

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/200505703>

Teamplanung mit Semantischen Webtechnologien

Thesis · April 2009

CITATIONS

0

READS

272

1 author:



Susanne Gottwald

Zuse-Institut Berlin

2 PUBLICATIONS 0 CITATIONS

SEE PROFILE

Diplomarbeit

Teamplanung mit semantischen Webtechnologien

Susanne Richter

betreut durch:

Prof. Dr. Robert Tolksdorf
Freie Universität Berlin, Institut für Informatik

Dr. Hans-Christoph Kürn
Siemens AG, München

Freie Universität Berlin
Fachbereich Mathematik und Informatik
Institut für Informatik

Berlin, 2. März 2009

Ehrenwörtliche Erklärung

Hiermit versichere ich, dass ich die vorliegende Diplomarbeit ohne Hilfe Dritter und ohne Zuhilfenahme anderer als der angegebenen Quellen und Hilfsmittel angefertigt habe. Die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Berlin, 2. März 2009

Inhaltsverzeichnis

1. Einleitung	5
1.1. Motivation	6
1.2. Zielsetzung	8
1.3. Gliederung	9
2. Grundlagen	10
2.1. Human Resources und Skills	10
2.1.1. Recruitment	11
2.1.2. Kompetenzen, Skills, Fähigkeiten	13
2.2. Teams und Teamplanung	16
2.3. Semantic Web	19
2.3.1. Definition und Eigenschaften	20
2.3.2. Ontologien	22
3. Problemanalyse und Anforderungsdefinition	30
3.1. Verwandte Arbeiten	32
3.1.1. Knowledge Management System, AIAI	32
3.1.2. Skillmanagement-System, BEKO	38
3.1.3. KMDL, Universität Potsdam	43
3.1.4. KOWIEN, Universität Duisburg-Essen	51
3.1.5. ePeople-System, DaimlerCrysler AG	55
3.2. Anforderungen	62
3.3. Anwendungsfälle	66
4. Konzeption und Entwurf	74
4.1. Human Resource Ontologie	74
4.2. Kriterien eines optimalen Teams	78
4.3. Einschränkungen der Auswahlmenge	86
4.4. Teamplanungsalgorithmen	87
4.4.1. Permutativer Ansatz	90
4.4.2. Erster Semantischer Ansatz	92
4.4.3. Zweiter Semantischer Ansatz	95
4.4.4. Dritter Semantischer Ansatz	99
4.4.5. Teamevaluation	102

5. Implementierung	108
5.1. Technische Realisierung	108
5.1.1. ServerApplication	109
5.1.2. ClientApplication	114
5.1.3. TeamplanningServer	116
5.1.4. Anwendungsfälle	118
5.1.5. Komplexität und Skalierbarkeit	121
5.2. Test	123
5.2.1. Vorbetrachtung	123
5.2.2. Eigenschaften der Testfälle	124
5.2.3. Untersuchung der Testergebnisse	125
6. Zusammenfassung und Ausblick	131
6.1. Bewertung der Ergebnisse	132
6.2. Ausblick	134
Abbildungsverzeichnis	136
Literaturverzeichnis	137
Glossar	144
A. Inhalt der CD	146

1. Einleitung

Als Tim-Berners Lee 2001 in der Zeitschrift „Scientific American“ das Semantic Web mit dem Artikel „The Semantic Web - A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities“ [BL01] erstmals erwähnte, war noch nicht abzusehen, dass wenige Jahre danach „Semantik in sein“ würde. So sehen es jedenfalls die Autoren der Studie „Semantische Technologien in der betrieblichen Anwendung“ [Joh06]. Mit der Bewegung des Semantic Web steht erstmals die umfassende semantische Integration der gesamten Datenbestände und Anwendungen auf der Agenda von Unternehmen. Ergebnisse der Studie zeigen, dass der Einsatz von semantischen Technologien im Unternehmen bei richtiger Implementierung in die Geschäftsprozesse Effizienzgewinne von bis zu 50% mit sich bringt. Dabei zeichnet sich ab, dass kleinere Einzelanwendungen, die einen zentralen Geschäfts- oder Informationsprozess unterstützen, effizienter sind und weit eher die gewünschten Resultate erbringen, als eine umfassende semantische Integration aller Datenbestände. Unter diesem Aspekt soll im Rahmen der vorliegenden Arbeit eine Applikation entwickelt werden, die Unternehmen bei der Planung für Projektteams mithilfe von semantischen Webtechnologien unterstützen kann.

Die Siemens AG steht dieser Arbeit als Wissensträger und Unternehmenshintergrund zur Seite. Dr. Hans-Christoph Kürn, Leiter der Abteilung e-Recruiting Siemens Deutschland, weiß durch seine Arbeit, wie informationsintensiv und aufwendig die Suche nach geeigneten Kandidaten für freie Arbeitsstellen ist und dass ein positiver Abgleich zwischen Stellen- und Bewerberprofilen nicht trivialerweise entsteht. Ein ähnliches Problem besteht bei der Zusammenstellung von Teams für Projekte, die ein spezifisches Aufgabenziel haben. Mitarbeiter eines Unternehmens müssen sich derart geeignet zu einem Team gruppieren, so dass alle notwendigen Kenntnisse zur Lösung des Problems im Team vorkommen. Eine auf semantischen Technologien basierende Anwendung kann den Auswahlprozess der Mitarbeiter

analysieren und so den Teamrecruiter, also die Person im Unternehmen, die Mitarbeiter für ein Team auswählt, bei seiner Arbeit unterstützen. Die Siemens AG beschäftigt Teamrecruiter, die sich hauptsächlich mit der passenden Vermittlung von Personen für Projektteams oder Abteilungen beschäftigen. Die im Rahmen dieser Arbeit entstehende prototypische Applikation soll deren effizientere Arbeitsweise ermöglichen.

1.1. Motivation

Teams und Teamarbeit sind aus Unternehmen als koordinierte Arbeitsmethode nicht mehr wegzudenken. West [Wes03] nennt Gründe dafür, Teamarbeit in Unternehmen einzusetzen: „Teams are the best way to enact organizational strategy [...] Team-based organizations with their flat structures can respond quickly and effectively in the fast-changing environments most organizations now encounter.“ Weitere Gründe seien, dass Teams in der richtigen Zusammensetzung Kreativität fördern, Innovation hervorbringen und Produkte schnell und kostengünstig entwickeln und ausliefern.

Für die Teambildung bzw. das Teambuilding existieren diverse theoretische Ansätze, wobei hier als populärer Vertreter das Phasenmodell "Forming - Storming - Norming - Performing" genannt sei. Die 4 Stufen der Teamentwicklung von Bruce Wayne Tuckman aus „Developmental Sequence in Small Groups“ (1965) wurden 1977 um eine fünfte Phase, dem Adjourning, ergänzt. Das Phasenmodell beschreibt, wie ein Team sich entwickelt. In der ersten Phase bildet es sich, in Phase zwei steht es Konfrontationen unter den Mitgliedern gegenüber, in der dritten Phase beginnt das Team zu kooperieren, in Phase vier bringt es Leistungen und Lösungen hervor und schließlich zerfällt es in der letzten Phase nach Abschluss der Arbeit wieder. Laut Tuckman sind alle Phasen notwendig und unumgänglich, damit das Team wachsen, sich Herausforderungen stellen, Probleme bewältigen, Lösungen finden, Arbeit planen und Ergebnisse liefern kann. Um Teams effizient und vor allem schnell zu bilden, werden Mitarbeiter von Unternehmen auf Teamtraining-Veranstaltungen oder Teambuilding-Kursen geschult. Dabei bezeichnet Teambuilding die Phasen und Strukturen, um aus einer bereits existierenden Gruppe von Mitarbeitern ein funktionierendes Team zu machen. Teambuilding beginnt also erst dann, wenn bereits bestimmt wurde, wer im Team mitwirkt. Für die in der vorlie-

genden Arbeit unterstützten Prozesse wurde daher bewusst der Begriff Teamplanung verwendet, um vom bekannten Teambuilding abzugrenzen, da die zu entwickelnde Applikation anhand einer Projektbeschreibung Mitarbeiter für das Team selektieren soll, wodurch der spätere Teambuilding-Prozess mittels einer effizienteren Zusammensetzung unterstützt werden kann. Teamplanung nimmt somit in der Einordnung in einen Projektzeitplan einen frühen Platz nach der Projektdefinition ein. Deiters [DL00] schreibt, dass vor allem die zunehmende Bedeutung der

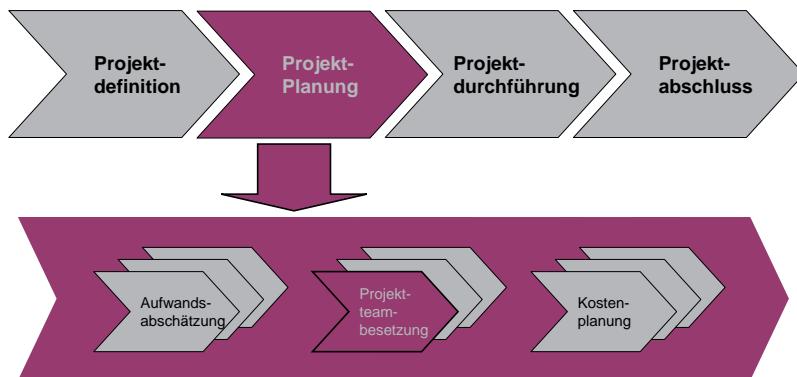


Abbildung 1.1.: Teamplanung im Projektzeitplan, angelehnt an [Bie06]

Projektarbeiten und der damit verbundene Einsatz flexibler Teams eine verstärkte informationstechnische Verwaltung dieser Ressourcen verlangen. Konsequentes Skillmanagement würde sowohl den Teambildungsprozess, den Wissenstransfer als auch die Weiterbildungsplanung innerhalb der Unternehmen unterstützen. In dieser Arbeit werden deshalb im Besonderen die Projektteams betrachtet, unter denen zeitlich begrenzte Arbeitsgruppen verstanden werden, die für die Bearbeitung eines Problems zusammengestellt werden und sich nach Erledigung des Projektes wieder auflösen. Die Teammitglieder stammen aus dem Unternehmen oder sind, wenn in Ausnahmefällen mit den bestehenden Mitarbeitern kein Team gebildet werden kann, Externe, die nach Projektabschluss das Unternehmen wieder verlassen bzw. als neue Mitarbeiter dort verbleiben. Teamrecruiter oder häufig auch die späteren Projektteamleiter beschäftigen sich in Unternehmen damit, geeignete Mitarbeiter für ein spezifisches Projekt zu finden. Die Grundlage hierfür bildet Kompetenzmanagement, das in großen Unternehmen mit wechselnden Mitarbeitern eine ständige Aktualität des Wissens über das Wissen der anderen erfordert. Der Teamrecruiter muss die Kenntnis darüber haben, welcher Mitarbeiter zu welchem Zeitpunkt welche Aufgabe wahrnimmt, welche Grundkenntnisse jeder besitzt und welches Spezialwissen erlangt wurde. Biesalski schreibt in [Bie06] dazu,

dass die Besetzung von vakanten Stelle in Projekten selten nach einer strukturierteren Vorgehensweise gewählt wird sondern vielmehr auf persönlichen Netzwerken und Vorlieben beruht. Dabei steigt das Risiko, dass fachliche Kriterien unbeachtet bleiben und geeignete Kandidaten übersehen werden. „Weiter steigen das zeitliche Risiko (rechtzeitiges Finden) mit der Anzahl der benötigten Kandidaten für Projektstellen stark an. Dieser Zustand ist weder gerecht, noch transparent und führt in aller Regel [...] zu einer suboptimalen Besetzung von Projektstellen.“ Herbst [Her00] definiert Wissensmanagement als „ein komplexes strategisches Führungs-konzept, mit dem ein Unternehmen sein relevantes Wissen ganzheitlich ziel- und zukunftsorientiert als wertsteigernde Ressource gestaltet. Die Wissensbasis aus individuellem und kollektivem Wissen wird bewusst, aktiv und systematisch ent-wickelt, sodass sie zum Erreichen der Firmenziele beiträgt.“ Selbst wenn ein Unternehmen ein solches Wissensmanagement-System besitzt und anwendet, bleibt es für den Teamrecruiter schwierig, bei allen Projekten verschiedener thematischer Domänen die essentiellen notwendigen Kenntnisse zu erkennen und Mitarbeiter, die jene Bedingungen am besten erfüllen, für diese Aufgaben innerhalb des Teams einzuteilen. Erst unter diesen Voraussetzungen kann ein Team optimal und effizi-ent arbeiten. Welche Kriterien für optimale Teams existieren, wird in Abschnitt 4.2 beschrieben.

1.2. Zielsetzung

Ziel dieser Diplomarbeit ist die Entwicklung einer prototypische Applikation, die einen Teamrecruiter eines Unternehmens dabei unterstützen soll, möglichst opti-male Teams zusammenzusetzen. Optimal bedeutet in einer ersten Annäherung, dass ein Team mit möglichst geringen personellen Ressourcen gebildet wird, das in einem vorgegebenen Zeitraum eine Lösung für das im Projekt zu behandelnde Problem entwickeln soll. Ein zeitnahe Arbeitsbeginn nach Projektdefinition und eine geeignete Zusammensetzung können hierbei Zeit und Kosten sparen, da nicht eine einzige Person aufwendig nach Teammitgliedern suchen muss. Dadurch wird eine effizientere Zusammensetzung anhand der Kenntnisse der einzelnen Team-mitglieder wahrscheinlicher. Das Unternehmen benötigt zur Realisierung ledig-lich Mitarbeiter- und Projektbeschreibungen, die auf Konzepten einer Human Re-source Ontologie (HR-Ontologie) basieren. Die vorliegende Ausarbeitung kann ei-

ne bereits entwickelte Ontologie der Ontonym GmbH Berlin nutzen. Eine weitere Grundlage bildet die im Rahmen dieser Arbeit entwickelten Teamplanungsalgorithmen, die in verschiedenen Ausprägungen die diversen Kriterien für optimale Teams berücksichtigen und aus der Menge der Mitarbeiter jene selektieren, die als Gruppe die Projektbeschreibung erfüllen. Dabei sollen allerdings ausschließlich fachliche Qualifikationen in eine Berechnung einfließen. Die Erfassung und Berechnung weiterer persönlicher Informationen der Mitarbeiter wie z.B. „soft skills“ (siehe Abschnitt 2.1.2) liegen außerhalb des Rahmens der vorliegenden Arbeit.

Im Ergebnis der Anwendung stehen nach Geeignetheitsgrad bewertete Teamzusammensetzungen, die dem Teamrecruiter eine Entscheidungsvorlage liefern. Die Applikation nimmt dem Teamrecruiter somit nicht die Auswahl der Zusammensetzung ab, aber sie kann dank semantisch annotierter Daten mehr Informationen aus vorhandenem Wissen verarbeiten und eine komplexere Bewertung ermöglichen, als dass manuell durch den Teamrecruiter machbar wäre.

1.3. Gliederung

Im Kapitel Grundlagen werden die für diese Arbeit zentralen Begriffe und der disziplinäre Rahmen erläutert. Dazu gehören einerseits Semantic Web und Ontologien als auch die Theorie der Teambildung und der Bereich Human Resources. Darauf aufbauend werden im Kapitel drei Eigenschaften existierender Wissensmanagement-Systeme bzw. Ansätze untersucht, die zur Planung von Teams benutzt werden können. Eine Anforderungsanalyse mit der Benennung von Anwendungsfällen des Prototyps schließt das Kapitel ab. Im sich anknüpfenden Kapitel Konzeption und Design werden die Kriterien optimaler Teams definiert und die relevanten Aspekte der HR-Ontologie vorgestellt sowie konzeptionelle Lösungsansätze präsentiert. Die Beschreibung der Teamplanungsalgorithmen und der Evaluation von Teams sind ebenfalls hier zu finden. Das Kapitel Implementierung der Anwendung enthält die Beschreibung der Architektur des Prototyps sowie Implementierungsdetails bei der technischen Umsetzung der Teamkriterien. Die Arbeit endet in Kapitel 6 mit einer Zusammenfassung und Bewertung der erreichten Ergebnisse sowie einem Ausblick auf mögliche Weiterentwicklungen des Themas.

2. Grundlagen

Das Kapitel Grundlagen soll in die Themenfelder der Arbeit einführen. Dazu gehören Fähigkeiten, Human Resources, Recruitment, Teams und Semantic Web.

2.1. Human Resources und Skills

Wie Biesalski in [Bie06] schreibt, gewinnt die Ressource Mensch für den wirtschaftlichen Erfolg eines Unternehmens zunehmend an Bedeutung. Drucker [Dru06] sagt es mit den Worten: „nur Humanressourcen besitzen die Fähigkeit zur Koordination, Integration und Bewertung“. Sie seien die Träger der Kompetenzen in einem Unternehmen und können nur dann optimal genutzt werden, wenn im Unternehmen ein strategisch ausgerichtetes ganzheitliches Personalmanagement betrieben werde.

Inhaltlich gleichbedeutend mit dem Personalmanagement ist der Begriff des Human Resource Management. Der positive Einfluss der Ressource Mensch ist stark von der Verwendung und Verwaltung des Wissens und der Fähigkeiten der Mitarbeiter abhängig. Personalmanagement ist deshalb eng mit dem Wissensmanagement verknüpft. Wie aus [FK03] zu erfahren ist, entstanden Ende der 80er Jahre für den Bereich der Büroarbeit erste Ansätze für eine Computer-Unterstützung der kooperativen Arbeit. Daraus entwickelte sich das interdisziplinäre Forschungsgebiet CSCW (Computer Supported Cooperative Work), das sich mit den Problemen der Verbesserung der Gruppenarbeit bezüglich deren Durchführung (Prozess) und Ergebnis (Inhalt) durch den Einsatz von Informations- und Kommunikationstechnologien beschäftigt. CSCW-Systeme bieten vor allem Unterstützung für eine enge Kommunikation, wohingegen der Austausch von explizitem Wissen vernachlässigt wird. Weiterhin schreiben die Autoren von [FK03], dass IT-basierte Lösungen für das Wissensmanagement den effizienten Umgang mit Wissen in Unternehmen

unterstützen sollen und zu diesem Zwecke existierende Technologien wie Datenbanken, Archive, Suchmaschinen oder Ontologien als Grundbausteine einsetzen. Der Austausch expliziten Wissens kann in Personal- oder Wissensmanagement-Systemen durch semantisch annotierte Daten und den Einsatz von Semantic Web Technik vonstatten gehen. Mit dieser Technik werden solche Systeme sinnvoll unterstützt und hinsichtlich des Effizienzverhaltens verbessert.

2.1.1. Recruitment

Recruitment bezeichnet die Personalbeschaffung eines Unternehmens. Im Beispiel der Siemens AG beginnt dieser Prozess in der Regel mit der internen oder externen Ausschreibung einer freien Stelle, in der die Zielsetzung, die Aufgaben und die Kompetenzen des gesuchten Stelleninhabers formuliert werden. Es folgt die Begutachtung daraufhin eingegangener Bewerbungen hinsichtlich der Stellenkriterien. Nach der Auswahl geeigneter Kandidaten finden Vorstellungsgespräche statt und schließlich entscheidet sich der Personalverantwortliche der jeweiligen Abteilung bzw. des Unternehmens zur Anstellung eines Bewerbers oder zur Neuauausschreibung, je nachdem ob die Anforderungen der Stelle erfüllt werden konnten.

Tolksdorf [TM06] schreibt, dass das Medium Internet bei der Vermittlung von Arbeitsplätzen immer mehr an Bedeutung gewinnt. In der Allensbacher Computer- und Technik-Analyse (ACTA) 2004 [Köc08] wird ersichtlich, dass die Bedeutung des Internet bei der Deckung des sporadisch auftretenden, stärker fokussierten Informationsbedarfs wächst, was auch Informationen im Zusammenhang mit Stellenmärkten betrifft. 2004 bezogen 27,9% der befragten Bevölkerung in der Altersgruppe der 14- bis 64-jährigen gelegentlich Informationen zu Stellenanzeigen aus dem Netz. Die aktuelle ACTA Umfrage [Köc08] ergab, dass 2008 38% der Befragten derselben Altergruppe Informationen des Stellenmarktes regelmäßig und 7% häufiger benutzen, genauso häufig wie aktuelle Wirtschaftsmeldungen und sogar häufiger als Informationen zu Computer- und Onlinethemen. In Abbildung 2.1 werden die relevanten Ergebnisse dargestellt.

Die Personalbeschaffung unter Einsatz elektronischer Medien oder Personalsysteme wird E-Recruiting genannt. Seit 2005 werden bei Siemens nur noch Online-Bewerbungen mittels einer elektronischen Bewerbungsmappe angenommen. Damit ist die Siemens AG in dieser Hinsicht eines der ersten unter den 500 welt-

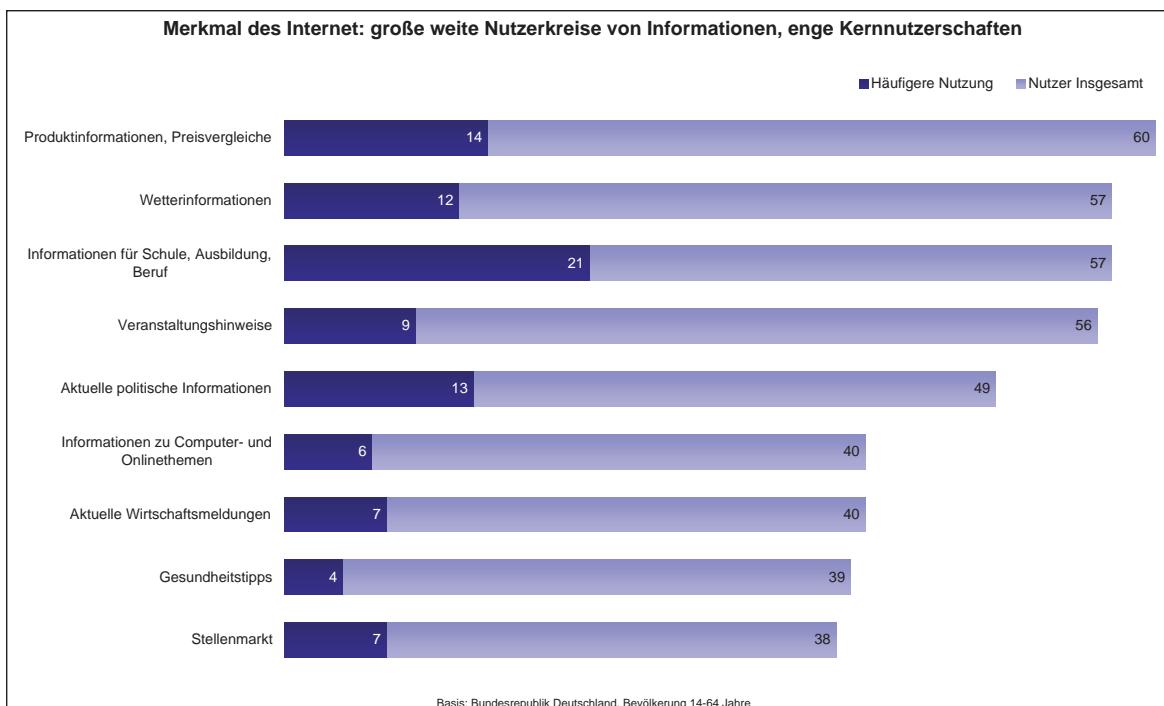


Abbildung 2.1.: Auszug aus der ACTA 2008, angelehnt an [Köc08]

weit größten Unternehmen. Der Leiter des E-Recruiting bei Siemens, Dr. Hans-Christoph Kürn, sagt dazu in einem Spiegel-Interview [Mar07]: „Inzwischen haben wir uns komplett von Papier verabschiedet. [...] Wir haben unser Verfahren so konzipiert, dass es der Post-Bewerbung sehr nahe kommt - nur eben elektronisch.“ Zur Siemens-Bewerbungsmappe gehören für eine Initiativbewerbung oder Bewerbung auf eine spezielle Ausschreibung ein Lebenslauf, Zeugnisse sowie ein frei verfasster Bericht ähnlich einem Anschreiben. In einem ersten Durchgang sichten dann Dr. Kürn und mehrere dezentral bei Siemens Deutschland eingesetzte Recruiter die Bewerbungen nach einem Grobraster und verschicken an jeden Bewerber entweder eine Eingangsbestätigung oder eine Absage. Die Bewerber befinden sich dann maximal vier Monate im Pool „Talents Link“ und können bei elektronischen Matchings zwischen Bewerbern und offenen Stellen von jedem Recruiter gefunden werden. Dr. Kürn schildert, dass diese Matchings allerdings nur Entscheidungsvorbereitung seien und nicht die Entscheidung über die Beschäftigung selbst. Der Recruiter empfiehlt dem Ausschreibenden der Abteilung einen oder mehrere Bewerber und erst dann finden eventuell Bewerbungsgespräche statt.

In ihrem Artikel schreiben Tolksdorf et al. [TM06], dass Unternehmen derzeit folgende Möglichkeiten haben, ihre offenen Stellen zu veröffentlichen:

- auf allgemeinen oder spezialisierten Jobportalen im Internet
(z.B. www.monster.de, www.jobpilot.de, www.stepstone.de, www.ansus.de)
- auf der firmeneigenen Webseite
(z.B. www.siemens.de/karriere)
- implizit bei sich entwickelnden Jobsuchmaschinen
(z.B. www.worldwidejobs.de)

Auf den genannten Webseiten können Arbeitssuchende freie Stellen finden, die ihrem Profil entsprechen. In [TM06] wird beschrieben, wie mittels Semantic-Web-Technologie die Auffindbarkeit und Vergleichbarkeit von Stellenanzeigen und Bewerberprofilen im Web verbessert werden können. Ausschreibungen jeglicher Art beginnen mit der Erstellung eines Stellenprofils, doch anstelle einer Freitextformulierung basiert bei den Autoren eine Ausschreibung auf kontrollierten Vokabularien (Ontologien). Ein semantischer Vergleichsmechanismus, der auf dem Wissen der Ontologien und verfügbarer Marktdaten beruht, berechnet die semantische Ähnlichkeit zwischen Stelle und Bewerber. Der Einsatz von semantischen Technologien für Recruiting-Prozesse bringt Vorteile für die beteiligten Akteure. Dienstleister können qualitativ bessere Vermittlungsdienste anbieten, für die Stellenanbieter steigen damit die Reichweite ihrer Angebote und die Präzision der Ansprache potentieller Bewerber. Beim Arbeitssuchenden verringert sich der Aufwand des Auffindens geeigneter Angebote.

2.1.2. Kompetenzen, Skills, Fähigkeiten

Eine einheitliche Definition für den Begriff der Kompetenz lässt sich nur schwer finden, da er in der Literatur vielfach eine andere Bedeutung besitzt. Für Huber [Hub99] bedeutet Kompetenz „eine Entwicklung grundlegender Fähigkeiten, die weder genetisch angeboren noch das Produkt von Reifungsprozessen sind, sondern vom Individuum selbst hervorgebracht werden.“ Dieser Interpretation kann sich für die vorliegende Arbeit angeschlossen werden, da auch die Entwicklung von Fähigkeiten aufgrund von Erfahrungen und beruflichen Tätigkeiten eine Rolle spielt. Die Begriffe Kompetenz, Skills und Fähigkeiten werden im Folgenden synonym verwendet, da keine genaue Trennung untereinander ermittelt wurde.

Zu der in der Einleitung genannten Definition von Wissensmanagement durch

Herbst sei hier eine weitere von Abecker [Abe04] erwähnt: „Wissensmanagement ist ein strukturierter, ganzheitlicher Ansatz für die nachhaltige Verbesserung der Behandlung von implizitem und explizitem Wissen in einer Organisation auf allen Ebenen (Individuum, Gruppe, Organisation, interorganisatorisch).“ Biesalski [Bie06] bezeichnet implizites Wissen als jenes, das in einem hohen Maße an Erfahrungen gebunden ist und meist sei sie dem Wissensträger nicht vollständig bewusst, was die Weitergabe nicht ohne weiteres möglich macht. Er nennt explizites Wissen dagegen artikuliertes und dokumentiertes Wissen. Ein entscheidendes Merkmal sei, dass es personenungebunden existieren kann und damit öffentlich zugänglich sei.

Gronau [Gro06] nutzt ebenfalls für seine Knowledge Modeling and Description Language (KMDL, siehe Abschnitt 3.1.3) eine Einteilung von Wissen in Wissensarten, nach der stillschweigendes oder explizites Wissen individuell für eine Person oder kollektiv in einer Gruppe vorkommen können. In diesem Zusammenhang können Fähigkeiten auch in objektive und subjektive Fähigkeiten eingeteilt werden. Objektive Fähigkeiten sind nachprüfbar und belegbar z.B. durch Zeugnisse. Subjektive Fähigkeiten hingegen können nicht per Beleg nachgewiesen werden wie z.B. die Gesamtheit der erlangten Kenntnisse in einem Projekt sondern viel mehr durch eine Einschätzung z.B. des Projektleiters oder Vorgesetzten. Für Hiermann und Höfferer [HH05] (siehe Abschnitt 3.1.2) stellen lediglich echte Qualifikationen, in diesem Fall eine Mischung aus „hard“ und „soft skills“, objektive Fähigkeiten dar, wohingegen Erfahrungen subjektive Fähigkeiten seien. Ein Teamrecruiter bei Siemens bestätigte in einem Interview, dass es gängige Praxis sei, Mitarbeiter nach einem Projekt oder einer anderen Tätigkeit mittels des Durchschnitts einer Selbst- und Fremdeinschätzung aussagekräftig zu bewerten. Demnach seien auch Erfahrungen durchaus realistisch beurteil- und vergleichbar. Biesalski [Bie06] schreibt, dass die Einschätzung oft anhand eines fest vorgegebenen Kompetenzkatalogs erfolgt. Das anfangs subjektive Bild durch die Selbsteinschätzung eines Mitarbeiters wird durch die Fremdeinschätzung derselben Person objektiviert. Denkbare Methoden seien Parallelbewertungen, Gruppenfeedback oder das Mitarbeitergespräch, was sich auch mit den genannten Mitteln des Siemens Teamrecruiters deckt. Für die Erfüllung der vorliegenden Arbeit stehen dennoch „hard skills“ im Vordergrund, da die Erarbeitung einer Lösung mit weichen Fähigkeiten aus dem zeitlichen Rahmen fallen würde.

Kompetenzmanagement weist nach Biesalski [Bie06] eine enge Affinität zum Be-

griff des Wissensmanagements auf, vor allem im Bezug auf die auch in der Definition des Wissensmanagements angeführte Ganzheitlichkeit des Ansatzes. Er definiert den Begriff folgendermaßen: „Kompetenzmanagement hat die Aufgabe, Mitarbeiterkompetenzen zu beschreiben, sie transparent zu machen, sowie den Transfer, die Nutzung und Entwicklung der Kompetenzen hinsichtlich strategischer Unternehmensziele sicherzustellen. Kompetenzmanagement umfasst dabei sowohl die Sichtweise des Unternehmens (strategische Ausrichtung), als auch die Sichtweise des Mitarbeiters (persönliche Ziele).“ Zelewski schreibt in [Zel04], dass ein mittelbarer aber kritischer Bedarf für die computerbasierte Unterstützung von effektivitäts- und effizienzfördernden Wissensmanagement-Technologien besteht. Dieser Bedarf ließe sich seiner Meinung nach speziell im Hinblick auf Projektgeschäft und Projektorganisation weiter konkretisieren. Im Fall einer Projektanfrage bereitet es oftmals große Schwierigkeiten, diejenigen Kompetenzen und Kompetenzträger zu identifizieren, über die ein Unternehmen einerseits verfügt und die andererseits für die Erfüllung einer wissensintensiven Aufgabe benötigt werden. Für Stader und Macintosh [SM99] besitzt ein Ontologie-basiertes Kompetenzmanagement-System das Potential, folgende Geschäftsprozesse in einem Unternehmen zu unterstützen:

- **Analyse von Wissenslücken:** Das System sollte die Menge der aktuell verfügbaren Fähigkeiten analysieren und untersuchen, welche Fähigkeiten gefährdet sind (z.B. durch Weggang von spezialisierten Mitarbeitern) oder sogar bereits fehlen. Es soll also anhand von semantischen Vergleichen die Differenz zwischen den notwendigen und existierenden Fähigkeiten im Unternehmen ermitteln und so dabei helfen, Wissenslücken frühzeitig zu erkennen und zu schließen.
- **Analyse zur Projektteam-Zusammensetzung:** Basierend auf Projektanforderungen soll das System genau die Mitarbeiter für ein Team identifizieren, die gemeinsam die Anforderungen am besten erfüllen, weil sie entweder die benötigten Fähigkeiten besitzen oder sie am leichtesten erlangen können.
- **Recruitment Planungen:** Zur Besetzung von vakanten Stellen sollte das System Mitarbeiter-Profile zur möglichen Rekrutierung vorschlagen und Bewerber gegen Stellenprofile bewerten.
- **Trainingsmaßnahmen:** Das System identifiziert bestimmte Mitarbeiterfähigkeiten, die im Rahmen von Bildungsmaßnahmen weiterentwickelt werden

könnten. Es wählt diese Fähigkeiten aus, weil ein hohes Vorkommen im Unternehmen essentiell ist oder die Mitarbeiter ihrem Profil nach besonders gut für Weiterbildungen geeignet sind.

Diese Auflistung der Aufgaben zeigt, dass Kompetenzmanagement im Kern einen ständigen Vergleich von vorhandenen und notwendigen Kompetenzen durchläuft. Durch den Einsatz von semantisch annotierten Mitarbeiterdaten und Semantic-Web-Technologien kann Kompetenzmanagement daher effizienter gestaltet werden.

2.2. Teams und Teamplanung

Zentraler Gegenstand dieser Arbeit sind Teams und ihre effiziente Zusammensetzung. An dieser Stelle sollen die Eigenschaften von Teams und Teamplanung näher beleuchtet werden.

Katzenbach [JKR93] definiert den Begriff des Teams folgendermaßen: „A team is a small number of people with complementary skills who are committed to a common purpose, performance goals, and approach for which they hold themselves mutually accountable.“ Damit ergeben sich fünf Eigenschaften eines Teams:

- geringe Mitgliederanzahl
- komplementäre Fähigkeiten
- Verpflichtung zu einer gemeinsamen Aufgabe und Effizienz-Zielen
- gemeinsame Herangehensweise
- gegenseitige Verantwortlichkeit

Katzenbach schreibt außerdem, dass die Fokussierung auf die Arbeitsleistung des Teams ein wichtigerer Erfolgsfaktor sei, als Kommunikation oder ein gutes Mit einander. Diese Definition stellt die Arbeitsleistung in den Vordergrund, die die Voraussetzung für Erfolg genau dann bildet, wenn ein Team die beschriebenen fünf Eigenschaften besitzt. Das ist insbesondere für die Teamplanungsalgorithmen wichtig (siehe Abschnitt 4.2). Die genannten Eigenschaften sind nicht einschränkend und gelten für jegliche Art oder Typ von Team. Stock [SH08] unterscheidet verschiedene Typen von Teams, je nachdem, wie stark innerhalb eines Teams

die Sach- oder Beziehungsebene ausgeprägt sind. Die Sachebene bezieht sich dabei auf die Relevanz aufgabenbezogener Aspekte im Team, während die Beziehungsebene Auskunft über die zwischenmenschliche Interaktion innerhalb eines Teams gibt. Es ergeben sich vier Teamtypen, was Abbildung 2.2 veranschaulicht. Eine Einzelkämpfer-Gruppe zeichnet sich weder durch emotionalen Zusammen-

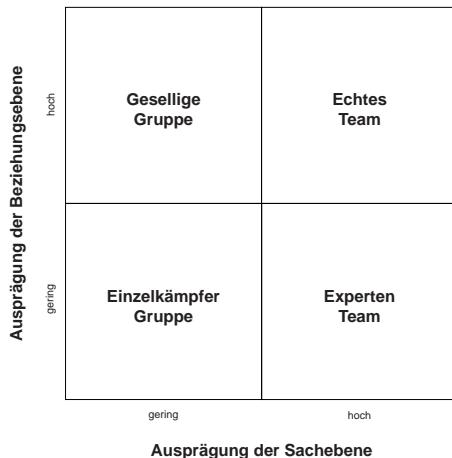


Abbildung 2.2: Die vier Team Typen, nach [SH08]

halt noch durch fachliche Kooperation aus, was eine geringe Teamleistung mit sich bringt. Ein echtes Team existiert dann, wenn Sach- und Beziehungsebene jeweils hoch ausgeprägt sind. In einem solchen Team sind Kooperation und die Kommunikationsflüsse klar geregelt und die partizipierenden Mitglieder unterstützen sich auch in persönlichen Dingen. Ziel der Applikation und der Teamplanungsalgorithmen sollte es also sein, möglichst solche Teams zusammenzusetzen, um dem Teamrecruiter sinnvolle und effiziente Teamzusammensetzungen zu liefern.

Stock [SH08] definiert ebenfalls Kriterien zur Beschreibung und Unterscheidung verschiedener Arten von Teams. Es wird zwischen klassischen und neueren Teamarten unterschieden. Abbildung 2.3 zeigt, dass eine ganze Reihe unterschiedlicher Teamtypen existiert. In Ausführungsteams ist der Grad der Routine im Rahmen der Aufgabenerfüllung deutlich höher als in Management-Teams, denn Ausführungsteams sind primär mit der Erfüllung operativer Aufgaben betraut. Permanente Teams können z.B. in Abteilungen eines Unternehmens vorkommen, die nach der Erledigung einer Aufgabe in derselben Zusammensetzung die nächste Aufgabe übernehmen. Projektteams hingegen zerfallen nach Abschluss des Projektes wieder. In funktionsbezogenen Teams stammen die Mitglieder weitestgehend aus demselben Funktionsbereich. Multifunktionale Teams weisen eine hohe funktiona-

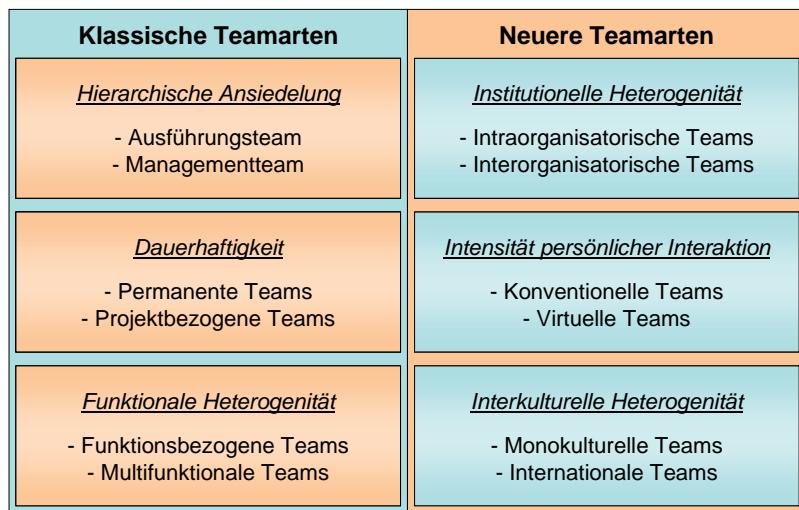


Abbildung 2.3.: Teamarten, angelehnt an [SH08]

le Heterogenität auf, z.B. Innovationsteams, die sich aus Mitgliedern verschiedener Bereiche eines Unternehmens zusammensetzen. Mitglieder intraorganisatorischer Teams stammen aus demselben Unternehmen und sie sind auch für die vorliegende Arbeit relevant. Interorganisatorisch ist ein Team dann, wenn seine Mitglieder aus verschiedenen Unternehmen kommen. In konventionellen Teams existiert eine „Face-to-Face“ Interaktion zwischen den Mitgliedern und ihre Arbeitsplätze befinden sich in räumlicher Nähe zueinander. Virtuelle Teams zeichnen sich dadurch aus, dass ihre Mitglieder regional verteilt an einer Aufgabe arbeiten. Sie kommunizieren hauptsächlich mittels moderner Informationstechnologien. Monokulturelle Teams bestehen aus Mitgliedern einer Länderkultur, bei interkulturellen Teams stammen die Mitglieder aus unterschiedlichen Kulturen.

Der Fokus der Anwendung liegt auf intraorganisatorischen Projektteams, da diese eine sehr häufig vorkommende Teamform sind und hier die beste Datenlage durch Wissensmanagement vorhanden ist.

Das Deutsche Institut für Normung (DIN) definiert in der Norm-69901 ein Projekt als: „[...] ein Vorhaben, das im Wesentlichen durch die Einmaligkeit der Bedingungen in ihrer Gesamtheit gekennzeichnet ist, wie z.B. Zielvorgabe, zeitliche, finanzielle, personelle und andere Begrenzungen, Abgrenzung gegenüber anderen Vorhaben, projektspezifische Organisation.“ Litke [LK06] gibt eine umfassendere Definition für ein Projekt an als: „ein abgrenzbares Einzelvorhaben mit definiertem Anfang und Ende, welches neuartig, risikoreich (terminlich, wirtschaftlich, technisch) und komplex ist, im Laufe der Abwicklung sich ändernde organisatorische

Bedürfnisse hervorruft und von hoher Bedeutung für die Unternehmung ist.“ Projektgruppen arbeiten kurzzeitig zusammen, sind nicht domänenspezifisch, also auf ein Anwendungsgebiet beschränkt, und besitzen in der Regel einen Projektleiter und mehrere Teammitglieder. Diese besitzen unterschiedliche Qualifikationen und kommen aus verschiedenen Abteilungen eines Unternehmens oder wirken als externe Mitarbeiter im Team mit.

Während es bei Recruitment und Stellenbesetzungen darum geht, eine passende Person für eine Stelle zu finden, besteht die Schwierigkeit bei Teamplanung darin, für eine Aufgabe Mitarbeiter derart zu einem Team zu kombinieren, dass sie u.a. die fünf Eigenschaften eines Teams nach Katzenbach besitzen und in der Summe alle notwendigen Fähigkeiten aufbringen, um miteinander arbeiten zu können. West [Wes03] beschreibt das Problem folgendermaßen: „It's about finding the magic mix, that blend of skills and experience that combines maturity, energy, determination and creativity. It's like a ballet when it happens“. Fuchs-Kittowski [FK03] fasst die Merkmale von Teams folgendermaßen zusammen: Der Fokus der Interaktion eines Teams liegt auf einem Ziel bzw. einer Aufgabe. Ihre Interaktion ist von kurzer Dauer und dient dem Zweck der Wissenserzeugung. Gegenstand der engen Interaktion sind gemeinsame Objekte. Die Gruppe bildet sich formell und wird auch formell legitimiert geführt. Die Zusammensetzung des Teams ist nicht variabel und eine Mitgliedschaft nicht nur freiwillig. Die Fachkompetenzen der kleinen Gruppe sind heterogen zueinander. Da die Mitglieder meist aus derselben Organisationswelt kommen, ist ein Team für Externe nicht offen.

2.3. Semantic Web

Ohne das Semantic Web, das die technische Grundlage dieser Arbeit bildet, wäre der komplexe Vergleich von Mitarbeiter- und Teamprofilen nicht möglich. Aus diesem Grund wird im folgenden Abschnitt Semantic Web näher beschrieben. Dem schließt sich ein Vergleich von existierenden Ontologien im HR-Bereich an, aus dem ersichtlich wird, warum die Entscheidung zur Nutzung der zur Verfügung gestellten Ontologie gefallen ist.

2.3.1. Definition und Eigenschaften

Das heutige Internet bzw. das World Wide Web (WWW) bietet vielleicht die einfachste Möglichkeit, Informationen zu verbreiten. Jeder kann mit HTML (Hyper-text Markup Language) Webseiten erstellen, sie nach eigenen Vorstellungen layouten und Links zu anderen Webseiten oder Ressourcen setzen. Das Resultat ist, dass das Web weiter enorm wächst und derzeit bereits mehr als 8 Billionen Webseiten beinhaltet [BCT06]. Noch immer werden die meisten Webseiten für den menschlichen Gebrauch geschrieben. [BCT06] schreiben, dass Computer in diesem Zusammenhang zur Darstellung der Informationen benutzt werden, also z.B. zum Dekodieren des Farbschemas, der Überschriften und Links. Darüber hinaus bieten Suchmaschinen eine populäre Möglichkeit, Webseiten aufzufinden. Das heutige Internet verändert auch die Art, wie Menschen miteinander kommunizieren oder Geschäfte abgewickelt werden. Die Autoren von [PPH06] sehen im „WWW das Herz einer voranschreitenden Revolution, die unsere Welt in eine Wissensgesellschaft transformiert“.

[HKRS08] zeigen aber auch die Probleme des heutigen Web. Eine unübersichtliche Fülle an Informationen ist ausgerichtet auf Menschen als Endnutzer, die die Bedeutung einer Webseite unabhängig von einer konkreten Repräsentation selbst erfassen und Zusammenhänge bilden. Die Informationen existieren lokalisiert im Web, das heißt sie sind angepasst an vorherrschende lokale, sprachliche und kulturelle Gegebenheiten. Ein Fakt, der problematisch bleibt, da Suchmaschinen zwar präzise aber nur stichwortbasiert sind. Die Suche nach dem Wort „Kohl“ lieferte bei Google am 28.01.2009 als erstes Ergebnis einen Artikel über den Altkanzler Helmut Kohl, dann einen Artikel über die Gattung der Kreuzblüter und an dritter Stelle die Internetpräsenz des Autohauses Kohl in Aachen. Das sind Suchergebnisse im deutschsprachigen Raum, eine weltweite Suche nach „Kohl“ lieferte erwartungsgemäß andere Ergebnislisten. Des Weiteren sind vorhandene Informationen heterogen auf verschiedensten Ebenen: Zeichenkodierung, natürliche Sprache und Anordnung der Informationen auf Webseiten. Informationen werden nicht explizit spezifiziert sondern folgen aus einer Kombination gegebener Daten und bilden damit implizites Wissen. Damit können Menschen umgehen, Maschinen aber nicht. Sie brauchen formal logische Methoden, um automatisch schlussfolgern zu können.

Wie das World Wide Web, so geht auch das Semantic Web zurück auf Tim Berners-

Lee. Auf der „The Emerging Technologies Conference“ am Massachusetts Institute of Technology (MIT) am 29. September 2004 nennt Berners-Lee die Anfänge des Internet: „a primeval soup of many things that know each other but haven't been put together“ und das Semantic Web nennt er: „killer application in the life sciences“. In seinem bereits in der Einleitung erwähnten Artikel von 2001 definiert er das Semantic Web als Erweiterung des existierenden Web, in der Informationen eine definierte Bedeutung erhalten, wodurch Computer und Menschen zusammenarbeiten können [BL01]. Er schreibt, dass es einen Unterschied gibt zwischen Informationen, die für den menschlichen Gebrauch produziert und jenen, die hauptsächlich für Maschinen geschaffen wurden. Damit das Semantic Web funktionieren kann, müssen Computer Zugang zu strukturierten Sammlungen von Informationen und Mengen von Inferenzregeln haben, die sie dann nutzen, um automatisch schlussfolgern zu können. Die Autoren von [HKRS08] sehen als Lösungsansätze einerseits die Verwendung von Methoden der Künstlichen Intelligenz zur Auswertung bestehender strukturierter Informationen im Web und andererseits die Strukturierung der Web-Informationen zur Erleichterung der automatischen Auswertung. Sie sehen ebenfalls wie Berners-Lee in seinem Artikel zwei Voraussetzungen zur Realisierung: offene Standards zur Beschreibung von Informationen und Methoden zur Gewinnung von Informationen aus derlei Beschreibungen. Bereits 2001 nennt Berners-Lee Standards wie XML (eXtensible Markup Language), mit deren Hilfe Benutzer ihren Dokumenten eine eigene Struktur mit eigenen Tags geben können, ohne Aussagen über dessen Bedeutung zu treffen und RDF (Resource Description Framework), womit Bedeutung ausgedrückt werden kann, in dem sie kodiert in Tripeln, also elementaren Sätzen mit Subjekt, Prädikat, Objekt, maschinenlesbar wird. Als dritte Basiskomponente sieht Berners-Lee die Ontologien, die philosophisch gesehen Theorien über die Natur der Existenz sind.

So wie die Autoren von [HKRS08] und [BCT06] könnte man das heutige Web als das syntaktische Web betrachten, in dem charakterisiert ist, was wohlgeformte Daten sind und das semantische Web erweitert diese Syntax um Bedeutung. Der Gartner Hype Cycle¹ für aufkommende Technologien 2007 ordnet Semantic Web in der dritten von fünf Phasen ein und erwartet noch mehr als 10 Jahre, bis es die Masse der Bevölkerung nutzen wird. Berners-Lee schreibt dazu: „The real power of the Semantic Web will be realized when people create many programs that collect Web content from diverse sources, process the information and exchange the results with

¹mehr Informationen unter <http://www.gartner.com/pages/story.php?id=8795.s.8.jsp>

other programs“.

Obwohl „Web“ in Semantic Web nahe legt, dass es sich dabei um verteilte Anwendungen für das Internet handelt, können die Vorteile von semantischen Webtechnologien durchaus auch für einzelne Nutzer oder betriebliche Anwendungen genutzt werden. Die Autoren von [Joh06] untergliedern den erwarteten Nutzen des Einsatzes semantischer Technologien in die folgenden Kategorien:

- Verbesserte Suche und Navigation
- Integration von Personen, Anwendungen und Daten
- Prozessoptimierung

Eine Anwendung zur Teamplanung mithilfe semantischer Webtechnologien kann ebenfalls diese Vorteile nutzen. Der Prozess der Teamplanung wird durch semantisch annotierte Daten effektiver gestaltet, indem Wissen über Mitarbeiter und Projekte integriert wird und eine projektspezifische Suche nach passenden Mitgliedern verbessert wird.

Darüber hinaus äußern sich viele Autoren sehr positiv über der Einsatz von ontologiebasierter Hilfe bei KnowledgeManagement-Systemen und speziell Teamplanungskomponenten. Zelewski [Zel04] beschreibt den Nutzen folgendermaßen: „Ontologien stellen aus der Perspektive des Knowledge-Level-Engineerings einen der Erfolg versprechendsten Ansätze für die Entwicklung computerbasierter Wissensmanagement-Systeme dar“. Gronau schreibt in [FSRG06] dazu: „Verbesserungspotentiale von Skillmanagement-Systemen bestehen besonders im Zusammenhang mit dem Einsatz von Ontologien oder semantischen Netzen zur Definition von Qualifikationsprofilen“.

2.3.2. Ontologien

Das Wort Ontologie stammt vom Griechischen *ontos* (sein) und *logos* (Wort). Es bildet eine zentrale Komponente des Semantic Web. Hesse [Hes02] schreibt, dass Ontologie ein überliefelter Begriff aus der Philosophie ist und dort für die Lehre vom Sein steht, genauer von den Möglichkeiten und Bedingungen des Seienden. Ontologien sind also eng verwandt mit der Erkenntnistheorie, die sich mit den Möglichkeiten und Grenzen menschlichen Wahrnehmens und Erkennens auseinandersetzt. Die Autoren von [Joh06] erklären, dass eine Ontologie genutzt werden

kann, um entweder einen Wirklichkeitsausschnitt mit seinen Kontextinformationen abzubilden, eine Terminologie zu repräsentieren oder die Daten in verschiedenen semantischen Informationsprozessen zu vereinheitlichen. Nach Gruber ist eine Ontologie in der Informatik folgendes: „A Ontology is a formal specification of a shared conceptualization of a domain of interest“ [Gru93]. Dahinter verbirgt sich, dass eine Ontologie maschinell interpretierbar ist, sie auf Konsens beruht, Begrifflichkeiten beschreibt und auf einen bestimmten Gegenstandsbereich bezogen ist. Zelewski schreibt in [Zel04], dass Ontologien mittels ihrer expliziten und formalen Semantik die computerbasierte Verarbeitung von Wissen unterstützen und daher einen maßgeblichen Beitrag zur allseits angestrebten, effizienzsteigernden Wiederverwendung von Wissen („knowledge reuse“) leisten. Weiterhin schreibt er, dass mithilfe der formalen Semantik Ontologien in unmittelbarem Zusammenhang stehen, mit den aktuellen Forschungs- und Entwicklungsanstrengungen, ein „weltumspannendes Semantic Web“ zu etablieren. Mithilfe von Ontologien können im Semantic Web Regeln formuliert und Schlussfolgerungen gezogen werden. Ontologien sind somit nach Hesse [Hes02] ein „Regelwerk zur Definition von Konzepten und deren Relationen“. Sie ordnen die Konzepte in einer Hierarchie an und definieren, welche Attribut-Wert-Paare in ihrem Gültigkeitsbereich (z.B. einer Anwendungsdomäne) eingeschränkt sind.

Ontologien können verschiedene Detaillierungsgrade besitzen, die sich anhand der Mächtigkeit ihrer semantischen Ausdrucksfähigkeit unterscheiden lassen. Die Verfasser von [SW01] erklären die Eigenschaften der einzelnen Stufen, die in Abbildung 2.4 dargestellt werden. Die einfachste Form einer Ontologie ist die des Kata-

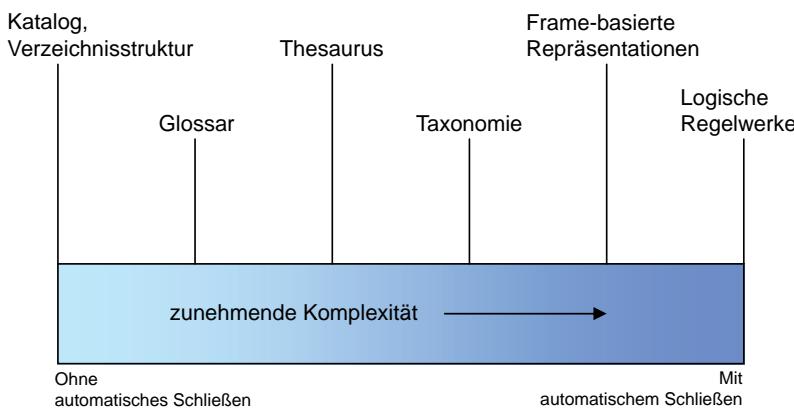


Abbildung 2.4: Typen von Ontologien, nach [SW01]

loges bzw. einer Verzeichnisstruktur. Verzeichnisstrukturen bilden einfache Hierarchien ohne semantische Metainformation ab, dennoch sind sie meist hierarchisch aufgebaut. Ein Glossar stellt ein zusammenhängendes Terminologiefeld und die darin verwendeten Begriffe in einen Zusammenhang. Die einzelnen Begriffe sind mitunter direkt verlinkt, allerdings ohne dass der dadurch entstehenden Relation eine Bedeutung zugeschrieben wird. Oftmals wird mit der Erklärung der Begriffe beabsichtigt, ein Verständnis für die Verwendung der Begriffe zu bilden. Als Ergänzung können die im Glossar verwendeten Begriffe in einem Thesaurus hierarchisch angeordnet werden. Die jeweiligen Konzepte werden dann in dem Verhältnis zu Ober- und Unterbegriffen angeordnet. Daten- oder Wissensmodelle beschreiben Objekte sowie die Relationen zwischen Objekten mithilfe von Typeninformation. Taxonomien ermöglichen es, Begriffe oder Gegenstände (Entitäten) nach unterschiedlichen Ordnungskriterien zu klassifizieren. Zur Darstellung von Gegenständen werden hierarchische Beziehungen verwendet. In der Linguistik wird unter Klassifikation die Segmentierung sprachlicher Begrifflichkeiten verstanden, um ein formales Sprachsystem zu beschreiben. Die Modellierung von Taxonomien in Form von Klassen, Kategorien oder Konzepten beabsichtigt letztlich immer ähnliche Ziele, und zwar die Merkmale gleichartiger Objekte mengentheoretisch zusammenzufassen. Die Verwandtschaftsgrade zwischen den Objekten lassen sich mit Graphen oder in Form von Baumstrukturen darstellen. Mithilfe von frame-basierten Repräsentationen können Prozeduren dargestellt werden. Logische Regelwerke wie z.B. die First Order Logic besitzen den höchsten Formalisierungsgrad.

Wie die Standards XML und RDF, die Berners-Lee in seinem Artikel [BL01] nennt, so existieren auch für Ontologien diverse standardisierte Sprachen. Seit 2004 ist OWL (Web Ontology Language) W3C Recommendation und bietet hinzukommend zu den Beschreibungssprachen wie RDF einen Satz logischer Axiome, die genutzt werden können, um Abhängigkeiten zwischen den Konzepten zu generalisieren. Mithilfe von logischen Ausdrücken können so regelhafte Beziehungen zwischen den Daten abgebildet werden. In OWL wurden Konzepte aus der frame-basierten Wissensrepräsentation, den Web Standards und der Prädikatenlogik zu einem Regelwerk zusammengeführt, um die Integration unterschiedlicher Ontologien zu ermöglichen.

Ontologien aus dem Bereich Human Resource

[BCT06] beschreibt, dass in der Regel drei Ebenen von Ontologien unterschieden werden: Top Level Ontologien, Mid Level oder auch Core Ontologien und die Vielzahl von Domain Ontologien. Mithilfe von Top Level Ontologien können generische Konzepte, wie z.B. „Ereignis“, „Prozess“ oder „Objekt“ modelliert werden. Die Mid Level oder auch Core Ontologien sind eine Zwischenform zwischen den generischen und anwendungsspezifischen Ontologien und drücken einzelne Basiskonzepte innerhalb der anwendungsspezifischen Ontologie aus. Die domänen spezifischen Ontologien beinhalten die Konzepte einer bestimmten Anwendungsdomäne wie z.B. die Taxonomien verschiedener Industriezweige. Die Konzepte von domänen spezifischen Ontologien werden häufig als Spezialisierung der Konzepte der generischen oder Kernontologien formuliert.

Nicht immer sind Ontologien frei zugänglich. Trotzdem existieren diverse Verzeichnisse, in denen man entwickelte Ontologien finden und wiederverwenden kann. In [JVM07] wird beschrieben, welche Ontologien im Bereich Human Resources existieren. Sie sollen im Folgenden kurz betrachtet werden.

KOWIEN Ontologie Das Projekt KOWIEN (KOoperatives WIssensmanagement in Engineering Netzwerken) wurde mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) des Rahmenkonzepts „Forschung für die Produktion von morgen“ gefördert und an der Universität Duisburg-Essen durchgeführt. Wie man aus [Zel04] erfährt, sollte mit dem Verbundprojekt eine technische Umsetzungslücke zwischen Techniken des Knowledge-Level-Engineerings und ihrer Anwendung in der betrieblichen Praxis durch anwendungsnahe Forschungs- und Entwicklungsarbeiten geschlossen werden. Die entwickelte Ontologie dient dem Wissenstransfer in die betriebliche Praxis und zeigt, wie sich Ontologien für Zwecke des Managements von Kompetenzprofilen erfolgreich konstruieren und anwenden lassen.

Die KOWIEN Ontologie ist nicht auf HR-Konzepte allein beschränkt. Sie ist vielmehr eine generische Top-Level-Ontologie, die ein domänen spezifisches Kompetenzprofil und Kompetenzmanagement bereitstellt. Derart ermöglicht sie die Darstellung und Schlussfolgerung über Aussagen, die über Mitarbeiterkompetenzen gemacht werden, um die Anwendungsfälle Expertensuche und Teamzusammen-

setzung zu realisieren. Das KOWIEN-Projekt wird im Abschnitt 3.1.4 näher betrachtet.

Wissensnetze Die Human Resource Ontology wurde an der Freien Universität Berlin und der Humboldt Universität zu Berlin im Wissensnetze Projekt für die Nutzung von semantischen Vergleichen zwischen Bewerber- und Stellenprofilen entwickelt. Sie basiert auf existierenden Standards und Klassifikationen. Wie in [MJW07] skizziert, dienen zur Beschreibung der Stelle bzw. der Arbeit die Berufskennziffern (BKZ) der Bundesagentur für Arbeit sowie das Standard Occupational Classification (SOC) System. Die einzelnen Branchen werden mithilfe der Wirtschaftszweige-Klassifikation (WZ2003) des Statistischen Bundesamtes sowie des North American Industry Classification System (NAICS) beschrieben. Fähigkeiten und Kompetenzen basieren auf der Skills Ontology des KOWIEN Projektes. Und schließlich wird zur Klassifizierung des Personalwesens auf den HR-XML-Standard des HR-XML-Konsortiums zurückgegriffen und mit HR-BA-XML eine Version für den deutschen Arbeitsmarkt der Bundesagentur für Arbeit genutzt. Der im Projekt entwickelte Prototyp kann Stellenangebote und Bewerberprofile semantisch vergleichen, indem Annotationen der importierten Vokabularen sowie Hintergrundwissen zur Anwendungsdomäne berücksichtigt werden. Die in dieser Arbeit benutzte HR-Ontologie von Ontonym ist die Weiterentwicklung des Prototyps der Wissensnetze Ontologie und ist mit Abstand die größte und branchenumfassendste Ontologie der hier beschriebenen.

TOVE Das TOVE (TOronto Virtual Enterprise) Projekt an der University of Toronto beinhaltet zwei Aufgaben: die Entwicklung einer großen generischen wiederverwendbaren Enterprise Ontology und einer Testumgebung [Fox92]. Diese umfasst mehrere Ontologien, die das Wissen über Aktivitäten, Zustände, Zeit, Kausalitäten, Ressourcen, Qualität und Kosten abbilden und so zur Modellierung der Unternehmen und ihrer Organisationsstruktur beitragen. Nähere Informationen konnten im Rahmen der Ausarbeitung nicht recherchiert werden.

LIP Ontology LIP steht für „Learning In Process“ und wurde in gleichnamigen EU-Projekt mit dem Ziel entwickelt, Lernprozesse in Arbeits- und Geschäftsprozesse zu integrieren [Sch05]. Der Fokus liegt auf der Automatisierung von On-

Demand Lernunterstützung und richtet sich an Mitarbeiter, ihr organisatorisches Umfeld und relevante Lernressourcen (z.B. Lernobjekte, Dokumente, Kollegen) eines Unternehmens. Dadurch soll die Lücke zwischen e-learning, Wissensmanagement, Human Resource Entwicklung und Betreuung geschlossen werden. Die drei

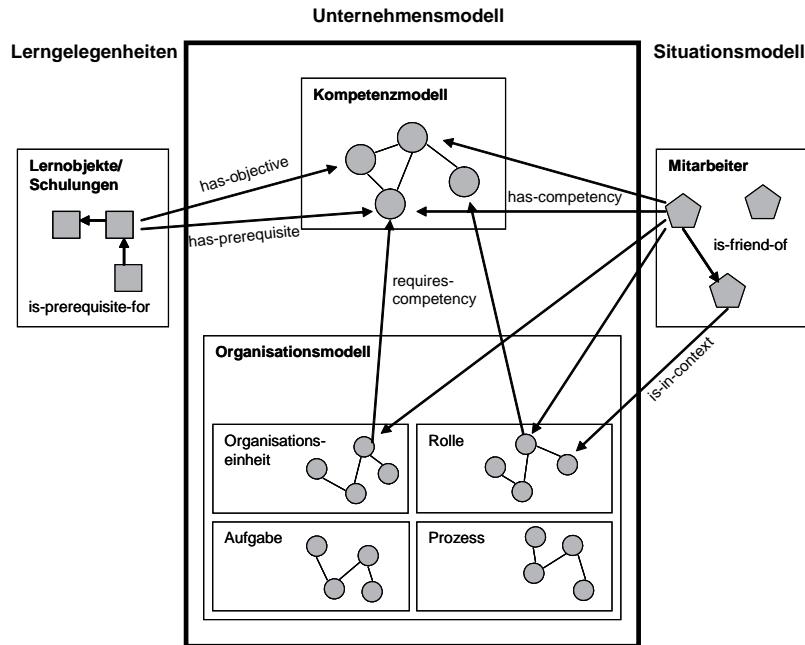


Abbildung 2.5.: Schema der drei Teile der LIP Ontologie, angelehnt an [Sch05]

Teile der Ontologie, die in Abbildung 2.5 zu sehen sind, sind Lerngelegenheiten, ein domänen spezifisches Unternehmensmodell und ein Situationsmodell in Form von Benutzern [KS07]. Sie sind durch Kompetenzanforderungen und Kompetenzobjekte miteinander verbunden. Die LIP Ontologie ist der Vorgänger der Professional Learning Ontology.

Professional Learning Ontology Diese Ontologie ist das Ergebnis der Zusammenlegung der LIP Ontologie mit Ansätzen des Kompetenzmanagements zur Verbesserung von Planungsabläufen im Rahmen von Lernprozessen und Weiterbildungen. Es werden verschiedene Disziplinen zusammengebracht, die sich mit formellen und informellen Lernprozessen in Organisationen beschäftigen. Abbildung 2.6 zeigt ihren schematischen Aufbau. Zur Modellierung der Ontologie wurde OWL-DL gewählt. Sie ist öffentlich verfügbar unter der Creative Commons Lizenz und wird am Forschungszentrum Informatik FZI Karlsruhe noch weiterentwickelt.

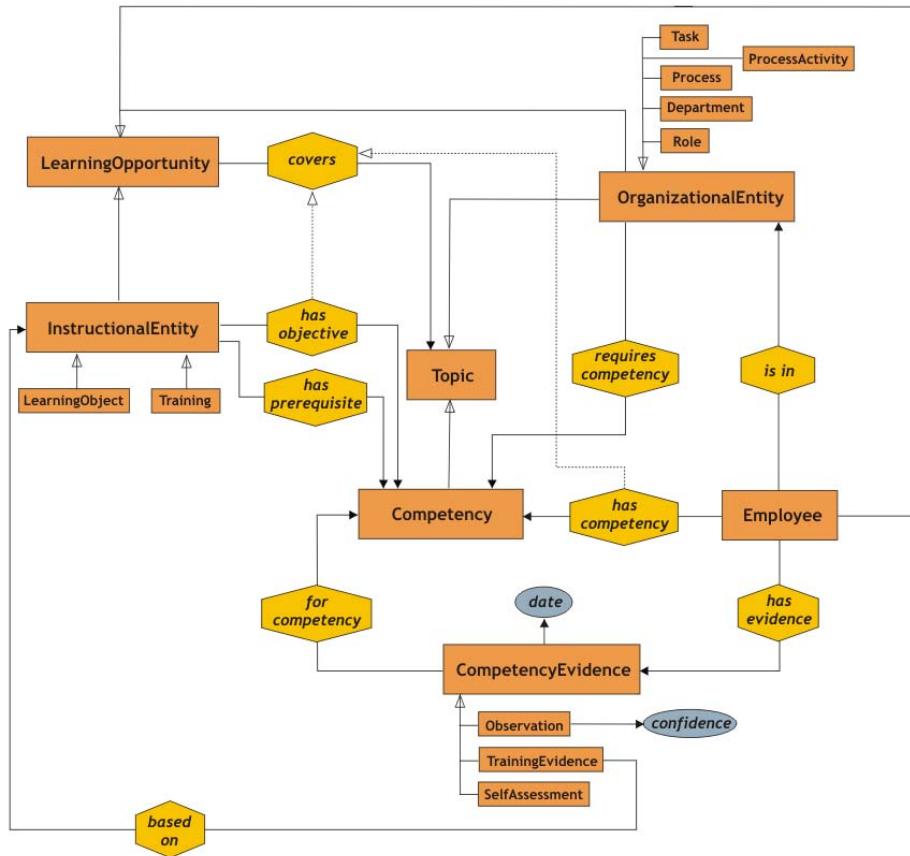


Abbildung 2.6.: Aufbau der Professional Learning Ontology

ePeople Diese Ontologie wurde bei der DaimlerCrysler AG im Zuge des neu eingeführten Personalmanagement-Systems ePeople [Bie04] in Kooperation mit dem Forschungszentrum Informatik FZI Karlsruhe entwickelt. Mittels des Kompetenzkatalogs werden Skillprofile der Mitarbeiter erstellt, die diese selber pflegen können. Auch Stellenbeschreibungen benutzen den Kompetenzkatalog, um eine Zuordnung von Soll-Anforderungen in Bezug auf Kompetenzen zu formulieren. Das Kompetenzmanagement wurde aus zwei unterschiedlichen Perspektiven betrachtet. Einerseits bietet es den Mitarbeitern die Möglichkeit, ihre Sprachkenntnisse, Kompetenzen und Zeugnisse online zu dokumentieren und sich damit transparent zu machen. Die andere Sichtweise ist die der Führungskraft, die das Kompetenzmanagement benutzt, um sich einen Überblick über die Kompetenzen der Mitarbeiter zu verschaffen. Diese Informationen geben einen Überblick über die Zusammensetzung und den aktuellen Bildungsstand der Mitarbeiter, aus dem sich dann z.B. Qualifikationsempfehlungen für Mitarbeiter ableiten lassen. Der andere Nut-

zen für Führungskräfte ist die Möglichkeit, gezielt nach mit entsprechenden Fähigkeiten ausgestatteten Mitarbeitern unternehmensweit zu suchen und diese unter Umständen dann auch zu rekrutieren.

Die Ontologie ist nicht öffentlich zugänglich und wird derzeit nicht weiter entwickelt. In Abschnitt 3.1.5 wird das System bezüglich seiner Eigenschaften im Kontext der Teamplanung näher beschrieben.

PROTON Die Proto Ontology wurde von Ontotext Lab im Rahmen des SEKT (Semantically Enabled Knowledge Technologies) Projektes entwickelt mit Partnern wie AIFB Universität Karlsruhe, Universität Innsbruck und anderen internationalen Institutionen. Diese Ontologie ist eine leichtgewichtige Top-Level Ontologie, die als Modellierungsbasis für eine Vielzahl von Aufgaben in unterschiedlichen Domänen dient. PROTON splittet sich in 4 Module auf: System, Top, Upper und Wissensmanagement. Wie in [TKM05] zu erfahren ist, enthält die Ontologie, die anfänglich BULO (Base Upper-Level Ontology) hieß, über 300 Klassen und 100 Properties. Geschrieben wurde die Ontologie in OWL Lite. Das Modul System enthält 5 Klassen und 5 Properties auf Metadatenebene. Das Modul Top bildet die höchste, allgemeinste konzeptuelle Ebene mit 20 Klassen. Upper enthält 200 generelle Klassen von Entitäten, die häufig in verschiedenen Domänen auftreten. Und schließlich enthält das Modul Wissensmanagement 38 Klassen spezieller Entitäten, die typisch für den Bereich Knowledge Management sind. Als Designziele werden die folgenden genannt:

- Unabhängigkeit von einer einzelnen Domäne
- leichtgewichtige logische Definitionen
- Anpassung an Standards
- Erfassen von benannten Entitäten und konkreten Domänen (z.B. Organisationen, Lokalitäten, Adressen)

Aufgrund des beschriebenen Vergleichs der einzelnen Ontologien bezüglich Größe, Verfügbarkeit, Aktualität und Domänenunabhängigkeit wurde die Ontologie des Projekts Wissensnetze für diese Arbeit gewählt. Sie wird näher in Abschnitt 4.1 beschrieben.

3. Problemanalyse und Anforderungsdefinition

Zur Durchführung einer Problemanalyse und der Definition von Anforderungen für die prototypische Applikation, sollen in diesem Kapitel themaverwandte Arbeiten, deren Vorgehensweise und Ergebnisse vorgestellt werden. Daran schließt sich die Anforderungsbeschreibung der Anwendung.

Die Autoren von [FK03] nennen und erörtern die vier grundlegenden Anforderungen an eine IT-Unterstützung kooperativer Wissensarbeit:

1. **Integration unterschiedlicher Wissensmanagement-Technologien in der globalen Wissensbasis:** Existierende Werkzeuge des Wissensmanagements helfen dabei, das bereits im Unternehmen kodiert in Dokumenten oder ideell in den Köpfen der Mitarbeiter vorhandene Wissen verfügbar zu machen. Verschieden strukturierte Wissensträger müssen in einer gemeinsamen transparenten Struktur abgebildet werden.

Für die Anwendung dieser Arbeit bedeutet das, eine gemeinsame Wissensstruktur für Aufgaben und Fähigkeiten der Mitarbeiter zu finden, wie in Abschnitt 2.3.2 beschrieben.

2. **Unterstützung der Bildung von Gruppen:** Wissensintensive Arbeitsprozesse benötigen das zeitnahe Auffinden und die flexible Integration von Wissens- und Kompetenzträgern, die häufig auch als Experten bezeichnet werden. Da die Relevanz einzelner Experten im Vorhinein nicht vollständig bestimmt werden kann, müssen die im Arbeitsprozess Tätigen die geeigneten Experten aus einem vorhandenen Angebot auswählen können.

Das Bilden von Gruppen bzw. Projektteams ist die Hauptaufgabe dieser Ausarbeitung. Ziel ist es, möglichst optimale Teams auf Basis der Kompetenzen

der Mitarbeiter im Unternehmen zusammenzusetzen.

3. **Unterstützung der Kommunikation und Kooperation als Voraussetzung für eine kooperative Wissenserzeugung:** Aufgrund von erforderlichen Reaktionsgeschwindigkeiten auf Kundenanforderungen sind neben asynchronen Kommunikationsmitteln wie email auch synchrone Werkzeuge wie Audio- und Videokonferenzsysteme vorzusehen, die eine Kommunikation in großen Gruppen ermöglichen. Bei vielen solcher Mittel sind die erzeugten Informationen flüchtig, da sie nur kurzzeitig als geistige Modelle existieren.

Die Nutzung gemeinsame zugänglicher Materialien ermöglicht den Beteiligten die Generierung eines einheitlichen Informationsstandes. Die Kompetenzen der Mitarbeiter der Wissensbasis sollten sowohl der Anwendung als auch den Teamrecruitern zur Verfügung stehen, um eine Entscheidung treffen bzw. die Berechnung einer optimalen Teamzusammensetzung vornehmen zu können. Die einzelnen Mitarbeiter sollten die Möglichkeit haben, ihre Kompetenzen in der Wissensbasis zu editieren. So wird das so genannte „gemeinsame Material“ konsistent gehalten und eine kooperative Wissensarbeit ermöglicht.

4. **Bewahrung des kooperativ erzeugten Wissens in der Wissensbasis:** Unter dem Gesichtspunkt der Wiederverwendung und Pflege von Wissensträgern sind die Kodierung des erzeugten Wissens und die Veröffentlichung des gemeinsamen Materials essentiell.

Da es sich bei den Kompetenzen einzelner Mitarbeiter um datenschutzrechtliche Informationen handelt, sollten nicht alle Mitarbeiter uneingeschränkten Zugang zu allen Daten des Wissensmanagement-Systems haben, sondern lediglich die Teamrecruiter, die für ihre Arbeit einen Überblick über das gesamte Unternehmen benötigen.

Deiters schreibt in [DL00] über den Nutzen von IT-Unterstützung für den speziellen Fall der Teambildung. Er formuliert, dass derzeitige Systeme lediglich die Selektion einzelner Mitarbeiter ermöglichen, ein flexibles Team aber selektiv zusammengestellt wird, indem nacheinander die geeigneten und noch fehlenden Personen ermittelt werden. Hilfreich wäre eine Anwendung, die das gleichzeitige Zusammenstellen eines kompletten Teams ermöglicht. Dadurch würden nicht nur die einzelnen Qualifikationen für die Suche nach einem geeigneten Mitarbeiter angegeben, sondern die Fähigkeiten, die das komplette Team abzudecken hat. Das Sys-

tem sollte anschließend automatisch ein Team zusammenstellen, das allen Anforderungen genügt. Als Vorteil dieser Vorgehensweise sieht er, dass Konstellationen möglich sind, die bei einer sequenziellen Auswahl nicht ersichtlich wären. Seiner Meinung nach kann es sein, dass Mitarbeiter, die mehrere Fachgebiete abdecken, wobei sie eins alleine beherrschen, für Aufgaben eingesetzt werden, die auch von anderen Personen übernommen werden könnten. Eine sequenzielle Suche würde in diesem Fall die einzelne Fachkenntnis als nicht abgedeckt ermitteln, während eine leistungsfähige Gruppensuche solche Besonderheiten berücksichtigen kann.

3.1. Verwandte Arbeiten

Die folgenden Arbeiten, die ähnliche Ziele wie die vorliegende Ausarbeitung verfolgen, werden in zeitlicher Reihenfolge ihres Erscheinens beschrieben. Im anschließenden Abschnitt 3.2 werden dann die Anforderungen basierend auf den Ergebnissen der folgenden Arbeiten formuliert.

3.1.1. Knowledge Management System, AIAI

Jussi Stader und Ann Macintosh vom AIAI (Artificial Intelligence Application Institute) der Universität Edinburgh entwickelten 1999 ein Knowledge Management System [SM99], damit ein Unternehmen „weiß, wer was weiß“. Ihrer Meinung nach müssen Unternehmen in der Lage sein, ihre Kernkompetenzen und vielversprechende und ausbaufähige Fähigkeiten der einzelnen Mitarbeiter einzuschätzen. Außerdem müssen sie die für ihr Überleben wichtigen Skills und diejenigen Mitarbeiter identifizieren, die solche Fähigkeiten besitzen. Ebenfalls wichtig ist die effiziente und effektive Zuordnung von Mitarbeitern zu Projekten. Das sind die vier implementierten Unternehmensziele.

Am AIAI wurden bereits zuvor andere Workflowmanagement-Systeme entwickelt. Das hier beschriebene Knowledge Management System soll Funktionalitäten dieser Systeme mittels strukturierter Fähigkeiten (Ontologien) unterstützen und verbessern. Das System soll strategische und taktische Entscheidungsvorlagen sowie Planungshilfen liefern, wobei es im Speziellen folgende Potentiale besitzt:

- **Skill gap analysis:** Identifizieren von fehlenden oder wegfallgefährdeten Fähigkeiten; Erkennen des Unterschieds zwischen Ist und Soll bezüglich der Fähigkeiten im Unternehmen; Identifizieren der gefährdeten Kernkompetenzen (z.B. durch Pensionierung von Mitarbeitern)
- **Project team building analysis:** Identifizieren von Mitarbeitern, die die Anforderungen eines Projektes erfüllen; Identifizieren der Mitarbeiter, die bezüglich Projektanforderungen weitergebildet werden können
- **Recruitment planning:** Identifizieren des Mitarbeitertyps, der eingestellt werden muss
- **Training analysis:** Identifizieren von Mitarbeitern, die für Weiterbildungen geeignet sind

Diese Potentiale bilden die Anforderungen an das Knowledge Management System von Stader und Macintosh.

Modelle Stader und Macintosh beschreiben die notwendigen Modelle für ihr System, wobei sich zusammengefasst die folgenden relevanten Basiskonzepte ergeben: Kompetenzen, Mitarbeiter, Projekte und Organisation. Abbildung 3.1 zeigt die Relationen zwischen diesen Konzepten. Die Autoren haben sich für eine hierar-

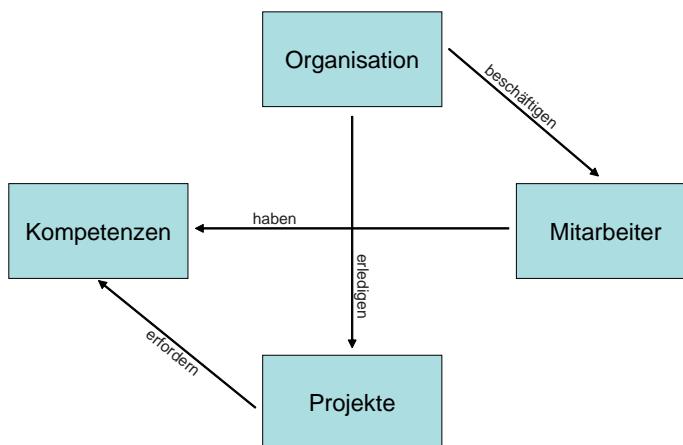


Abbildung 3.1.: Relevante Basiskonzepte, angelehnt an [SM99]

chische Strukturierung der Kompetenzen entschieden, da dadurch Spezialisierungen von Kompetenzen auf verschiedenen Ebenen möglich werden. Terme, die weiter unten in der Hierarchie stehen, sind spezifischer als Terme, die höher stehen.

Sie zeigen auf, dass viele Kompetenzen in zwei Komponenten aufgeteilt werden können: die Fähigkeit, etwas zu tun und die Dinge, mit denen es getan werden kann. Als Beispiel nennen sie „design software systems“, wobei „design“ die Fähigkeit an sich ist und „software systems“ der Bereich, auf den die Fähigkeit angewendet wird. Aus dieser Erkenntnis resultieren zwei verschiedene Hierarchien für das System: Anwendungsgebiete und Kompetenzen. Sie unterscheiden vier Haupttypen von Kompetenzen: technische, physische, Projektkompetenzen und Unternehmenskompetenzen. Darüber hinaus existieren verschiedene Erweiterungen für die genannten Hierarchien. Zu den allgemeinen Erweiterungen gehören etwa Einschränkungen der Anwendungsgebiete, die definieren, welche Kompetenzen für welche Sachbereiche anwendbar sind. Die Wissensbasis ist die Gesamtheit der intern im Unternehmen existierenden Fähigkeiten der Mitarbeiter und derer, die extern für das Unternehmen arbeiten. Erweiterungen bezüglich Anforderungen an Kompetenzen betreffen spezielle Projekte oder auch die gesamte Organisation. Mit ihnen soll beschrieben werden, wie viele Personen, die die notwendigen Fähigkeiten mit einem gewissen Erkenntnisstand und Erfahrungslevel besitzen, für ein Projekt gebraucht werden. Weiter gehend sollen Rollen definiert werden, die für eine bestimmte Kombination von Fähigkeiten stehen, die eine einzelne Person inne hat. Dadurch kann ausgedrückt werden, welche Fähigkeiten so wichtig sind, dass sie im Projekt von einer und derselben Person vertreten sein sollen.

Die Mitarbeiter sollen über die Kompetenzen hinausgehend über weitere Informationen beschrieben werden: Name, eindeutiger Bezeichner, Position im Unternehmen, Kontaktinformationen und aktuelle sowie zukünftige Aufgaben im Unternehmen.

Projekte werden über ihren Namen, Status (z.B. laufend, in Planung), Wichtigkeit und Relevanz fürs Unternehmen spezifiziert. Je wichtiger ein Projekt ist, desto priorisierter wird das zugehörige Project Staffing, also die Auswahl von geeigneten Mitarbeitern, behandelt. Die Relevanz eines Projekts gibt an, in welche Richtung sich die Kernkompetenzen eines Unternehmens entwickeln können.

Systemdetails Zur Erfüllung der Systemanforderungen wie unter 3.1.1 beschrieben, begannen Stader und Macintosh mit der Spezifikation der Ontologien, der Profile der Mitarbeiter und der Projekte. Die Ontologien werden mittels eines Ontology Manager gepflegt, dessen Aufgabe es einerseits ist, sicherzustellen, dass die Ontologien die im Unternehmen notwendigen Spezifikationen beinhalten und anderer-

seits die Wahrung der Kohärenz und Wohlgeformtheit der Ontologien. Die Autoren beschreiben ferner ihren Skill Manager, der für jede Kompetenz einen Namen, eine Beschreibung, die zugehörigen Anwendungsbereiche, die Vater- und Kindkompetenzen und den Grad der Spezialisierung erfasst.

Damit ein Knowledge Management System ausreichend akzeptiert und genutzt werden kann, müssen die Mitarbeiter ihre eigenen Kompetenzen leicht spezifizieren und ändern können. Der genannte Skill Manager erlaubt es auch den Mitarbeitern, ihre generellen Informationen (z.B. Name) zu editieren, als auch eine Zuordnung von Kompetenzen vorzunehmen. Diese Kompetenzen können mit einem erreichten Level, Erfahrungsgrad (Anzahl der angewendeten Monate) und der Information versehen werden, in welchen Projekten die jeweilige Kompetenz angewendet wurde.

Die Projekte werden durch einen Manager auf Projektebene spezifiziert. Als Hilfestellung können häufig vorkommende Projekttypen als Templates angelegt werden, die für andere Projekte einen Ausgangspunkt darstellen.

Nach der Spezifikation der relevanten Daten folgt die Bestimmung der Wissenslücken, der Mitarbeiter für Weiterbildungsmaßnahmen (möglicherweise in anderen Projekten) und die Erstellung von Recruitment Profilen der einzelnen Mitarbeiter und des Gesamtunternehmens. Stader und Macintosh definieren eine Wissenslücke für eine Fähigkeit, wenn sie gar nicht vorhanden ist, nicht in ausreichender Anzahl vorhanden ist, nicht in einem ausreichendem Maß an Erfahrung vorhanden ist, nicht für das entsprechende Anwendungsbereich genutzt werden kann oder zwar vorhanden ist aber kein Interesse der Nutzung besteht. Zur Schließung von Wissenslücken identifizieren die Autoren diejenigen Mitarbeiter, die sich für Weiterbildungsmaßnahmen eignen; diese als Trainees bezeichneten Mitarbeiter haben folgende Eigenschaften:

- Die Person besitzt relevante Kompetenzen: diese liegen in der Kompetenz-Hierarchie nah bei der Wissenslücke oder in einer Sequenz von aufeinander aufbauenden Fähigkeiten vor der fehlenden Kompetenz.
- Die Person hat Interesse an der Kompetenz geäußert.
- Die Person besitzt die fehlende Kompetenz zwar, aber nicht auf einem ausreichendem Level.
- Die Person hat die Kompetenz in einem anderen Anwendungsbereich.

- Die Person besitzt die meisten anderen Kompetenzen einer spezifischen Rolle.

Als Trainingsmaßnahmen können sich nach Meinung der beiden Autoren auch Projekte selber eignen, in dem z.B. die Wissenslücke „on-the-job“ durch einen Mitarbeiter geschlossen wird, der die fehlende Kompetenz auf einem zu niedrigen Level besitzt. Geeignet dafür sind Projekte, die nicht von zu großer Bedeutung und Wichtigkeit für das Unternehmen sind und in dem auch Mitarbeiter involviert sind, die die fehlende Kompetenz in einem ausreichenden Maß mitbringen. Auf diesem Weg kann der Trainee von dem geschulteren Mitarbeiter lernen. Ein Projekt sollte außerdem nie aus zu vielen Trainees bestehen, da sonst kein Lernprozess mehr stattfinden kann.

Als dritte und letzte Aufgabe des Knowledge Management Systems bleibt die Zuordnung von Mitarbeitern zu Projekten. Die Lösung dieser Aufgabe erfolgt in zwei Stufen: potentielle Mitglieder identifizieren und schließlich ein Projektteam selektieren. Zur Identifizierung von geeigneten Mitarbeitern durchläuft das System ebenfalls zwei Schritte: 1) Mitarbeiter mit relevanten Kompetenzen finden und Wissenslücken erfassen; 2) zur Schließung der Wissenslücken geeignete Mitarbeiter für Weiterbildungen finden. Während dieses Prozesses werden drei Ergebnislisten erstellt. Die erste Liste besteht aus Mitarbeitern mit relevanten Fähigkeiten, die im Projekt eingesetzt werden können, eine zweite Liste aus Trainees, die sich in jenen Kompetenzen weiterbilden können, die nicht in der ersten Liste abgedeckt werden und schließlich eine dritte Liste aus Fähigkeiten, die weder in der ersten Liste vorkommen noch durch Trainees der zweiten Liste geschult werden können. Abschließend folgt die Selektion eines Projektteams. Die Autoren schreiben, dass es schwer vorzustellen sei, wie sehr automatisiert die effektive Selektion eines Teams überhaupt sein kann. Selbst wenn alle notwendigen Kriterien vom System benutzt werden können, sei es doch zweifelhaft, dass der Benutzer bei der Vielzahl von Permutationen der Mitarbeiter versteht, was das System tut und welche der Teambesetzungen für ein spezielles Projekt zu wählen sei. Um diesem Umstand entgegenzuwirken, sollte der Benutzer in der Lage sein, ein Initialteam zu bilden, von dessen geeigneten Mitgliedern er überzeugt ist und das dann mit weiteren selektierten Mitarbeitern automatisch vervollständigt wird. Weiterhin hat der Benutzer die Möglichkeit, Mitarbeiter aus dem Prozess der Teamselektion auszuschließen. Zur Hilfestellung soll das System die geeigneten Teammitglieder in einem Ranking anhand bestimmter Kriterien ausgeben. Diese wären z.B. Passgenauigkeit der Projekt-Anforderungen, Güte der Komplettierung des Teams oder

Schwierigkeitsgrad der Zuordnung zu Projekten.

Ergebnisse Die beiden Komponenten der Entwicklung sind die Spezifizierung von Ontologien und die Implementierung von Knowledgemanagement Prozessen. Die Autoren schreiben, dass sie außer für Rollen und externe Kompetenzen Editoren zur Spezifikation der Basiskonzepte entwickelt haben. Sie implementierten je eine Version der „skill gap analysis“ sowie „training analysis“. Nicht implementiert wurden „recruitment planning“ und „project team building analysis“. Umgesetzt wurde lediglich die erste Stufe der Zuordnung von Mitarbeitern zu Projekten. Die Autoren schreiben, dass sie zur Realisierung der zweiten Stufe noch mehr Informationen über die Selektion von Teams innerhalb von Unternehmen benötigen. Die Implementierung der fehlenden Funktionalitäten sehen die Autoren als zukünftige Arbeit, ebenso wie die Verbesserung der Benutzerschnittstellen und eine bessere Anpassung an die „echte Welt“ - das betrifft z.B. die Skalierung für größere Unternehmen und der Umgang mit großen Ontologien. Als Fazit schreiben Stader und Macintosh, dass selbst mit der begrenzten Funktionalität der hohe Nutzwert eines solchen Knowledgemanagement-Systems gezeigt werden konnte.

Positiv sei vermerkt, dass die Autoren bereits 1999 vor der Definition des Semantic Web zur Erstellung eines Knowledgemanagement-Systems strukturierte Kompetenzen einer Ontologie verwenden haben. Allerdings handelt es sich bei diesen Ontologien noch nicht um die in Abschnitt 2.3.2 beschriebenen weitaus komplexeren Ontologien sondern vielmehr um einfachere Taxonomien. Interessant bei der Modellierung der Daten bleibt die beschriebene Trennung von Fähigkeit und Anwendungsgebiet. Das erscheint sinnvoll und praxisnah und sollte in der Applikation dieser Arbeit durchaus Anwendung finden, z.B. durch unterschiedliches Erfassen von Fähigkeiten und Aufgaben der Mitarbeiter. Ebenso bestätigend wird das Ranking der einzelnen Personen für Teamzusammensetzungen angesehen. Erst eine automatisierte Bewertung der einzelnen Teams macht eine Auswahl durch den Benutzer (den Teamrecruiter) möglich. Weiterhin positiv ist die Selektion von Mitarbeitern für Trainingsmaßnahmen, um Wissenslücken zu schließen. Diese Funktionalität soll auch der in dieser Ausarbeitung zu entwickelnde Prototyp für den Fall erfüllen, dass kein optimales Team gebildet werden kann. Wünschenswert wäre es gewesen, wenn die Autoren das entwickelte System um die fehlenden Funktionalitäten erweitert und erneut dokumentiert hätten. So hätte man noch von den

Ergebnissen der Teamselektionen profitieren können, aber anscheinend blieb es bei dem anfänglichen System.

3.1.2. Skillmanagement-System, BEKO

Hiermann und Höfferer von der Wiener BEKO Engineering und Informatik AG veröffentlichten 2005 ihr Skillmanagement-System (SMS) auf der 5. International Conference on Knowledge Management in Graz [HH05]. Nach eigenen Angaben unterscheidet sich ihr System von anderen, in dem sie erstens ein Netzwerk von Fähigkeiten benutzen und zweitens eine zeitliche Betrachtungsweise miteinbeziehen. Die Autoren unterscheiden Skill-von Knowledgemanagement, wobei der markanteste Unterschied ist, dass Skillmanagement implizit und Knowledgemanagement sowohl implizit als auch explizit abläuft. Die wichtigste Aufgabe des Knowledge-management sei das Darstellen des Wissens der Mitarbeiter in einer Datenbank, so dass die Verteilung und der Nutzen des unternehmensweiten Wissens optimiert werden. In ihrem Ansatz repräsentiert eine Kompetenz eine Assoziation von individuellen Fähigkeiten, Abbildung 3.2 zeigt die Klassifizierung der Konzepte. Falls

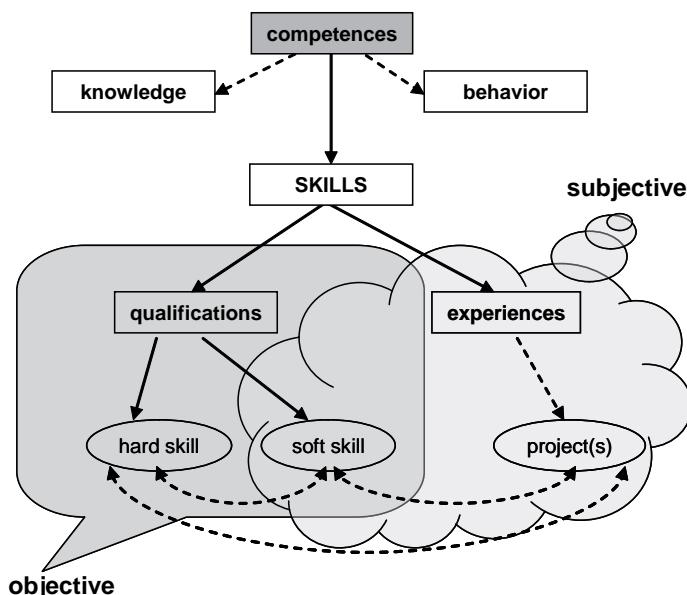


Abbildung 3.2.: Modell der Kompetenzen, angelehnt an [HH05]

mehrere Fähigkeiten häufiger für die Lösung einer Aufgabe benötigt werden, sind sie über eine Relation miteinander verbunden und bilden den „Kompetenzraum“

eines Mitarbeiters. Eine Teamzusammensetzung benötigt über die Beschreibung der Kompetenzen der Mitarbeiter hinausgehend Informationen wie Qualifikationsprofil, Erfahrungsprofil und Verfügbarkeit der Personen. Die Kompetenzen eines Mitarbeiters werden durch sein Wissen, seine Fähigkeiten und sein Verhalten spezifiziert. Die Autoren unterteilen Qualifikationen in „hard skills“ und „soft skills“ (siehe Abbildung 3.2), wobei „hard skills“ Qualifikationen objektiv bewerten und „soft skills“ aus einer subjektiven Selbst-und Fremdeinschätzung hervorgehen. Ein Qualifikationsprofil beschreibt das Wissen eines Mitarbeiters. Ein Erfahrungsprofil besteht aus Beschreibungen von involvierten Projekten und dort angewendeten Qualifikationen.

Um Teams für ein spezifisches Projekt zu bilden, muss eine detaillierte Beschreibung der Mitarbeiterfähigkeiten und der Verfügbarkeit der möglichen Teammitglieder vorhanden sein. Hiermann und Höfferer definieren die folgenden Attribute für Fähigkeiten:

- Name: eindeutiger Name der Fähigkeit
- Version: wird hauptsächlich für technische Qualifikationen benutzt und bezeichnet z.B. die Version eines Betriebssystems
- Funktion: Rolle (z.B. Benutzer, Programmierer), mit der die Fähigkeit benutzt wurde
- Erfahrung: Dauer der Anwendung der Fähigkeit
- zuletzt benutzt: Datum, an dem die Fähigkeit zuletzt genutzt wurde
- Erfahrungsgrad: beschreibt die Qualität einer Qualifikation über eine Zahl von 1 bis 5
- Verlinkungen: Menge von gemeinsam genutzten Fähigkeiten („Erfahrungsraum“)

Suchprozess Durch die Verlinkungen von Fähigkeiten untereinander entsteht der „Erfahrungsraum“ eines Mitarbeiters. Abbildung 3.3 zeigt solche Räume beispielhaft. Da es pro Mitarbeiter je einen dieser Räume gibt, kann eine Suche nach einem bestimmten Profil durchgeführt werden. Ausgangspunkt dafür ist wiederum ein „Erfahrungsraum“, der den Anforderungen eines Projekts entspricht. Die Suche teilt sich in zwei Schritte auf: zuerst werden alle Räume, die nun als Bäume

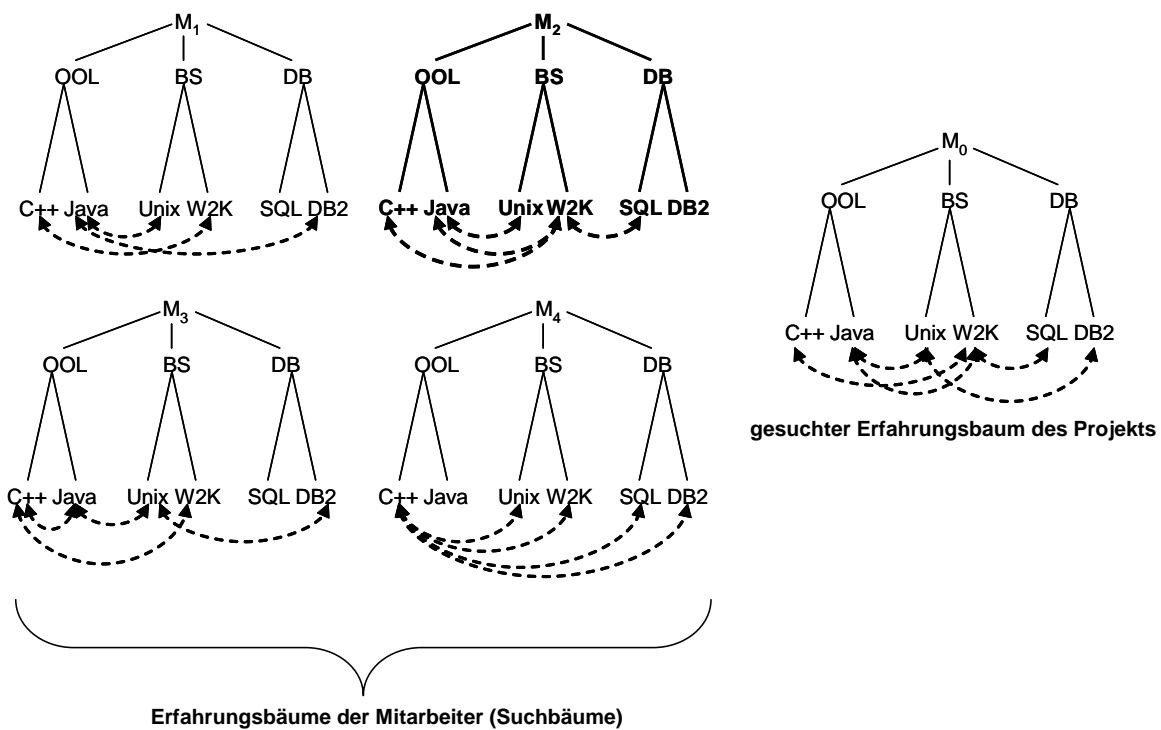


Abbildung 3.3.: Beispiel für „Erfahrungsräume“

betrachtet werden, selektiert, die mindestens die Blätter des Projektbaums haben und anschließend erfolgt eine Bewertung der gewählten Bäume. Eine Bewertung wird höher, wenn die Anzahl der Blätter des Projektbaums entspricht, ebenso wie die Anzahl der Verlinkungen jedes Blattes. In dem Beispiel aus Abbildung 3.3 wäre für den gesuchten Projektbaum M_0 der Baum M_2 die geeignete Wahl.

Fallstudie Hiermann und Höfferer testeten ihr System in einem IT-Unternehmen mit ca. 550 erfassten Mitarbeitern und rund 200 erfassten Projekten. Sie geben an, dass die „Ontologie der Fähigkeitsdatenbank“ rund 2500 Einträge enthielt, wobei nicht ersichtlich ist, was genau die Autoren mit Ontologie meinen, da sie eine Erläuterung schuldig bleiben. Die Kategorien auf der ersten Ebene der Unternehmensontologie seien Programmiersprachen, Datenbanken, Betriebssysteme, Transaktionsmonitore, Werkzeuge, Zertifikate, Software, Methoden, Sprachen und technisches Wissen. Der hierfür gebildete Baum hat zum damaligen Zeitpunkt eine Tiefe von 4.

Ein Mitarbeiter kann seine Profile selber erstellen und bearbeiten. Darüber hinaus hat er die Möglichkeit, maximal fünf seiner Fähigkeiten als primär einzustufen, um

damit auszudrücken, dass er ihre Nutzung bevorzugt.

Auf Managementebene wird die neue Position des Skill Managers eingeführt, für den eine spezielle Abteilungsontologie geschaffen wird. Diese Person kümmert sich darum, die Projektanforderungen entsprechend der Ontologie zu übersetzen. Der direkte Vorgesetzte eines jeden Mitarbeiters ist dafür verantwortlich, dass die Erfahrungs- und Qualifikationsprofile der Mitarbeiter aktuell bleiben. Als Zeitpunkte wird zweimal pro Jahr und nach Projektabschluss vorgeschlagen.

Auf Projektebene gibt es fünf Projektmanager, die das Skillmanagement-System nutzen können, um für ein Projekt geeignete Mitarbeiter zu finden. Dafür werden weitere Eingaben pro Mitarbeiter benötigt: Projektbeschäftigung, Urlaubsplanung, Krankheitsstatus, Trainingsmaßnahmen usw. Wie genau ein Projekt beschrieben wird, nennen die Autoren allerdings nicht. Nach eigenen Angaben werden folgende Vorteile während der Fallstudie gezeigt:

1. unternehmensweite Standardisierung der Fähigkeitennamen
2. leichtere, schnellere und exaktere Team- und Ressourcenplanung
3. sinkende Abbruchquote von Projekten
4. Kundenzufriedenheit erreicht schnell ein hohes Level
5. schneller Zugriff auf implizites Wissen
6. exaktere Trainingsplanungen
7. Unterstützung bei der Personalentwicklung
8. Koordinierung der Unternehmensstrategie

Die Autoren nennen auch die Nachteile ihres Systems. Allgemein sei es im Skillmanagement schwierig, eine ständige Datenaktualität zu gewährleisten. Im Speziellen sehen sie den Nachteil, dass ein Vollzeit-Skillmanager beschäftigt werden muss, der das System betreut. Die Koordination der speziellen Ontologien jeder Abteilung durch einen Spezialisten gestaltet sich ebenso schwierig. Vorgesetzte müssen zusätzliche Arbeitszeit aufwenden, um die Profile der ihnen zugewiesenen Mitarbeiter aktuell zu halten. Die Mitarbeiter selber müssen ebenfalls Zeit dafür aufwenden, die Autoren schätzen einen zusätzlichen Arbeitstag im Jahr. Ständige Betreuung und Pflege des Systems z.B. durch ein Service Center wird notwendig, was wiederum höhere Betriebskosten für das Unternehmen nach sich zieht.

Ergebnisse Die Autoren ziehen das Fazit, dass Skillmanagement-Systeme nur funktionieren können, wenn die Fähigkeiten so mit den Attributen gespeichert werden, wie von den Autoren beschrieben. An dieser Stelle sei positiv angemerkt, dass sie sich kritisch mit den Kosten ihres entwickelten Systems auseinanderge setzt haben. Es ist nicht von der Hand zu weisen, dass viele Personen Zeit für ihr Skillmanagement-System aufwenden müssen, statt sich ihrer vorrangigen Aufgabe zu widmen. Die Kosten dürften den Nutzen für ein Unternehmen weit überstei gen, da auch noch ein Experte beschäftigt werden muss. Die Autoren bleiben leider einige Erklärungen schuldig, so z.B. die Erläuterung ihrer Ontologie. Das System entstand in einer Zeit, in der Semantic Web und Ontologien bereits bekannt wa ren. Die Autoren scheinen aber nur den Begriff der Ontologie, nicht aber deren Ei genschaften zu nutzen. Sie beschreiben einzig und allein ihre „Erfahrungsräume“, die nicht mehr als statische Verlinkungen von gemeinsam genutzten Fähigkeiten sind. Dass diese Herangehensweise geeignet ist, sei an dieser Stelle angezweifelt. Was passiert z.B., wenn ein Mitarbeiter während des nächsten Einsatzes in einem Projekt bestimmt Fähigkeiten nicht gebraucht? Werden diese dann aus dem Baum gelöscht?

In der Modellierung der Fähigkeiten werden Eigenschaften mit Historie-Daten ver mischt. Wie wird aber verfahren, wenn solch ein Datensatz geändert werden muss? Das erklären die Autoren ebenfalls nicht. Darüber hinaus erscheint die Aufteilung von objektivem und subjektivem Wissen sinnlos bzw. falsch, denn Projekterfahrung wird subjektiviert und „soft skills“ objektiviert. Durch Recherche für die vorliegen de Ausarbeitung konnte jedoch nur das Gegenteil dieser Aussage festgestellt wer den. Gerade weil „soft skills“ nicht objektiv oder nur schwer objektiv bewertbar sind, wurde von ihrem Gebrauch für den Teamplanungsprozess abgesehen. Weiterhin ist das System auf den Bereich der Informationstechnologie beschränkt, was eine Anwendung in anderen Branchen erschwert und größere Anpassungen erfor dert. Für die Applikation dieser Arbeit kann der Umstand mitgenommen werden, dass die Mitarbeiter ihre Beschreibungen selbständig aktuell halten sollen. Es sollte für sie Motivation sein, dass sie durch aktuelle Daten wertvoller für das Unterneh men werden und häufiger in interessanten Projekten eingesetzt werden können.

3.1.3. KMDL, Universität Potsdam

An der Universität Potsdam entwickelte die Gruppe um Norbert Gronau, dem Inhaber des Lehrstuhls für Wirtschaftsinformatik und Electronic Government, die KMDL (Knowledge Modeling and Description Language) und das zugehörige Werkzeug K-Modeler. Skillmanagement kann für Gronau et al. [FSRG06] als ein Teil des Wissensmanagements definiert werden, der sich mit dem Wissen in den Köpfen der Mitarbeiter (stillschweigendes Wissen) beschäftigt. Als Aufgaben von Skillmanagement-Systemen nennen sie die Suche nach Experten, Personalbeschaffung und Personalentwicklung mit „Skill-Gap-Analysen“ und als dritten Schwerpunkt das Projektmanagement und Staffing. Letzteres mit dem Ziel, ein ausgewogenes Team zusammenzustellen, das in der Summe möglichst alle benötigten Kenntnisse in sich vereint, die zur Erledigung der Gesamtaufgabe notwendig sind. Problematisch sehen die Autoren die Erfassung und Beurteilung der einzelnen Mitarbeiterfähigkeiten, die Zuordnung und Bewertung einzelner Skills sowie die Aktualität der Daten.

Im aktuellsten Arbeitsbericht der Gruppe heißt es, die „KMDL in der Version 2.1 ist eine semiformale Modellierungssprache und dient der Visualisierung, Analyse, Bewertung und Verbesserung von Wissenskonversionen in wissensintensiven Geschäftsprozessen.“ [FF07] Wissensintensive Geschäftsprozesse würden sich durch folgende Kennzeichen auszeichnen [Gro06]:

- innovative Anteile, Kreativität
- hoher Entscheidungsspielraum, Autonomie
- Quellen- und Medienvielfalt
- nicht eindeutig bestimmbarer Prozessablauf
- viele Beteiligte
- nicht vollständig definierte Aufgaben

Wissenskonversionen zwischen stillschweigendem und explizitem Wissen treten nach Meinung der Autoren in Organisationen entlang der Geschäftsprozesse und zwischen den Prozessbeteiligten eines Unternehmens auf. Sie unterscheiden die folgenden vier Konversionstypen, die bildlich in Abbildung 3.4 gezeigt werden: Sozialisation (Erfahrungsaustausch, bei dem stillschweigendes Wissen entsteht),

Externalisierung (Artikulation von stillweigendem Wissen), Kombination (Verknüpfung von explizitem Wissen zu neuem expliziten Wissen) und Internalisierung (Eingliederung von explizitem in stillschweigendes Wissen).

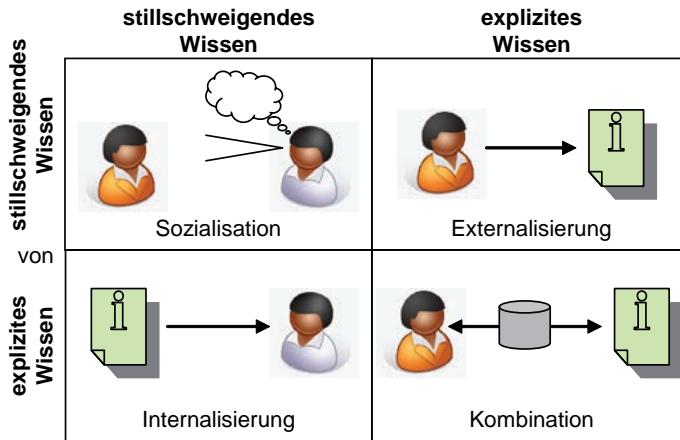


Abbildung 3.4.: Wissenskonversionen bei KMDL, angelehnt an [Gro06]

Sichten Die KMDL ist ein Framework zur Modellierung und ganzheitlichen Integration der Geschäftsprozesse und des Wissensflusses in Unternehmen. Dazu werden in der KMDL zwei Sichten definiert: Prozesssicht und Aktivitätssicht.

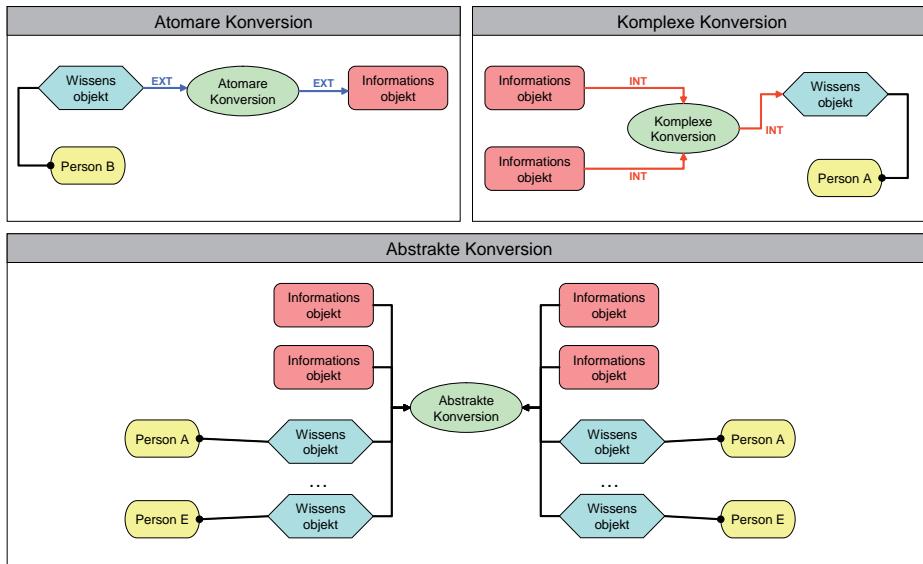


Abbildung 3.5.: Modellierung der Konversionsarten mit verfügbaren Objekten

Die Prozesssicht beschreibt den relevanten betrieblichen Ablauf aus der Perspek-

tive von Tätigkeitsfolgen (Prozessschritten). Auf dieser Ebene ist erkennbar, welche Aufgaben nacheinander abgearbeitet werden müssen und welche Alternativen existieren. Außerdem werden in der Prozesssicht den Aufgaben die Ressourcen zugeordnet, die zur Bearbeitung der Aufgabe genutzt werden.

Die Aktivitätssicht erlaubt eine detaillierte Beschreibung der ablaufenden Wissensumwandlungen zwischen Informations- und Wissensobjekten bei der Aufgabenbefüllung. Damit werden hier Wissenskonversionen auf einer feingliedrigen Ebene beschrieben. Auf Grund des Modellierungsaufwandes wird nur für wissensintensive Aufgaben eine Modellierung der Aktivitätssicht angestrebt. In dieser Sicht werden atomare, komplexe und abstrakte Konversionen unterschieden (siehe Abbildung 3.5).

Vorgehensmodell Das verwendete Vorgehensmodell besteht aus sieben Phasen. Nach der „Projektanbahnung“ folgt die Identifikation wissensintensiver Geschäftsprozesse. In Phase 3 erfolgt deren Aufnahme in drei Schritten. Zunächst werden die wesentlichen Prozesse aufgenommen. Im zweiten Schritt erfolgt die Nacherfassung und Modellierung der erhobenen Informationen. Die erzeugten Prozessmodelle werden dem Projektpartner im dritten Schritt zur Überprüfung übergeben. Diese Schritte werden iterativ bis zur vollständigen Erhebung des Untersuchungsgebiets im gewünschten Abstraktionsgrad durchgeführt.

Es folgt Phase 4 des Vorgehensmodells, in der die aufgenommenen Prozesse auf ihre Stärken und Schwächen hin analysiert werden. Darauf folgt die Erarbeitung eines Soll-Konzepts. In Phase 6 werden je nach definiertem Ziel mit dem Projektpartner die Verbesserungsmaßnahmen umgesetzt. Zum Abschluss werden die Ergebnisse anhand der Zielvorgabe evaluiert.

„Staffing - Konzeption und prototypische Implementierung einer Teambildungskomponente“

Uwe Rüßbüldt aus der Arbeitsgruppe um Gronau entwickelt in [RÖ6] eine Teambildungskomponente für die KMDL. Voraussetzungen dafür sind:

- semantisches Netz zur Darstellung der Konzepte (Anforderung, Wissen)
- Beziehung zwischen den Konzepten für den Inferenzmechanismus

- Anforderungen für jede Aufgabe
- Definition von Mitarbeiterprofilen

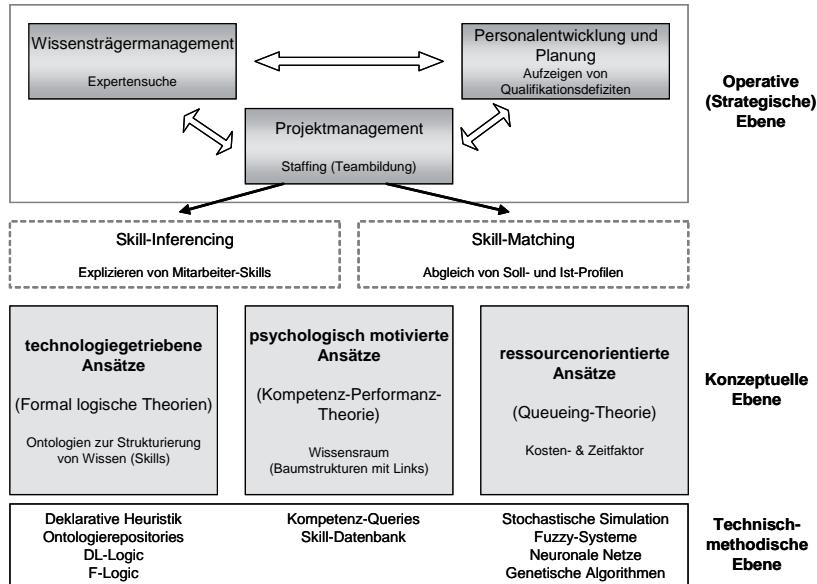


Abbildung 3.6.: Staffing-Ansätze im Überblick, angelehnt an [RÖ6]

Der Autor arbeitet diverse Ansätze zum Staffing heraus, wie zusammengefasst in Abbildung 3.6 zu erkennen ist. Er definiert Staffing als die Besetzung von Projektteams. Er unterteilt die genannten Ansätze auf konzeptioneller Ebene in technologiegetriebene, psychologisch motivierte und ressourcenorientierte Ansätze. Letzteres kann als reines Planungsverfahren angesehen werden, da keine Informationen über die Teammitglieder im Einzelnen berücksichtigt werden. Aus diesem Grund wurde dieser Ansatz nicht weiter verfolgt. In den Bereich der psychologisch motivierten Ansätze gehört auch die unter 3.1.2 beschriebene Arbeit von Hiermann und Höfferer. Rüßbüldt kommt zu dem Schluss, dass zur Konzeption des Teambildungsprozesses sowohl ein ontologiebasierter als auch ein psychologisch motivierter Ansatz geeignet sind. Beide Vorgehensweisen erlauben seiner Meinung nach eine rechnergestützte formale Abbildung von Kompetenz oder inhaltlich etwas eingeschränkten Skill-Profilen. Technologisch anspruchsvoller seien die formal ontologiebasierten Anwendungen. Die Entscheidung ist dennoch für die Nutzung eines semantischen Netzes gefallen und die Definition und Nutzung eines taxonomischen Ähnlichkeitsmaßes bildet eine Grundlage der notwendigen Bewertungsfunktion von Personengruppen in seiner Arbeit.

Im ersten Schritt wird ein semantisches Netz erstellt, in dem die Fähigkeiten hierar-

chisiert werden. Dann werden Qualifikationsprofile erzeugt, die dauerhaft in einem Repository abgespeichert werden. Basierend auf KMDL folgt die Auswahl der zu erledigenden Aufgaben im Projekt und zusätzlicher Bedingungen. Aus diesen Eingaben errechnet der Teambildungsalgorithmus Teamzusammensetzungen, die z.B. als Reports dem Benutzer zur Verfügung gestellt werden.

Teambildungsalgorithmus Der Algorithmus zur Suche nach geeigneten Mitgliedern für Teams bildet das zentrale Element bei der Lösung von Rüßbüldt. Der Algorithmus soll nach eigenen Angaben:

- ein „gleichzeitiges“ Zusammenstellen von Teams ermöglichen und dem nachteiligen Effekt des sukzessiven Zusammenfindens der Teammitglieder entgegenwirken
- lediglich die an das Team gestellten Anforderungen berücksichtigen
- konfigurierbar sein
- einen Inferenzmechanismus nutzen, um verborgenes Mitarbeiterwissen mit einzubeziehen

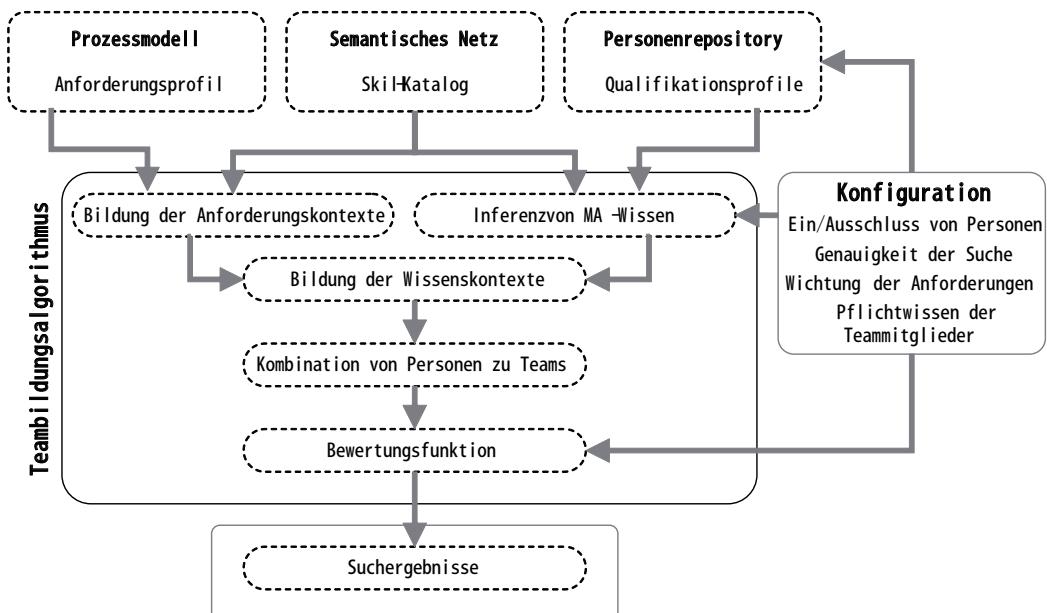


Abbildung 3.7.: Teambildungsalgorithmus, angelehnt an [RÖ6]

Abbildung 3.7 zeigt den Ablauf des Algorithmus. Für die Suche werden zunächst entsprechend des Anforderungsprofils alle Anforderungen analysiert, die das Team

erfüllen muss. Anschließend wird für jede zu berücksichtigende Person deren Wissenskontext bezüglich einer gestellten Anforderung ermittelt. Die Menge der Personen, die einen „nichtleeren“ Wissenskontext bezüglich einer Anforderung besitzen, werden sukzessive zu Teams vervollständigt. Im letzten Schritt folgt die Auswertung, in der die gefundenen Teams anhand automatisch generierter Diagramme bewertet werden.

Suchstrategien Bevor die Suche gestartet wird, hat der Benutzer die Möglichkeit, Parameter der Suche zu beeinflussen. Diese Parameter betreffen die Auswahl der Mitarbeiter, aus denen ein Team gebildet werden soll. Anhand nicht-relevanter Qualifikationen oder z.B. auch Krankheit von Mitarbeitern werden Mitarbeiter aus dem Suchprozess ausgeschlossen und so der Suchraum von vornherein eingeschränkt. Zusätzlich kann eine Gewichtung der einzelnen Anforderungen vorgenommen werden. So werden bestimmte Fähigkeiten höher priorisiert als andere. Außerdem kann das direkte Vorhandensein einer gesuchten Fähigkeit bei allen Personen des Teams als weitere Einschränkung vorgegeben werden. Drittens hat der Benutzer die Möglichkeit, die Genauigkeit der Suche zu bestimmen. Dieser Parameter spiegelt das im semantischen Netz genutzte Abstandsmaß zweier Konzepte wieder und gibt an, in welchem „Umfeld“ einer Anforderung Konzepte ermittelt werden. Das Anforderungskonzept und die Menge der benachbarten Konzepte spannen dann den Wissensraum einer Person bezüglich dieser Anforderung auf. Personen, die dann zu den Anforderungen relevante Fähigkeiten besitzen, bilden einen nicht-leeren Wissenskontext. Eine „sehr genaue“ Suche führt zu qualitativ hochwertigen aber wenigen, möglicherweise gar keinen Teams. Eine gesonderte Expertensuche und weitere Teambildungsstrategien wie „Finde das optimale Team“ oder „Finde alle Teams mit minimaler Personenanzahl“ ergänzen die Funktionalität.

Als Experte bezeichnet der Autor eine Person, die über spezielle Fähigkeiten verfügt oder über umfangreiches Wissen auf einem oder mehreren Fachgebieten. Da seiner Meinung nach die Identifizierung solcher Experten kontextabhängig ist und eine objektive Ermittlung und Erfassung der Kriterien für Expertentum unmöglich sei, wird nicht zwischen Experten und anderen Mitarbeitern unterschieden. Ein „Expertenteam“ besteht dann nur aus Personen, die genau alle Anforderungen mit mindestens dem geforderten Level abdecken.

Bewertung von Teams Um ein Team zu erhalten, müssen alle möglichen Personenkombinationen gefunden werden, die alle Aufgabenanforderungen erfüllen. Dazu wird sukzessive je eine Person mit nicht-leerem Wissenskontext dem Team hinzugefügt, bis alle Anforderungen erfüllt sind. Dadurch entsteht potentiell eine große Anzahl von Teams. Zudem wird für jedes Team jede Konstellation von Person-Aufgaben-Zuordnung generiert. Zur Kompensation dieser beiden Mengen wird nach dem Kombinieren von Personen eine sofortige Bewertung jedes Teams vorgenommen. Der Autor untersuchte verschiedene Evaluationsmethoden, darunter auch die Bewertung der „Erfahrungsbäume“ von Hiermann, Höfferer aus dem Abschnitt 3.1.2. Manko der Methoden ist durchgängig, dass sie auf die Bewertung des Skill-Matchings zwischen Einzelpersonen und Stellenprofilen fokussiert sind und keinerlei Hinweise auf die Bewertung mehrerer Personenprofile geben. Rüßbüldt definierte seine Kriterien zur Bewertung von Teams deshalb folgendermaßen:

1. Distanzmaß: Konzepte in einem Wissenskontext eines Mitarbeiters stellen alternativ nutzbare Fähigkeiten zu einer Anforderung dar. Je geringer das Abstandsmaß zwischen Alternativen und Anforderung ist, desto höher die Ähnlichkeit zwischen den Konzepten.
2. Alternativen: Die Menge der Alternativen, also die Mächtigkeit des Wissenskontexts, eines Mitarbeiters ist ein Indiz für eine mögliche Eignung. Je größer die Menge, desto breiter ist die Wissensbasis einer Person.
3. Skill Gewichte: Der Benutzer kann den Anforderungen eines Projekts die Gewichte „normal“, „hoch“ und „sehr hoch“ zuweisen. Ein Team sei höher zu bewerten, wenn es eine bessere Alternative bezüglich einer Anforderung höheren Gewichts bereitstellt.
4. Ausprägung der Fähigkeiten: Ein Team sei zu bevorzugen, wenn dessen Erfahrungswerte in der Summe über den geforderten Anforderungsausprägungen liegen. Allerdings führt ein „Übererfüllen“ nicht zu einer wesentlich höheren Bewertung.
5. Team-Effizienz: Je mehr Personen nötig sind, um die Anforderungen zu erfüllen, um so „uneffizienter“ ist ein Team.

Diese Kriterien werden basierend auf einer Bewertungsfunktion zu einem gemeinsamen Wert verdichtet. Dieser wird anschließend aufbereitet und mit anderen Aus-

wertungsformen kombiniert. Zur Repräsentation entschied sich der Autor für ein Punktesystem, dessen „Justierung“ er anhand von Testläufen vorgenommen hat, da sich keinerlei Richtwerte finden ließen. Die ersten drei Bewertungsschritte werden für jede Anforderung durchlaufen und anschließend zu einer Summe aufaddiert. Danach erfolgt die Bestimmung der Effizienz eines Teams, in dem die Teamgröße zur Anzahl der Anforderungen in Relation gesetzt wird. Das bedeutet, dass ein Team eine höhere Bewertung erhält, wenn die Anzahl der Personen geringer als die Anzahl der erfüllten Anforderungen ist. Die Ergebnisse werden verschiedenartig dargestellt mittels diverser Diagramme.

Ergebnisse Rüßbüldt schreibt, dass er keinen Praxistest mit seinem System durchgeführt hat. Daher bleibt es offen, wie sich die Laufzeit der Anwendung für ein großes Unternehmen verhält. Sein System ist nicht ohne KMDL benutzbar, wenn auch der konzeptionelle Teambildungsalgorithmus ohne KMDL für andere Systeme adaptierbar ist. Der Benutzer muss im ersten Schritt immer erst einen KMDL-Prozess erzeugen und die ersten Phasen des Vorgehensmodell durchlaufen, bis eine Teamplanung beginnen kann. Während des Projektes und danach wird die Modellierung mit KMDL immer wieder angepasst, so dass die Zuordnung der einzelnen Objekte der Prozess- und Aktivitätssichten konsistent bleiben. Diese Vorgehensweise ist stark prozessorientiert, was durchaus seine Berechtigung hat. Dadurch wird allerdings eine Teamplanung gerade vor Beginn des Projektes erschwert, was den Fokus der vorliegenden Arbeit bildet. Nachteilig bleibt auf jeden Fall, dass der Autor ein semantisches Netz und keine Ontologie für seine Lösung benutzt hat. Die größere Ausdrucksstärke und Erfassung mehrerer Informationen über die Unternehmensstruktur sind mit einem semantischen Netz nicht kompensierbar. Rüßbüldt beläßt diesen Punkt für eine eventuelle zukünftige Weiterentwicklung seines Systems.

Positiv bleiben die Bewertung der Teams und die Einschränkung der Personenmenge, aus der Teams generiert werden sollen. Das ist gerade für Wissensmanagement-Systeme mit Teamplanungsfunktionalität großer Unternehmen ein wichtiger Laufzeitfaktor. Problematisch erscheint allerdings, dass die Einschränkungen durch den Benutzer auch hinsichtlich Mitarbeiterqualifikationen erfolgen sollen.

Als schwierig wird der Parameter „Genauigkeit der Suche“ eingeschätzt. Der Benutzer muss ihn einstellen und nimmt somit direkt Einfluss auf den Wissenskontext einer Person. Das bedeutet, je genauer eine Suche sein soll, um so kleiner wird ein

Wissenskontext einer Person, also die Anzahl der relevanten Fähigkeit zu einer Anforderung. Es steht in Frage, ob der Benutzer immer weiß, wie er diesen Parameter zu bestimmen hat, um eine sinnvolle Suche durchzuführen. Weiterhin entspricht das sukzessive Füllen der Teams eher dem permutativen Algorithmus der vorliegenden Arbeit (siehe Teil 4.4), der zu Test- und Vergleichszwecken implementiert wird.

3.1.4. KOWIEN, Universität Duisburg-Essen

Das Projekt Kooperatives Wissensmanagement in Engineering-Netzwerken (KOWIEN) befasst sich mit der Unterstützung der arbeitsteiligen Erfüllung wissensintensiver Engineering-Aufgaben in Netzwerken durch computerbasierte Arbeitsmethoden. Für die Koordinierung dieser Aufgaben sieht Zelewski [Zel04] es als wichtig, diejenigen Kompetenzen zusammenzuführen und aufeinander abzustimmen, die als Wissen zur Aufgabenerfüllung benötigt werden. Kompetenzen beschreibt er „allgemein als handlungsbefähigendes Wissen [...] Kenntnisse und Fertigkeiten werden nicht um ihrer selbst Willen erworben und angewendet.“ Die Gesamtaufgabe, Wissen über Kompetenzen zu managen, umfasst im KOWIEN-System drei Teilaufgaben:

1. Identifizierung derjenigen Kompetenzen, die zur Erfüllung von wissensintensiven Engineering-Aufgaben erforderlich sind und die Zusammenführung der zur Aufgabenerfüllung erforderlichen Kompetenzen in einem aufgaben-spezifischen Kompetenzprofil
2. Identifizierung der aktuellen Kompetenzträger und die Erfassung ihrer Kompetenzen in personenspezifischen Kompetenzprofilen
3. Zuordnung von Kompetenzprofilen zu Engineering-Aufgaben als „Matching“ zwischen den Profilen der Personen und den Aufgaben

In [AZ03] schreiben Alan und Zelewski, dass insbesondere im Projektgeschäft Kompetenzprofilen eine besondere Bedeutung zukommt. Durch sie werde der Projektleitung ermöglicht, für wissensintensive Teilprozesse eine Zuordnung zwischen prozessinduzierten Wissensanforderungen und verfügbaren Kompetenzen zu gewährleisten. Kompetenzprofile unterstützen damit auch den Teilprozess der Teambildung, indem sie Auskünfte darüber ermöglichen, welche potenziellen

Teammitglieder welche Kompetenzen in ein Team einbringen könnten. Somit kann ein Teamrecruiter die Zusammensetzung eines Teams auf der Grundlage einer breiten Wissensbasis vornehmen. Kompetenzprofile haben ihrer Meinung nach auch während eines Projekts positive Effekte. Beispielsweise kann bei einem Ausfall eines Teammitglieds schnell nach Ersatz gesucht werden, der zumindest über die im Projekt benötigten Kompetenzen des Teammitglieds verfügt. Des Weiteren würden sich Kompetenzprofile auch für die Suche nach ergänzenden Teammitgliedern anbieten, wenn in einer Phase der Projektdurchführung erkannt wird, dass das Projektteam für die Problembewältigung nicht mehr ausreicht. Aus Sicht des Autors von [Zel04] können Unternehmen der zunehmenden Dynamik und Turbulenz in ihrem Umfeld nur mit flexiblen Organisationsformen entgegenwirken, wobei sich die Projektorganisation durchgesetzt hat. Die Planung und Steuerung solcher Projekte mit Projektmanagement-Techniken wie z.B. Netzplantechnik und Workflowmanagement-Systemen wird durchaus zufriedenstellend beherrscht. Die Zusammenstellung von Projektteams als weitere Aufgabe des Projektmanagements wird hingegen nur mangelhaft mit praxistauglichen Instrumenten unterstützt. Aus technologischer Perspektive soll mit dem KOWIEN Projekt aufgezeigt werden, wie sich Ontologien in der betrieblichen Praxis für Zwecke des Managements von Wissen über Kompetenzen, Kompetenzträger und Kompetenzprofile erfolgreich konstruieren und anwenden lassen. Für die Autoren stellt das Knowledge-Level-Engineering eine der wichtigsten und derzeit fruchtbaren „Schlüsseltechniken“ auf dem Gebiet der Erforschung Künstlicher Intelligenz und der Informatik dar.

Unterstützung der Projektarbeit In [BH04] beschreiben Bäumgen und Hübbers, wie mit dem KOWIEN-Prototyp die Projektarbeit unterstützt werden kann. Das ontologiebasierte Kompetenzmanagement-System basiert auf *infonea®* („information network architecture“), das von einem Bonner IT-Unternehmen entwickelt wurde.

Das KOWIEN-Vorgehensmodell sah im ersten Schritt die Erstellung der Unternehmensontologie vor. Dazu erfolgt die Identifikation der wesentlichen Wissensträger und Dokumente, um die Kompetenzbegriffe und die Relationen zwischen den Begriffen in strukturierter Form zu erfassen. Der Vorteil der webbasierten Architektur des KOWIEN-Prototyps zeigt sich bei der kontinuierlichen Erweiterung und Pflege der Kompetenzontologie im laufenden Betrieb. Ein verantwortlicher Ontologie-Administrator übernimmt diese Aufgaben.

Projekte werden zentral erfasst und beschrieben, indem ihnen Kompetenzen zugeordnet werden, die für ihre Durchführung erforderlich sind. Es gibt zwei Möglichkeiten, ein Projekt ins System einzugeben. Einerseits ist die Eintragung unabhängig vom KOWIEN-Geschäftsprozessmodell möglich. Zum anderen kann es bei seiner Eingabe direkt an das entsprechende Unternehmensmodell gekoppelt werden, d.h. es wird festgelegt, welche Prozesse des Unternehmensmodells bei diesem Projekt durchlaufen werden. Dieses prozessorientierte Vorgehen hat den Vorteil, dass dem Projekt automatisch die erforderlichen Kompetenzen zugeordnet werden, da sie für die Prozesse erfasst worden sind.

Die Autoren schreiben, dass eine Integration in die Geschäftsprozesse für die Einführung eines Kompetenzmanagement-Systems erfolgentscheidend sei. Auch sind sie der Meinung, dass man bei der Pflege der Kompetenzprofile auf die Mitwirkung jedes einzelnen Mitarbeiters angewiesen ist. Das Unternehmen sollte sich deshalb darum bemühen, diese Tätigkeit in den Arbeitsalltag der Mitarbeiter zu integrieren und sie zu motivieren, durch ihre aktive Mitarbeit zu einer hohen Qualität und Aktualität der Datenbasis beizutragen.

Der erste Schritt des Projektmanagements bildet in der Regel die Bestimmung eines Projektleiters. Um einen geeigneten Projektleiter zu finden, sind bei der Suche sowohl die fachliche Eignung als auch die organisatorischen Rahmenbedingungen wie Zeitplanung oder Projektstandort zu berücksichtigen. Der Projektleiter kann dann damit beauftragt werden, das Projektteam zu bilden, also Mitarbeiter zu suchen, die zusammen mit ihm die Kompetenzanforderungen des Projekts erfüllen. Der Teamleiter, also das Teammitglied mit leitender Funktion innerhalb des Teams, nimmt in diesem Fall die Rolle des Teamrecruiters ein. Die prozessorientierte Arbeitsweise im KOWIEN-Prototyp erfordert die Auswahl der projektrelevanten Prozesse und damit spezifischer Aufgaben. Die für die Prozesse erfassten erforderlichen Kompetenzen sind im Prinzip diejenigen Kompetenzen, die eine Person haben muss, die in den entsprechenden Prozessen eine ausführende oder eine mitwirkende Rolle übernimmt. Das System schlägt Mitarbeiter und ihre Rollen vor, in dem es überprüft, welche Rollen in vorangegangen Projekten ausgeführt wurden.

Ergebnisse Die im KOWIEN Projekt konzipierten, prototypisch implementierten und evaluierten computerbasierten Instrumente für das Management von Kompetenzen umfassen grob drei Komponenten:

1. ein computerbasiertes Vorgehensmodell für die Erstellung und Anwendung ontologiebasierter Kompetenzmanagement-Systeme
2. ein prototypisches Software-Tool für ontologiebasierte Kompetenzmanagement-Systeme
3. eine E-Learning-Umgebung zur Unterstützung der praktischen Anwendung des Vorgehensmodells und des Software-Tools

Da der entwickelte Prototyp bereits in Unternehmen benutzt wird, kann von einer Praxistauglichkeit des Systems ausgegangen werden. Wie bereits bei der KMDL aus Abschnitt 3.1.3 gesehen, nutzt auch der KOWIEN-Prototyp eine unternehmensabhängige Prozesssicht zur Erfüllung der Wissensmanagement Aufgaben. Die beschriebenen Vorteile können dann genutzt werden, wenn eine Applikation speziell für ein Unternehmen entwickelt wird. Im Fall der vorliegenden Arbeit jedoch wird eine unternehmensunabhängige Anwendung im Hinblick auf Teamplanung entwickelt und deshalb auf die Modellierung von Geschäftsprozessen verzichtet. Der Teamleiter eines Projekts sucht dessen Mitglieder aus, d.h. er übernimmt damit die Rolle des Teamrecruiters. Bisher wurde davon ausgegangen, dass es sich bei den beiden Rollen um verschiedene Personen handelt. Aber [Zel04] zeigt auf, dass auch diese Vorgehensweise möglich ist. Offen bleibt dabei allerdings die Frage, wie geeignet die Zusammenstellung eines Teams wirklich ist, die durch ein im Team involviertes Mitglied gewählt wurde. Vorstellbar ist durchaus, dass der Teamleiter von vornherein Personen aufgrund von zwischenmenschlichen Problemen nicht mit in die Betrachtung einer möglichen Teamzugehörigkeit einbezieht, obwohl sie fachlich geeignet wären. Dieses Problem wird auch von Biesalski in [Bie06] angesprochen. Vorbildlich wirkt die Auseinandersetzung mit der Erfassung von Kompetenzprofilen. Die Mitarbeiter haben diverse Möglichkeiten, ihre Kompetenzen und deren Herkunft zu beschreiben. Diese komplexe Datengrundlage garantiert eine qualitativ hochwertige Suche nach Kompetenzträgern. Inwieweit die Nutzung der genannten ausführenden und mitwirkenden Rollen erfolgversprechend ist, wurde von den Autoren nicht belegt. Die Bedeutung von Rollen in Teams im Zusammenhang der vorliegenden Ausarbeitung wird in Abschnitt 4.2 beleuchtet. Leider fehlt eine konkrete Beschreibung der Teambildung mit KOWIEN bezüglich einer algorithmischen Vorgehensweise. Es werden Personen mit ihren Kompetenzen einzeln betrachtet und anhand der Projektanforderungen sortiert nach Geeignetheit dem Teamleiter präsentiert. Der Nachteil dabei ist, dass kein Vergleich zwi-

schen Projektanforderungen und einer Menge von Teammitgliedern statt findet, sondern „nur“ viele Vergleiche zwischen Einzelpersonen und den Anforderungen. Der Teamleiter wird sich im Anschluss bzw. in der Selektionsphase mutmaßlich für die Personen entscheiden, die besser bewertet wurden. Es ist ersichtlich, dass diese Art der Mitgliederselektion nicht in jedem Fall sicherstellt, dass alle notwendigen Kompetenzen in ausreichendem Maße in der Teamzusammensetzung vorhanden sind.

3.1.5. ePeople-System, DaimlerCrysler AG

Im März 2003 begann bei der DaimlerCrysler AG im Werk Wörth die Umstellung der dezentral host-basierten verteilten Personalsysteme auf ein Client-Server-basiertes Personalsystem mit dem internen Namen ePeople. Eines der Hauptziele dabei war laut Biesalski in [Bie06] die Etablierung von Standardprozessen. „Diese Standardprozesse [...] sollten zu einer erhöhten Transparenz der Personalarbeit führen, wie auch entsprechende Kostensenkungspotentiale im Personalbereich eröffnen, wenn Prozesse schlanker, nachvollziehbarer und systemunterstützt letztendlich schneller ablaufen, als vorher.“

Im Rahmen des vorgestellten Systems werden verschiedene Profiltypen definiert, die im Kompetenzkatalog vorkommen können. Grundsätzlich bestehen die Profile aus einer Menge einzelner mit einer Gewichtung versehener Kompetenzen. Biesalski verwendet eine vierstufige Gewichtung: 1-Grundkenntnisse, 2-erweiterte Grundkenntnisse, 3-Experte und 4-Trainer. Es werden folgende Profile verwendet:

- **Stellenanforderungsprofil:** umfasst die Kompetenzen, die für die optimale Erfüllung der Aufgaben einer Stelle benötigt werden.
- **Mitarbeiterkompetenzprofil:** erfasst die Kompetenzen eines Mitarbeiters. Es entsteht durch Selbst- und/oder Fremdbeurteilung.
- **Differenzkompetenzprofil:** entsteht aus einem Vergleich eines Stellenanforderungsprofils mit einem Mitarbeiterkompetenzprofil und enthält diejenigen Kompetenzen, die zur Erreichung der Stellenanforderung fehlen.
- **Schulungsanforderungsprofil:** spezifiziert, welche Voraussetzungen erfüllt sein müssen, um an einer Weiterbildungsmaßnahme teilnehmen zu können.

- **Schulungskompetenzprofil:** gibt an, welche Kompetenzen mit welcher Gewichtung in einer Schulung erworben wurden.

Die Datenhaltung des entwickelten Personalentwicklungssystems wird zentral in einem Human Resource Data Warehouse (HRDW) vorgenommen. Die einzelnen technisch wie inhaltlich heterogenen Datenquellen werden über einen Extraktions-, Lade- und Transformationsprozess in das HRDW geladen. Dadurch sind alle Daten an einem Ort verfügbar. Die Ontologie bezieht alle benötigten Daten direkt aus den entsprechend zugeordneten Feldern des HRDW und bildet einen semantischen „Überbau“ oberhalb des HRDW. Zusätzlich zu diesen Daten sind in der Ontologie Zusammenhänge modelliert, die man nicht im HRDW vorfindet.

Das entworfene Personalentwicklungssystem besteht aus mehreren Modulen. Ziel des Moduls Nachfolgeplanung ist es, ein IT-System zu konzipieren, welches Führungskräfte wie auch Mitarbeiter des Personalbereichs bei der Nachfolgeplanung unterstützt. Über die Integration eines ontologiebasierten Kompetenzkataloges besteht die Möglichkeit, mit Hilfe eines „Matching“ zwischen dem Stellenanforderungs- und den Mitarbeiterkompetenzprofilen geeignete Nachfolgekandidaten zu finden.

Hauptziele des Moduls Bildungsplanung bilden die Entlastung und Unterstützung der in diesem Bereich verantwortlichen Person bei der Bildungsbedarfsanalyse und der Bildungsplanung. Diese Person legt fest, wo er die strategischen Schwerpunkte in seiner Abteilung in den nächsten Jahren sieht. Zur Strukturierung dieser Informationen wird die Ontologie benutzt, die eine Gewichtung von Kompetenzen zulässt. Anschließend ermittelt die Anwendung aus einem Vergleich zwischen den Stellenanforderungsprofilen und den Mitarbeiterkompetenzprofilen eine eventuelle Differenz in Form eines Differenzkompetenzprofils, das dann dazu benutzt werden kann, nach geeigneten, den Bedarf deckenden Weiterbildungsmaßnahmen zu suchen.

Im semantischen Datenmodell kommen folgende Basiskonzepte vor: Unternehmensstruktur, Mitarbeiter, Bildung, Bewerber, Kompetenzen und Zeit. Abbildung 3.8 zeigt die beispielhafte Ausprägung des Konzepts Unternehmensstruktur. Dieses Modell ermöglicht beispielsweise die Gruppierung von Mitarbeitern zu Teams, die Modellierung der organisationalen Struktur oder die Zuordnung von Fähigkeiten zu Mitarbeitern.

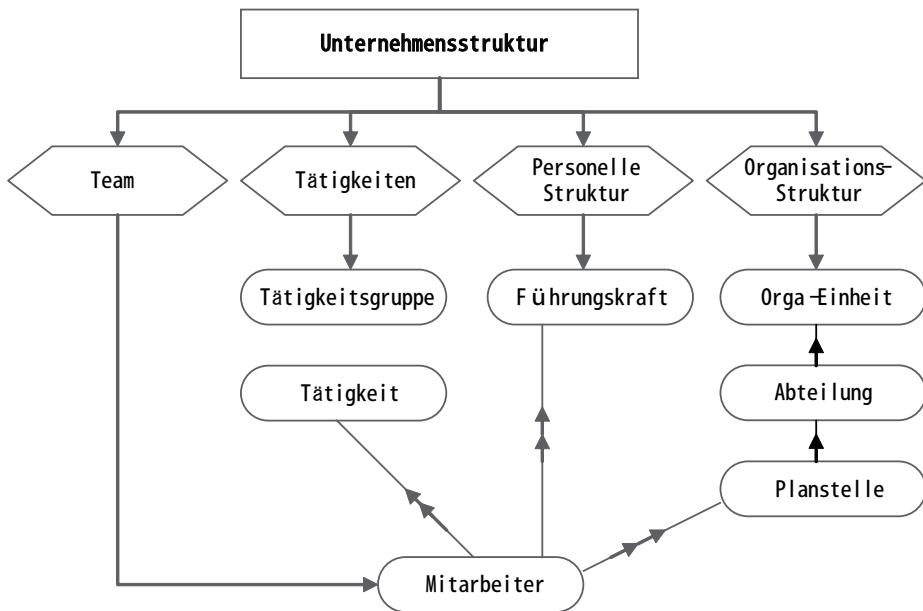


Abbildung 3.8.: mögliche Unternehmensstruktur, angelehnt an [Bie06]

Modul Projektteambesetzung Im Teil 1.1 wurde erwähnt, welche Nachteile Biesalski bei einer unstrukturierten Stellenbesetzung vakanter Projektstellen sieht: ein erhöhtes Risiko des Übersehens von geeigneten Kandidaten und ein zeitliches Risiko. Er schreibt, dass „dieser Zustand weder gerecht, noch transparent ist und in aller Regel [...] zu einer suboptimalen Besetzung von Projektstellen führt“. Deshalb sind die Ziele des Moduls Projektteambesetzung ein automatisiertes Auffinden geeigneter Mitarbeiter und die Zuordnung zu möglichen Projektstellen auf Basis von Kompetenzprofilen der Mitarbeiter.

Der Ansatz von Biesalski ist auf die Durchführung von einzelnen Projekten begrenzt. Die Funktionalität des Moduls steht ausschließlich wie die Abbildung 1.1 aus dem Abschnitt 1.1 zeigt, in der Phase der Projektplanung zur Verfügung. Die Funktionalitäten im Einzelnen sind:

1. Projektverwaltung: Systemanmeldung, Administrationsaufgaben
2. Stellenfunktionen: Stellenverwaltung, Stellenanforderungskompetenzprofil definieren, Projektstelle besetzen
3. Erweiterte Analyse- und Systemfunktionen: Besetzungsvorschlag abrufen, Differenzprofil abrufen, Weiterbildungsinformationen anfordern, aggregierte Mitarbeiterfunktionen abrufen, Stellenprofil exportieren

Damit ein Projektleiter einen Vorschlag für die Besetzung einer Stelle oder eines gesamten Projektteams abrufen kann, muss er zuerst ein Stellenanforderungsprofil definieren. Dazu sucht er im Kompetenzkatalog nach benötigten Kompetenz-Arten und ordnet sie dem Profil zu. Anschließend legt er die Gewichte jeder Kompetenz fest und definiert bestimmte Pflichtkompetenzen als sogenannte „KO-Kriterien“. Diese sind dann zwingend bei der Suche nach Mitarbeiterkompetenzprofilen. Der Projektleiter hat die Möglichkeit, sich eine Liste von geeigneten Mitarbeitern für eine Projektstelle anzeigen zu lassen. Er entscheidet dann nach einer individuellen Prüfung der Eignung der Kandidaten und aufgrund der im Einzelfall zu berücksichtigenden Auswahlkriterien, welcher Mitarbeiter die Stelle besetzt. Weiterhin hat er die Möglichkeit, sich einen Vorschlag für die Besetzung eines gesamten Projektteams generieren zu lassen. Zu diesem Zweck kann eine Einschränkung der Organisationsstruktur erfolgen, um die Selektion nicht aus dem gesamten Unternehmen durchzuführen. Daraufhin werden Vergleichskennzahlen berechnet. Diese beruhen auf „KO-Kriterien“, semantisch ähnlichen Profilen oder Über- und Teilqualifikationen hinsichtlich des Kompetenzprofils. Die Ergebnisse werden dann graphisch übersichtlich präsentiert.

Der ontologiebasierte Vergleich von Stellenanforderungsprofilen mit Mitarbeiterkompetenzprofilen stellt für Biesalski [Bie06] eine der wichtigsten Funktionalitäten dar. Die Vergleichskomponente wird im Gesamtsystem zentral implementiert und operiert auf der modellierten Ontologie. Er nutzt ein aggregiertes Gesamtmaß als Ergebnis des Vergleichs, welches aus vier Teilen zusammengesetzt ist. Dabei ist R ein Anforderungsprofil mit einer Liste von Kompetenzen r mit je einer Bezeichnung rs und einer Gewichtung rl . E stellt das Mitarbeiterprofil dar mit einer Liste von Kompetenzen e , die je eine Bezeichnung es und eine Gewichtung el haben.

- **Direkter Vergleich:** Einzelne Kompetenzen können als „KO-Kriterien“ definiert werden. Die Berechnung dieses Vergleichswertes erfolgt sowohl auf Ebene der Kompetenzbezeichnung als auch der Gewichtung. Die Ähnlichkeit („sim“) zwischen einem Anforderungsprofil R und einem Mitarbeiterprofil E in einem direkten Vergleich ($sim_{dVergleich}(R, E)$) ist das Verhältnis von allen Vergleichswerten der Fähigkeitsbezeichnungen und ihrer Gewichte zu den im Projektprofil geforderten Fähigkeitsgewichten. Formel 3.1 zeigt dieses Verhältnis, die Formeln 3.2 und 3.3 sind benutzte Hilfsformeln und geben jeweils 0 oder 1 zurück, wenn die Bezeichnung der verglichenen Fähigkeiten

und ihre Gewichte übereinstimmen.

$$sim_{dVergleich}(R, E) := \frac{\sum_{r \in R} \sum_{e \in E} weight(r) * sim_{Gewicht}(r, e) * sim_{Bezeichnung}(r, e)}{\sum_{r \in R} weight(r)} \quad (3.1)$$

$$sim_{Bezeichnung}(r, e) := \begin{cases} 1, & \text{wenn } rs = es \\ 0, & \text{sonst} \end{cases} \quad (3.2)$$

$$sim_{Gewicht}(r, e) := \begin{cases} 1, & \text{wenn } rl = el \\ 0, & \text{sonst} \end{cases} \quad (3.3)$$

Die beiden Hilfsformeln ermitteln im direkten Vergleich nur die Gleichheit zweier Profile. Wenn beide Profile gleich sind, liefert der direkte Vergleich sim im Ergebnis 1, sonst einen Wert zwischen 0 und 1.

- **Proportionale Ähnlichkeit:** Teilweise erfüllte Anforderungen, also Unterqualifikationen bzw. nicht erfüllte Mindestanforderungen des Projekts, sollen als solche erkannt werden und entsprechend als Abwertung in die Berechnung einfließen. Im direkten Vergleich wurde auf Gleichheit überprüft, was für „KO-Kriterien“ eines Projektes benutzt werden kann.

In Formel 3.4 werden nun Mitarbeiterprofile betrachtet, in denen Fähigkeiten vorkommen, die eine geringere Gewichtung haben als im Projektanforderungsprofil gefordert. Die Hilfsformel 3.5 liefert eine 1, wenn die Gewichte von Fähigkeiten im Projektprofil kleiner sind, als die Fähigkeitsgewichte von Mitarbeitern. Wenn die Anforderungen des Projekts größer sind als im Profil eines Mitarbeiters angegeben, errechnet die Formel je nach Abstand der Gewichte aus Projekt- und Mitarbeiterprofil einen Wert zwischen -0,5 und 0. Dieser fließt in die Summenberechnung der Formel 3.4 ein und wertet die proportionale Ähnlichkeit zwischen Projektprofil und Mitarbeiterprofil entsprechend ab.

$$sim_{pVergleich}(R, E) := \frac{\sum_{r \in R} \sum_{e \in E} weight(r) * sim_{pGewicht}(r, e) * sim_{Bezeichnung}(r, e)}{\sum_{r \in R} weight(r)} \quad (3.4)$$

$$sim_{pGewicht}(r, e) := \begin{cases} 1, & \text{wenn } rl \leq el \\ \left(1 - (rl - el) * \frac{1}{4}\right), & \text{sonst} \end{cases} \quad (3.5)$$

Diese beiden Formeln überprüfen wiederum auf Gleichheit. Für den Vergleich der Bezeichnung wird die Gleichung 3.2 des direkten Vergleichs benutzt.

- **Kompensatorische Ähnlichkeit:** Eine Erweiterung der Unterqualifikation ist die Überqualifikation, die Defizite bei anderen Kompetenzen abmildern kann und bei vielen sehr guten Kandidaten als zusätzliches Kriterium der Differenzierung dient. Die Berechnung des Vergleichswerts funktioniert ähnlich wie bei der proportionalen Ähnlichkeit. Aber der Vergleich der Fähigkeitsgewichte in der Gleichung 3.7 liefert grundsätzlich einen negativen Wert, wenn die Projektanforderungen größer sind als die Fähigkeiten des Mitarbeiters und positiv im umgedrehten Fall. Eine Überqualifikation eines Mitarbeiters wird also in der Formel 3.6 durch einen positiven Wert berücksichtigt, eine Unterqualifikation fließt mit einem negativen Wert in die Summe ein.

$$sim_{kVergleich}(R, E) := \frac{\sum_{r \in R} \sum_{e \in E} weight(r) * sim_{kGewicht}(r, e) * sim_{Bezeichnung}(r, e)}{\sum_{r \in R} weight(r)} \quad (3.6)$$

$$sim_{kGewicht}(r, e) := 1 - (rl - el) * \frac{1}{4} \quad (3.7)$$

Durch diese Vorgehensweise und Summierung aller Werte können Unterqualifikationen durch Überqualifikationen ausgeglichen werden.

- **Taxonomische Ähnlichkeit:** Die taxonomische Struktur der Kompetenzontologie wird hierbei genutzt, um einen unscharfen Vergleich zwischen ähnli-

chen Kompetenzen durchzuführen. Dieser Vergleich basiert auf einem Ähnlichkeitsmaß. Fähigkeiten werden nicht mehr nur über Gleichheit verglichen, sondern über ihren „räumlichen“ Abstand in der baumartigen Taxonomie. Die Ähnlichkeit berechnet sich allgemein durch folgende Gleichung:

$$sim_{tAehnlichkeit}(R, E) := 1 - distance(r, e) \quad (3.8)$$

In der Gleichung ist *distance* eine Funktion zur Berechnung des Abstandes zwischen einer Anforderung *r* und einer Fähigkeit *e* des Mitarbeiters. Dieser Wert liegt zwischen 0 (gleiches Konzept) und 1 (total verschieden). Der Wert der Ähnlichkeit liegt damit ebenfalls zwischen 0 (gar keine Ähnlichkeit) und 1 (übereinstimmend). Auf diese Art können Fähigkeiten einer Taxonomie auch dann verglichen werden, wenn sie nicht dieselbe Bezeichnung haben, also unterschiedlich sind.

Ergebnisse In der Evaluation des Systems zeigte sich nach eigenen Angaben eine positive Resonanz der benutzenden Mitarbeiter der DaimlerCrysler AG. Sie bauten viele Vorurteile gegenüber existierenden Kompetenzmanagement-Systemen ab. Die Einführung eines flexibleren Kompetenzkatalogs in Form einer Kompetenz-Ontologie wurde positiv aufgenommen. Sie führt zu einer deutlich verbesserten und effizienteren Einschätzung der Kompetenzen. Die differenzierten Kennzahlen beim Vergleich von Stellenprofilen mit Mitarbeiterkompetenzprofilen können ebenso genutzt werden, um hier eine fundiertere Grundlage bei der Besetzung von Stellen oder den Vorschlägen für Bildungsmaßnahmen einzubeziehen. Mit ihrem Testszenario zeigen die Autoren, wie die einzelnen Module einen echten Mehrwert für die Mitarbeiter bieten, z.B. die automatisch generierten Vorschläge für Bildungsmaßnahmen oder Teamzusammensetzungen. Ebenso wie bei der vorliegenden Arbeit, so liegt auch der Fokus des ePeople-Systems bezüglich Projektteambildung auf der Planungsphase einzelner Projekte und damit auf deren kompetenzorientierten Zusammensetzung. Biesalski arbeitet ausführlich die Vorteile von Ontologien gegenüber sonstigen Arten der Kompetenzerfassung heraus. Ebenso ausführlich beschreibt er, wie Kompetenzprofile bewertet werden. Dabei sei erwähnt, dass die drei ersten Vergleichsmaße einen reinen Vergleich auf Gleichheit durchführen und noch keine Ähnlichkeitsberechnung nutzen. Das vierte definierte Maß schließlich greift auf die taxonomische Struktur der Kompetenzstrukturierung

zurück und berechnet die Ähnlichkeit zweier Kompetenzen basierend auf ihrem Abstand in der Hierarchie. Wenn diese Unterscheidung auch nicht ganz den Kern vorliegenden Arbeit treffen mag, so profitiert sie dennoch von den Ergebnissen der einzelnen Betrachtungen.

Zusammengefasst bildet die Arbeit Biesalskis eine sehr wertvolle Grundlage für die nun folgende Anforderungsdefinition.

3.2. Anforderungen

Aus den Erfahrungen der verwandten Arbeiten ergeben sich schließlich die Anforderungen der zu entwickelnden Anwendung. Hinzu kommen die Interviews mit Siemens Teamrecruitern, deren Ergebnisse in einer Anforderungsanalyse niedergeschrieben wurden. Vergleichend stellen Deiters und Lucas [DL00] folgende technische Anforderungen an Skillmanagement-Systeme:

- **Variabilität der Kenntnisstrukturierung:** Die Kenntnisse eines Mitarbeiters setzten sich aus Aspekten mehrerer Kategorien zusammen. Die unterschiedlichen Aspekte, die bei der Bewertung der Kenntnisse eine Rolle spielen, werden zu einem Kriterienkatalog verbunden. Es muss in einem Skillmanagement-System die Möglichkeit bestehen, mehrere parallele Kriterienkataloge zu definieren, z.B. Produktkatalog, Projektkatalog, und die Möglichkeit diese Kataloge in Abhängigkeit der durchzuführenden Aufgabe in Beziehung zueinander zu bringen.
- **Objektive Bewertungen:** Für den Vergleich von Mitarbeiterkenntnissen ist es notwendig, die Kenntnisse in einem Bewertungsschema einzustufen. Eine Bewertung von Kenntnissen sollte jedoch von den Bewerteten selbst oder in Zusammenarbeit mit ihnen erfolgen, um die Akzeptanz der Beurteilungen zu sichern. Ein Vergleich der Kenntnisse ist nur möglich, wenn ein einheitlicher Bewertungsmaßstab zu Grunde liegt.
- **Flexible Datenaktualisierungen:** Ein Skillmanagement-System muss Techniken bieten, teilweise bearbeitete Profile zu speichern, verschiedene Versionen zu verwalten und auf Vollständigkeit zu prüfen. Die Daten der Mitarbeiterkenntnisse besitzen eine starke Dynamik, weshalb das System in der Lage sein muss, diese Daten regelmäßig zu aktualisieren.

- **Such- und Auswertungsfunktionalität:** Skillmanagement-Systeme müssen umfangreiche Such- und Auswertungsfunktionalitäten bieten, damit der Benutzer das eingetragene Wissen auch wieder zurück erhält. Die Komplexität der Umsetzung hängt im Wesentlichen von der Form der Profilstrukturierung ab.
- **Datenschutz:** Die Erfassung von personenbezogenen Daten, wie sie im Rahmen des Skillmanagements notwendig ist, erfordert neben organisatorischen Regelungen auch technische Maßnahmen wie Anonymisierung, Datenreduktion oder automatische Benachrichtigungsmechanismen.

Ist-Zustand bei der Siemens AG In einem Guide der Siemens AG für IT-basierte Großprojekte [Kuh99] heißt es: „Die optimale Abwicklung eines Projektes ist nur in einem Umfeld möglich, das nicht nur über die erforderliche Menge von Ressourcen (Personal, Infrastruktur, Prozesse) verfügt, sondern auch über eine adäquate Mischung der Mitarbeiterskills.“ Damit wird erneut die Praxisrelevanz dieser Ausarbeitung erkennbar. Weiterhin wird geschrieben, dass der Teamrecruiter „sich schon während der Angebotserstellung ein Bild über das für sein Projekt erforderliche Mitarbeiterprofil“ machen sollte. Durch eine frühzeitige Besetzung von Schlüsselpositionen im Projekt könne der Projekterfolg abgesichert werden, was ebenso in Teil 1.1 beschrieben wurde.

Als Voraussetzung für diesen Vorgang werden folgende Prozesse genannt: Mitarbeiter vorausschauend frei machen, Mitarbeiter schulen oder umschulen, sie bei anderen Abteilungen anfordern oder neue Mitarbeiter einstellen. Diese Prozesse werden teilweise in den Anwendungsfällen der vorliegenden Arbeit berücksichtigt (siehe 3.3) und umgesetzt.

Die in dem Guide empfohlene Vorgehensweise sieht die Mitarbeiterprofil-Planung in Form eines Stufendiagramms vor, in dem, aufgeschlüsselt nach Fähigkeitenniveaus und Zeitraum, die Anzahl und der Typ von Mitarbeitern sichtbar wird, die für das Projekt erforderlich sind. Auf der Zeitachse des Diagramms sollen nicht nur die Projektlaufzeit und die Dauer der Phasen dargestellt werden, sondern auch markante Meilensteine. Auf der Mitarbeiterachse soll die Anzahl der Mitarbeiter je Fähigkeitenniveau aufgetragen werden, die für die Abwicklung der jeweiligen Projektphase erforderlich ist. Von einer detailliereren Mitarbeitereinsatzplanung sollte abgesehen werden, da Grobplanung und Projektcharakterisierung die Ziele die-

ses Vorgangs bestimmen.

Ähnlich wie in dem genannten Diagramm soll die Anwendung dieser Ausarbeitung im ersten Schritt die Auswahl der relevanten Mitarbeiter dem Projektzeitraum anpassen. Mitarbeiter, die bereits für den Projektzeitraum eine Beschäftigung haben, sollen nicht betrachtet werden. Ansonsten beruht die Mitgliederauswahl auf fachlichen Eigenschaften, den bei Siemens bezeichneten Fähigkeiteniveaus. Somit wird der größte Teil der empfohlenen Vorgehensweise auch für den Rahmen dieser Arbeit berücksichtigt.

Anforderungsanalyse In der vorliegenden Arbeit soll eine prototypische Applikation entwickelt werden, die Teamrecruiter eines Unternehmens dabei unterstützen soll, geeignete, möglichst optimale Teams mit den vorhandenen Mitarbeitern basierend auf einer gegebenen Projektbeschreibung zusammenzusetzen. Die Teams sollen mit einem Geeignetheitsgrad bewertet werden. Kann kein Team mit den vorhandenen Mitarbeitern gebildet werden, soll die Anwendung dem Teamrecruiter diverse Alternativen vorschlagen.

Es sei darauf hingewiesen, dass die Anwendung nicht alle benannten Funktionen eines Kompetenzmanagement-Systems beinhalten wird. Vielmehr wird ein Prototyp einer IT-gestützten Teamplanung entwickelt, der als Komponente in ein Kompetenzmanagement-System integriert werden kann. Wie Zelewski in [Zel04] schreibt, wird „die Zusammenstellung von Projektteams in aktuellen Kompetenzmanagement-Systemen nur mangelhaft mit praxistauglichen Instrumenten unterstützt.“ Die zu entwickelnde Applikation soll zur Verbesserung genau dieser Funktionalität beitragen. Aus diesem Grund werden auch zwei der oben genannten Anforderungen aus [DL00] für die Anforderungsanalyse nicht berücksichtigt, wozu einerseits die variable Kenntnisstrukturierung gehört. Wie bereits erwähnt, wird das System eine HR-Ontologie benutzen, die umfangreich und brachenübergreifend modelliert wurde. Deshalb wird von der Erfüllung dieser Systemanforderung abgesehen. Andererseits existieren zwar diverse Suchmöglichkeiten in den Anwendungsfällen des Prototyps, diese dienen aber ausschließlich dem Zweck der Teamplanung.

Der Teamrecruiter übergibt der Anwendung als Eingabe eine Projektbeschreibung. Außerdem kann er, wenn notwendig, nicht verfügbare Mitarbeiter aus dem Prozess der Teamzusammensetzung ausschließen. Er erhält als Ergebnis mehrere mögliche

Teamzusammensetzungen, die anhand von Kriterien eines optimalen Teams und der Projektbeschreibung bewertet werden. Die funktionalen Anforderungen einer Applikation zur Erfüllung dieser Zielstellung in einem realen Unternehmensumfeld sehen wie folgt aus:

1. Speicherung der notwendigen Mitarbeiterdaten; das sind neben den persönlichen Informationen auch Kenntnisse und bereits erledigte Aufgaben
2. Mitarbeiter pflegen ihre Daten selber, das heißt, sie tragen sie ein und bearbeiten diese dann später
3. Teamrecruiter können nach Projektabschluss etwaige Änderungen an diesen Beschreibungen vornehmen
4. Teamrecruiter erhalten auf Eingabe der Projektbeschreibung eine Ergebnisliste mit bewerteten Teamzusammensetzungen
5. mögliche Speicherung der Teamzusammensetzungen zu den jeweiligen Projektbeschreibungen
6. aus datenschutzrechtlichen Gründen darf die Anwendung ausschließlich im internen Firmennetzwerk genutzt werden
7. Ausgaben erfolgen in einem maschinenlesbaren Format

Die prototypische Anwendung wird nicht alle genannten Anforderungen umsetzen. Ihr Fokus liegt vielmehr auf der Analyse des Konzepts zur Teamplanung mit semantischen Webtechnologien und der Implementierung der entsprechenden Algorithmen (siehe dazu Abschnitt 5).

Die nichtfunktionalen Anforderungen gestalten sich dahingehend umfangreicher. Die Benutzer der Anwendung werden die Mitarbeiter des Unternehmens sein. Eine einfache Bedienung und eine leicht verständliche Benutzeroberfläche bieten sich daher an. Anwendungen, die semantische Webtechnologien nutzen, führen meist sehr komplexe und aufwändige Berechnungen durch. Trotzdem sind Performanz und schnelle Suchläufe für eine betriebliche Anwendung wichtig.

Mögliche zu behandelnde Fehler werden in den Anwendungsfällen im Abschnitt 3.3 beschrieben. Das System sollte mit Fehlern umgehen können und stets in einem aktiven Zustand bleiben. Im Prototypen der Anwendung stammen Eingaben nicht aus anderen Systemen. Projektbeschreibungen werden mittels eines maschinenlesbaren Dokumentes oder per Formular übergeben. Ebenso werden Ausgaben nicht

automatisch an andere Systeme weitergeleitet. Der Prototyp des Systems braucht nicht portabel zu sein. Er sollte aber die Möglichkeit bieten, flexible Importschnittstellen für vorhandene Systeme, z.B. für den Import von Personalprofilen aus SAP zu implementieren. Die prototypische Anwendung wird direkt auf die in Teil 4.1 beschriebene HR-Ontologie zugreifen.

3.3. Anwendungsfälle

Anwendungsfälle oder Use Cases beschreiben die Abläufe bzw. Prozesse einer Anwendung, indem Interaktionen zwischen Akteuren und dem zu betrachteten System erläutert werden. Folgende für die Teamplanungsfunktionalitäten relevante Anwendungsfälle sollen in der im Rahmen dieser Arbeit entstandenen Anwendung realisiert werden:

- Generierung einer Teamzusammensetzung anhand eines Projektanforderungsprofils
- Generierung der Kriterien einer Ausschreibung mittels eines Differenzkompetenzprofils
- Vergleich von Mitarbeiterkompetenzprofilen mit einem Differenzkompetenzprofil bzgl. Weiterbildungsmaßnahmen
- Vergleich eines Bewerberkompetenzprofils mit einem Teamkompetenzprofil
- Vergleich von Mitarbeiterkompetenzprofilen bezüglich Ähnlichkeit

Es werden verschiedene Profile verwendet, die allgemein gesehen Mengen von gewichteten oder ungewichteten Kompetenzen darstellen. Ein Projektanforderungsprofil enthält, wie das Stellenanforderungsprofil aus Abschnitt 3.1.5, diejenigen Kompetenzen, die zur Erfüllung der im Projekt vorgesehenen Aufgaben benötigt werden. Das Differenzkompetenzprofil besteht aus, wie in der Definition von Biesalski, der Differenz der Kompetenzen zweier Profile. In diesem Fall soll ein Vergleich zwischen einem Projektanforderungsprofil und einem Teamkompetenzprofil durchgeführt werden. Die zur Erfüllung der Projektanforderungen fehlenden Kompetenzen werden im Differenzprofil dargestellt. Mitarbeiterkompetenzprofile sind solche, die die Fähigkeiten jedes einzelnen Mitarbeiters bündeln. Ein Bewerberkompetenzprofil beschreibt die Kompetenzen einer noch nicht im Unternehmen

befindlichen Person. Dabei soll überprüft werden, ob sich ein bestimmter Bewerber eignen würde, in einem Team mitzuarbeiten.

Im Folgenden werden die genannten Anwendungsfälle detaillierter beschrieben.

Teamzusammensetzung Dieser Anwendungsfall bildet den Kern dieser Arbeit. Einem Teamrecruiter soll eine Entscheidungsvorlage für die Auswahl von Mitarbeitern für ein Team unter Angabe Projektanforderungsprofils geliefert werden. Er erhält als Ergebnis eine Liste von möglichen Teams, wobei jedes davon mit einem Geeignetheitsgrad bezüglich der Projektanforderungen bewertet wird. Diese Bewertung richtet sich danach, inwieweit die einzelnen Teams die Kriterien optimaler Teams (siehe Abschnitt 4.2) erfüllen. Ziel eines den Prozess des Teamrecruitments unterstützenden Teamplanungsalgorithmus sollte es sein, möglichst optimale Teams zusammenzusetzen. Je nach vorhandenen Mitarbeiterkompetenzprofilen kann dies für ein Projekt möglich sein oder nicht. Dabei sei erwähnt, dass es dem Teamrecruiter überlassen bleibt, welche der vorgeschlagenen Zusammensetzungen er als so geeignet einschätzt, dass er sie in die Tat umsetzt. Für den Fall, dass kein optimales Team gebildet werden kann bzw. das bestmögliche Team für den Teamrecruiter nicht ausreichend ist, können die weiteren Anwendungsfälle dem Anwender alternative Lösungen ermöglichen.

- Akteure: Teamrecruiter
- Auslöser: Der Teamrecruiter möchte für ein anstehendes Projekt geeignete Mitarbeiter aussuchen.
- Vorbedingungen: Für jede Person der Mitarbeitermenge, aus denen Teammitglieder selektiert werden sollen, existiert ein Mitarbeiterkompetenzprofil, auf das vom System zugegriffen werden kann.
- Eingabe: Die vorgenannten Mitarbeiterprofile und ein Projektanforderungsprofil werden zur Generierung einer entsprechenden Teamzusammensetzung benötigt.
- Ablauf:
 1. a) Der Teamrecruiter erstellt ein Projektanforderungsprofil, in dem er ein neues Projekt mit seinen Daten (z.B. Titel, Datum des Beginns, Enddatum) erzeugt und ihm relevante Aufgaben sowie notwendige Fähig-

keiten zuordnet. Diese Fähigkeiten können mit einem der Gewichte 1-Student, 2-Junior, 3-Middle und 4-Senior versehen werden.

- b) Als andere Möglichkeit kann er eine bereits vorbereitete Projektbeschreibung in das System eingeben.
2. Er markiert bestimmte Mitarbeiter als nicht verfügbar und schließt sie damit aus der Berechnung aus. Grundlage für diese Entscheidung sollten einzig terminliche Verhinderungen wie z.B. Urlaub oder Krankheit sein. Der Ausschluss sollte nicht auf vermeintlich ungeeigneten fachlichen Eigenschaften beruhen, denn dies wird von der Anwendung automatisiert geleistet.
 3. Er startet die Generierung der Teamzusammensetzungen.
 4. Der Teamplanungsalgorithmus beginnt zu arbeiten. Er berechnet diverse Teamzusammensetzungen und evaluiert sie, was mittels eines Geeignetheitsgrades ausgedrückt wird. Er präsentiert dem Teamrecruiter die Liste der einzelnen Teams absteigend sortiert nach ihrer Bewertung.
- Ausgabe: Liste von Teams. Jedes Team besteht aus Mitarbeitern des Unternehmens und wurde mit einer Bewertung versehen.
 - Nachbedingungen: Falls der Teamrecruiter aufgrund zu geringer Teambewertungen keines der präsentierten Teams zur Projektdurchführung auswählen möchte oder mit den vorhandenen Mitarbeitern kein optimales Team gebildet werden kann, kann er einen der alternativen Anwendungsfälle nutzen.

Ausschreibungskriterien Eine weitere Einsatzmöglichkeit der Anwendung ist die Generierung von Kriterien einer Ausschreibung. Ziel ist es, mit dieser Ausschreibung die Kompetenzen zu ergänzen, die zur Bildung eines optimalen Teams fehlen. Der Teamrecruiter kann diesen Prozess nach eigener Einschätzung auslösen, wenn ihm keines der durch die Teamzusammensetzung vorgeschlagenen Teams zur Erfüllung der Projektaufgaben als geeignet erscheint. Das System erstellt daraufhin ein Differenzkompetenzprofil, das die fehlenden Kompetenzen enthält. Dieses Profil bildet dann die Grundlage einer Ausschreibung.

- Akteure: Teamrecruiter

- Auslöser: Es konnte kein optimales Team gebildet werden, das bedeutet, dass keines der generierten Teams eine hinreichend große Bewertung bezüglich des Projektanforderungsprofils erhalten hat.
- Vorbedingungen: Der Teamplanungsalgorithmus hat eine Liste von bewerteten Teamzusammensetzungen generiert (siehe oben). Die Bewertungen der einzelnen Teams reichen dem Teamrecruiter nicht aus.
- Eingabe: Teamkompetenzprofil des bestbewerteten Teams und Projektanforderungsprofil.
- Ablauf:
 1. Der Teamrecruiter wertet die Ergebnisse der Teamzusammensetzungen aus und entscheidet sich für die Generierung von Ausschreibungskriterien. Er startet den Vorgang.
 2. Der Teamplanungsalgorithmus vergleicht das während der Generierung der Zusammensetzung entstandene beste Team mit den Projektanforderungen. Er ermittelt die fehlenden Kompetenzen.
 3. Das generierte Differenzkompetenzprofil wird dem Teamrecruiter präsentiert.
- Ausgabe: Differenzkompetenzprofil, das diejenigen Kompetenzen enthält, die zur Bildung eines optimales Team bezüglich der Projektanforderungen benötigt werden.
- Nachbedingungen: Nachdem eine Ausschreibung stattgefunden hat und Bewerber zur Auswahl stehen, kann deren Eignung mittels des Anwendungsfalls „Bewerbertauglichkeit“ geprüft werden.

Bewerbertauglichkeit Wie bereits beschrieben, besteht die Möglichkeit, zu überprüfen, ob die Kompetenzen eines Bewerbers zur Vervollständigung eines Teams geeignet sind. Ein positiver Vergleich kann dann zur Einstellung des Bewerbers führen und den Projektbeginn einleiten. Dabei muss dieser Prozess nicht notwendigerweise nach dem Prozess der Generierung der Ausschreibungskriterien erfolgen. Der Vergleich eines Bewerberprofils mit einem Teamprofil kann jederzeit erfolgen.

- Akteure: Teamrecruiter
- Auslöser: Es konnte kein optimales Team gebildet werden. Ein Bewerber soll dieses Team ergänzen kann.
- Vorbedingungen: Es existiert ein unvollständiges Teamkompetenzprofil.
- Eingabe: Für die Berechnung werden das Bewerberkompetenzprofil, ein unvollständiges Teamkompetenzprofil sowie ein Projektanforderungsprofil benötigt.
- Ablauf:
 1. a) Der Teamrecruiter entscheidet sich nach Generierung von Teams zur Erstellung von Ausschreibungskriterien, um eine geeignete Person zur Komplettierung des Teams zu finden. Daraufhin wurde ein Bewerber gefunden, der auf Eignung überprüft werden soll.
b) Der Teamrecruiter findet unabhängig von einer Teamzusammensetzung einen Bewerber, von dem er glaubt, dass er geeignet wäre für ein zukünftiges Projekt.
 2. Er erstellt das Kompetenzprofil des Bewerbers.
 3. a) Das gespeicherte beste Team der zuvor durchgeföhrten Teamzusammensetzungen wird zur Vervollständigung aufgerufen.
b) Er startet den Vorgang der Teamzusammensetzungen inklusive des Bewerberprofils.
 4. Der Teamrecruiter startet den Vergleich des Bewerbers mit dem gewählten Team.
 5. Der Grad der Geeignetheit des Bewerbers wird dem Teamrecruiter präsentiert.
- Ausgabe: Grad der Geeignetheit des Bewerbers für ein Team, das bezüglich bestimmter Projektanforderungen zusammengesetzt wurde.
- Nachbedingungen: Sollte im Ergebnis ein hoher Geeignetheitsgrad stehen, kann der Teamrecruiter die Einstellung des Bewerbers veranlassen und diesen im anstehenden Projekt einsetzen. Das Projekt wird dann mit der neuen Teamzusammensetzung gespeichert und durchgeführt. Sollte der Vergleich

hingegen zu Ungunsten des Bewerbers ausfallen, kann der Teamrecruiter entweder nach einem weiteren Bewerber suchen oder eine der anderen Alternativen in Betracht ziehen.

Weiterbildungsmaßnahmen Der Vorschlag von effizienten Weiterbildungsmaßnahmen bietet dem Teamrecruiter einen weiteren Ausweg, wenn aus den vorhandenen Mitarbeitern keine hinreichend geeigneten Team gebildet werden konnten. Wenn der Teamrecruiter diesen Anwendungsfall gestartet hat, vergleicht die Anwendung die bestbewertete Teamzusammensetzung mit den gestellten Projektanforderungen. Dieses Differenzkompetenzprofil wird dann daraufhin untersucht, welche der Teammitglieder mit geringstem Aufwand derart geschult werden können, dass sie im Anschluss an die Weiterbildung ein optimales Team bilden.

- Akteure: Teamrecruiter
- Auslöser: Es konnte kein optimales Team gebildet werden. Der Teamrecruiter startet dann die Generierung eines Vorschlags für Weiterbildungsmaßnahmen.
- Vorbedingungen: Der Teamplanungsalgorithmus generiert eine Liste von bewerteten Teamzusammensetzungen, wobei die Bewertungen der einzelnen Teams den Teamrecruiter nicht zufriedenstellen.
- Eingabe: Es werden das Teamkompetenzprofil eines ausgewählten (z.B. des bestbewerteten) Teams, die Mitarbeiterprofile dieses Teams sowie das Projektanforderungsprofil benötigt.
- Ablauf:
 1. Der Teamrecruiter wertet die Ergebnisse der Teamzusammensetzungen aus und entscheidet sich für die Generierung von Vorschlägen zu Weiterbildungsmaßnahmen. Er startet den Vorgang.
 2. Der Teamplanungsalgorithmus vergleicht das während der Generierung der Zusammensetzung entstandene Team mit den Projektanforderungen. Er ermittelt die fehlenden Kompetenzen und die Mitarbeiter, die mit geringstem Aufwand jene Kompetenzen in Schulungsmaßnahmen lernen können, um die Wissenslücken zu schließen.

3. Die generierten Mitarbeiter mit ihren zu schulenden Kompetenzen werden dem Teamrecruiter präsentiert.

- Ausgabe: Liste von Mitarbeitern und Kompetenzen, die geschult werden könnten.
- Nachbedingungen: Nachdem die vorgeschlagenen Mitarbeiter an den Weiterbildungsmaßnahmen teilgenommen haben, kann der Teamrecruiter erneut eine Liste der Teamzusammensetzungen erzeugen. Zuvor müssen die Kompetenzprofile der geschulten Mitarbeiter entsprechend geändert werden.

Ähnlichkeit zwischen Mitarbeitern Dieser Anwendungsfall kann genutzt werden, wenn nach Projektbeginn eines der Mitglieder nicht mehr zur Verfügung steht oder aus diversen Gründen ausfällt bzw. wenn schon absehbar ist, dass eine der Personen in nächster Zeit das Projekt verlassen muss. Dann ist es wie unter 4.2 beschrieben essentiell, schnell guten Ersatz zu finden, damit das Projekt dennoch zum erfolgreichen Abschluss kommen kann. Der Teamrecruiter oder in diesem Fall auch der Teamleiter hat mit dem Prozess „ähnlichen Mitarbeiter finden“ die Möglichkeit, einen ähnlichen Mitarbeiter für den ausgefallenen zu bestimmen. Der Teamplanungsalgorithmus schließt automatisch für die Berechnung nicht verfügbare Mitarbeiter aus und gibt im Ergebnis die Person aus, die der ausgefallenen am ähnlichsten ist. So ist es bei einem hohen Übereinstimmungsgrad wahrscheinlich, dass die neue Person die Aufgaben des ausgefallenen Teammitglieds übernehmen kann.

- Akteure: Teamrecruiter oder Teamleiter
- Auslöser: Ausfall eines der Teammitglieder.
- Vorbedingungen: Es existiert ein Team. Für ein möglicherweise ausfallendes Mitglied wird Ersatz gesucht.
- Eingabe: Es werden das Mitarbeiterkompetenzprofil des ausfallenden Mitarbeiters sowie das Projektanforderungsprofil benötigt.
- Ablauf:
 1. Der Teamrecruiter oder Teamleiter weiß, dass eines der Teammitglieder demnächst das Projekt verlassen wird bzw. dass das bereits geschehen ist.

2. Er startet die Suche nach einem ähnlichen Mitarbeiter.
 3. Der Teamplanungsalgorithmus schließt nicht verfügbare Mitarbeiter (z.B. aus anderen Projekten) aus der Ähnlichkeitsberechnung aus und sucht in der Menge der verbleibenden nach dem Mitarbeiter, der dem ausfallenden Mitglied am ähnlichsten ist.
 4. Eine Liste von Mitarbeitern mit Angabe ihrer Ähnlichkeit wird präsentiert.
- Ausgabe: Eine Liste von Mitarbeitern und der Grad ihrer Ähnlichkeit zum ausfallenden Mitarbeiter.
 - Nachbedingungen: Der Akteur entscheidet sich für einen der Mitarbeiter auf der Liste und beschäftigt diesen dann ersatzweise im Projekt.

4. Konzeption und Entwurf

Nachdem in den vorangegangen Kapiteln die Thematik, das Umfeld der Arbeit und Begriffe erläutert und die Anforderungen an die Anwendung definiert wurden, folgt nun die Beschreibung der Konzeption und des Designs dieser Lösung.

4.1. Human Resource Ontologie

Der Vorgänger der HR-Ontologie wurde bereits in Abschnitt 2.3.2 erwähnt und soll an dieser Stelle genauer beschrieben werden.

Die verwendete Ontologie ist die Weiterentwicklung der Wissensnetze Ontologie und basierte in dieser Version auf bereits existierenden Spezifikationen und Ontologien im Human Resource (HR) Sektor zur Beschreibung von Berufen und zur Klassifikation der Wirtschaftszweige. Das sind im Detail die in [TM06] genannten:

- **Berufskennziffer (BKZ):** Die BKZ werden von der Bundesagentur für Arbeit zur Klassifikation von Berufstätigkeiten verwendet und enthalten insgesamt 5597 Berufskategorien.
- **Standard Occupational Classification (SOC) System:** ist das amerikanische Gegenstück zur BKZ und ordnet die Berufstätigkeiten in 23 Hauptgruppen, 96 Nebengruppen und 449 Berufe ein.
- **Wirtschaftszweige 2003 (WZ2003):** Die Klassifikation der Wirtschaftszweige ist eine deutsche Standardklassifikation der Wirtschaftsaktivitäten, die Firmen nach von ihnen durchgeführten Tätigkeiten in entsprechende Gruppen einteilt. Die Ontologie umfasst insgesamt 1442 Konzepte.

- **North American Industry Classification System (NAICS):** Dieses System definiert vergleichbar zum deutschen WZ2003 die Wirtschaftszweige für Kanada, Mexiko und USA.
- **Human-Resource-XML (HR-XML):** Der meist genutzte Standard für den Austausch von Prozessdokumenten wie Stellenangeboten und Bewerbungen ist HR-XML, dessen Standardisierung vom HR-XML-Consortium, einer unabhängigen, nichtkommerziellen Organisation zur Entwicklung von Standards, geleitet wird. Ziel ist die Ermöglichung des automatischen Datenaustausches im Bereich Human Resource auch unternehmensübergreifend. Die Bundesagentur für Arbeit entwickelte aus dem internationalen HR-XML die deutsche Erweiterung HR-BA-XML.
- **KOWIEN Qualifikations-Ontologie:** Aus dieser Ontologie, in der Kompetenzen beschrieben werden, wurden Teile in die HR-Ontologie importiert.

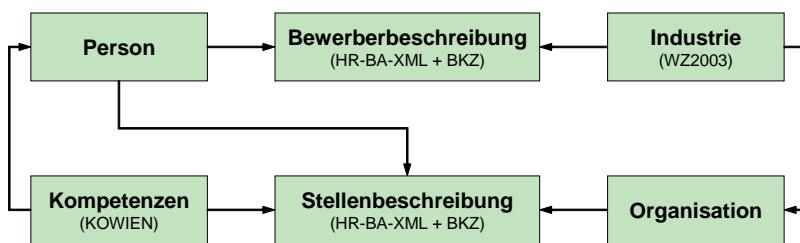


Abbildung 4.1.: Nutzung von kontrollierten Vokabularen

In der internationalen Version der Ontologie werden NAICS und SOC anstelle von WZ2003 und BKZ eingesetzt.

Insgesamt umfasst die HR-Ontologie 9666 Konzepte in der Version, die der vorliegenden Arbeit als Grundlage dient. Für die Umsetzung der in 3.3 beschriebenen Anwendungsfälle sind davon insbesondere die Modellierung von Fähigkeiten und Aufgaben relevant. Die HR-Ontologie enthält insgesamt weit über tausend Konzepte für Fähigkeiten und Aufgaben. Fähigkeiten werden genutzt, um Anforderungen für ein Projekt zu definieren. Durch die Markierung mit Leveln werden bestimmte Fähigkeiten wichtiger als andere. Ein grobe Einteilung der Fähigkeiten in Typen ist in Abbildung 4.2 zu sehen. Die Fähigkeiten gliedern sich in charakteristische, kognitive, kommunikative, physische, technische und unternehmerische Fähigkeiten und Kompetenz. Sie werden folgendermaßen definiert und beispielhaft verwendet:

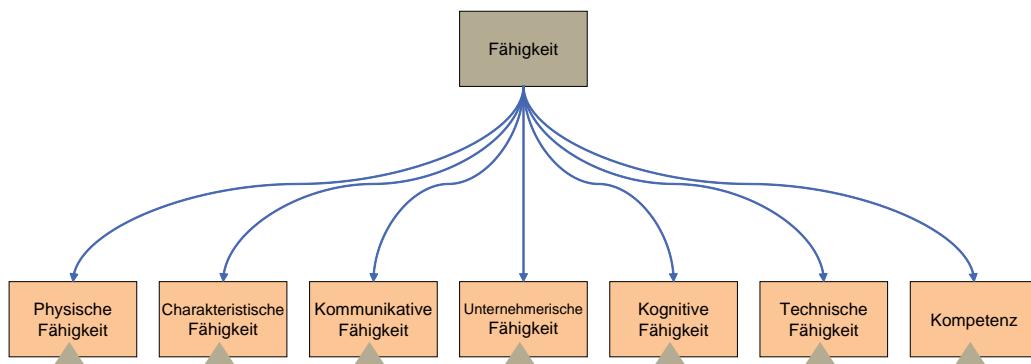


Abbildung 4.2.: Strukturierung der Fähigkeiten

- charakteristisch: Fähigkeiten, die die spezifischen, kennzeichnenden Eigenschaften einer Person betreffen.
Beispiele: Kritikfähigkeit, Flexibilität, Pünktlichkeit, engagiert
- kognitiv: Fähigkeiten, die die Wahrnehmung und das Erkennen betreffen und meist unbewusst existieren.
Beispiele: Prozessverständnis, Formulieren, Ideenvielfalt, Strukturierung
- kommunikativ: Fähigkeit, konstruktiv, effektiv und bewusst zu kommunizieren, also Informationen auszutauschen.
Beispiele: Antworten, Kundenberatung, Überzeugungskraft, Informieren
- physisch: Fähigkeiten, die eine Person in die Lage versetzen, etwas körperlich zu tun.
Beispiele: Demontage, Reparatur, absolutes Gehör, Verbinden
- technisch: Fähigkeiten, die auf bestimmte Verfahrensweisen bezogen sind oder mit Werkzeugen und Geräten durchgeführt werden.
Beispiele: Anwendungssoftwareerweiterung, Datenmodellierung, Netzwerkkentwicklung, Systemintegration
- unternehmerisch: Fähigkeiten, die das Bestehen eines Unternehmens betreffen.
Beispiele: Kostenbewusstsein, Verhandlungsgeschick, vertriebsorientiertes Handeln, unternehmerisches Denken
- Kompetenz: Eine Kompetenz ist, wie in Teil 2.1.2 beschrieben, „eine Entwicklung grundlegender Fähigkeiten, die weder genetisch angeboren noch das Produkt von Reifungsprozessen sind, sondern vom Individuum selbst

hervorgebracht werden.“ Kompetenz besteht aus den Kompetenzbereichen, die in Abbildung 4.3 dargestellt werden: Fach-Kompetenz (Bsp: Chinesisch-Kenntnis, Teilprojektleitungserfahrung, Diplom-Informatiker), Methodenkompetenz (Bsp: Konfliktlösungstechniken, logisches Denken, Budgeterstellung), Personale Kompetenz (Bsp: Einsatzbereitschaft, Sorgfältigkeit, Lernbereitschaft), Querschnittskompetenz (Bsp: Innovationsfreude, Führungs-persönlichkeit, Medienkompetenz) und Soziale Kompetenz (Bsp: Durchset-zungsstärke, Solidarität, Motivierungsvermögen)

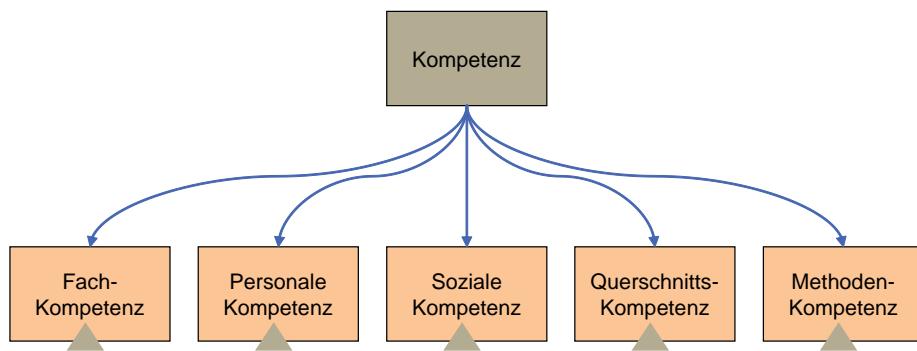


Abbildung 4.3.: Ausprägungen von Kompetenz

An der Stelle sei erwähnt, dass es sich bei der Einordnung in die verschiedenen Klassen der Fähigkeiten nicht um eine ausschließliche Einordnung handelt. Fähigkeiten können in mehreren verschiedenen Klassen vorkommen, z.B. taucht logisches Denken sowohl als kognitive Fähigkeit auf als auch als Methodenkompetenz. Die Modellierung der Fähigkeiten ähnelt zum Teil in ihrer Struktur der „Hierarchy of Capabilities“, wie durch Stader und Macintosh in [SM99] dargestellt und in ihrem Knowledge Management System eingesetzt (siehe Abschnitt 3.1.1). Auf der ersten Ebene ihrer Modellierung existieren technische Fähigkeiten, kognitive Fähigkeiten, physische Fähigkeiten sowie Projektfähigkeiten und Unternehmens-fähigkeiten.

Die Modellierung der Aufgaben wird gebraucht, um einem Projekt Aufgaben zuzuordnen, die während des Projekts gelöst werden sollen. Mitarbeitern werden gleichermaßen Aufgaben zugeordnet, bei deren Absolvierung sie beteiligt waren. Genau wie bei den Fähigkeiten wird dadurch ein Vergleich zwischen einer Projektbeschreibung und mehreren Mitarbeiterbeschreibungen möglich. Da die Modellierung der Aufgaben sehr umfangreich ist, seien hier lediglich einige Beispiele genannt:

- Kontrolle ⇒ Investitionscontrolling
- Planung ⇒ Personaleinsatzplanung
- Software-Engineering-Aufgaben ⇒ Mikrocontroller-Programmierung

Der Pfeil zeigt eine von mehreren möglichen über diverse Klassifikationsebenen hinweg gehende Spezialisierung der genannten Aufgaben an. Wie bei den Fähigkeiten, so können auch Aufgaben durchaus in mehreren Klassen erscheinen.

Die umfangreiche Menge von Synonymen ermöglicht es, syntaktisch verschiedenen Konzepten der Ontologie dieselbe Semantik zukommen zu lassen sowie die Einordnung derselben Bedeutung in verschiedenen Klassen. Beispiele von synonymen Begriffen:

1. IT-Technologie ≡ Informationstechnologie
2. Bodenkundler ≡ Geologe
3. teamorientiert ≡ teamfähig

4.2. Kriterien eines optimalen Teams

Die Frage, was ein optimales Team ist, ist nicht trivial zu beantworten und doch essentiell für die vorliegende Arbeit. Intensive Recherche und Interviews mit Siemens Teamrecruitern ergaben, dass ein Team grundsätzlich dann optimal ist, wenn es mit minimalen personellen Ressourcen und vorhandenen Fähigkeiten in einem definierten Zeitraum mit einem vorgegebenen Budget eine Aufgabe oder ein Problem lösen kann. Deiters [DL00] umschreibt das Ziel eines Skillmanagement-Systems mit Teamplanungsfunktionalität folgendermaßen: „[...] besteht das Ziel darin, ein ausgewogenes Team zusammenzustellen, welches in der Summe möglichst alle benötigten Kenntnisse in sich vereint. Die Kriterien, anhand derer beurteilt werden kann, ob ein Team ausgewogen ist, ergeben sich aus der Gesamtaufgabe, die das Team gemeinsam bearbeitet.“ Diese generelle Definition wird durch die folgenden Aussagen über optimale Teams noch weiter vertieft und verfeinert. Sie bilden sowohl die Grundlage für die Teamplanungsalgorithmen als auch die Evaluation der gebildeten Teams.

Teamgröße

Eine zentrale Eigenschaft von Teams bildet für viele Autoren die Teamgröße. Katzenbach [JRK93] schreibt dazu, dass ein arbeitsfähiges Team eine kleine Mitgliederanzahl besitzen sollte. Für sein Buch untersuchte er verschiedene Teams mit 2 bis 25 Mitarbeitern, die meisten hatten jedoch unter 10 Mitglieder. Die Gefahr bei größeren Teams besteht darin, dass unter Umständen nicht genügend Platz für alle an einem Ort existiert und gemeinsame Terminabsprachen kompliziert werden. Solche Teams zerfallen meist in diverse Untergruppen.

Fuchs-Kittowski [FK03] arbeitete heraus, dass als ideale Teamgröße häufig eine Mitgliederanzahl von fünf angesehen wird und bei einer deutlich höheren Zahl damit zu rechnen sei, dass die Gruppe in mehrere Teams zerfällt. Lindemann [Lin06]

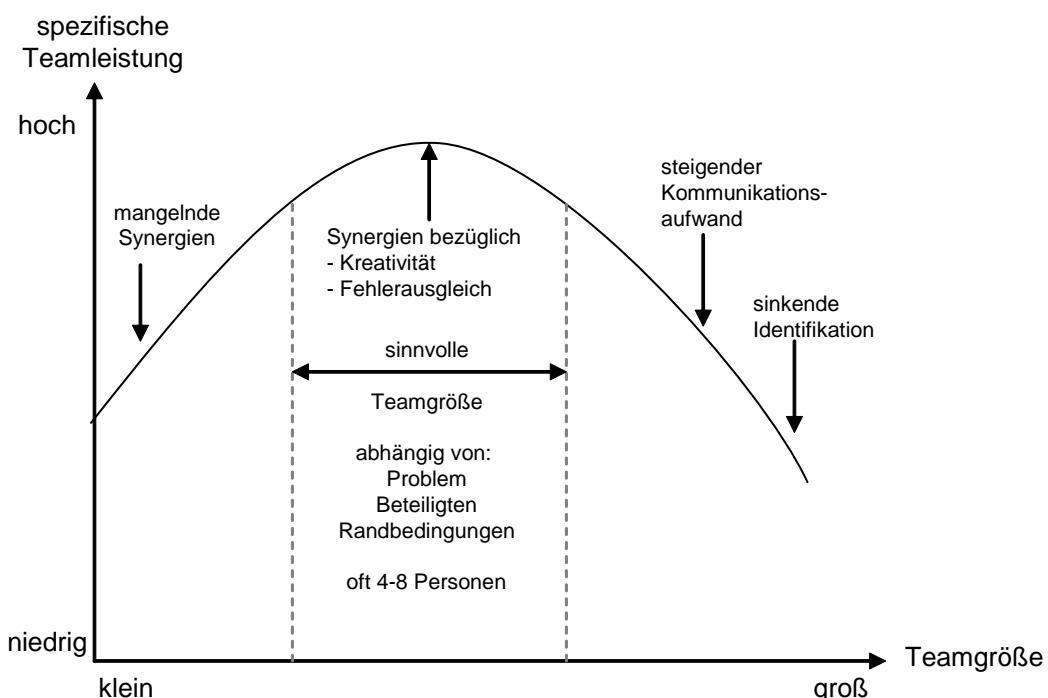


Abbildung 4.4.: Teamleistung in Abhängigkeit der Teamgröße, Quelle: [Lin06]

verschärft diese Aussagen noch, indem er die Teamleistung von der Teamgröße abhängig macht und eine sinnvolle Teamgröße bei vier bis acht Personen ansetzt. Seiner Ansicht nach sinkt in zu großen Teams die Identifikation mit dem Team bei gleichzeitigem Anstieg des Kommunikationsaufwandes. In Abbildung 4.4 erkennt man außerdem, dass mangelnde Synergien schließlich die Folge zu kleiner Teams sind.

Heterogenität - Homogenität

Die Zusammensetzung eines Teams kann bezüglich der Eigenschaften der Mitarbeiter eher homogen oder heterogen ausfallen. Welche Faktoren die Heterogenität oder Homogenität eines Teams bestimmen, zeigt Tabelle 4.1 in Anlehnung an Jackson et al. [JMW95]. Sie unterteilen diese Faktoren in aufgaben- und beziehungsorientierte Attribute, die jeweils entweder sichtbar oder versteckt sein können.

	aufgabenbezogen	beziehungsorientiert
sichtbare Attribute	Abteilungs- und Gruppenzugehörigkeit	Geschlecht
	Dauer der Organisationszugehörigkeit	Alter
	Zeugnisse und Titel	Nationalität
	Bildungsstand	Ethnische Herkunft
	Mitgliedschaft in beruflichen Vereinigungen	Religion
		Politische Gesinnung
versteckte Attribute		Physische Erscheinung
	Wissen und Expertise	Sozio-ökonomischer Status
	Mentale Fähigkeiten	Eigenschaften
	physische Fähigkeiten	Werte
	Erfahrung	Persönlichkeit

Tabelle 4.1.: Bereiche der Heterogenität der Teammitglieder

Autoren, die sich mit diesem Thema auseinandersetzen, haben ganz unterschiedliche Meinungen darüber, ob heterogene oder homogene Teams geeigneter sind. Belbin [Bel04] schreibt, dass Teams, in denen sich die Personen ihrem Profil nach zu sehr ähneln, weniger produktiv sind als gemischte Teams. Von der Oelsnitz [Oel07] ist der Meinung, dass Forschungs- und Entwicklungsteams einen höheren Grad an Heterogenität benötigen als Teams, die eher koordinativ-integrative Aufgaben wahrnehmen. Die unterschiedlichen gedanklichen Zugänge und Interpretationen würden zusätzliche Einsichten eröffnen und das Erkennen zusätzlicher

Handlungsoptionen ermöglichen. Magjuka, Baldwin [MB91] und Vernon [Ver03] sind zu dem Ergebnis gekommen, dass Teams, in denen die Mitglieder über ganz unterschiedliches Wissen und Fähigkeiten verfügen, dieses selbst als effektiver bewerten. Steinmann und Schreyögg [SS91] haben verschiedene Studien untersucht und festgestellt, dass jedoch in der Regel homogene Teams über eine höhere Kohäsion verfügen und aus diesem Grund auch zufriedener sind. Einen direkten Zusammenhang zwischen Homogenität und Zufriedenheit hat Vernon allerdings empirisch nicht feststellen können. Hackman [Hac87] ist zu dem Schluss gekommen, dass es in einer Gruppe eine Balance zwischen Homogenität und Heterogenität geben muss. Nur so können die Nachteile der Homogenität durch die Vorteile der Heterogenität und umgekehrt ausgeglichen werden.

Zusammenfassend kann vermerkt werden, dass die Mehrheit der genannten Autoren in Heterogenität als ein wichtiges Merkmal von Teams einschätzen. Heterogene Teams seien geeigneter und arbeiten effizienter.

Transaktives Wissen und Überlappungen

Von der Oelsnitz [Oel07] untersuchte die Bedeutung von transaktivem Wissen für Teamarbeit. Darunter versteht man das wechselseitige Wissen der Teammitglieder darüber, wer in einem Team welche Fähigkeiten besitzt und was der Einzelne kann. Es dient dazu, eine effiziente Informationsverarbeitung bzw. Kompetenzkoordination zu ermöglichen und bildet eine Vorbedingung, um sinnvoll miteinander zu kooperieren. Transaktives Wissen schafft damit eine gemeinsame Verständigungsbasis durch die gesteigerte Sensibilität im Hinblick auf unterschiedliche disziplinäre Standpunkte und Problemzugänge. In diesem Fall werden teaminterne Diskussionen ermöglicht, in denen eine Lösung aus verschiedenen Blickwinkeln betrachtet wird. Von der Oelsnitz zitiert Mohrman [Moh95]: „Teams require members to have, at a minimum, enough understanding of the skills of their teammates to be able to discuss issues and trade-offs as the team goes through the cycle of considering divergent views and arriving at convergence on a direction.“ In der Entwicklung der Applikation muss darauf geachtet werden, dass genügend Überlappungen der Mitarbeiterkompetenzen existieren, um die Entstehung solcher Diskussionen zu zulassen. Wübbelmann [Wüb01] schreibt zu Diskussionen, dass die Teammitglieder so in der Lage sind, eigene wertvolle Ideen zu fachlichen Fragen einzubringen, fachliche Vorschläge zu unterbreiten und Ergebnisse zu bewerten. Gemein-

sam würden sie schneller den Kern eines fachlichen Problems erkennen und einander Vor- und Nachteile verschiedener Alternativen aufzeigen können. Gleichzeitig sollte in einem Team aber eine Deckungsgleichheit der Mitarbeiterfähigkeiten vermieden werden, da dies nach von der Oelsnitz [Oel07] Kompetenzredundanz nach sich zieht, das heißt die Aufgabe wäre theoretisch auch durch weniger Mitarbeiter realisierbar. Transaktives Wissen in Teams ist letztlich also notwendig, um korrekte Informationszuweisungen und -verknüpfungen herzustellen, Doppelarbeit zu vermeiden, Kompetenzstreitigkeiten zu verhindern und den internen Vorgang des Wissensaufbaus zu unterstützen.

Wissenslücken

Fehlende Fähigkeiten bedeuten Wissenslücken in einem Team. Sie treten laut Stader [SM99] dann auf, wenn die Fähigkeit:

- nicht vorhanden ist
- bei zu wenigen Mitarbeitern verfügbar ist
- nicht in einem ausreichenden Maß an Erfahrung und Expertise vorhanden ist
- nicht für das entsprechende Anwendungsgebiet genutzt werden kann oder
- bei Mitarbeitern vorhanden ist, die an ihr kein Interesse haben

Bei der Zusammensetzung eines optimalen Teams sollte versucht werden, Wissenslücken möglichst zu vermeiden, um die Aufgabenlösung im zeitlichen Rahmen zu schaffen. Wissenslücken müssen allerdings nicht immer gleich das Scheitern des Teams bedingen. Katzenbach [JRK93] sieht in Teams auch ein Instrument zum Lernen und zur Weiterentwicklung und schreibt, dass Wissenslücken sogar behebbar sind, wenn die Teammitglieder die Lücken erkennen und versuchen, diese durch einen Lernprozess zu schließen. Dieser Antrieb entsteht seiner Meinung nach durch die gemeinsame Angst, mit dem Team zu versagen.

Bei der Konzeption der Teamplanungsalgorithmen muss darauf geachtet werden, dass möglichst keine Teams mit Wissenslücken generiert werden. Sollten sie sich jedoch aufgrund der vorhandenen Mitarbeiter nicht vermeiden lassen, müssen Teams mit Wissenslücken in der Evaluation entsprechend abgewertet werden. Teams mit geringen Wissenslücken müssen, wie durch Katzenbach erörtert, nicht

zum Scheitern des Teams und damit des Projekts führen. Sie können mit lernbereiten Mitarbeitern dennoch produktiv sein.

Rollen

Belbin [JRK93] definiert eine Rolle als „A team role is a pattern of behaviour characteristic of the way in which one of the member interacts with others in facilitating the progress of the team.“ Die sogenannten Belbin-Rollen sind die Grundlage vieler Teamzusammensetzungen und beruhen auf Untersuchungen Belbins, inwiefern sich verschiedene Persönlichkeitstypen auf die Teamleistung auswirken. Seiner Aussage nach wären Teams dann effektiv, wenn sie aus einer Vielzahl heterogener Persönlichkeits- und Rollentypen bestehen. Er unterscheidet drei Haupttypen, die jeweils drei der neun Teamrollen umfassen:

- 3 handlungsorientierte Rollen
 - Shaper: Der „Macher“ übernimmt die Verantwortung für das Team. Er formuliert Teilziele, sucht Strukturen und sorgt dadurch für eine schnelle Entscheidungsfindung. Außerdem veranlasst er die Erledigung von Aufgaben.
 - Implementor: Die Rolle des „Umsetzers“ setzt formulierte Konzepte in durchführbare Arbeitspläne um ist verantwortlich für eine strukturierte Vorgehensweise.
 - Completer: Der „Perfektionist“ achtet auf eine genaue Einhaltung von zeitlichen Vorgaben und wird insbesondere dann wichtig, wenn die Arbeitsweise des Teams zu oberflächlich oder langsam wird.
- 3 kommunikationsorientierte Rollen
 - Coordinator: Der „Koordinierer“ stimmt den Arbeitsprozess ab, setzt Ziele und Prioritäten, erkennt relevante Problemstellungen und delegiert Aufgaben an jene Kollegen, die zu deren Erledigung am besten geeignet sind. Insbesondere eignen sich „Koordinierer“ aufgrund ihrer Eigenschaften als Teamleiter.

- Teamworker: Die Rolle des „Arbeiters“ erzeugt ein angenehmes Arbeitsklima und Harmonie. Er schafft es, auch introvertierte Teammitglieder zur aktiveren Teilnahme zu motivieren.
- Resource Investigator: Der „Wegbereiter“ stellt die Kontakte zur Welt außerhalb des Teams her. Die so gefundenen Quellen werden durch ihn für die Ideenfindung innerhalb des Teams genutzt.

- 3 wissensorientierte Rollen

- Plant: Die Rolle des „Erfinders“ wird bei schwierigen Problemen wertvoller, da er weiterhin Lösungen findet.
- Monitor Evaluator: Der „Beobachter“ berücksichtigt relevante Lösungs- und Handlungsmöglichkeiten des Teams und verfügt über ein gutes Urteilsvermögen.
- Specialist: Der „Spezialist“ gleicht Informationsdefizite des Teams aus, in dem er sein unfangreiches und notwendiges Fachwissen beiträgt.

Bäumgen und Hübber [BH04] beschreiben einen Projektablauf bei KOWIEN (siehe Abschnitt 3.1.4) mit dem Beginn der Bestimmung eines Projektleiters, der möglichst viel Projekterfahrung, Projektleiterkompetenz, soziale Kompetenz und spezifisch fachliche Kompetenzen besitzen sollte. Der Projektleiter bildet also die Basis des Projektes und wählt die Projektprozesse aus. Anschließend definiert er die ausführenden und mitwirkenden Rollen mit bestimmten Funktionen anhand der Kompetenzen der Teammitglieder. Rollen basieren auf empirisch ermittelten sozial-psychologischen Eigenschaften der Teammitglieder. Das Erfassen und Auswerten dieser „weichen“ Kenntnisse übersteigen jedoch den Rahmen der vorliegenden Arbeit, weshalb auf die Benennung von Rollen innerhalb des Teams verzichtet wird. Lediglich die Markierung eines Teamleiters sollte durch die Applikation gewährleistet werden, da diese Position ein Mindestmaß an Steuerung und Koordination in einem Team gewährleistet und die Schnittstelle nach außen bildet. Ein Interview mit einem Siemens Teamrecruiter ergab, dass sich die Rolle eines Teamleiters in jedem Fall herausbilden wird, auch wenn niemand diese Position fest besetzt.

Die Anwendung wird angelehnt an das KOWIEN-Vorgehen, das Teammitglied als Teamleiter aussuchen, das insgesamt am kompetentesten für die Projektaufgabe erscheint. Je mehr relevante Fähigkeiten ein Teamleiter besitzt, um so häufiger und effizienter kann er Entscheidungsprozesse voranbringen.

Verfügbarkeit

Deiters [DL00] schreibt zum Punkt der Verfügbarkeit: „ein leistungsfähiges System optimiert den Abgleich dahingehend, dass nur die Personen gemeldet werden, die zeitlich komplett verfügbar sind oder bei denen die geringsten zeitlichen Konflikte auftreten.“ Laut West [Wes03] ist die richtige Zusammensetzung eines erfolgreichen Teams ein zweidimensionales Matching-Problem: Mitglieder müssen einerseits ihren Fähigkeiten entsprechend zu passenden Aufgaben zugeordnet werden und andererseits müssen die zu einem Team kombinierten Mitglieder menschlich und fachlich miteinander harmonieren können. Da Personen über unterschiedliche methodische und technische Kompetenzen verfügen, müssen sie den Aufgaben jeweils zugeordnet werden, die sie am besten erfüllen können, damit sie weder unter noch überfordert werden. Das bedeutet, dass jedes Mitglied eine exakte Aufgabe innerhalb des Teams wahrnimmt. Um so wichtiger ist damit die Verfügbarkeit jedes ausgewählten Teammitglieds über den gesamten Projektzeitraum. Der Ausfall eines Teammitglieds, beispielsweise durch Arbeit in anderen Projekten oder durch plötzlich eintretende Situationen wie Krankheit während eines laufenden Projektes, kann also unter Umständen nicht nur zu zeitlichen Problemen führen sondern sogar den Misserfolg eines Projektes bedeuten. Der Teamrecruiter sollte in der Lage sein, unmittelbar vor Erstellung einer Teamzusammensetzung nicht verfügbare Mitarbeiter als solche zu kennzeichnen, um andere Projekte nicht zu gefährden. Ein Team kann auch bei einem Ausfall eines Teammitglieds durchaus optimal bleiben, wenn möglichst schnell ein ähnlicher Mitarbeiter gefunden wird, der den ausfallenden ersetzen kann. Für den Fall, dass ein Ausfall eines Mitglieds einfach durch andere Teammitglieder vertreten werden kann, kann davon ausgegangen werden, dass es sich nicht um ein optimales Team gehandelt hat. Dann wäre Kompetenzredundanz vorhanden und der ausfallende Mitarbeiter sowieso nicht notwendig gewesen (siehe Transaktives Wissen 4.2).

Zusammenfassung

Aus den gewonnenen Erkenntnissen können die Eigenschaften der Teamplanungsalgorithmen und der Evaluation abgeleitet werden, damit möglichst optimale Teamzusammensetzungen erstellt werden können. Die zu bildenden Teams müssen eine geeignete Teamgröße besitzen, um Synergien nutzen zu können. Ein dedizierter,

kompetenter Teamleiter und weitere Teammitglieder müssen sich in der Projekt domäne oder möglichst ähnlichen Themenfeldern auskennen. Die Zusammensetzung sollte bezüglich der Kenntnisse möglichst heterogen ausfallen. Die Fähigkeiten der Mitglieder müssen sich gleichzeitig derart überlappen, dass ein Verständnis für fremde Kompetenzen (transaktives Wissen) innerhalb des Teams existiert und fachliche Diskussionen entstehen können. Wissenslücken sollten vermieden werden, in dem relevante Fähigkeiten in einem ausreichenden Maß an Erfahrung vorhanden sind.

4.3. Einschränkungen der Auswahlmenge

Je nach Unternehmensgröße kann die Menge der Mitarbeiter, aus denen Teams für ein Projekt gebildet werden können, sehr groß sein, was die Berechnungen der Mitgliederauswahl und Evaluation entsprechend aufwendiger macht. Deshalb erscheint es sinnvoll und durchaus notwendig, die Auswahlmenge bereits vor der Teamselektion zu reduzieren. Um dies in der Anwendung der vorliegenden Arbeit zu leisten, werden zwei Ansätze verfolgt. Einerseits wird die Auswahlmenge um nicht verfügbare Mitarbeiter verringert und andererseits sollen deckungsgleiche Mitarbeiter bezüglich ihrer Fähigkeiten anhand eines Clusterings nicht in die Auswahlmenge gelangen.

Verfügbarkeit

Es werden für jeden Mitarbeiter bestimmte Daten gespeichert: Fähigkeiten, beteiligte Aufgaben und persönliche Daten. Bei den letzteren wird auch erfasst, von wann bis wann ein Mitarbeiter mit einer Aufgabe beschäftigt ist. Diese Datumsangaben ermöglichen durch einen Vergleich mit dem Projektzeitraum eine automatische Selektion verfügbarer Mitarbeiter.

Der Teamrecruiter hat darüber hinaus die Möglichkeit, Mitglieder als nicht verfügbar zu markieren, z.B. weil sie wegen Krankheit oder Urlaub nicht eingesetzt werden können.

Clustering der Kompetenzen

Deckungsgleichheit wird durch von der Oelsnitz [Oel07] definiert und bedeutet, dass mehrere Mitarbeiter dieselben Fähigkeiten besitzen wodurch eine Kompetenzredundanz entsteht, da die einzelnen Mitarbeiter keinen Kenntnisgewinn anbieten. Diese Deckungsgleichheit sollte vermieden werden, da mit gleichen Mitarbeitern keine heterogenen Teams gebildet werden können, was ausführlich in Abschnitt 4.2 erläutert wird.

Um zu gewährleisten, dass deckungsgleiche Mitarbeiter nicht gemeinsam im Auswahlprozess enthalten sind, wird einmalig vor Beginn der ersten Teamplanung ein Clustering über die Fähigkeiten der Mitarbeiter angewendet. Das bedeutet, dass deckungsgleiche Mitarbeiter bezüglich ihrer Fähigkeiten im selben Cluster gruppiert werden, sie sind sozusagen in derselben Mitarbeiterkategorie einzuordnen. Für die Selektion von Teammitgliedern gilt es dann, einen Mitarbeiter je Cluster für die Auswahlmenge zu ermitteln. Im Ergebnis können die anderen im jeweiligen Cluster beteiligten Mitarbeiter als Alternative vorgeschlagen werden.

Solange sich die Fähigkeiten der Mitarbeiter nicht ändern, können die generierten Cluster behalten und genutzt werden. Erst nach Änderung der Mitarbeiterfähigkeiten muss ein neues Clustering erzeugt werden.

4.4. Teamplanungsalgorithmen

Bei der Konzipierung einer Lösung zur effizienten Zusammensetzung von Teams ist die Wahl der Informationen, auf denen die Teamplanung beruhen soll, ein wichtiger Aspekt. Angelehnt an die bisher vorgestellten Ergebnisse und die Vorgehensweise bei der Siemens AG werden folgende Profile definiert:

- **Projektanforderungsprofil:** Enthält alle relevanten Informationen eines Projekts. Das sind einerseits Basisinformationen wie Projekttitel, Datum des Beginns und Enddatum. Andererseits werden Fähigkeiten, die für die Arbeit im Projekt benötigt werden, mit dem Profil verknüpft. Unter Angabe eines Levels pro Fähigkeit erreicht man eine Gewichtung. Weiterhin werden die Aufgaben erfasst, die während des Projekts erarbeitet werden sollen.

- **Mitarbeiterkompetenzprofil:** Enthält alle relevanten Informationen über einen Mitarbeiter. Neben einigen persönlichen Daten wie Name, Geschlecht und Alter werden für einen Mitarbeiter ebenfalls seine gewichteten Fähigkeiten sowie erledigte Aufgaben gespeichert.

Zum Verständnis der Algorithmen dienen weiterhin die folgenden Parameter:

- P-Skills: Menge der Fähigkeiten, die für zur Erfüllung der Aufgaben des Projekts benötigt werden
- P-Tasks: Menge der Aufgaben, die in einem Projekt gelöst werden sollen
- E-Skills: Menge der Fähigkeiten eines Mitarbeiters
- E-Tasks: Menge der Aufgaben, bei dessen Lösung ein Mitarbeiter bereits mitgearbeitet hat

Im Rahmen dieser Arbeit werden drei semantische Algorithmen zur Zusammensetzung von Teams konzipiert.

Die drei Ansätze erhalten jeweils als Eingabe ein Projektanforderungsprofil und die Profile der Mitarbeiter. Sie unterscheiden sich in den Informationen, die aus den Profilen zur Berechnung der Teamzusammensetzungen verwendet werden. Je fortgeschritten der Ansatz, desto umfangreicher werden Angaben der Profile bezüglich projektrelevanter Informationen betrachtet.

Die Ausgabe ist bei allen Algorithmen dieselbe: eine Liste von bewerteten Teamzusammensetzungen, deren Aufbau sich nach Kriterien möglichst optimaler Teams richtet. Der Teamrecruiter, dessen Aufgabe die effiziente Zusammenstellung von Projektteams ist, kann anhand der Ergebnisse entscheiden, welche Mitarbeiter des Unternehmens Mitglieder des Teams werden.

Der Fokus beim Bau der Algorithmen sowie der umgebenden Anwendung liegt auf der Ganzheitlichkeit einer Lösung, nicht im Hinblick auf ein komplettes Kompetenzmanagement-System aber auf die in einem solchen System vorkommende Komponente Teamplanung. Das bedeutet, die Lösung beruht auf einer umfassenden Berücksichtigung möglichst vieler Aspekte und Zusammenhänge im Hinblick auf Teamplanung, wie in den Anwendungsfällen in Abschnitt 3.3 erwähnt. Weiterhin sollen sie komplettete Teams für die Aufgaben eines Projekts bilden und nicht ausschließlich einzelne Mitarbeiterkompetenzprofile auf ihre Geeignetheit hin untersuchen (siehe Abschnitt 3.1.5 beschrieben, ePeople). Der Teamrecruiter soll mit der Anwendung die Möglichkeit erhalten, ohne Konfiguration einzelner

Parameter, wie z.B. Bestimmung der Genauigkeit der Ähnlichkeitsvergleiche (siehe Ergebnisse in Abschnitt 3.1.3, KMDL), möglichst einfach und unaufwendig geeignete Teamzusammensetzungen für eine gegebene Projektbeschreibung zu finden. Die unter 3.3 beschriebenen Anwendungsfälle sollen ihn auch bei seiner Tätigkeit unterstützen, wenn basierend auf fachlichen Eigenschaften der vorhandenen Mitarbeiter kein optimales Team eingesetzt werden kann.

Im Folgenden wird erst der permutative Lösungsansatz vorgestellt, dessen Eigenschaften in weiten Teilen dem Lösungsansatz der KMDL entsprechen (3.1.3). Die in der Implementierung verwendeten Mechanismen stellen jedoch aufgrund unzureichender Beschreibungen lediglich eine Annäherung an die von Rüßbüldt präsentierten Ergebnisse dar. Zusätzlich wird eine auf Permutationen basierende Vorgehensweise dem sukzessiven Ansatz vorgezogen. Mit der Umsetzung des permutativen Algorithmus können deshalb dessen Ergebnisse in Bezug auf Umfang und Güte mit den entwickelten semantischen Ansätzen verglichen werden. Dies lässt außerdem Rückschlüsse auf die Qualität verwandter Ansätze zu.

Anschließend werden die semantischen Algorithmen dargestellt, die durch Umsetzung der Kriterien optimaler Teams zusammen mit deren Evaluation den Schwerpunkt dieser Ausarbeitung bilden. Sie erzeugen während der Berechnung eine Skill-Matrix. Die Spalten dieser bilden allgemein die Projektanforderungen (P-Skills und P-Tasks), die Zeilen repräsentieren die Mitarbeiter des Unternehmens. Einträge in der Matrix markieren die höchste Ähnlichkeit einer der Mitarbeitereigenschaften (E-Skills und E-Tasks) zu einer Projektanforderung. Damit bildet die Matrix die Grundlage bei der Auswahl von Mitarbeitern für Teams.

Die gefüllte Matrix erlaubt es, neben den einzelnen Ähnlichkeiten zwischen den Fähigkeiten weitere Maße zur Güte einer Teamzusammensetzung abzulesen. Im Folgenden werden sie zusammengefasst:

- normalisierte Zeilensumme: Das ist der fachliche Wert, den ein Mitarbeiter durchschnittlich einem speziellen Projektteam beitragen kann. Die normalisierte Zeilensumme entsteht durch Bilden des Mittelwert einer Matrixzeile.
- normalisierte Spaltensumme: Das ist der durchschnittliche Wert, mit dem eine bestimmte Fähigkeit durch die gewählten Mitglieder im Projektteam vertreten ist. Der Wert entsteht durch Bilden des Mittelwerts einer Matrixspalte.

- Funktionalwert: Dies ist der Mittelwert über die fachlichen Werte aller Teammitglieder.

Durch weitergehende Interpretationen der Skill-Matrix können zusätzliche für das Unternehmen relevante Informationen gewonnen werden. Das Bilden der normalisierten Spaltensummen über im gesamten Unternehmen vorkommende Mitarbeiter und Fähigkeiten erzeugt eine Aufstellung markant vertretener (Kern-) Kompetenzen sowie möglicher Wissenslücken.

4.4.1. Permutativer Ansatz

Der permutative Algorithmus ist der einfachste untersuchte Ansatz zur Teamplanung. Er heißt permutativ, weil er alle möglichen Kombinationen von Mitarbeitern zu Teams generiert.

Als Eingabe dienen die Mitarbeiterkompetenzprofile der verfügbaren Mitarbeiter des Unternehmens sowie ein Projektanforderungsprofil. Zusätzlich wird die Menge um diejenigen Mitarbeiter reduziert, die keine der Fähigkeiten aus dem Projektanforderungsprofil besitzen, mit dem Ziel, die Anzahl der möglichen Permutationen zu reduzieren. Diese Mitarbeiter würden aufgrund einfacher Soll-Ist-Vergleiche im nicht-semantischen Umfeld keine Berücksichtigung finden. Die Reduktion nimmt somit keinen negativen Einfluss auf die Ausgabe des Algorithmus. Es wird hierdurch bei einem großen Anteil von ungeeigneten Mitarbeitern ein erheblicher Performance-Gewinn erzielt, da die Größe der Eingabe und somit die Rechenzeiten verringert werden können.

Der Algorithmus betrachtet die Fähigkeiten der Mitarbeiter und vergleicht diese mit den Fähigkeiten in der Projektanforderung. Für jeden Mitarbeiter, der die Projektanforderungen erfüllt, wird rekursiv ein Baum gebildet, in dem an jedem inneren Knoten ein weiterer Mitarbeiter zu einem Team hinzugefügt wird. Auf diesem Baum wird dann eine Tiefensuche durchgeführt, wobei jeder Pfad von der Wurzel bis zu einem Blatt ein mögliches Team darstellt.

Um sicherzustellen, dass die Menge der hierbei gefundenen Permutationen auf diejenigen Teams beschränkt bleibt, in denen alle Projektanforderungen vollständig erfüllt sind, wird auf jeder Ebene des Baumes eine Reduktion der auf dem Weg zu einem Blattknoten zu ergänzenden Fähigkeiten vorgenommen. Im Rahmen der Rekursion des Algorithmus werden in jedem Rekursionsschritt die Fähigkeiten des

hinzugefügten Mitarbeiters aus der Menge der verbleibenden durch das Team zu erfüllenden Fähigkeiten entfernt. Durch diese stetige Minimierung der Anforderungen wird der Rekursionsanker definiert, wobei unvollständige Pfade (es werden an einem Blattknoten nicht alle Projektfähigkeiten erfüllt) keine Teams definieren, die in die Ausgabe des Algorithmus eingehen. Ebenfalls werden auf einem Pfad keine weiteren Mitglieder gesucht, wenn das aktuelle Team alle Anforderungen erfüllt. Mit der Menge der gültigen Permutationen entsteht die Menge aller vollständigen Teams, deren Mitglieder alle Projektanforderungen erfüllen. Der Algorithmus sieht im Pseudocode folgendermaßen aus:

Prozedur: Füge Mitglied zu Team hinzu

Zweck: Rekursive Funktion zur Bildung vollständiger Teams
hinsichtlich der Projektanforderungen

Parameter: Liste von Mitarbeitern im Team, Liste mit
verbleibenden Fähigkeiten aus Projektanforderung

Ergebnis: Menge von Teams, die der Tiefensuche im
generierten Baum entspricht

- 1 Falls keine verbleibenden Fähigkeiten aus
Projektanforderungen
- 2 Füge Team zur Resultatliste der möglichen Teams hinzu
- 3 Return. //Rekursionsanker
- 4 Für jeden Mitarbeiter aus der Liste der verfügbaren
Mitarbeiter
- 5 Wenn Mitarbeiter hat Fähigkeit aus Liste der
verbleibenden Projektfähigkeiten
- 6 Füge Mitarbeiter zu Team hinzu, wenn nicht bereits
Mitglied
- 7 Entferne Mitarbeiterfähigkeiten aus Liste der
verbleibenden Projektfähigkeiten
- 8 Füge Mitglied zu Team hinzu (Team, verbleibende
Projektfähigkeiten) //Rekursionsschritt

4.4.2. Erster Semantischer Ansatz

Für den ersten semantischen Ansatz werden als Eingaben das Projektanforderungsprofil und die Mitarbeiterkompetenzprofile benötigt, aus denen die Fähigkeiten für die Berechnung der Geeignetheit betrachtet werden. Die verwendeten Fähigkeiten sind nicht gewichtet, womit sie alle als gleich wichtig betrachtet werden.

Grob gesehen durchläuft der Algorithmus die folgenden Schritte, was bildlich in Abbildung 4.5 dargestellt wird:

1. Ähnlichkeit zwischen allen P-Skills und allen E-Skills berechnen.
2. Jeweils die höchste Ähnlichkeit in der Skill-Matrix (E,P) speichern.
3. Mitgliederauswahl.
4. Evaluation.

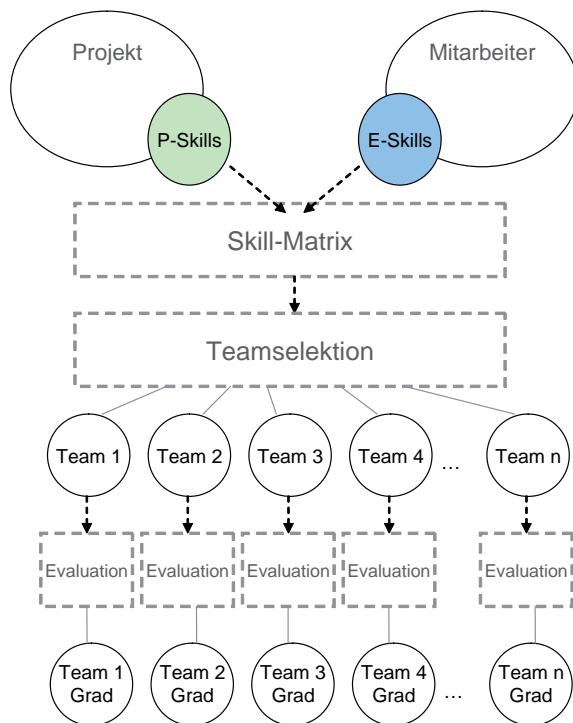


Abbildung 4.5.: Ablauf des ersten semantischen Algorithmus

Die Berechnung der Ähnlichkeit wird anhand der HR-Ontologie zwischen allen Projektfähigkeiten und allen Mitarbeiterfähigkeiten vorgenommen. Dabei ist allerdings jeweils die höchste Ähnlichkeit zwischen je einer Projektfähigkeit und allen Mitarbeiterfähigkeiten für die Berechnung relevant. Weniger ähnliche Fähigkei-

ten werden nicht im Auswahlprozess betrachtet, weil diese nicht maßgeblich zur Ausübung der Projektaktivität beitragen. Ein Eintrag x in der Skill-Matrix bedeutet dann, dass ein Mitarbeiter E (Zeile) mindestens eine Fähigkeit besitzt, die mit dem Wert x ähnlich zu einer Projektfähigkeit P (Spalte) ist. Die errechneten Ähnlichkeiten in der Matrix haben einen Wert zwischen 0 und 1. Die Eigenschaften der Ähnlichkeitsfunktion werden im Kapitel 5 beschrieben.

In Abbildung 4.6 wird ein Beispiel für die Skill-Matrix des ersten Algorithmus gezeigt.

Mitarbeiter Projekt	Skill A	Skill B	normalisierte Zeilensumme
1 	0,8	0,7	$1,5/2 = 0,75$
2 	1	0,2	$1,2/2 = 0,6$
3 	1	1	$2/2 = 1$
Normalisierte Spaltensumme	$2,8/3 = 0,93$	$1,9/3 = 0,63$	Funktionalwert 0,78

Abbildung 4.6.: Beispiel für die Skill-Matrix des ersten semantischen Ansatzes

Das Unternehmen besteht aus drei Mitarbeitern. Zur Projektdurchführung werden die Fähigkeiten A und B benötigt. Der erste Eintrag 0,8 bedeutet, dass Mitarbeiter 1 diverse Fähigkeiten besitzt, von denen eine mit 80% am ähnlichsten zu Skill A ist. Er besitzt keine weiteren Fähigkeiten, die zur Projektanforderung A eine höhere Ähnlichkeit besitzen. Der Eintrag 1 an drei Stellen bedeutet, dass die entsprechenden Mitarbeiter genau die geforderten Skills besitzen. Da keine größere Ähnlichkeit als 1 erreicht werden kann und irrelevant ist, welche der Mitarbeiterfähigkeiten in die Skill-Matrix eingetragen wurde, kann der Algorithmus an dieser Stelle abbrechen und zum Vergleich der nächsten Projektfähigkeit mit dem aktuellen Mitarbeiter übergehen. 0,2 markiert für den Mitarbeiter 2 eine recht geringe Ähnlichkeit zu Skill B des Projekts. Er besitzt ebenfalls keine die zweite Projektanforderung besser erfüllende Fähigkeit.

Die ausgefüllte Skill-Matrix ermöglicht schließlich die Berechnung der normalisierten Zeilen- und Spaltensummen. Dabei wird ersichtlich, dass Skill A mit einem

Wert von 0,93 für die gewählten Mitarbeiter in einem Projekt besser vertreten ist, als Skill B mit 0,63. Da in diesem Fall die Skill-Matrix für das gesamte Unternehmen gebildet wurde, kann an der höheren normalisierten Spaltensumme abgelesen werden, dass Skill A wahrscheinlich eine der Kernkompetenz des Unternehmens bildet. Diese Berechnung muss allerdings mit allen Fähigkeiten vorgenommen werden, deren Ausprägung man überprüfen möchte, um eine realistische Aussage machen zu können. Den größten fachlichen Wert für das Projekt besitzt Mitarbeiterin 3. Sie erfüllt alle Projektanforderungen zu 100% und wird deshalb, wie in Abschnitt 4.2 erläutert, als kompetentestes Teammitglied für die Rolle des Teamleiters gekennzeichnet.

Die Selektion der Teams erfolgt anhand der Skill-Matrix. Dafür werden jeweils pro Projektfähigkeit mindestens zwei Mitarbeiter gewählt, die für diese Fähigkeit die höchsten Ähnlichkeiten besitzen. Dadurch wird gewährleistet, dass jede der im Projektanforderungsprofil enthaltenen Fähigkeiten durch mindestens zwei Teammitglieder ausgeführt werden kann, wodurch transaktives Wissen innerhalb des Teams entsteht (siehe 4.2). Im Beispiel aus Abbildung 4.6 sind das für Skill A Mitarbeiter 2 und 3 und für Skill B Mitarbeiter 3 und 1. Gleichzeitig müssen sie im Vergleich untereinander die höchsten normalisierten Zeilensummen haben. Dadurch werden genau die Mitarbeiter für Teams ausgewählt, die besonders gut für die Fähigkeiten qualifiziert sind und zusätzlich einen hohen durchschnittlichen fachlichen Wert haben. Sollten in einer Spalte beispielsweise 10 Mitarbeiter mit einer Ähnlichkeit von 1 erfasst sein, so werden nur diejenigen für Teams selektiert, die darüber hinaus den höchsten fachlichen Wert besitzen, also mit hoher Wahrscheinlichkeit auch für andere Projektfähigkeiten relevant sind.

Die gebildete Menge dieser Mitarbeiter wird anschließend anhand derselben Baumstruktur, wie beim permutativen Ansatz beschrieben, zu Teams von 4 bis 10 Mitgliedern permutiert. Im Vergleich zum permutativen Ansatz ist die Menge der zu permutierenden Mitarbeiter weitaus geringer und besteht aus den geeignetsten Mitarbeitern bezüglich der Projektanforderungen. Wenn ein Mitarbeiter für mehrere Projektanforderungen ausgewählt wurde, würde er theoretisch mehrfach in dem Baum vorkommen. Er wird allerdings nur dann zu einem Team hinzugefügt, wenn er nicht bereits enthalten ist. Diese Vorgehensweise berücksichtigt bereits bei der Teamselektion zwei der fünf abprüfbareren Eigenschaften von Teams nach der Definition von Katzenbach (siehe 2.2): komplementäre Fähigkeiten und eine

geringe Mitgliederanzahl.

Nach der Bildung von Teams folgt deren Evaluation, um dem Teamrecruiter eine Einschätzung der Geeignetheit zu ermöglichen. Der Ablauf der Evaluation wird in Abschnitt 4.4.5 erläutert.

4.4.3. Zweiter Semantischer Ansatz

Für den zweiten Ansatz werden ebenfalls das Projektanforderungsprofil und die Mitarbeiterkompetenzprofile als Eingaben benötigt. Der Algorithmus betrachtet nun für die Berechnung die darin enthaltenen Fähigkeiten mit ihren Gewichten. Im zweiten Ansatz wird damit die Erfahrung bzw. der Grad an Expertise einer Fähigkeit in den Auswahlprozess mit einbezogen. Durch die verwendeten Level wird außerdem die Wichtigkeit bestimmter Projektanforderungen ausgedrückt. Bei unterschiedlichen Leveln sind höher gewichtete Fähigkeiten wichtiger als andere, da sie mit fortgeschrittenener Erfahrung gebraucht werden.

Analog wie bei Biesalski [Bie06] wird in dieser Ausarbeitung die von der Siemens AG benutzte vierstufige Skala für Level verwendet:

- 1 - Student (Anfängerkenntnisse, die meist nicht beruflich erworben wurden)
- 2 - Junior (erweiterte Grundkenntnisse aus dem Berufsleben)
- 3 - Middle (sehr gut ausgeprägte Kenntnisse)
- 4 - Senior (Expertenwissen durch langjährige Anwendung)

Der zweite semantische Algorithmus durchläuft die folgenden Schritte (schematisch dargestellt in Abbildung 4.7):

1. Ähnlichkeit zwischen allen P-Skills und allen E-Skills berechnen.
2. Die berechneten Ähnlichkeiten werden durch das Verhältnis vom leistbaren zum geforderten Level auf- oder abgewertet.
3. Jeweils den höchsten Wert in der Skill-Matrix (E,P) speichern.
4. Mitgliederauswahl.
5. Evaluation.

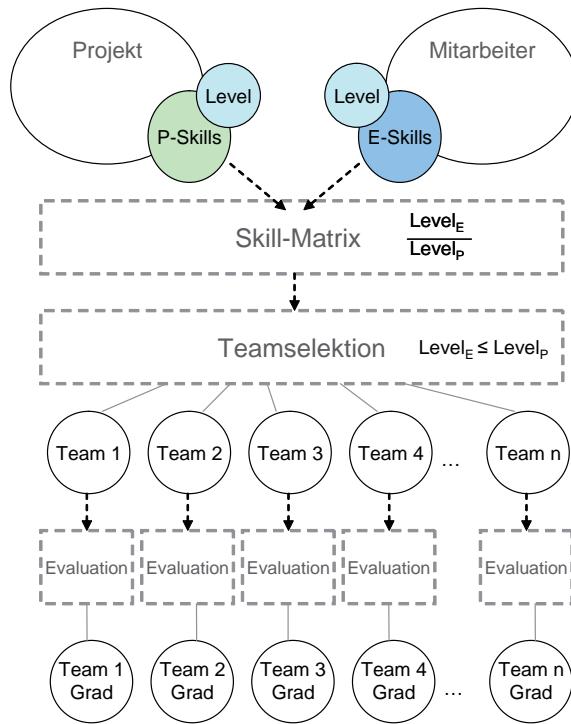


Abbildung 4.7.: Schema des zweiten Algorithmus

Der Unterschied zum ersten Ansatz liegt in der Generierung der Skill-Matrix. Die Berechnung der Ähnlichkeiten wird erneut anhand der HR-Ontologie zwischen allen Projektfähigkeiten und allen Mitarbeiterfähigkeiten vorgenommen. Jede dieser Ähnlichkeiten wird zusätzlich um Level auf- oder abgewertet. Die Gleichung 4.1 skizziert die Berechnung der Matrixeinträge:

$$sim_{2.Anansatz} = \begin{cases} \frac{sim_{1.Anansatz} * Level_M}{Level_P}, & \text{wenn } Level_P \geq Level_M \\ sim_{1.Anansatz} * \left(1 - \frac{Level_M - Level_P}{4}\right), & \text{wenn } Level_P < Level_M \end{cases} \quad (4.1)$$

$Level_P$ bezeichnet den Projektlevel und $Level_M$ den Mitarbeiterlevel. Die eigentliche Ähnlichkeit zwischen den Fähigkeiten, wie sie im ersten Ansatz ($sim_{1.Anansatz}$) berechnet wurde, wird an dieser Stelle durch Unter- oder Überfüllung der notwendigen Fähigkeitslevel abgewertet und durch Erfüllung aufgewertet. Die Ähnlichkeiten werden also bezüglich der leistbaren Level relativiert. Biesalski [Bie06] hat die Vorgehensweise in diesem Fall betrachtet, was in Abschnitt 3.1.5 beschrieben wurde. Je größer die Differenz zwischen gefordertem und leistbarem Level ist, um so höher ist der Grad der Unter- oder Übererfüllung und desto größer werden

Ähnlichkeiten entsprechend verringert. Im Fall der Übererfüllung ergibt $sim_{1,Ansatz}$ einen Wert über 1, der durch $\left(1 - \frac{Level_M - Level_P}{4}\right)$ entsprechend der Differenz der Level einer Fähigkeit verringert. 4 steht in der Gleichung für die vier verschiedenen Level. Eine große Levdendifferenz resultiert dann in einem größeren Wert als eine kleinere Levdendifferenz. Durch diese Art der Berechnung werden weiterhin Werte zwischen 0 und 1 in der Matrix erzeugt.

- Untererfüllung: $Projektlevel > Mitarbeiterlevel$
- Übererfüllung: $Projektlevel < Mitarbeiterlevel$

Eine Übererfüllung einer Projektanforderung tritt beispielsweise auf, wenn eine Fähigkeit mit Level 1 mit einem Mitarbeiter besetzt wird, der diese Fähigkeit mit einem Level 4 ausfüllen kann. Dieser Mitarbeiter kann so seinen eigentlichen Wert nicht im Team einbringen, da seine Kenntnisse nur zu einem Bruchteil genutzt werden. Darüber hinaus verursacht wahrscheinlich ein Mitarbeiter mit höherem Level, unter Umständen sogar mit Expertenwissen, dem Unternehmen höhere Kosten. Deshalb sollte er besser in Projekten eingesetzt werden, in denen seine Erfahrung und Kenntnisse gebraucht werden. Aus diesem Grund wird in der Matrix die Ähnlichkeit im Fall einer Übererfüllung verringert. Dasselbe geschieht im Falle einer Untererfüllung. Ist für ein Projekt z.B. eine Fähigkeit auf Level 3 notwendig und besitzt ein Mitarbeiter diese Fähigkeit nur auf Level 1, wird der Wert seiner Ähnlichkeit ebenfalls entsprechend reduziert.

Jeweils der höchste der errechneten Werte ist erneut für die weitere Betrachtung relevant. Ein Eintrag x in der Skill-Matrix bedeutet in zweiten Ansatz, dass ein Mitarbeiter E (Zeile) mindestens eine gewichtete Fähigkeit besitzt, die mit dem Wert x ähnlich zu einer gewichteten Projektfähigkeit P (Spalte) ist.

Die Abbildung 4.8 zeigt ein Beispiel für die Skill-Matrix des zweiten Algorithmus. Es handelt sich bei dem Unternehmen um das aus Beispiel 4.6. Zur Projektdurchführung werden wiederum die Fähigkeiten A und B benötigt. Der Matrixeintrag 0,4 bedeutet nun, dass die Fähigkeit des Mitarbeiters 1, die am ähnlichsten zu Skill A ist und beim ersten Ansatz mit 80% gewertet wurde, durch Untererfüllung des Levels für Skill A abgewertet wird. Er kann diese Fähigkeit nur mit dem Level 1 ausführen, er besitzt also lediglich Anfängerkenntnisse, obwohl erweiterte Grundkenntnisse gefordert sind. Dieser Umstand resultiert in einem Wert 0,4.

Der Eintrag 1 bei Mitarbeiter 2 bedeutet, dass dieser die geforderte Fähigkeit A be-

Projekt Mitarbeiter	Skill A (2)	Skill B (4)	normalisierte Zeilensumme
1	$(0,8 \cdot 1)/2 = 0,4$	$(0,7 \cdot 4)/4 = 0,7$	$1,1/2 = 0,55$
2	$(1 \cdot 2)/2 = 1$	$(0,2 \cdot 3)/4 = 0,15$	$1,15/2 = 0,58$
3	$(1 \cdot 2)/3 = 0,67$	$(1 \cdot 2)/4 = 0,5$	$1,17/2 = 0,59$
Normalisierte Spaltensumme	$2,07/3 = 0,69$	$1,35/3 = 0,45$	Funktionalwert 0,57

Abbildung 4.8.: Beispiel für die Skill-Matrix des zweiten semantischen Ansatzes

sitzt und sogar mit dem notwendigen Level. Mitarbeiterin 3 besitzt ebenfalls Skill A, allerdings mit Level 3, also mit sehr guten Kenntnissen, obwohl lediglich erweiterte Grundkenntnisse gebraucht werden. Damit ist der Umstand einer Übererfüllung gegeben und der Wert wird, wie in Gleichung 4.1 angegeben, verringert auf 0,67. Der erste Algorithmus hat Mitarbeiter 2 und 3 bezüglich Skill A gleich berechnet, weil er keine Level betrachtet. Durch den zweiten Algorithmus wird ersichtlich, dass Mitarbeiter 2 für das gegebene Projekt geeigneter ist als Mitarbeiterin 3. Bei der Berechnung der normalisierten Zeilen- und Spaltensummen wird ersichtlich, dass Skill A mit einem Wert von 0,69 für die gewählten Mitarbeiter in einem Projekt besser, wenn auch nicht mehr mit dem hohen Wert aus Beispiel 4.6, vertreten ist, als Skill B mit 0,45. Das liegt in der Verwendung der Level begründet. Den größten fachlichen Wert für das Projekt besitzt weiterhin Mitarbeiterin 3, auch wenn ihre Fähigkeiten deutlich im Vergleich zum ersten Algorithmus abgewertet werden mussten. Sie wird deshalb für ein Team ausgewählt, da sie im Vergleich zu den anderen Mitarbeitern trotz übererfüllender Fähigkeiten die insgesamt geeignete Mitarbeiterin für das Projekt ist. Sie würde aus diesem Grund wieder für die Rolle des Teamleiters markiert werden.

Bei der Auswahl von Mitgliedern für Teams werden höherwertige und damit für das Projekt wichtigere Fähigkeiten priorisiert betrachtet. Die Fähigkeiten werden nach ihren Leveln sortiert. Angefangen mit der am höchsten gewichteten Projektanforderung werden mindestens zwei Mitarbeiter ausgewählt, die für diese Anforderung die höchsten Skill-Matrixeinträge besitzen und gleichzeitig die höchsten

fachlichen Werte bieten. Für die absteigenden anderen Level wird ebenso verfahren. Im Beispiel würden also für Skill B Mitarbeiter 1 und Mitarbeiterin 3 gewählt werden und für Skill A Mitarbeiter 2 und erneut Mitarbeiterin 3.

Die Menge der Mitarbeiter, aus denen schließlich Teams selektiert werden sollen, wird, wie bereits beim zweiten semantischen Ansatz beschrieben, pro Spalte der Skill-Matrix und anhand einer Baumstruktur zu Teams von 4 bis 10 Mitgliedern permutiert. Durch die nach Leveln sortierte Mitarbeiterselektion erscheinen die besten Mitarbeiter der wichtigsten Level in Nähe des Wurzelknotens des Baums. Dadurch wird gewährleistet, dass sie die ersten Mitarbeiter sind, die für Teams ausgesucht und nicht aufgrund z.B. zu großer Mitgliederanzahl „aussortiert“ werden. Nach der Bildung der Teams folgt erneut deren Evaluation (siehe Abschnitt 4.4.5).

4.4.4. Dritter Semantischer Ansatz

Dass Erfahrung eine besondere Bedeutung zukommt, ist in Interviews mit Siemens Teamrecruitern festgestellt worden. Außerdem definiert Jackson et al. [JMW95] (siehe Tabelle 4.1) die Erfahrung als eines der aufgabenbezogenen Attribute, die Heterogenität in Teams sicherstellen. In Abschnitt 4.2 wurde der Einfluss von Rollen innerhalb eines Teams erläutert. Bäumgen und Hübber [BH04] schreiben, dass ein Projektleiter bzw. Teamleiter möglichst viel Projekterfahrung haben sollte. Von der Oelsnitz schreibt in [Oel07], dass auch das Vorhandensein von Erfahrung und insbesondere gemeinsamen Erfahrungen zur Bildung von transaktivem Wissen führt. Das bedeutet, dass es dem Team einen Effektivitätsgewinn einbringt, wenn die involvierten Mitglieder Erfahrung im Lösen der im Projekt gestellten Aufgaben haben und wissen, nach welchen Abläufen gehandelt werden soll.

Aus den genannten Gründen werden beim dritten semantischen Ansatz nunmehr aus dem Projektanforderungsprofil und den Mitarbeiterkompetenzprofilen alle enthaltenen Informationen betrachtet, also benötigte gewichtete Fähigkeiten und zu lösende Aufgaben für das Projekt sowie gewichtete Fähigkeiten der Mitarbeiter und Aufgaben, bei denen sie bereits mitgearbeitet haben. Diese Vorgehensweise berücksichtigt nicht nur, wie beim zweiten Ansatz, Erfahrung in einer Fähigkeit sondern auch Erfahrung mit der Lösung von Aufgaben und bestimmten Abläufen bzw. Prozessen und trifft somit die Aussagen der zitierten Autoren. Außerdem

wird in diesem Ansatz durch die Einbeziehung von Erfahrungswissen auch berücksichtigt, dass, wie in 4.2 beschrieben, Wissenslücken behebbar sind. Denn die Teammitglieder müssen sowohl die notwendigen Fähigkeiten besitzen, als auch die zu lösenden Aufgaben kennen. Der Ablauf des dritten Algorithmus gestaltet sich, wie in Abbildung 4.9 dargestellt, folgendermaßen:

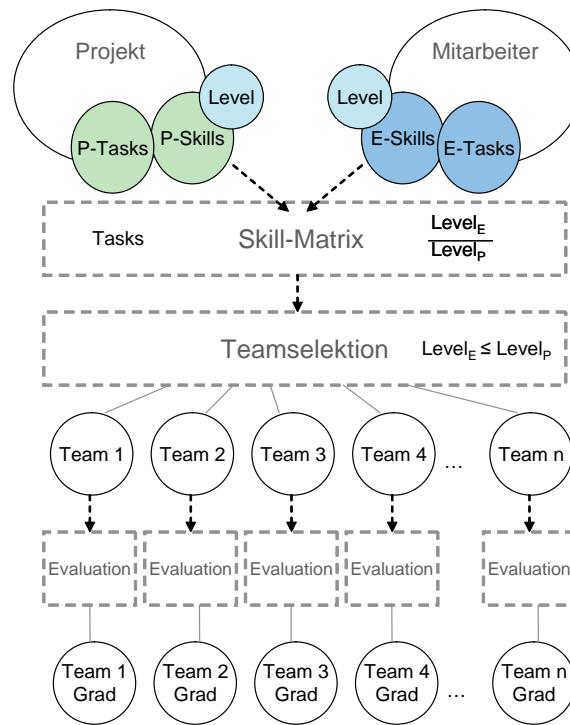


Abbildung 4.9.: Phasen des dritten Ansatzes

1. Ähnlichkeit zwischen allen P-Skills und allen E-Skills berechnen.
2. Diese Werte werden durch das Verhältnis vom verfügbaren Mitarbeiterlevel zum geforderten Projektlevel auf- oder abgewertet.
3. Ähnlichkeiten zwischen allen P-Tasks und allen E-Tasks berechnen.
4. Die jeweils höchsten errechneten Werte von Fähigkeiten und Aufgaben zu einem Mittelwert aggregieren.
5. Werte in der Skill-Matrix (E,P) speichern.
6. Mitgliederauswahl.
7. Evaluation.

Der Unterschied liegt, wie schon bei den anderen semantischen Ansätzen, in der Generierung der Skill-Matrix. Die Berechnung der Ähnlichkeit wird wieder anhand der HR-Ontologie zwischen allen Projektfähigkeiten und allen Mitarbeiterfähigkeiten vorgenommen. Jede dieser Ähnlichkeiten wird wie beim zweiten Ansatz durch Betrachtung der Level auf- oder abgewertet. Analog dazu werden nun basierend auf der HR-Ontologie die Ähnlichkeiten zwischen allen Projektaufgaben und allen Mitarbeiteraufgaben berechnet. Jeweils der höchste dieser beiden Gesamtwerte ist erneut für die weitere Betrachtung relevant. Für jeden Mitarbeiter wird in der Skill-Matrix der Mittelwert aus der höchsten Fähigkeitsähnlichkeit und höchsten Aufgabenähnlichkeit für jede Projektanforderung eingetragen. Ein Eintrag x in der Skill-Matrix bedeutet somit, dass ein Mitarbeiter E (Zeile) mindestens eine gewichtete Fähigkeit sowie eine Aufgabe besitzt, die zusammen genommen die Durchschnittsähnlichkeit x zu einer gewichteten Projektfähigkeit P (Spalte) sowie einer Projektaufgabe bildet.

Die Abbildung 4.10 zeigt ein Beispiel für die Skill-Matrix des dritten Algorithmus.

Mitarbeiter \ Projekt	Skill A (2)	Skill B (4)	normalisierte Zeilensumme		Projekt \ Mitarbeiter	Task I	Task II
1	$(0,8*1)/2$ $(0,4+1)/2 = 0,7$	$(0,7*4)/4$ $(0,7+1)/2 = 0,85$	$1,55/2 = 0,78$		1	1	0,5
2	$(1*2)/2$ $(1+0,7) = 0,85$	$(0,2*3)/4$ $(0,15+0,7)/2 = 0,43$	$1,28/2 = 0,64$		2	0,6	0,7
3	$(1*2)/3$ $(0,67+0,3)/2 = 0,49$	$(1*2)/4$ $(0,5+0,3)/2 = 0,4$	$0,89/2 = 0,45$		3	0,3	0,3
Normalisierte Spaltensumme	$2,04/3 = 0,68$	$1,68/3 = 0,56$	Funktionalwert 0,62				

Abbildung 4.10.: Beispiel für die Skill-Matrix und Aufgaben des dritten semantischen Ansatzes

Es handelt sich bei dem Unternehmen erneut um das aus Beispiel 4.6. Zur Projektdurchführung werden in diesem Fall die Fähigkeiten A und B für die Lösung der Aufgaben I und II benötigt. Der Matrixeintrag 0,7 des Mitarbeiters 1 ist nun der Mittelwert aus der zu Skill A ähnlichen und um den Level 1 abgewerteten Fähigkeit sowie der höchsten Ähnlichkeit 1 zwischen der Projektaufgabe I und einer Mit-

arbeiteraufgabe. Der Durchschnitt aus höchster Fähigkeitsähnlichkeit und höchster Aufgabenähnlichkeit bildet den Eintrag jeder Matrixzelle. Für die Berechnung der Werte für Mitarbeiter 2 wird jeweils 0,7 als höchste Aufgabenähnlichkeit benutzt. Interessant ist die Entwicklung der Werte von Mitarbeiterin 3: die anfänglichen 1-Einträge wurden sowohl durch über- und untererfüllte Level abgewertet und schließlich durch die geringe Erfahrung in den gestellten Projektaufgaben weiter vermindert. Ihre Zeilenwerte werden trotzdem auf geeignete Weise herabgestuft, so dass sie sich nicht völlig „unbrauchbar“ für das Projekt darstellt. Im Vergleich mit den anderen Mitarbeitern erreicht sie allerdings nicht deren Grad der Geeignetheit. Aufgrund des höchsten fachlichen Wertes wird der Algorithmus Mitarbeiter 1 als Teamleiter vorschlagen. Er bietet durchschnittlich gesehen die meiste Erfahrung und vor allem durch Erfüllung der höchst gewichteten Fähigkeit die meiste Kompetenz.

Bei der Auswahl von Mitgliedern für Teams wird wie beim zweiten Algorithmus verfahren. Nach Leveln sortiert werden mindestens zwei der besten Mitarbeiter pro Matrixspalte ausgewählt. Für das letzte Beispiel würden demnach für Skill B Mitarbeiter 1 und 2 gewählt werden und für Skill A Mitarbeiter 2 und 1. Die zu Teams permutierten Mitarbeiter werden schließlich mit einem Geeignetheitsgrad evaluiert.

4.4.5. Teamevaluation

Das Konzept der Evaluation zielt darauf ab, unabhängig von einer vorangegangenen Mitgliederselektion, eine Bewertung anhand der Kriterien optimaler Teams vorzunehmen. Durch diese Vorgehensweise wird ein Vergleich verschiedener Auswahltechniken möglich. Dazu gehören auch jene, die nicht im Rahmen der vorliegenden Arbeit entstanden sind. In Abschnitt 4.2 werden die Kriterien optimaler Teams dargelegt, wobei in den meisten Fällen bei den genannten Autoren keine Angabe konkreter Quantitäten zu finden war. Gemeint sind Aussagen wie z.B. „transaktives Wissen einer Fähigkeit muss mindestens bei zwei Teammitgliedern vorhanden sein“. Zur Schließung dieser Lücke und zur Konzipierung der Berechnung der Kriterien sollen an dieser Stelle die notwendigen Quantitäten definiert werden.

Ähnlich einer Bewertung der Leistung bei Sportarten, wurde pro Kriterium immer

von einem Idealfall ausgegangen. Bei Nichterfüllung von Details wurden von diesem Idealfall anteilig „Punkte“ bzw. Werte abgezogen. Zum Verständnis sei noch ein Wert definiert, der häufig verwendet wird. Der Funktionalwert eines Teams ist der Mittelwert aller durchschnittlichen fachlichen Werte jedes Teammitglieds, also der Mittelwert der normalisierten Zeilensummen. Damit beschreibt der Funktionalwert die fachliche Güte eines Teams und wird von zum Zweck der Berechnung einiger Kriterien betrachtet

Teamgröße

Bei der Teamgröße waren die Autoren sehr konkret: ein optimales Team besteht aus 4 bis 10 Mitgliedern. Teams mit Größen außerhalb dieses Bereiches werden deshalb nicht an den Teamrecruiter weitergeleitet. Bereits bei der Auswahl von Teammitgliedern achten die Algorithmen darauf, dieses Kriterium zu erfüllen. Wie in Teil 1.2 genannt, sollte ein optimales Team möglichst geringe personelle Ressourcen brauchen. Bei der Evaluation wird berücksichtigt, dass im direkten Vergleich zweier Teams mit demselben Funktionalwert das Team gewählt werden soll, das kleiner ist. Zur Erläuterung sei an dieser Stelle verwiesen auf den Abschnitt 4.2, in dem Kompetenzredundanz als Folge von deckungsgleichen Mitarbeitern beschrieben wird. Das bedeutet, dass die Aufgabe auch durch weniger Mitarbeiter realisierbar wäre. Beispielsweise muss ein Team mit vier Mitgliedern, dass denselben Funktionalwert besitzt wie ein Team mit sieben Mitgliedern, besser bewertet werden, da weniger Mitarbeiter notwendig sind, um die benötigten fachlichen Qualifikationen aufzubringen. Dieser Umstand wird durch Gleichung 4.2 umgesetzt.

$$Evaluation_{Teamgroesse} = \frac{Funktionalwert + \frac{4}{Anzahl_Teammitglieder}}{2} \quad (4.2)$$

Der Idealfall für ein Team bezüglich seiner Größe liegt vor, wenn vier Mitglieder einen Funktionalwert von 1 erreichen. Der Evaluationswert der Teamgröße ist aus diesem Grund der Durchschnitt aus Funktionalwert des Teams und der zu vier Mitgliedern relativierten Teamgröße. Teams, die aus vier Mitarbeitern bestehen, erhalten für den oberen Bruch mit einer 1 den höchsten Wert. Je mehr Mitglieder ein Team braucht, desto kleiner wird der Bruch. Ein Team kann bezüglich seiner Größe nur dann gut bewertet werden, wenn sowohl Funktionalwert als auch die Anzahl stimmen.

Heterogenität

In Teil 4.2 wird beschrieben, dass ein optimales Team möglichst heterogen aufgebaut sein sollte, und welche Faktoren die Homogenität und Heterogenität beeinflussen. Für die in Tabelle 4.1 genannten beziehungsorientierten Attribute fällt es schwer, eine objektive Erfassung zu gewährleisten, außerdem sind sie nicht Teil der Zielsetzung. Heterogenität kann in der vorliegenden Arbeit ausschließlich aufgrund der verarbeiteten Daten auf Wissen und Expertise sowie Erfahrung basieren. Deshalb soll ein Team dann als heterogen definiert werden, wenn es keine deckungsgleichen Mitglieder bezüglich ihrer Fähigkeiten gibt. Eine von zwei gleichen Personen bringt dem Team keinen Kenntnisgewinn, weshalb nur eine dieser Person gewertet werden sollte. Dies wird in Formel 4.3 berücksichtigt.

$$Evaluation_{Heterogenitaet} = 1 - \frac{1}{Anzahl_Teammitglieder} * Anzahl_gleicher_Paare \quad (4.3)$$

Je zwei gleiche Teammitglieder bilden ein Paar. Da nur eines der Teammitglieder im Team gebraucht wird, wird der Teamwert sozusagen um eine Person reduziert. Für den Fall, dass alle Mitglieder dieselben Fähigkeiten besitzen, wird die Anzahl gleicher Paare Anzahl der Teammitglieder - 1.

Überlappung

Teamaufgabe kann nicht funktionieren, wenn es kein transaktives Wissen im Team gibt (siehe 4.2). Wenn es keine überlappenden Fähigkeiten gibt, entstehen keine Diskussionen und in der Folge auch weniger gute Lösungen. Deshalb sollte es pro Fähigkeit wenigstens zwei Personen geben, die diese mit einem hohen Grad bedienen. Formel 4.4 geht bei der Berechnung des Kriteriums wiederum von einem Idealfall aus: pro Projektanforderung sollte es mindestens zwei Personen geben, die die jeweilige Fähigkeit zu 100% erfüllen.

$$Evaluation_{Ueberlappung} = \sum_{i=1}^n \left(\frac{\Sigma x + y}{2} * \frac{1}{n} \right) \quad (4.4)$$

In der Formel ist n die Anzahl der im Projekt benötigten Fähigkeiten, x und y bezeichnen die beiden höchsten Expertisengrade pro Spalte der Skill-Matrix. Im Bestfall wird $\frac{\Sigma x + y}{2}$ pro Spalte jeweils 1, wodurch auf jede der Fähigkeiten der Anteil

$\frac{1}{Anzahl_Fähigkeiten}$ entfällt, was in der Summe 1 bedeutet. Jede Fähigkeit, für die diese Eigenschaft nicht zutrifft, wertet den Gesamtwert für Überlappung entsprechend ab.

Wissenslücken

Wie beschrieben, sollten Wissenslücken verständlicherweise in einem Team vermieden werden. Das Vorhandensein einer Wissenslücke muss aber nicht zum Abbruch oder Misserfolg führen, wenn die Mitglieder Willens sind, die Lücke durch Lernen zu schließen. Für Stader, wie in Abschnitt 4.2 gezeigt, gibt es eine Wissenslücke hauptsächlich dann, wenn eine Fähigkeit fehlt oder nicht ausreichend vorhanden ist. Eine genauere Definition, wie z.B. eine Fähigkeit, die höchstens zu 30% erfüllt ist, markiert eine Wissenslücke, konnte leider nicht gefunden werden. Ziel der Einzelwerte bei der Evaluation ist es, einen Wert zwischen 0 und 1 zu erhalten. Für das Kriterium der Wissenslücke würde eine 0 das Fehlen einer Projektanforderung und 1 eine hundertprozentige Erfüllung bedeuten. Aus diesem Grund soll der Wert der am geringsten vertretenen Fähigkeit, also der kleinsten normalisierten Spaltensumme, zurückgegeben werden (siehe Formel 4.5).

$$Evaluation_{Wissensluecken} = \text{kleinste_normalisierte_Spaltensumme} \quad (4.5)$$

Ist die kleinste normalisierte Spaltensumme groß (annähernd 1), heißt das, dass im Prinzip keine Wissenslücke vorhanden ist. Ein sehr kleiner Wert geht dann in die Gesamtevaluation als Wissenslücke ein.

Rollen

In Teil 4.2 werden Teamrollen beschrieben und dass diese auf Persönlichkeitsprofilen basieren. Dadurch rücken Rollen allgemein aus dem Fokus dieser Arbeit. Die markanteste Rolle des Teamleiters soll allerdings durch das Mitglied mit der höchsten normalisierten Zeilensumme gekennzeichnet werden. In der Gleichung 4.6 wird dies gezeigt.

$$Evaluation_{Teamleiter} = \text{hoechste_normalisierte_Zeilensumme} \quad (4.6)$$

Somit geht der Wert des Teamleiters in den Gesamtwert des Teams ein. Je höher sein Wert und demnach seine Geeignetheit, desto besser kann ein Team evaluiert werden.

Verfügbarkeit

Dass die Verfügbarkeit aller Mitglieder über den gesamten Projektzeitraum als sehr wichtig für ein Team betrachtet wird, wurde in Abschnitt 4.2 erörtert. Die entwickelten Teamplanungsalgorithmen beachten dieses Kriterium bereits bei der Mitgliederselektion (siehe 4.3). Da die Evaluation auf jegliche Teamzusammensetzung unabhängig von den entwickelten Algorithmen angewendet werden soll, wird die Verfügbarkeit dennoch überprüft. Jeder nicht verfügbare Mitarbeiter schwächt das Team. Dieser Umstand kann dadurch ausgedrückt werden, in dem der Funktionalwert des Teams um den Wert des ausfallenden Teammitglieds verringert wird. Gleichung 4.7 zeigt diese Berechnung.

$$Evaluation_{\text{Verfügbarkeit}} = \text{Funktionalwert} - \sum_{i=1}^n \text{normalisierte_Zeilensumme}_{E_n} \quad (4.7)$$

In der Formel bezeichnet n die Anzahl der nicht verfügbaren Teammitglieder. Sollten alle Teammitglieder nicht komplett im Projekt arbeiten können, ist der in die Gesamtevaluation eingehende Wert 0.

Geeignetheitsgrad

Die einzelnen während der Evaluation errechneten Werte müssen im letzten Schritt zu einem Geeignetheitsgrad aggregiert werden.

Allgemein ist der Geeignetheitsgrad der durchschnittliche Wert aller Evaluationswerte. Wenn man allerdings Kriterien auf ihre Bedeutung für das Team vergleicht, wird ersichtlich, dass sie nicht alle in gleichem Maße Einfluss auf dem Verlauf des Projektes nehmen. Als Beispiel sei an dieser Stelle der Vergleich zwischen Teamgröße und Wissenslücken genannt. Während Wissenslücken wie in Abschnitt 4.2 beschrieben behebbar sind, zerfallen zu große Teams in Subteams oder zu kleine Teams können keine Synergien nutzen. Aus diesem Grund werden die einzelnen Kriterien priorisiert betrachtet:

1. Überlappungen und transaktives Wissen
2. heterogene Zusammensetzung
3. andauernde Verfügbarkeit
4. Teamgröße
5. geeigneter Teamleiter
6. Wissenslücken

Da Überlappungen und transaktives Wissen durch die Entstehung von Diskussionen direkt die Teamarbeit beeinflussen, wird dieses Kriterium am wichtigsten angesehen. Eine heterogene Zusammensetzung sichert eine produktive Arbeitsweise und wird deshalb direkt hinter der Überlappung eingeordnet. Mit diesen beiden Kriterien wird eine produktive Teamarbeit gewährleistet. Damit dieser Zustand über die Dauer des Projektes anhalten kann, müssen alle Mitglieder verfügbar bleiben. Es folgt das Kriterium der Teamgröße, wobei insgesamt sieben mögliche Größen als geeignet erachtet werden und somit ein etwas vergrößerter Spielraum geboten wird. Ein geeigneter, kompetenter Teamleiter hält das Team dynamisch und bildet die Verbindung zur Außenwelt (siehe Teil 4.2). Wissenslücken schließlich sind am einfachsten zu kompensieren und stehen deshalb in der Ordnung an unterster Stelle.

Damit ergibt sich folgende Gleichung zur Ermittlung des Geeignetheitsgrades:

$$\text{Geeignetheitsgrad} := \frac{6 * \text{Evaluation}_{\text{Ueberlappung}} + 5 * \text{Evaluation}_{\text{Heterogenitaet}} + 4 * \text{Evaluation}_{\text{Verfuegbarkeit}} + 3 * \text{Evaluation}_{\text{Teamgroesse}} + 2 * \text{Evaluation}_{\text{Teamleiter}} + 1 * \text{Evaluation}_{\text{Wissensluecken}}}{6 + 5 + 4 + 3 + 2 + 1} \quad (4.8)$$

Der Geeignetheitsgrad ist somit der Mittelwert von allen nach Priorität gewichteten Evaluationswerten.

5. Implementierung

Nach der Erstellung der konzeptionellen Lösung wird in diesem Kapitel die eigentliche Realisierung der zuvor beschriebenen Anwendung vorgestellt. Die Beschreibung der Implementierung erfolgt nach einzelnen Schwerpunkten getrennt und umfasst Vorgehensweise, Methoden und Ergebnisse.

Die Umsetzung in Java erfolgte anhand der Sun Code Konvention für Java [Sun99]. Die entstandene Anwendung ist kompatibel zu Java in Version 6. Bezeichner und Kommentare sind in englischer Sprache verfasst.

5.1. Technische Realisierung

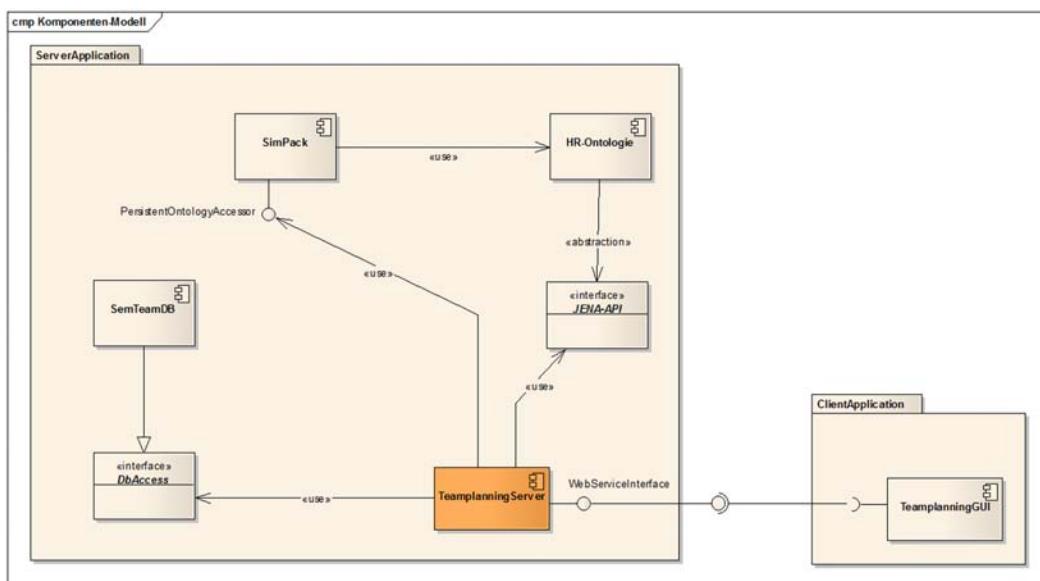


Abbildung 5.1.: Komponentenmodell der Anwendung

Abbildung 5.1 zeigt das UML-Komponentenmodell, das den Rahmen für die Entwicklung und Ausführung der Anwendungskomponenten festlegt und zeigt, wie

diese voneinander abhängen. Die Anwendung teilt sich grob in eine Server- und eine Client-Anwendung. Zur Kommunikation der beiden Packages werden die in Java 6 verfügbaren Techniken zur Definition von Webservices verwendet. Über das `WebServiceInterface` veröffentlicht der TeamplanningServer einen Webservice, der von der `ClientApplication` konsumiert wird. Durch den Informationsaustausch zwischen Client und Server über einen Webservice kann der komplexere TeamplanningServer-Prozess auf entsprechend ausgelegter Hardware installiert und verfügbar gemacht werden. Die leichtgewichtige TeamplanningGUI hingegen kann auf dem Arbeitsplatzrechner des Teamrecruiters verwendet werden. Zusätzlich ist die Portabilität und eine mögliche Heterogenität der Systemplattformen (Linux, Windows oder sonstige) gewährleistet. Im folgenden Abschnitt werden die einzelnen Komponenten näher erläutert.

5.1.1. ServerApplication

Die `ServerApplication` enthält alle Komponenten, die zur Ausführung der in Abschnitt 3.3 vorgestellten Anwendungsfälle gebraucht werden. Der Server beinhaltet zu diesem Zweck sowohl die HR-Ontologie, als auch die Datenbank der Mitarbeiter. Außerdem werden hier die Ähnlichkeitsberechnungen zur Erstellung der Skill-Matrizen vorgenommen.

SemTeamDB

Bei dieser Komponente handelt es sich um eine relationale Datenbank. In `SemTeamDB` werden berechnungsrelevante Daten der Mitarbeiter gespeichert. Der Entwurf der Datenbank wird in Abbildung 5.2 dargestellt. Für einen `Employee` werden persönliche Daten erfasst wie Name, Geschlecht, Alter und mit den Attributen `BusyFrom` und `BusyTo` die Dauer seiner aktuellen Beschäftigung, was die Einschränkung seiner Verfügbarkeit für eventuelle Projekte bedeutet. Die Verbindung `pass` drückt aus, bei welchen Aufgaben ein Mitarbeiter bereits involviert war. Die parametrisierte Relation `have` ordnet über die Angabe von `Leveln` einem Mitarbeiter eine Menge von Fähigkeiten zu und mittels `participate` kann die Tätigkeit eines Mitarbeiters in einem Team gespeichert werden.

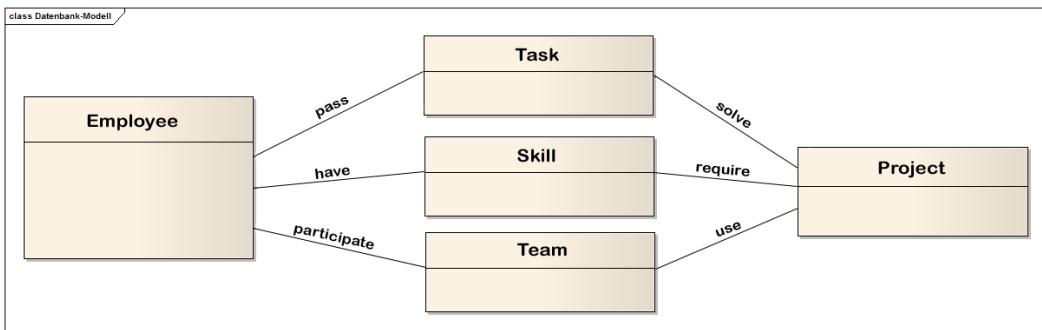


Abbildung 5.2.: Datenbankmodell SemTeamDB

Ein Project wird analog zu einem Mitarbeiter mit beschreibenden Attributen gespeichert. Das sind ein eindeutiger Bezeichner, ein Titel sowie Beginn- und Enddatum. Über die Relation solve werden einem Projekt die Aufgaben zugeordnet, die während der Ausführung gelöst werden sollen. Require ist die Verbindung zwischen einem Projekt und seinen Anforderungen. Schließlich wird über use ein Team mit einem Geeignetheitsgrad zu einem Projekt zugeordnet.

Einträge in der Tabelle der Aufgaben (Task) werden beschrieben durch eine eindeutige Nummer, einen Aufgabennamen, der so auch in der Ontologie zu finden ist und ein optionales Werkzeug Tool, unter dessen Nutzung die Aufgabe erledigt wird. Eine Fähigkeit (Skill) besitzt ebenfalls eine eindeutige Id, einen Namen, eine bestimmte Methode und ein gegebenenfalls für diese Fähigkeit relevantes Werkzeug. Außerdem wird eine Fähigkeit mit einem Typ beschrieben, der aus den Fähigkeitenklassen der HR-Ontologie stammt (siehe Abschnitt 4.1). Ein Team kann zum Zweck der Historisierung bei Bedarf in der Datenbank gespeichert werden und beinhaltet eine eindeutige Nummer sowie implizit zu ihm in Relation gesetzte Mitarbeiter und ein Projekt.

DBAccess

Die Schnittstelle DBAccess erlaubt den gekapselten Zugriff auf das Datenbanksystem. Durch ihre Verwendung wird deklariert, welche Zugriffsmethoden existieren und wie diese genutzt werden können. Dadurch wird der Austausch der verwendeten Datenbank vereinfacht. Durch Erfüllung der Schnittstellenspezifikation kann die Anwendung mit geringem Aufwand an die im Unternehmen vorhandene Infrastruktur angepasst werden.

SimPack

Die Entwicklung einer geeigneten Ähnlichkeitsfunktion liegt außerhalb des Fokus dieser Ausarbeitung. Aus diesem Grund wurde für die Berechnung der Ähnlichkeiten SimPack benutzt, eine „generische Java Bibliothek für Ähnlichkeitsmaße in Ontologien“ [BKKB05], die an der Universität Zürich entwickelt wurde. SimPack ist für den freien Gebrauch unter der LGPL Lizenz verfügbar. Die Bibliothek bietet diverse Ähnlichkeitsmaße, welche je nach Datenstruktur über generische Accessoren benutzt werden.

Für die Implementierung sind zwei der verfügbaren Maße ausgewählt worden: die Methode des Kantenzählens nach Resnik [Res99] und die konzeptuelle Ähnlichkeit nach Wu, Palmer [WP94].

Die Formel zur Ermittlung der Ähnlichkeit zwischen zwei Konzepten in einer Ontologie nach Resnik wird in 5.1 dargestellt.

$$sim_{Resnik}(Node_1, Node_2) = \frac{2 * MAX - len(Node_1, Node_2)}{2 * MAX} \quad (5.1)$$

Der Abstand zweier Knoten wird dabei über die Anzahl der Kanten, die zwischen

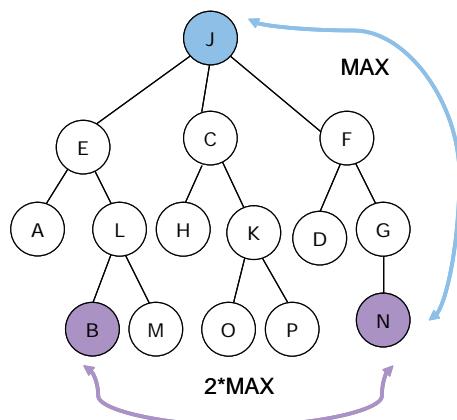


Abbildung 5.3.: Kantenzählmethode nach Resnik

ihnen liegen, berechnet. MAX ist die Länge des längsten Pfades in der Ontologie vom Wurzelknoten zu einem Blatt. Die doppelte Länge des längsten Pfades entspricht damit dem Abstand der zwei am weitesten entfernten Knoten in der Ontologie. $len(Node_1, Node_2)$ bezeichnet die Länge des kürzesten Weges, um von Knoten $Node_1$ zu Knoten $Node_2$ zu gelangen. Damit ist die Ähnlichkeit der beiden Knoten der Anteil des Weges zwischen $Node_1$ und $Node_2$ am größtmöglichen Ab-

stand zweier Knoten. Beispiel: Der längste Pfad zwischen Wurzel und einem Blatt verläuft über 100 Kanten und der kürzeste Weg zwischen zwei gesuchten Knoten beträgt 4, dann besitzen diese beiden eine Ähnlichkeit von 98%. Das Verhältnis des Weges zwischen den beiden Knoten zum längsten Pfad in der Gesamtontologie ist sehr gering, was zu einer hohen Ähnlichkeit führt.

Gleichung 5.2 skizziert die Berechnung der konzeptuellen Ähnlichkeit nach Wu und Palmer.

$$sim_{Wu\&Palmer}(Node_1, Node_2) = \frac{2 * N_3}{N_1 + N_2 + 2 * N_3} \quad (5.2)$$

Die Berechnung bezieht den am nächsten gelegenen gemeinsamen Elternknoten MRCA (most recent common ancestor) mit ein. Dabei bezeichnet N_3 den Abstand vom MRCA zum Wurzelknoten der Ontologie. N_1 ist die Pfadlänge von $Node_1$ zu MRCA, und analog dazu bezeichnet N_2 den Abstand zwischen $Node_2$ und MRCA. Abbildung 5.4 zeigt ein Beispiel. Der MRCA zwischen den Knoten H und P ist

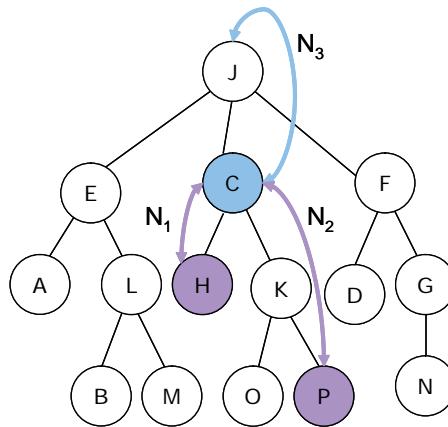


Abbildung 5.4.: Konzeptuelle Ähnlichkeit nach Wu und Palmer

Knoten C. Die Pfadlänge zwischen H und P ist die Summe der „aufsteigenden“ Weglänge von H zu Knoten C und der „absteigenden“ Weglänge von C zu Knoten P. Die Ähnlichkeit der beiden ist analog zu Resnik der Anteil ihrer Entfernung zur Position in der Ontologie, also zum Abstand vom Wurzelknoten. Dies führt dazu, dass zwei Konzepte um so ähnlicher sind, je weiter unten sich in der Hierarchie ihr MRCA befindet. Liegt er jedoch in Wurzelnähe, bedeutet das, dass sich die beiden Konzepte in zwei relativ weit voneinander entfernt liegenden Teilbäumen der Ontologie befinden und deshalb nicht so ähnlich sind. Für die Knoten H und P aus dem Beispiel würde eine Ähnlichkeit von 20% errechnet werden. Würde

ihr MRCA zur Wurzel statt einer Entfernung von einer Kante 10 Kanten bei sonst unveränderten Entfernungen haben, wären H und P zu 83% ähnlich.

Die Berechnung der Matrizen wird in der Implementierung standardmäßig mit der konzeptuellen Ähnlichkeit nach Wu und Palmer vorgenommen. Die beiden vorgestellten Maße können allerdings je nach Belieben ausgetauscht werden.

Zur Nutzung der Ähnlichkeitsmaße konnte keiner der vorhandenen SimPack Accessoren genutzt werden, da die Anwendung die persistierte Ontologie nutzt. Aus diesem Grund wurde für die Implementierung ein `PersistentOntology-Accessor` entwickelt, der dieselben Funktionalitäten wie vorhandene Accessoren bietet, aber auf persistierte Ontologien angewendet werden kann.

HR-Ontologie

Die HR-Ontologie wurde zur effizienten Nutzung in einer Datenbank persistiert, so dass Ontologie-Zugriffe performant durchgeführt werden können. Der entwickelte Accessor initialisiert bei seinem ersten Aufruf notwendige Variablen für die Ähnlichkeitsberechnungen und serialisiert diese für einen schnelleren Zugriff. Beispielsweise erzeugt er eine Matrix der Dimension $|HR - \text{Ontologie}| \times |HR - \text{Ontologie}|$, in dem die kleinsten Pfadlängen der Knoten zueinander gespeichert werden. Dadurch werden aufwendige Berechnungen einmalig durchgeführt und für folgende Ähnlichkeitsberechnungen lediglich abgefragt, was direkt in einem Performanzgewinn resultiert.

Die Implementierung verwendet für die Skill-Matrizen, die mit Ähnlichkeitswerten gefüllt werden, die `COLT-API`, die am CERN, der Europäischen Organisation für Kernforschung, entwickelt wurde [Hos04]. Diese Schnittstelle stellt frei verfügbare Bibliotheken für „High Performance Scientific and Technical Computing“ in Java zur Verfügung. Die Skill-Matrix, die in den semantischen Algorithmen zum Einsatz kommt, basiert auf einer `SparseDoubleMatrix2D`, einer zweidimensionalen Matrix von Dezimalzahlen, die besonders effiziente Methoden zum Abspeichern und Auslesen von Einträgen bereitstellen. Die gefüllten Matrizen werden aus Performanzgründen ebenfalls für spätere Zugriffe serialisiert.

JENA API

JENA ist ein Java Rahmenwerk für die Entwicklung von Semantic Web Anwendungen [DS08]. Es ist frei verfügbar und wird ständig durch die Hewlett Packard Development Company weiter entwickelt. Für die Implementierung wurde die aktuelle Version 2.5.7 benutzt.

Die JENA API (Application Programming Interface) wird als Programmierschnittstelle zur Verarbeitung und Speicherung der semantischen Daten genutzt. Auf diese Art werden alle Zugriffe auf die Ontologie verwaltet. Die Persistierung der Ontologie erfolgt ebenfalls mithilfe von JENA.

TeamplanningServer

Die Komponente des TeamplanningServers bildet die zentrale Steuereinheit der Anwendung. Nähere Einzelheiten werden in Abschnitt 5.1.3 erläutert.

5.1.2. ClientApplication

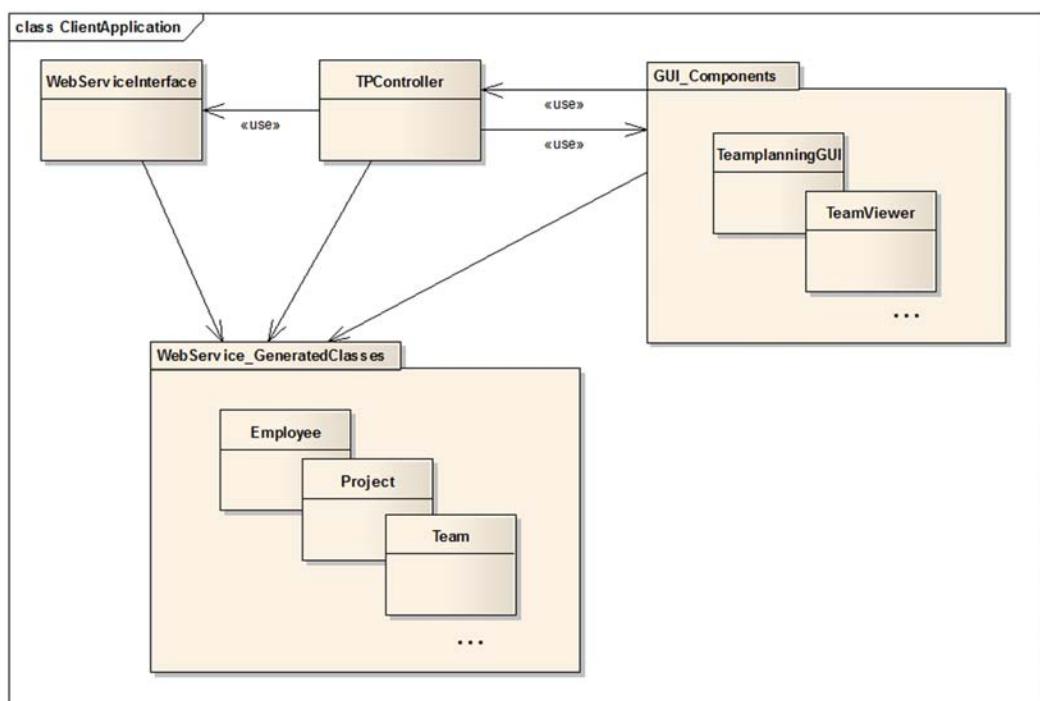


Abbildung 5.5.: Klassendiagramm der ClientApplication

Die ClientApplication ist der Teil der Anwendung, der vom Teamrecruiter verwendet wird. In Abbildung 5.5 wird das UML-Klassendiagramm dargestellt. Die Nutzung der Java Tools für Webservices JAX-WS generiert zu dem von der ServerApplication angebotenen Webservice passende Klassen, die eine objektorientierte Verarbeitung der bereitgestellten Ergebnisse ermöglichen. Die Klasse WebServiceInterface implementiert die vorgesehenen Zugriffsmethoden des Clients auf den TeamplanningServer. Der TPController dient als zentrale Komponente der Steuerung des Informationsflusses zwischen den anzeigenenden GUI-Komponenten und der Kommunikationsschnittstelle WebServiceInterface. Der Benutzer kann mithilfe der GUI-Komponenten über grafische Benutze-

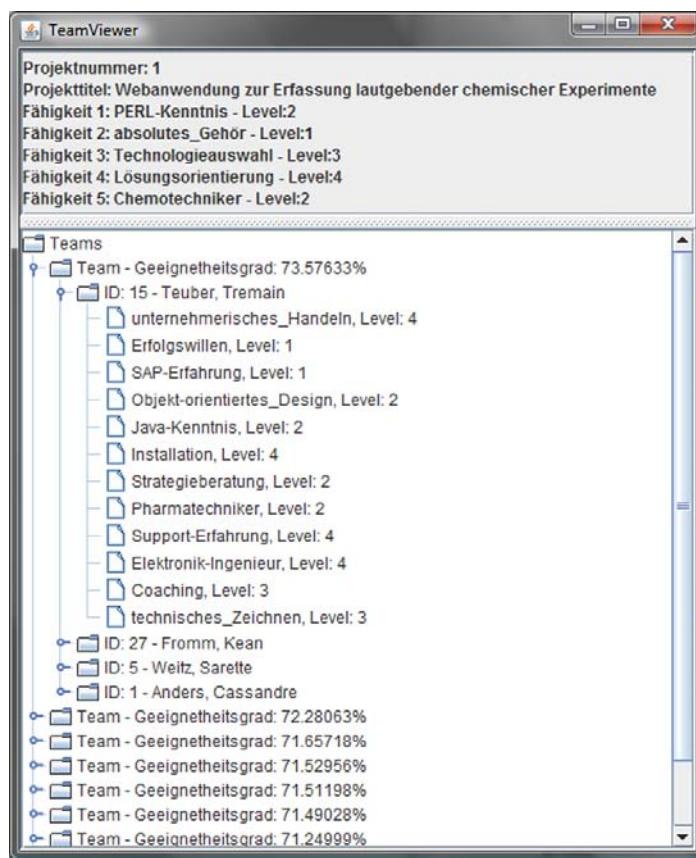


Abbildung 5.6.: Anzeige selektierter Projektteams im TeamViewer

oberflächen Projekte verwalten sowie die gewünschten Anwendungsfälle durchführen. Ein Beispiel für die Anzeige von erzeugten Teams für ein Projekt im TeamViewer wird in Abbildung 5.6 dargestellt. Für die Anzeige wird eine Baumstruktur verwendet, in der die Teams übersichtlich und hierarchisch gruppiert angeordnet sind.

5.1.3. TeamplanningServer

Der TeamplanningServer ist die zentrale Komponente der ServerApplication. Die Interaktionen mit dem TeamplanningServer laufen folgendermaßen ab:

- Über das WebServiceInterface bietet der Server einen Webservice an.
- Ein Teamrecruiter nutzt die Funktionalität z.B. für die Generierung von Teams, in dem er die TeamplanningGUI bedient, die den Webservice konsumiert.
- Nach der Übermittlung einer Projektbeschreibung holt der TeamplanningServer zur Berechnung im ersten Schritt die verfügbaren Mitarbeiter aus der Datenbank SemTeamDB, indem er die Schnittstelle DBAccess nutzt.
- Im nächsten Schritt wird der Algorithmus mit der Berechnung der Skill-Matrix beauftragt, die die verfügbaren Mitarbeiter als Zeilen und die Projektforderungen als Spalten enthält.
- Die Einträge der Matrix sind Ergebnisse von Ähnlichkeitsberechnungen zwischen den Mitarbeiter- und Projektprofilen. Die Berechnung dieser Werte erfolgt durch Ähnlichkeitsfunktionen aus SimPack. Diese verwenden für den Zugriff auf die HR-Ontologie die JENA-API.
- Der TeamplanningServer startet nach der Generierung der Skill-Matrix die Mitarbeiterselektion.
- Anschließend veranlasst er die Evaluation der erstellten Teams und leitet dieses Ergebnis über das WebServiceInterface an die ClientApplication weiter.

Abbildung 5.7 zeigt den für die implementierten Algorithmen relevanten Ausschnitt des UML-Klassendiagramms, dessen Inhalte an dieser Stelle erläutert werden.

Teamplanungsalgorithmen Die Algorithmen sowie die Evaluation erben Methoden und Variablen der Klasse TeamSelectionBase. Sie implementiert die

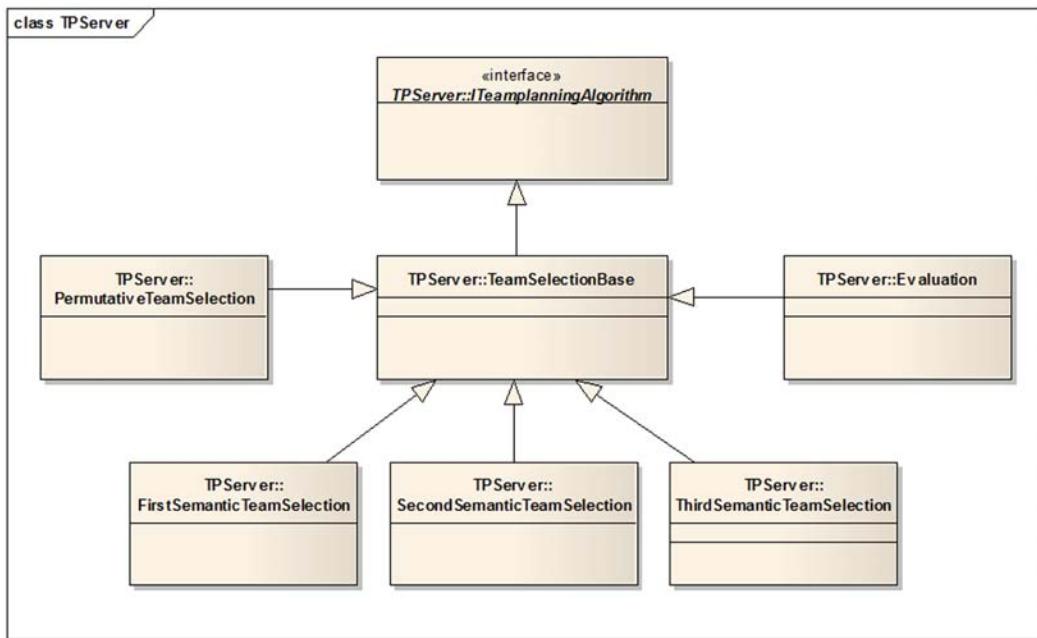


Abbildung 5.7.: Klassendiagramm der Algorithmen und Evaluation

Schnittstelle `ITeamPlanningAlgorithm`, die Methoden für den gekapselten Zugriff auf die Algorithmen erlaubt. Die beiden für den Prozess der Teamplanung wichtigsten sind:

- `List<Team> activateTeamSelection():` gibt eine Liste von generierten Teams zurück
- `List<Evaluation> activateEvaluation(List<Team> teams):` bewertet die gegebenen Teams jeweils mit einem Geeignetheitsgrad und liefert eine Liste von zugehörigen Evaluationen zurück

Sie werden durch alle vier Templanungsalgorithmen implementiert.

In der Klasse `TeamSelectionBase` befinden sich Methoden und Klassenvariablen, die von den einzelnen Algorithmen überschrieben bzw. mit Werten belegt werden, da sich ihre Funktionalitäten je nach Ansatz unterscheiden. Die wichtigsten im Überblick:

- `SparseDoubleMatrix2D buildSkillMatrix():` berechnet die Skill-Matrix für einen Algorithmus
- `List<Team> buildTeams(SparseDoubleMatrix2D matrix):` auf der Grundlage der Skill-Matrix werden Mitarbeiter für Teams selektiert

Evaluation von Teams Die Evaluation implementiert die Kriterien optimaler Teams. Sie baut einmalig eine das gesamte Unternehmen betreffende Skill-Matrix für das gegebene Projekt. Die Zeilen der Matrix werden entsprechend den vorhandenen IDs der Mitarbeiter mit Ähnlichkeitswerten belegt.

Die Matrix basiert auf dem Konzept des zweiten Ansatzes, der somit als Vergleichsmaß für die Güte der Algorithmen benutzt wird. Die Verwendung von gewichteten Fähigkeiten wird als die Eigenschaft betrachtet, die den vier konzipierten Algorithmen einer gemeinsamen Basis am nächsten kommt. Anhand des vorliegenden Teamprofils werden diejenigen Zeilen der Skill-Matrix zu einer eigenen Teammatrix ausgelesen, die den IDs der Mitglieder entsprechen. Auf dieser gebildeten Matrix wird schließlich die Evaluation der Teams berechnet.

5.1.4. Anwendungsfälle

Der Anwendungsfall der Teamzusammensetzung wurde in 5.1.3 beschrieben. In diesem Teil wird die Funktionsweise der weiteren Anwendungsfälle erläutert. Für

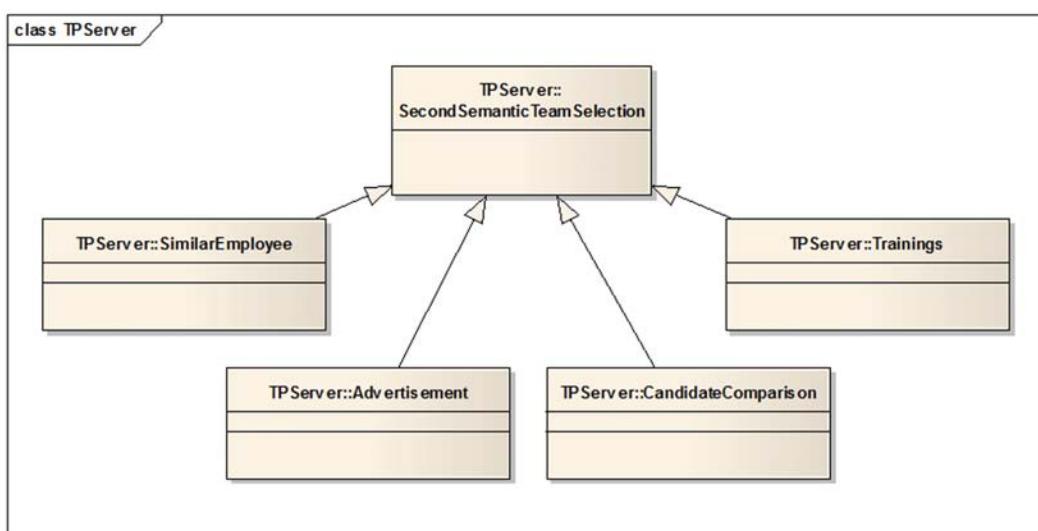


Abbildung 5.8.: Anwendungsfälle nutzen den 2. semantische Algorithmus

den Fall, dass keine geeigneten Projektteams zusammengesetzt werden, können die unter 3.3 beschriebenen Anwendungsfälle genutzt werden. Ihre Realisierung basiert auf dem Konzept des zweiten semantischen Ansatzes, zu sehen in Abbildung 5.8. Die Anwendungsfälle nutzen bereits generierte Skill-Matrizen, um die resultierende Rechenzeit gering zu halten.

Advertisement Zur Erzeugung von Kriterien einer Ausschreibung wird ein bereits vorhandenes Teamkompetenzprofil benötigt. Am geeignetsten ist dafür das während der Teamzusammensetzung generierte beste Team, um den Umfang einer Ausschreibung möglichst gering zu halten.

Die Skill-Matrix des Teams wird deserialisiert und enthält die Projektanforderungen als Spalten und die Teammitglieder als Zeilen. Die beiden höchsten Einträge jeder Spalte werden selektiert. Dabei markiert eine 1, dass der Mitarbeiter der entsprechenden Zeile die Projektfähigkeit mit dem geforderten Level besitzt. Werte kleiner als 1 zeigen eine Unter- oder Übererfüllung der Fähigkeit an. Das sind die Fälle, in denen eine Ausschreibung sinnvoll sein kann. Aus diesem Grund wird für die beiden höchsten Werte pro Spalte überprüft, ob die dazu gehörenden Mitarbeiter die Fähigkeit der Spalte besitzen oder lediglich eine sehr ähnliche Fähigkeit. Bei einem positiven Abgleich, werden die Level folgendermaßen verglichen:

- $pLevel < eLevel \Rightarrow$ Der Mitarbeiter besitzt einen höheren als den geforderten Level. Es ist keine Ausschreibung für diese Fähigkeit notwendig.
- $pLevel > eLevel \Rightarrow$ Die Fähigkeit sollte unter Angabe von pLevel ein Kriterium einer Ausschreibung sein.

Wenn der Mitarbeiter die Fähigkeit der Spalte nicht besitzt, wird auch diese Fähigkeit als Ausschreibungskriterium vorgeschlagen.

CandidateComparison Um zu überprüfen, ob ein Bewerber ein Team vervollständigen kann bzw. den Geeignetheitsgrad des Teams erhöht, wird für ihn ein Employee-Objekt erzeugt, das die Fähigkeiten des Bewerbers enthält. Ein Team wird um den Bewerber ergänzt und anschließend evaluiert. Der dabei entstandene Geeignetheitsgrad wird als Ergebnis ausgegeben.

Trainings Analog zu Advertisement wird hier ein existierendes Teamprofil verwendet. Idealerweise sollte es das bestbewertete Team sein, wodurch gewährleistet werden kann, dass möglichst wenig Weiterbildungen als Vorschläge generiert werden. Anwendbar ist Trainings aber auf jedes Teamkompetenzprofil.

Die Skill-Matrix des Teams wird zur Berechnung erneut deserialisiert. Aus der Matrix werden wie bei Advertisement die beiden höchsten Einträge pro Spalte selektiert, wobei ein 1-Eintrag bedeutet, dass der Mitarbeiter der entsprechenden Zei-

le die Projektfähigkeit bereits in seinem Profil und mit dem geforderten Level besitzt. In diesem Fall braucht keine Weiterbildung für die Fähigkeit vorgeschlagen zu werden. Kleinere Werte als 1 zeigen entweder eine Unter- oder Übererfüllung der Fähigkeit an. Aus diesem Grund wird für die beiden höchsten Werte pro Spalte überprüft, ob die dazu gehörenden Mitarbeiter die Fähigkeit der Spalte besitzen. Wenn dieser Vergleich positiv ausfällt, werden die Level folgendermaßen gegenübergestellt:

- pLevel < eLevel \Rightarrow Der Mitarbeiter besitzt bereits einen höheren als den geforderten Level. Er wird nicht für eine Weiterbildung vorgeschlagen.
- pLevel > eLevel \Rightarrow Der Mitarbeiter sollte sich in der Fähigkeit weiterbilden, um den höheren Level zu erreichen.

Wenn der Mitarbeiter die Fähigkeit der Spalte nicht besitzt, wird er für eine Weiterbildung in dieser Fähigkeit empfohlen.

SimilarEmployee Zwei Mitarbeiter sind dann sich für Projekt ähnlich, wenn sie möglichst die gleichen oder ähnlichen Fähigkeiten auf einem gleichen oder ähnlichen Niveau ausüben können, das bedeutet, dass ihre Werte in der Skill-Matrix beinahe deckungsgleich sind. Für diesen Vergleich werden zusätzlich zu dem gegebenen ausfallenden Teammitglied und einem Teamprofil die anderen zeitlich verfügbaren Mitarbeiter des Unternehmens benötigt.

Die Matrix, die während der Evaluation von Teams für das vorliegende Projekt entstanden ist, wird deserialisiert. Wie bereits genannt, richten sich die Einträge der Zeilen der Matrix nach der ID des Mitarbeiters. Die Matrix wird entsprechend der verfügbaren Mitarbeiter zusammengesetzt. Die folgende Gleichung zeigt die Berechnung der Ähnlichkeit zwischen zwei Mitarbeitern E_1 und E_2 :

$$sim(E_1, E_2) = \frac{\sum_{i=1}^n 1 - abs(matrix[E_1, i] - matrix[E_2, i])}{n} \quad (5.3)$$

In der Gleichung bezeichnet n die Anzahl der im Projekt benötigten Fähigkeiten i . $matrix[E_2, i]$ ist der Wert, der in der Zeile des Mitarbeiters an Spalte i zu finden ist. Die Ähnlichkeit zwischen zwei Mitarbeitern ist demnach der Mittelwert der absoluten Differenzen zwischen den Ähnlichkeiten ihrer Fähigkeiten. Sie wird also bestimmt durch die Abstände der projektrelevanten Fähigkeiten. Als Ergebnis wird eine Liste der Mitarbeiter mit ihrer Ähnlichkeit ausgegeben.

5.1.5. Komplexität und Skalierbarkeit

Die vorliegende prototypische Implementierung der Anwendung wird in diesem Abschnitt bezüglich Komplexität und weiterer Skalierungsindikatoren überprüft. Gerade bei der Betrachtung eines möglichen Einsatzes in einem großen Unternehmen, stellt sich die Frage, wie sich die Implementierung bei erhöhten Anforderungen verhält. Folgende Eigenschaften sollen untersucht werden:

- Performanz: bedeutet in diesem Zusammenhang, wieviel Zeitbedarf die Generierung der Teams beansprucht sowie die Länge der Antwortzeiten in Abhängigkeit der Eingaben
- Ressourcenbedarf: darunter fällt die Untersuchung bezüglich der Nutzung von Arbeitsspeicher
- Skalierbarkeit: beschreibt das Verhalten einer Anwendung, wenn sich ihr Umfeld vergrößert, also Rechen- und Speicheraufwände in Folge von wachsenden Eingaben steigen

Der permutative Ansatz macht einen einfachen Soll-Ist-Abgleich zwischen allen Projektanforderungen und Fähigkeiten der Mitarbeiter. Findet er hierbei eine Übereinstimmung, ist der entsprechende Mitarbeiter im Rahmen der zu bildenden Permutationen und damit der Teams zu berücksichtigen. Im rechenaufwendigsten Fall erfüllt jeder der Mitarbeiter mindestens eine der Projektanforderungen, so dass die Menge, aus der Permutationen gebildet werden, genau der Menge der Mitarbeiter entspricht. Die Anzahl der Permutationen einer Menge mit n Elementen beträgt $n!$ (Fakultät von n). Für ein Unternehmen mit 50 Mitarbeitern, wobei jeder eine projektrelevante Fähigkeit besitzt, würden insgesamt $50! = 3.041 * 10^{64}$ mögliche Teams gebildet werden. Der permutative Ansatz reduziert diese Menge beim Traversieren des Permutationsbaumes bereits um Teams, die außerhalb der optimalen Teamgröße von 4 bis 10 Mitgliedern liegen. Ein Team wird außerdem nur dann in das Ergebnis aufgenommen, wenn alle geforderten Projektanforderungen von ihm erfüllt werden. Mit steigender Unternehmensgröße wachsen daher bei der Verwendung des permutativen Ansatzes die Komplexität der Berechnung sowie der Rechen- und Speicheraufwand in einem erheblichen Maß. Es ist zu erwarten, dass der permutative Ansatz eine unzureichende Skalierbarkeit aufweist, da mit der Anzahl der Permutationen auch der Arbeitsspeicheraufwand in hohem Maße zunimmt.

Die Performanz der semantischen Algorithmen ist zu einem hohen Anteil von der Berechnung der Skill-Matrizen abhängig. Die Berechnung einer solchen Matrix erfolgt im ungünstigsten Fall über den Vergleich jeder Mitarbeiterfähigkeit mit jeder Projektfähigkeit. Die Anzahl der Vergleiche wird reduziert, wenn exakt die gesuchte Fähigkeit beim Mitarbeiter vorhanden ist. Um ständige Zugriffe auf die Ontologie zu vermeiden, werden die für die Ähnlichkeitsberechnung verwendeten kürzesten Pfade zwischen zwei Knoten in der Ontologie in einem Vorverarbeitungsschritt serialisiert und sind somit zur Laufzeit des Algorithmus effizient zugreifbar. Die Anzahl der Zellen einer Skill-Matrix für den ersten, zweiten und dritten semantischen Ansatz beträgt:

$$Zellen_{Skill-Matrix} = Anzahl_{verfügbarer Mitarbeiter} * Anzahl_{P-Skills} \quad (5.4)$$

In jeder Zelle dieser Skill-Matrix wird für die Ähnlichkeit ein Dezimalwert doppelter Genauigkeit zwischen 0 und 1 eingetragen, der in Java 8 Byte Speicherplatz belegt.

Hierzu ein Beispiel: für ein Unternehmen mit 30 verfügbaren von insgesamt 50 Mitarbeitern verbraucht eine Skill-Matrix für ein Projekt mit 5 Fähigkeiten somit insgesamt 1200 Byte für die Speicherung der Vergleichswerte. Besitzt ein Mitarbeiter 10 Fähigkeiten und befindet sich unter ihnen im schlechtesten Fall nicht die geforderte Fähigkeit, sind insgesamt 10 Ähnlichkeitsvergleiche notwendig. Für die gegebenen 5 Projektanforderungen werden daher im ungünstigsten Fall 50 Ähnlichkeitsberechnungen durchgeführt.

Die Menge der für den Selektionsvorgang relevanten Mitarbeiter wird durch die Auswahl einer konstant geringen Anzahl von m Mitarbeitern für jeweils eine Projektfähigkeit deutlich reduziert. Die Konstante m wurde für die Anwendung mit 3 initialisiert um eine Erfüllung des Kriteriums der Überlappung zu gewährleisten, ist jedoch nach Bedarf anpassbar. Diese m Mitarbeiter können mittels einer linearen Operation auf der Matrix selektiert werden. Diese vergleichsweise kleinen Listen relevanter Mitarbeiter werden dann zu der Menge möglicher Teams permutiert. Aufgrund der im Vergleich zum permutativen Algorithmus wesentlich kleineren Permutationsmengen pro Fähigkeit ergibt sich hier eine deutlich reduzierte Laufzeit des Auswahlvorgangs und in gleichem Maß ein geringerer Speicheraufwand. Die zuvor beschriebenen Mechanismen verbessern die Skalierbarkeit des Algorithmus wesentlich. Für eine entsprechend große Eingabemenge erfordert die Berech-

nung der Skill-Matrix erwartungsgemäß den hauptsächlichen Aufwand. Dem wird durch eine Serialisierung bereits berechneter Matrizen begegnet, die dann effizient wiederverwendet werden können. So kann die Teamplanung für ein bereits zuvor definiertes Projekt mit einer entsprechend geringen Berechnungsdauer durchgeführt werden.

Der zweite semantische Algorithmus unterscheidet sich in seiner Komplexität durch die Berücksichtigung eines weiteren Parameters bei der Erstellung der Skill-Matrizen. Der Aufwand hierfür wird daher um eine vernachlässigbare Konstante erhöht. In der Selektion führt die Berücksichtigung der Priorität der Projektfähigkeiten ebenfalls zu keinem relevanten Mehraufwand.

Für den dritten semantischen Algorithmus ist zusätzlich zur Skill-Matrix die Berechnung einer Task-Matrix erforderlich und die Berücksichtigung dieser bei der Selektion von Teams. Der Gesamtaufwand sowohl an Speicher als auch an Rechenzeit wird daher näherungsweise verdoppelt.

5.2. Test

Um eine Aussage über die Leistungsfähigkeit der entworfenen Algorithmen treffen zu können, wurden in einem automatisierten Verfahren auf Basis von eigens generierten Firmen- und Projektdaten umfangreiche Tests durchgeführt. Die statistische Auswertung dieser lässt Rückschlüsse auf die qualitativen und quantitativen Unterschiede zwischen den Ansätzen zu.

5.2.1. Vorbetrachtung

Zur Bestimmung der Testkriterien und -umgebung werden die folgenden Vorüberlegungen zu möglichen Problemen und zu erwartenden Ergebnissen angestellt. Nach der vorangegangenen Konzeption wird erwartet, dass sich die Ergebnisse der semantischen Algorithmen untereinander ähneln. Es wird aufgrund der zum Einsatz kommenden Ähnlichkeitsberechnungen davon ausgegangen, dass sie auch in unterschiedlich beschaffenen Unternehmensstrukturen in der Lage sind, die am

besten geeigneten Teams zu finden und diese entsprechend ihres Grades zu bewerten. Je größer das betrachtete Unternehmen, desto größer wird die Wahrscheinlichkeit, dass die Menge der Projektanforderungen bei den Fähigkeiten der Mitarbeiter gefunden werden kann. Es wird daher erwartet, dass in größeren Unternehmen auch eine größere Menge geeigneter Teams zusammengestellt wird.

Der permutative Algorithmus wird voraussichtlich in manchen Fällen eine große Menge von Teams finden, unter denen dann auch die bestmöglichen Zusammensetzungen von Mitarbeitern sind. Bei Nichtvorhandensein einer genauen Übereinstimmung für jede Anforderung zu dieser Kombination aus Projekt und Unternehmen werden keine Teams ermittelt. Die Ergebnisse der semantischen Algorithmen werden sich deutlich vom nicht-semantischen Ansatz abheben, da sie auch ohne vollständige Übereinstimmungen zwischen Projektanforderung und Mitarbeiterfähigkeit Teams generieren können.

Die Rechenzeit der semantischen Ansätze wird sich mit ansteigender Unternehmensgröße erhöhen, da sich die Erstellung der Skill-Matrizen rechen- und somit zeitintensiver gestaltet. Durch deren Wiederverwendung bei wiederholtem Aufruf des Teambildungsalgorithmus für ein Projekt kann die Berechnungszeit reduziert werden.

Der dritte semantische Algorithmus stützt sich auf eine erweiterte Datengrundlage, indem er auch die Aufgaben aus dem bisherigen Einsatzgebiet der Mitarbeiter berücksichtigt. Dies verhindert die Vergleichbarkeit der Ergebnisse zu den anderen Ansätzen. Der Algorithmus wird daher im Rahmen der Testfälle nicht betrachtet.

5.2.2. Eigenschaften der Testfälle

Die Tests wurden auf Basis dreier Unternehmenskategorien durchgeführt. Es wird unterschieden in kleine Unternehmen mit jeweils 30 Mitarbeitern, mittelgroße Unternehmen mit 100 Mitarbeitern und große Unternehmen mit 500 Beschäftigten. Auf eine Betrachtung noch größerer Unternehmen wurde verzichtet, da in Konzernen, deren Mitarbeiteranzahl die genannten Mengen übertrifft, eine Aufteilung in Abteilungen und Divisionen üblich ist, was die Anzahl der für ein spezifisches Projekt zur Verfügung stehenden Mitarbeiter wiederum einschränkt.

Zur Generierung der Test-Unternehmen wurde ein zufallsbasierter Ansatz verfolgt. Jedem generierten Mitarbeiter wurde eine Menge von bis zu 20 Fähigkeiten zugeordnet, von denen jede Bestandteil der Fähigkeitenklasse der HR-Ontologie

ist, und die auf einem individuellen Level beherrscht wird. Zusätzlich wurde die Verfügbarkeit einiger Mitarbeiter zeitlich eingeschränkt, um die nebenläufige Ausführung mehrerer Projekte im Unternehmen nachzubilden.

Auch die für die Tests benötigten Projekte wurden zufällig erstellt. Den Projekten wurde jeweils ein Zeitraum, sowie eine Menge von fünf benötigten Fähigkeiten zugeordnet. Dies dient der Annäherung an reale Vorhaben und sichert die Vergleichbarkeit der Ergebnisse. Die Untersuchung von Projekten mit wesentlich mehr benötigten Fähigkeiten führt in der Realität voraussichtlich zu einer Aufteilung des Projekts in mehrere Teilprojekte, weil zu viele Aufgaben von einem einzelnen Projektteam nicht erledigt werden können.

Als Basis für die Bewertung der durch die jeweiligen Algorithmen ermittelten Teams dient der unter 4.4.5 beschriebene Prozess der Evaluation unter der Berücksichtigung von Ähnlichkeiten zwischen gewichteten Fähigkeiten.

Es wurden jeweils pro Unternehmenstyp 50 vollständige Testsätze erzeugt. Für jeden dieser Sätze wurden anhand von zehn konstant verwendeten Testprojekten mit jedem der drei Algorithmen Teams ermittelt. Die Testergebnisse werden im Folgenden in Bezug auf Qualität und Quantität vorgestellt und untersucht.

5.2.3. Untersuchung der Testergebnisse

Die Qualität wird von der Team-Evaluation mit einem Dezimalwert zwischen 0 und 1 angegeben. Abbildung 5.9 zeigt die mittlere Verteilung von Durchschnitt, Maximum und Minimum in der Bewertung der vom jeweiligen Algorithmus ermittelten Teams für den kleinen Unternehmenstyp. Die Säulen im Diagramm stellen den jeweiligen Wert dar, während die Abweichungsindikatoren in den Säulen die obere und untere Maximalabweichung anzeigen.

Die beiden semantischen Algorithmen liefern in dieser Darstellung auf Basis einer identischen Datengrundlage ein sehr ähnliches Ergebnis, der zweite Ansatz zeigt hier in geringem Maße bessere Resultate. Zur weiteren Interpretation der Ergebnisse bietet sich die in Abbildung 5.10 gezeigte quantitative Darstellung an. Hierbei ist die logarithmische Skalierung der x-Achse zu beachten.

Für den permutativen Algorithmus wird hier eine geringe durchschnittliche Anzahl von Teams gefunden. Die Abweichungsindikatoren zeigen jedoch, dass in Einzelfällen bis zu mehreren tausend Teams erzeugt wurden. Aus dieser Menge von Teams eine passende Auswahl zu treffen, dürfte einem Teamrecruiter deutlich

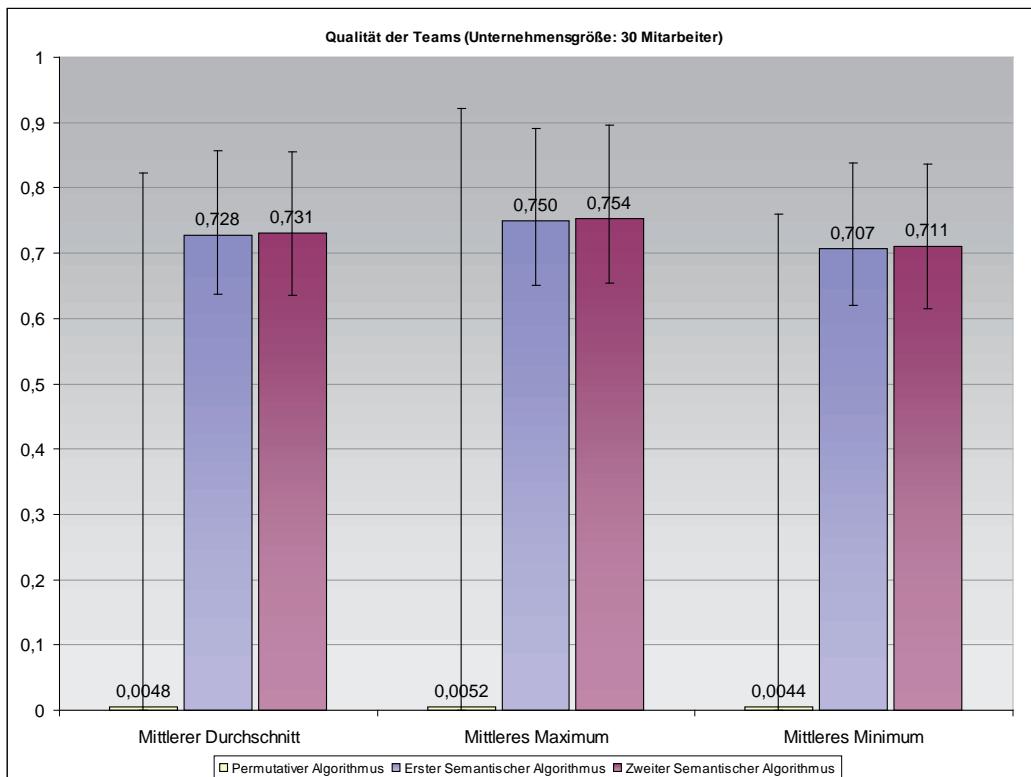


Abbildung 5.9.: Auswertung der Teamqualität bei Unternehmen mit 30 Mitarbeitern

schwerfallen, zumal der Geeignetheitsgrad der Teams ihm nur durch die Unterstützung durch die semantische Evaluation zur Verfügung steht. In Abbildung 5.11 ist die Verteilung der Anzahlen der pro Unternehmen und Projekt gefundenen Teammengen gezeigt. Hieraus wird der sehr geringe Wirkungsgrad des permutativen Algorithmus ersichtlich, in den meisten Fällen werden keine Teams generiert. Um diesen Ansatz erfolgreich einzusetzen wird vom Teamrecruiter ein deutlich größeres Wissen über die Mitarbeiter des Unternehmens und die hierzu passenden Projekte erforderlich.

Der erste und zweite semantische Ansatz liegen auch quantitativ erwartungsgemäß auf einem ähnlichen Niveau, wobei der zweite Ansatz im Durchschnitt eine leicht größere Anzahl Teams ermittelt.

In den Testreihen für Unternehmen ab 100 Mitarbeitern konnte der permutative Algorithmus nicht weiter beobachtet werden, da der auf dem Testsystem in der Java-Umgebung verfügbare Arbeitsspeicher von maximal 1,4 GByte für die Berechnung der Permutationen nicht mehr ausreichend war. Im Folgenden reduziert sich die Betrachtung daher auf den Vergleich der beiden semantischen Ansätze.

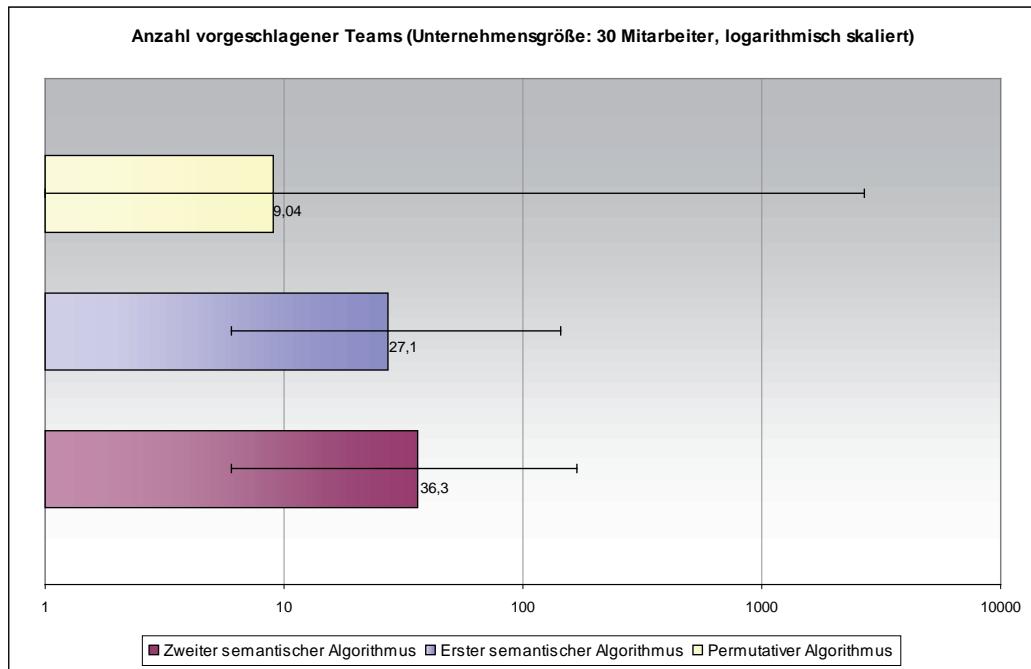


Abbildung 5.10.: Menge der gefundenen Teams für Unternehmen mit 30 Mitarbeitern

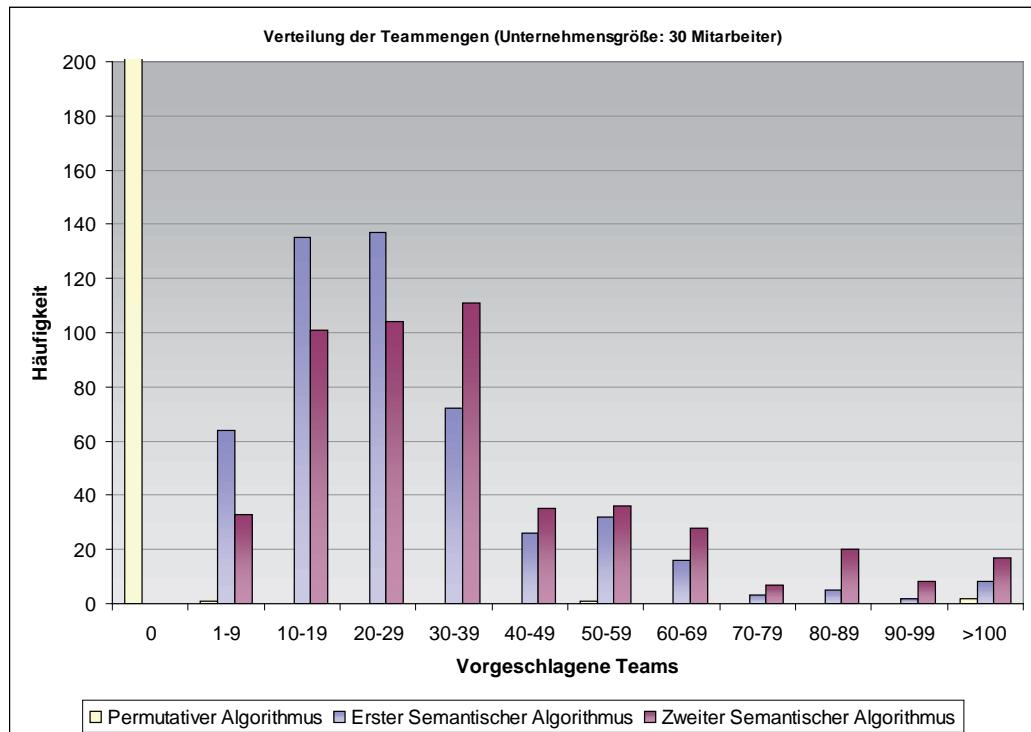


Abbildung 5.11.: Verteilung der Anzahl gefundener Teams für Unternehmen mit 30 Mitarbeitern

Bei der Untersuchung des nächstgrößeren Unternehmenstyps zeigt sich hier, dass die Unterschiede bei der Teamqualität deutlicher zugunsten des zweiten Algorithmus ausfallen, wobei die Qualität insgesamt im Vergleich zu den Unternehmen mit 30 Mitarbeitern gestiegen ist (5.12). Dies liegt in der besseren Verfügbarkeit von Fähigkeiten in einem größeren Unternehmen begründet. Auch die Anzahl der

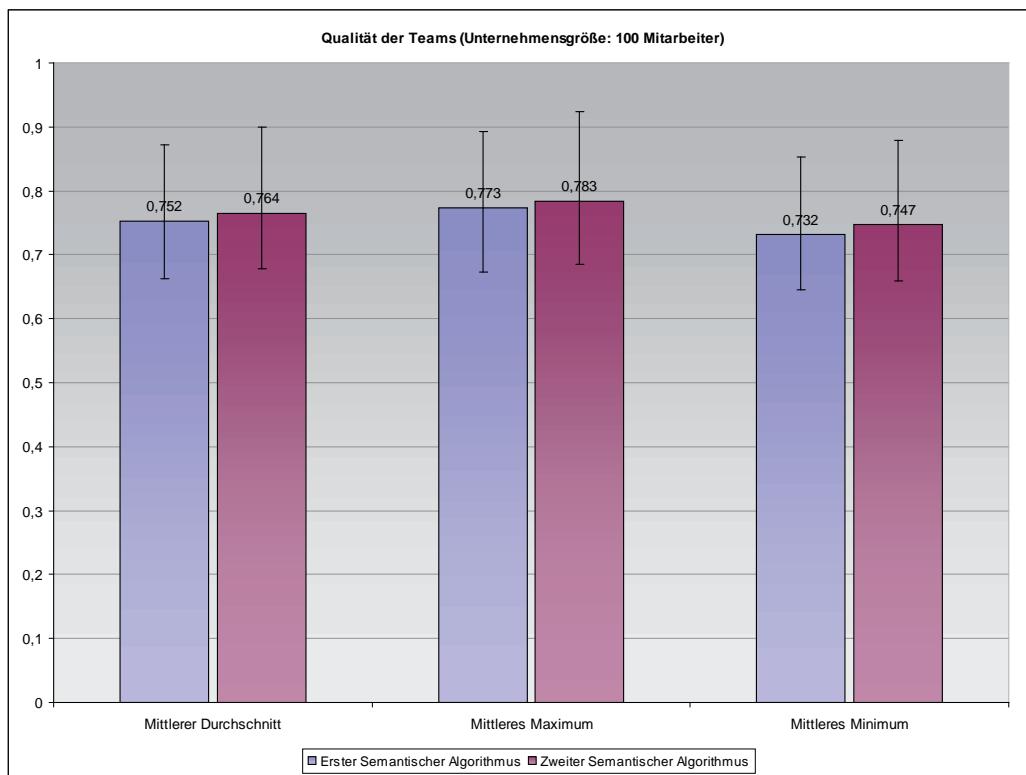


Abbildung 5.12.: Auswertung der Teamqualität bei Unternehmen mit 100 Mitarbeitern

gefundenen Teams steigt im Vergleich deutlich an, durchschnittlich werden beim mittleren Unternehmenstyp mit dem ersten semantischen Algorithmus 82,0 Teams und mit dem zweiten semantischen Algorithmus 99,7 Teams gebildet.

Die Ergebnisse des größten Unternehmenstyps zeigen wiederum die höhere Qualität des zweiten Algorithmus. In Abbildung 5.13 zeigt sich zusätzlich wieder, dass der durchschnittliche Geeignetheitsgrad der Teams im Vergleich zu den kleineren Unternehmen gestiegen ist. Dies bestätigt, dass eine höhere Verfügbarkeit von Fähigkeiten zu besser bewerteten Teams führt. Tabelle 5.1 stellt die Anzahl der im Durchschnitt gefundenen Teams je Unternehmenstyp dar. Es ist ersichtlich, dass mit steigender Unternehmensgröße die Menge der gefundenen Teams überschau- bar bleibt. In den Testresultaten sind hiervon auch keine extremen Abweichungen

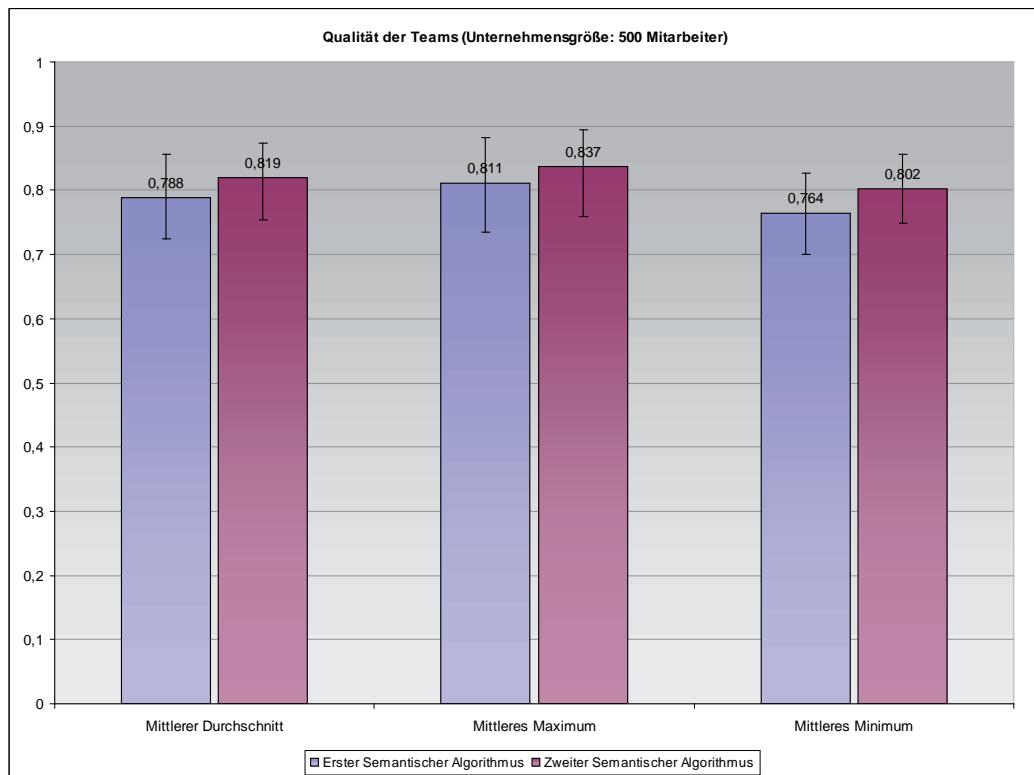


Abbildung 5.13.: Auswertung der Teamqualität bei Unternehmen mit 500 Mitarbeitern

zu beobachten. Während der Testreihen konnte bestätigend festgestellt werden,

Unternehmenstyp	klein	mittel	groß
Erster semantischer Algorithmus	27,1	82,0	105,6
Zweiter semantischer Algorithmus	36,3	99,7	121,8

Tabelle 5.1.: Vergleich der durchschnittlichen Anzahl gefundener Teams

dass die Berechnung der Skill-Matrizen den größten Anteil der Laufzeit in Anspruch nimmt. Der Zeitbedarf für diese Operation blieb über den Verlauf der Tests für jeweils eine Unternehmensgröße weitgehend konstant. Für die kleinen Unternehmen wurden die Matrizen auf dem Testsystem in unter einer Minute, für die mittleren Unternehmen in vier und für die großen Unternehmen in etwa 15 Minuten berechnet. Durch die Wiederverwendung der Matrizen bei gleichen Anfragen wurden in weiterführenden Untersuchungen die Berechnungszeit auf jeweils wenige Sekunden reduziert.

Mit Unterstützung der Tests konnte festgestellt werden, dass die beiden seman-

tischen Algorithmen in Bezug auf die Qualität der ermittelten Teams zufriedenstellende Ergebnisse liefern. Die Evaluationswerte dieser befinden sich in einem angemessenen Intervall. Qualitativ schlechte Ergebnisse konnten ebenso wenig beobachtet werden wie der Fall, dass der Algorithmus keine Teams für ein Projekt bilden konnte. Insgesamt ist der zweite Algorithmus in der Qualität seiner Ergebnisse dem ersten Ansatz überlegen und es erscheint sinnvoll, ihn bei entsprechender Datenlage (Mitarbeiterfähigkeiten liegen gewichtet vor) bevorzugt einzusetzen.

6. Zusammenfassung und Ausblick

Die vorliegende Ausarbeitung lässt sich in drei wesentliche Blöcke unterteilen, die in diesem Abschnitt zusammengefasst betrachtet werden sollen:

- Grundlagen und Rahmen der Arbeit
- Anforderungen und konzeptionelle Lösung
- Technische Umsetzung einer prototypischen Applikation

Beginnend mit den Grundlagen werden die Bereiche Personal- und Human Resource Management sowie Wissens- und Kompetenzmanagement erläutert und der Kompetenzbegriff definiert. Recruitment wird als mögliches Einsatzszenario des entwickelten Prototyps beleuchtet und der Wert von Informationstechnologien in diesem Zusammenhang betrachtet. Die Eigenschaften und Arten von Teams werden ausführlich begutachtet. Als technische Grundlage werden Semantic Web und Ontologien beschrieben. Insbesondere werden mit dem Vergleich der existierenden HR-Ontologien der Forschungsstand und der bisherige Einsatz von semantischen Webtechnologien im Bereich Human Resources gezeigt.

Anhand von Ergebnissen der themaverwandten Arbeiten werden zusätzlich zu Interviews mit Siemens Teamrecruitern eine Anforderungsanalyse und die Anwendungsfälle der prototypischen Applikation erstellt. Der wichtiger Teil der Arbeit wird durch die ausführliche Betrachtung und Definition der Kriterien optimaler Teams gebildet. Sie formen die Basis der drei entwickelten semantischen Teamplanungsalgorithmen sowie der Evaluation von Teams.

Die Implementierung wird durch das Aufzeigen ihrer Architektur und einer Beschreibung der darin enthaltenen Komponenten vorgestellt. Die durchgeföhrten Tests in unterschiedlichen Szenarien zeigen abschließend das Verhalten der prototypischen Implementierung unter verschiedenen Bedingungen.

6.1. Bewertung der Ergebnisse

Ziel der vorliegenden Ausarbeitung war die Entwicklung einer Applikation, die einen Teamrecruiter eines Unternehmens dabei unterstützen soll, möglichst optimale Projektteams zusammenzusetzen. Die Testergebnisse aus mehreren differenzierten Testläufen in Abschnitt 5.2.3 zeigen, dass die Teamplanung unter Nutzung von semantischen Webtechnologien durchführbar ist und zufriedenstellende Ergebnisse liefert. Die entwickelten Algorithmen basieren anstatt auf Soll-Ist-Vergleichen auf konsequent durchgeführten Ähnlichkeitsberechnungen, wodurch das Erbringen von Teamzusammensetzungen gewährleistet werden kann. Dies bestätigen die Testergebnisse, in denen für jede überprüfte Eingabe Teams gebildet werden konnten. Insbesondere aufgrund der eingeschränkten Nutzbarkeit des herkömmlichen permutativen Ansatzes in einem zufälligen Unternehmensumfeld wurde erkennbar, wie groß der Mehrwert von semantischen Technologien im HR-Bereich sein kann.

Recherchierte und untersuchte Arbeiten in diesem Umfeld liefern bereits gute Ansätze zur Schaffung einer IT-gestützten Planung von Projektteams (siehe Abschnitt 3.1.5, DaimlerCrysler AG). Die Ergebnisse dieser Arbeiten konnten zunächst durch die Nutzung semantischer Technologien und zusätzlich auch über die Implementierung der recherchierten Kriterien optimaler Teams verbessert werden. Dadurch wurde eine ganzheitliche Lösung bezüglich der Teamplanung sowie der nachhaltige Nutzen zur algorithmenunabhängigen Evaluation von Teamzusammensetzungen basierend auf fachlichen Qualifikationen geschaffen. Biesalski kritisiert in [Bie06], dass die Besetzung von vakanten Stellen in Projekten meistens auf persönlichen Netzwerken und Vorlieben beruht. Den Nachteilen dieser vorherrschenden Herangehensweise wird begegnet, indem eine objektive Besetzung von Teams ermöglicht wird. Das betrifft die Auswahl der verfügbaren Mitarbeiter anhand einer automatisierten Verarbeitung ihrer Daten, den objektiven Abgleich zwischen Projekt- und Mitarbeiterprofilen sowie eine objektive Evaluation anhand der Teamkriterien.

Des Weiteren schildert Deiters in [DL00] (siehe Abschnitt 3), wie Kompetenzmanagement-Systeme mit Teamplanungsfunktionalität aufgebaut sein sollten. Es kann zusammengefasst werden, dass der entwickelte Prototyp, wie von Deiters gefordert, ein gleichzeitiges automatisiertes Zusammenstellen eines kompletten Teams ermöglicht. Dadurch wird den Nachteilen einer sukzessiven Lösung begegnet.

Bei der Erstellung einer Lösung wurde konsequent auf durch den Benutzer einzustellende Parameter verzichtet. Wie Rüßbüldt in [RÖ6] beschreibt, muss der Benutzer seines Systems z.B. die Genauigkeit der Suche bestimmen, wodurch diese Entscheidung direkten Einfluss auf die Ähnlichkeitsberechnungen nimmt. Für solche Herangehensweisen ist es unabdingbar, dass der Benutzer ausreichend Wissen über die Verwendung und die Bedeutung der Parameter besitzt und sich über die Folgen der gewählten Einstellung bewusst ist. Um den Benutzer von dieser Aufgabe zu entlasten, beruhen Ergebnisse in der prototypischen Anwendung vollständig auf dem Wissen und den Informationen, die die semantischen Algorithmen aus den vorliegenden Mitarbeiter- und Projektbeschreibungen erlangen.

Die konzipierten Skill-Matrizen erlauben weitere unternehmensrelevante Begutachtungen bezüglich der Verteilung von Kompetenzen. Die Skill-Matrix über Kombinationen von zu untersuchenden Mitarbeitern und Fähigkeiten ermöglicht zusätzlich zu den in Abschnitt 3.3 genannten Anwendungsfällen die auf Ähnlichkeitsberechnungen beruhende Ermittlung von Kernkompetenzen sowie Skill-Gap-Analysen, die beide als Basisaufgaben eines Kompetenzmanagement-Systems von diversen Autoren (z.B. Gronau [FSRG06]) betrachtet werden. Damit besitzen die entwickelten Konzepte noch umfassenderes Potential und belässt Raum für Weiterentwicklungen.

Offen bleibt die statistische Untersuchung der Wirkungsweise des dritten semantischen Algorithmus. Er wurde aufgrund einer unzureichend vergleichbarer Funktionsweise zu den anderen Algorithmen nicht in die Testreihen integriert. Außerdem gestaltete sich die Generierung von Testdaten schwierig, da Fähigkeiten und Aufgaben in ihrem gemeinsamen Vorkommen auch eine gemeinsame Semantik besitzen. Das Testen in einem zufälligen Szenario, vor allem bei Berücksichtigung verschiedener Unternehmensgrößen, hätte diesen Umstand jedoch nicht gewährleisten können. In diesem Fall wären Ähnlichkeiten zwischen Fähigkeiten und Aufgaben berechnet worden, die in einem realen Umfeld in dieser Kombination voraussichtlich nicht auftauchen würden, wodurch die Aussagekraft der Ergebnisse verfälscht würde.

Im Rahmen der Ausarbeitung wurde ein Prototyp für die Implementierung erstellt. Für den produktiven Einsatz in einem Unternehmen sollten Funktionen bezüglich der unter 3.2 genannten Anforderungsanalyse und einer Systemspezifikation ergänzt werden. Im Abschnitt 3.1.1 wird beschrieben, wie im Knowledge Management System von Stader und Macintosh auch Trainees in Projektteams beteiligt

werden. In der entwickelten Applikation werden für Projektteams generell die besten verfügbaren Mitarbeiter für die Projektanforderungen gewählt. Diese Vorgehensweise besitzt den Vorteil, dass Teams von höherer Geeignetheit gebildet werden aber auch den Nachteil, dass niedriger qualifizierte Mitarbeiter unberücksichtigt bleiben. Da diese Konstellation in mehreren Projekten nacheinander auftreten kann, werden bestimmte Mitarbeiter immer für Projekte vorgeschlagen und andere nie. Diesem Umstand kann man entgegenwirken, in dem die Lösung von Stader und Macintosh auf den Anwendungsfall der Weiterbildungsmaßnahmen adaptiert wird: Mitarbeiter können sich auch durch ihre Mitwirkung in Projekten weiterbilden. In Folge dessen wird eine homogenerere Weiterentwicklung der Kompetenzen von Mitarbeitern gewährleistet.

6.2. Ausblick

Im Folgenden werden zukünftige Fragestellungen und mögliche Weiterentwicklungen des Themas vorgestellt.

Die Verarbeitung von Fähigkeiten und Aufgaben ermöglicht eine kompetenzorientierte Zusammensetzung von Teams. Bei der Konzeptionierung einer Lösung sind „soft skills“ nicht betrachtet worden, da deren objektive Erfassung als schwierig erachtet wird. Eine geeignete Teamzusammensetzung beruht allerdings zu einem beachtlichen Teil auch auf charakterlichen Fähigkeiten und deren angemessener Kombination in einem Team. Die Belbin-Rollen [Bel04] zur Teambildung beispielsweise basieren auf psychologischen Persönlichkeitsprofilen. Aus diesem Grund wird eine Weiterentwicklung des Themas durch die zusätzliche Betrachtung weicher Fähigkeiten für die Teamplanung als sinnvoll erachtet. Im Abschnitt 2.1.2 wurde aufgezeigt, dass Methoden zur Bewertung von „soft skills“ existieren. Eine umfassende Recherche zur Nutzung dieser Methoden im Bereich von semantischen Webtechnologien wird für die Weiterentwicklung notwendig werden. Im Idealfall sollte die Erarbeitung einer Lösung in dieser Richtung interdisziplinär erfolgen.

Für die Ähnlichkeitsberechnungen wurde mit Fähigkeiten und Aufgaben ein relativ kleiner Ausschnitt der HR-Ontologie verwendet. Sie besitzt allerdings noch mehr Potential zur Modellierung von Mitarbeitern bezüglich des Teamplanungsprozesses. Die Erweiterung der Ontologie um Teamaspekte zur Unterstützung

vollständiger semantischer Mitarbeiterprofile wird deshalb als vernünftig angesehen.

In dem Zusammenhang der Ähnlichkeitsberechnungen sollte evaluiert werden, welche existierende Ähnlichkeitsfunktion für die Teamplanung am geeignetsten ist bzw. ob die spezialisierte Entwicklung eines solchen Maßes zu noch effizienteren Teamzusammensetzungen führen kann.

Eine weitere Performanzsteigerung kann durch Caching von Ähnlichkeitsberechnungen erreicht werden. Dazu sollten die berechneten Zeilen der Skill-Matrix der entsprechenden Mitarbeiter auch in anderen Projekten einfacher wiederverwendbarer sein. Die bereits vorhandenen Matrizen sollten ohne erneute Berechnung genutzt werden und müssten lediglich nach einer Änderung der Mitarbeiterfähigkeiten (z.B. durch Weiterbildungsmaßnahmen) oder aber auch bei Veränderungen der Ontologie angepasst werden.

Für die Begegnung einer prozessorientierten Vorgehensweise für Teamzusammensetzungen, wie bei der Betrachtung von KMDL in Abschnitt 3.1.3 beschrieben, wäre die Weiterentwicklung des dritten Ansatzes denkbar. Durch eine Verknüpfung von Fähigkeiten zu Aufgaben bereits bei der Generierung von Teams, können Mitarbeiter direkt mit einer ausführenden Rolle, wie bei der Analyse von KOWIEN in Abschnitt 3.1.4 dargestellt, markiert werden.

Denkbar wäre auch die Erweiterung um die parallele Generierung von Teams für mehrere Projekte. In diesem Fall müssen die verwendeten Algorithmen die vorhandenen Kompetenzen gerecht auf die Projekte aufteilen, da sonst einige Projekte bei ihrer Durchführung benachteiligt würden. Gerade in großen Unternehmen herrscht eine erhöhte Fluktuation von Projekten, d.h. zu vielen Zeitpunkten beginnen und enden die Arbeiten in Projekten. Diese Unternehmen könnten von einer parallelen Berechnung von Teamzusammensetzungen für verschiedene Projekte profitieren.

Abschließend betrachtet wird deutlich, dass dieses Thema noch enorm viel Potential besitzt, weil es aktuell und von großer Bedeutung für eine Unterstützung entsprechender Prozesse im HR-Bereich ist. Die Ergebnisse lassen den erhöhten Mehrwert von Semantic Web Technologien gerade auch im Unternehmensumfeld erkennen. Eine weitergehende Untersuchung der Nutzungsmöglichkeiten von semantischen Technologien für den Bereich der Human Resources erscheint vielversprechend.

Abbildungsverzeichnis

1.1.	Teamplanung im Projektzeitplan, angelehnt an [Bie06]	7
2.1.	Auszug aus der ACTA 2008, angelehnt an [Köc08]	12
2.2.	Die vier Team Typen, nach [SH08]	17
2.3.	Teamarten, angelehnt an [SH08]	18
2.4.	Typen von Ontologien, nach [SW01]	23
2.5.	Schema der drei Teile der LIP Ontologie, angelehnt an [Sch05]	27
2.6.	Aufbau der Professional Learning Ontology	28
3.1.	Relevante Basiskonzepte, angelehnt an [SM99]	33
3.2.	Modell der Kompetenzen, angelehnt an [HH05]	38
3.3.	Beispiel für „Erfahrungsräume“	40
3.4.	Wissenskonversionen bei KMDL, angelehnt an [Gro06]	44
3.5.	Modellierung der Koverisionsarten mit verfügbaren Objekten	44
3.6.	Staffing-Ansätze im Überblick, angelehnt an [RÖ6]	46
3.7.	Teambildungsalgorithmus, angelehnt an [RÖ6]	47
3.8.	mögliche Unternehmensstruktur, angelehnt an [Bie06]	57
4.1.	Nutzung von kontrollierten Vokabularien	75
4.2.	Strukturierung der Fähigkeiten	76
4.3.	Ausprägungen von Kompetenz	77
4.4.	Teamleistung in Abhängigkeit der Teamgröße, Quelle: [Lin06]	79
4.5.	Ablauf des ersten semantischen Algorithmus	92
4.6.	Beispiel für die Skill-Matrix des ersten semantischen Ansatzes	93
4.7.	Schema des zweiten Algorithmus	96
4.8.	Beispiel für die Skill-Matrix des zweiten semantischen Ansatzes	98
4.9.	Phasen des dritten Ansatzes	100
4.10.	Beispiel für die Skill-Matrix und Aufgaben des dritten semantischen Ansatzes	101

5.1.	Komponentenmodell der Anwendung	108
5.2.	Datenbankmodell SemTeamDB	110
5.3.	Kantenzählmethode nach Resnik	111
5.4.	Konzeptuelle Ähnlichkeit nach Wu und Palmer	112
5.5.	Klassendiagramm der ClientApplication	114
5.6.	Anzeige selektierter Projektteams im TeamViewer	115
5.7.	Klassendiagramm der Algorithmen und Evaluation	117
5.8.	Anwendungsfälle nutzen den 2. semantische Algorithmus	118
5.9.	Auswertung der Teamqualität bei Unternehmen mit 30 Mitarbeitern	126
5.10.	Menge der gefundenen Teams für Unternehmen mit 30 Mitarbeitern	127
5.11.	Verteilung der Anzahl gefundener Teams für Unternehmen mit 30 Mitarbeitern	127
5.12.	Auswertung der Teamqualität bei Unternehmen mit 100 Mitarbeitern	128
5.13.	Auswertung der Teamqualität bei Unternehmen mit 500 Mitarbeitern	129

Literaturverzeichnis

- [Abe04] ABECKER, Andreas: *Tools im Wissensmanagement - Ein Überblick.* DTT Symposium 2004 „Terminologie und Wissensmanagement“, Köln, 2004
- [AZ03] ALAN, Yilmaz ; ZELEWSKI, Stephan: Ontologiebasierte Wissensräume / Universität Duisburg-Essen. 2003. – Forschungsbericht
- [BCT06] BREITMAN, Karin ; CASANOVA, Marco A. ; TRUSZKOWSKI, Walter: *Semantic Web: Concepts, Technologies and Applications (NASA Monographs in Systems and Software Engineering).* Springer Verlag Berlin, 2006
- [Bel04] BELBIN, R. M.: *Management Teams. Why They Succeed Or Fail.* Second Edition. Oxford : Butterworth Heinemann, 2004
- [BH04] BÄUMGEN, Christof ; HÜBBERS, Roger: Ontologiebasiertes Kompetenzmanagementsystem auf Basis von infonea®. In: *Industrieerprobte Lösungen und Werkzeuge für Produktentwicklung, Engineering und Kompetenzmanagement*, 2004, S. 161–173. – Abschlussworkshop der Verbundprojekte GINA, Koeffizient und KOWIEN
- [Bie04] BIESALSKI, Ernst: Ontologien als Bindeglied bei der Personalentwicklung. In: *34. Jahrestagung der Gesellschaft für Informatik, Workshop: Skill Management - Potenziale, Konzepte, Werkzeuge.* Ulm, September 2004. – URL: http://www.fzi.de/KCMS/kcms_file.php?action=link&id=426, gesehen am 14.01.2009
- [Bie06] BIESALSKI, Ernst: *Unterstützung der Personalentwicklung mit ontologiebasiertem Kompetenzmanagement*, Universität Frediriana zu Karlsruhe, Diss., 2006. – präsentiert: 4. Symposium für Wissensmanagement in Karlsruhe, 12.10.2006

- [BKKB05] BERNSTEIN, Abraham ; KAUFMANN, Esther ; KIEFER, Christoph ; BÜRKI, Christoph: SimPack: A Generic Java Library for Similiarity Measures in Ontologies / Department of Informatics, University of Zurich. 2005. – Forschungsbericht
- [BL01] BERNERS-LEE, Tim: The Semantic Web - A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. In: *Scientific American Magazine* (2001), Mai. – URL: <http://www.sciam.com/article.cfm?id=the-semantic-web>, gesehen am 28.01.2009
- [DL00] DEITERS, Wolfgang ; LUCAS, Reinhard: Skill-Management: ein Baustein für das Management flexibler Teams. In: *Information Management and Consulting* 15 (2000), Nr. 3, S. 54–60
- [Dru06] DRUCKER, Peter F.: *The Practice of Management*. Reissue. HarperCollins, 2006
- [DS08] DICKINSON, Ian J. ; SEABORNE, Andy: *Jena - A Semantic Web Framework for Java*. 2008. – <http://jena.sourceforge.net/index.html>, gesehen am 24.02.2009
- [FF07] FRÖMING, Jane ; FÜRSTENAU, Daniel: KMDL v2.1 - Eine semiformale Beschreibungssprache zur Modellierung von Wissenskonversionen / Universität Potsdam. 2007. – Forschungsbericht
- [FK03] FUCHS-KITTOWSKI, Frank: Wissensmanagement und E-Collaboration - Ein Framework für Communities, Teams und Netze zur Unterstützung kooperativer Wissensarbeit? / Fraunhofer ISST Berlin. 2003. – Forschungsbericht
- [Fox92] FOX, Mark S.: The TOVE Project Towards a Common-Sense Model of the Enterprise. In: *IEA/AIE '92: Proceedings of the 5th international conference on Industrial and engineering applications of artificial intelligence and expert systems*. London, UK : Springer-Verlag, 1992, S. 25–34
- [FSRG06] FRÖMING, Jane ; SCHMIDT, Simone ; RÜSSBÜLDT, Uwe ; GRONAU, Norbert: Anforderungs- und wissensorientierte Zusammenstellung von Projektteams. In: *Industrie Management* (2006), Nr. 22, S. 12–14

- [Gro06] GRONAU, Norbert: *Softwaregestütztes Staffing zur kompetenzorientierten Zusammenstellung von Projektteams*. Vortragspräsentation auf der Fachtagung Kompetenzmanagement in Potsdam, Oktober 2006
- [Gru93] GRUBER, T.R.: Toward the principles for the Design of Ontologies Used for Knowledge Sharing. In: *International Journal of Human and Computer Studies* 43 (1993), S. 907–928
- [Hac87] Kapitel in: Handbook of organizational behaviour. In: HACKMAN, Richard J.: *The design of work team*. Jay W. Lorsch, Eaglewood Cliffs, New Jersey, 1987, S. 315–342
- [Her00] HERBST, Dieter: *Erfolgsfaktor Wissensmanagement*. Cornelsen, 2000
- [Hes02] HESSE, Wolfgang: Ontologie(n). In: *Informatik-Spektrum* 25 (2002), Nr. 6, S. 477–480
- [HH05] HIERMANN, Wolfgang ; HÖFFERER, Max: *Skill Management: Searching Highly Skilled Employees for Teambuilding and Projetc Managament Tasks*. Graz (Austria), Juni 2005. – URL: http://i-know.tugraz.at/blog/wp-content/uploads/2008/11/6_skill-management.pdf, gesehen am 01.Februar 2009
- [HKRS08] HITZLER, Pascal ; KRÖTZSCH, Markus ; RUDOLPH, Sebastian ; SURE, York: *Semantic Web. Grundlagen*. Springer, 2008
- [Hos04] HOSCHEK, Wolfgang: *Colt - Open Source Libraries for High Performance Scientific and Technical Computing in Java*. 2004. – <http://acs.lbl.gov/~hoschek/colt/index.html>, gesehen am 24.02.2009
- [Hub99] HUBER, Hans-Dieter: *Interkontextualität und künstlerische Kompetenz. Eine kritische Auseinandersetzung*. Vortrag auf der 7. Internationalen Performance-Konferenz, Kunsthaus Glarus, März 1999
- [JMW95] Kapitel 7. In: JACKSON, Susan E. ; MAY, Karen E. ; WHITNEY, Kristina: *Understanding the dynamics of diversity in decision-making teams*. Jossey-Bass, San Francisco : R. A. Guzzo, E. Salas: Team Effectiveness and Decision Making in Organizations, 1995, S. 204–261
- [Joh06] JOHN, Michael: Semantische Technologien in der betrieblichen Anwendung. In: *Ergebnisse einer Anwenderstudie, Technical Report FIRST*

- Fraunhofer Institut Rechnerarchitektur und Softwaretechnik, 2006. – URL: <http://publica.fraunhofer.de/eprints/urn:nbn:de:0011-n-534743.pdf>, gesehen am 20.11.2008
- [JRK93] JON R. KATZENBACH, Douglas K. S.: *The Wisdoms of Teams*. Boston : Havard Business School Press, 1993
- [JVM07] JARRAR, Mustafa ; VERVENNE, Luk ; MAYNARD, Diana: HR-Semantics Roadmap - The Semantic challenges and opportunities in the Human Resources domain / OOA Ontology Outreach Advisory, Knowledge-Web Initiative. 2007. – Forschungsbericht. – Ergebnis des OOA-HR Kick-off Workshops am 11.10.2006
- [Köc08] KÖCHER, Prof. Dr. R.: ACTA 2008: *Veränderungen der Informations- und Kommunikationskultur*. Präsentation in München, Institut für Demoskopie Allensbach, Oktober 2008. – http://www.acta-online.de/praesentationen/acta_2008/acta_2008_Information%2390EDC.pdf, gesehen am 03.11.2008
- [KS07] KUNZMANN, Christine ; SCHMIDT, Andreas: Kompetenzorientierte Personalentwicklung: Auf dem Wege zum Lernen bei Bedarf. In: *ERP Management* 2007 (2007), Nr. 1, S. 38–41. – URL: http://www.andreas-p-schmidt.de/publications/kunzmann_schmidt_ERP_1-07.pdf, gesehen am 14.01.2009
- [Kuh99] KUHLMANN, Michael: *Guide für IT-basierte Großprojekte*. 1999. – Dokument für die interne Nutzung
- [Lin06] LINDEMANN, Udo: *Methodische Entwicklung technischer Produkte*. 2. bearbeitete Auflage. Springer, 2006
- [LK06] LITKE, Hans-D. ; KUNOW, Ilonka: *Projektmanagement*. 5. Auflage. Haufe Verlag, 2006
- [Mar07] MARTENS, Heiko: Fischen im Pool. In: *SPIEGEL SPECIAL: Was studieren?* (2007), Nr. 2, S. 134–136. – URL: <http://www.spiegel.de/spiegelspecial/0,1518,479197,00.html> gesehen am 06.11.2008
- [MB91] MAGJUKA, Richard J. ; BALDWIN, Timothy T.: Team-based employee involvement programs: effects of design and administration. In: *Personnel Psychology* 44 (1991), Nr. 4, S. 793–812

- [MJW07] MOCHOL, Małgorzata ; JENTZSCH, Anja ; WACHE, Holger: Suitable employees wanted? Find them with semantic techniques / Freie Universität Berlin, University of Applied Sciences Northwestern Switzerland (FHNW). 2007. – Forschungsbericht. – URL: <http://wissensnetze.ag-nbi.de/publ/mswfb2007.pdf>, gesehen am 14.01.2009
- [Moh95] MOHRMAN, Susan A.: *Designing Team-based Organizations*. Jossey Bass Publishers, San Francisco, 1995
- [Oel07] OELSNITZ, Dietrich von d.: Kompetenzsteuerung in Teams durch transaktives Wissen. In: *Jahrbuch Strategisches Kompetenzmanagement*. Rainer Hampp Verlag, 2007 (1 - Dynamische Theorien der Kompetenzentstehung und Kompetenzverwertung im strategischen Kontext), S. 113–146
- [PPH06] PODGORELEC ; PAVLIC ; HERICKO: Using Semantic Web Technologies for Project Team Building. 2006. – Forschungsbericht. – vorgetragen auf der KMO 2006 (Knowledge Management in Organizations)
- [RÖ6] RÜSSBÜLDT, Uwe: *Staffing - Konzeption und prototypische Implementierung einer Teambildungskomponente*. 2006
- [Res99] RESNIK, Philip: In: *Journal of Artificial Intelligence Research* 11 (1999), 95–130 S.
- [Sch05] SCHMIDT, Andreas: Bridging the Gap between Knowledge Management and E-Learning with Context-Aware Corporate Learning. In: *Lecture Notes in Artificial Intelligence (LNAA)*, Springer 3782 (2005), April, S. 203–213
- [SH08] STOCK-HOMBURG, Ruth: *Personalmanagement: Theorien - Konzepte - Instrumente*. Gabler Verlag, 2008
- [SM99] STADER, Jussi ; MACINTOSH, Ann: Capability Modeling and Knowledge Management / Artificial Intelligence Applications Institute, Division of Informatics, The University of Edinburgh. 1999. – Forschungsbericht
- [SS91] STEINMANN, Horst ; SCHREYÖGG, Georg: *Management: Grundlagen der Unternehmensführung*. Gabler Verlag, 1991
- [Sun99] SUN: *Code Conventions for the Java Programming Language*. 1999. – URL: <http://java.sun.com/docs/codeconv/>

- [SW01] SMITH, Barry ; WELTY, Christopher: *Ontology: Towards a New Synthesis*. Oktober 2001. – URL: <http://www.cs.vassar.edu/~weltyc/papers/fois-intro.pdf>, gesehen am 29.01.2009
- [TKM05] TERZIEV, Ivan ; KIRYAKOV, Atanas ; MANOV, Dimitar: D1.8.1 Base upper-level ontology (BULO) Guidance / Ontotext Lab, Sirma Group. 2005. – Forschungsbericht. – URL: http://proton.semanticweb.org/D1_8_1.pdf, gesehen am 29.01.2009
- [TM06] TOLKSDORF, Robert ; MOCHOL, Małgorzata: Semantic-Web-Technologien im Arbeitsvermittlungsprozess. In: *Wirtschaftsinformatik* 48 (2006), Januar, Nr. 1, S. 17–26
- [Ver03] VERNON, Rachel N.: *Similarity vs. Diversity: The Impact of Personality Congruence on Teams*. 18. Jährliche Konferenz der Society for Industrial and Organizational Psychology, Orlando, Florida, 2003
- [Wüb01] Kapitel 4 Manager Competence Audit. In: WÜBBELMANN, Klaus: *Management Audit*. Gabler Verlag, 2001
- [Wes03] WEST, Michael A.: *Effective Teamwork; Practical Lessons from Organizational Research*. Second Edition. Blackwell Publishers, 2003
- [WP94] WU, Zhibiao ; PALMER, Martha: Verb semantics and lexical selection. In: *32nd. Annual Meeting of the Association for Computational Linguistics*. New Mexico State University, Las Cruces, New Mexico, 1994, S. 133–138
- [Zel04] ZELEWSKI, Stephan: Kooperatives Wissensmanagement in Engineering-Netzwerken (KOWIEN). In: *Industrieerprobte Lösungen und Werkzeuge für Produktentwicklung, Engineering und Kompetenzmanagement*, 2004, S. 21–83. – Abschlussworkshop der Verbundprojekte GINA, KOEFFIZIENT und KOWIEN

Glossar

Begriff	Bedeutung
ACTA	Allensbacher Computer- und Technik-Analyse
AIAI	Artificial Intelligence Application Institute
API	Application Programming Interface, Programmierschnittstelle
BKZ	Berufskennziffer
CSCW	Computer Supported Cooperative Work
DIN	Deutsches Institut für Normung e.V.
E-Skills	Menge der Fähigkeiten eines Mitarbeiters (angewendet bei den Algorithmen dieser Arbeit)
GUI	Graphical User Interface, grafische Benutzeroberfläche
HR-Ontologie	Human Resource-Ontologie, wurde für diese Diplomarbeit von der Ontonym GmbH zur Verfügung gestellt
HRDW	Human Resource Data Warehouse
HRM	Human Resource Management
HTML	Hypertext Markup Language
infonea	information network architecture

Begriff	Bedeutung
KMDL	Knowledge Modeling and Description Language, entwickelt am Lehrstuhl Wirtschaftsinformatik und Electronic Government der Universität Potsdam
Kohäsion	Zusammenhalt (hier: in einem Team)
KOWIEN	Kooperatives Wissensmanagement in Engineering-Netzwerken
MIT	Massachusetts Institute of Technology
MRCA	Most Recent Common Ancestor, gemeinsamer Elternknoten zweier Knoten, beispielsweise in einer Ontologie
NAICS	North American Industry Classification System
OWL	Web Ontology Language
P-Skills	Menge der Fähigkeiten, die für die Arbeit im Projekt benötigt werden (angewendet bei den Algorithmen dieser Arbeit)
RDF	Resource Description Framework
SEKT	Semantically Enabled Knowledge Technologies
SOC	Standard Occupational Classification
Teamrecruiter	Person innerhalb eines Unternehmens, die mit der Zusammensetzung von Teams beschäftigt ist
UML	Unified Modeling Language, Sprache zur Modellierung von Anwendungen und Systemen
XML	extensible Markup Language

A. Inhalt der CD

Prototypische Applikation

- Die im Rahmen der Arbeit entwickelten Anwendungen.
- Testdaten, Testdatengenerator, Auswertung

Schriftliche Diplomarbeit

- Die schriftliche Ausarbeitung im PDF-Format
- Alle Abbildungen der Ausarbeitung in hoher Qualität als PDF.

Dokumentation

- UML-Diagramme für Enterprise Architect
- Javadoc-Dokumentation der Sourcen