

February 2007

# Individualisierung von Prozessen und E-Services mithilfe von Case Based Reasoning

Günter Schicker

*Universität Erlangen-Nürnberg*

Carolin Kaiser

*Universität Erlangen-Nürnberg*

Freimut Bodendorf

*Universität Erlangen-Nürnberg, bodendorf@wiso.uni-erlangen.de*

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

---

## Recommended Citation

Schicker, Günter; Kaiser, Carolin; and Bodendorf, Freimut, "Individualisierung von Prozessen und E-Services mithilfe von Case Based Reasoning" (2007). *Wirtschaftsinformatik Proceedings 2007*. 43.

<http://aisel.aisnet.org/wi2007/43>

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](mailto: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

# **Individualisierung von Prozessen und E-Services mithilfe von Case Based Reasoning**

Günter Schicker, Carolin Kaiser, Freimut Bodendorf

Lehrstuhl für Wirtschaftsinformatik II

Universität Erlangen-Nürnberg

Lange Gasse 20

90403 Nürnberg

bodendorf@wiso.uni-erlangen.de

## **Abstract**

Mit steigender Wettbewerbsdynamik, kürzeren Strategie- und Produktlebenszyklen sowie zunehmender Leistungsindividualisierung steigen die Anforderungen an die Flexibilisierung der Prozess- und IT-Landschaft, um eine effiziente Implementierung neuer Geschäftsstrategien sicherzustellen. Gleichzeitig ermöglichen neue Technologien und Standards wie Web Services und Serviceorientierte Architekturen (SOA) die IuK-technische Umsetzung flexibler Wertschöpfungssysteme. Es wird ein Konzept zur Individualisierung von Prozessen und E-Services mittels Case Based Reasoning vorgestellt. Anhand von Prozesskontext-Informationen werden Vorschläge für individualisierte Prozesse und unterstützende E-Services automatisch erstellt. Der Ansatz erlaubt die effiziente fachliche Modellierung und E-Service-Unterstützung individueller Prozesse auf Basis einer Serviceorientierten Architektur.

## **1 Motivation**

In den vergangenen Jahren dominierten in den Prozessmanagement-Abteilungen von Unternehmen Diskussionen über die Standardisierung und Harmonisierung von Prozessen. Aufgrund des internen Kostendrucks wurden Anstrengungen unternommen, um Synergien auf fachlicher Ebene z.B. durch Organisation von Shared Services oder Outsourcing von Unterstützungsprozessen zu realisieren [FeSe05, S.14ff]. Grundlage hierfür war die Definition

geeigneter Prozesse und eine an den neu gestalteten Prozessen ausgerichtete IuK-Infrastruktur. Neben fachlichen Treibern wurde die Standardisierung von Prozessen auch vorangetrieben, um die Konsolidierung der Applikationslandschaft zu befördern. Eine Vielzahl heterogener Anwendungen (z.B. unterschiedliche Softwareprodukte oder Systeminstanzen) für vergleichbare Einsatzbereiche führte zu hohen Lizenz-, Wartungs- und Betriebskosten und erforderte – als Basis einer Applikationsharmonisierung – zunächst ein gemeinsames Prozessverständnis sowie abgestimmte Prozesse.

Die Standardisierung von Prozessen stößt jedoch an Grenzen. Der Hauptgrund hierfür liegt in der strategiekonformen Ausrichtung der Prozesslandschaft, die einer rein effizienzorientierten Ausrichtung – zumindest teilweise - entgegen läuft. Um sich vom Wettbewerb zu differenzieren, müssen sich Kernprozesse von Wettbewerbern unterscheiden. Prozesse setzen Strategie um. Die IT-Landschaft setzt Prozesse um. Diese zentralen Aussagen von Porter geben die logische Abhängigkeit vor: Strategie – Prozesse – IT [Port96]. Rückkopplungen sind dabei möglich, z.B. können IT-Innovationen wie RFID Auswirkungen auf Prozesse und Strategie haben. Prozesse sind daher gemäß den strategischen Vorgaben zu modellieren und zu implementieren, um die Unternehmensziele zu erreichen. Aus diesem Grund ist sowohl die Adaption von Referenzprozessmodellen, die Übertragung von Best Practices auf das eigene Unternehmen als auch die Standardisierung von Prozessen mit Bedacht vorzunehmen. Gleichzeitig steigt der Bedarf an Flexibilisierung oder sogar Individualisierung von Prozessen:

- Die Vernetzung nimmt zu und verstärkt Arbeitsteilung und Spezialisierung, um strategische Wertschöpfungsnetze oder Virtuelle Unternehmen zu gestalten [Flei01, S. 17ff].
- Fusionen und Akquisitionen erfordern eine flexible Anpassung der Prozess- und IT-Landschaft, um Synergien im Zuge von Transitionen zügig zu realisieren.
- Die Intensivierung des Wettbewerbs sowie kürzere Produktlebenszyklen führen in dynamischen Branchen (z.B. Consumer Electronics) zu kürzeren Strategiezyklen und zur Notwendigkeit, Strategien prozessseitig zügig umzusetzen [Moor95].
- Die zunehmende Produkt- und Dienstleistungsindividualisierung als strategisches Wettbewerbsinstrument erfordert eine Flexibilisierung der Leistungserstellungsprozesse.

Die Flexibilisierung der Prozesse ist zunehmend wettbewerbskritisch.<sup>1</sup> Die IT-Architektur und die zugehörige Applikationslandschaft ist daher auf eine effiziente und zugleich flexible Unterstützung der Prozesse auszurichten. Dies wird u.a. durch Serviceorientierte Architekturen (SOA) technisch ermöglicht [KrBS04]. Standards zur dynamischen Orchestrierung elektronischer Dienste (Web Services) zur Prozessunterstützung (z.B. BPML, BPEL) etablieren sich, kommerzielle Softwareprodukte zur technischen Konfiguration und Ausführung von Web-Service-basierten Workflows und Orchestrierungen gewinnen an Reife [Chan06, S. 157ff].

Allerdings bleiben zahlreiche Herausforderungen in Bezug auf die fachliche Gestaltung flexibler Prozesse bestehen. Im Folgenden wird ein Ansatz zur Spezifikation individueller Prozesse sowie zur Konfiguration unterstützender E-Services mithilfe von Case Based Reasoning vorgestellt. Ziel ist es, das Design individueller Prozesse zumindest teilweise zu automatisieren und darüber hinaus im Rahmen der Ausführung der Prozessbeschreibung zur richtigen Zeit die richtigen E-Services<sup>2</sup> für die Prozessbeteiligten zur Verfügung zu stellen.

## **2 Bestehende Individualisierungsansätze**

Zur Individualisierung von Prozessen und E-Services existieren verschiedene Konzepte, die nach Art des eingesetzten Instruments unterschieden werden können.

Beim klassischen Ansatz der Prozessindividualisierung wird jeder Prozess für den jeweiligen Kontext passend unter Verwendung von Richtlinien neu modelliert [Lang 97, S. 2]. Diese permanente Neugestaltung von individuellen Prozessen ist sehr zeitintensiv und verhindert die Wiederverwendung von Erfahrungen aus früheren Prozessen [Rupp02, S. 2]. Im Gegensatz dazu ermöglichen die Modellierungsansätze mithilfe von Referenzmodellen [Brock03, S. 31ff], Prozessskeletten [Remme97, S. 114ff] und Prozessbausteinen [Lang97, S. 4ff] die Nutzung von Prozesswissen. Problematisch bei allen Ansätzen ist jedoch die aufwändige Suche und Anpassung an den aktuellen Kontext.

Die Methoden zur Individualisierung von E-Services sind den Verfahren der Prozessindividualisierung sehr ähnlich. Analog zur Prozessindividualisierung besteht der

---

<sup>1</sup> Flexibilität ist die Fähigkeit, sich auf geänderte Anforderungen und Gegebenheiten der Umwelt einstellen zu können. Im Prozesskontext bezieht sich die Änderungsfähigkeit auf Ablauforganisation (Struktur), Geschäftsregeln, Prozessakteure sowie E-Services und deren Provider zur Design- und Laufzeit von Prozessen.

<sup>2</sup> E-Services sind Softwarekomponenten, die aus Servicekontrakt, Schnittstellen und Implementierung bestehen und fachliche Funktionen auf grobgranularer Ebene kapseln (Informations-/Anwendungsdienste) [KrBS04].

klassische Ansatz der E-Service-Individualisierung darin, für jeden Kontext einen E-Service anhand vorgegebener Programmierrichtlinien neu zu erstellen. Diese Vorgehensweise ist jedoch sehr aufwändig und erlaubt keine Wiederverwendung von E-Service-Wissen. Eine Wiederverwendung ermöglichen die Ansätze zur Erzeugung eines E-Services durch den Einsatz von Entwurfsmustern, Frameworks und Bibliotheken.

Abstrakt betrachtet besteht die Vorgehensweise aller Ansätze zur Prozess- und E-Service-Individualisierung aus der Suche eines geeigneten Instruments und der daran anschließenden Anpassung an den individuellen Kontext. Beide Schritte führen bei manueller Durchführung oft zu einem hohen Zeitaufwand. Intelligente Ansätze zur Individualisierung beschäftigen sich deshalb mit der Entwicklung von automatischen Such- und Adaptionenverfahren. Die Systeme „WorkBrain“ [WaWT97, S. 3ff] und „Flexware“ [WaWe97, S. 52f] ermöglichen es zum Beispiel mittels eines fallbasierten Retrieval-Systems bestehende Prozessmodelle zu finden, die einen zur aktuellen Situation ähnlichen Prozesskontext aufweisen. Der „ARIS Process Generator“ [HaRS99, S. 21ff] und das Konzept zur projektspezifischen Individualisierung [Rupp02, S. 67ff] erlauben die automatische Adaption von Prozessmodellen an den Prozesskontext mithilfe von benutzerdefinierten Regeln. Problematisch bei diesen Ansätzen ist, dass nur einer der beiden Schritte, Suche oder Adaption, automatisiert wird, während der andere weiterhin manuell durchzuführen ist. Ziel dieses Projektes ist es daher ein System zu entwickeln, das sowohl intelligente Automatismen zur Suche bestehender Prozessmodelle und ihrer E-Services als auch zur Adaption gefundener Prozessmodelle und ihrer E-Services an den aktuellen individuellen Kontext bietet. Dabei kommt die Methodik der Künstlichen Intelligenz Case Based Reasoning zum Einsatz.

### **3 Case Based Reasoning**

Case Based Reasoning (CBR) ermöglicht das Lösen von Problemen auf Basis von Erfahrungswissen, das in Form von Fällen in einer Fallbasis abgelegt ist [MaTS01, S. 1]. Ein Fall besteht aus zwei grundsätzlichen Teilen, einer Problembeschreibung und der dazugehörigen Lösung. Um ein neues Problem zu lösen, wird in der Fallbasis das dazu ähnlichste Problem gesucht und dessen Lösung wieder verwendet [MaTS01, S. 1]. Im Rahmen der Prozess- und E-Service-Individualisierung können CBR-Systeme dazu verwendet werden, um zu einem gegebenen Prozess- und E-Service-Kontext (Problembeschreibung) den

ähnlichsten in der Fallbasis enthaltenen Prozess- und E-Service-Kontext aufzufinden und dessen Prozess mit E-Services (Lösung) wieder zu verwenden und an den neuen Kontext anzupassen. CBR erfüllt damit die Grundanforderungen der automatischen Selektion und Adaption.

Die Funktionsweise eines CBR-Systems kann als zyklisches Phasenmodell [AaPI94, S. 46] dargestellt werden. In der Retrieve-Phase wird zu dem gegebenen neuen Kontext der Prozess mit E-Services in der Fallbasis gesucht, der den dazu ähnlichsten Kontext enthält. Dieser dient in der Reuse-Phase als Basis zur Lösungsfindung. Im Rahmen der Revise-Phase erfolgt die tatsächliche Ausführung des neuen Prozesses und seiner E-Services. In Abhängigkeit des Erfolges erhält er eine Bewertung, die innerhalb der Retain-Phase über seine Aufnahme in die Fallbasis entscheidet. Abb. 1 veranschaulicht den CBR-Zyklus.

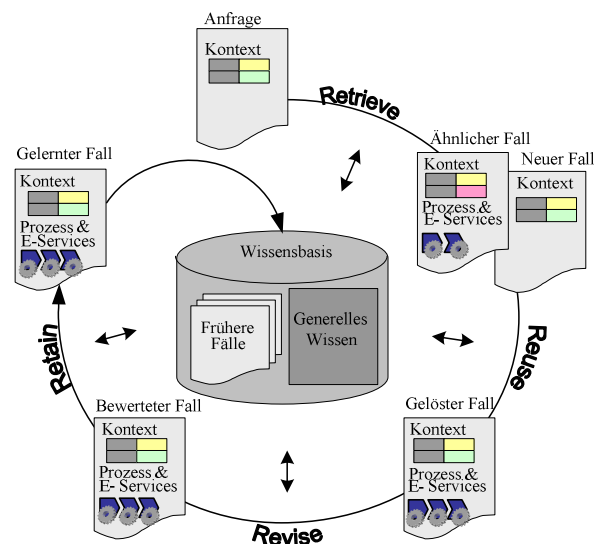


Abb. 1: CBR-Zyklus nach [AaPI94, S. 46]

Bei der Entwicklung eines CBR-Systems zur Prozess- und E-Service-Individualisierung sind eine geeignete Form der Repräsentation der Fälle sowie passende Verfahren zur Ähnlichkeitsbestimmung, Suche, Adaption und Wartung zu bestimmen.

## 4 Prozess- und E-Service-Individualisierung mittels Case Based Reasoning

### 4.1 Definition und Repräsentation von Kontext, Prozess und E-Services

Ein Fall setzt sich aus dem Prozess- und E-Service-Kontext sowie dem ausgeführten Prozess und seinen E-Services zusammen. Der Prozess- und E-Service-Kontext spiegelt die Rahmenbe-

dingungen wider, unter denen der Prozess und seine E-Services sinnvoll eingesetzt werden können (z.B. Branche, Kundenanforderungen, strategisches Umfeld, Phase im Produktlebenszyklus, Prozesskomplexität). Die Repräsentation des Kontextes erfolgt mit Hilfe von Attribut-Werte-Vektoren [Rich03, S. 412]. Hierbei wird der Prozess- und E-Service-Kontext durch eine Anzahl  $n$  von Attributen  $A_1, A_2, \dots, A_n$  näher charakterisiert. Ein konkreter Prozess- und E-Service-Kontext  $P$  wird durch den Vektor der Attributwerte  $a_1, a_2, \dots, a_n$  aus den Wertebereichen  $W_1, W_2, \dots, W_n$  spezifiziert:  $P = (a_1, a_2, \dots, a_n) \in W_1 \times W_2 \times \dots \times W_n$ . Hierbei können metrische, ordinale und nominale Attribute verwendet werden. Um unbekannte Attributwerte zuzulassen, wird der Wertebereich jedes Attributs um die Ausprägung „unbekannt“ erweitert.

Auf Grund ihrer komplexen Struktur wird für Prozesse und E-Services die ausdrucksstarke objektorientierte Repräsentationsform herangezogen. Hierbei werden die Prozesse und E-Services mit ihren Elementen durch Klassen näher charakterisiert. Ein konkreter Prozess mit E-Services besteht aus einer Menge von Objektinstanzen dieser Klassen. Das Klassendiagramm in Abb. 2 zeigt auf, wie Prozesse und E-Services definiert sind.

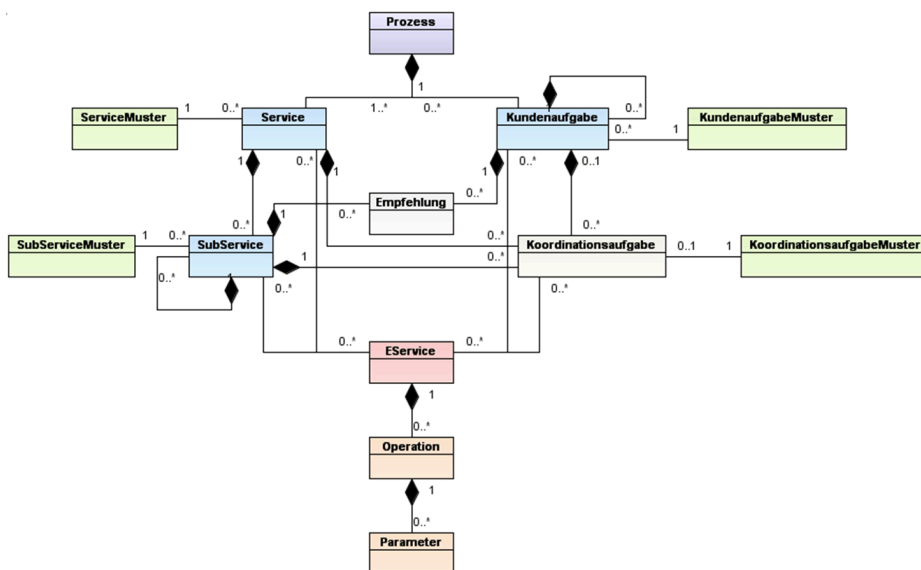


Abb. 2: Prozess- und E-Service-Modell

Ein Prozess besteht aus einem oder mehreren Services, die von einem Serviceanbieter für einen Kunden erbracht werden. Er kann verschiedene Aufgaben beinhalten, die der Kunde selbst zu übernehmen hat. Jeder Service kann mehrere Sub-Services umfassen. Kundenaufgaben und Sub-Services können sich selbst als Element enthalten und ermöglichen somit eine rekursive hierarchische Schachtelung. Reihenfolgebedingte Beziehungen werden nur auf Serviceebene modelliert. Die Prozesselemente Service, Kundenaufgabe und Sub-Service können weiterhin



Empfehlungen und Koordinationsaufgaben beinhalten. Alle Services, Kundenaufgaben, Sub-Services und Koordinationsaufgaben beruhen auf Mustern, die allgemeine Informationen zu den Prozesselementen enthalten (z.B. Name, Beschreibung, Funktion, Ziel). Die Ausführung von Services, Kundenaufgaben, Sub-Services und Koordinationsaufgaben kann durch E-Services unterstützt werden. Jeder E-Service kann mehrere Funktionalitäten in Form von Operationen umfassen. Operationen können wiederum mehrere Parameter enthalten, die ihre Funktionalität näher spezifizieren.

#### 4.2 Bestimmung ähnlicher Prozess- und E-Service-Kontexte

Zielsetzung der CBR-Retrieve-Phase ist es zu einem neuen Kontext den Prozess mit E-Services in der Fallbasis zu ermitteln, der den dazu ähnlichsten Kontext enthält. Zur Bestimmung der Ähnlichkeit zweier Kontexte kommen lokale und globale Ähnlichkeitsmaße zum Einsatz. Während lokale Ähnlichkeitsmaße die Ähnlichkeit zwischen einzelnen Attributausprägungen messen, ermitteln globale die Ähnlichkeit zwischen zwei gesamten Kontexten durch Aggregation der lokalen Ähnlichkeiten [Stah03, S. 50ff; Wess95, S. 125ff].

Lokale Ähnlichkeitsmaße werden in Abhängigkeit des Skalenniveaus des jeweiligen Attributs definiert. Zur lokalen Ähnlichkeitsmessung nominaler Attribute werden Ähnlichkeitstabellen [Goos96, S. 90f] herangezogen. Diese enthalten für jede Kombination aus Attributwerten  $a_i$  des Anfragefalles  $a$  und des Vergleichfalles  $f$  aus der Fallbasis einen Ähnlichkeitswert  $x_{ij} \in [0;1]$ .

$a/f$	$a_1$	$a_2$	...	$a_n$
$a_1$	$I$	$x_{12}$		$x_{1n}$
$a_2$	$x_{21}$	$I$		$x_{2n}$
...				
$a_n$	$x_{n1}$	$x_{n2}$		$I$

Abb. 3: Ähnlichkeitstabelle

Bei metrischen Attributen erfolgt die lokale Ähnlichkeitsbestimmung unter Verwendung von Distanzmaßen, die die Differenz zweier Attributwerte berücksichtigen. Zur lokalen Ähnlichkeitsbestimmung metrischer Attribute stehen verschiedene Maße  $m$  zur Wahl. Als Maße  $m$  können schwellenwertbasierte, lineare, exponentiale und sigmoide Funktionen gewählt werden

[Stah03, S. 54ff]. Nachstehende Abbildung zeigt die Graphen dieser Funktionen für den Wertebereich  $f > a$  auf.

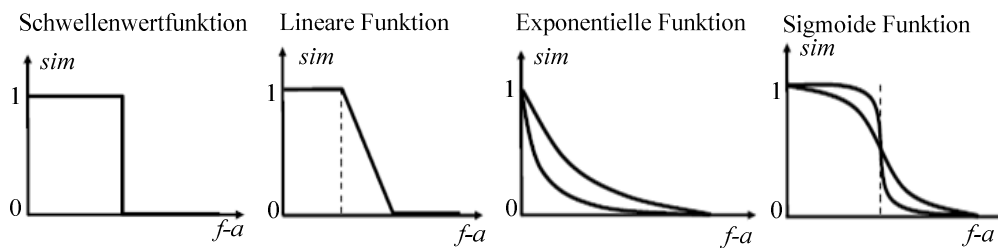


Abb. 4: Ähnlichkeitsfunktionen nach [Stah03, S.56]

Zur Ähnlichkeitsmessung ordinaler Attribute existieren keine eigenen Ähnlichkeitsmaße. Es können jedoch die Maße metrischer Attribute verwendet werden [Goos96, S. 91]. Hierbei müssen die Ausprägungen des ordinalen Attributs auf Zahlenwerte abgebildet werden, über die die Ähnlichkeitsmessung erfolgt. Bei unbekanntem Attributwert erfolgt die Ähnlichkeitsmessung für Attribute aller Skalenniveaus durch eine Schätzung.

Als globales Ähnlichkeitsmaß wird die gewichtete Summe der lokalen Ähnlichkeiten herangezogen. Hierbei kann der Domänenexperte (z.B. Fachprozess-Owner) jedem Attribut ein Gewicht zuweisen, welches die Relevanz des Attributs widerspiegelt.

Die manuelle Definition der Ähnlichkeitsmaße ist zeitaufwändig und erfordert ein hohes Maß an explizitem Domänenwissen. Daher kommt ein Lernalgorithmus zum Einsatz, der die Definition des globalen Ähnlichkeitsmaßes vereinfacht und das Maß optimiert. Mit Hilfe des Lernalgorithmus werden die initialen uniformen oder benutzerdefinierten Gewichte so modifiziert, dass das globale Ähnlichkeitsmaß das Auffinden von nützlichen und adaptierbaren Prozessen und E-Services ermöglicht. Als Lernalgorithmus wird das Gradientenabstiegsverfahren eingesetzt [Stah03, S. 105ff]. Dieses benötigt Trainingsdatensätze, die Informationen vom Benutzer des CBR-Systems über die Rangfolge der Nützlichkeit adaptierten Prozesse und E-Services zu einem gegebenen Problem enthalten. Diese Informationen können während der Benutzung des Systems erfasst werden. Auf Basis des Vergleichs der vom Benutzer angegebenen Rangfolge der adaptierten Lösungen und der durch das Ähnlichkeitsmaß berechneten Rangfolge ist eine Fehlerfunktion definiert, die das Gradientenabstiegsverfahren durch Modifikation der Gewichte minimiert.

### 4.3 Suche ähnlicher Prozesse und E-Services-Kontexte

Ziel der Suchverfahren, die im Rahmen der CBR-Retrieve-Phase eingesetzt werden, ist es, zu einer Anfrage  $n$  Fälle aus der Fallbasis zu finden, die bezüglich des definierten Ähnlichkeitsmaßes die höchste Ähnlichkeit aufweisen [Wess95, S. 159]. Die Suche der ähnlichsten Fälle kann mittels verschiedener Strategien realisiert werden, die alle einen Kompromiss aus Qualität der gefundenen Fälle, Zeitverhalten und Flexibilität erfordern. Zur Prozess- und E-Service-Individualisierung werden zwei alternative Suchverfahren, die sequenzielle Suche und die wissensarme Indexierung mittels erweitertem k-d-Baum [Wess95, S. 209ff], bereitgestellt. Beide Verfahren finden mit Sicherheit die ähnlichsten Fälle in der Fallbasis, unterscheiden sich aber in Effizienz und Flexibilität. Während die sequenzielle Suche beliebige Ad-hoc-Anfragen erlaubt, ermöglicht der erweiterte k-d-Baum eine schnelle standardisierte Suche. Bei der sequenziellen Suche wird die Ähnlichkeit zwischen der Anfrage und jedem Fall der Fallbasis nacheinander mit Hilfe des definierten Ähnlichkeitsmaßes ermittelt und die Fälle werden mit aufsteigender Ähnlichkeit zur Anfrage geordnet [Wess95, S. 163ff].

Die Grundidee des erweiterten k-dimensionalen Baumes besteht darin, die Fallbasis mithilfe eines Baumes nach  $k$  Dimensionen zu partitionieren [Wess95, S. 180f]. Die Dimensionen entsprechen dabei den Attributen des Prozess- und E-Service-Kontextes [Goos96, S. 69f]. Abb. 5 zeigt einen erweiterten 2-d-Baum.

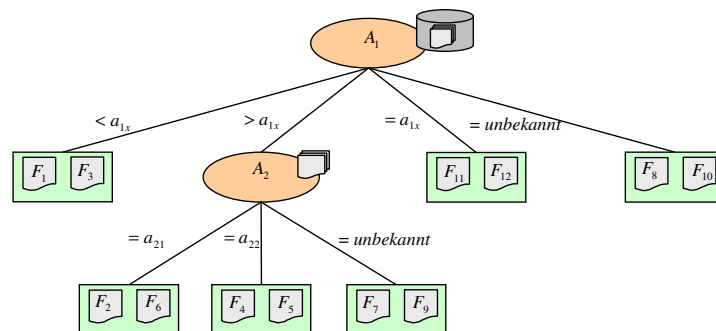


Abb. 5: Erweiterter 2-d-Baum

Der erweiterte k-d-Baum wird automatisch generiert [Wess95, S. 198ff]. Hierbei wird die Baumstruktur so gestaltet, dass zur Ermittlung der ähnlichsten Fälle nur für wenige Fälle die Ähnlichkeit zur Anfrage berechnet werden muss und somit eine effiziente Suche möglich ist [Goos96, S. 71]. Bei der Suche im erweiterten k-d-Baum [Wess95, S. 188, S. 234f] werden zunächst ausgehend von dem Wurzelknoten die Pfade im Baum verfolgt, deren Kantenbedingungen die Attributwerte der Anfrage erfüllen, bis ein Blattknoten erreicht ist. Die

im Blattknoten enthaltenen Fälle werden gemäß ihrer Ähnlichkeit zur Anfrage geordnet. Mithilfe der Attribut-Werte-Bedingungen der Knoten wird getestet, ob noch ähnlichere Fälle existieren, und die Suche gegebenenfalls rekursiv am Vaterknoten fortgesetzt.

#### 4.4 Adaption bestehender Prozesse und E-Services

Ziel der Adaption, die im Rahmen der CBR-Reuse-Phase stattfindet, ist es, ein aktuelles Problem mithilfe des in der CBR-Retrieve-Phase gefundenen ähnlichsten Falles zu lösen. Die Lösung des ähnlichsten Falles dient dabei als Basis für die Bestimmung der Lösung des aktuellen Problems [FLMN99, S. 105].

Die Adaption verläuft, abhängig von dem Ziel der Anfrage, in verschiedenen Schritten. Als Anfrage-Ziele können die Erstellung eines neuen individuellen Prozesses mit E-Services und die individuelle Erweiterung eines bestehenden Prozesses mit seinen E-Services unterschieden werden. Bei der Neuerstellung wird eine Kopie des Prozesses und der E-Services des ähnlichen Falles als Grundlage für die Anfrage verwendet. Im Falle einer Erweiterung wird aus dem Prozess und den E-Services der Anfrage und des ähnlichen Falles ein neuer Prozess mit E-Services zusammengesetzt. Anschließend erfolgt durch die substitutionale und strukturelle Adaption sowie die Konsistenzsicherung eine Anpassung des kopierten bzw. konfigurierten Prozesses und seiner E-Services an den aktuellen Kontext. Nachstehende Grafik veranschaulicht den Ablauf der Adaption.

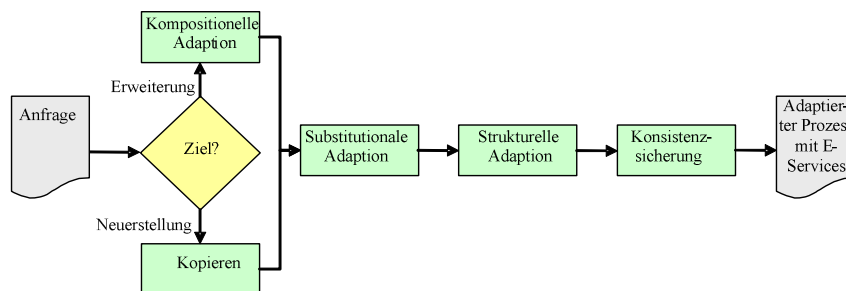


Abb. 6: Ablauf der Adaption

Ziel der kompositionellen Adaption ist es, einen neuen Prozess mit E-Services aus den Prozess- und E-Serviceelementen der Anfrage und des ähnlichen Falles zusammenzustellen. In den neuen Prozess werden alle abgeschlossenen und in Bearbeitung befindlichen Prozess- und E-Serviceelemente der Anfrage übernommen und Prozess- und E-Servicebestandteile des ähnlichen Falles hinzugefügt. Um eine sinnvolle Reihenfolge auf Serviceebene zu ermöglichen, wer-

den in der Retrieve-Phase nur solche Fälle selektiert, deren Serviceabfolge mit den Services, die in der Anfrage abgeschlossen sind oder sich in Bearbeitung befinden, übereinstimmt.

Im Rahmen der substitutionalen Adaption erfolgt die Anpassung der Attributwerte der Prozess- und E-Serviceelemente, die aus dem ähnlichen Fall entnommen sind. Hierbei wird zum Beispiel bei allen Services der Status auf „geplant“ gesetzt, als Editierungsdatum das aktuelle Datum eingetragen und ein Serviceanbieter ausgewählt, für den Präferenzen des Kunden vorliegen.

Zielsetzung der strukturellen Adaption ist es, die Struktur des Prozesses und der E-Services des ähnlichen Falles an den Prozess- und E-Service-Kontext des Anfragefalles anzupassen. In Abhängigkeit von den Gemeinsamkeiten und Unterschieden der Kontextattribute werden Komponenten des Prozesses und der E-Services des ähnlichen Falles entfernt, neue Komponenten hinzugefügt und die Reihenfolge der Komponenten verändert.

Zur Realisierung der strukturellen Adaption wird ein zusätzliches CBR-System, im Folgenden A-CBR-System genannt, verwendet. Die Fälle des A-CBR-Systems werden mit Hilfe der Fallbasis des Haupt-CBR-Systems generiert [JaCR01, S. 1013]. Dadurch werden die Adaptionfälle genau auf die Adaptionsbedürfnisse des Haupt-CBR-Systems abgestimmt. Ein Adaptionsfall setzt sich aus den Anwendbarkeitsbedingungen und dem Adaptionsbedarf (Problembeschreibung) sowie den Adaptionsaktionen (Lösung) zusammen [WiCR02, S. 426ff]. Die Anwendbarkeitsbedingungen beschreiben die Kontextattributwerte und den Prozessablauf, bei dem der Einsatz der Adaptionsaktionen sinnvoll ist. Der Adaptionsbedarf gibt die Unterschiede zwischen den Kontextattributen des Anfragefalles und des ähnlichen Falles an, die die Durchführung von Adaptionsaktionen bedingen. Die Adaptionsaktionen beinhalten die Modifikationen, die an dem ähnlichen Prozess durchgeführt werden müssen, um ihn an den neuen Kontext anzupassen. Hierbei stehen *FügeHinzu*-, *Lösche*-, *ÄndereReihenfolge*- und *ÄndereStrukturelleBeziehung*-Aktionen zur Verfügung.

Im Rahmen der strukturellen Adaption wird für einen Anfragefall und einen dazu ähnlichen Fall der Adaptionsfall gesucht, der die am besten geeigneten Adaptionsaktionen enthält. Zu diesem Zweck erzeugt das Haupt-CBR-System aus dem Anfragefall und seinem ähnlichen Fall einen Adaptionsanfragefall, den es dem A-CBR-System übergibt. Dieses selektiert aus seiner Fallbasis den dazu ähnlichsten Adaptionsfall und übermittelt ihn an das Haupt-CBR-System, das daraufhin die Adaptionsaktionen des Adaptionsfalles ausführt. Abb. 7 veranschaulicht das Zusammenspiel zwischen Haupt-CBR-System und A-CBR-System.

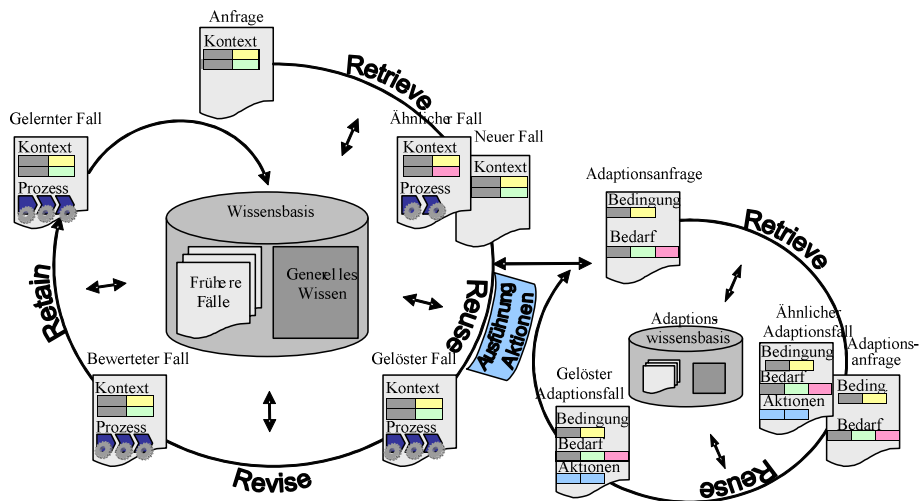


Abb. 7: Strukturelle Adaption

Da es sich bei den Adaptionenmechanismen um Heuristiken handelt, ist die Konsistenz des so erzeugten Prozesses und seiner E-Services nicht garantiert. Um die Konsistenz und damit auch die Qualität des Prozesses mit seinen E-Services zu erhöhen, kommen lokale und globale Konsistenzsicherungsmaßnahmen zur Anwendung. Im Rahmen der lokalen Konsistenzsicherung wird die Eignung einzelner Prozess- und E-Serviceelemente bezüglich des Prozess- und E-Service-Kontextes überprüft und gegebenenfalls werden Korrekturmaßnahmen vorgenommen. Die globale Konsistenzsicherung hingegen beschäftigt sich mit den Interdependenzen zwischen den einzelnen Prozess- und E-Serviceelementen und führt Maßnahmen zur Verbesserung der Konsistenz des gesamten Prozesses und seiner E-Services durch.

#### 4.5 Ausführung des Prozesses und seiner E-Services

Ergebnis der Adaption ist ein individueller Prozess- und E-Servicevorschlag auf Basis gesammelten Prozesswissens. Ziel ist es jedoch, nicht nur die Definition individualisierter Prozesse zu unterstützen, sondern vielmehr den Ansatz einer prozessbasierten E-Service-Logistik zu realisieren. Die richtigen E-Services sollen zur richtigen Zeit am richtigen Ort im Prozess zur Verfügung gestellt werden und so die Koordinations- und Informationsbedarfe der Prozessbeteiligten befriedigen. Die dazu notwendige Ausführung der durch das CBR-System vorgeschlagenen Prozess- und E-Service-Modelle wird durch die Meta-Orchestration unterstützt, die in Abschnitt 5 skizziert wird. Zu diesem Zweck wird der Prozessvorschlag des CBR-Systems in eine XML-basierte Prozessbeschreibung transformiert und ausgeführt.

#### **4.6      Wartung des CBR-Systems**

Veränderungen in der Umwelt, dem Aufgabenfokus, den Benutzeranforderungen und den Wissenscontainern des CBR-Systems können zu einer Verschlechterung der Ergebnisqualität und Effizienz des CBR-Systems führen [Wils01, S. 1]. Die Zielsetzung der Wartung, die im Rahmen der CBR-Retain-Phase stattfindet, besteht darin, das Wissen eines CBR-Systems (Prozess- und E-Service-Wissen) zu bewahren und gegebenenfalls zu korrigieren, um eine hohe Qualität und Effizienz zu gewährleisten [Roth02, S. 30]. Im Rahmen der Individualisierung von Prozessen und E-Services werden die Wissenscontainer „Ähnlichkeitswissen“, „Adaptionswissen“ und „Fallbasis“ mit jeweils eigenen Verfahren automatisch gewartet. Der Wissenscontainer „Vokabular“, bestehend aus Kontextattributen und Prozess- und E-Serviceelementen, wird manuell von einem Domänenexperten gewartet. Das globale Ähnlichkeitswissen wird manuell initialisiert und anschließend fortlaufend interaktiv mittels der benutzerpräferierten Rangfolgen der adaptierten Prozesse und E-Services optimiert. Die Generierung des strukturellen Adaptionswissens aus Fällen der Fallbasis ermöglicht die Erweiterung und Optimierung des Adaptionswissens. Die Wartung der Fallbasis besteht aus den Retain- sowie Review- und Restore-Schritten [Roth02, S. 55ff]. Im Rahmen des Retain-Schrittes entscheiden Intrafallqualitätsmaße über die Aufnahme eines Prozesses mit seinen E-Services in die Fallbasis. Innerhalb des Review-Schrittes werden die Prozesse und E-Services hinsichtlich ihrer Intrafallqualität, Interfallkonsistenz und Redundanz überprüft und gegebenenfalls im Restore-Schritt gelöscht.

### **5      Systemarchitektur**

Die CBR-Lösung ist Teil der Gesamtarchitektur „Individual Value Web System (IVWS)“ zur prototypischen Realisierung des Forschungsansatzes einer prozessbasierten E-Service-Logistik anhand interorganisatorischer Behandlungsprozesse in Gesundheitsnetzen [ScBo06]. Die Architektur unterstützt als Prozess- und Integrationsplattform die Konfiguration und Ausführung individueller Prozesse. Sie wurde auf Basis von .NET realisiert und gliedert sich in vier Ebenen. Präsentations-Ebene: Application Frontends initiieren und steuern alle Aktivitäten des IVWS. Typischerweise werden im Projekt Application Frontends über graphische Benutzeroberflächen

implementiert, welche die direkte Interaktion der Endanwender im Rahmen rollenspezifischer Prozessportale mit dem System ermöglichen (in C# realisierte WebParts des Portal Server).

Customization- und Prozess-Steuerungs-Ebene: Diese Ebene besteht aus drei Komponenten – der Prozess- und E-Service-Customization, der Meta Orchestration sowie dem Service Bus, welcher u.a. Funktionen für die Orchestration und Ausführung von Web Services beinhaltet.

- Service Bus: Diese Komponente verbindet alle Netzwerkakteure. Sie ermöglicht die Kommunikation zwischen Application Frontends und E-Services. Die erforderliche Funktionalität wird durch den MS Biztalk Server zur Verfügung gestellt (u.a. Connectivity, Integration Services, Communication). Der Service Bus unterstützt auch die technische Konfiguration und Ausführung feingranularer, Nachrichtenfluss-orientierter Web Service-Orchestrations (WSO), z.B. repräsentiert in BPEL oder XLANG.
- Prozess- und E-Service-Customization: Für die Individualisierung von Prozessen und der Anpassung der zugeordneten E-Service-Logistik wird Case Based Reasoning genutzt (vgl. Kapitel 4). Die vom CBR-System adaptierte Prozess- und E-Service-Spezifikation wird zur Ausführung an die Meta-Orchestration-Engine übermittelt.
- Meta Orchestration: Die Architektur basiert auf dem Prinzip Web-Service-basierter Workflows und dem Einsatz der WSO. Defizite bei der Anwendung der WSO und der dafür verfügbaren Systeme sind geringe Flexibilität, unzureichende Individualisierbarkeit und fehlende Benutzernähe. Die Meta Orchestration liefert einen Beitrag zur Behebung dieser Defizite, indem sie als Zwischenschicht zur Orchestrierungsebene (z.B. Biztalk Server) fungiert. Der Prozess-Owner kann die aus der CBR-Komponente übermittelten Prozessmodelle und E-Service-Parameter modifizieren (z.B. E-Services entfernen/hinzufügen) und das Modell instanziiieren. Die technische Konfiguration wird in der Laufzeitumgebung ausgeführt, d.h. die Meta-Orchestration-Engine stößt die Ausführung „normaler“ WSO des Biztalk-Servers an [ScBo06].

Applikations-Ebene: Neben den geschilderten IVWS-spezifischen CBR- und Orchestrierungs-Diensten kann jede Form existierender Web Services Informationen und Anwendungen den Netzwerk-Teilnehmern zur Verfügung stellen, über SOAP und WSDL über den Service Bus aufrufen und in die E-Service-Logistik integrieren.

Daten-Ebene: Diese Ebene stellt die für die Umsetzung notwendige Datenbasis bereit. In einer MS SQL-Datenbank werden die Prozess- und E-Service-Daten gespeichert und verwaltet.



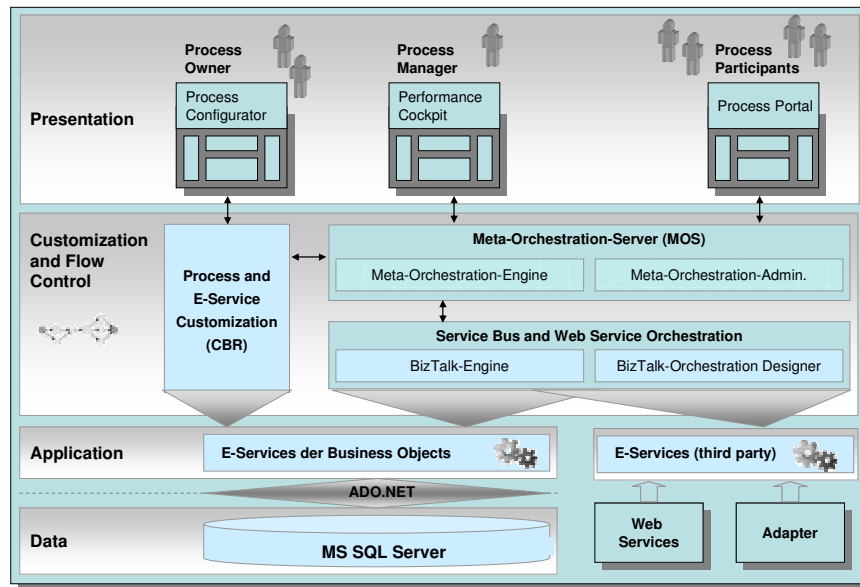


Abb. 8: Individual Value Web System

## 6 Zusammenfassung und Ausblick

Um Strategien in der Prozess- und IT-Landschaft effizient und effektiv umzusetzen, nimmt der Bedarf zur Flexibilisierung bzw. Individualisierung von Prozessen zu. Gleichzeitig ermöglichen neue Technologien und Standards wie Web Services und Serviceorientierte Architekturen die IuK-technische Umsetzung flexibler Wertschöpfungssysteme. Ausgehend von den bestehenden Ansätzen der Individualisierung und deren Defizite wurde ein System zur Individualisierung von Prozessen und E-Services mithilfe von Case Based Reasoning entwickelt. Zielsetzung war es, den manuellen Aufwand für die Definition und Modellierung von Prozessen für den Benutzer gering zu halten und den Vorschlag automatisch an den gegebenen Prozesskontext anzupassen. Das Konzept basiert auf intelligenten Automatismen zur Selektion geeigneter bestehender Prozesse und ihrer E-Services sowie zur Adaption gefundener Prozesse und ihrer E-Services. Die Adaption erfolgt anhand automatischer transformationsorientierter Verfahren. Hierbei werden nacheinander kompositionelle, substitutionale und strukturelle Adaptionsmechanismen sowie eine Konsistenzsicherung ausgeführt.

Das vorgestellte CBR-System zur Individualisierung von Prozessen und E-Services lässt sich dort einsetzen, wo Serviceorientierte Architekturen und Möglichkeiten der technischen Konfiguration von E-Services geschaffen wurden. Der Ansatz unterstützt die anwenderbezogene, fachliche Prozessdefinition anhand des Prozesskontextes und ermöglicht so die dynamische

Konfiguration von Prozessen, die Auswahl relevanter Prozessakteure und die Orchestrierung der E-Services. Der Ansatz ergänzt daher technisch orientierte Ansätze der Prozessrepräsentation und -konfiguration wie BPEL oder BPML. Eine aufwändige Modellierung zahlreicher Prozessvarianten oder gar individueller Prozessinstanzen entfällt.

Die Einrichtung und Wartung solcher Systeme ist mit Aufwand verbunden. So müssen vor der Nutzung des Systems einheitliche Kontextattribute, Prozesselement-Muster und E-Serviceelemente definiert sowie bestehende Prozesse mit ihren Kontexten und E-Services in dieses Format transformiert werden. Während der Nutzungsphase des Systems sind diese Daten stets an veränderte Umweltbedingungen und Benutzeranforderungen anzupassen. Gleichzeitig erfolgt damit jedoch die Explizierung und Formalisierung von Prozesswissen. Das CBR-System stellt ein Basisverfahren zur Individualisierung von Prozessen und E-Services dar, das sich noch durch zusätzliche Konzepte (z.B. Prognose unbekannter Kontextattribute) erweitern lässt. Auch die Kombination mit Prozess Repositories, Modellierungssystemen oder die Integration in Business Process Management Systeme (z.B. SAP XI, Intalio n<sup>3</sup>) weist Forschungspotenzial auf, um Wertschöpfungssysteme zu flexibilisieren und die Möglichkeiten Serviceorientierter Architekturen aus fachlicher Sicht auszuschöpfen.

## Literaturverzeichnis

- [AaP194] Aamodt, Edgar; Plaza, Enric: Case Based Reasoning: Foundational Issues, Methodological Variations and System Approaches. In: AI Communications 7 (1994) 1, S. 39-59.
- [Chan06] Chang, James F.: Business Process Management Systems: Strategy and Implementation. Taylor & Francis, Boca Raton 2006.
- [FeSe05] Feldmayer, Johannes; Seidenschwarz, Werner: Marktorientiertes Prozessmanagement – Wie Process Mass Customization Kundenorientierung und Prozessstandardisierung integriert. Vahlen, München 2005.
- [Flei01] Fleisch, Elgar: Das Netzwerkunternehmen: Theorien, Strategien und Prozesse zur Steigerung der Wettbewerbsfähigkeit in der "Networked economy". Springer, Berlin et. al. 2001.

- [FLMN99] Fuchs, Béatrice; Lieber, Jean; Mille, Alain; Napoli, Amedeo: Towards a Unified Theory of Adaptation in Case Based Reasoning. In: Althoff, K.-D.; et. al. (Hrsg.): Case Based Reasoning Research and Development. Springer, Berlin 1999, S. 104-117.
- [Goos96] Goos, Klaus: Fallbasiertes Klassifizieren: Methoden, Integration und Evaluation. Infix Verlag, Sankt Augustin 1996.
- [HaRS99] Hagemeyer Jens, Rolles Roland, Scheer August-Wilhelm: Der schnelle Weg zum Sollkonzept: Modellgestützte Standardsoftwareeinführung mit dem ARIS Process Generator. In: Scheer, August-Wilhelm (Hrsg.): Veröffentlichungen des Instituts für Wirtschaftsinformatik an der Universität Saarbrücken, Heft 152, Saarbrücken 1999.
- [JaCR01] Jarmulak, Jacek; Craw, Susan; Rowe, Ray: Using Case-Based Data to Learn Adaptation Knowledge for Design. In: Proceedings of the 17th IJCAI Conference. Morgan Kaufmann, o.O. 2001, S. 1011-1016.
- [KrBS04] Krafzig, Dirk; Banke, Karl; Slama, Dirk: Enterprise SOA, Service-Oriented Architecture Best Practices. Prentice Hall PTR, Maryland 2004.
- [Lang97] Lang, Klaus: Gestaltung von Geschäftsprozessen mit Referenzbausteinen. Universität Erlangen-Nürnberg, Dissertation. 1997.
- [MaTS01] Main, Julie; Dillon, Tharam S.; Shiu, Simon C. K.: A Tutorial on Case Based Reasoning. In: Pal, Sankar K.; Dillon, Tharam S.; Yeung, Daniel S. (Hrsg.): Soft Computing in Case Based Reasoning. Springer, London 2001, S. 1-28.
- [Moor95] Moore, Geoffrey A.: The tornado – marketing strategies from Silicon Valley's cutting edge. Harper Collins, New York 1995.
- [Port96] Porter, Michael E.: What is strategy? In: Harvard Business Review, (1996) 11.
- [Remm97] Remme, Markus: Konstruktion von Geschäftsprozessen – Ein modellgestützter Ansatz durch Montage genetischer Prozesspartikel. Gabler, Wiesbaden 1997.

- [Rich03] Richter, Michael M.: Fallbasiertes Schließen. In: Görz, Günther; Rollinger, Claus-Rainer; Schneeberger, Josef (Hrsg.): Handbuch der Künstlichen Intelligenz. 4. Aufl. Oldenbourg Wissenschaftsverlag, München 2003, S. 407-430.
- [Roth02] Roth-Berghofer, Thomas: Knowledge Maintenance of Case Based Reasoning Systems - The SIAM Methodology. Universität Kaiserslautern, Dissertation. 2002.
- [Rupp02] Rupprecht, Christian: Ein Konzept zur projektspezifischen Individualisierung von Prozessmodellen. Universität Karlsruhe, Dissertation. 2002.
- [ScBo06] Schicker, Günter; Bodendorf, Freimut: Process-based E-Service-Logistics for Healthcare Networks. In: Gesellschaft für Informatik (Hrsg.): Lecture Notes in Informatics. European Conference on eHealth. Fribourg 2006, S. 37-46.
- [Stah03] Stahl, Armin: Learning of Knowledge-Intensive Similarity-Measures in Case Based Reasoning. Universität Kaiserslautern, Dissertation. 2003.
- [WaWe97] Wargitsch, Christoph; Wewers, Thorsten: Flexware: Fallorientiertes Konfigurieren von komplexen Workflows – Konzepte und Implementierungen. In: Müller, M.; et. al. (Hrsg.): Beiträge zum 11. Workshop Planen und Konfigurieren im Rahmen der 4. Tagung Wissensbasierte Systeme. Erlangen 1997, S. 45-55.
- [WaWT97] Wargitsch, Christoph; Wewers, Thorsten; Theisinger, Felix: WorkBrain: Merging Organizational Memory and Workflow Management Systems. In: Workshop ‚Knowledge Based Systems for Knowledge Management in Enterprises‘ im Rahmen der 21. KI-Jahrestagung. Freiburg 1997.
- [Wess95] Wess, Stefan: Fallbasiertes Problemlösen in wissensbasierten Systemen zur Entscheidungsunterstützung und Diagnostik. Infix Verlag, Sankt Augustin 1995.
- [WiCR02] Wiratunga, Nirmalie; Craw, Susan; Rowe, Ray: Learning to Adapt for Case-Based Design. In: Lecture Notes in Computer Science, Band 2416. Springer Verlag, o.O. 2002, S. 421-435.
- [Wils01] Wilson, David C.: Case-Base Maintenance: The Husbandry of Experience. Indiana University, Dissertation. 2001.