

February 2007

Kontextbasiertes Information Retrieval zur Unterstützung von Entwicklungsprozessen

Andreas Henrich

Otto-Friedrich-Universität Bamberg, andreas.henrich@wiai.uni-bamberg.de

Karlheinz Morgenroth

Otto-Friedrich-Universität Bamberg, karlheinz.morgenroth@wiai.uni-bamberg.de

Follow this and additional works at: <http://aisel.aisnet.org/wi2007>

Recommended Citation

Henrich, Andreas and Morgenroth, Karlheinz, "Kontextbasiertes Information Retrieval zur Unterstützung von Entwicklungsprozessen" (2007). *Wirtschaftsinformatik Proceedings 2007*. 89.

<http://aisel.aisnet.org/wi2007/89>

This material is brought to you by the Wirtschaftsinformatik at AIS Electronic Library (AISeL). It has been accepted for inclusion in Wirtschaftsinformatik Proceedings 2007 by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

In: Oberweis, Andreas, u.a. (Hg.) 2007. *eOrganisation: Service-, Prozess-, Market-Engineering*; 8. Internationale Tagung Wirtschaftsinformatik 2007. Karlsruhe: Universitätsverlag Karlsruhe

ISBN: 978-3-86644-094-4 (Band 1)

ISBN: 978-3-86644-095-1 (Band 2)

ISBN: 978-3-86644-093-7 (set)

© Universitätsverlag Karlsruhe 2007

Kontextbasiertes Information Retrieval

zur Unterstützung von Entwicklungsprozessen

Andreas Henrich, Karlheinz Morgenroth

Lehrstuhl für Medieninformatik
Fakultät Wirtschaftsinformatik und Angewandte Informatik
Otto-Friedrich-Universität Bamberg
96045 Bamberg
{andreas.henrich|karlheinz.morgenroth}@wiai.uni-bamberg.de

Abstract

Mitarbeitern die richtige Information zum richtigen Zeitpunkt zugänglich zu machen, stellt heute einen entscheidenden Wettbewerbsfaktor dar. Auch wenn in den vergangenen Jahren durch Maßnahmen und Systeme aus dem Bereich des Wissensmanagements in Organisationen umfangreiche Wissenssammlungen angelegt worden sind oder auch nur durch den Einsatz entsprechender IT-Systeme mehr Information in digitaler Form vorliegt, so wird die Identifikation von relevanter Information für die einzelne Person immer schwieriger und zeitaufwändiger.

Der von uns präsentierte Ansatz erfasst, beschreibt und nutzt historische und aktuelle Kontextinformation sowohl über Mitarbeiter als auch über die Entstehung von Dokumenten, um zum einen manuelle Suchanfragen von Mitarbeitern zu präzisieren und zum anderen das Entstehen von Informationsbedürfnissen bei Mitarbeitern automatisch erkennen und mit geeigneten Suchanfragen selbstständig beantworten zu können. Unser Ansatz und dessen Implementierungen decken dabei insbesondere ingenieurmäßige Produktentwicklungsprozesse ab.

1 Motivation

Mit der zunehmenden Menge an verfügbarer Information wird die Identifizierung relevanter Information für die einzelne Person immer schwieriger. Für die einzelnen Mitarbeiter reicht die in einer Organisation technisch abrufbare Information von eher allgemeinen Inhalten über neue Themen und Erkenntnisse im eigenen oder in angrenzenden Fachgebieten, die zur Weiter- und Fortbildung dienen können, über die Anwendung von und Erfahrungen mit Vorgehensmodel-

len, Methoden und Werkzeugen, bis zu konkreten Teilergebnissen oder Komponenten, die direkt in einem aktuell bearbeiteten Projekt Verwendung finden können. Die Anwendung bewährter Vorgehen, aber auch die Wiederverwendung von Komponenten stellt einen Faktor dar, der zur Steigerung der Effizienz und Produktivität innerhalb eines Entwicklungsprozesses – bspw. im Software Engineering oder Maschinenbau – und zu einer höheren Qualität bei den entwickelten Produkten führen kann. Dies bedeutet, dass der Identifikation von relevanten Inhalten, die zu einer Wiederverwendung herangezogen werden können, ein erheblicher Stellenwert zukommt.

Trotz des Einsatzes von Suchmaschinen sind die gefundenen Ergebnisse für eine anfragende Person oft nicht zielführend und erschöpfend im Hinblick auf das bestehende Informationsbedürfnis. Gründe hierfür können in einem zu unscharf artikulierten Informationsbedürfnis oder schlichtweg in der Wahl falscher Suchbegriffe liegen. In der Konsequenz unterlassen Anwender häufig die Durchführung einer Suche wegen des damit verbundenen Aufwands und der ungewissen bzw. in der Vergangenheit unbefriedigenden Qualität der Ergebnisse (vgl. z. B. [Wils99]).

Einen Ansatz die Ergebnisqualität zu erhöhen, bildet die Einbeziehung des Kontextes des Anfragestellers. Dabei werden neben seinen aktuellen und zurückliegenden Tätigkeiten und deren Inhalten unter anderem auch persönliche Kenntnisse erfasst und zur Verfeinerung von manuell gestellten Anfragen verwendet. Darüber hinaus kann der Kontext auch dazu verwendet werden, auf ein latent vorliegendes Informationsbedürfnis zu schließen, automatisch Suchanfragen zu stellen und die Suchergebnisse dem Anwender proaktiv zu präsentieren. Auf der anderen Seite kann u. a. der Aufgabenkontext, in dem bestimmte Dokumente erstellt worden sind, sowie deren Bearbeiter erfasst werden, um neue Beschreibungs- und Suchkriterien für diese Dokumente zu erhalten.

In Abbildung 1 ist ein Szenario dargestellt, in dem ein Entwickler innerhalb eines laufenden Entwicklungsprojektes mit der Arbeit an einem Gesamtentwurf beginnt. Die Grundlage für den Gesamtentwurf sind Anforderungslisten, prinzipielle Lösungen sowie ein Vorentwurf. Um nun für den aktuellen Gesamtentwurf Elemente existierender Vorentwürfe oder ganze Teilkomponenten wieder verwenden zu können, muss der Entwickler Kenntnis über entsprechende Artefakte entweder in früheren Iterationen des eigenen Projektes oder anderen Projekten haben. Diese Kenntnis könnte bei einer Person durch eine ständige Einbeziehung in andere Projekte geschaffen werden. Diesem Vorgehen sind aber offensichtlich zeitliche und organisatorische

Grenzen gesetzt. Eine Alternative bildet der Einsatz eines kontextbasierten Information Retrieval Systems, das eine Person – in diesem Fall einen Entwicklungsingenieur – mit der für ihre aktuell vorliegende Aufgabe benötigten Information versorgt.

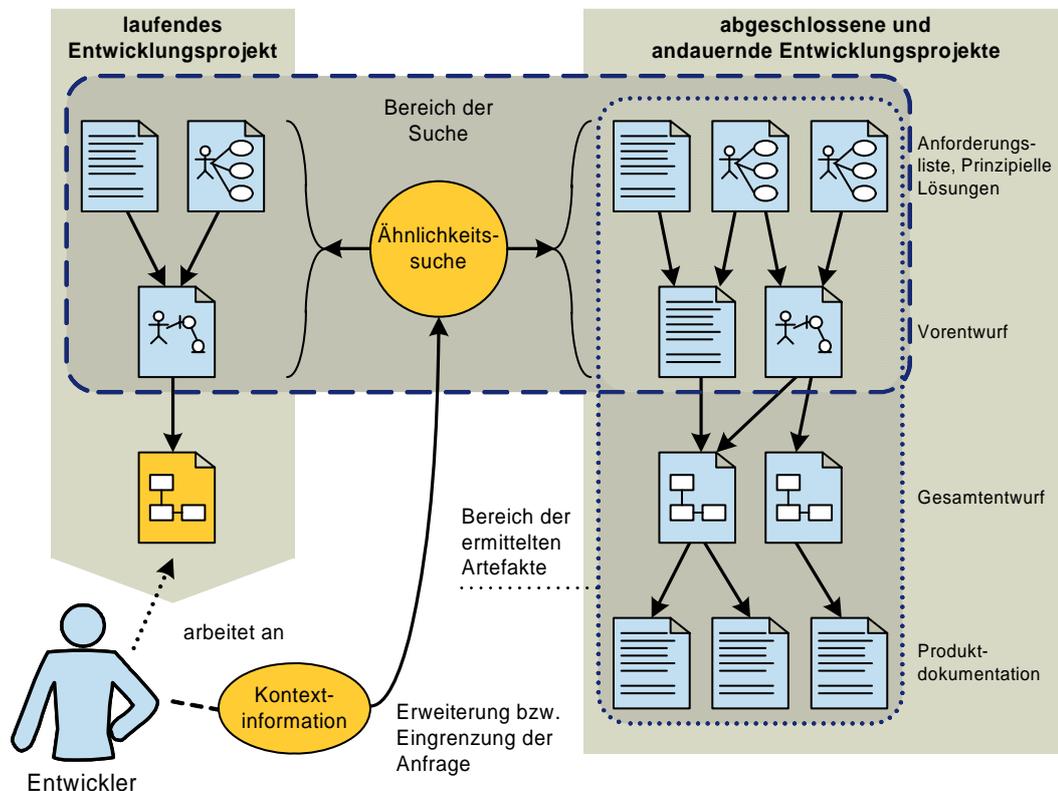


Abb. 1: Szenario eines laufenden Entwicklungsprojektes (in Anlehnung an VDI 2221 [VDI93]) mit einem Entwickler, der aktuell einen Gesamtentwurf beginnt

Auch wenn Entwicklungsprozesse, wie z. B. VDI 2221 [VDI93] im Maschinenbau oder ein Rational Unified Process [Kruc04] im Software Engineering, zunächst eine starke Strukturierung aufweisen mögen, so bieten sie in den einzelnen Teilaufgaben den beteiligten Personen eine sehr hohe Entscheidungsvielfalt in der Art und Weise wie die gestellten Aufgaben im Detail auszuführen sind und welche Werkzeuge oder Materialien dazu zum Einsatz kommen sollen. Gleichzeitig lässt sich ein Erfolg oder Misserfolg dieser Entscheidungen erst in folgenden Aufgaben und Phasen beurteilen.

2 Verwandte Ansätze

In der Vergangenheit wurde eine Reihe von Forschungsansätzen präsentiert, die sich sowohl mit der Erfassung und Modellierung des Kontextes von Individuen als auch mit der automatischen Belieferung von Personen mit zu deren aktueller Tätigkeit relevanter Information beschäftigten.

Über den folgenden kurzen Abriss hinausgehend ist eine ausführliche Würdigung von themenverwandten Forschungsansätzen bei [Morg06] zu finden.

Eine Gruppe verwandter Ansätze bilden Systeme, die den Text eines aktuell von einer Person bearbeiteten Dokumentes zur Formulierung von Anfragen verwenden (vgl. [RhSt96], [BuKr99], [DCSH04]). Gesucht werden dabei relevante Dokumente, wobei der betrachtete Kontext sich aber im Wesentlichen über den aktuell bearbeiteten Text definiert. Andere Ansätze beobachten einen Benutzer beim „Surfen“ im Web und geben Empfehlungen zu thematisch als ähnlich erachteten Webseiten (vgl. [Libe95], [JoFr97], [PaMB96]). Darüber hinaus kommen Methoden des Collaborative Filtering sowohl in Empfehlungssystemen für Texte als auch für Produkte im E-Commerce zum Einsatz (vgl. [RIS+94], [ScKR01], [ShMa95]). Eine Integration entsprechender kontextbasierter Suchdienste zeigen bspw. [Meie00] für Managementinformationssysteme und [PPK03] für Portalsysteme.

Im Forschungsgebiet des Workflow-Management verfolgte das Forschungsprojekt KnowMore sowie das sich daran anschließende DECOR-Projekt einen partiell ähnlichen Ansatz [ABH+00, ABN+01]. Dabei wurden allerdings existierende Workflow-Schemata zusätzlich um weitere Metadaten ergänzt, die u. a. für jede Aufgabe eine generische Anfrage enthielten [Maus01]. Im Unterschied zu unserem im Folgenden skizzierten Ansatz wurden die Anfragen für ausgewählte und vordefinierte Aufgaben also a priori ausformuliert. Zum Ausführungszeitpunkt einer Aufgabe wurden diese generischen Anfragen mit Instanzdaten gefüllt und zur Ausführung gegenüber einer Reihe von so genannten Suchagenten gebracht. Ebenfalls an einer starren Prozessmodellierung orientiert sich das PreBIS-Projekt [BöHä05].

Unser Ansatz hebt sich von den genannten Ansätzen durch eine umfassendere Betrachtung des Kontextes und durch ein wesentlich mächtigeres Konzept zur Relevanzbewertung sowie durch den Bezug auf den Produktentwicklungsprozess ab. Darüber hinaus findet sowohl eine Unterstützung von stark strukturierten Prozessrahmen als auch von eher unstrukturierten Einzeltätigkeiten statt.

3 Der COBAIR-Ansatz

Mit dem COBAIR-Ansatz [Morg06] wird ein grundlegender methodischer Ansatz zur Nutzung des Kontextes von Personen und deren Tätigkeiten für die Suche nach relevanter Information vorgestellt. Darüber hinaus wird ebenso der Kontext von Dokumenten, bzw. Artefakten i. A.

betrachtet, in dem diese entstanden sind. Der COBAIR-Ansatz umfasst dabei die Erfassung und Beschreibung von Kontextinformation, die Strukturierung und gleichzeitige Vernetzung der erfassten Kontextinformation, die Ableitung von Informationsbedürfnissen als auch die Präzisierung und Durchführung von Ähnlichkeitssuchen, teilweise oder gänzlich auf Basis der Kontextinformation.

3.1 Erfassung und Beschreibung des Kontextes

Die Ausgangsbasis von Suchvorgängen von Personen ist ein mehr oder minder stark ausgeprägtes Informationsbedürfnis. Dieses lässt sich jedoch nicht direkt beobachten, vielmehr können nur die aus einem Informationsbedürfnis motivierten Handlungen einer Person beobachtet werden. Gleichzeitig ist ein Informationsbedürfnis durch Eigenschaften einer Person selbst, ihrer Umgebung sowie die aktuellen Aufgaben und ausgeführten Tätigkeiten – allgemein den Kontext einer Person – motiviert. Zu diesen Eigenschaften zählen unter anderem psychologische, kognitive und affektive Faktoren, Erfahrungen und Kenntnisse einer Person, aktuelle wie auch vergangene Aufgaben und Tätigkeiten einer Person, die Einordnung einer Person in eine Organisation, das Arbeitsumfeld sowie die politische, ökonomische und technische Umgebung. Wie bei anderen Ansätzen und Systemen (vgl. Abschnitt 2) wird auch hier implizit von der Annahme ausgegangen, dass eine entsprechende Person zur Verrichtung ihrer Tätigkeit, in deren Rahmen bestimmte Informationsbedürfnisse auftreten können, Softwaresysteme (z. B. CAD-, Office-, Groupware-Systeme) einsetzt, die zur Ableitung von Informationen über die aktuell verrichtete Tätigkeit in der Lage sind bzw. diesbezüglich angepasst werden können.

Der Kontext eines Anwenders während der Ausübung seiner Tätigkeit mit Hilfe eines Softwaresystems lässt sich daher zum einen aus der von ihm vollführten Benutzerinteraktion mit dem Softwaresystem ableiten. Zum anderen umfasst der Kontext auch die zu einer Aktion gehörenden Vor- und Ergebnisdokumente.

Für die Erfassung von Informationen über aktuelle und zurückliegende Kontexte von Personen dienen verschiedenste Softwaresysteme als Quelle. Im Folgenden werden diese nach ihren technischen Architekturmerkmalen unterschieden und Beispiele für zu erfassende Kontextinformation gegeben.

- *Klassische Client-Anwendungen.* In diese Kategorie fallen neben Office-Programmen (inkl. E-Mail und Terminverwaltung) auch CAD-Anwendungen, die auf dem Arbeitsplatzrechner eines Anwenders laufen. Derartige Client-Anwendungen können heute durchweg mittels so

genannter Plug-Ins um Funktionalitäten erweitert werden. Auf diesem Weg ist sowohl ein Zugriff auf die aktuell von einem Anwender bearbeiteten Daten möglich (z. B. vorliegende prinzipielle Lösungen oder einen betrachteten Vorentwurf) als auch die Erfassung der Aktionen, die eine Person ausführt.

- *Client-Server-Anwendungen.* Unter Client-Server-Anwendungen fallen bspw. Anwendungen wie Groupware-Systeme (z. B. Lotus Notes), Werkzeuge zur Projektplanung (z. B. Microsoft Project) oder PDM-Systeme, teilweise in direkter Verbindung mit CAD-Systemen. Die eigentliche Anwendung setzt sich aus mindestens zwei Teilen zusammen, die auf dem einzelnen Arbeitsplatzrechner (Client) und auf einem zentralen Server laufen und miteinander kommunizieren. Der Erweiterbarkeit dieses Typus von Softwaresystemen kommt die Tatsache entgegen, dass zunehmend so genannte offene Architekturen zum Einsatz kommen. Je nach eingesetzter Technologie und Architektur der Implementierung können derartige Anwendungen sowohl auf der Client- als auch der Server-Seite durch Plug-Ins ergänzt werden.
- *Server-Anwendungen.* In diese Kategorie fallen z. B. unternehmensweite Verzeichnisdienste (X.500, Active Directory, LDAP), denen sich Kontextinformation über die organisatorische Eingliederung einer Person in einem Unternehmen sowie deren Rollen und Aufgaben entnehmen lässt. Weitere Quellen stellen abteilungs- oder unternehmensweite Dokumenten-Management-Systeme (DMS), Content-Management-Systeme (CMS) oder Repositories und PDM-Systeme dar. Diese Systeme erfassen neben den eigentlichen Nutzdaten automatisch auch eine Reihe von Metadaten, wie z. B. Autor und Zeitpunkt einer Änderung an einem Dokument. Für die Erfassung von Kenntnissen, Erfahrungen, Interessen und auch von persönlichen Präferenzen lassen sich entweder spezialisierte Skill-Management-Werkzeuge oder erweiterte Human-Resource-Systeme einsetzen.

Sowohl die für eine Ableitung von Kontextinformation in Betracht kommenden Systeme als auch deren Inhalte sind dabei sehr heterogen. Für eine einheitliche und eindeutige Erfassung, Benennung und Kennzeichnung der Inhalte kommen daher in unserem Ansatz verschiedene Technologien des Semantic Web [Bern98] zum Einsatz.

Für die Beschreibung der erfassten Kontextinformation wird das Resource Description Framework (RDF, [KICa04]) verbunden mit Ontologien auf Basis der Web Ontology Language (OWL, [DeSc04]) genutzt. Der Kontext einer Person wird somit in Form einer automatisch ge-

wonnenen Annotation mit Metadaten beschrieben, während zur eindeutigen Bezeichnung der Inhalte der Kontextbeschreibung verschiedene in Ontologien definierte Konzepte Verwendung finden.

Um die resultierende Fülle an verfügbarer Information sinnvoll für die Erschließung eines Informationsbedürfnisses verwenden zu können, gilt es diese zunächst anhand eines Modells zu gliedern. Abbildung 2 zeigt einen Ansatz für ein einheitliches Nutzermodell, das die verschiedenen Kontextinhalte in einzelne Dimensionen einteilt und darüber hinaus auch deren Veränderung berücksichtigt. Gleichzeitig findet aber eine Vernetzung der einzelnen Kontextinhalte statt [HeMo01, HeMo03].

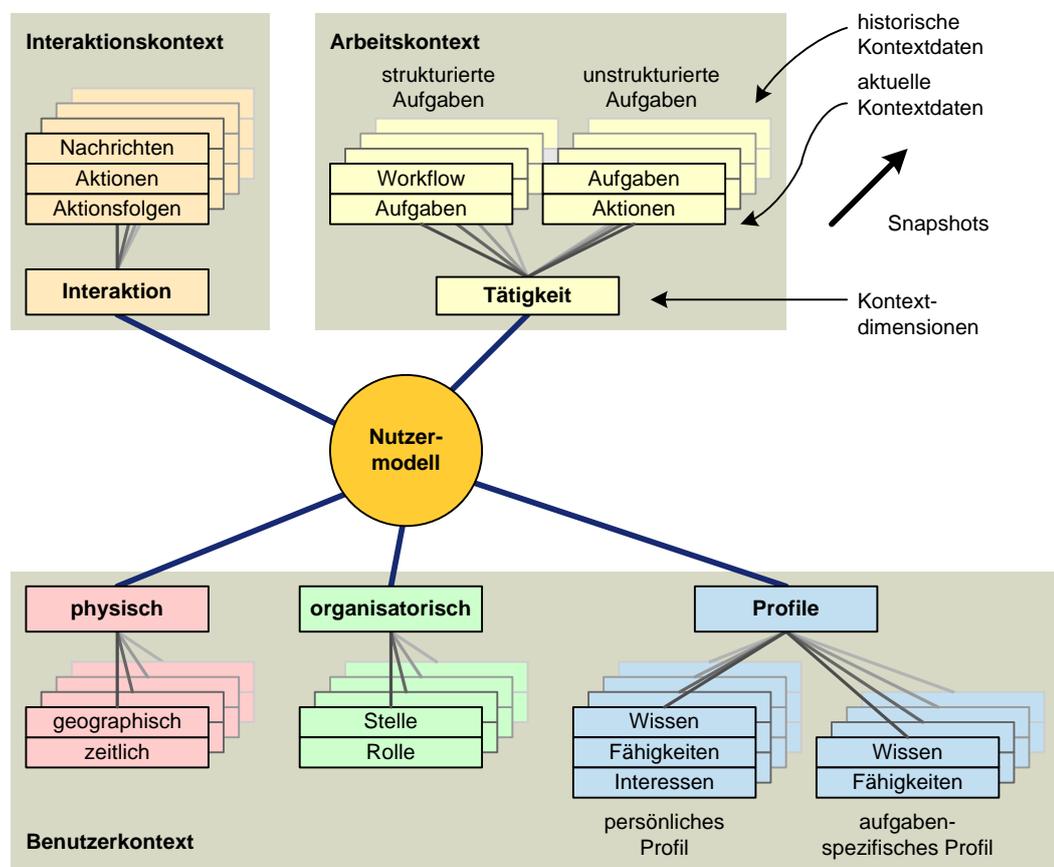


Abb. 2: Einheitliches Nutzermodell

Dieses einheitliche Nutzermodell differenziert grundsätzlich zwischen Benutzer-, Arbeits- und Interaktionskontext:

- Der *Benutzerkontext* umfasst sowohl den physikalischen und organisatorischen Kontext als auch das Profil des Benutzers. Diese einzelnen Aspekte werden im Folgenden als Kontextdimensionen bezeichnet.

Die physikalische Kontextdimension beinhaltet beispielsweise die geographischen Daten

über den Arbeitsplatz eines Mitarbeiters.

Die organisatorische Kontextdimension beinhaltet sowohl die Stelle, die eine Person innerhalb der Organisation einnimmt, als auch die einzelnen Rollen, die sie ausfüllt. Dies schließt eine Stellenbeschreibung sowie die erwarteten Fähigkeiten und das Wissen eines entsprechenden Stelleninhabers mit ein. Darüber hinaus enthält diese Kontextdimension Informationen über die einzelnen Organisationseinheiten selbst, also beispielsweise über die Abteilung, in der ein Mitarbeiter tätig ist.

Die Profildimension enthält Informationen über die Wissensgebiete und Fähigkeiten einer Person. Im Modell wird zwischen dem persönlichen Profil, das von einer Person selbst vorgegeben werden kann und beispielsweise persönliche Präferenzen und Interessen enthält, und einem aufgabenspezifischen Profil unterschieden, das sich aus den Aufgaben und Tätigkeiten bestimmt, die eine Person in der Vergangenheit ausgeführt hat.

Diese verschiedenen Teildimensionen des Benutzerkontexts haben gemein, dass sie sich über die Zeit nur sehr langsam verändern.

- Der *Arbeitskontext* charakterisiert die aktuelle Tätigkeit, die eine Person ausführt. Dabei lässt sich diese Tätigkeit durch die wohl definierten Aufgaben in einem Workflow oder durch die ausgeführten Aufgaben und Aktionen beschreiben, welche eine unstrukturierte Tätigkeit mit sich bringt. Im ersten Fall wird eine konkrete Beschreibung der einzelnen Aufgaben oder der gesamten Tätigkeit vorliegen, während im zweiten Fall typischerweise nur vage Informationen über die Tätigkeit verfügbar sind.

Die technische Sicht auf den Arbeitskontext schließt die aktuell verwendeten Module eines Anwendungssystems mit ein, die eine Person zur Ausübung ihrer Tätigkeit verwendet. Der Arbeitskontext einer betrachteten Person wird sich deutlich häufiger ändern als dies vom Benutzerkontext zu erwarten ist.

- Die aktuellen Interaktionen eines Benutzers mit den von ihm eingesetzten Anwendungssystemen spiegelt der *Interaktionskontext* wider. Diese Interaktionen werden durch Nachrichten von Menübefehlen oder Dialogen repräsentiert, die ein Benutzer während der Arbeit mit einem Anwendungssystem ausführt bzw. verwendet, beispielsweise das Öffnen eines Vorentwurfs in einem CAD-System. Der Interaktionskontext ändert sich damit dynamisch innerhalb von Minuten oder Sekunden.

Neben den aktuellen Werten der einzelnen Kontextdimensionen ist auch deren Veränderung über die Zeit für die Bestimmung eines Kontextes von Interesse. So werden in bestimmten Zeitabständen sog. Snapshots aufgezeichnet, deren Inhalt zudem über verschiedene Mechanismen aggregiert und verdichtet wird (siehe [Morg06]).

3.2 Ableitung eines Informationsbedürfnisses aus der erfassten Kontextinformation

Für die Ableitung eines Informationsbedürfnisses aus der erfassten Kontextinformation erfolgt zunächst die Bildung von verschiedenen Profilen. Dabei besteht die Hauptaufgabe in der Filterung und Aggregation der im einheitlichen Nutzermodell enthaltenen Kontextinformation. Die Filterung erfolgt in Bezug auf die Veränderungshäufigkeit der verschiedenen Kontextdimensionen typischerweise mittels Fensterfunktionen entlang des zeitlichen Verlaufs. Technisch werden dabei sowohl Histogramme, welche die Häufigkeit einzelner Konzepte der Ontologie abbilden, als auch gewichtete RDF-Daten erzeugt und weiterverarbeitet.

- So wird das *Profil der aktuellen Tätigkeit* aus der Kontextinformation des Interaktions- und Arbeitskontextes für einen vom aktuellen Zeitpunkt ausgehenden Zeitraum von z. B. 10 Minuten in die Vergangenheit ermittelt. Ziel ist dabei die Abbildung der aktuellen Tätigkeit einer Person.
- Das *Profil der zurückliegenden Tätigkeit* wird aus der Dimension Tätigkeit des Arbeitskontextes im Nutzermodell gebildet. Ziel dieses Profils ist es, die zurückliegende Tätigkeit einer Person und die von ihr bearbeiteten Inhalte abzubilden. Im Gegensatz zum Profil der aktuellen Tätigkeit soll mit dem Profil der zurückliegenden Tätigkeit nicht eine bestimmte ausgeführte Tätigkeit beschrieben werden, sondern vielmehr das Tätigkeitsgebiet des Entwicklers und dessen Inhalte in einer allgemeineren Form.
- Das *persönliche Profil* einer Person umfasst deren Wissen, Fähigkeiten und auch Interessen. Die Bildung eines persönlichen Profils kann entweder nur aus den manuell vorgegebenen Eigenschaften erfolgen oder aus den in zurückliegenden Tätigkeiten gezeigten Eigenschaften.
- Ein *Profil des organisatorischen Benutzerkontextes* enthält Kontextinformation aus der gleichlautenden Kontextdimension des Nutzermodells über die Stelle, die eine Person innerhalb einer Organisation innehat, bzw. die Rollen, die sie ausübt.

- Das *Profil des physischen Benutzerkontextes* enthält physische Kontextinformation aus der entsprechenden Dimension des Nutzermodells wie die geographische Lage eines Benutzers sowie auch zeitliche Angaben.

Die eigentliche Ableitung eines Informationsbedürfnisses erfolgt nun über mehrere Verarbeitungsstufen, die in Abbildung 3 dargestellt sind. Dieses einfache Modell motiviert sich u. a. durch verschiedene Modelle des menschlichen Informationsverhaltens (vgl. z. B. [Wils99]), die davon ausgehen, dass ein Profil einen mehr oder minder starken Einfluss auf das Entstehen eines zunächst latenten Informationsbedürfnisses hat. Im Gegenzug wirken die einzelnen Profile gleichzeitig hemmend auf das Entstehen eines tatsächlich wahrgenommenen Informationsbedürfnisses, das im Modell als erwartetes tatsächliches Informationsbedürfnis bezeichnet wird.

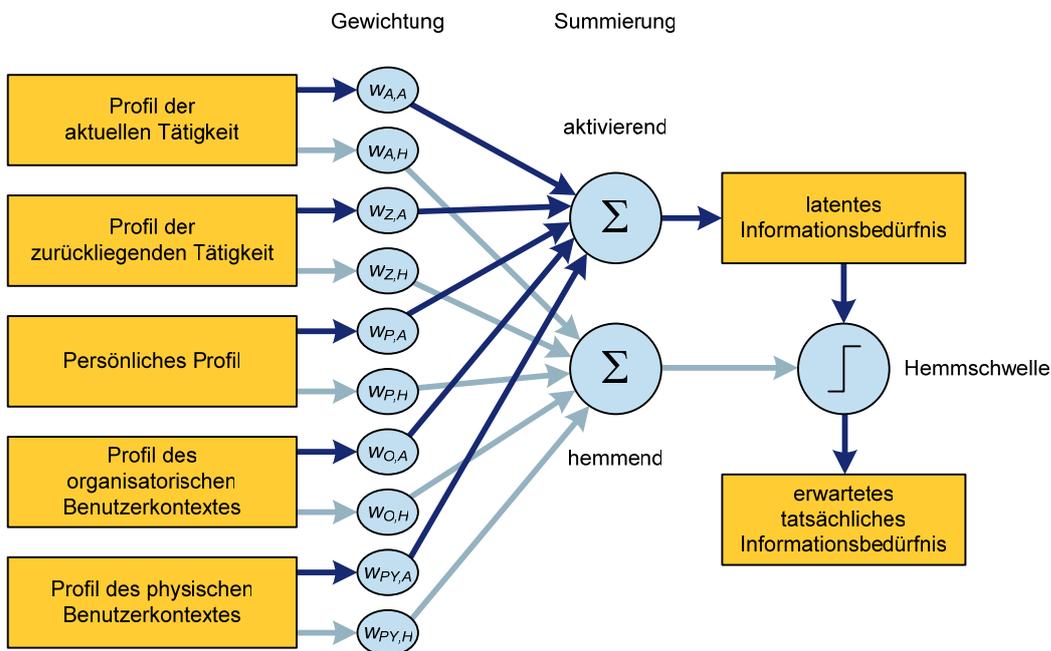


Abb. 3: Ableitung eines erwarteten tatsächlichen Informationsbedürfnisses aus den einzelnen Profilen des einheitlichen Nutzermodells (einfaches Modell)

Sehr grob gesprochen ergibt sich das Informationsbedürfnis als Abweichung des aktuellen Kontextes vom langfristigen Mittel. Einem Entwickler, der sich im Rahmen einer Analysetätigkeit z. B. mit für ihn bisher unbekanntem Legacy-Systemen beschäftigt, werden von unserem System entsprechende Dokumente über die Anbindung dieser Systeme geliefert.

Neben diesem einfachen Gewichtungmodell sind ebenfalls Regelsysteme und Systeme zur Erkennung bestimmter Handlungsweisen (z. B. mittels Hidden-Markov-Modellen) verfügbar und können alternativ eingesetzt werden (vgl. [Morg06]).

Der Inhalt des erwarteten tatsächlichen Informationsbedürfnisses kann nun zum einen zur Verfeinerung einer manuell gestellten Suchanfrage genutzt werden. Zum anderen kann bei einer systembedingten Auslösung der Hemmschwelle der Inhalt des erwarteten tatsächlichen Informationsbedürfnisses zu einer Suchanfrage umformuliert und an einen oder mehrere Suchdienste weitergeleitet werden.

3.3 Ähnlichkeitssuche mittels RDF Query by Example

Der COBAIR-Ansatz sieht sowohl die Anbindung externer Suchdienste, die z. B. im Rahmen eines existierenden Wissensmanagementsystems bereits vorhanden sind, als auch eine eigene Suchfunktionalität vor. Während typische Suchdienste auf Basis des Vektorraummodells [SWY75] arbeiten, setzt der eigene Suchdienst auf dem RDF Query by Example Ansatz [HeMo05] auf. Basis für die Präzisierung einer manuell gestellten Anfrage oder die automatische Formulierung einer Anfrage ist ein erwartetes tatsächliches Informationsbedürfnis, das in Form von gewichteten RDF-Daten vorliegt, mithin bestimmte Vernetzungen mit Gewichtungen zwischen Konzepten und externen Informationen, z. B. Dokumenten, Aufgaben oder Personen, aufzeigt. Wenngleich eine Umwandlung in eine für das Vektorraummodell adäquate Form möglich ist, so gehen dabei zahlreiche, explizite Beziehungen zwischen Konzepten verloren. Basis der in den RDF-Daten enthaltenen Konzepte sind in OWL [DeSc04] vorliegende Ontologien, die gleichzeitig den Ausgangspunkt für die Bestimmung eines Ähnlichkeitsmaßes mittels der sog. ontologischen Distanz zwischen einzelnen Konzepten darstellt.

Der RDF Query by Example Ansatz nutzt als Anfrage einen RDF-Beispieldatensatz, der z. B. im einfachsten Fall ein erwartetes tatsächliches Informationsbedürfnis repräsentiert. Gleichzeitig kommen vage Anfragebearbeitung, die Unterstützung von Volltextanfragen sowie die Ermittlung sortierter Ergebnislisten zum Einsatz. Über die Identifikation ähnlicher, aber zeitlich zurückliegender Informationsbedürfnisse anderer Personen können nicht nur deren Arbeitsergebnisse gefunden werden, sondern in der Relevanzbewertung die Bewertungen dieser Arbeitsergebnisse z.B. in Reviews mit einfließen.

RDF-Daten repräsentieren fast immer einen allgemeinen Graphen mit komplexen Strukturen. Eine Suche entspricht damit dem bekannten Problem der Subgraphisomorphie und liegt zunächst in der Komplexitätsklasse NP. Beschränkt sich die Form des Anfragegraphen auf Bäume, so erreicht die Komplexität zwar die Klasse P, bleibt aber für praktische Zwecke ungeeignet. Durch den Einsatz geeigneter Datenstrukturen, hier zweistufig verketteter invertierter Lis-

ten, und einer strombasierten Anfragebearbeitung ist dennoch eine effiziente Anfragebearbeitung möglich [HeMo05].

4 Das COBAIR-Framework

Zur Realisierung entsprechender kontextbasierter Information Retrieval Systeme geben wir neben einer Reihe von Methoden und Ansätzen auch einen Architekturrahmen an, der neben der Anbindung verschiedenster Quellen für Kontextinformation insbesondere auch die Handhabung hoher Volumina an Kontextinformation ermöglicht [HeRo00, HeRo03]. Für die effiziente Bearbeitung vager Anfragen wurde der RDF Query by Example Ansatz integriert.

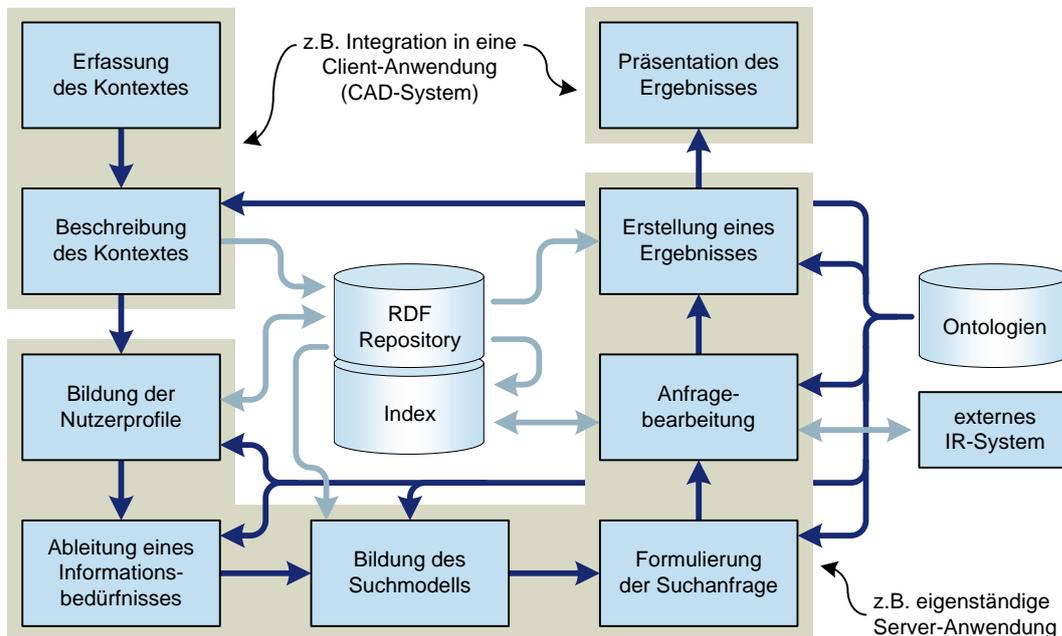


Abb. 4: Komponenten des COBAIR-Frameworks

Dieses Framework unterstützt dabei im Einzelnen die:

- Anbindung von verschiedenen Client-, Client-Server- sowie Server-Anwendungen zur Erfassung von Informationen über aktuelle und zurückliegende Kontexte von Personen,
- Beschreibung der Kontextinformation mittels RDF und Einsatz von Ontologien zur eindeutigen Benennung der einzelnen Elemente,
- Verwaltung und Abfrage von Kontextinformation sowie Gliederung der Kontextinformation in einem einheitlichen Nutzermodell, technische Speicherung in einem RDF-Repository,

- Bildung von Nutzerprofilen und Ableitung von Informationsbedürfnissen einzelner Benutzer aus deren Kontextinformation,
- Formulierung eines erwarteten tatsächlichen Informationsbedürfnisses für einen Anwender,
- Formulierung von Suchanfragen auf einem Suchindex,
- Ausführung der Suchanfragen gegenüber dem eigenen RDF-Repository oder externen IR-Systemen sowie
- Präsentation der Suchergebnisse und deren weitere Verwendung durch einen Anwender.

Das obige Framework ermöglicht mit verschiedenen Architekturmerkmalen vor allem die Verarbeitung sehr großer Mengen an Kontextinformation und gleichzeitig die Anfragebearbeitung in für die Anwender akzeptabler Zeitdauer. Die Anfragebearbeitung unterstützt dabei sowohl vage Anfragen als auch die Lieferung der nach ihrer Relevanz gegenüber der Anfrage sortierten Ergebnisse.

5 Einsatz und Evaluierung im Gebiet des Software Engineerings

Für die Anwendungsdomäne des Software Engineering erfolgte exemplarisch die Realisierung eines entsprechenden kontextbasierten Information Retrieval Systems. Mittels dieser Implementierung wurden die im Rahmen des Forschungsprojektes aufgezeigten Methoden und Modelle auf ihre Nützlichkeit und Performanz beim Einsatz in Softwareentwicklungsprozessen hin evaluiert [Morg06]. Konkrete Zielsetzung war hier, wie bereits in der Motivation und in Abbildung 1 abstrakt aufgezeigt, die Unterstützung von Softwarearchitekten und -entwicklern mit zu ihrer aktuellen Tätigkeit relevanten Artefakten aus anderen Softwareprojekten. Neben einer direkten Verwendung von Komponenten stand dabei auch das Aufzeigen von möglichen Lösungswegen, bspw. durch die Wahl geeigneter Softwarearchitekturen oder -designs im Mittelpunkt.

Wenngleich der zugrunde liegende Rational Unified Process [Kruc04] einen strengen Prozessrahmen vorgibt, so verbleiben auch hier in den einzelnen Teilaufgaben zahlreiche Entscheidungsfreiheiten für einen Entwickler, die teilweise erst in späteren Phasen negative wie positive Auswirkungen zeigen können. Dies sind z. B. Fragen der Softwarearchitektur, des Softwaredesigns, der Einsatz bestimmter Integrationswerkzeuge und Middleware-Systeme, aber auch der Einsatz von geeigneten Algorithmen.

Die prototypische Realisierung eines kontextbasierten Information Retrieval Systems basierend auf dem COBAIR-Framework integriert sich in die typische Anwendungslandschaft mit Werkzeugen zur Prozess- und Projektplanung, Verwaltung von Anforderungen, Änderungen und Fehlern sowie Dokumentenmanagementsysteme, Repositories und Verzeichnisdienste. Auf Client-Seite der Entwickler wurde die Integration in Form eines Plug-Ins in eine integrierte Entwicklungsumgebung realisiert (siehe Abbildung 5). Das zur Anwendung kommende Nutzermodell berücksichtigt neben den Fähigkeiten der einzelnen Personen auch deren in zurückliegenden Projekten durchgeführte Tätigkeiten. Auf Basis der aktuellen Tätigkeit wird so ein latentes Informationsbedürfnis antizipiert und automatisch in entsprechende Suchanfragen übersetzt. Für die Evaluierung der Suchqualität und -performanz kam eine Testkollektion aus 72 Open Source Softwareprojekten zum Einsatz.

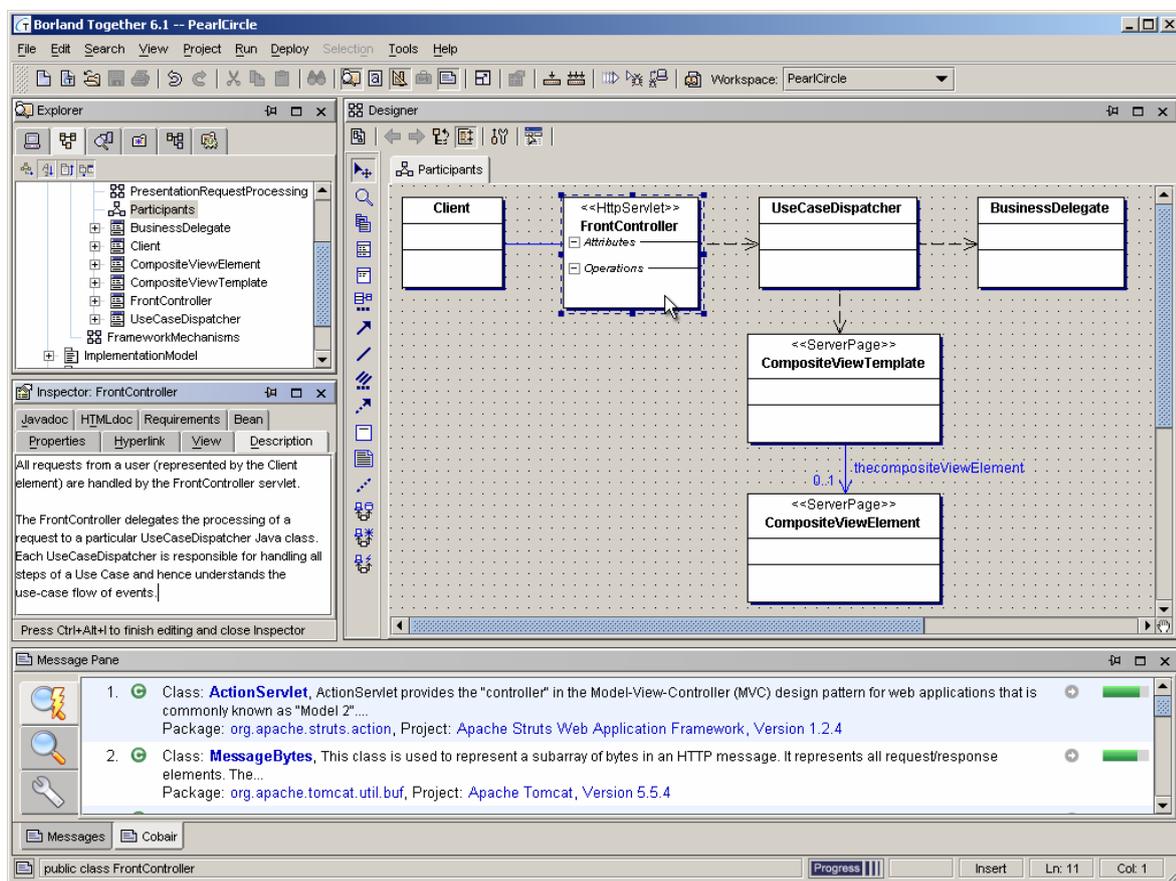


Abb. 5: Einsatz eines Plug-Ins in Borland Together zur Erfassung von Kontextinformation über einen Mitarbeiter sowie zur Anzeige von Suchergebnissen und deren weiteren Verwendung.

Das der Evaluierung zugrunde liegende Szenario umfasst die Entwicklung eines einfachen, webbasierten Auktionssystems in Form einer Mehrschichtenarchitektur, das technisch auf Basis der Java 2 Platform, Enterprise Edition (J2EE) realisiert werden soll. Dieses Beispielszenario, der Verlauf des Softwareentwicklungsprozesses sowie alle dabei entstehenden Artefakte sind

Eeles et al. [EHK02] entnommen. Damit war auch für wiederholte Evaluierungen ein stabiler Verlauf eines Softwareentwicklungsprozesses gewährleistet.

Die Ergebnisse (ausführlich in [Morg06] zu finden) zeigen neben der Tragfähigkeit des Ansatzes auch die Qualität der mit einem entsprechenden kontextbasierten Information Retrieval Ansatz erzielbaren Suchergebnisse. So waren relevante Suchergebnisse im Rahmen der Evaluierung häufig an erster oder zweiter Stelle zu finden (siehe auch Abbildung 5). Die Bearbeitung auch komplexer Suchanfragen, die automatisch auf Basis der vorliegenden Kontextinformation erzeugt wurden, dauerte dabei selten länger als eine Sekunde.

6 Zusammenfassung und Ausblick

Eine abschließende Bewertung der gesamten prototypischen Implementierung eines kontextbasierten Information Retrieval Systems für das Anwendungsgebiet des Software Engineerings zeigt, dass ein derartiges System in der Lage ist, einen Anwender automatisch mit hilfreicher und relevanter Information während seiner Tätigkeit zu unterstützen. Da diese Bewertung in einem allerdings beschränkten Testszenario stattgefunden hat, ist eine weitere Evaluierung in einem größeren Projektumfeld, die dann auch verschiedene Nutzerprofile und -modelle mit einbeziehen könnte, anzustreben.

Dies ist für die Zukunft in einem entsprechenden Rahmen geplant. Dabei sollen Produktentwicklungsprozesse in der Domäne der Konstruktionstechnik und des Maschinenbaus in einem interdisziplinären Forschungsumfeld untersucht werden.

Literaturverzeichnis

- [ABH+00] Abecker, Andreas; Bernardi, Ansgar; Hinkelmann, Knut; Kühn, Otto; Sintek, Michael: Context-Aware, Proactive Delivery of Task-Specific Information: The KnowMore Project. *Information Systems Frontiers*, 2(3/4):253–276, 2000.
- [ABN+01] Abecker, Andreas; Bernardi, Ansgar; Ntioudis, Spyridon; Herterich, Rudi; Houy, Christian; Legal, Maria; Mentzas, Gregory, Müller, Stephan: The DECOR Toolbox for Workflow-Embedded Organizational Memory Access. In: ICEIS 2001, 3rd International Conference on Enterprise Information Systems, Setúbal, Portugal, 7–10 July, 2001.

- [Bern98] Berners-Lee, Tim: A roadmap to the Semantic Web. <http://www.w3.org/DesignIssues/Semantic.html>, September 1998, Zugriff am 1.7.2006.
- [BöHä05] Böhm, Karsten; Härtwig, Jörg: Prozessorientiertes Wissensmanagement durch kontextualisierte Informationsversorgung aus Geschäftsprozessen. In: Ferstl, O. K.; Sinz, E. J.; Eckert, S.; Isselhorst, T. (Hrsg.): Wirtschaftsinformatik 2005. Heidelberg: Physica-Verlag, 2005 (eEconomy eGovernment eSociety), S. 943–962.
- [BuKr99] Budzik, Jay; Hammond, Kristian: Watson: Anticipating and Contextualizing Information Needs. In 62nd Annual Meeting of the American Society for Information Science, Medford, New Jersey, USA, 1999.
- [DCSH04] Dumais, Susan; Cutrell, Edward; Sarin, Raman; Horvitz, Eric: Implicit Queries (IQ) for Contextualized Search. In Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2004), Seite 594, Sheffield, UK, 25.–29. Juli 2004. ACM Press.
- [DeSc04] Dean, Mike; Schreiber, Guus: OWL Web Ontology Language Reference. <http://www.w3.org/TR/owl-ref/>, Februar 2004, Zugriff am 1.7.2006.
- [EHK02] Eeles, Peter; Houston, Kelli; Kozaczynski, Wojtek. Building J2EE Applications with the Rational Unified Process. Addison Wesley, August 2002.
- [HeMo02] Henrich, Andreas; Morgenroth, Karlheinz: Integration von kontextunterstütztem Information Retrieval in Portalsysteme. In: Tagungsband zur Teilkonferenz Management der Mitarbeiter-Expertise in IT-Beratungsunternehmen im Rahmen der Multikonferenz Wirtschaftsinformatik (MKWI 2002), 9.–11. September 2002, Nürnberg.
- [HeMo03] Henrich, Andreas; Morgenroth, Karlheinz: Supporting Collaborative Software Development by Context-Aware Information Retrieval Facilities. In: Proceedings of the 14th International Workshop on Database and Expert Systems Applications (DEXA) (3rd International Workshop on Web Based Collaboration Prague, Czech Republic, September 1–5, 2003). USA: IEEE Computer Society, 2003, Seite 249–253.

- [HeMo05] Henrich, Andreas; Morgenroth, Karlheinz: A Query Engine for RDF based Similarity Queries on Software Artifacts. In: Witold Abramowicz (Ed.): BIS 2005. Proceedings of 8th International Conference on Business Information Systems. Poznan, Poland, 20.–22. April 2005, Seite 169–184.
- [HeRo00] Henrich, Andreas; Robbert, Günter: Combining Multimedia Retrieval and Text Retrieval to Search Structured Documents in Digital Libraries. In Proceedings 1st DELOS Workshop on Information Seeking, Searching and Querying in Digital Libraries, Zürich, Schweiz, Dezember 2000.
- [HeRo03] Henrich, Andreas; Robbert, Günter: Ein Ansatz zur Übertragung von Rangordnungen bei der Suche auf strukturierten Daten. In Tagungsband der 10. GI-Fachtagung Datenbanksysteme für Business, Technologie und Web (BTW'03), Leipzig, 2003.
- [JoFr97] Joachims, Thorsten; Freitag, Dayne; Mitchell, Tom M.: Web Watcher: A Tour Guide for the World Wide Web. In Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI), Seite 770–777, 1997.
- [KlCa04] Klyne, Graham; Carroll, Jeremy J.: Resource Description Framework (RDF): Concepts and Abstract Syntax. <http://www.w3.org/TR/rdf-concepts/>, Februar 2004, Zugriff am 1.7.2006.
- [Kruc04] Philippe Kruchten. The Rational Unified Process: An Introduction. Addison Wesley, 3. Auflage, 2004.
- [Libe95] Lieberman, Henry: Letizia: An Agent That Assists Web Browsing. In Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95), Seite 924–929, Montreal, Quebec, Kanada, 1995. Morgan Kaufmann.
- [Maus01] Maus, Heiko: Workflow Context as a Means for Intelligent Information Support. In Proceedings of 3rd International Conference on Modeling and Using Context (CONTEXT'01), Seite 261–274, 2001.
- [Meie00] Meier, Marco: Integration externer Daten in Planungs- und Kontrollsysteme – Ein Redaktions-Leitstand für Informationen aus dem Internet. Gabler, Wiesbaden, 2000.

- [Morg06] Morgenroth, Karlheinz: Kontextbasiertes Information Retrieval. Modell, Konzeption und Realisierung kontextbasierter Information Retrieval Systeme. Dissertation, Otto-Friedrich-Universität Bamberg, 2006.
- [PaMB96] Pazzani, Michael J.; Muramatsu, Jack; Billsus, Daniel; Syskill & Webert: Identifying Interesting Web Sites. In Proceedings of the Thirteenth National Conference on Artificial Intelligence, Seite 54–61, Portland, Oregon, USA, 1996.
- [PPK03] Priebe, Torsten; Pernul, Günther; Krause, Peter: Ein integrativer Ansatz für unternehmensweite Wissensportale. In Tagungsband 6. Internationale Tagung Wirtschaftsinformatik (WI 2003), Dresden, September 2003.
- [RhSt96] Rhodes, Bradley J.; Starner, Thad: Remembrance Agent: A continuously running automated information retrieval system. In The Proceedings of The First International Conference on The Practical Application Of Intelligent Agents and Multi Agent Technology (PAAM '96), Seite 487–495, London, UK, April 1996.
- [RIS+94] Resnick, Paul; Iacovou, Neophytos; Suchak, Mitesh, Bergstrom, Peter; Riedl, John: GroupLens: an open architecture for collaborative filtering of netnews. In Proceedings of the 1994 ACM conference on Computer supported cooperative work, Seite 175–186, Chapel Hill, North Carolina, USA, 1994. ACM Press.
- [ScKR01] Schafer, J. Ben; Konstan, Joseph A.; Riedl, John: E-Commerce Recommendation Applications. *Data Min. Knowl. Discov.*, 5(1-2):115–153, 2001.
- [ShMa95] Shardanand, Upendra; Maes, Pattie: Social information filtering: algorithms for automating word of mouth. In Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems, Seite 210–217, Denver, Colorado, USA, 1995. ACM Press.
- [SWY75] Salton, Gerard; Wong, Anita; Yang, Chung shu. A vector space model for automatic index. *Communications of the ACM*, 18(11):613–620, November 1975.
- [VDI93] VDI-Richtlinie 2221: Methodik zum Entwickeln und Konstruieren technischer Systeme, 1993.
- [Wils99] Wilson, Tom D.: Models in information behaviour research. *Journal of Documentation*, 55(3):249–270, 1999.