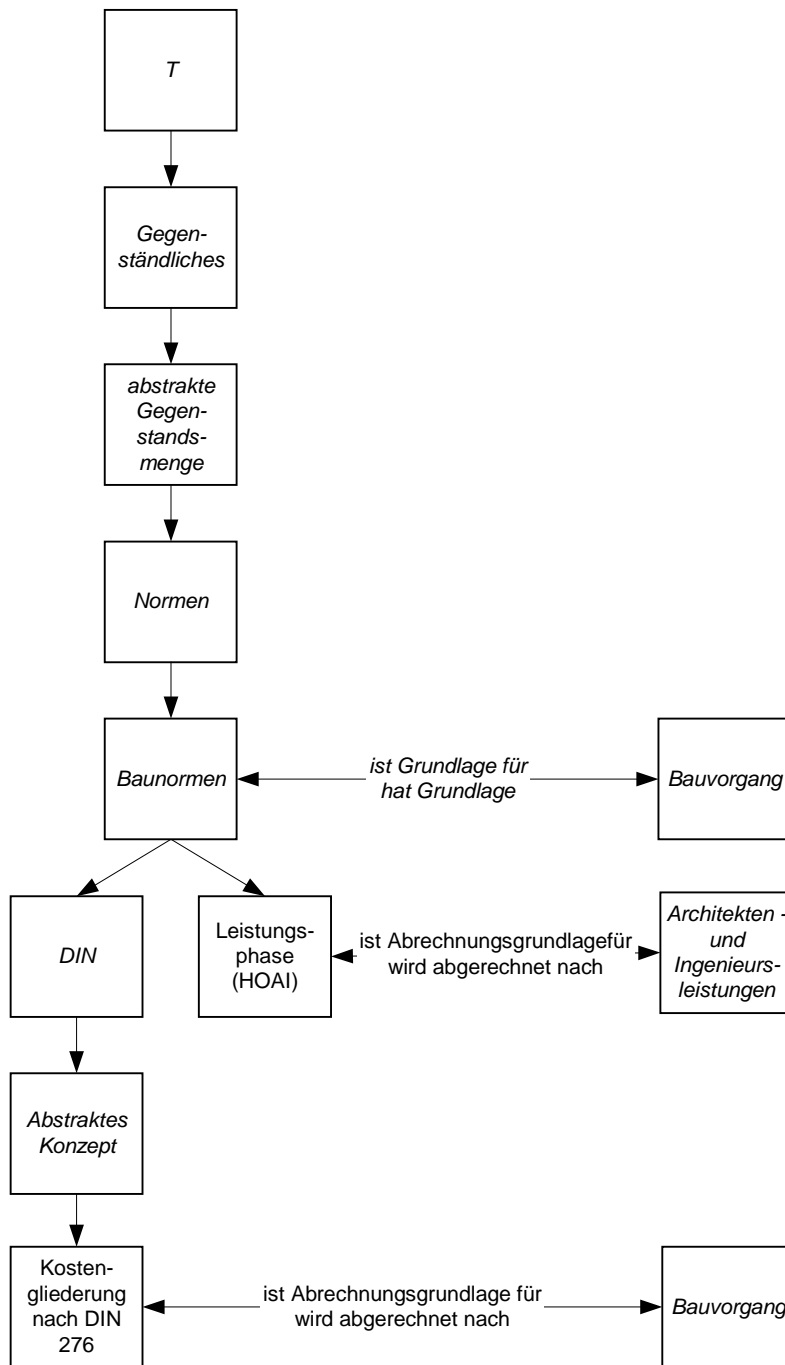


Abbildung 41: Baugrundnetz – Baunormen

Attribute

- Keine

Synonyme

- Regel
- Gesetz
- Auflage

- Vorschrift
- Pflicht
- Gesetzmäßigkeit

Relationen

- *Ist Grundlage für Bauvorgang*. Diese Relation dient als Grundlage für eine Beziehung zwischen einer Norm und einem Bauvorgang. Die Relation ist abstrakt und wird durch die Unterrelationen genauer spezifiziert.

Mögliche Rollen

- Keine

4.6.3.1 Leistungsphasen (HOAI)

Die einzelnen Leistungsphasen der HOAI werden als Individuen ausgebildet, die Individuen von *Architekten- und Ingenieursleistungen* als Abrechnungsgrundlage zugeordnet werden können.

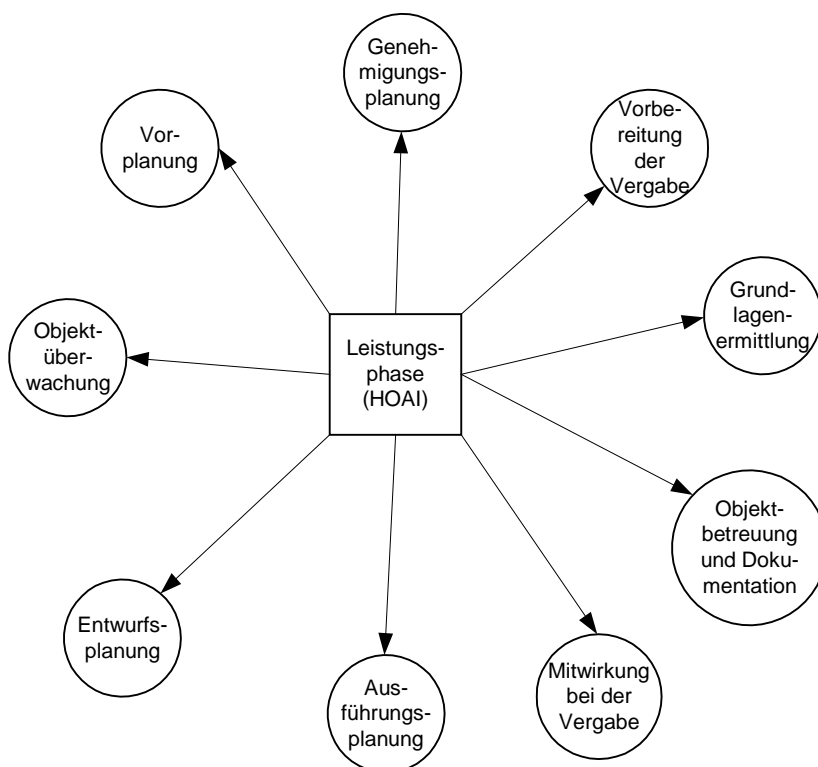


Abbildung 42: Baugrundnetz – Leistungsphasen (HOAI)

Attribute

- Gewichtung. Das Attribut Gewichtung dient zur Angabe der einzelnen Gewichtungen der Leistungsphasen bei der Berechnung..

Synonyme

- Honorarverordnung für Architekten und Ingenieure

Relationen

- Ist Abrechnungsgrundlage für *Architekten- und Ingenieursleistungen*. Die Relation wird von ist Grundlage für abgeleitet.

Mögliche Rollen

- Keine

4.6.3.2 Kostengliederung nach DIN 276

Die verschiedenen Ebenen der Kostengliederung werden als individuenfähige Begriffe angelegt. Die einzelnen Punkte werden dann als Individuen angelegt, die Individuen von *Ausführungen* als Abrechnungsgrundlage zugeordnet werden können.

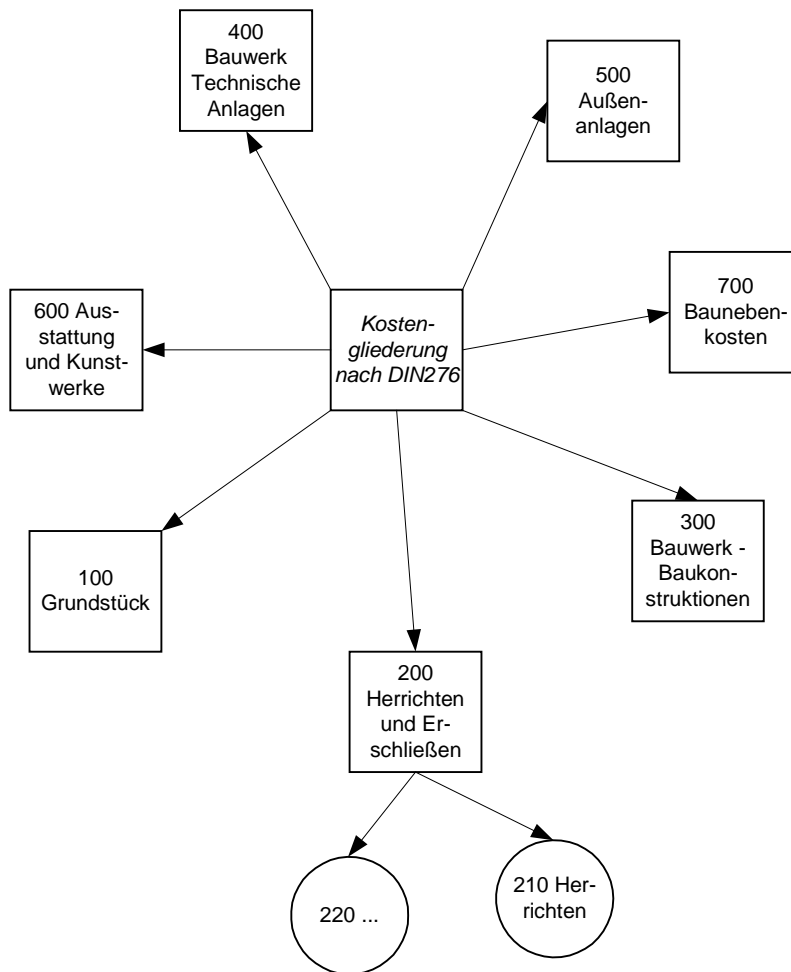


Abbildung 43: Baugrundnetz - Kostengliederung nach DIN 276

Attribute

- Keine

Synonyme

- Keine

Relationen

- Ist Abrechnung für Ausführungen

Mögliche Rollen

- Keine

4.6.4 Projektbeteiligter

Alle an einem Bauprojekt beteiligten Personen oder Firmen sind selbst im Netz vertreten. Sie werden als ein Individuum von einem der Unterbegriffe von **Aktor** angelegt (z.B. Firma, Person oder Organisation). Um Zugriff auf das System zu erhalten, muss jeder Beteiligte um die Rolle **Projektbeteiligter** erweitert werden. Ein **Aktor** muss die Rolle Projektbeteiligter besitzen, um einen Bauvorgang ausführen zu können.

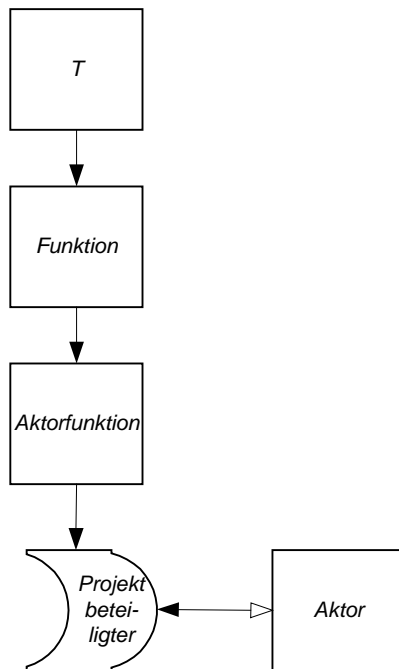


Abbildung 44: Baugrundnetz - Projektbeteiligter

Attribute

- **Login** ist der Name, mit dem sich ein Projektbeteiligter am System anmelden muss. Der Login wird vom Systemadministrator vergeben und ist frei wählbar.
- **Passwort**. Jeder Benutzer erhält ein freiwählbares Passwort zur Authentifizierung im System.

Synonyme

- Projektteilnehmer

Relationen

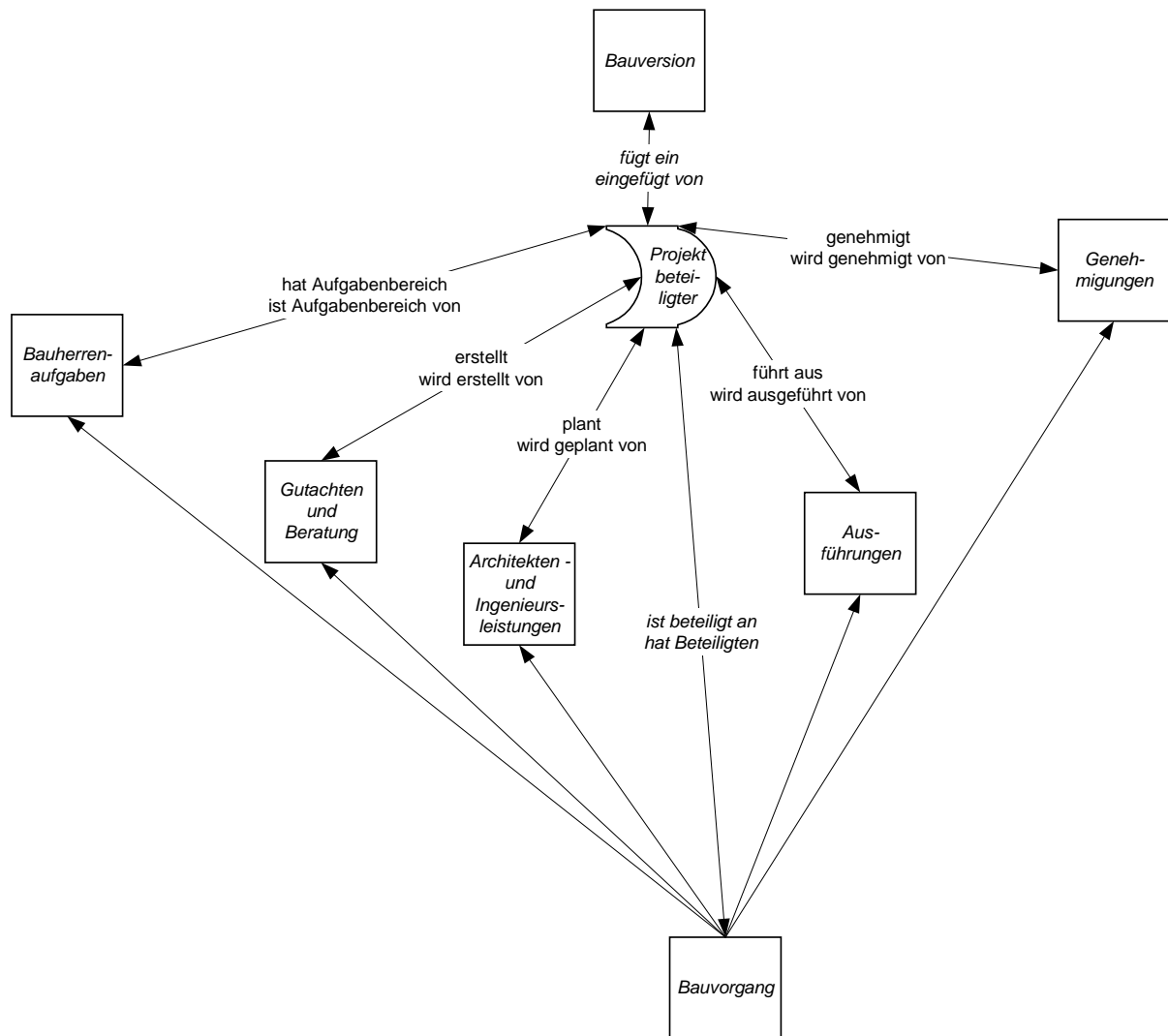


Abbildung 45: Baugrundnetz - Projektbeteiligter (Relationen)

- ***Ist beteiligt an Bauvorgang.*** Hiermit wird eine Beziehung eines Projektbeteiligten mit einem Bauvorgang hergestellt. Da die Relation abstrakt ist, kann sie nicht direkt hergestellt werden. Die jeweilige Beziehung wird durch die Unterrelationen genauer definiert. Über diese Unterrelationen lässt sich z.B. auch eine spezialisierte Suche nach einem bestimmten Projektbeteiligten realisieren. Um alle Projektbeteiligten, die eine planende Tätigkeit ausführen (Architekten bzw. Ingenieure) zu finden, muss nur nach der Relation plant gesucht werden.
 - Erstellt ***Gutachten und Beratung***
 - Führt aus ***Ausführungen***
 - Genehmigt ***Genehmigungen***
 - Hat Aufgabenbereich ***Bauherrenaufgaben***
 - Plant ***Architekten- und Ingenieursleistungen***

- Fügt ein Bauversion. Fügt ein Projektbeteiligter eine neue Bauversion ein, wird automatisch diese Relation gezogen, damit jederzeit feststellbar ist, welche Version von wem angelegt wurde.

Mögliche Rollen

- Keine

4.6.5 Bauobjekt

Ein Bauobjekt ist ein abstrakter Oberbegriff für alle in einem Bauprojekt zu erstellenden Objekte.

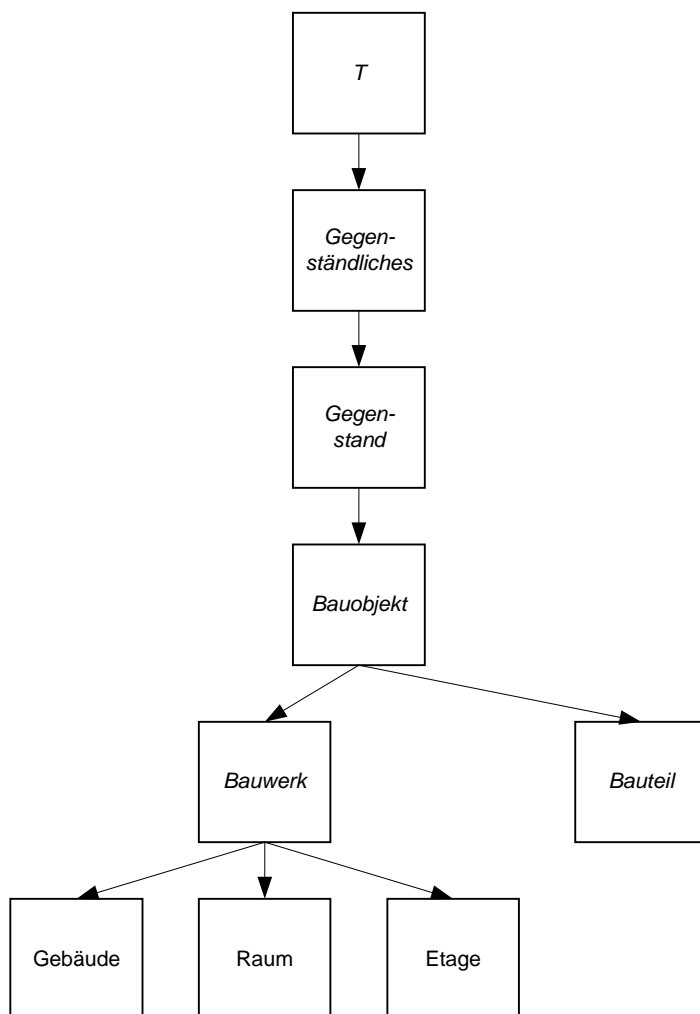


Abbildung 46: Baugrundnetz - Bauobjekt

Attribute

- **Beschreibung**

- **Kurzbeschreibung**

Synonyme

- Gebilde
- Konstrukt

Relationen

- hat Objektbeziehung mit **Bauobjekt**. Diese Relation dient zum Erstellen einer Beziehung zwischen zwei Bauobjekten. So können z.B. zwei nebeneinander liegende Räume in Beziehung miteinander gesetzt werden.
- hat Objektbestandteil / ist Objektbestandteil von **Bauobjekt**. Besteht ein Bauobjekt aus weiteren Bauobjekten, kann diese Relation verwendet werden.
- ist Projektbestandteil von **Bauprojekt**. Diese Relation sollten nur Bauobjekte auf höchster Ebene eingehen. Dies kann z.B. ein Gewerk sein.
- wird ausgeführt **Bauvorgang**. Hiermit werden alle am Bauobjekt ausgeführten Bauvorgänge angezeigt.

Mögliche Rollen

- Keine

4.6.5.1 Bauwerk

Bauwerke dienen zur Gruppierung weiterer Bauteile. So kann z.B. beim Hausbau das Gebäude zunächst in Etagen, diese dann in Räume und diese schließlich in einzelne Wände, Türen usw. zerlegt werden.

4.6.5.2 Bauteile

Die Bauteile sollten die letzten Elemente in der Bauobjektstruktur sein, d.h. sie sollten im Allgemeinen nicht weiter aufgeteilt werden. Dies ist aber nicht unbedingt zwingend. So kann z.B. eine Wand, die in einem Raum als atomarer Bestandteil behandelt wird, in einem anderen noch Tür- oder Elektrikobjekte beinhalten.

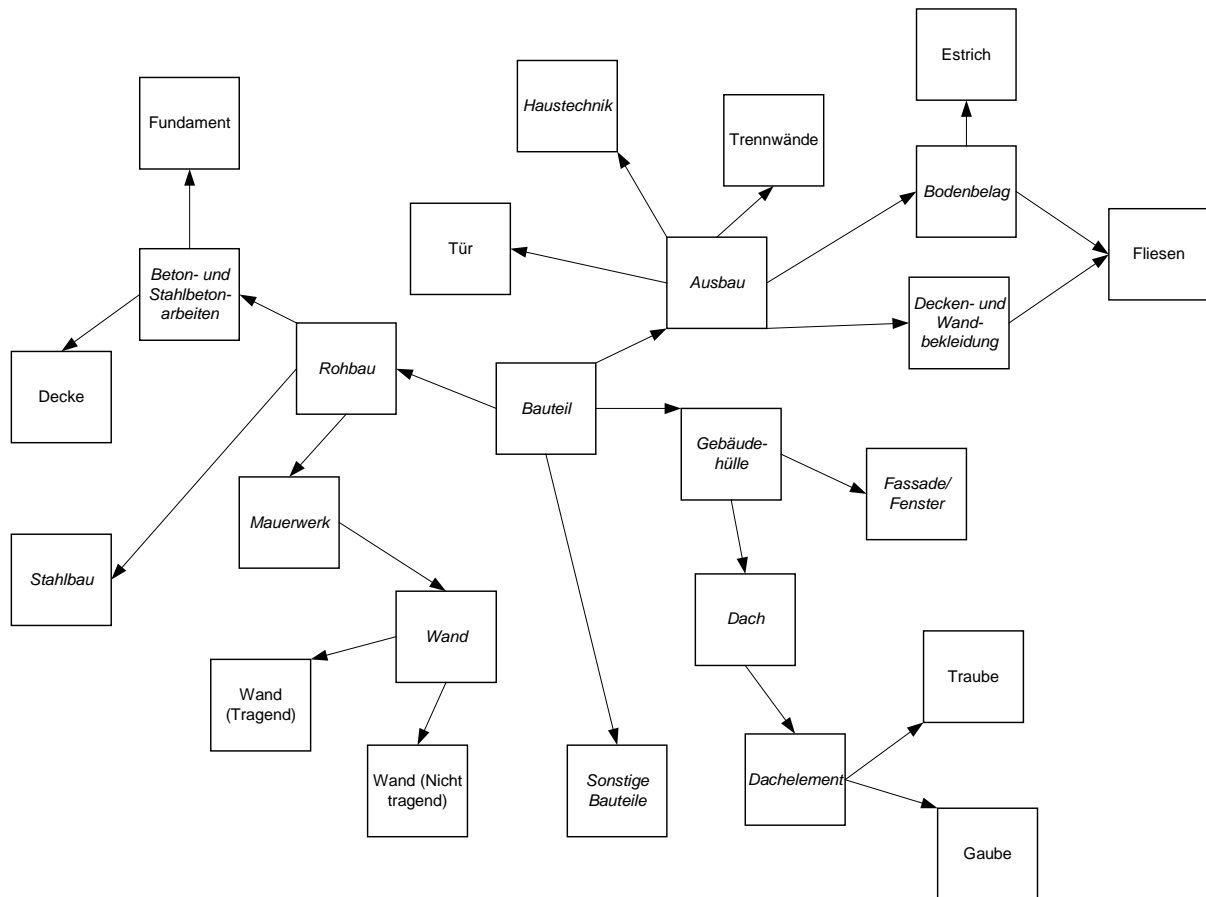


Abbildung 47: Baugrundnetz - Bauteil

4.6.6 Dokumente

Dokumente werden als eigenständiger Begriff implementiert. So ist gewährleistet, dass ein Dokument mehreren Bauvorgängen zugeordnet werden kann. Da sich im Laufe der Zeit auch Dokumente ändern können, kann über eine Rolle eine neue Dokumentversion angelegt werden. Dies hat den Vorteil, dass jederzeit auf alte Versionen zugegriffen werden kann. Der Dokumentenbegriff kann auch durch weitere Unterbegriffe verfeinert werden. So kann z.B. zu jedem Dokumenttyp (Word-Datei, CAD-Dokument) ein eigener Unterbegriff angelegt werden, der dann weitere Synonyme annehmen kann.

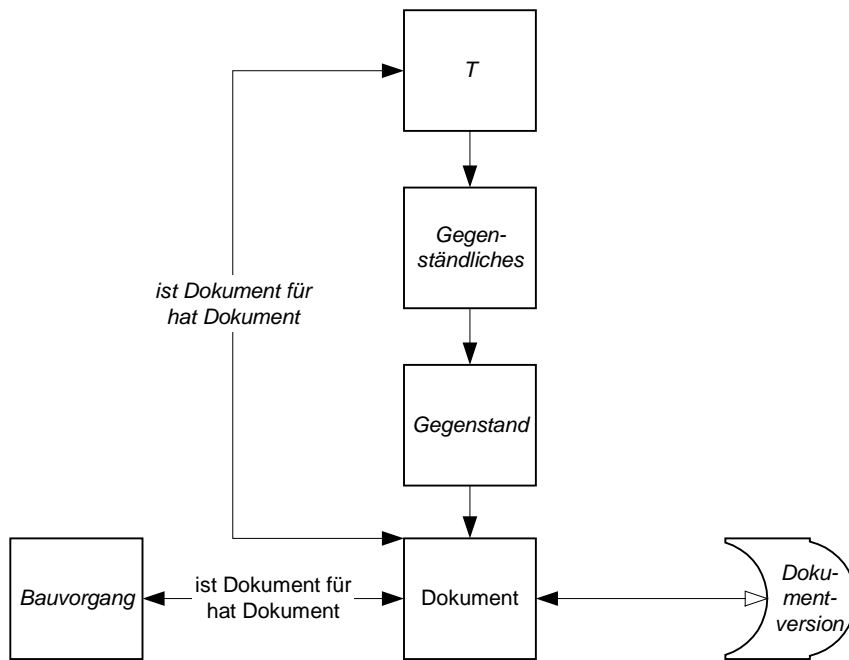


Abbildung 48: Baugrundnetz - Dokument

Attribute

- **Beschreibung.** Hier kann eine ausführliche Beschreibung in natürlicher Sprache abgelegt werden.
- **Kurzbeschreibung** dient als kurze, stichpunktartige Inhaltsangabe.

Synonyme

- Datei

Relationen

- Ist Dokument für T wird als abstrakte Relation angelegt. So ist es durch Unterrelationen später möglich, allen Begriffen ein Dokument zuzuordnen.
 - Ist Dokument für Bauvorgang ist die konkrete Umsetzung der Dokumentenrelation. Sie gewährleistet, dass nur einem Bauvorgang ein Dokument zugeordnet werden kann.

Mögliche Rollen

- **Dokumentversion.** Über die Erweiterung **Dokumentversion** wird eine Versionierung von Dokumenten erreicht.

4.6.7 Version

Sowohl Dokumente als auch Bauvorgänge können verschiedene Versionen haben. Hierzu wird der abstrakte Begriff Version angelegt. Dieser wird durch seine Unterbegriffe spezialisiert.

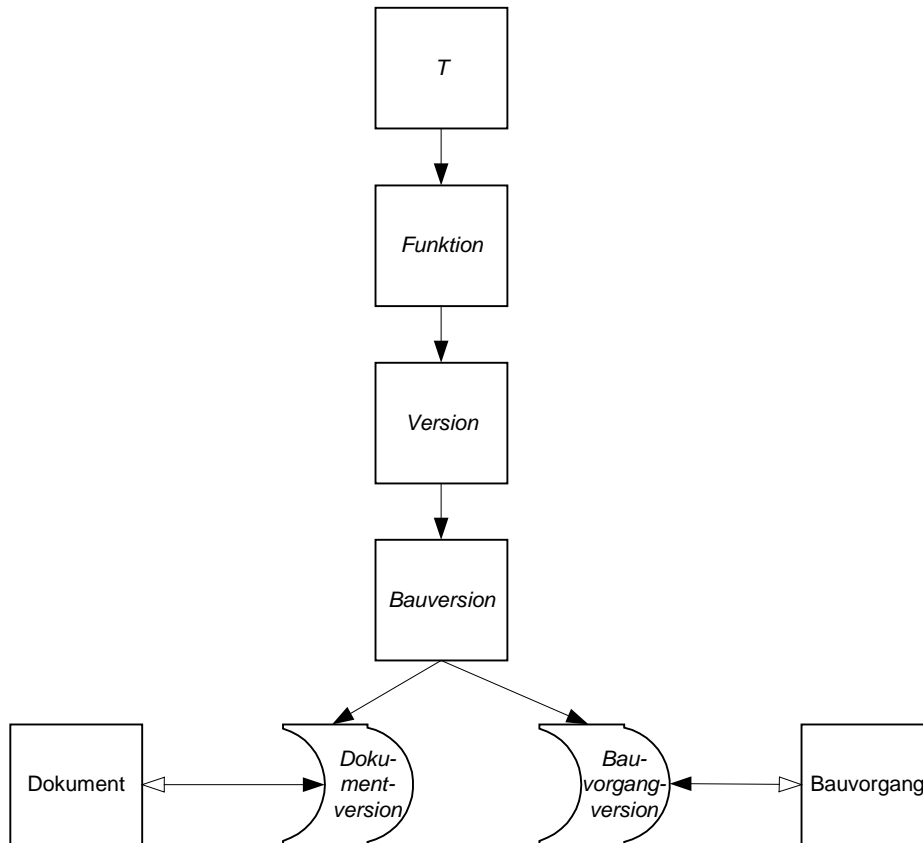


Abbildung 49: Baugrundnetz - Version

Attribute

- **Änderungen.** Hier kann ein natürlichsprachlicher Text eingegeben werden, der die Änderungen beschreibt.
- **Eingefügt am** steht für das Datum, an dem die neue Version eingefügt worden ist. Das Einfügen des Datums sollte automatisch erfolgen und sich auf die Systemzeit beziehen, damit eine Vergleichbarkeit der Datumsangaben gewährleistet ist.

Synonyme

- Variante

Relationen

- Keine

Rollen

- Keine

4.6.7.1 Bauversion

Um feststellen zu können, von welchem Projektbeteiligten eine Version eingefügt worden ist, wird ein neuer Unterbegriff von Version erstellt. Dieser erhält eine neue Relation, die die Beziehung zu einem Projektbeteiligten herstellt.

Attribute

- Keine

Synonyme

- Keine

Relationen

- Eingefügt von *Projektbeteiligter*. Diese Relation zeigt an, welcher Projektbeteiligte die Version eingefügt hat. Auch dieser Eintrag sollte automatisch vom System vorgenommen werden.
- Typ der *Änderung* *Änderung*. Eine Version lässt sich mit einem beliebigen Satz von vordefinierten Änderungstypen kombinieren. Über diese Relation können auch Bauteile bzw. Bauvorgänge gefunden werden, die nicht direkt von einer *Änderung* betroffen sind. (s. auch *Änderung*)

Rollen

- Keine

4.6.7.2 Dokumentversion

Die Dokumentversion wird als Rolle realisiert, die von einem Dokument erweitert werden muß.

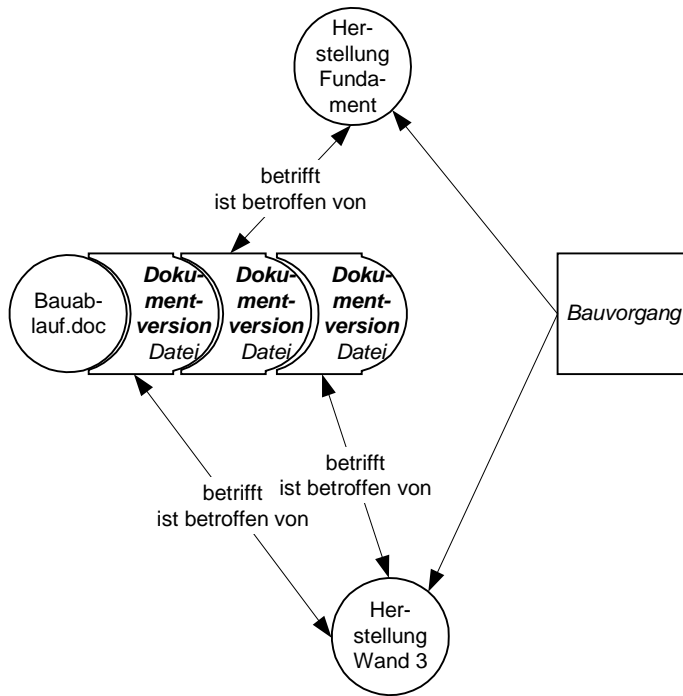


Abbildung 50: Baugrundnetz - Beispiel Dokumentversionen

Attribute

- **Datei**. Ist ein BLOB-Feld (Binary large Object). Darin können beliebige binäre Daten gespeichert werden. So können alle möglichen Dateien, wie Bilder oder serialisierte Objekte, eingefügt werden.

Synonyme

- Keine

Relationen

- **Betrifft Bauvorgang**. Ein Dokument gilt normalerweise für mehrere Bauvorgänge. Eine Änderung kann aber auch nur einen einzigen oder sehr wenige Bauvorgänge betreffen. Daher können durch die Relation **betrifft**, die direkt von einer Änderung betroffenen Bauvorgänge angezeigt werden. Über diese Relation geht auch die Suche nach möglicherweise betroffene Bauvorgängen (Falls ein Änderungstyp angegeben wurde).

Rollen

- Keine

4.6.7.3 Bauvorgangversion

Die Bauvorgangversion wird, wie die Dokumentversion, als Rolle umgesetzt. Sie enthält die letztendlich interessanten Daten eines Bauvorgangs.

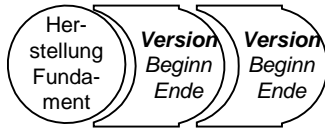


Abbildung 51: Baugrundnetz - Beispiel Bauvorgangversion

Attribute

- **Beginn** bezeichnet den geplanten Starttermin eines Bauvorgangs.
- **Ende** steht für das geplante Ende des Bauvorgangs.
- **Kosten.** Dies sind die geplanten Kosten, die der Bauvorgang verursacht.

Synonyme

- Keine

Relationen

- Keine

Rollen

- Keine

4.6.8 Änderung

Änderungen können als eine Art Template für eine Suche nach betroffenen Objekten angesehen werden. Bei der Änderung wird eine benutzerdefinierte Suche (*Konfigurierbare Suche*) hinterlegt, die versucht, möglicherweise betroffene Bauvorgänge zu finden.

Beispiel:

Eine Wand hat eine Tür. Neben der Tür liegen ein Lichtschalter und die Kabel. Die Abmessungen der Wand und der Tür sind im Dokument *Bauplan* eingezeichnet. Die Lage des Lichtschalters und der Kabel im Plan *Elektrik*. Die Tür soll nun in ihrer Lage verschoben werden. Hierfür wird eine neue *Bauplan*-Dokumentversion angelegt, in der die neuen Abmessungen eingetragen sind.

Da aber die Tür verschoben wurde, müssen auch der Lichtschalter und die Kabel verschoben werden. Dies ist aber nicht im *Bauplan* ersichtlich, da die Elektrik in einem anderen Plan dokumentiert ist.

Daher kann bei einer neuen Version angegeben werden, um welchen Typ Änderung es sich handelt. Hier könnte z.B. ein Typ *Veränderung der Abmessungen* existieren, bei dem eine konfigurierbare Suche hinterlegt ist, die in der Nähe der betroffenen Bauvorgänge ein Bauteil von Typ Elektrik sucht.

Beim verschieben einer Tür, ist nicht immer ein Lichtschalter davon betroffen. Daher ist die beim Änderungstyp abgelegte Suche auch nur eine unterstützende Hilfe, möglicherweise betroffene Bauteile bzw. Vorgänge zu finden.

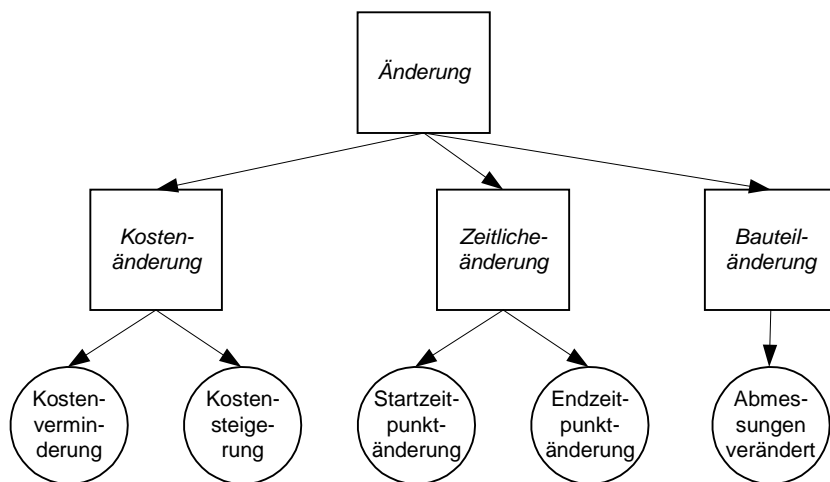


Abbildung 52: Baugrundnetz - Änderungen

Attribute

- **Context** bezeichnet den Namen einer konfigurierbaren Suche, die für diesen Änderungstyp aufgerufen werden kann.

Synonyme

- Keine

Relationen

- Ist Änderungstyp für *Bauversion*. S. hat Änderungstyp

Rollen

- Keine

4.7 Modellierungsbeispiel

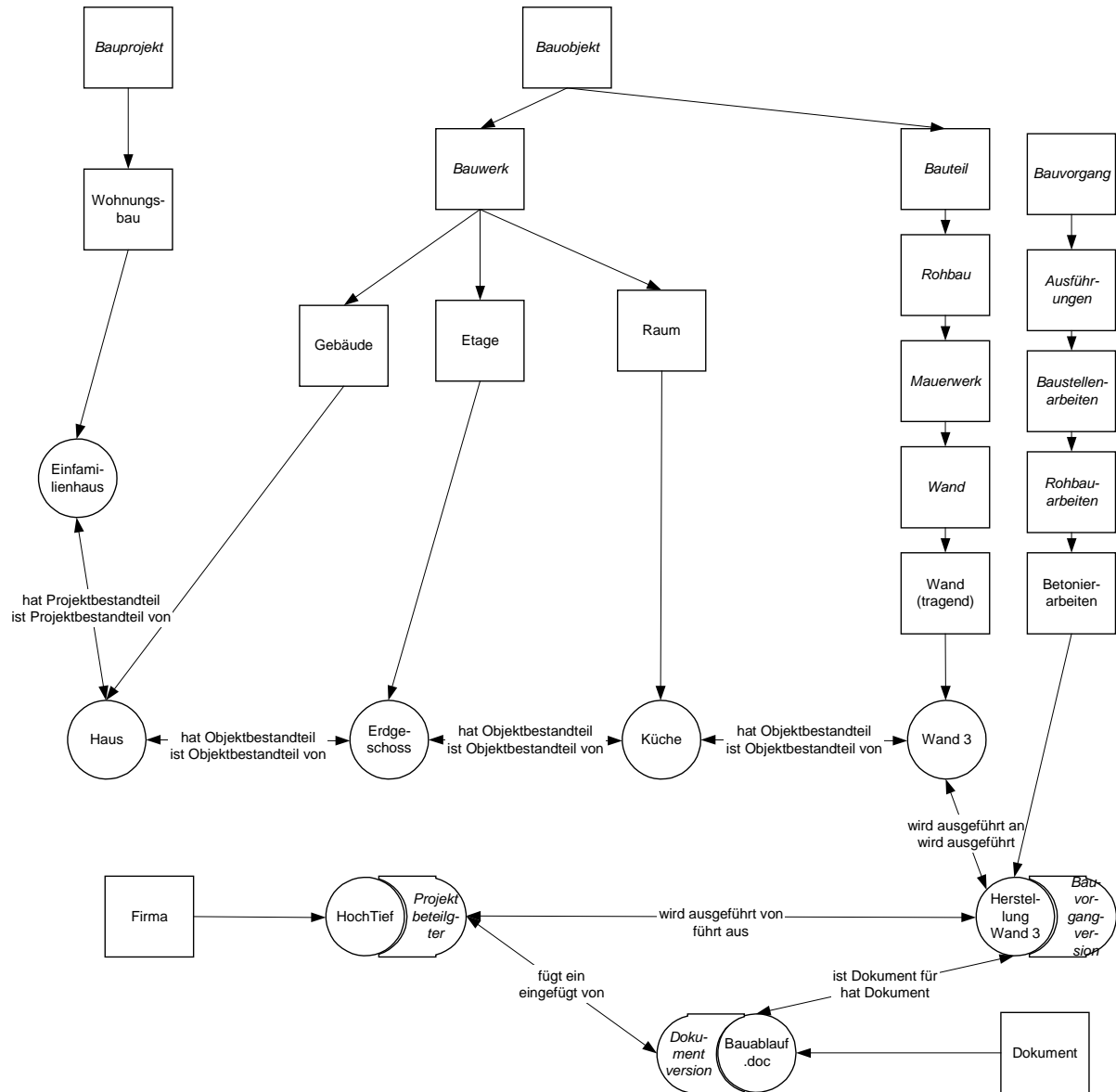


Abbildung 53: Baunetz - Modellierungsbeispiel

5 Modellierung des Grundnetzes mit K-Infinity

Die Modellierung des Baunetzes wird an einem kurzen Beispiel erklärt. Das hier angelegte Netz besteht aus den sechs Begriffen *Aktor*, *Firma*, *Aktorfunktion*, *Baufirma*, *Bauvorgang*, *Betonierarbeiten* und *Planung*.

Zuerst werden ein neues Wissensnetz und die Benutzer angelegt. Hierzu wird das *Admin-Tool* verwendet. Darin wird der Name des K-Infinity-Servers, des Wissensnetzes und eines Benutzers mit Passwort angegeben.

The screenshot shows a window titled "K-Infinity Administration" with the following elements:

- Benutzer anmelden:**
 - Server:
 - Wissensnetz: ...
- Benutzer:** ...
- Passwort:**
 - erstellen
 - löschen
- Datenbestand:**
 - sichern
 - kopieren auf:
 - löschen
 - wiederherst.
 - aktualisieren
- Ende**

Abbildung 54: K-Infinity – Admin-Tool

Danach wird der K-Organizer geöffnet.

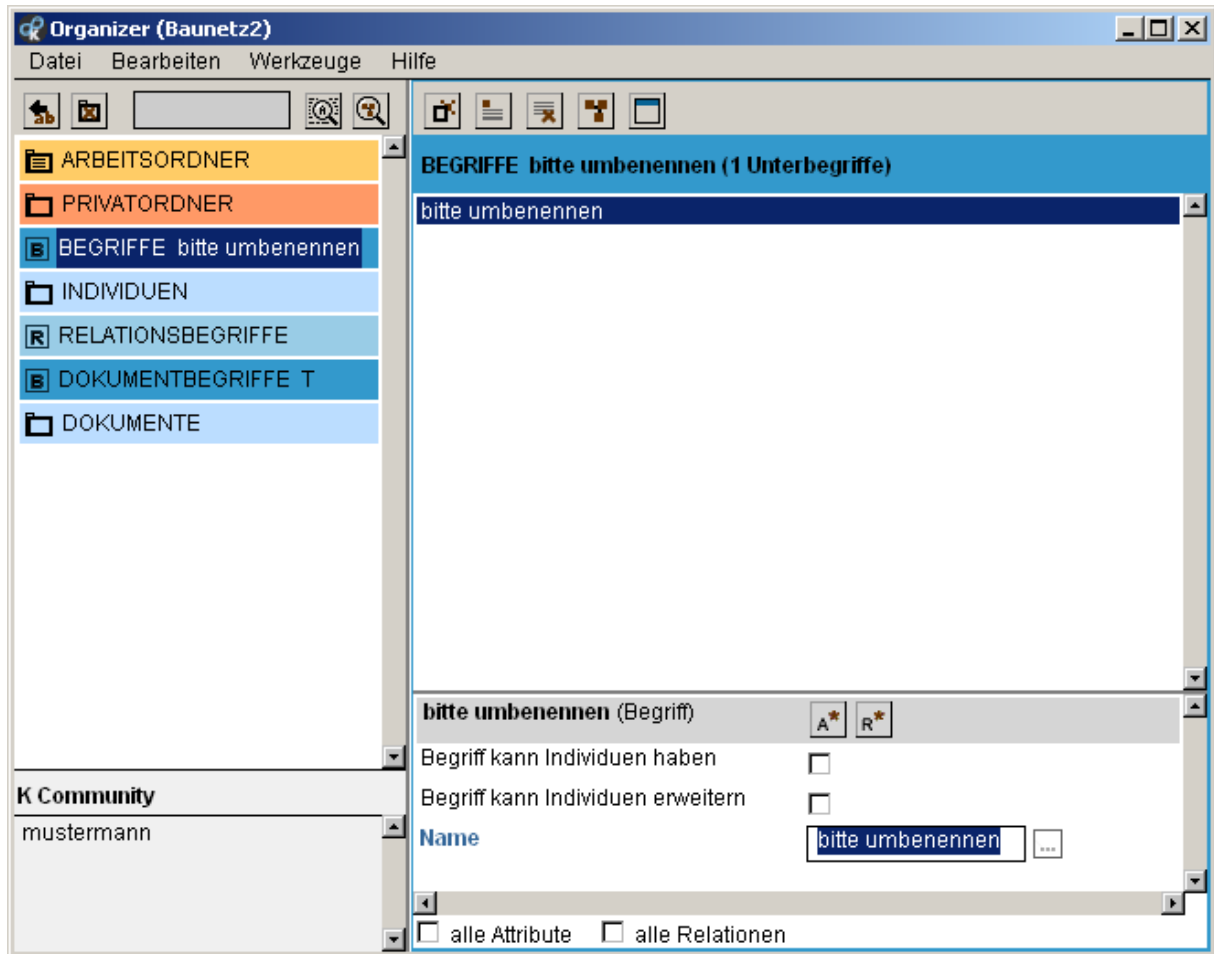


Abbildung 55: K-Infinity - Organizer

Im Organizer muss zuerst der Wurzelbegriff umbenannt werden. Er erhält den Namen *T*. Ein neuer Unterbegriff von *T* wird erstellt, indem *T* selektiert wird und über die Menü- oder Symbolleiste *Neu* ausgewählt wird. Zuerst werden die drei Oberbegriffe *Aktor*, *Aktorfunktion* und *Bauvorgang* als Unterbegriffe von *T* angelegt. Hierzu muss lediglich der Name angegeben werden.

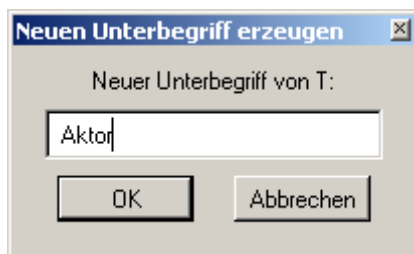


Abbildung 56: K-Infinity - Neuen Begriff anlegen

Die angelegten Begriffe sind automatisch abstrakt und erben von *T* nur das System-Attribut *Name* und die Systemrelationen.

Genauso werden die Begriffe *Firma* als Unterbegriff von *Aktor*, *Baufirma* als Unterbegriff von *Aktorfunktion* angelegt. *Bauvorgang* erhält als Unterbegriff *Betonierarbeiten* und *Planung*.

Zunächst sollten alle Bauvorgänge eine Beschreibung und einen Start- bzw. Endtermin erhalten.

Ein Doppelklick im Organizer auf den Begriff *Bauvorgang* öffnet den Begriffs-Editor. Hier können jetzt die Schemata für den Begriff und die Individuen angelegt werden.

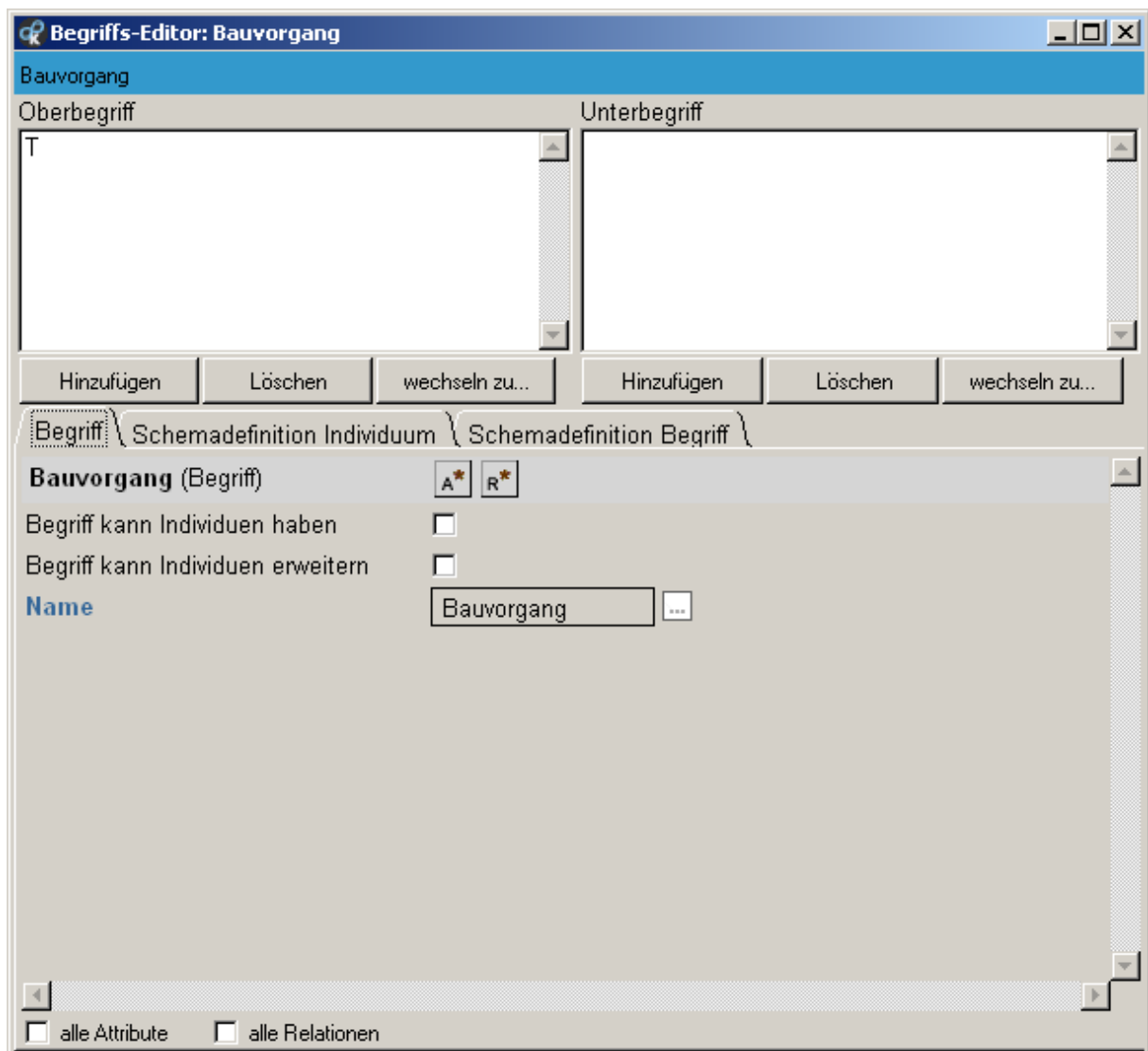


Abbildung 57: K-Infinity – Begriffseditor (1)

Um die Schemadefinition für die Individuen von *Bauvorgang* festzulegen, wird in den Reiter *Schemadefinition Individuum* gewechselt.

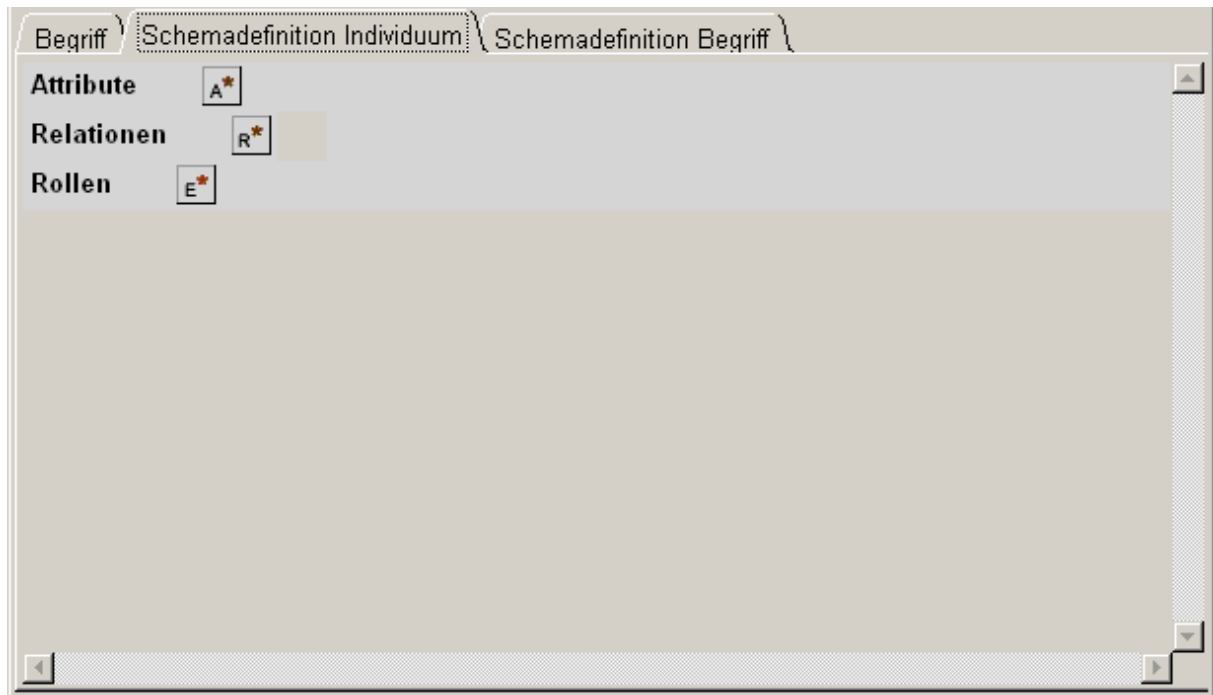


Abbildung 58: K-Infinity - Schemadefinition Individuum

Hier wird über den Button *Neues Attribut* ein Attribut für die Individuen erstellt. Für jedes Attribut muss zuerst ein Attributtyp ausgewählt werden. Für die *Beschreibung* wird der Datentyp Zeichenkette ausgewählt. *Start-* und *Endtermin* erhalten den Datentyp *Datum*.

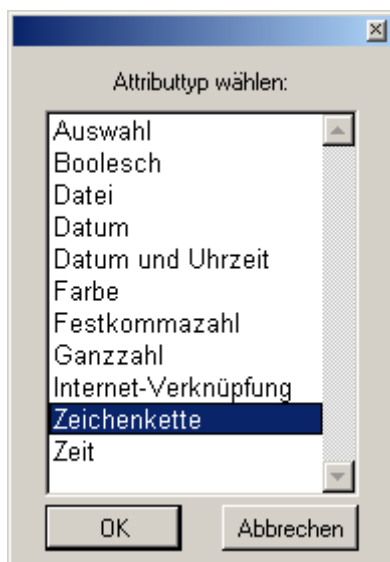


Abbildung 59: K-Infinity - Attributtyp

Danach wird der Name des Attributes angegeben.

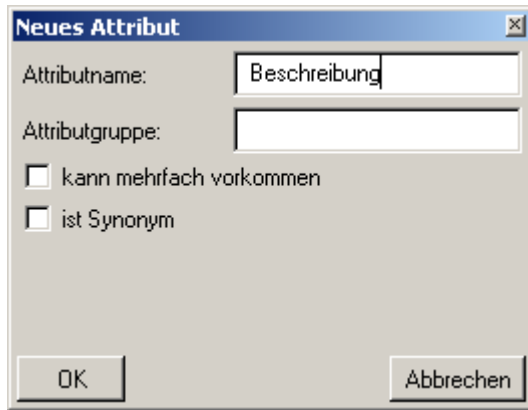


Abbildung 60: K-Infinity - Neues Attribut anlegen

Die Attributgruppe dient zur Verwaltung von Rechten, d.h. es lassen sich Rechte auf Attributgruppen vergeben, so dass z.B. nicht jeder Anwender eine Adresse ändern kann.

Kann ein Attribut mehrfach vorkommen, muss die Check-Box *kann mehrfach vorkommen* selektiert werden. Soll das Attribut gleichzeitig ein Synonym für das Individuum sein, so kann die Check-Box *ist Synonym* ausgewählt sein.

Ein Bauvorgang soll nur von einer Baufirma ausgeführt werden können. Daher wird Baufirma als Rolle für Firmen realisiert, d.h. Individuen von Firma können die Erweiterung Baufirma annehmen, und erst dann einen Bauvorgang ausführen.

Ein Doppelklick auf Baufirma im *Organizer* öffnet wieder den *Begriffs-Editor*.

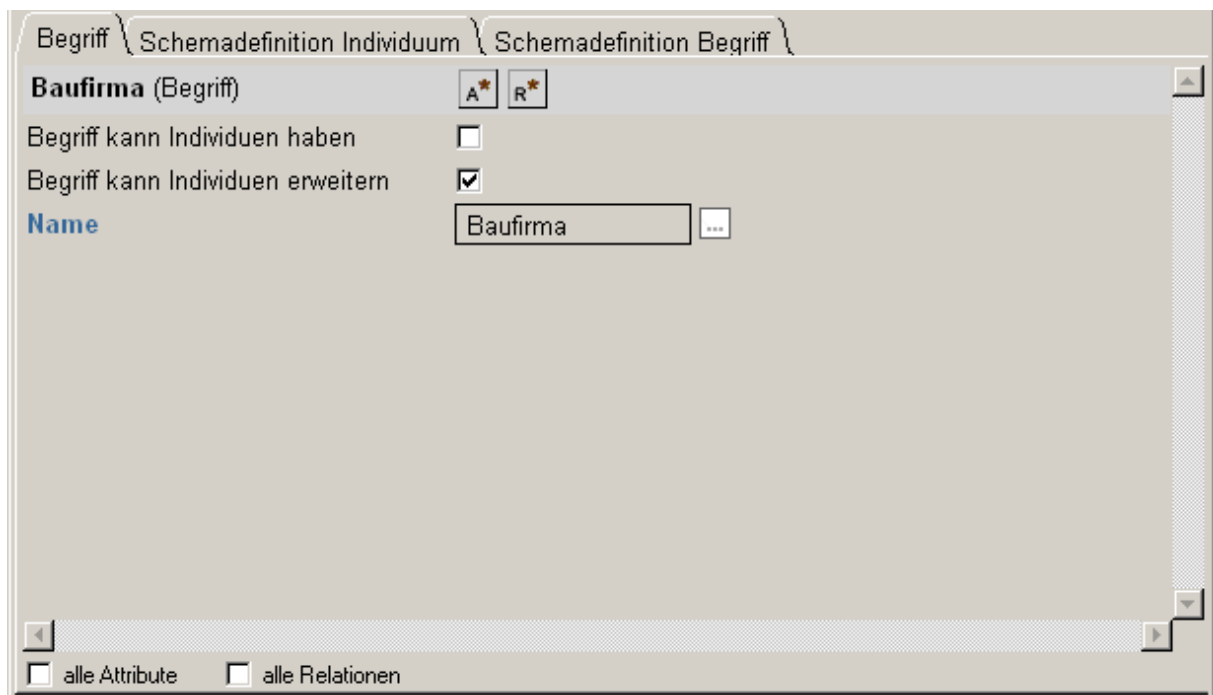


Abbildung 61: K-Infinity - Begriffseditor (2)

Damit ein Begriff Individuen erweitern kann, muss die Check-Box *Begriff kann Individuen erweitern* selektiert werden.

Jetzt muss noch eine Relation erstellt werden, damit Baufirmen einen Bauvorgang ausführen können. Hierzu wird wieder in den Reiter *Schemadefinition Individuum* gewechselt. Eine neue Relation wird über den Button *Relation hinzufügen* angelegt.

Abbildung 62: K-Infinity - Neue Relation anlegen

Hier müssen zuerst ein Name für die Relation und die inverse Relation angegeben werden. Für die inverse Relation muss das Ziel angegeben werden, hier Individuen vom Begriff *Bauvorgang*. Wird *obligatorisch* aktiviert, muss diese Relation immer bei einem neuen Individuum gezogen werden. Für eine Relation kann ebenfalls angegeben werden, ob sie mehrfach vorkommen kann. Da eine *Baufirma* mehrere *Bauvorgänge* ausführen kann, wird die Check-Box aktiviert.

Als Oberrelation wird die System-Relation *relates to* ausgewählt.

Um genauer zu spezifizieren, welche Art von *Bauvorgang* ausgeführt wird, soll die Relation *führt aus* abstrakt sein. Hierzu muss die neu angelegte Relation bearbeitet werden. Dies geschieht über den Button neben der neu angelegten Relation *Relation editieren*.

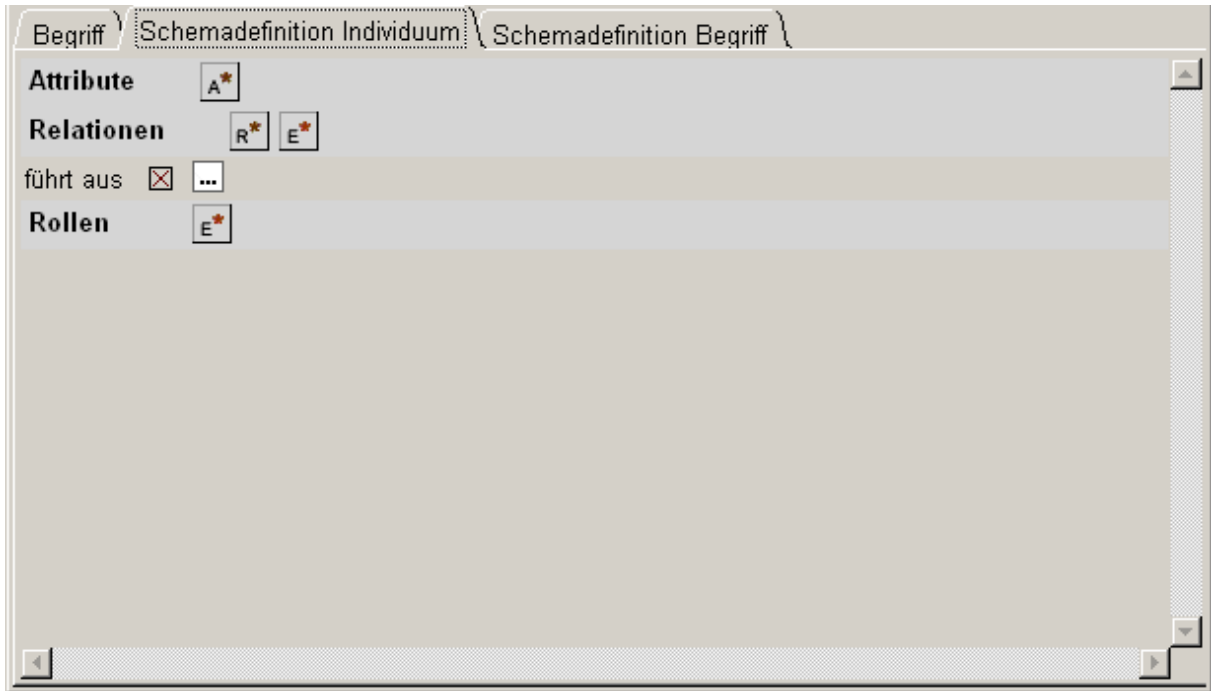


Abbildung 63: K-Infinity - Schemadefinition Individuum (2)

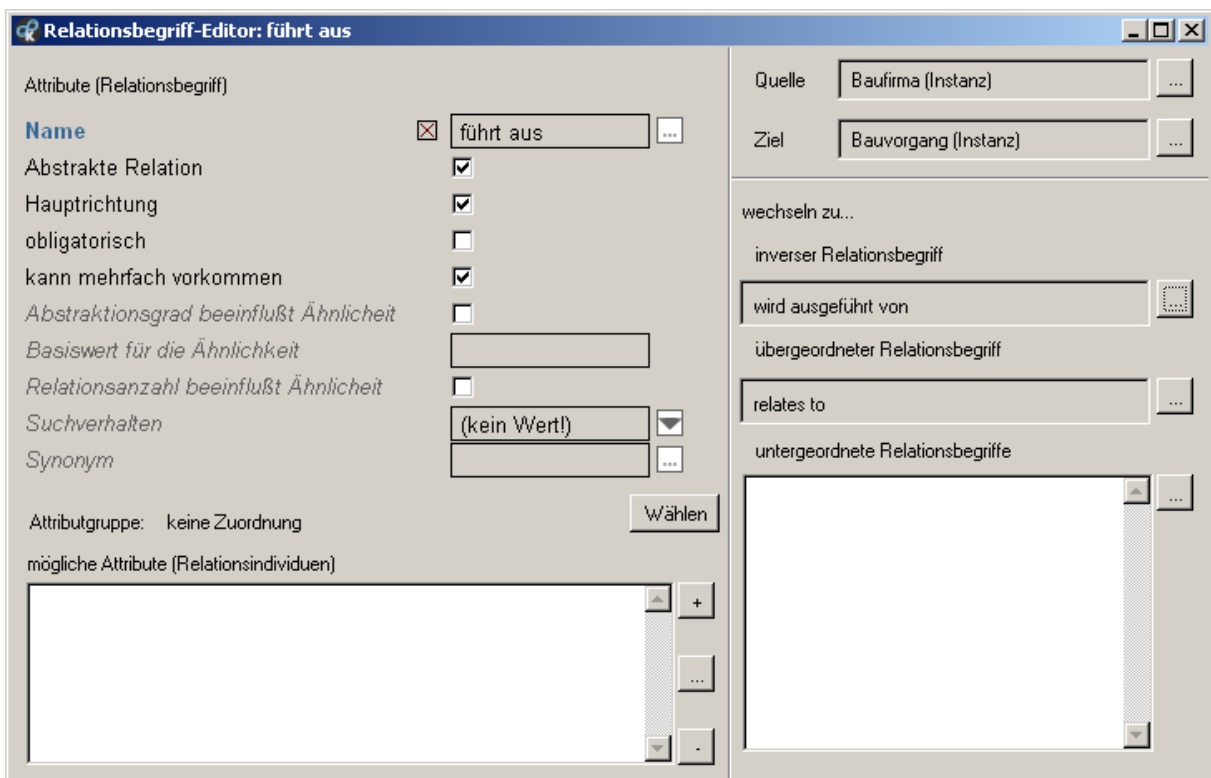


Abbildung 64: K-Infinity - Relationsbegriff-Editor

Bei Relation bzw. der inversen Relation wird die Check-Box *Abstrakte Relation* aktiviert. Hier können auch Synonyme für die Relation angegeben werden.

Die Unterrelationen von *führt aus* sollen dann genauer spezifizieren, welche Art von Vorgang ausgeführt wird. Daher wird eine Relation *betoniert* als Unterbegriff von *führt aus* erstellt.

Abbildung 65: K-Infinity - Neue Relation anlegen (2)

Sie erhält als Ziel *Betonierarbeiten*. Ebenso wird die Relation *plant* als Unterrelation von *führt aus* erstellt. Sie erhält als Ziel *Planung*. Über diese Hierarchie ist es später möglich, alle Bauvorgänge (Planungs- und Betonierarbeiten), oder eben nur die *Betonierarbeiten* die eine *Baufirma* ausführt, zu suchen.

Da die Begriffe *Betonierarbeiten*, *Planung* und *Firma* individuenfähig sein sollen, muss die Check-Box *Begriff kann Individuen haben* im *Begriffs-Editor* aktiviert werden.

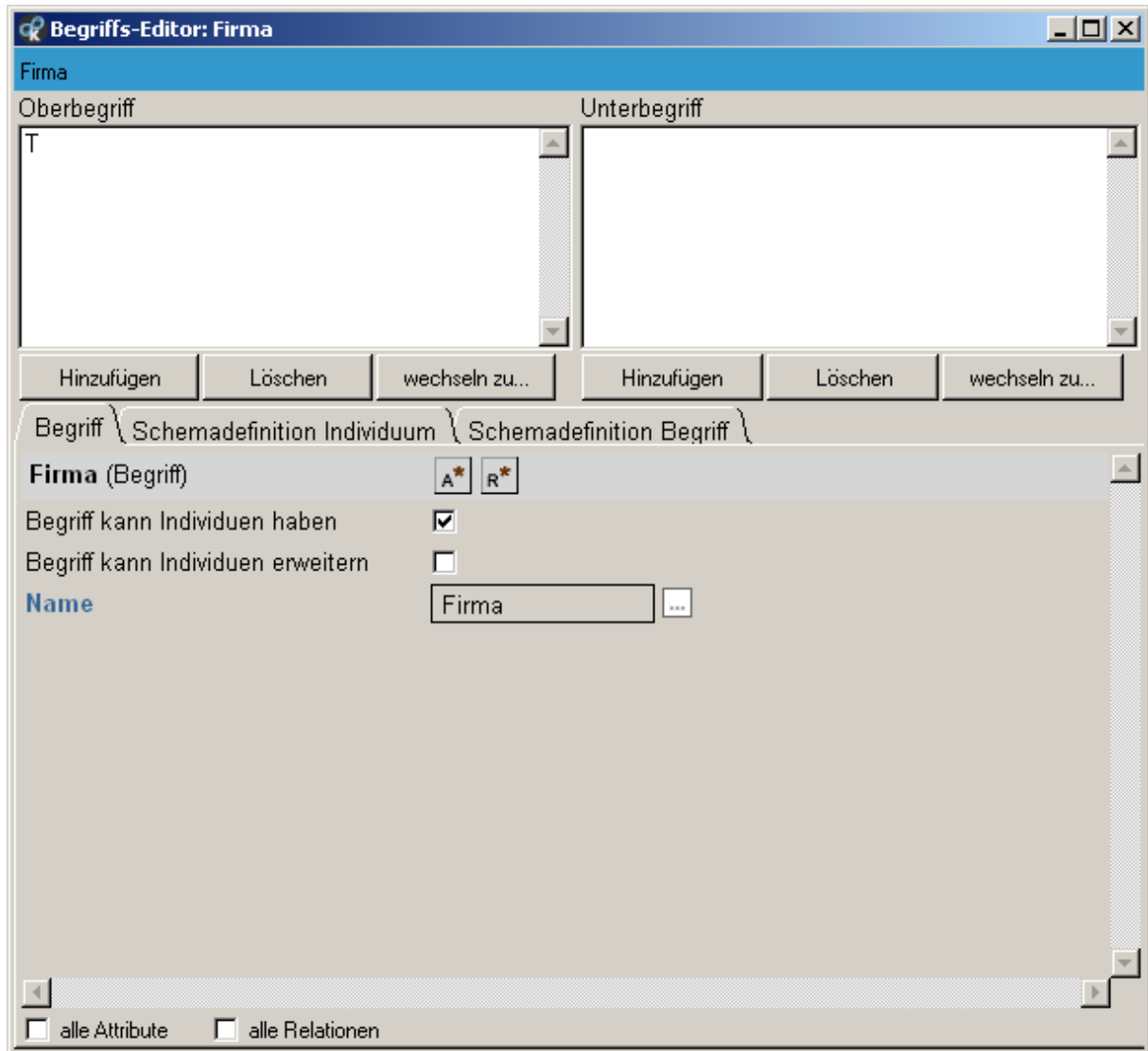


Abbildung 66: K-Infinity - Begriffseditor (3)

Jetzt muss nur noch die Rollenerweiterung *Baufirma* für die Individuen von *Firma* spezifiziert werden. Dies geschieht wieder im Reiter *Schemadefinition Individuum* über den Button *Rollenerweiterung hinzufügen*.

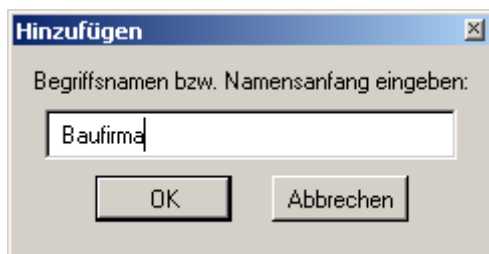


Abbildung 67: K-Infinity - Neue Rollenerweiterung anlegen

Hierzu muss der Name der Rolle angegeben werden. Es reicht, auch nur den Namensanfang einzugeben, um eine Auswahl zu erhalten.

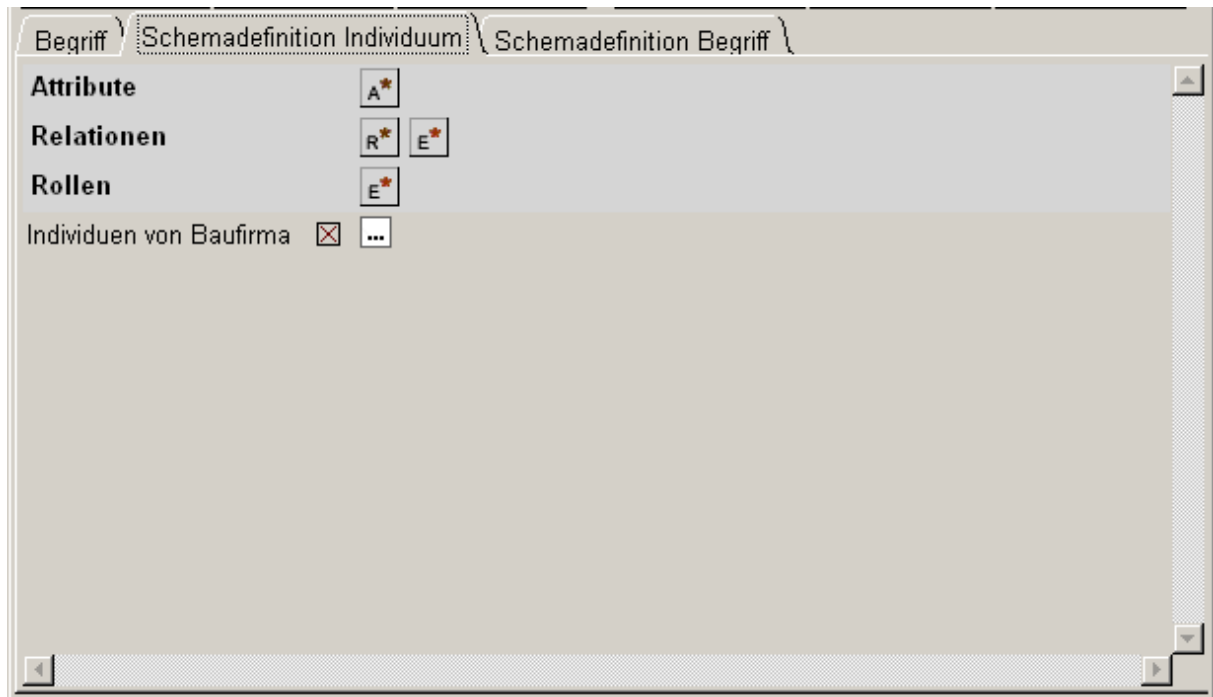


Abbildung 68: K-Infinity - Schemadefinition Individuum (3)

Das angelegte Netz kann jetzt im Graph-Editor betrachtet werden. Hierzu wird ein Begriff selektiert und in der Menü- oder Symbolleiste *Graph* ausgewählt.

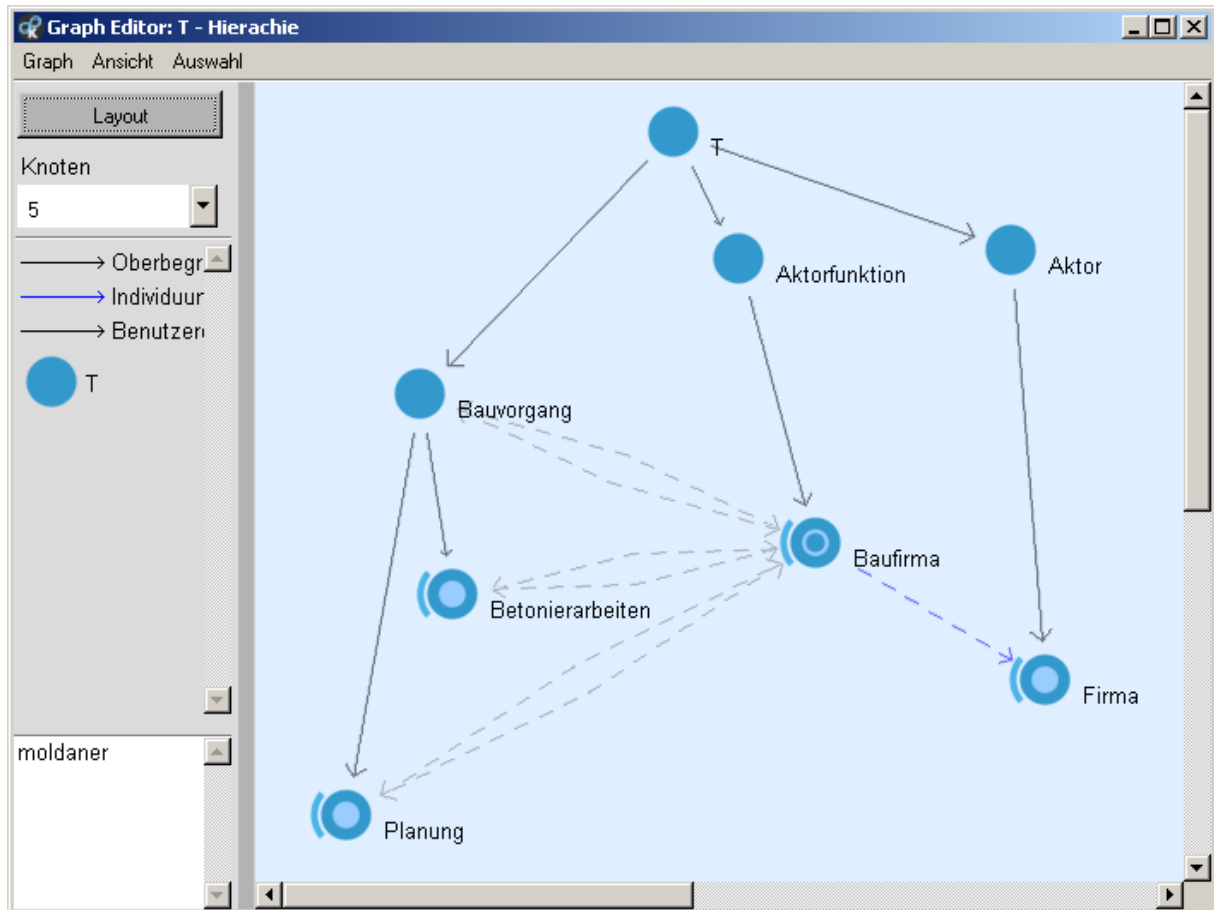


Abbildung 69: K-Infinity - Graph-Editor

Im *Graph-Editor* können ebenso wie im *Organizer* alle Begriffe und Relationen über das Kontext-Menü bearbeitet werden.

Jetzt wird eine neue Firma als Individuum von *Firma* angelegt. Dies geschieht über das Kontext-Menü.

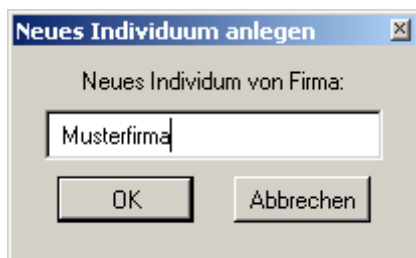


Abbildung 70: K-Infinity - Neues Individuum anlegen

Es muss zuerst der Name des neuen Individuums angegeben werden. Ebenso werden die zwei Individuen *Herstellung Fundament* (abgeleitet von *Betonierarbeiten*) und *Statische Berechnung* (abgeleitet von *Planung*) erstellt. Die neuen Individuen erscheinen sofort im *Graph-Editor*.

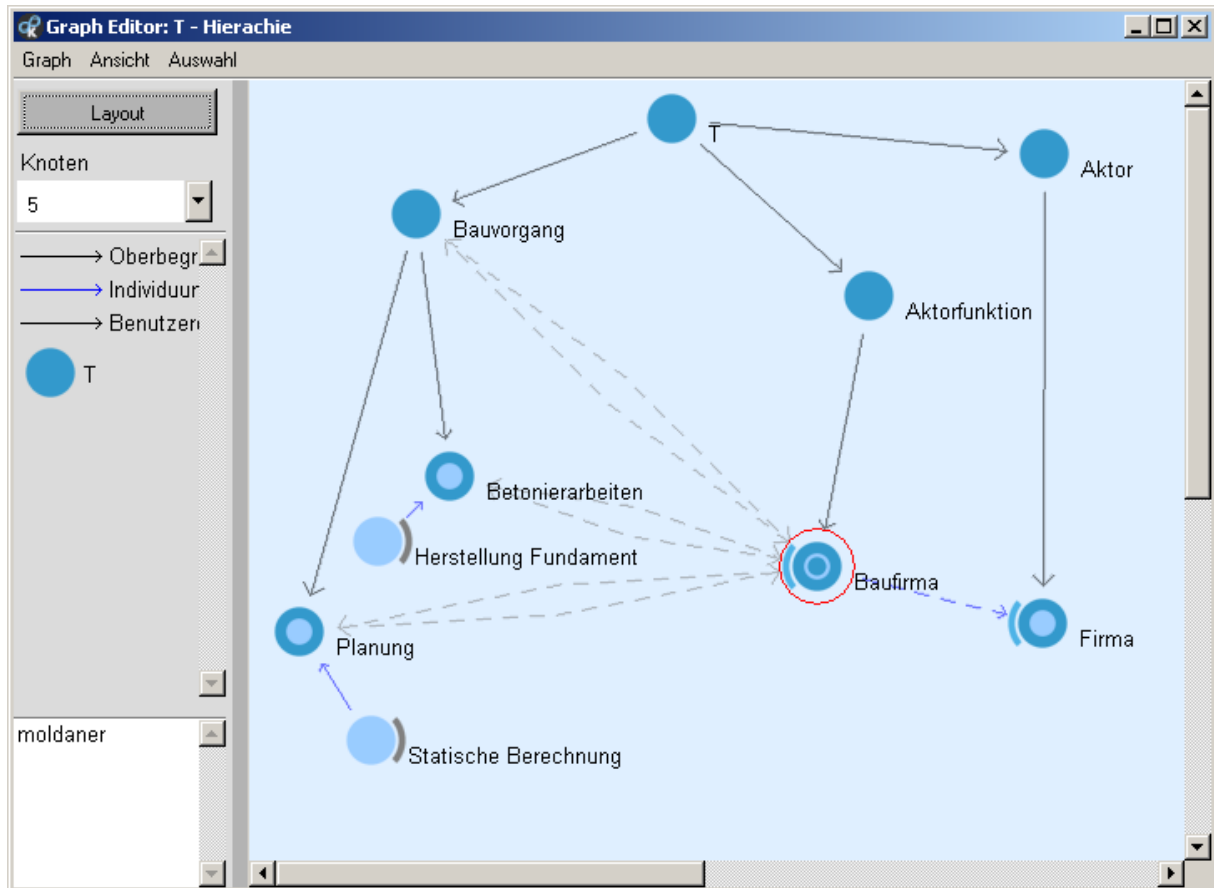


Abbildung 71: K-Infinity - Graph-Editor (2)

Um *Musterfirma* zu einer *Baufirma* zu machen, muss sie mit der Rolle *Baufirma* erweitert werden. Hierzu erscheinen beim Überfahren eines Knotens mit der Maus „Anfasser“, die per Drag-and-Drop auf einen anderen Knoten gezogen werden können.

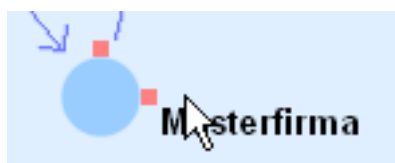


Abbildung 72: K-Infinity - "Anfasser "

Wird der „Anfasser“ auf die Rolle *Baufirma* gezogen, nimmt die Firma *Mustermann* die Rolle *Baufirma* an. Jetzt kann ebenfalls per Drag-and-Drop eine Relation zwischen *Musterfirma* und *Herstellung Fundament* bzw. *Statische Berechnung* hergestellt werden. K-Infinity erkennt automatisch, welche Relationen zwischen zwei Begriffen gezogen werden können. Ist nur eine Relation möglich, wird diese sofort gezogen. Stehen mehrere Relationen zur Auswahl, werden diese angezeigt, und der Anwender muss die entsprechende auswählen.

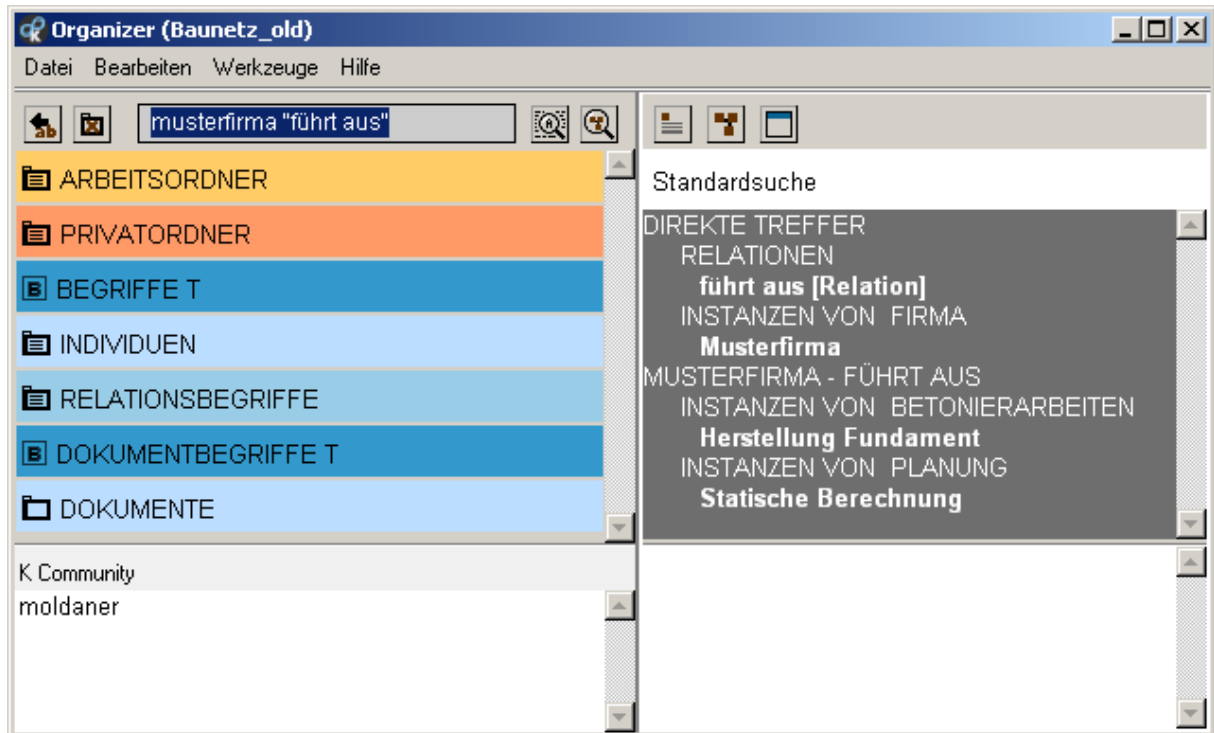


Abbildung 74: K-Infinity - Semantische Suche (1)

Die Suche „Musterfirma plant“ liefert tatsächlich nur die *Statische Berechnung*.

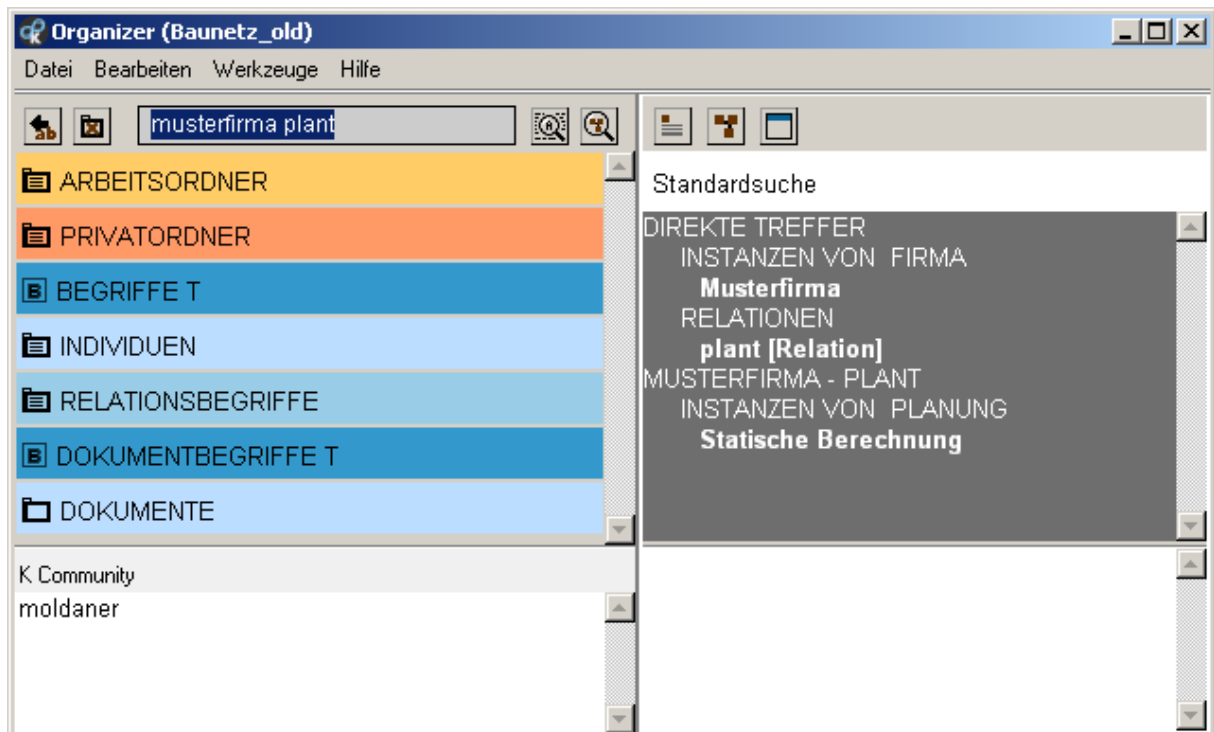


Abbildung 75: K-Infinity - Semantische Suche (2)

6 Anwendungsbeispiel

Ziel der Umsetzung war es, eine internetbasierte Plattform zu erstellen, die den Zugriff auf die im Wissensnetz abgelegten Informationen bietet und ebenso das Einfügen neuer Informationen bzw. Änderungen ermöglicht.

6.1 Anforderungen

- Der Zugriff sollte von jedem PC aus erfolgen können, ohne dass zuerst eine gesonderte Anwendung installiert werden muss. Die Anwendung sollte plattformübergreifend, und nicht auf einen speziellen Browser ausgerichtet sein
- Mehrere Projekte sollten gleichzeitig verwaltbar sein. Ebenso sollte der Anwender nur die Projekte sehen, an denen er beteiligt ist.
- Das System sollte eine passwortgeschützte Anmeldung bieten, um unautorisierten Zugriff zu verhindern.
- Das Einfügen von Informationen und Dokumenten sollte möglichst automatisiert erfolgen, d.h. das System sollte dem Anwender die Zuordnung so leicht wie möglich machen.
- Es sollte leicht erweiterbar sein, um an neue Aufgabenstellungen angepasst werden zu können. Die Anwendung sollte sich leicht in einen bestehenden Internetauftritt integrieren lassen.

6.2 Verwendete Werkzeuge

6.2.1 K-Infinity

K-Infinity ist die Plattform zur Entwicklung des Wissensnetzes.

6.2.2 Usage Tools Server

Der *Usage Tools Server* wird als Brücke zu K-Infinity verwendet. Er nimmt die Anfragen an das Wissensnetz als HTTP-Requests entgegen, und liefert das Ergebnis als XML-Datenstrom zurück.

6.2.3 Web-Server

Als Webserver wird die offizielle Referenz Implementierung für die *JAVA Servlet* und *JAVA Server Pages (JSP)* Technologie, der *Jakarta Tomcat Server* verwendet (<http://jakarta.apache.org/>). Dieser steht unter der *Apache Software License* und ist frei verfügbar.

6.2.4 JAVA Servlet Technologie

Die *JAVA Servlet* Technologie der Firma SUN (<http://www.java.sun.com>) ermöglicht den Einsatz von dynamischen Webseiten. Ein Servlet kann wie ein Applet angesehen werden, mit dem Unterschied, dass es nicht auf der Client-Seite, sondern auf einem Webserver läuft. Servlets haben die volle Unterstützung der JAVA-API und bieten darüber hinaus erweiterte Möglichkeiten zur Abarbeitung HTTP-spezifischer Anfragen.

6.2.5 XML-Parser

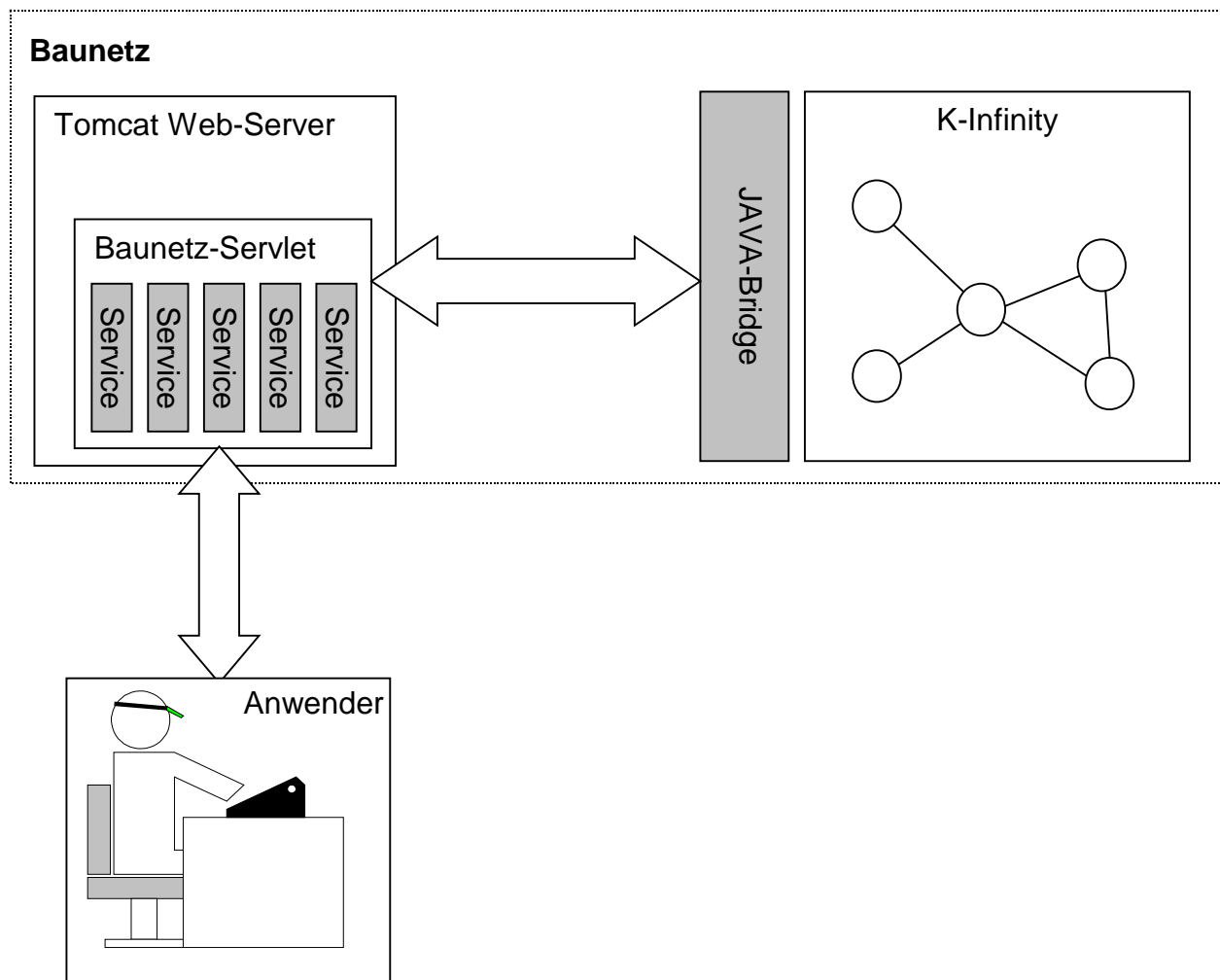
Um spezielle Anfragen an die JAVA-Bridge im XML-Format zu erzeugen, und die ebenfalls im XML-Format erhaltenen Antworten aufzubereiten, wird der *XERCES*-Parser des „Apache XML Project“ (<http://xml.apache.org/xalan-j/>) verwendet.

6.2.6 XSL und XSLT Stylesheet Processor

Antworten der JAVA-Bridge, die direkt verwendet werden können, müssen, da sie im XML-Format vorliegen, nach HTML transformiert werden. Hierfür ist der XSLT-Prozessor *XALAN* des „Apache XML Project“ zuständig. Dieser benötigt zur Transformation von XML nach HTML zusätzlich ein XSL-Stylesheet, das die Transformationsregeln enthält.

6.2.7 Das com.oreilly.servlet Paket

Das `com.oreilly.servlet` Paket enthält Klassen, die die Basis-Funktionalität der Servlets erweitern. Konkret werden sie für die Abarbeitung des Dateitransfers vom Anwender zum Servlet verwendet.



6.2.8 Grundkonzept

Abbildung 76: Gesamtkonzept Baunetz

Ein neues Projekt wird grundsätzlich mit der K-Infinity-Software angelegt. Generell sollten Änderungen am Netz bzw. das Anlegen von neuen Projekten nur von Personen vorgenommen werden, die sich mit der Netzstruktur eines bestimmten Projektes auskennen. Dies vermeidet das Anlegen von Dubletten und falschen Relationen. Da nur wenige geschulte Personen für die Administration des Netzes zuständig sind (dies sollte in der Regel der Projektleiter sein), kann zum Anlegen neuer Objekte und Relationen auf internetbasierte und browserfähige Werkzeuge verzichtet werden. Darüber hinaus bietet die K-Infinity-Software mehrere grafische Darstellungsarten des Netzes, was die Übersicht der komplexen Netzstruktur erhöht.

Der Zugriff für alle Projektbeteiligten auf das System sollte in einem Web-Browser erfolgen. Um den Installationsaufwand für alle Projektbeteiligten so gering wie möglich zu halten, wurde eine Umsetzung mittels dynamischer Web-Seiten gewählt. Dies erfordert keine

Installation gesonderter Werkzeuge wie Plugins oder ActiveX-Komponenten. Umgesetzt wird es mit Hilfe der JAVA Servlet Technologie und einem Tomcat Web-Server.

Um ein möglichst offenes und erweiterbares Konzept zu gewährleisten, werden die einzelnen Anfragen als Dienst implementiert. Der Zugriff auf die Dienste erfolgt über einen Servicemanager. Dies soll es ermöglichen, auch während der Laufzeit neue Dienste zu implementieren. Diese müssen nur dem Servicemanager bekannt gemacht werden und können dann bei einem Aufruf gestartet werden.

Die Kommunikation des Servlets mit K-Infinity erfolgt über eine JAVA-Bridge. Alle Zugriffe auf das Wissensnetz erfolgen als HTTP-Anfragen. Diese können gegebenenfalls im Post-Teil der Anfrage XML-Syntax enthalten. Die JAVA-Bridge liefert die Antworten als XML-Datenstrom. Es ist die Aufgabe des jeweiligen Dienstes, die Anfragen des Anwenders in die entsprechenden HTTP-Anfragen zu übersetzen und die Ergebnisse entsprechend aufzubereiten.

6.3 Umsetzung des Internetauftritts mit JAVA-Servlets

6.3.1 Das Paket *com.iviews.bn*

Das Servlet nimmt die Anfragen des Anwenders entgegen, arbeitet die Anfrage auf und generiert daraus eine Antwort. Um zu gewährleisten, dass das Servlet für beliebige Anfragen erweiterbar ist, wird ein modularer Aufbau eingesetzt.

Die Klassen des Baunetz-Servlet sind im Paket *com.iviews.bn* zusammengefasst.

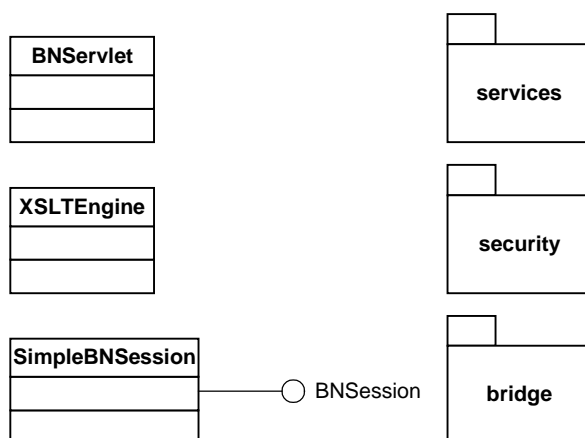


Abbildung 77: Servlet - Übersicht Paket *com.iviews.bn*

6.3.1.1 Die Klasse BNServlet

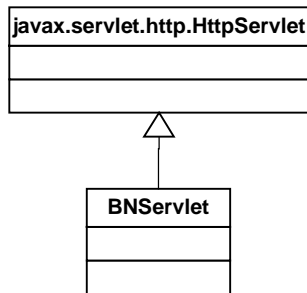


Abbildung 78: Servlet - BNServlet

Die Klasse *BNServlet* ist das eigentliche Servlet und wird vom Servlet-Container (Tomcat) geladen. Sie kümmert sich um die Benutzerauthentifikation und das Anlegen neuer, benutzerspezifischer Session-Daten.

6.3.1.2 Die Klasse XSLTEngine

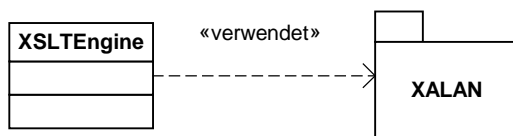


Abbildung 79: Servlet - XSLTEngine

Klasse *XSLTEngine* ist für die Transformation von XML nach HTML zuständig. Sie verwendet den XALAN XSLT-Prozessor.

6.3.1.3 SimpleBNSession

SimpleBNSession implementiert das Interface *BNSession* und dient zur Verwaltung der benutzerspezifischen Session-Daten. Sie überprüft, ob der Benutzer bereits am System angemeldet ist und leitet dann die Anfragen an die spezifischen Dienste weiter.

6.3.1.4 Das Paket com.iviews.bn.services

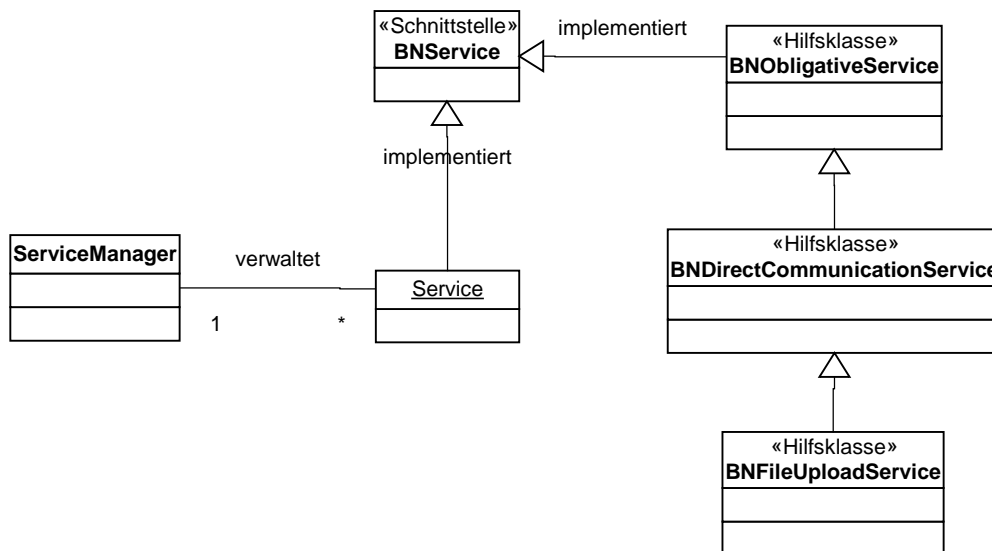


Abbildung 80: Servlet - Paket com.iviews.bn.services

Der *ServiceManager* verwaltet die verschiedenen Baunetz-Dienste. Er weiß, wo die Dienste zu finden sind, und welcher Dienst für eine bestimmte Anfrage zu laden ist. Das Servlet bzw. die *BNSession* greift über den *ServiceManager* auf die verschiedenen Baunetz-Dienste zu. Diese müssen das Interface *BNService* implementieren. Um die Implementierung der Schnittstelle zu erleichtern, existieren die drei Helferklassen *BNObligativeService*, *BNDirectCommunicationService* und *BNFileUploadService*.

Die einzelnen Dienste wissen dann, welche Anfragen an das Wissensnetz zu generieren sind, und wie die Antwort zu verarbeiten ist. Hierfür werden entweder die Antworten direkt verwendet, oder sie werden vor der Transformation entsprechend aufbereitet. Der Service weiß ebenso, welche XSL-Datei zur Transformation verwendet werden muss.

6.3.1.5 Das Paket com.iviews.bn.bridge



Abbildung 81: Servlet -Paket com.iviews.bn.bridge

Die Klasse *CoastBridge* dient zur direkten Kommunikation mit der *JAVA-Bridge*, d.h. die Antwort auf eine Anfrage kommt direkt als XML-Datenstrom zurück und kann sofort in HTML transformiert werden.

Die Klasse *InstanceFactory* bietet Funktionen zum Erstellen bzw. Bearbeiten von Objekten im Wissensnetz. Da hierfür zum Teil mehrere Transaktionen nötig sind, nimmt die Klasse diese Aufgabe den verschiedenen Diensten ab und bietet einfache Schnittstellen zum generieren bzw. Bearbeiten von Objekten.

6.3.1.6 Das Paket `com.iviews.bn.security`

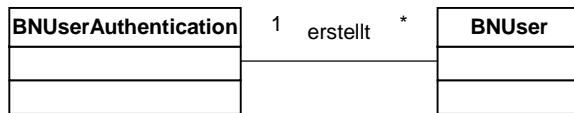


Abbildung 82: Servlet -Paket `com.iviews.bn.security`

Die Klasse *BUserAuthentication* übernimmt die Authentifizierung der Anwender im System. Ist ein Benutzer im System registriert, so erstellt sie ein *BUser*-Objekt, das die für die Dienste notwendigen anwenderspezifischen Informationen enthält.

6.3.1.7 Abläufe

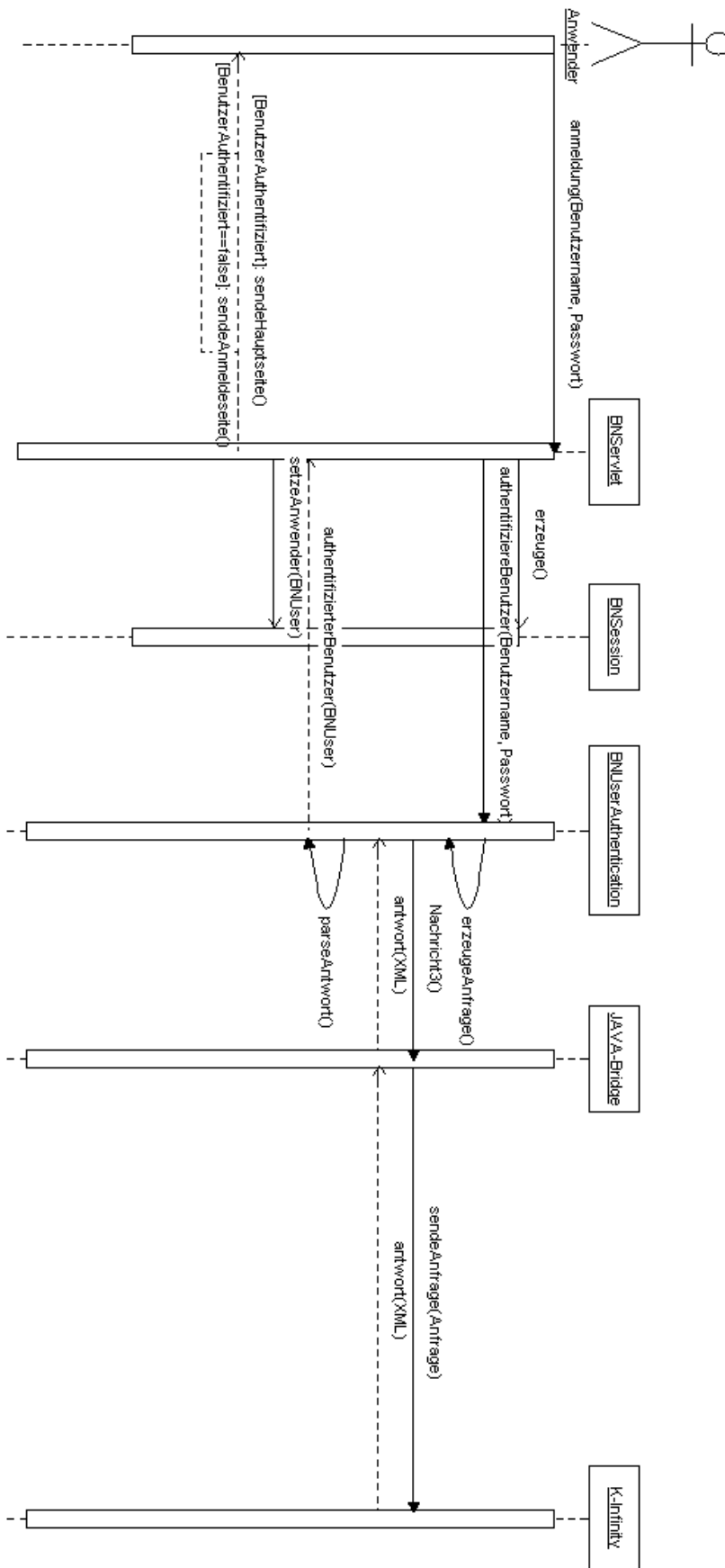


Abbildung 83: Servlet - Anmeldevorgang

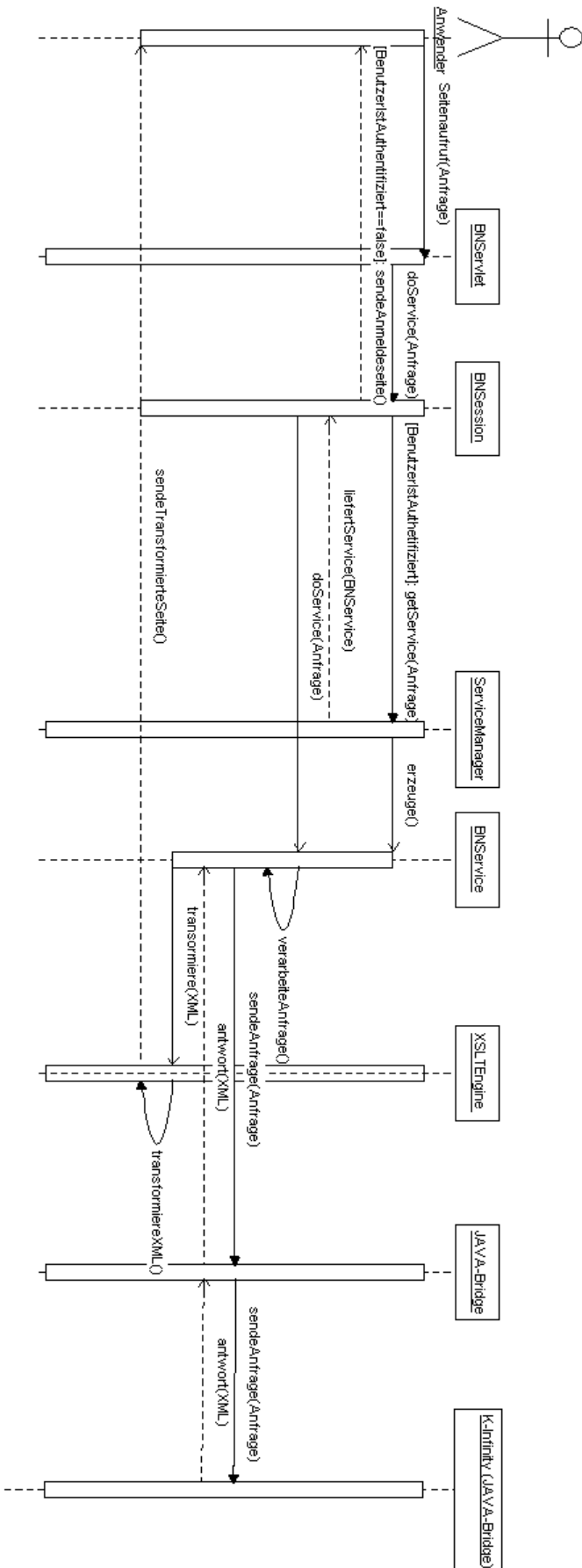


Abbildung 84: Servlet - Dynamische Seitengenerierung

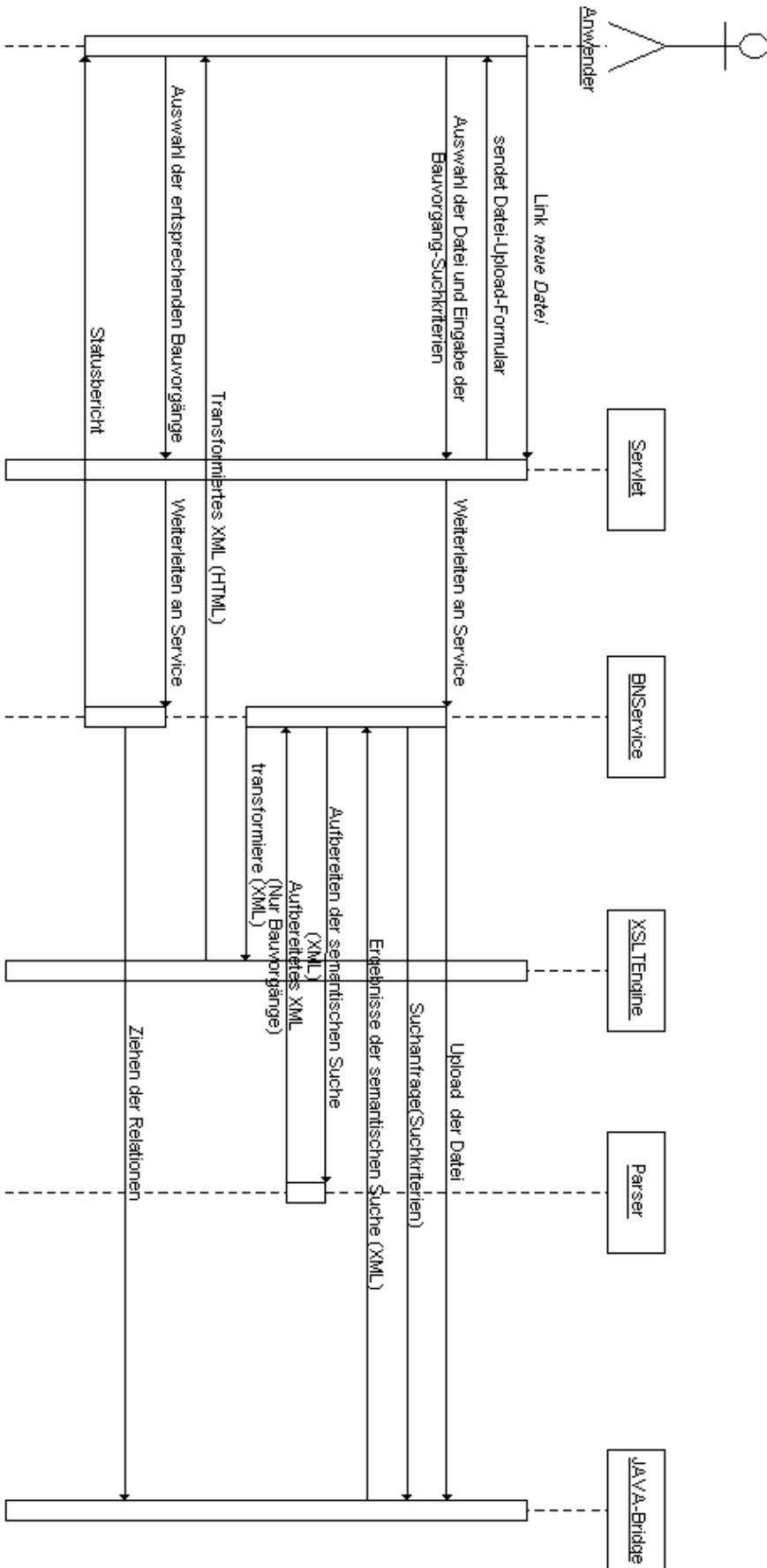


Abbildung 85: Servlet - Dateiupload

6.3.2 XSL

Um die XML-Daten entsprechend aufzubereiten, benötigt der XSLT-Prozessor XSL-Stylesheets. Mit Hilfe der Stylesheets kann der Prozessor die XML-Dokumente in unterschiedliche Formate transformieren. Um die Ergebnisse im Browser betrachten zu können, werden die Daten in das HTML-Format transformiert.

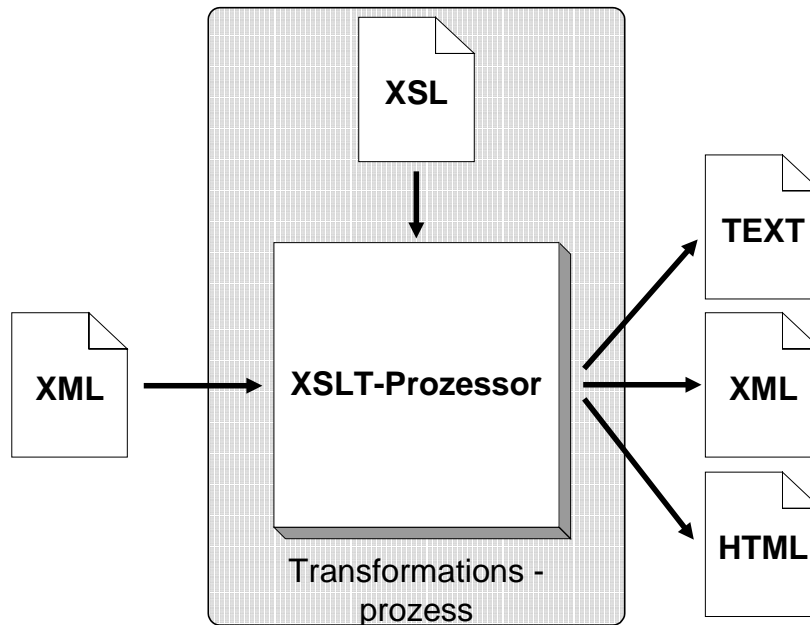


Abbildung 86: Servlet - XSLT-Prozessor

Die Stylesheets befinden sich in einem bestimmten Verzeichnis. So können sie von mehreren Diensten gleichzeitig verwendet werden. Die Dienste kennen nur den Namen des Stylesheets und laden dieses bei Bedarf. So lassen sich durch Ändern der Stylesheets jederzeit der Stil und das Layout der generierten Seiten anpassen, um diese z.B. in einen bestehenden Internetauftritt zu integrieren.

6.4 Konzeption des Internetauftritts

6.4.1 Aufbau des Internetportals

Um Zugang zum System zu erhalten, muss sich der Anwender zunächst anmelden. Nach der Anmeldung gelangt der Benutzer direkt auf die Baunetz-Hauptseite. Von hier aus gelangt er zu allen Bereichen des Systems.

Die Navigation ist in zwei Bereiche gegliedert. Im projektspezifischen, allgemeinen Teil stehen alle Informationen, die direkt das ausgewählte Projekt betreffen.

Der zweite Bereich (*MeinBaunetz*) ist auf den angemeldeten Anwender bezogen, d.h. hier stehen alle anwenderspezifische Informationen bereit.

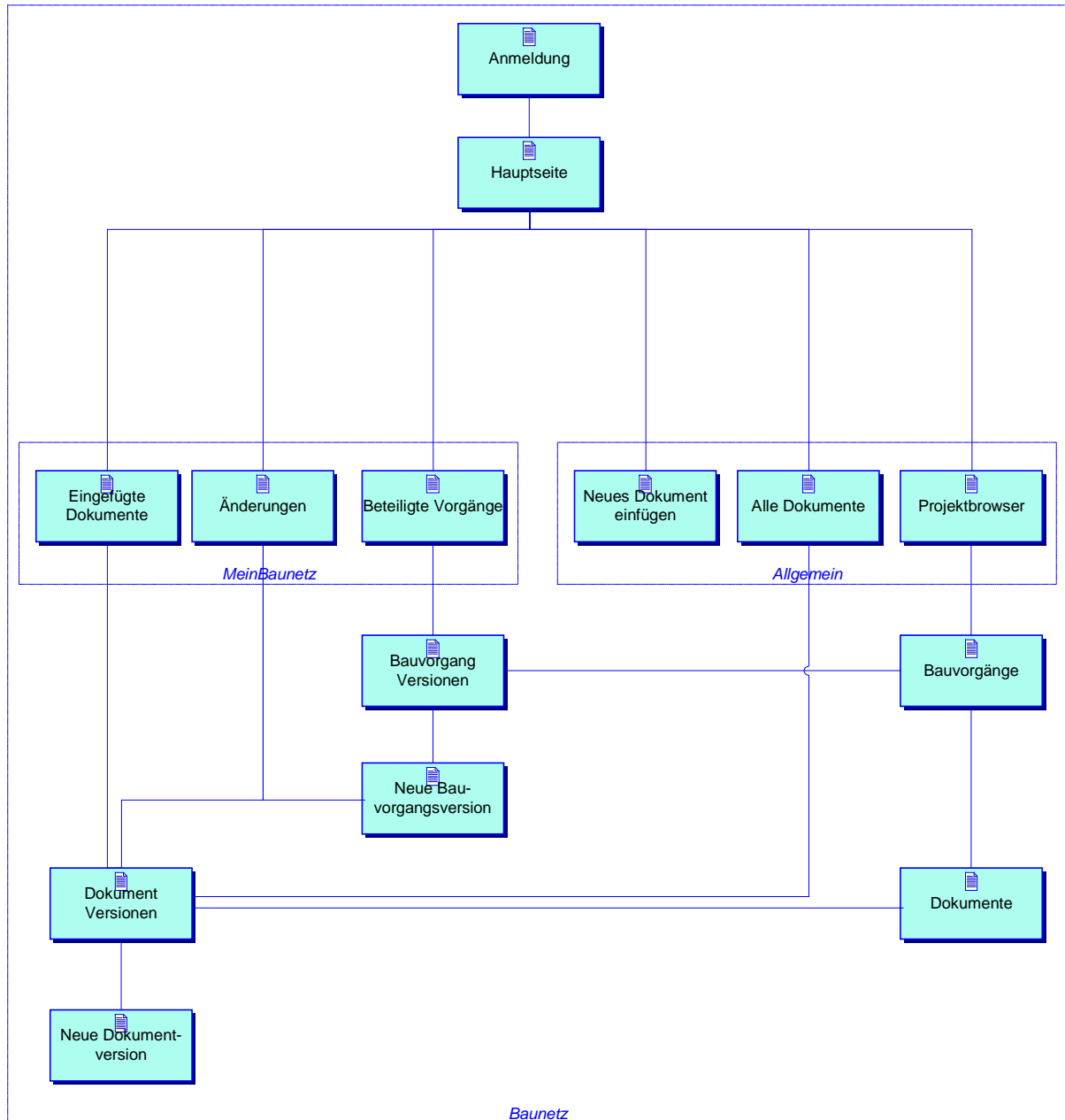


Abbildung 87: Baunetz – Konzeption der Internetseiten

Der Aufbau der Baunetz-Hauptseite ist in drei Bereiche (Frames) gegliedert:

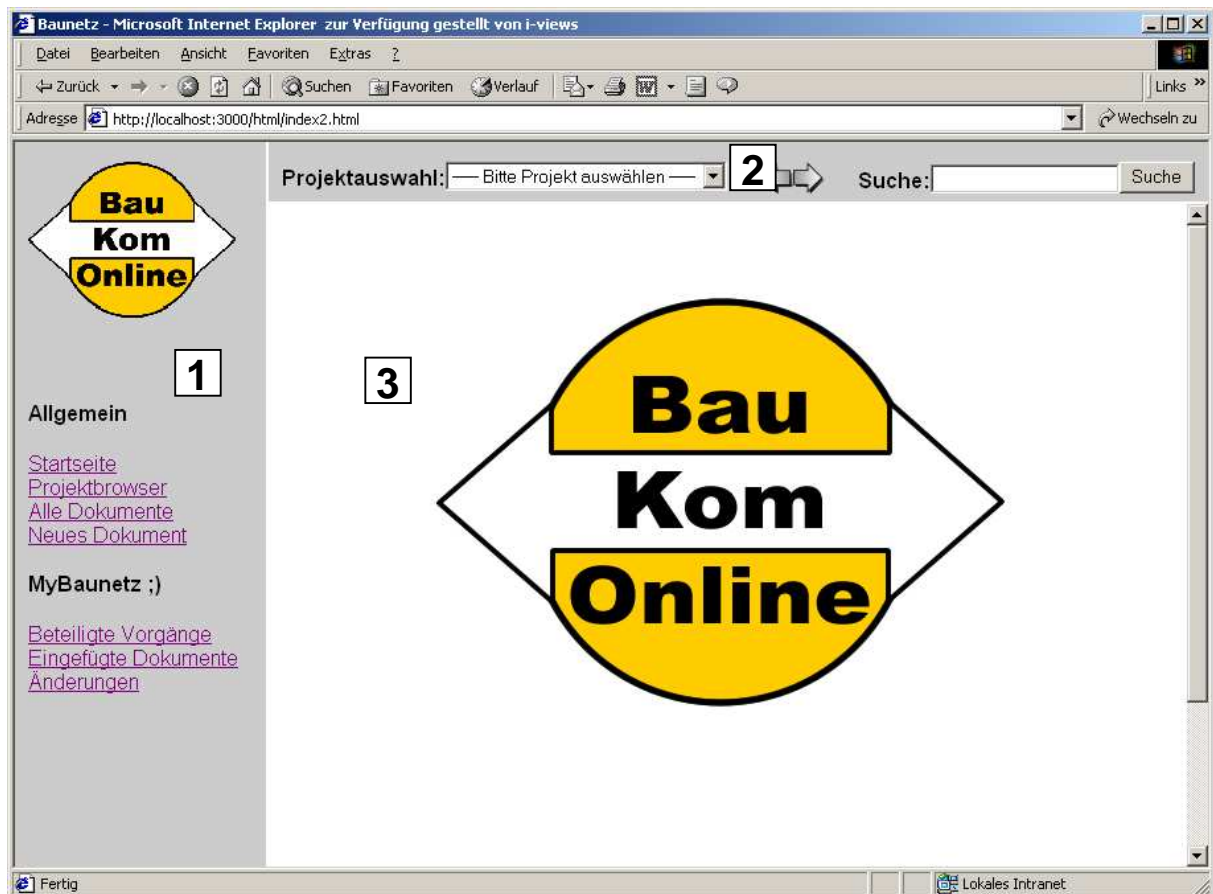


Abbildung 88: Baunetz - Frames

1. Die Navigationsleiste

Die Navigationsleiste befindet sich auf der linken Seite des Browserfensters. Von hier gelangt man zu allen wichtigen Funktionen des Baunetzes. Die Navigation ist in zwei Bereiche unterteilt. Unter *Allgemein* sind alle, das in der *Projektauswahl-* und *Suchleiste* (2) ausgewählte Projekt betreffenden Funktionen vereint. Unter *MeinBaunetz* sind alle, den aktuell angemeldeten Benutzer betreffenden Funktionen zu finden.

Die einzelnen Links werden in den anschließenden Kapiteln näher erläutert.

2. Projektauswahl- und Suchleiste

In der Projektauswahl- und Suchleiste kann der Benutzer zwischen den verschiedenen Projekten, an denen er beteiligt ist, wechseln.

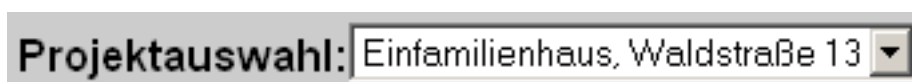


Abbildung 89: Baunetz - Projektauswahl

Anmerkung: Generell ist das Baunetz für die gleichzeitige Verwaltung von mehreren Projekten vorgesehen. Zur Zeit unterstützen aber nicht alle K-Infinity-Schnittstellen die Möglichkeit, für eine Suchanfrage ein Projekt als Startknoten auszuwählen. Daher kann mit der aktuellen Version nur ein Bauprojekt verwaltet werden.

Ebenso ist ein Eingabefeld für eine semantische Volltextsuche vorhanden.



Abbildung 90: Baunetz - semantische Suche

Mit der semantischen Suche können Objekte im Wissensnetz durch Eingabe einfacher Suchbegriffe gefunden werden. Hierbei werden auch Synonyme und mögliche Relationen der Suchbegriffe berücksichtigt. Siehe auch *semantische Suche*.



Abbildung 91: Baunetz - Verlauf

Über die beiden Pfeile kann durch den Verlauf der bisher angezeigten Seiten geblättert werden.

3. Hauptfenster

Das Hauptfenster dient zur Anzeige der ausgewählten Funktionen.

6.4.2 Funktionalität

6.4.2.1 Der Projektbrowser

Der Projektbrowser dient zur hierarchischen Gliederung der Projektbestandteile. Ausgehend vom ausgewählten Projekt, kann in die Tiefe der einzelnen Bauobjekte „hineingetaucht“ werden.



Abbildung 92: Baunetz - Projektbrowser

Die Übersicht ist, je nach Vorhandensein, gegliedert in:

- **Hat Objektbestandteil.** Hierunter werden alle Bestandteile des aktuellen Bauobjektes angezeigt.
- **Hat Objektbeziehung mit** zeigt alle mit dem aktuellen Bauobjekt in Beziehung stehenden Bauobjekte an.
- **Ist Objektbestandteil von** ist die Beziehung zum Übergeordneten Objekt.

Die jeweiligen Teilbereiche sind identisch aufgebaut. Über den Link eines Bauobjektes kommt man direkt zur Projektbrowser-Ansicht diese Bauobjektes.

Der Link *Details* öffnet eine Detailansicht des jeweiligen Bauobjektes, in der alle Eigenschaften und Relationen des Bauobjektes angezeigt werden.



Abbildung 93: Baunetz - Detailansicht

Über die Spalte Vorgang gelangt man zu den Bauvorgängen eines Bauobjektes. Der Link *Direkt* zeigt nur die direkt am Bauteil ausgeführten Vorgänge an. Durch das Anklicken des Links *Alle*, werden auch die Bauvorgänge der Unterobjekte mit angezeigt.

The screenshot shows a web browser window titled 'Baunetz - Microsoft Internet Explorer'. The address bar shows 'http://localhost:3000/Baunetz/html/index2.html'. The main content area displays 'Vorgänge für Haus, Waldstraße 13:' followed by a table. The table has five columns: 'wird ausgeführt', 'Beginn', 'Ende', 'Am Vorgang Beteiligte', and 'Dokumente'. The table lists five construction processes with their respective start and end dates and links to participants and documents. On the left side, there is a navigation menu with sections 'Allgemein' and 'MeinBaunetz' containing various links.

wird ausgeführt	Beginn	Ende	Am Vorgang Beteiligte	Dokumente
Boden auffüllen und verdichten	2002-02-01	2002-02-02	Beteiligte	Dokumente
Bodenaushub	2001-10-05	2001-10-08	Beteiligte	Dokumente
Bodengutachten	2001-09-10	2001-10-01	Beteiligte	Dokumente
Projektmanagement, Haus Waldstr. 73	2001-09-01	2002-05-16	Beteiligte	Dokumente
Statische Berechnung, Haus Waldstraße 13	2001-10-01	2001-10-10	Beteiligte	Dokumente

Abbildung 94: Baunetz - Vorgänge eines Bauobjekts

Die Bauvorgänge werden auch in tabellarischer Form präsentiert. Zu den einzelnen Bauvorgängen wird jeweils das der aktuellsten Version des Bauvorgangs entsprechende Start- bzw. Enddatum angezeigt.

Über den Link *Beteiligte* werden alle am ausgewählten Bauvorgang beteiligte Firmen angezeigt.

Der Link *Dokumente* listet alle den Bauvorgang betreffenden Dokumente auf. Diese Ansicht ist die gleiche wie bei *Alle Dokumente* beschrieben.

Über den Namen des Bauvorgangs gelangt man in die entsprechende Detailansicht des Bauvorgangs.

Bau Kom Online

Allgemein

[Startseite](#)
[Projektbrowser](#)
[Alle Dokumente](#)
[Neues Dokument](#)
[NetNavigator](#)

MeinBaunetz

[Beteiligte Vorgänge](#)
[Eingefügte Dokumente](#)
[Änderungen](#)

Projektwahl: Einfamilienhaus, Waldstraße 13 Suche: Suche

Bodengutachten:

Kurzbeschreibung:

Beschreibung:

[Neue Version](#)

Historie:

Beginn	Ende	Kosten	Eingefügt am	Eingefügt von:	Änderungen	Möglicherweise betroffene Vorgänge/Typ der Änderung
2001-09-10	2001-10-01	10.000,-	2001-09-05	Gussek-Kellerbau	Original	

Fertig Lokales Intranet

Abbildung 95: Baunetz - Detailansicht Bauvorgang

Hier werden alle Details des Bauvorgangs angezeigt. Es ist auch möglich, neue Versionen des Bauvorgangs anzulegen. (S. *Neue Versionen anlegen*)

6.4.2.2 Alle Dokumente

Der Link *Alle Dokumente* bietet eine Übersicht über alle im aktuellen Projekt vorhandenen Dokumente.

Projektwahl: Suche: Suche

Alle Dokumente:

Dokument	Kurzbeschreibung	Historie	Neue Version	Download aktuellste Version
Bauablauf Erdgeschoss.doc	Bauablauf EG	Versionen	Neu	Download/ZIP
Bauablauf Treppenhaus.txt	Bauablauf Treppenhaus	Versionen	Neu	Download/ZIP
Bauplan Erdgeschoss	Bauplan EG	Versionen	Neu	Download/ZIP

Abbildung 96: Baunetz - Übersicht Dokumente

Die gefundenen Dokumente werden in tabellarischer Form präsentiert. Zu jedem Dokument wird die jeweilige Kurzbeschreibung angezeigt. Über die Spalte *Download aktuellste Version* kann sofort die aktuellste Version des Dokumentes heruntergeladen werden. Es gibt zwei Möglichkeiten, die Datei herunterzuladen. Zum einen kann die Datei direkt heruntergeladen und sofort geöffnet werden. Zum anderen existiert die Möglichkeit, die Datei als komprimiertes ZIP-Archiv zu erhalten. Dies verringert die Dateigröße und beschleunigt den Download.

Über den Link *Neue Version* kann eine neue Dokumentversion angelegt werden. (s. *Version anlegen*)

Über die Spalte *Historie* gelangt man zu den verschiedenen Versionen des Dokumentes.

The screenshot shows a web browser window titled 'Baunetz - Microsoft Internet Explorer zur Verfügung gestellt von i-views'. The address bar shows 'http://localhost:3000/html/index2.html'. The page content includes a logo for 'Bau Kom Online' and a navigation menu with links like 'Startseite', 'Projektbrowser', 'Alle Dokumente', and 'Neues Dokument'. The main content area displays 'Projektauswahl: Einfamilienhaus, Waldstraße 13' and 'Suche:'. The title is 'Dokumentversionen für Bauablauf Erdgeschoss.doc:'. Below this is a 'Beschreibung:' section with the text 'Bauablauf für das Erdgeschoss'. The 'Versionen:' section contains a table with the following data:

Eingefügt von	Eingefügt am	Änderungen	Betrifft	Möglicherweise betroffene Vorgänge/Typ der Änderung	Download
HochTief AG	2001-10-01	Änderung an Wand 3	Herstellung Wand 3	Abmessungen verändert	Download/ZIP
HochTief AG	2001-09-27	Original			Download/ZIP

Abbildung 97: Baunetz - Dokumentversionen

Hier können auch die älteren Versionen heruntergeladen werden. Es werden auch die vorgenommenen Änderungen angezeigt, sowie die davon betroffenen Bauvorgänge. Wurde für eine Änderung ein *Typ der Änderung* spezifiziert, gelangt man über den Link zu den möglicherweise von der Änderung betroffenen Bauteilen.

6.4.2.3 Neue Dokumente einfügen

Zu einem Projekt können jeder Zeit neue Dokumente hinzugefügt, und mit den entsprechenden Bauvorgängen verknüpft werden. Hierbei kann zu jedem Dokument eine Beschreibung in Kurz- und Langform angelegt werden.

Da ein großes Bauprojekt aus über 1000 Bauvorgängen bestehen kann, sollte die Auswahl der zu verknüpfenden Bauvorgänge möglichst einfach sein. Daher können beim Einfügen eines neuen Dokumentes mehrere Suchbegriffe angegeben werden. Aufgrund der angegebenen Suchbegriffe wird eine semantische Suche nach Bauvorgängen ausgeführt, und dem Anwender zur Auswahl angezeigt. Dies hat den Vorteil, dass auch nach entsprechenden Synonymen gesucht wird. So werden bei der Eingabe von „Betoniervorgang“ auch Individuen von *Betonierarbeiten* gefunden.

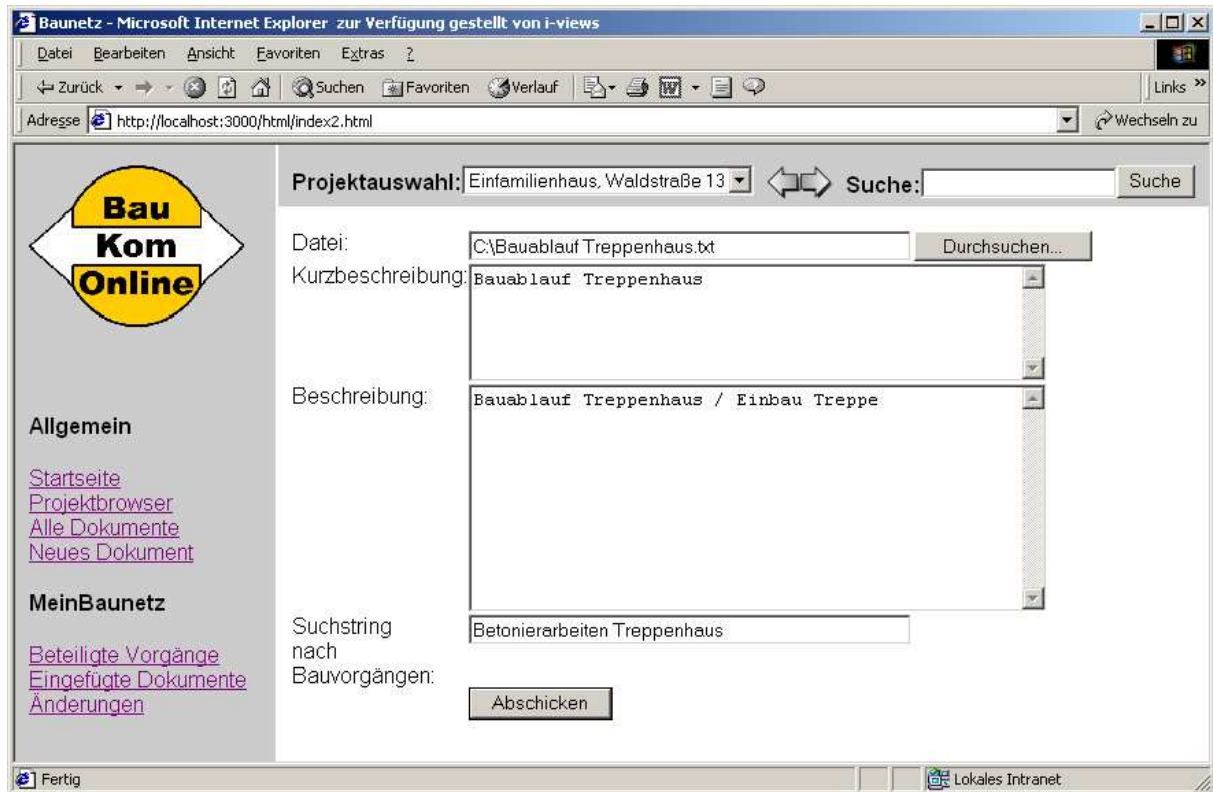


Abbildung 98: Baunetz - Einfügen neuer Dokumente

Nachdem der Button *Abschicken* betätigt wurde, wird im Netz ein neues Individuum des Begriffs *Dokument* angelegt. Zu diesem Dokument wird automatisch eine neue Version angelegt und das aktuelle Datum sowie der Benutzer werden eingetragen. Somit lässt sich jederzeit feststellen, wer welche Dokumente wann eingefügt hat. Danach werden dem Anwender die der Suche entsprechenden Bauvorgänge zur Auswahl angeboten.

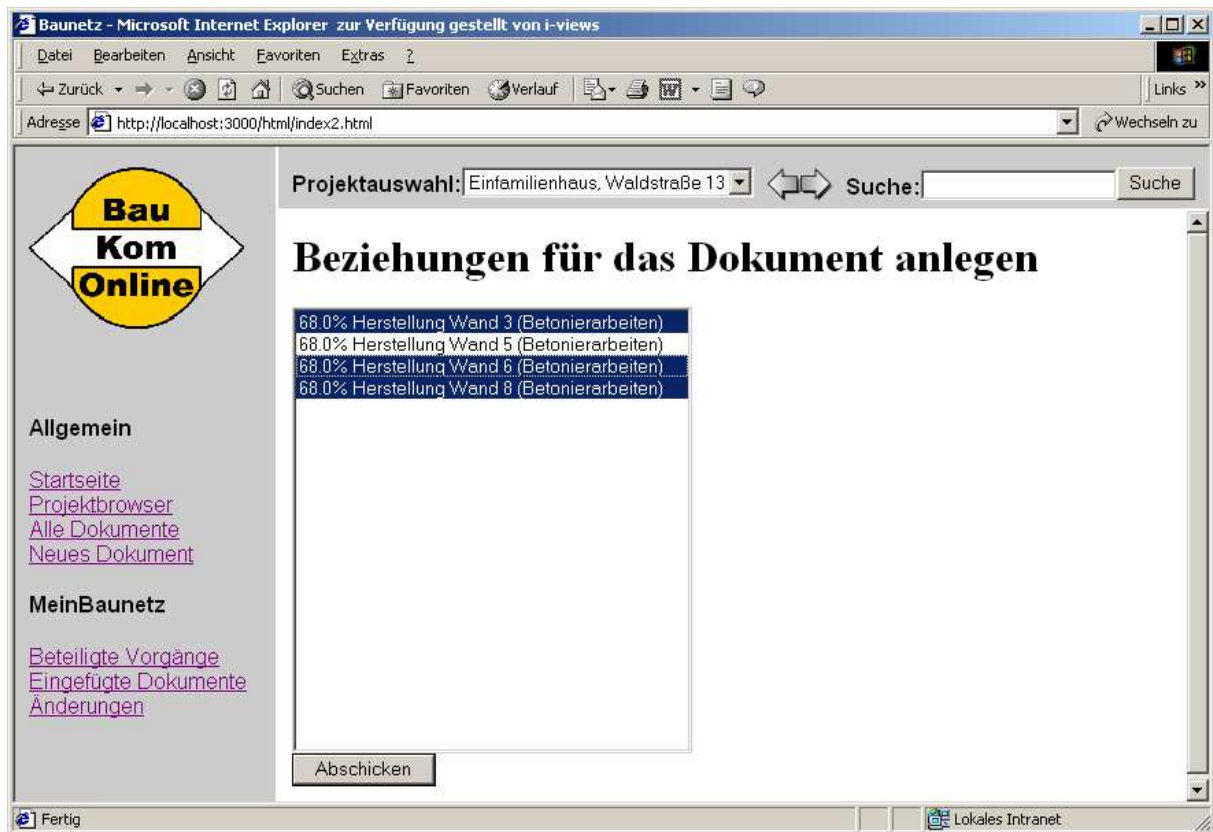


Abbildung 99: Baunetz - Anlegen der Relationen

Zu jedem gefundenen Bauvorgang wird die entsprechende Trefferwahrscheinlichkeit angezeigt. Der Anwender kann nun einen oder mehrere Bauvorgänge auswählen. Durch erneutes Betätigen des Buttons *Abschicken*, werden die ausgewählten Relationen gezogen und der Anwender erhält eine Statusmeldung über das erfolgreiche Einfügen des Dokumentes.

6.4.2.4 Beteiligte Vorgänge

Hier werden alle Bauvorgänge, an denen der aktuell angemeldete Benutzer beteiligt ist, angezeigt. Dies ermöglicht einen schnellen und direkten Zugriff auf die relevanten Informationen des Systems. Diese Ansicht ist identisch mit der Ansicht *Vorgänge eines Bauobjekts*.

6.4.2.5 Eingefügte Dokumente

Über den Link *Eingefügte Dokumente* erhält der Anwender eine Übersicht über alle von ihm eingefügten Dokumente. Hiermit wird das Einfügen neuer Dokumentversionen erleichtert. Die Darstellung entspricht der Ansicht *Dokumente*.

6.4.2.6 Änderungen

Über den Link *Änderungen* kann nach Änderungen, die im ausgewählten Projekt vorgenommen wurden, gesucht werden. Hierfür muss zuerst ein Datum im ISO 8601-Format eingegeben werden, mit dem beginnend nach Änderungen gesucht werden soll.

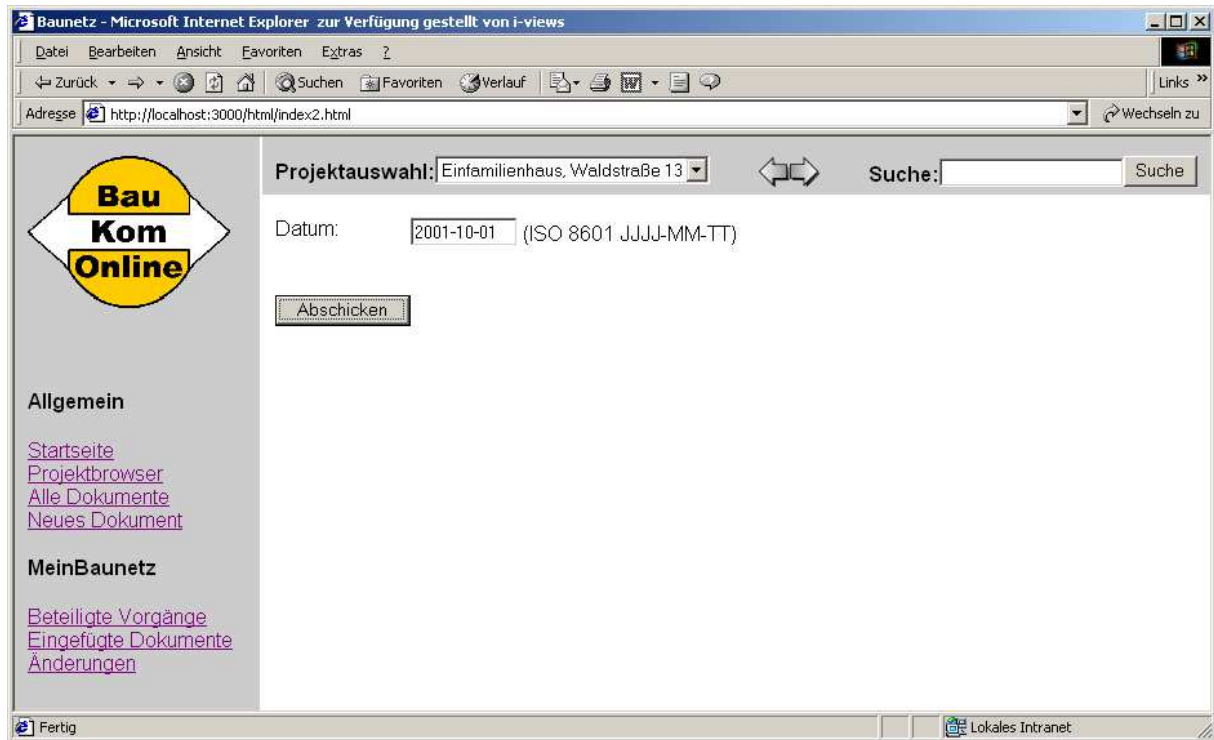


Abbildung 100: Baunetz - Eingabe des Datums

Die gefundenen Änderungen werden getrennt nach geänderten bzw. neu eingefügten Bauvorgängen und Dokumenten dargestellt.

Bau Kom Online

Projektauswahl: Einfamilienhaus, Waldstraße 13 Suche: Suche

Vorgangsänderungen seit dem 2001-10-01:

Vorgang	Änderung	Eingefügt am	Eingefügt von:
Herstellung Wand 4		2001-10-01	HochTief AG

Dokumentänderungen seit dem 2001-10-01:

Dokument	Änderung	Eingefügt am	Eingefügt von:
Bauplan Erdgeschoss		2001-10-06	Bauhof Tresen GmbH
Bauplan Erdgeschoss	Änderungen an Wand 3	2001-10-07	Bauhof Tresen GmbH
Bauablauf Erdgeschoss.doc	Änderung an Tür Wand 3	2001-10-01	HochTief AG

Fertig Lokales Intranet

Abbildung 101: Baunetz - Änderungen

Über den Namen eines Vorgangs bzw. Dokumentes kommt man in die entsprechende Detailansicht.

6.4.2.7 Semantische Suche

Über das Eingabefeld kann, ähnlich einer Suchmaschine im Internet, eine beliebige Suchanfrage gestellt werden. Die semantische Suche bietet aber im Gegensatz zu einer reinen Volltextsuche mehrere Vorteile. Zum einen werden gefundene Begriffe, die miteinander in Beziehung stehen, höher gewertet. Es kann nach Relationen selbst gesucht werden bzw. die Beziehung zweier Begriffe untereinander mitangegeben werden. So kann eine Suchanfrage, ähnlich der natürlichen Sprache, angegeben werden, z.B. "Bauvorgang wird ausgeführt an Wand 3". Findet das System Individuen von Bauvorgang, die auch eine Beziehung „wird ausgeführt an“ mit einem Individuum mit Namen „Wand 3“ haben, so werden diese Individuen höher bewertet.

Ein weiterer Vorteil dieser Suche ist, dass der Anwender nicht die genaue Bezeichnung eines Begriffs kennen muss, denn die semantische Suche berücksichtigt auch alle angelegten Synonyme.

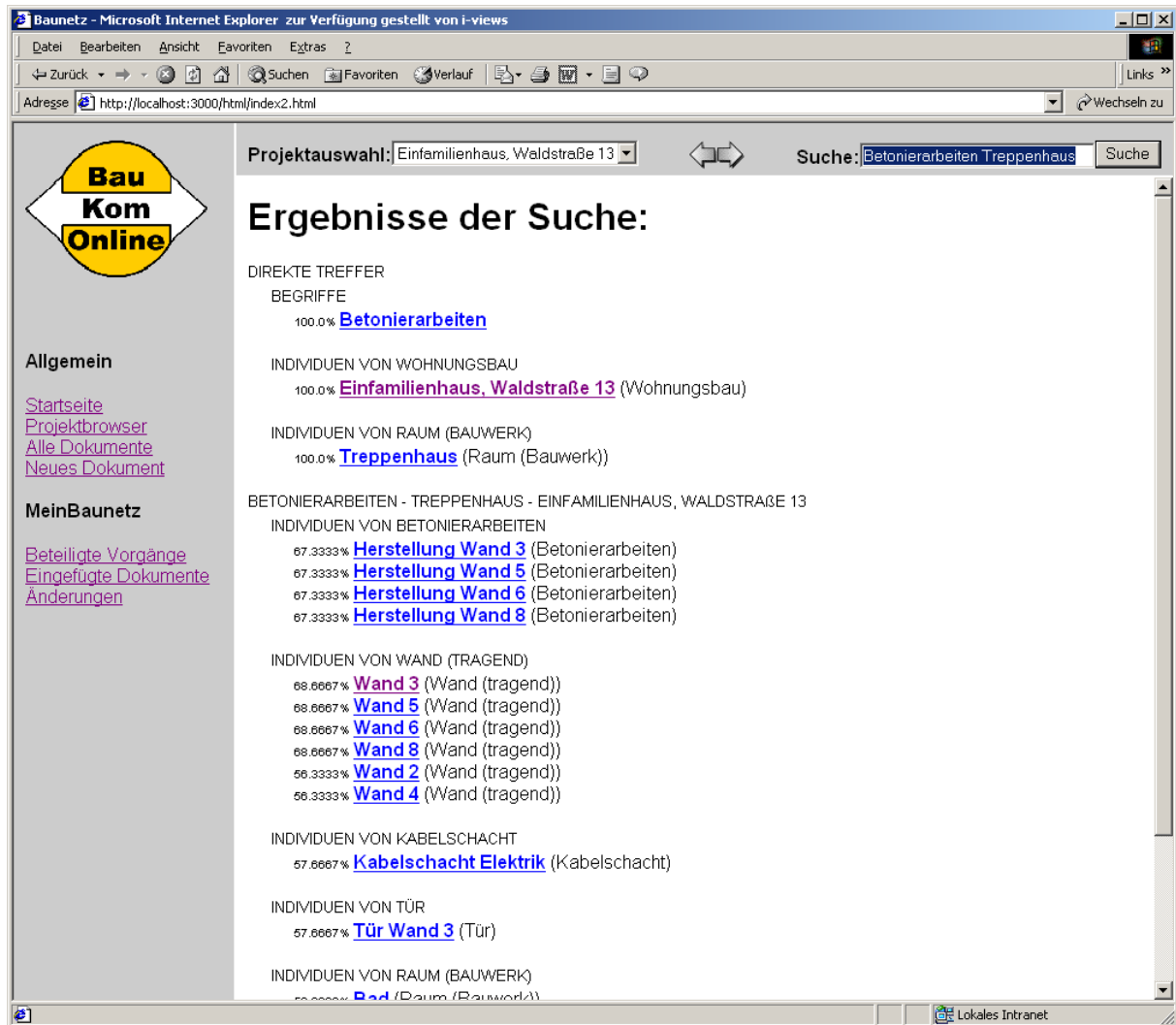


Abbildung 102: Baunetz - Ergebnisse der semantischen Suche

Die gefundenen Ergebnisse werden thematisch, mit Angabe der Trefferwahrscheinlichkeit, dargestellt, d.h. wurden zwei Begriffe oder Individuen, die über eine konkrete Relation in Beziehung stehen, gefunden, wird der Suchpfad, über den die Begriffe gefunden wurden, mit angezeigt. Die gefundenen Ergebnisse können über Links in der *Detailansicht* betrachtet werden.

6.4.2.8 Neue Versionen anlegen

Sowohl zu Dokumenten als auch zu Bauvorgängen können verschiedene Versionen angelegt werden. Beim Anlegen einer neuen Version wird, wie beim Anlegen neuer Dokumente, automatisch der angemeldete Benutzer und das Datum mit der Version gespeichert.

Neue Dokumentversionen

Wird eine neue Dokumentversion angelegt, muss die einzupflegende Datei ausgewählt werden. Danach kann eine Beschreibung der Änderungen als Freitext erfolgen. Existiert für die Art der Änderung ein Typ (*Typ der Änderung*), kann der entsprechende ausgewählt werden. Dieser ermöglicht das Auffinden eventuell von der Änderung betroffener Bauvorgänge.

Da einem Dokument mehrere Bauvorgänge zugeordnet sein können, von der Änderung aber nur einzelne betroffen sind, können die betroffenen Bauvorgänge ausgewählt werden. Um dem Anwender die Auswahl der betroffenen Bauvorgänge zu erleichtern, kann wieder eine Suchanfrage spezifiziert werden.

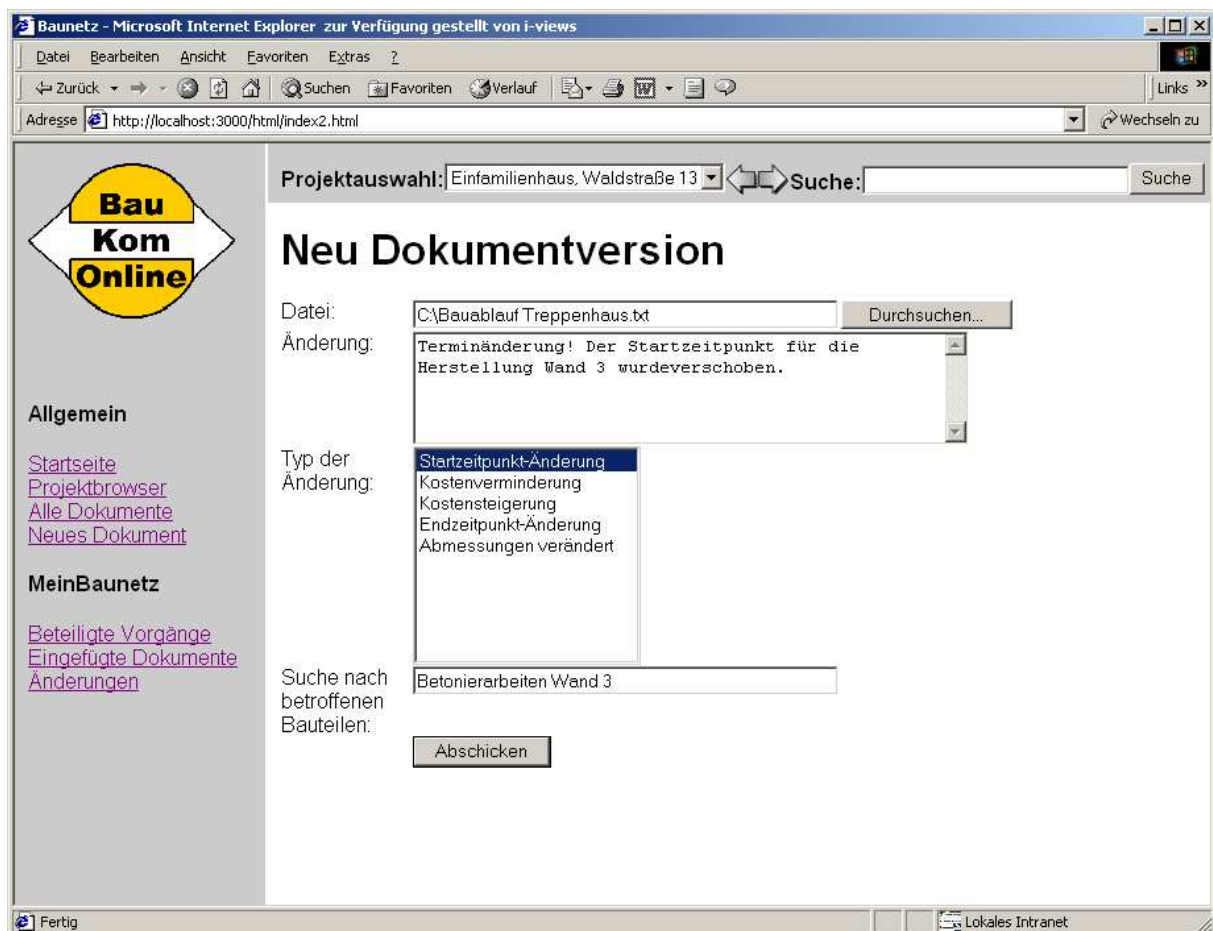


Abbildung 103: Baunetz - Neue Dokumentversion anlegen

Die betroffenen Bauvorgänge können dann wie bei *Ziehen der Relationen* (s. Abbildung 99) ausgewählt und eingetragen werden.

Bauvorgangversionen

Bei einer neuen Bauvorgangsversion können die geänderten Daten, wie Start- bzw. Endzeitpunkt oder Kosten, eingetragen werden. Analog zum Anlegen einer Dokumentversion können eine freitextliche Beschreibung und der *Typ der Änderung* angegeben werden.

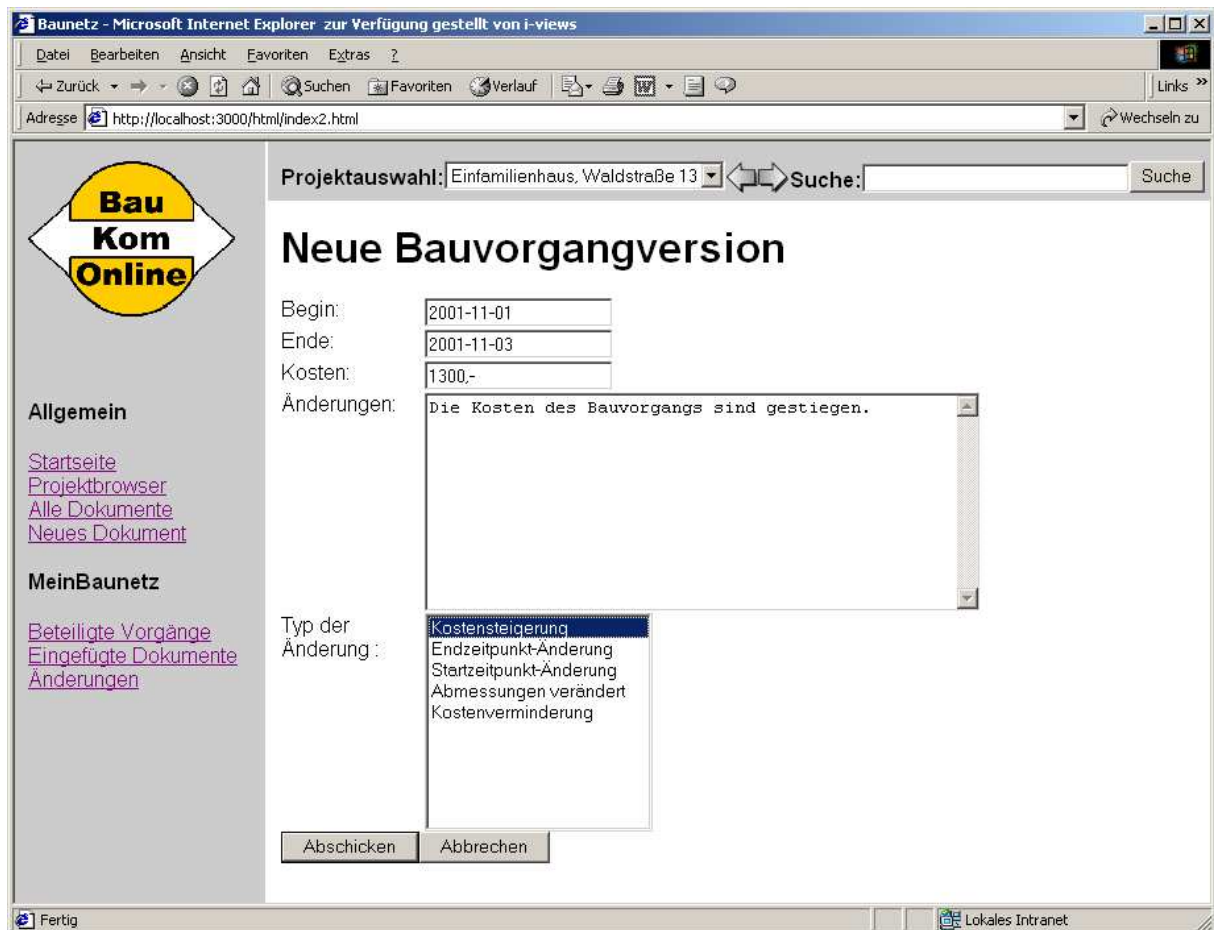


Abbildung 104: Baunetz - Neue Bauvorgangversion anlegen

7 Zusammenfassung und Ausblick

Die vorliegende Arbeit zeigt, dass der Einsatz von Wissensmanagementsystemen die Abbildung von Bauplanungs- und Ausführungsprozessen in Bauprojekten erheblich vereinfachen kann. Basierend auf einem allgemeinen, bauprojektübergreifenden Grundnetz, lassen sich beliebige Bauprojekte verwalten. Aufgrund der dynamischen Struktur, die durch frei definierbare Relationen und den Vererbungsmechanismus von Objekten erreicht wird, lässt sich das System auf beliebige Bauvorhaben anwenden und gegebenenfalls an neue Aufgabenstellungen anpassen.

Zunächst muss ein Bauprojekt in seine kleinsten Bestandteile zerlegt werden, d.h. die einzelnen Bauobjekte, wie Räume oder Bauteile, und die einzelnen Bauprozesse müssen erkannt und, basierend auf dem Grundnetz, im System abgebildet werden. Was schließlich als kleinste Einheit angesehen wird, hängt von der gewünschten Funktionalität und der Genauigkeit des Netzes ab. Dieser Mehraufwand wird aber durch die Entlastung der einzelnen Projektbeteiligten bei der Eingabe und Suche, aufgehoben. Dies ist der Hauptvorteil von Wissensmanagementsystemen gegenüber herkömmlichen Ansätzen. Da bei jedem neuen Bauprojekt Firmen, die zuvor noch nie mit diesem System gearbeitet haben, hinzukommen können, muss die Einarbeitungszeit in ein Projektverwaltungssystem so gering wie möglich sein. Das System sollte aber so komplex wie möglich sein, um die einzelnen Zusammenhänge der verschiedenen Projektbestandteile abbilden und dadurch Auswirkungen von Änderungen auf andere Projektbestandteile sichtbar zu machen.

Wie das Anwendungsbeispiel gezeigt hat, wird der Anwender bei Anlegen neuer Dokumente dahingehend entlastet, dass er bei der Verknüpfung des Dokumentes mit den vom Dokument betroffenen Bauvorgängen durch eine semantische Volltextsuche unterstützt wird. Die semantische Volltextsuche erlaubt es dem Anwender, relativ einfach zu definieren, welche Vorgänge er sucht. Da das System zu den Begriffen die entsprechenden Synonyme kennt, muss der Anwender nicht mehrere Bedeutungen für ein Wort mit Booleschen Operatoren verknüpfen. Ebenso kann das System Relationen zwischen mehreren Suchbegriffen erkennen und auswerten.

Ein weiterer Vorteil eines Wissensnetzes ist, dass es anhand der semantischen Beziehungen der Objekte untereinander, Beziehungen von Objekten erkennen kann, die dem Anwender nicht auf Anhieb ersichtlich sind. Dies kann, wie die konfigurierbare Suche gezeigt hat, dazu genutzt werden, mögliche Auswirkungen von Änderungen an einem Objekt auf ein anderes Objekt zu erkennen.

Durch die objektorientierten Mechanismen, wie Vererbung und Abstraktion von Objekten, ist es möglich, die Projektbestandteile während der Laufzeit eines Bauprojektes dynamisch zu ändern. Es können jederzeit, ohne den Code der Anwendung zu ändern, beliebige Begriffe und Relationen angelegt werden, um das System entsprechend zu erweitern bzw. zu verfeinern.

Um den Anwender von einer detaillierten Kenntnis des Systems zu entlasten, ist es denkbar, dass das System automatisch Dokumente jeglicher Form selbstständig analysiert und

klassifiziert. Bei der automatischen Analyse von Dokumenten könnte das Prinzip der neuronalen Netze eine große Rolle spielen, da hier sehr gute Ergebnisse mit unscharfen Dokumenten erreicht werden. Gerade in der Analyse von Zeichnungen und Bauplänen, die keinem festen Schema folgen, könnte der Vorteil der neuronalen Netze liegen, die versuchen, entsprechende Bauteile aus einem Dokument zu extrahieren.

Ebenso ist das automatische Erstellen eines neuen Bauprojektnetzes, aufgrund bereits bestehender Dokumente, möglich. So könnten die ersten Entwurfspläne eines Bauprojektes automatisch analysiert und in das Wissensnetz übertragen werden. Dies würde den organisatorischen Aufwand bei Erstellen eines Projektnetzes erheblich vereinfachen, da hierfür kein spezieller „Wissensarbeiter“ nötig wäre, der sich mit der Struktur des Netzes auskennt.

Die Lernfähigkeit eines Netzes könnte auch dazu genutzt werden, sich an den jeweiligen Anwender anzupassen. So könnte sich das System an spezielle Eigenarten eines Anwenders bei der Dokumentation anpassen, und somit bei der Klassifizierung bessere Ergebnisse erzielen.

8 Anhang

8.1 Installation

8.1.1 Softwarevoraussetzungen zur Installation

- Eine installierte JAVA Runtime, ab Version 1.2 (www.java.sun.com). Die Installation der JAVA-Runtime wird hier nicht näher erläutert, da sie über eine eigene Installerroutine verfügt.
- XERCES-Parser (<http://xml.apache.org/xerces2-j>).
- XALAN-XSLT-Prozessor (<http://xml.apache.org/xalan-j>).
- Das *com.oreilly.servlet* Paket (<http://www.servlets.com/cos>).
- K-Infinity Server.
- JAVA-Bridge.

8.1.2 Installation Jakarta-Tomcat 3.2.3

1. Die ZIP-Datei *jakarta-tomcat-3.2.3.zip* in ein beliebiges Verzeichnis entpacken. Hierbei wird in diesem Verzeichnis ein Verzeichnis *jakarta-tomcat-3.2.3* erstellt.
2. Setzen der Umgebungsvariablen *TOMCAT_HOME=Verzeichnis\jakarta-tomcat-3.2.3*
3. Setzen der Umgebungsvariablen *JAVA_HOME*, die in das Installationsverzeichnis der JAVA-Runtime zeigt.

8.1.3 Installation Baunetz Servlet

1. In das *%TOMCAT_HOME%/webapps*-Verzeichnis das Verzeichnis Baunetz aus dem Ordner *Baunetz-Servlet* kopieren. Hierbei werden die eigentlichen Servlet-Klassen und die XSL- bzw. HTML-Dateien angelegt.
2. Die drei im *Tomcat_Lib*-Verzeichnis enthaltenen JAR-Dateien in das *%TOMCAT_HOME%/lib*-Verzeichnis kopieren. Um Konflikte mit den im Tomcat enthaltenen Parser-Klassen zu umgehen, sind die XERCES- und XALAN-JAR-Dateien umbenannt worden, da Tomcat die Klassen in alphabetischer Reihenfolge lädt.
3. Anpassen der *Server.xml* Datei. Dem Element *ContextManager* ein neues Kind-Element hinzufügen:

```
<Context path="/Baunetz" docBase="webapps/Baunetz" crossContext="true" debug="0" reloadable="true"
trusted="false"/>
```

4. In der *bn.properties* Datei muss die Standardadresse der JAVA-Bridge angegeben werden. Dies geschieht durch den Eintrag:

defaultserviceurl=http://Rechnername:Port/

8.1.4 Installation Wissensnetz

Hierzu muss das Verzeichnis *Baunetz* aus dem Ordner *Wissensnetz* in das *volume*-Verzeichnis des K-Infinity Servers kopiert werden.

8.1.5 Starten der Anwendung

Im *%TOMCAT_HOME%/bin* Verzeichnis die Datei *startup.bat* aufrufen. Dies startet den Tomcat-Server. Das Baunetz kann dann unter *http://Computername/Baunetz/html* aufgerufen werden.

8.2 Abbildungsverzeichnis

Abbildung 1: Unterschied konventionelle Programme zu wissensbasierten Systemen.....	8
Abbildung 2: Darstellung der Wissensrepräsentationsarten nach [HH]	9
Abbildung 3: Beispiel semantisches Netz.....	11
Abbildung 4: O-A-W-Tripel - semantisches Netz	12
Abbildung 5: Darstellung Objekt-Attribut-Wert-Tripel.....	12
Abbildung 6: generischer Frame – Instanz.....	12
Abbildung 7: Frame - semantisches Netz	13
Abbildung 8: Beispiel Topic Map	15
Abbildung 9: Biologisches Neuron	16
Abbildung 10: Künstliches Neuron [EINN].....	17
Abbildung 11: Schichten eines neuronalen Netzes [EINN].....	19
Abbildung 12: Feedforward Netz.....	19
Abbildung 13: Feedforward-Netz mit "Shortcut"	20
Abbildung 14: Gegenüberstellung symbolischer und subsymbolischer Wissensverarbeitung	22
Abbildung 15: Gegenüberstellung symbolischer und subsymbolischer Ansätze nach [HWPD]	22
.....	22
Abbildung 16: USU KnowledgeMiner [USU].....	27
Abbildung 17: KnowledgeMiner - Topic Map	28
Abbildung 18: SER Brainware – Layer [SER]	29
Abbildung 19: SERbrainware - Lernen und Klassifizieren	30
Abbildung 20: SERbrainware – Lernzeit [SER]	30
Abbildung 21: Definition - Begriff	36
Abbildung 22: Definition - Rolle	36
Abbildung 23: Definition - Individuen.....	36
Abbildung 24: Definition - Super/Sub Relation.....	36
Abbildung 25: Definition - Relationen.....	37
Abbildung 26: Definition - Ist auch Relation.....	37
Abbildung 27: Grundprinzipien - Ober-/Unterbegriff	38
Abbildung 28: Grundprinzipien - Gruppierung von Objekten (1)	39
Abbildung 29: Grundprinzipien - Gruppierung von Objekten (2)	39
Abbildung 30: Grundprinzipien - Abstraktionsgrad der Begriffe (1)	40
Abbildung 31: Grundprinzipien - Abstraktionsgrad der Begriffe (2)	40
Abbildung 32: Grundprinzipien - Mehrfachvererbung	41
Abbildung 33: Grundprinzipien – Rollenerweiterung (1).....	42
Abbildung 34: Grundprinzipien - Rollenerweiterung (2)	42
Abbildung 35: Grundprinzipien - Individuenfähige Begriffe	42
Abbildung 36: Grundnetz - T	46
Abbildung 37: Baugrundnetz - Bauprojekt	47
Abbildung 38: Baugrundnetz - Bauvorgang	48
Abbildung 39: Baugrundnetz – Unterbegriffe von Bauvorgang (1)	50
Abbildung 40: Baugrundnetz – Unterbegriffe von Bauvorgang (2)	51
Abbildung 41: Baugrundnetz – Baunormen.....	52
Abbildung 42: Baugrundnetz – Leistungsphasen (HOAI).....	53
Abbildung 43: Baugrundnetz - Kostengliederung nach DIN 276.....	55
Abbildung 44: Baugrundnetz - Projektbeteiligter	56
Abbildung 45: Baugrundnetz - Projektbeteiligter (Relationen)	57
Abbildung 46: Baugrundnetz - Bauobjekt	58
Abbildung 47: Baugrundnetz - Bauteil	60

Abbildung 48: Baugrundnetz - Dokument	61
Abbildung 49: Baugrundnetz - Version	62
Abbildung 50: Baugrundnetz - Beispiel Dokumentversionen	64
Abbildung 51: Baugrundnetz - Beispiel Bauvorgangversion.....	65
Abbildung 52: Baugrundnetz - Änderungen	66
Abbildung 53: Baunetz - Modellierungsbeispiel	67
Abbildung 54: K-Infinity – Admin-Tool	68
Abbildung 55: K-Infinity - Organizer	69
Abbildung 56: K-Infinity - Neuen Begriff anlegen.....	69
Abbildung 57: K-Infinity – Begriffseditor (1)	70
Abbildung 58: K-Infinity - Schemadefinition Individuum	71
Abbildung 59: K-Infinity - Attributtyp	71
Abbildung 60: K-Infinity - Neues Attribut anlegen	72
Abbildung 61: K-Infinity - Begriffseditor (2)	72
Abbildung 62: K-Infinity - Neue Relation anlegen.....	73
Abbildung 63: K-Infinity - Schemadefinition Individuum (2).....	74
Abbildung 64: K-Infinity - Relationsbegriff-Editor	74
Abbildung 65: K-Infinity - Neue Relation anlegen (2)	75
Abbildung 66: K-Infinity - Begriffseditor (3)	76
Abbildung 67: K-Infinity - Neue Rollenerweiterung anlegen	76
Abbildung 68: K-Infinity - Schemadefinition Individuum (3).....	77
Abbildung 69: K-Infinity - Graph-Editor	78
Abbildung 70: K-Infinity - Neues Individuum anlegen	78
Abbildung 71: K-Infinity - Graph-Editor (2)	79
Abbildung 72: K-Infinity - "Anfasser "	79
Abbildung 73: K-Infinity - Graph-Editor (3)	80
Abbildung 74: K-Infinity - Semantische Suche (1)	81
Abbildung 75: K-Infinity - Semantische Suche (2)	81
Abbildung 76: Gesamtkonzept Baunetz.....	84
Abbildung 77: Servlet - Übersicht Paket com.iviews.bn	85
Abbildung 78: Servlet - BNServlet	86
Abbildung 79: Servlet - XSLTEngine.....	86
Abbildung 80: Servlet - Paket com.iviews.bn.services	87
Abbildung 81: Servlet -Paket com.iviews.bn.bridge.....	87
Abbildung 82: Servlet -Paket com.iviews.bn.security	88
Abbildung 83: Servlet - Anmeldevorgang	89
Abbildung 84: Servlet - Dynamische Seitengenerierung	90
Abbildung 85: Servlet - Dateiupload	91
Abbildung 86: Servlet - XSLT-Prozessor	92
Abbildung 87: Baunetz – Konzeption der Internetseiten	93
Abbildung 88: Baunetz - Frames	94
Abbildung 89: Baunetz - Projektauswahl.....	94
Abbildung 90: Baunetz - semantische Suche	95
Abbildung 91: Baunetz - Verlauf	95
Abbildung 92: Baunetz - Projektbrowser	96
Abbildung 93: Baunetz - Detailansicht	97
Abbildung 94: Baunetz - Vorgänge eines Bauobjekts	98
Abbildung 95: Baunetz - Detailansicht Bauvorgang.....	99
Abbildung 96: Baunetz - Übersicht Dokumente	100
Abbildung 97: Baunetz - Dokumentversionen.....	101
Abbildung 98: Baunetz - Einfügen neuer Dokumente	102

Abbildung 99: Baunetz - Anlegen der Relationen	103
Abbildung 100: Baunetz - Eingabe des Datums	104
Abbildung 101: Baunetz - Änderungen	105
Abbildung 102: Baunetz - Ergebnisse der semantischen Suche	106
Abbildung 103: Baunetz - Neue Dokumentversion anlegen.....	107
Abbildung 104: Baunetz - Neue Bauvorgangversion anlegen.....	108

8.3 Literaturverzeichnis

- [EMAXPS] Die grafische Expertensystemschale EMA-XPS (<http://www.iai.uni-wuppertal.de/EMA-XPS/de/pguide.html>)
- [DVW] Darstellung von Wissen (<http://www.fb10.uni-bremen.de/linguistik/khwagner/semantik/wissen.htm>)
- [HH] **Helbig, Hermann:** Die semantische Struktur natürlicher Sprache: Wissensrepräsentation mit MultiNet, Berlin, Heidelberg, 2001
- [TM] Topic Maps (<http://linux.exit.mytoday.de/srmurphy/daten/topicmaps.pdf>)
- [ISOTM] ISO/IEC 13250 Topic Maps (<http://www.y12.doe.gov/sgml/sc34/document/0129.pdf>)
- [EINN] Script zu "Einführung in neuronale Netze" (<http://www.math.uni-muenster.de/SoftComputing/lehre/material/nnscrip/>)
- [HWPD] **Andreas Koller:** „Hybride Wissensverarbeitung in der präventivmedizinischen Diagnostik“ Paderborn 1999
- [TIK] **Devlin, K.:** Infosense. Turning Information into Knowledge. New York Freeman, 1999
- [GKA] Graphentheoretische Konzepte und Algorithmen (http://www-info1.informatik.uni-wuerzburg.de/de/lehre/2001_ss/graphentheorie/vorlesung.pdf)
- [RU] **Reimer, Ulrich:** Einführung in die Wissensrepräsentation. Leitfaden der Informatik, Stuttgart, 1991
- [USU] USU AG (<http://www.usu.de>)
- [SER] SER Systems AG (<http://www.ser.de>)

9 Erklärung zur Diplomarbeit gemäß § 19 Abs. 6 DPO/AT

Hiermit versichere ich, die vorliegende Diplomarbeit ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus den Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den _____

(Unterschrift)