



*Recherche und vergleichende
Evaluation von verfügbaren Ansätzen
für
"Content-oriented Workflows"*

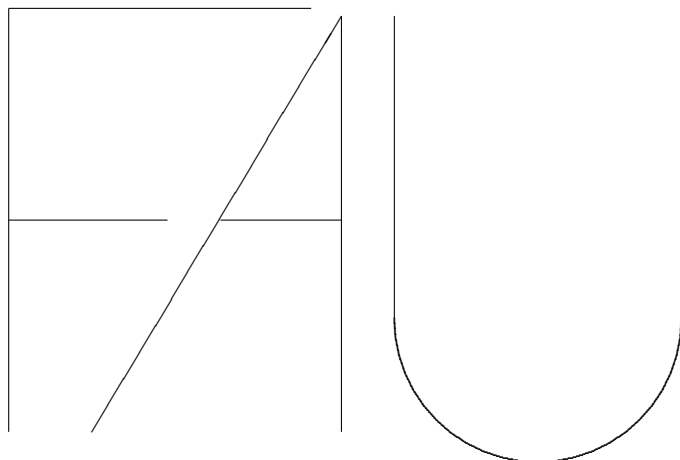
Diplomarbeit

André Kreuter

Lehrstuhl für Informatik 6
(Datenmanagement)

Department Informatik
Technische Fakultät

Friedrich Alexander-
Universität
Erlangen-Nürnberg



Recherche und vergleichende Evaluation von verfügbaren Ansätzen für "Content-oriented Workflows"

Diplomarbeit im Fach Informatik

vorgelegt von

André Kreuter

geb. 22.12.1984 in Castrop-Rauxel

angefertigt am

**Department Informatik
Lehrstuhl für Informatik 6 (Datenmanagement)
Friedrich-Alexander-Universität Erlangen-Nürnberg**

Betreuer: Univ.-Prof. Dr.-Ing. habil. Richard Lenz
Dipl.-Inf. Christoph P. Neumann

Beginn der Arbeit: 01.10.2010

Abgabe der Arbeit: 01.04.2011

Erklärung zur Selbständigkeit

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass diese Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Der Universität Erlangen-Nürnberg, vertreten durch den Lehrstuhl für Informatik 6 (Datenmanagement), wird für Zwecke der Forschung und Lehre ein einfaches, kostenloses, zeitlich und örtlich unbeschränktes Nutzungsrecht an den Arbeitsergebnissen der Diplomarbeit einschließlich etwaiger Schutzrechte und Urheberrechte eingeräumt.

Erlangen, den 01.04.2011

(André Kreuter)

Kurzfassung

Recherche und vergleichende Evaluation von verfügbaren Ansätzen für "Content-oriented Workflows"

Im medizinischen Umfeld unterstützen heutzutage moderne Softwaresysteme nur bedingt inter-institutionelle Arbeitsabläufe. Mittels eines Prozessunterstützungssystems die Behandlung eines Patienten abzubilden, parallel medizinische Informationen aufzunehmen und diese denjenigen wieder zur Verfügung zu stellen, die sie benötigen, optimiert die Prozesseffizienz. Die Zusammenarbeit zwischen Institutionen erfordert definierte Schnittstellen bezüglich des Arbeitsgebietes und der Verantwortung der an einem Prozess beteiligten. Ansätze, die nur das inhaltsbasierte oder aktivitätsbasierte Paradigma beschreiben, werden die an sie gestellten Forderungen nur bedingt erfüllen. Zur Optimierung der Zusammenarbeit muss zusätzlich noch die soziale Interaktion unterstützt werden. Für eine vergleichende Bewertung wurden fünf aktuelle Ansätze ausgewählt. Zur besseren Differenzierung werden die Ansätze neben der inhaltsbasierten und aktivitätsbasierten Steuerung auch bezüglich der Strukturiertheit ihres Prozessverlaufes bewertet. Basierend auf den Eigenschaften der Ansätze wurden Vergleichskriterien definiert, die für einen Vergleich der unterschiedlichen Ansätze verwendet werden. Bei der Bewertung der Ansätze bezüglich eines möglichen Einsatzes im medizinischen Umfeld, konnten deren Stärken und Schwächen aufgezeigt werden. Für den Einsatz im medizinischen Umfeld eignet sich primär α -Flow, dessen großer Vorteil die Verwendung eines elektronischen Dokuments ist, auf Basis dessen Informationen ausgetauscht werden. Die restlichen Ansätze besitzen Schwächen vor allem in dem Bereich der Anpassbarkeit an geänderte Rahmenbedingungen, eine Eigenschaft die vor allem im medizinischen Umfeld benötigt wird.

Abstract

Research and comparative evaluation of available approaches to "Content-oriented Workflows"

Nowadays, healthcare inter-institutional processes are only limited supported by modern software systems. By means of a process supporting system representing the patient treatment, assimilating medical information in parallel and distributing them to those, who need the information, will optimize the process efficiency. The collaboration between institutions requires clear interfaces regarding the working areas and the corresponding responsibilities of the process participants. Workflows, which follow either the content- or the activity-based paradigm, will basically fail to fulfill the requirements they are facing. Furthermore, to facilitate the cooperation, the social interaction between the process participants has to be supported. For a comparative evaluation five up to date process supporting systems are selected. To improve the differentiation between them, beside the content- and activity-based process control also the structure of the process itself is evaluated. Based on the properties of the process control systems comparative criteria are defined, which are used for a comparison of the applied process control systems. An evaluation of the process control system regarding their potential use in the healthcare, shows their advantages and disadvantages. For a potential use in the healthcare α -Flow is the primary candidate. Its big advantage is the application of an electronic document, which is the basis for the information exchange. The remaining process control systems show weaknesses especially regarding the adaptation to changed boundary conditions, which is a big challenge in the healthcare.

Danksagung

Ich möchte mich hiermit bei Herrn Prof. Dr.-Ing. Richard Lenz und Herrn Christoph P. Neumann für ihre Unterstützung bedanken. Sie standen mir immer mit Rat und Tat zur Seite und ich hätte mir keine besseren Betreuer vorstellen können. Außerdem möchte ich mich bei meinen Eltern für ihre unendliche Geduld und Korrektur bedanken. Ein weiterer Dank geht an alle diejenigen, die mich in irgendeiner Art unterstützt haben, insbesondere Kristian Rathert und Stefan Hanisch, die sich Zeit für die Korrektur der Arbeit genommen haben.

Inhaltsverzeichnis

| | |
|--|------------|
| Abbildungsverzeichnis | v |
| Tabellenverzeichnis | vii |
| Abkürzungsverzeichnis | xi |
| 1 Einleitung | 1 |
| 1.1 Motivation und Herausforderung | 1 |
| 1.2 Ziele der Arbeit | 3 |
| 1.3 Abgrenzung | 5 |
| 2 Methodik | 7 |
| 3 Arbeitsabläufe, Inhalte, Kooperationen im medizinischen Kontext | 9 |
| 3.1 Begriffsdefinition der Anpassbarkeit | 11 |
| 3.2 Aktivitätsbasiertes Paradigma | 11 |
| 3.3 Inhaltsbasiertes Paradigma | 13 |
| 3.4 Groupware | 15 |
| 3.4.1 Was ist Groupware | 15 |
| 3.4.2 Groupware Paradigma | 17 |
| 3.5 Pradigmen im Kontex der Medizin | 18 |
| 3.6 Paradigmen in wissensintensiven, semistrukturierten Prozess in der Medizin | 19 |
| 3.7 Spannungsdreieck der drei Paradigmen | 20 |
| 4 α-Flow Kurzdarstellung und Einsatzumgebung | 23 |
| 4.1 Kurzdarstellung des Hintergrunds von α -Flow | 23 |
| 4.2 Realsierung der Paradigmen | 24 |
| 4.3 Paradigmabewertung | 25 |

| | | |
|----------|--|-----------|
| 5 | Kurzdarstellung der betrachteten Ansätze zur Prozessunterstützung | 27 |
| 5.1 | α -Flow | 28 |
| 5.1.1 | Kurzdarstellung von α -Flow | 28 |
| 5.1.2 | Beispiel für die Anwendung von α -Flow | 30 |
| 5.2 | Business Process Modelling and Notation | 31 |
| 5.2.1 | Kurzdarstellung des Ansatz BPMN | 31 |
| 5.2.2 | Beispiel für eine Anwendung von BPMN | 32 |
| 5.3 | Adaptive Case Management | 33 |
| 5.3.1 | Kurzdarstellung des Ansatzes ACM | 34 |
| 5.3.2 | Beispiel für die Anwendung von ACM | 36 |
| 5.4 | Artifact-Centric Business Process Models | 37 |
| 5.4.1 | 4-Dimensionen | 38 |
| 5.4.2 | Herausforderungen vom ACBPM-Ansatz | 39 |
| 5.4.3 | Beispiel für die Anwendung von ACBPM | 40 |
| 5.5 | Datenorientiertes Prozess-Management-System | 41 |
| 5.5.1 | Kurzdarstellung des DPMS Ansatzes | 41 |
| 5.5.2 | Beispiel für die Anwendung von DPMS | 43 |
| 5.6 | Data-driven Process Structures | 44 |
| 5.6.1 | Kurzdarstellung des Ansatzes DDPS | 45 |
| 5.6.2 | Beispiel für die Anwendung von DDPS | 46 |
| 5.7 | Gemeinsame Eigenschaften der Ansätze zur Prozessunterstützung | 47 |
| 6 | Vergleichsparameter | 49 |
| 6.1 | Aspekte eines Ansatzes zur Prozessunterstützung | 49 |
| 6.2 | Begriffserklärung | 50 |
| 6.3 | Vergleichskriterien der unterschiedlichen Aspekte | 51 |
| 6.3.1 | Gesamtmethodische Aspekte | 51 |
| 6.3.2 | Funktionale Aspekte | 55 |
| 6.3.3 | Informative Aspekte | 58 |
| 6.3.4 | Dynamische Aspekte | 60 |
| 6.3.5 | Integrative Aspekte | 61 |
| 6.3.6 | Technisch-operative Aspekte | 62 |
| 6.3.7 | Software-operative Aspekte | 63 |
| 6.3.8 | Evolutionäre Aspekte | 68 |
| 6.4 | Aspekte der Systeme zur Prozessunterstützung und deren Zusammenhang | 69 |

| | | |
|----------|---|------------|
| 7 | Vergleich der unterschiedlichen Ansätze zur Prozessunterstützung | 71 |
| 7.1 | Aufbau des Vergleichs der Ansätze zur Prozessunterstützung | 71 |
| 7.2 | Vergleichsdurchführung | 72 |
| 7.2.1 | Gesamtmethodische Aspekte | 72 |
| 7.2.2 | Funktionale Aspekte | 77 |
| 7.2.3 | Informative Aspekte | 81 |
| 7.2.4 | Dynamische Aspekte | 84 |
| 7.2.5 | Integrative Aspekte | 86 |
| 7.2.6 | Technischoperative Aspekte | 87 |
| 7.2.7 | Softwareoperative Aspekte | 88 |
| 7.2.8 | Evolutionäre Aspekte | 93 |
| 7.3 | Bewertung der Ansätze zur Prozessunterstützung | 95 |
| 7.3.1 | Was leisten die Ansätze zur Prozessunterstützung | 96 |
| 7.3.2 | Wo liegen die Grenzen der Ansätze zur Prozessunterstützung . . . | 101 |
| 7.3.3 | Lebenszyklus | 105 |
| 7.4 | Einordnung der Ansätze ins Paradigmadreieck | 106 |
| 7.5 | Erfahrungswerte aus dem Vergleich | 112 |
| 8 | Evaluation | 117 |
| 8.1 | Evaluation der Anforderungen | 117 |
| 8.2 | Ausblick | 120 |
| 9 | Zusammenfassung | 125 |
| A | Begriffsdefinition | 129 |
| A.1 | Prozess | 129 |
| A.2 | Workflow | 129 |
| A.3 | Organisatorische Sicht | 129 |
| A.4 | IT Sicht | 129 |
| A.5 | Aktivität | 129 |
| A.6 | Anwendungsfunktion | 130 |
| A.7 | Objektlebenszyklus | 130 |
| A.8 | Konfiguration | 130 |
| B | Internet Kommunikationsmittelverteilung | 131 |
| C | BPMN-Poster | 135 |

Literaturverzeichnis

I

Abbildungsverzeichnis

| | | |
|------|--|----|
| 1.1 | Schematische Darstellung eines Behandlungsablauf in der Medizin | 2 |
| 1.2 | Skizze eines Beinbruchbehandlung | 3 |
| 1.3 | Positionierung von Systemen zur Prozessunterstützung | 6 |
| 3.1 | Skizze eines Workflow-Systems | 10 |
| 3.2 | Skizze eines Prozessablaufes nach dem aktivitätsbasiertem Paradigma . . | 13 |
| 3.3 | Skizze eines Prozessablaufes nach dem inhaltsbasiertem Paradigma . . . | 15 |
| 3.4 | Raum-Zeit-Matrix von Bullen und Johansen | 16 |
| 3.5 | Skizze eines Prozessablaufes nach dem Groupware Paradigma | 18 |
| 3.6 | Skizze des Zusammenspiels von unterschiedlichen Paradigmen | 21 |
| 5.1 | Skizze eines α -Flow-Anwendungsfalls | 30 |
| 5.2 | Schematische Darstellung einer Beinbruchbehandlung | 33 |
| 5.3 | ACM-Systemkomponentenskizze | 35 |
| 5.4 | Skizze eines ACM-Anwendungsfalls | 37 |
| 5.5 | ACBPM-Systemkomponentenskizze | 38 |
| 5.6 | Skizze eines ACBPM-Anwendungsfalls | 40 |
| 5.7 | DPMS-Datenobjektskizze | 42 |
| 5.8 | DPMS-Prozessablaufsskizze | 43 |
| 5.9 | Skizze eines DPMS-Anwendungsfalls | 44 |
| 5.10 | DDPS Modellierungsschritte | 46 |
| 5.11 | Skizze eines DDPS-Anwendungsfalls | 47 |
| 6.1 | Abbildungsbeispiel der hierarchischen Struktur einer Institution | 52 |
| 6.2 | Lebensphasen von Prozessen | 52 |
| 6.3 | Beschreibungssprachen für Prozesselementen | 55 |
| 6.4 | Hierarchie von Aktionen | 57 |
| 6.5 | Aufbau eines Datenobjektes | 58 |
| 6.6 | Benutzerverwaltung für ein Rollensystem | 64 |
| 6.7 | Koordinierung von Duplikaten | 67 |

| | | |
|-----|--|-----|
| 6.8 | Paralleler Zugriff auf Datenobjekten | 68 |
| 6.9 | Neudefinition von Datenobjekt und Anwendungsfunktion | 69 |
| 7.1 | Einordnung der Ansätze in die Paradigmen | 112 |
| 7.2 | Steuerung und Struktur Positionierung der Ansätze | 115 |
| 8.1 | Erweiterung des Spannungsdreiecks zum Viereck | 122 |
| 8.2 | Hauptcharakteristika von inhaltsgesteuerten Prozesssystemen | 124 |
| B.1 | Wie viele Menschen sind in einem Land online | 131 |
| B.2 | Wie viele Stunden sind Internetnutzer online | 132 |
| B.3 | Wie viele Stunden sind Jugendliche im Internet | 132 |
| B.4 | Welche Kommunikationsmittel werden von allen Internetnutzern verwendet | 133 |
| C.1 | BPMN Symbolposter Teil 1 | 136 |
| C.2 | BPMN Symbolposter Teil 2 | 137 |

Tabellenverzeichnis

| | | |
|------|---|----|
| 1.1 | Arten des Schriftverkehrs im medizinischen Umfeld | 2 |
| 5.1 | Zusammenfassung der Steuerung der Ansätze | 48 |
| 6.1 | Aufzählung der Aspekten | 49 |
| 7.1 | Abkürzungen der Ansätze zur Prozessunterstützung | 72 |
| 7.2 | Vergleich: Modellierungsmethoden | 73 |
| 7.3 | Vergleich: Explizite Definition von Lebensphasen in einem Prozess | 74 |
| 7.4 | Vergleich: Analyse der Prozesse | 74 |
| 7.5 | Vergleich: Datenobjekt oder Rekord basierte Datenspeicherung | 74 |
| 7.6 | Vergleich: Zusammenführung von Nutzdaten, Verwaltungsdaten, Verwaltungssystem | 75 |
| 7.7 | Vergleich: Kommunikationsmittel für soziale Interaktion | 75 |
| 7.8 | Vergleich: Natürliche Sprache als Prozess-Beschreibungssprache | 76 |
| 7.9 | Vergleich: Anpassbare Benutzeroberfläche | 76 |
| 7.10 | Vergleich: Benutzerinteraktion | 77 |
| 7.11 | Vergleich: Ereignisbehandlung | 78 |
| 7.12 | Vergleich: Online- / Offline- Modus | 78 |
| 7.13 | Vergleich: Echtzeitfähigkeit von Anwendungsfunktionen | 78 |
| 7.14 | Vergleich: Abhängigkeit von Anwendungsfunktion | 79 |
| 7.15 | Vergleich: Granularität von Anwendungsfunktionen | 79 |
| 7.16 | Vergleich: Wiederverwendbarkeit von Prozesselementen | 80 |
| 7.17 | Vergleich: Langzeiteinsatzfähigkeit | 80 |
| 7.18 | Vergleich: Aktive Datenobjekte | 81 |
| 7.19 | Vergleich: Inhalt: Nutz- und Verwaltungsdaten | 82 |
| 7.20 | Vergleich: Identifikation von Daten | 82 |
| 7.21 | Vergleich: Speicherung der Prozessverlaufsinformationen in den Daten | 82 |
| 7.22 | Vergleich: Anzahl der Datenobjekte für einen Anwendungsfall | 83 |
| 7.23 | Vergleich: Einheit der Validierung | 83 |

| | | |
|------|--|-----|
| 7.24 | Vergleich: Daten- oder Aktivitätssteuerung | 84 |
| 7.25 | Vergleich: Datenlebenszyklus | 85 |
| 7.26 | Vergleich:Offenes Ende oder Terminierung einer Prozessinstanz | 85 |
| 7.27 | Vergleich: Echtzeitfähigkeit von Prozessen | 85 |
| 7.28 | Vergleich: Transparenz | 86 |
| 7.29 | Vergleich:Inter-institutionelle Prozessausführung | 86 |
| 7.30 | Vergleich: Kommunikationsnetz | 87 |
| 7.31 | Vergleich: Spezialgeräte (Pads, Mobiltelefone) | 87 |
| 7.32 | Vergleich: Kenntnis der Systemarchitektur | 87 |
| 7.33 | Vergleich: Regelsystem | 89 |
| 7.34 | Vergleich: Sicherheitssysteme | 89 |
| 7.35 | Vergleich: Benutzerverwaltung | 90 |
| 7.36 | Vergleich: Protokollierungssystem | 90 |
| 7.37 | Vergleich: Ein- und Ausgabeverwaltung | 90 |
| 7.38 | Vergleich: Anmerkungs-system | 91 |
| 7.39 | Vergleich: Lokalisierung von Anwendungsfunktionen | 91 |
| 7.40 | Vergleich: Koordinierung von Daten | 91 |
| 7.41 | Vergleich: Versionsverwaltung | 91 |
| 7.42 | Vergleich: Fehlererkennung | 91 |
| 7.43 | Vergleich: Fehlerkorrektur | 92 |
| 7.44 | Vergleich: Verklemmungsvermeidungssystem | 92 |
| 7.45 | Vergleich: Anpassungsfähigkeit des Datenschemas zur Laufzeit | 93 |
| 7.46 | Vergleich: Anpassungsfähigkeit der Anwendungsfunktionen zur Laufzeit . | 94 |
| 7.47 | Vergleich: Anpassungsfähigkeit der Anzahl von Anwendungsfunktionen zur Laufzeit | 94 |
| 7.48 | Vergleich: Anpassungsfähigkeit der Benutzerverwaltung zur Laufzeit . . . | 94 |
| 7.49 | Vergleich: Anpassungsfähigkeit des Regelsystems zur Laufzeit | 95 |
| 7.50 | Vergleich: Anpassungsfähigkeit des Routingsystems zur Laufzeit | 95 |
| 7.51 | Vergleich: Anpassungsfähigkeit des Protokollierungssystems zur Laufzeit . | 95 |
| 7.52 | Symbolbeschreibung für den tabellarischen Vergleich | 106 |
| 7.53 | Zusammenfassung des gesamtmethodischen Vergleichs | 106 |
| 7.54 | Zusammenfassung des funktionalen Vergleichs | 107 |
| 7.55 | Zusammenfassung des informativen Vergleichs | 107 |
| 7.56 | Zusammenfassung des dynamischen Vergleichs | 107 |
| 7.57 | Zusammenfassung des integrativen Vergleichs | 108 |

| | |
|---|-----|
| 7.58 Zusammenfassung des technische-operativen Vergleichs | 108 |
| 7.59 Zusammenfassung des software-operativen Vergleichs | 108 |
| 7.60 Zusammenfassung des evolutionären Vergleichs | 109 |

Abkürzungsverzeichnis

| | |
|--------------|---|
| ACBPM | Artifact-Centric Business Process Models |
| ACID | Atomicity, Consistency, Isolation, Durability |
| ACM | Adaptive Case Management |
| BPEL | Business Process Execution Language |
| BPMN | Business Process Modelling and Notation |
| BPMS | Business Process Management-System |
| CH | Case Handling |
| CM | Case Management |
| CSCW | Computer Supported Cooperative Work |
| CT | Computertomograph |
| DBS | Datenbanksystem |
| DDPS | Data-driven Process Structures |
| DPMS | Datenorientiertes Prozess-Management-System |
| EPK | Ereignisgesteuerten Prozessketten |
| ERH | Electronic Health Record |
| GUI | Graphical User Interface |
| ID | Identifizierungsnummer |
| IM | Instant Messenger |
| MRT | Magnetresonanztomograph |

| | |
|-------------|---|
| OCR | Optical Character Recognition |
| OMG | Object Management Group |
| PDA | Personal Digital Assistant |
| SBVR | Semantics of Business Vocabulary and Business Rules |
| SzP | System zur Prozessunterstützung |
| PKI | Public Key Infrastructure |
| TAV | Transaktionsverwaltung |
| WfMS | Workflow-Management-Systeme |

1 Einleitung

Inter-institutionelle Arbeitsabläufe im medizinischen Umfeld werden heutzutage noch immer nur teilweise durch moderne Softwaresysteme unterstützt. Speziell bei dem Austausch von Patienteninformationen zwischen den einzelnen Institutionen besteht ein Defizit. In Abhängigkeit von der jeweils vorliegenden Erkrankung eines Patienten und der dazugehörigen medizinischen Behandlung, wird dieser gegebenenfalls fallspezifisch von mehreren medizinischen Fachabteilungen betreut. Dieser Prozess erfordert einen Austausch der Patienteninformationen zwischen den jeweiligen Fachabteilungen. Die Behandlung eines Patienten mittels eines Prozessunterstützungssystems abzubilden, parallel medizinische Informationen aufzunehmen und diese denjenigen wieder zur Verfügung zu stellen, die sie benötigen, kann die Effizienz der einzelnen Institutionen steigern. Ziel dieser Arbeit ist die Recherche und die vergleichende Evaluation von verfügbaren Ansätzen für *content-oriented* Workflows. Sie ist Teil des ProMed Forschungsprojekts zur Entwicklung eines lose gekoppelten Prozessunterstützungssystems für den medizinischen Bereich.

1.1 Motivation und Herausforderung

Die Optimierung des Prozessablaufes führt zu einer Verbesserung des Informationsaustausches zwischen den beteiligten Institutionen im medizinischen Umfeld. Die klassische Vorgehensweise basiert auf einer lokalen Informationsverarbeitung und bleibt daher hinter den Möglichkeiten zurück, die ein modernes System zur Prozessunterstützung mit sich bringt. Die medizinische Behandlung, die normalerweise aus diagnostischen und therapeutischen Zyklen (Abbildung 1.1) besteht, generiert in jedem Schritt Information, auf denen basierend die Ärzte über den Fortgang der Behandlung entscheiden. Dieser Prozess ist zu Beginn der medizinischen Behandlung nicht vollständig planbar. Daher muss ein Informationsverarbeitungssystem in der Lage sein, sich dem Behandlungsverlauf anzupassen.

Durch den Einzug von digitalen Informationsverarbeitungssystemen im medizinischen Umfeld, erscheint die klassische papierbasierte Informationsverarbeitung antiquiert.

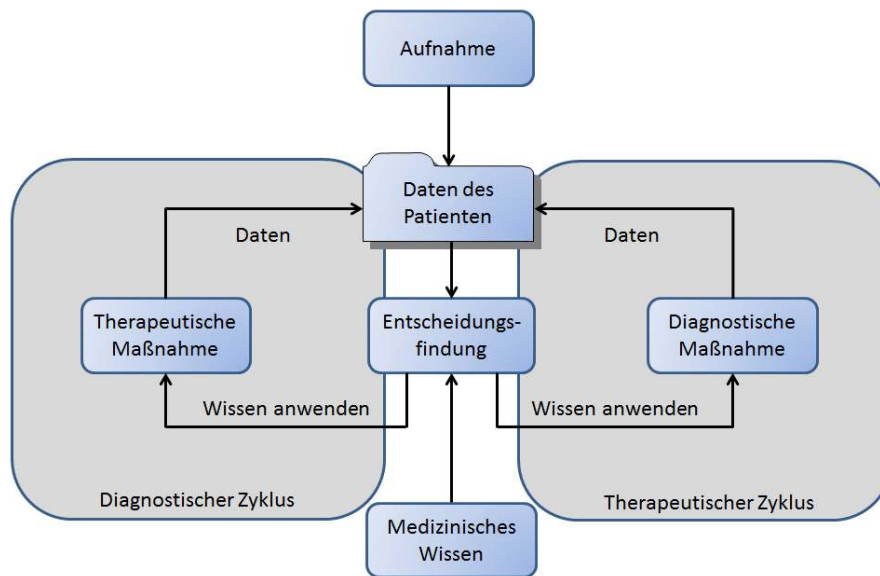


Bild 1.1: Schematische Darstellung eines Behandlungsablaufs in der Medizin

Ohne Vorliegen einer klaren Strukturierung des Informationsverlaufs und definierten Verantwortungsbereichen können unterschiedliche Möglichkeiten der Informationsübertragung verwendet werden. Die Tabelle 1.1 visualisiert die verschiedenen Formen der Informationsweitergabe im medizinischen Umfeld. Aufgrund fehlender einheitlicher Vorschriften können Informationen beispielsweise verloren gehen oder treffen nicht rechtzeitig bei den jeweiligen Empfängern ein. Ein weiterer Nachteil ist die mögliche Fragmentierung der Patienteninformationen.

| Daten | Beispiele |
|------------------------------------|--------------------|
| unstrukturiert, nicht elektronisch | Brief, Notiz, Bild |
| strukturiert, nicht elektronisch | Formular |
| unstrukturiert, elektronisch | E-Mail, Text-Datei |
| strukturiert, elektronisch | Online-Formulare |

Tabelle 1.1: Arten des Schriftverkehrs im medizinischen Umfeld

Am Beispiel eines Beinbruchs skizziert die Abbildung 1.2 schematisch die unterschiedlichen Stationen einer medizinischen Behandlung. Auf jeder Station fallen mehrere Informationen an, die gespeichert und wenn notwendig weiteren, an der Behandlung beteiligten Stationen, zur Verfügung gestellt werden müssen. Nach jedem Prozessschritt entscheidet ein Arzt über den Fortgang der Behandlung. Diese Entscheidung basiert im Wesentlichen auf den vorliegenden Informationen.

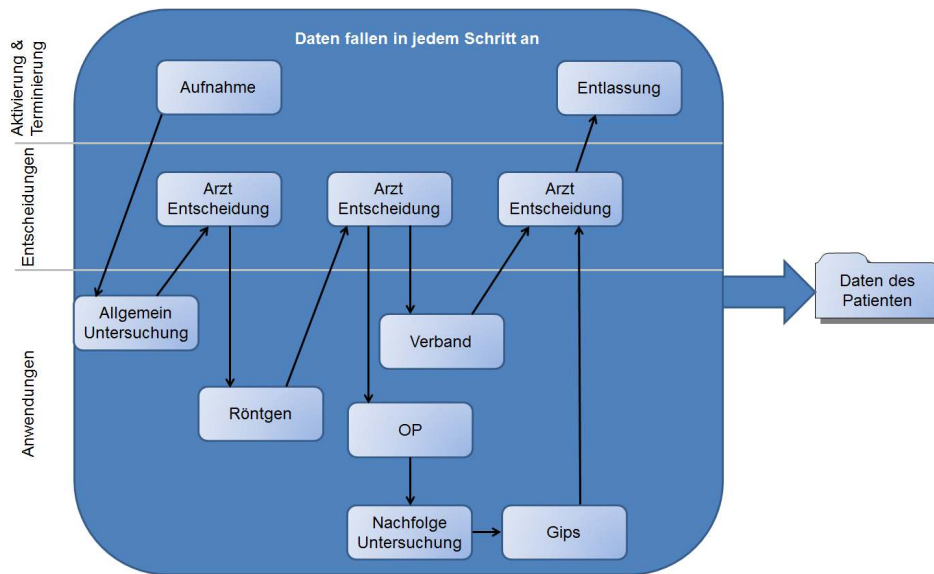


Bild 1.2: Skizze eines Beinbruchbehandlung

Liegen die Information nicht rechtzeitig und/oder nur fragmentiert vor, kann dies die Entscheidungen des Arztes beeinflussen. Dies kann zusätzliche Kosten verursachen und schlimmstenfalls den Genesungsprozess eines Patienten beeinträchtigen. Durch die Integration aller Daten und durch Versenden der Patientendaten auf geschützten Kommunikationskanälen, kann dies mit Hilfe eines Systems zur Prozessunterstützung effektiv verhindert werden.

1.2 Ziele der Arbeit

Das Ziel dieser Arbeit ist die Recherche und die vergleichende Evaluation von existierenden *content-oriented* Workflows. Besonderer Wert wird dabei auf die Leistungsfähigkeit der einzelnen Ansätze zur Prozessunterstützung und deren Evaluation in Bezug auf einen möglichen Einsatz im medizinischen Umfeld gelegt.

Im medizinischen Umfeld existieren verschiedene Prozesse, die für sich betrachtet immer einen eigenen Zweck erfüllen, zusammen aber ein gemeinsames Ziel verfolgen. Die in den Laboratorien, in den Apotheken und von den Ärzten angewandten Prozesse können als Versorgungsprozess eines Patienten zusammengefasst werden. Der Versorgungsprozess beschreibt diejenigen Maßnahmen, die zur Genesung eines Patienten oder zur Erhaltung seiner Gesundheit notwendig sind. Soll der Patient in einem Informationssystem abgebildet werden, dann müssen hierfür sowohl die persönlichen Daten als auch die Krankheitsgeschichte des Patienten als Input zur Verfügung stehen. Aus diesem

Grund steht der Patient im Mittelpunkt und dieser wird mittels seiner persönlichen Daten repräsentiert, die den Prozessverlauf steuern [NL10]. Bei der Recherche geeigneter Ansätze wird bei der vergleichenden Evaluation auf die *content-oriented* Workflows eingegangen. Dabei werden die allgemeinen Eigenschaften der Ansätze sowie die besonderen Eigenschaften der jeweils verfügbaren Umsetzungen erörtern.

Diese Arbeit ist Teil des ProMed Forschungsprojektes des Lehrstuhls 6 Informatik (Datenmanagement) der Friedrich-Alexander-Universität Erlangen-Nürnberg. Eine Teilaufgabe der ProMed Forschungsgruppe ist die Entwicklung von α -Flow. α -Flow ist ein Prozess-Ansatz, der die Abläufe im medizinischen Umfeld unterstützen soll. Im Zuge dieser Arbeit werden aktuelle, verwandte Ansätze mit dem als Referenz dienenden α -Flow Ansatz verglichen und die Eigenschaften aller Ansätze kritisch bewertet.

Die Daten in *content-oriented* Workflow-Instanzen besitzen einen Lebenszyklus, der von der Instanziierung bis zum Ende der Ausführung reicht. Dabei erfahren sie Veränderungen mittels der Aktivitäten, die auf sie einwirken. Diese Änderungen werden als Zustandsübergänge bezeichnet. Ein Zustand beschreibt eine Konfiguration von Werten in den Attributen einer Prozessinstanz. Der Lebenszyklus kann formal definiert werden, wodurch er für die Konsistenzprüfung verwendbar wird [GS10] [RKG07a]. Die formale Beschreibung und die damit verbundene Konsistenzprüfung von Prozessmodell und Daten sind Eigenschaften, die für den α -Flow-Ansatz adaptiert werden können. Aus diesem Grund wird der Lebenszyklus besonders betrachtet.

Im medizinischen Umfeld wird von Richtlinien für Entscheidungen gesprochen, die den Entscheidungsprozess qualitativ hochwertig unterstützen sollen [BCM⁺10]. Der Entscheidungsprozess wird im Wesentlichen durch die medizinisch relevanten Daten gelenkt. Diese Daten werden in einem Prozess während seines Ablaufs von unterschiedlichsten Informationssystemen bearbeitet, die je nach Prozessverlauf an einer Prozessinstanz partizipieren können. Aus diesem Grund müssen heterogene Informationssystemkomponenten adäquat gekoppelt werden. Da autonome Systeme nicht vermeidbar sind, wird ein Weg benötigt, medizinische Informationssysteme lose zu koppeln [Len09]. Deshalb wird untersucht, wie sich adaptive Systeme bei Veränderungen der Rahmenbedingungen der Außenwelt oder systembedingten Veränderungen bei der jeweiligen Kopplungsmethode zwischen Daten und Aktivitäten verhalten.

Ein weiterer Fokus der vergleichenden Evaluation liegt auf der Anpassungsfähigkeit der Ansätze an geänderte Rahmenbedingungen der Außenwelt. Die Anpassungsfähigkeit wird auch als Adaptivität bezeichnet. Die Traditionellen Ansätze müssen sich dynamisch an neue Rahmenbedingungen anpassen können [CWW⁺06] [LHHM05], um diese An-

forderung zu erfüllen. Bei der Anpassungsfähigkeit wird die Integration von anderen medizinischen Informationssystemen, wie beispielsweise Computertomograph (CT) oder Magnetresonanztomograph (MRT), nicht berücksichtigt. Die zum Vergleich herangezogenen prozessunterstützenden Ansätze müssen statt einer Schnittstelle zu medizinischen Spezialgeräten, die Daten von diesen Geräten aufnehmen können.

Institutionen, die sich schnell an veränderte Situationen anpassen müssen, das gilt insbesondere für den medizinischen Bereich, beschreiben Scheer und Köppen [AWS01] als lebendigen Organismus, „der sich ändert bzw. verändern muss, um weiterhin lebensfähig zu bleiben“. Deshalb wird die Möglichkeit des medizinischen Personals, Veränderungen an einem System zur Prozessunterstützung vorzunehmen, ebenfalls im Laufe des Vergleiches bewertet. Rechnergestützte Informationssysteme werden als Mensch-Aufgabe-Techniksysteme verstanden, die die Eigenschaften besitzen sollen, den Nutzer bei seiner Arbeit zu unterstützen.

1.3 Abgrenzung

Es existieren viele vergleichende Evaluationen von Systemen zur Prozessunterstützung, die einzelne Teilaspekte, wie die Meta-Ebene von Instanzen [RM98], die Workflow-Engine [HPA05] oder die Konsistenz von dynamischen Modellen [RRD03] berücksichtigen. Zu den Systemen der Prozessunterstützung gehört auch Computer Supported Cooperative Work (CSCW). In die gleiche Kategorie wie CSCW kann auch Groupware eingeordnet werden. Groupware gehört zu denjenigen Systemen, die Informationen mittels kooperativer Arbeit verarbeiten [Joh88]. Diese Ansätze stehen nicht im Focus der Betrachtung.

Im medizinischen Umfeld existieren viele Untersuchungen, die Informationssysteme evaluieren [AGH⁺03] [ABN⁺04] [KP04]. Allerdings werden hierbei nur Teilprozesse betrachtet. Ganzheitliche Vergleiche von Prozessunterstützungssystemen in der Medizin, deren primäres Ziel die Definition von Richtlinien für Informationssysteme im medizinischen Umfeld ist [ADLP⁺06], existieren nur wenige.

Es existieren viele unterschiedliche Ansätze, die sich mit der inhaltsbasierten Steuerung eines Prozesses befassen. Entsprechend der Struktur der Aktivitäten und der Daten können sie in separate Kategorien eingeteilt werden [AWG05b]. Diese Arbeit nimmt Bezug auf lose gekoppelte, kooperative Arbeit unterstützende und adaptive Ansätze zur Prozessunterstützung. Deren Eigenschaften werden dann in Bezug auf den möglichen Einsatz im medizinischen Umfeld bewertet. Abbildung 1.3 beschreibt nach [RRA03] die unterschiedlichen Prozess-Kategorien.

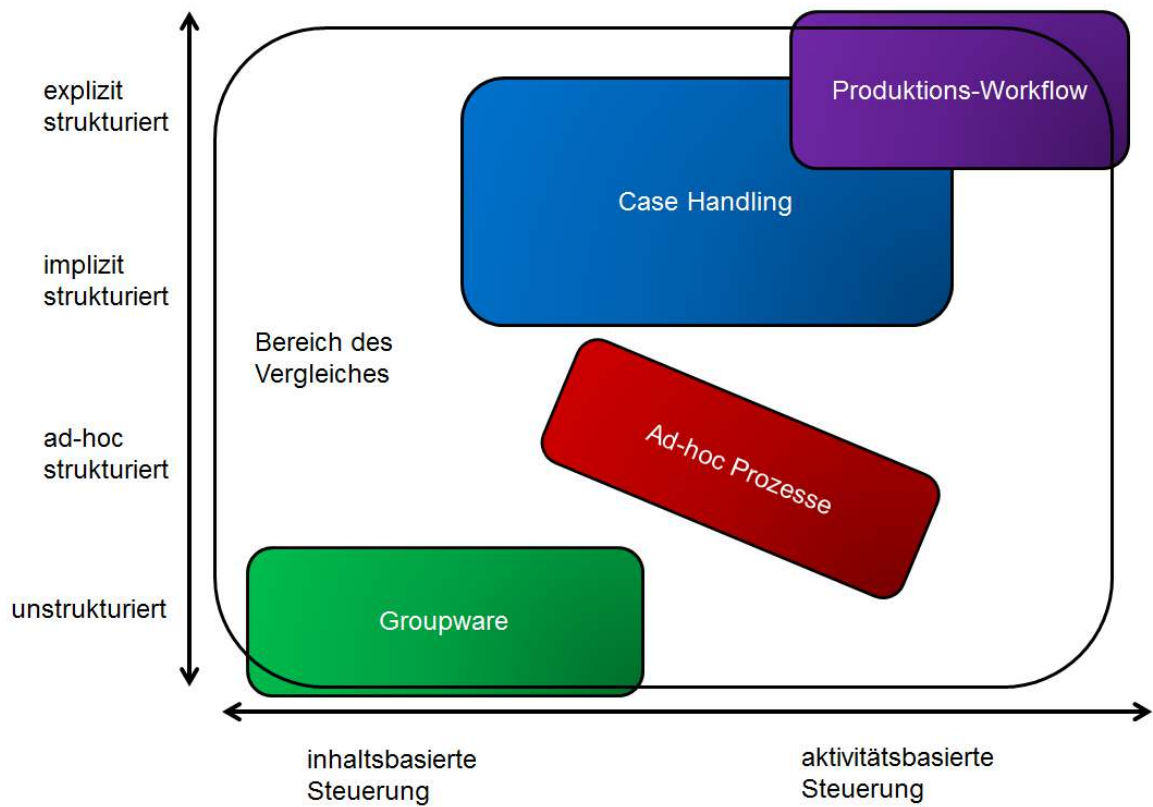


Bild 1.3: Positionierung von Systemen zur Prozessunterstützung nach [RRA03]

2 Methodik

Die vorliegende Arbeit befasst sich mit der Recherche und der vergleichenden Evaluation von existierenden Ansätzen des *content-oriented* (deutsch: inhaltsbasierte) Workflows. In Kapitel 3, Grundlagen, werden die drei unterschiedlichen Paradigmen zur Prozessunterstützung: das inhaltsbasierte, das aktivitätsbasierte und das Groupware Paradigma, miteinander verglichen. Die gesammelten Informationen basieren auf einer explorativen Literaturrecherche. Nach der Beschreibung der Paradigmen wird thematisiert, wie die jeweiligen Paradigmen zusammenhängen. Daraufhin wird ihre mögliche Verwendung im medizinischen Umfeld untersucht und diskutiert, welche vor- und nachteiligen Eigenschaften die Paradigmen im Bezug zum medizinischen Umfeld jeweils besitzen. Zusätzlich werden in diesem Kapitel Begrifflichkeiten, die im weiteren Verlauf der Arbeit relevant sind, definiert.

Neben der vergleichenden Evaluation der Prozessunterstützungsansätze werden diese dahingehend bewertet, in weit sie im medizinischen Umfeld eingesetzt werden können. α -Flow ist ein Prozessunterstützungsansatz, der für den medizinischen Bereich von der Forschungsgruppe ProMed des Informatik 6 Lehrstuhls (Datenmanagement) der Friedrich-Alexander-Universität Erlangen-Nürnberg entwickelt wird. Er beschreibt einen Prozess-Ansatz, in dem verschiedene Paradigmen (im Wesentlichen das inhalts- und das aktivitätsbasierte Paradigma) zu einem Modell fusionieren. Wie mehrere Paradigmen zusammen in einem Prozess-Ansatz integriert werden, wird in Kapitel 4 am Beispiel von α -Flow beschrieben.

In Kapitel 3 und 4 wird eine ganzheitliche Sicht auf die drei Paradigmen beschrieben. Ab Kapitel 5 erfolgt eine spezifische Beschreibung der Prozessunterstützungsansätze. Für den Vergleich müssen unterschiedliche Prozessunterstützungsansätze untersucht werden. Als Kriterium für die Auswahl der Ansätze zur Prozessunterstützung, die für den Vergleich herangezogen werden dient, das inhaltsbasierte Paradigma. Zusätzlich wird aus Vergleichsgründen ein Prozessunterstützungsansatz ausgewählt, der das Auswahlkriterium inhaltsbasiertes Paradigma nicht erfüllt. Mittels dieses Prozessunterstützungsansatzes wird der Unterschied zu den inhaltsbasieren Ansätzen verdeutlicht. Als

sekundäres Auswahlkriterium werden die Prozessunterstützungsansätze auf der Basis von unterschiedlichen Realisierungen der Lebenszyklen der Daten in einem Ansatz bewertet.

Das Kapitel 5 fasst prägnant die Ansätze zur Prozessunterstützung entsprechend ihrer wesentlichen Merkmale zusammen. Dabei wird auf die Kerneigenschaften der jeweiligen Prozessunterstützungsprogramme, die für die vergleichende Evaluation notwendig sind, eingegangen. Nicht alle Prozessunterstützungsprogramme beschreiben Implementierungseigenschaften. Aus diesem Grund wird bei der weiteren Diskussion der Ansätze auf Implementierungsdetails nicht mehr eingegangen. Abschließend wird zur besseren Darstellung ein Fallbeispiel jedes Ansatzes beschrieben.

Als Grundlage der vergleichenden Evaluation der Prozessunterstützungsansätze werden in Kapitel 5 die Kerneigenschaften der Prozessunterstützungsansätze extrahiert und als Vergleichskriterium aufgearbeitet. Zuerst werden Synonyme für die jeweilig verwendeten Begriffe beschrieben und anschließend vereinheitlicht. Zum Zwecke der Übersichtlichkeit werden die Vergleichskriterien in acht Kategorien eingeordnet. Die acht Kategorien werden beschrieben und es wird dargelegt, welche Vergleichskriterien in die jeweilige Kategorie eingeordnet werden.

Die vergleichende Evaluation wird in Kapitel 7 durchgeführt. Zu Beginn dieses Kapitels werden die Rahmenbedingungen und der Aufbau des Vergleichs beschrieben. Danach wird ein Vergleich der Prozessunterstützungsansätze aus dem Kapitel 5 anhand der Vergleichskriterien aus Kapitel 6 durchgeführt und die erzielten Ergebnisse bewertet. Dabei wird der Vergleich auf der Basis der acht Kategorien strukturiert. Unterstützend werden Tabellen eingesetzt, die die Vergleichsergebnisse zusammenfassen. Anschließend folgt eine Bewertung der Prozessunterstützungsansätze basierend auf deren Stärken und ihren jeweiligen Schwächen. Die Ergebnisse des Vergleichs ermöglichen die Einordnung der Ansätze nach den verschiedenen Paradigmen.

Die Prozessunterstützungsansätze werden unabhängig von einer fachlichen Anwendungsdomäne verglichen. Kapitel 8 ergänzt die Diskussion durch die Betrachtung der jeweiligen Leistungsfähigkeit im medizinischen Umfeld. Der allgemeine Vergleich wird Fragen offen lassen, die auch durch den direkten Bezug auf das medizinische Umfeld nicht abschließend beantwortet werden können. Zusätzlich werden anhand der gewonnen Erkenntnisse Vorschläge unterbreitet, wie z.B. durch eine Verbesserung der Anforderungen an die Ansätze zur Prozessunterstützung, diese optimaler im medizinischen Umfeld eingesetzt werden können. Die hierbei gemachten Vorschläge basieren im Wesentlichen auf dem Vergleich im zweiten Abschnitt in Kapitel 8. In Kapitel 9, Zusammenfassung, werden die Arbeit und speziell die erzielten Ergebnisse der Evaluation zusammengefasst.

3 Grundlagen: Arbeitsabläufe, Inhalte, Kooperationen im medizinischen Kontext

Wie in vielen anderen Geschäftsbereichen, müssen sich auch im medizinischen Bereich die Geschäftsprozesse immer an die vorliegenden Rahmenbedingungen anpassen. Allgemein wird versucht den jeweiligen Arbeitsablauf (engl: *Workflow*) zu automatisieren. Ein **Workflow** ist eine automatisierte, in Algorithmen gegossene, ablaufende Gesamtheit von Aktivitäten, die sich auf Teile eines gesamten Anwendungsprozesses bezieht [Ste98]. Das Organisieren, Planen, Entscheiden, Kontrollieren, Steuern und Führen von Workflows sowie das Ausüben dieser Tätigkeiten, unterliegt dem Workflow-Management-System oder in Kurzform **Worklow-Management** [OBS97]. Ein Workflow-Management besteht in der Regel aus zwei Komponenten. Erstens: Einer Modellierungskomponente für die Beschreibung der Anwendungen in einem Anwendungsprozess und zweitens: aus einem Monitoring bzw. Regelsystem, das die Ausführungszeit des Workflows überwacht [JBS97] [Coa99]. Bei der Implementierung wird das Workflow-Management in **Workflow-Elemente** abgebildet. Die Abbildung 3.1 skizziert die Komponenten eines klassischen Workflow-Systems.

Zusätzlich zu den klassischen Eigenschaften sollen Workflow-Technologien die Modellierung von Geschäftsabläufen (Business Prozesse) mittels Softwareunterstützung erleichtern. Des Weiteren sollen Workflow-Technologien deren Modifikation und/oder deren Neudefinierung unterstützen. Zudem sollen Workflow-Management-Systeme (WfMS), die Ausführung einer Workflow-Instanz überwachen [SAMS01].

WfMS sind mittlerweile mächtige Werkzeuge für die computerunterstützte Ausführung von Geschäftsprozessen. Letztere können unabhängig von spezifischen Anwendungen modelliert, ausgeführt und überwacht werden. Trotzdem existieren auf dem Software-Markt noch zahlreiche Anwendungen mit im Code fest verdrahteter Prozesslogik. Die Ursache dafür, dass konventionel-

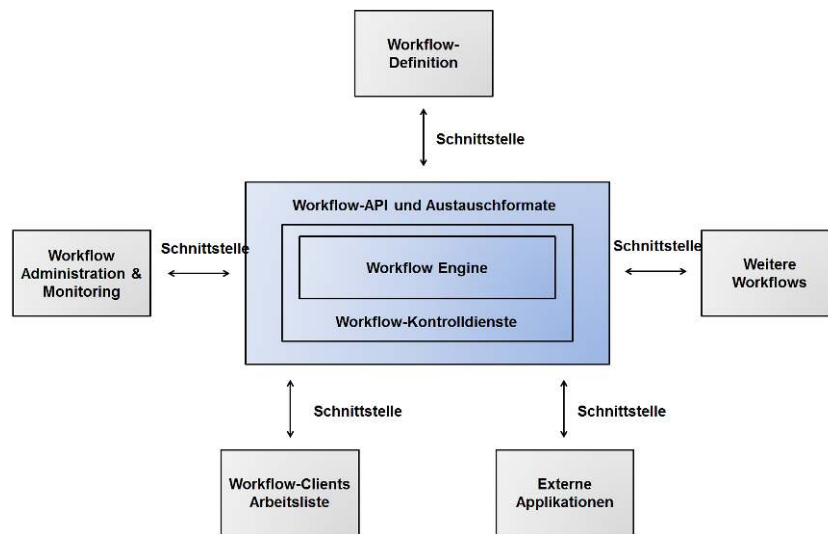


Bild 3.1: Skizze eines Workflow-Systems nach [ST01]

le WfMS bei der Entwicklung prozessorientierter Anwendungen noch nicht die erhoffte Verbreitung erfahren haben, ist zum einen auf ihre Architektur und zum anderen auf ihr aktivitätsorientiertes Paradigma zurückzuführen. [KR09a]

Ein Workflow-System arbeitet nach einem bestimmten Paradigma. Die verschiedenen Paradigmen werden in den folgenden Abschnitten beschrieben. In Abschnitt 3.2 wird beschrieben, welche Eigenschaften ein aktivitätsbasiertes Paradigma erfüllt. Der darauf folgende Abschnitt 3.3 beschreibt die Eigenschaften des inhaltsbasierten Paradigmas. Ein Workflow wird im Regelfall nicht nur von einer Person verwendet, sondern von gleich mehreren Anwendern, die auf die Informationen der jeweils anderen Anwender angewiesen sind.

Groupware stellt ein Konzept der Gruppenarbeit dar. In Abschnitt 3.4 wird dargestellt, wie Groupware-Eigenschaften in einem Workflow partizipieren. Des Weiteren werden die Ergebnisse dieser Untersuchungen im Absatz 3.5 zusammengefasst und analysiert, in wie weit die unterschiedlichen Ansätze im medizinischen Bereich angewendet werden können. Abschließend wird im Abschnitt 3.7 ein eigenes Schema zur Einordnung der Ansätze zur Prozessunterstützung beschrieben.

3.1 Begriffsdefinition der Anpassbarkeit

Bezüglich der Anpassbarkeit von Prozessen werden in dieser Arbeit, verschiedene Begriffe benutzt. Die Begriffe **adaptiv**, **dynamisch**, **flexible** und **agil** werden gerne als Synonyme verwendet und ihre jeweilige Definition wird nachfolgend in Anlehnung an [Swe10] spezifiziert.

Adaptiv: Veränderung einer Entität, die durch Änderung in der Außenwelt notwendig ist und permanent wird. Die Entität wird an die neuen Bedingungen der Außenwelt angepasst, um sie effizient und effektiv abzubilden. Die Entität wird durch sich selbst verändert und nicht durch eine externe Kraft.

Dynamisch: Eine Änderung von Parametern oder Bedingungen, die in Echtzeit ablaufen.

Flexibel: Die Möglichkeit der Veränderung eines Objektes während der Laufzeit.

Agil: Die Fähigkeit etwas schnell zu erledigen, ohne große Vorbereitung.

Aus der Einleitung wurde ein Workflow wie folgt definiert.

Allgemein wird versucht den jeweiligen Arbeitsablauf (engl: *Workflow*) zu automatisieren. Ein **Workflow** ist eine automatisierte, in Algorithmen gegossene, ablaufende Gesamtheit von Aktivitäten, die sich auf Teile eines gesamten Anwendungsprozesses bezieht [Ste98].

Ein Workflow beschreibt danach eine Sequenz von Anwendungsprozessen. Eine Sequenz von Anwendungsprozessen impliziert das aktivitätsbasierte Paradigma. Das aktivitätsbasierte Paradigma wird im folgenden Abschnitt beschrieben. Für den weiteren Verlauf der Arbeit, soll der Begriff Workflow nur noch für aktivitätsbasierte Prozessunterstützungsansätze verwendet werden. Aus diesem Grund wird alternative zu dem Begriff Workflow-Ansatz der Begriff Prozessunterstützungsansätze verwendet. Weitere Begriffsdefinitionen befinden sich in Kapitel A.

3.2 Aktivitätsbasiertes Paradigma

Im Mittelpunkt des aktivitätsbasierten Paradigma stehen die Aktivitäten, die den Prozessunterstützungsablauf steuern. Während der Design-Phase eines Prozessunterstützungssystems wird die Reihenfolge der Aktivitäten festgelegt, die einen Prozessablauf

definieren. Die Aktivitäten enthalten alle Informationen, wie beispielsweise zu welchem Zeitpunkt welche Funktionen durchzuführen sind. Dies kommt der Funktionsweise von Business Process Modelling and Notation (BPMN) [AK10] nahe.

Das aktivitätsbasierte Paradigma beschreibt geregelte und arbeitsteilige Abläufe durch die sukzessive Ausführung von Aktivitäten, die durch einen Sequenzfluss verbunden werden. Dabei werden die Aktivitäten eines Prozesses mittels Stufen und Übergängen beschrieben, wie bei endlichen Automaten oder Petri-Netzen. Die Partizipation von Anwendern ist bei der Ausführung der Anwendungsfunktionen möglich. Allerdings stellt die Unterstützung der Partizipation von Anwendern nur eine untergeordnete Anforderung im aktivitätsbasierten Paradigma dar. Daher werden keine Informationen zur sozialen Interaktion von Anwendern integriert.

Zu jedem Zeitpunkt der Laufzeit ist der Sequenzfluss von Aktivitäten wohldefiniert. Um ein bestehendes Geschäftsziel zu erreichen, ist die Ausführung der Aktivitäten fest vorgegeben. Die Definition zusätzlicher Aktivitäten während der Laufzeit ist ausgeschlossen. Fehlt eine Anwendungsfunktion, dann gilt der Sequenzfluss als nicht existent. Dieses Vorgehen spiegelt das *close world* [Min82] Szenario wieder, nach dem auch Datenbanksysteme (DBS)¹ arbeiten. In einem DBS werden Informationen abgespeichert, die mittels entsprechender Suchfunktionen wieder ausgelesen werden können. Existiert für eine Anfrage kein Ergebnis, dann wird davon ausgegangen, dass die gesuchten Informationen nicht existieren, obwohl diese in der Außenwelt durchaus existieren können.

Der Sequenzfluss der Aktivitäten wird explizit beschrieben. Allerdings existiert kein zentrales Konzept für die Verwendung von Daten. Entsprechend dem Sequenzfluss der Aktivitäten werden die Daten abgebildet. Die Daten selbst können den Verlauf des Sequenzflusses nicht beeinflussen.

¹Bei dem *close world* Konzept wird angenommen, dass Informationen, die nicht im Datenbanksystem (DBS) vorhanden sind, auch nicht existieren [Min82]. Im Vergleich dazu erlaubt das *open world* Konzept die Existenz nicht definierter Informationen.

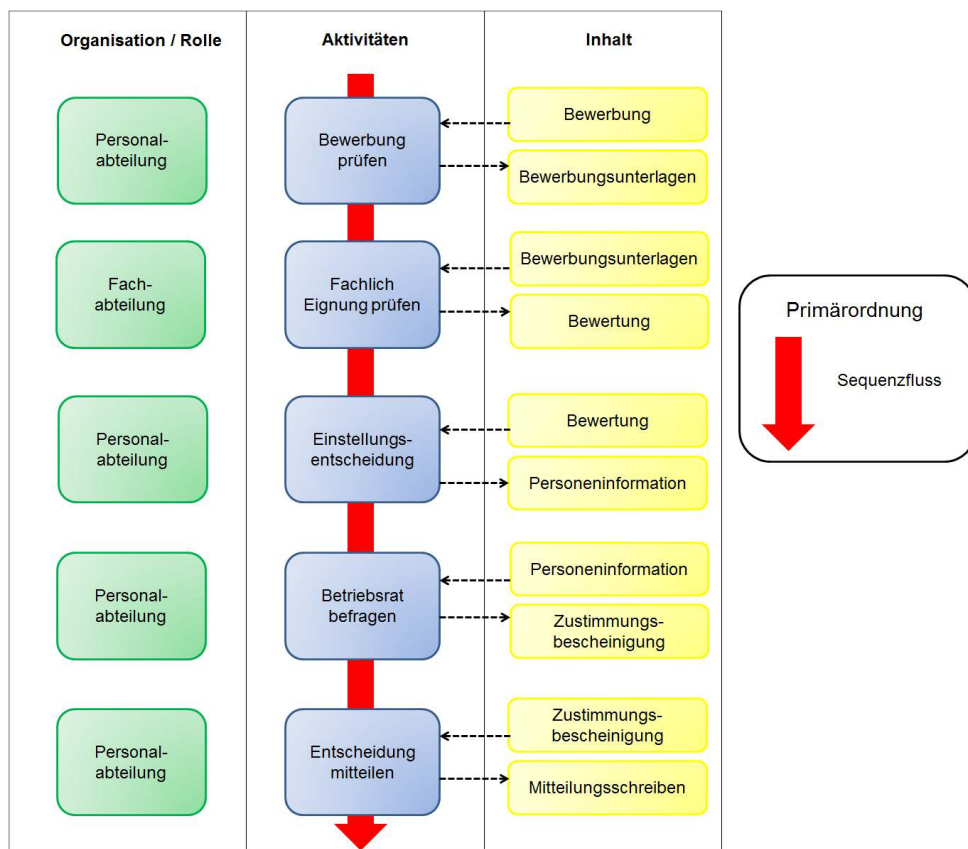


Bild 3.2: Skizze eines Prozessablaufes nach dem aktivitätsbasierten Paradigma

Abbildung 3.2 skizziert ein Bewerbungsbeispiel, bei dem die einzelnen Aktivitäten im Zentrum der Ausführung stehen. Der Sequenzfluss der Aktivitäten verläuft von oben nach unten. Bei der Modellierung werden die Daten nur am Rande betrachtet. Sie werden zwar für die Ausführung beschrieben; allerdings sind sie während der Ausführung für den Entscheidungsprozess nicht relevant. Eine Aktivität sendet die Datenobjekte nicht weiter.

Ein besonderes Merkmal der aktivitätsbasierten Ansätze ist das wiederholte Modellieren derselben Datenobjekte. Dies ist deshalb möglich, weil die Datenobjekte den Aktivitätsfluss lediglich flankieren. Beispielspielerhaft ist dies in der Abbildung 3.2 wiedergegeben.

3.3 Inhaltsbasiertes Paradigma

Das inhaltsbasierte Paradigma steuert einen Prozessunterstützungsansatz anhand der gespeicherten Nutzinformationen, [CH09] [MRH07a] [KR09b]. Die Daten repräsentieren

den Mittelpunkt des inhaltsbasierten Paradigmas. Bei dem Ablauf von prozessunterstützenden Systemen wird ein Datenfluss beschrieben, bei dem die Daten von einem Zustand in den nächsten überführt werden. Bei der Modellierung eines Prozesses werden diejenigen Datenzustände beschrieben, die benötigt werden, um das Geschäftsziel zu erreichen. Die Daten können für bestimmte Entitäten der Außenwelt oder der konzeptionellen Welt strukturiert werden. Somit wird eine Sammlung von Daten immer einer bestimmten Entität zugeordnet.

Im Verlauf eines Prozesses werden die Daten mittels Aktivitäten verändert. Merkmal der inhaltsbasierten Prozessunterstützungsansätze ist das wiederholte modellieren derselben Aktivitäten, wie es in Abbildung 3.3 dargestellt wird, da die Aktivitäten den Datenfluss nur flankieren.

Die Partizipation von Anwendern ist bei der Ausführung der Anwendungsfunktionen möglich. Allerdings stellt die Unterstützung der Partizipation von Anwendern eine untergeordnete Anforderung im inhaltsbasierten Paradigma dar. Daher werden keine Informationen zur sozialen Interaktion von Anwendern integriert.

Das inhaltsbasierte Paradigma verfolgt das *close world* Szenario. Alle Informationen, die nicht in den Daten gespeichert sind, existieren nicht. Aktivitäten können nur mit solchen Daten arbeiten, die auch im System vorhanden sind. Allerdings können Daten, die nicht im Prozessunterstützungssystem integriert sind, in der Außenwelt existieren. Aus diesem Grund besteht die Möglichkeit die vorhandenen Datensätze zu erweitern. Allerdings werden die neuen Attribute erst im Verlauf der Ausführung mit Informationen gefüllt.

Abbildung 3.3 beschreibt den vereinfachten Verlauf einer Bewerbung nach dem inhaltsbasierten Paradigma. Für den Bewerbungsprozess werden die Informationen bezüglich der Bewerbung in den Daten abgebildet. Bei jedem Schritt greifen die Anwendungsfunktionen auf die Daten zu und verändern deren Werte. Dadurch werden die Daten von einem Zustand in den nächsten übertragen. Der Datenfluss von einem Zustand in den nächsten beschreibt den Fortschritt der Ausführung, und welche Aktivität auf den Daten Veränderungen vornehmen darf. Die Daten können zusätzlich Verwaltungsinformationen beinhalten, die beschreiben, welcher Benutzer auf die Daten zugreifen darf und auch welche Anwendungsfunktionen auf die Daten zugreifen dürfen.

Wie der Abbildung 3.3 zu entnehmen ist, verläuft der Datenfluss von oben nach unten. Dabei reagieren Aktivitäten mit den Daten. Die Organisation/die Rollen interagieren nur peripher mit den Daten und nehmen keine Steuerungsfunktion wahr.

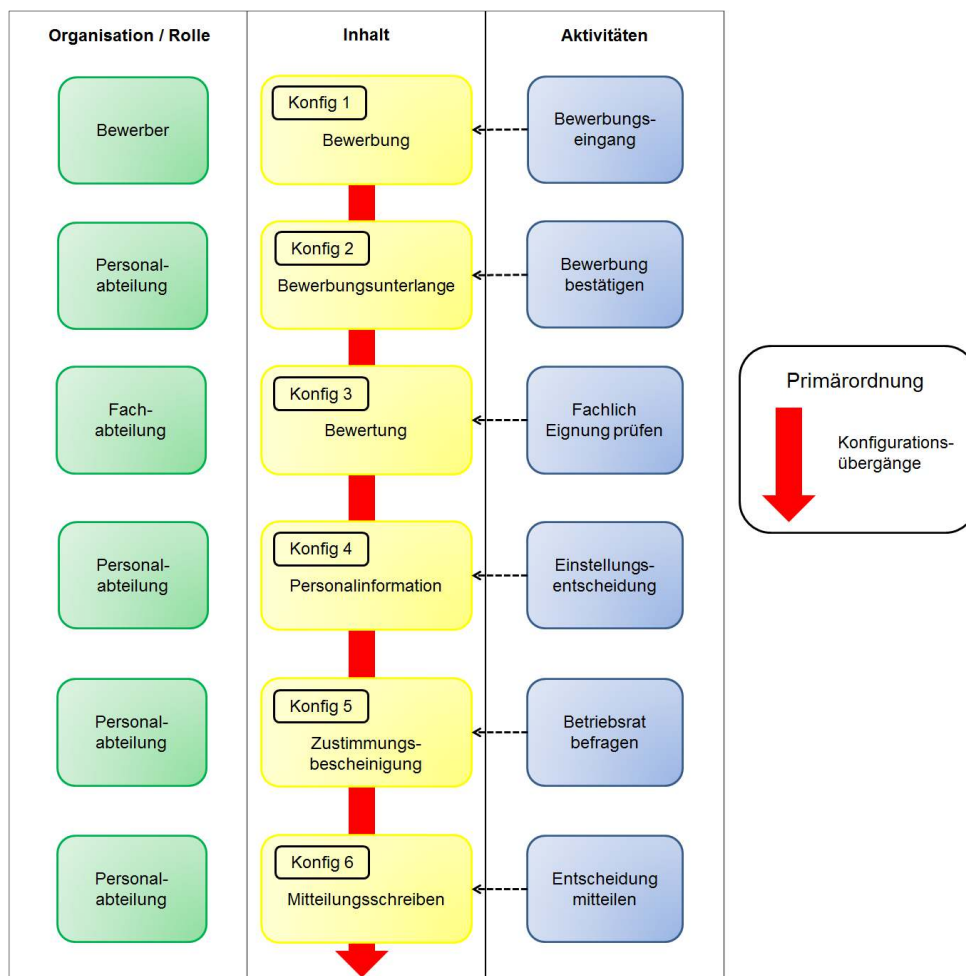


Bild 3.3: Skizze eines Prozessablaufes nach dem inhaltsbasierten Paradigma

3.4 Groupware

Groupware stellt neben der Software-Komponente ein eigenes Forschungsgebiet dar. Groupware und Computer Supported Cooperative Work (CSCW) stellen verwandte Forschungsgebiete dar, die aber getrennt behandelt werden [Gru88]. Welche Eigenschaften Groupware besitzt, wird im nachfolgenden Abschnitt diskutiert. Basierend auf der Kerneigenschaft von Groupware wird daraus ein Paradigma abgeleitet.

3.4.1 Was ist Groupware

Die kooperative Zusammenarbeiten mittels eines Informationssystems in einer Gruppe soll durch Groupware gefördert werden [SRKM01]. Der Begriff Groupware tritt oftmals im Zusammenhang mit CSCW auf. CSCW beschreibt das Forschungsgebiet zur Unter-

stützung von Gruppenarbeit mit Hilfe von Informationssystemen [Wil91] [Hut04]. Im Gegensatz dazu beschreibt Groupware die implementierte Softwarelösung (Abbildung 3.4), für die unterschiedliche Charakterisierungsmöglichkeiten existieren, um kooperative Gruppenarbeit zu unterstützen [Joh88] [Hut04]. Zusammen mit der kooperativen Arbeit steht die Kommunikation zwischen den jeweiligen Anwendern im Vordergrund. Im Vergleich zu DBS, die aus einem zentralen, normalisierten Datenmodell bestehen, basiert Groupware auf elektronischen Nachrichten-Objekten. Das Nachrichten-Objekt basiert auf einer unabhängig verteilten Datenverwaltung [NH99], [SB92]. Einzelne Groupware-Komponenten werden heute schon von vielen Internetnutzern verwendet B.

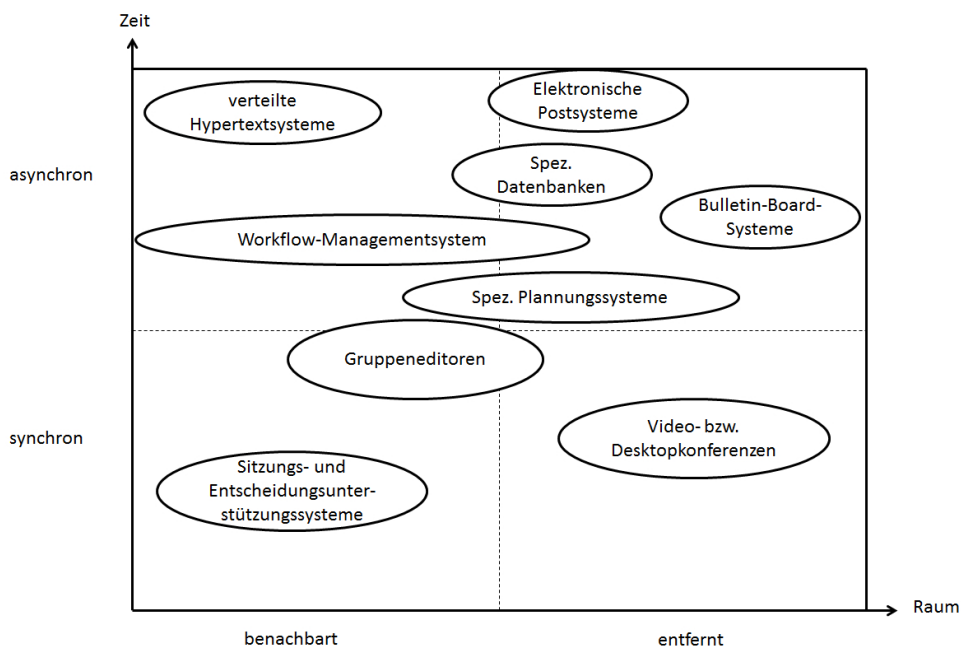


Bild 3.4: Raum-Zeit-Matrix von Bullen und Johansen [BJ88]

Die Abbildung 3.4 zeigt die Raum-Zeit-Matrix von Bullen und Johansen [BJ88]. Mit ihrer Hilfe kann gezeigt werden, wie einzelne Softwarelösungen das kooperative Arbeiten unterstützen können. Die Raum-Zeit-Matrix beschreibt die Einordnung der Groupware-Systemkomponenten nach Zeit und Raum. Die Dimension Zeit beschreibt, ob eine direkte kontinuierliche Kommunikation zwischen den Anwendern stattfindet (synchron) oder ob nur eine zeitlich versetzte Kommunikation (asynchron) möglich ist. Ähnlich kann die Dimension Raum in ‚benachbart‘ und ‚entfernt‘ unterteilt werden. Hiermit wird beschrieben, ob die Anwender in einem Raum zusammen arbeiten oder räumlich getrennt sind. Die entsprechende Einordnung von WfMS ist nicht möglich, denn ein WfMS kann aus vielen unterschiedlichen Systemkomponenten bestehen, wie

beispielsweise aus einem DBS, einem Planungssystem und einer Benutzerverwaltung. Des Weiteren kann ein WfMS sowohl synchron als auch asynchron verwendet werden oder auch ‚benachbart‘ oder ‚entfernt‘, was eine definierte Einteilung in die oben genannten Dimensionen unmöglich macht. Die Erweiterung der Raum-Zeit-Matrix von Grudin [Gru02] mittels der Einführung der Absehbarkeit oder Unabsehbarkeit in die Dimensionen Raum und Zeit bringt ebenfalls keine Verbesserung der Einordnung mit sich.

3.4.2 Groupware Paradigma

Das Groupware Paradigma stellt die kooperative Arbeit in den Vordergrund, da mehrere Anwender eines Prozesses gemeinsam an einer Lösung arbeiten können. Bei diesem Paradigma werden die Anwender eines Prozessunterstützungssystems in den Mittelpunkt gestellt. Wenn mehrere Personen beteiligt sind, entsteht eine Gruppendynamik. Die Bearbeitung eines Groupware-basierten Prozesses, kann daher auch als digitale Gruppenarbeit beschrieben werden.

Eine starre Reihenfolge, nach der die Anwender einen Prozess bearbeiten, existiert nicht. Die Anwender arbeiten frei und greifen selbstständig auf die verfügbaren Daten und die Aktivitäten zu, die sie bearbeiten wollen. Auch muss die Anzahl der Bearbeiter vor der Instanziierung eines Prozesses nicht feststehen, denn während des Prozessablaufes können weitere Personen an diesem partizipieren, die vorher nicht registriert waren. Diese Eigenschaft repräsentiert das *open world* Konzept, denn Personen, die in einem Prozess noch nicht bekannt sind, werden nicht als nicht existent gewertet, weil sie bei der Prozessinstanziierung noch nicht eingetragen waren. Das dynamische Partizipieren an einem Workflow, wird auch als *publish subscribe*-System [EFGK03] bezeichnet.

Die Kontrolle des Prozessablaufes liegt bei den Anwendern selbst. Sie kommunizieren miteinander und tauschen dabei diejenigen Informationen aus, um eine Prozess erfolgreich abzuschließen. Aus diesem Grund sind die Informationen über eine Prozessdomäne getrennt von den Steuerinformationen, die sich die Anwender gegenseitig zu senden.

Abbildung 3.5 veranschaulicht erneut am Beispiel eines Bewerbungsprozess das Groupware Paradigma. Das Hauptaugenmerk liegt auf den Anwendern und deren Kommunikation. Die Kommunikation zwischen den Anwendern verläuft ungeordnet bidirektional. Daten und Aktivitäten stehen links und rechts davon. Sie haben keine Reihenfolge und werden von den Anwendern nach ihrer jeweiligen Aufgabe verwendet. Zwischen den Benutzer können spezifische Aufgaben verteilt werden. Mittels der Kommunikation zwischen den Anwendern wird dabei definiert, wer welche Aktivitäten ausführt.

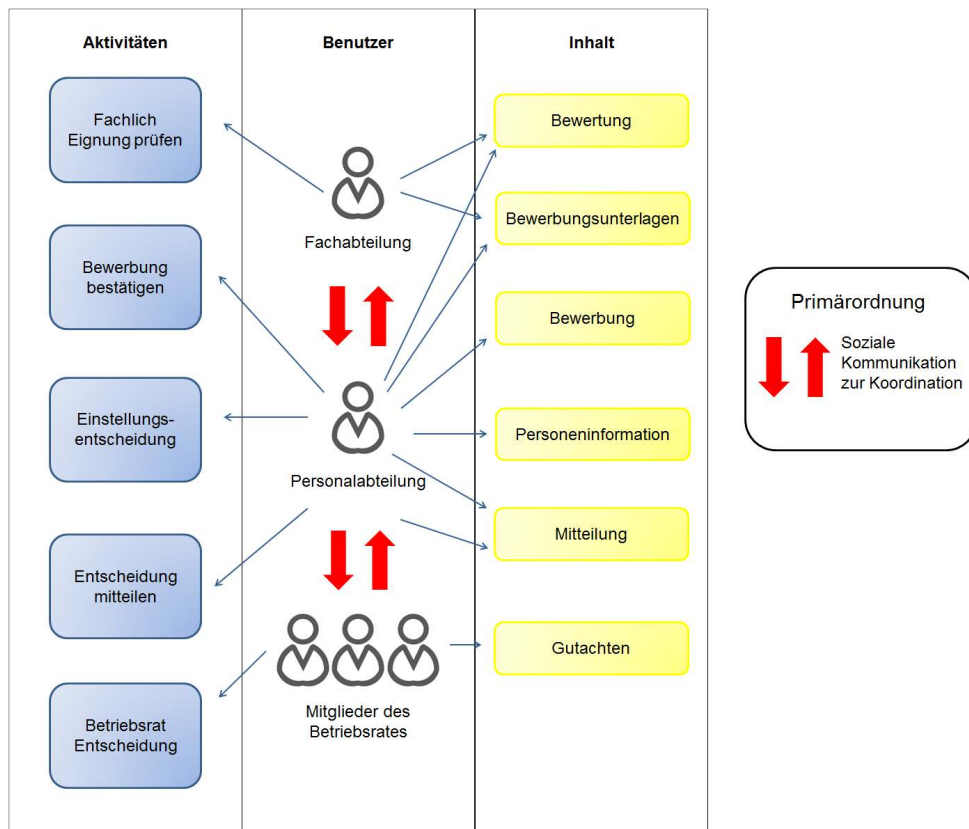


Bild 3.5: Skizze eines Prozessablaufes nach dem Groupware Paradigma

3.5 Pradigmen im Kontex der Medizin

Während eines diagnostischen oder therapeutischen Zyklus fallen Informationen an, die von einem Arzt interpretiert werden. Für die weitere Behandlung werden alle existierenden Informationen der vorlaufenden Diagnosen und/oder Therapien benötigt, die ein Patient durchlaufen hat. Dabei beziehen sich alle Informationen, die während der diagnostischen und therapeutischen Zyklen anfallen, auf jeweils einen Patienten. Neben der direkten Patientenbehandlung existieren noch weitere Prozesse, die die Ausführung der diagnostischen und therapeutischen Zyklen unterstützen. Im Mittelpunkt des Prozessablaufes steht aber immer der Patient und seine personenbezogenen Informationen.

Anhand dieser Informationen treffen Ärzte ihre Entscheidungen über den Fortgang der Behandlung eines Patienten. Dies entspricht dem inhaltsbasierten Paradigma. Bei jedem diagnostischen und therapeutischen Zyklus werden die anfallenden Daten gespeichert. Die Daten werden ad-hoc zum Arzt transferiert, der die Informationen prüft und auf Basis der Informationen entscheidet, welche Behandlung der Patient benötigt. Die

Entscheidung wird zu den anderen Daten persistiert. Da jede Entscheidung auf der Basis der Daten getroffen wird, wird der Verlauf der Behandlung durch die Daten gesteuert. Ein Behandlungsverlauf kann oft nicht auf der Basis der bei der Erstuntersuchung generierten Informationen definiert werden, weil ohne weitergehende Untersuchungen und / oder Behandlungen, zu wenige Informationen vorliegen.

Das Groupware Paradigma stellt die kooperative Arbeit und vor allem die Kommunikation in den Mittelpunkt. Durch die mögliche Parallelverarbeitung der Daten kann ein Arzt Informationen zu einer Diagnose eintragen, während gleichzeitig das Labor beispielsweise die Ergebnisse einer Bluteruntersuchung eingeben kann. Dabei behindern sich die Anwender nicht gegenseitig, vielmehr ergänzen sie sich. Werden von einem Anwender Information eingetragen, dann wird gleichzeitig vermerkt, wer die Informationen hinzugefügt hat. Diese Funktionalität erweitert die bisher etablierte papierbasierte Informationsweitergabe. Sie wird dadurch nicht ersetzt, sondern die anfallenden Informationen werden digitalisiert. Dadurch kann man von einer Dokumentenbasierten Informationsübergabe sprechen, an der die verschiedenen Ärzte, die räumlich verteilt arbeiten, partizipieren können.

3.6 Potpourri der Paradigmen in wissensintensiven, semistrukturierten Prozess in der Medizin

Der Ablauf von diagnostischen und therapeutischen Zyklen beschreibt eine Aneinanderreihung von medizinischen Maßnahmen, die auch als Sequenzfluss von Aktivitäten verstanden wird. Das aktivitätsbasierte Paradigma ist allerdings zu starr und zu unflexibel und kann sich daher nicht an die jeweiligen Entscheidungen der Ärzte anpassen. Eine allgemeine Prozesskontrolle über alle Aktivitäten hinweg kann nur mit großem Aufwand betrieben werden, wenn beispielsweise lose gekoppelte Informationssysteme betrachtet werden sollen.

An der Behandlung eines Patienten sind oft mehrere Ärzte beteiligt. Fallbedingt können sie gemeinsam einen Patienten therapieren und sich die Informationen per Informationssystem zusenden. Dieses Vorgehen spiegelt das Groupware Paradigma wieder. Das Groupware Paradigma folgt keiner festen vorab definierten Organisationsstruktur, sondern strukturiert sich anhand der Interaktion der einzelnen Anwender. Allerdings können bei einem solchem unstrukturierten Prozessablauf Informationen verloren gehen, die gegebenenfalls für die medizinische Behandlung eines Patienten wichtig sind.

Ärzte treffen ihre Entscheidung auf der Basis der Informationen, die aus abgeschlossenen Diagnosen und Therapien resultieren. Dabei werden alle Diagnosen und Therapien mit einbezogen, die einem Patienten im Zuge seines Heilungsprozesses zugeordnet werden. Dies repräsentiert das inhaltsbasierte Paradigma.

Durch die kontinuierliche Generierung von Informationen, die aus Diagnosen und Therapien resultieren, und die darauf beruhende Entscheidungsfindung der Ärzte, wird das inhaltsbasierte Paradigma zum zentralen Paradigma im medizinischen Bereich. Trotzdem ist es nicht vorbehaltlos einsetzbar. Für ein Datenbasiertes Prozessunterstützungssystem wird noch eine Prozesssteuerung benötigt, die mit den anfallenden Daten interagieren muss. Die Prozesssteuerung muss mit ad-hoc partizipierenden Informationssystemen arbeiten können, die unterschiedliche Hardware verwenden. Die Aufnahme neuer Daten bringt eine Schemaänderung während der Laufzeit mit sich. Trotzdem erfüllt das inhaltsbasierte Paradigma am besten die Anforderungen des medizinischen Umfeldes. Des Weiteren arbeiten mehrere Institutionen an der Behandlung eines Patienten und tauschen Informationen per Informationssystemen aus. In diesem Fall müssen Teilschritte durch einen Sequenzfluss von Anwendungsfunktionen geplant werden. Somit sind sowohl das aktivitätsbasierte als auch das Groupware Paradigma neben dem inhaltsbasierten Paradigma zu berücksichtigen. Die drei Paradigmen spannen ein Dreieck auf, das die jeweilige Beziehung zueinander darstellt.

3.7 Spannungsdreieck der drei Paradigmen

Ein System zur Prozessunterstützung (SzP) realisiert nicht nur ein Paradigma, da für einen Prozessablauf dieser neben Daten auch Aktivitäten benötigt werden, um ein definiertes Prozessziel zu erreichen. SzP werden heute nicht nur von einer Person verwendet, normalerweise partizipieren mehrere Anwender an einem SzP [Jab95]. Diese müssen die Ergebnisse ihrer jeweiligen Aktivitäten kommunizieren und auch koordinieren können. Es besteht aber die Möglichkeit die SzP einem Paradigma zuzuordnen, dass diese am stärksten beeinflusst.

Die Abbildung 3.6 beschreibt ein Spannungsdreieck zwischen den drei oben beschriebenen Paradigmen. An den Seitenlinien, die das Dreieck aufspannen, liegt jeweils eine reine Form des jeweiligen Paradigmas vor, ohne dabei mit den jeweils anderen Paradigmen vermischt zu werden. Beim inhaltsbasierten Paradigma stehen die Daten, die die Informationen enthalten, im Mittelpunkt. Bei dem aktivitätsbasierten Paradigma stellen die Aktivitäten den Mittelpunkt dar. Bei Groupware liegt der Schwerpunkt auf der

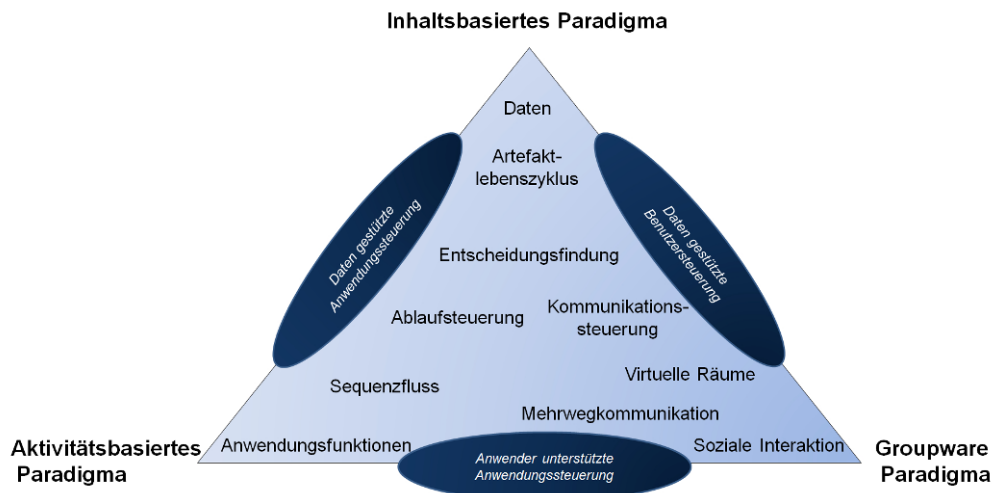


Bild 3.6: Skizze des Zusammenspiels von unterschiedlichen Paradigmen

sozialen Interaktion zwischen den einzelnen Anwendern eines SzP. Je weiter man sich von den Seiten eines einzelnen Paradigmas in Richtung des Dreieckszentrums entfernt, umso mehr vermischen sich die Eigenschaften der Paradigmen. Das inhaltsbasierte Paradigma beschreibt an dieser Position den Lebenszyklus der Daten (Beschreibung in Abschnitt A). Beim aktivitätsbasierten Paradigma stellt der Sequenzfluss von Aktivitäten den Verlauf eines Prozesses dar. Das Groupware Paradigma wird aufgeteilt, zum einen in Mehrweg-Kommunikation und zum anderen in virtuelle Räume¹. Im Zentrum des Dreiecks interagieren die drei Paradigmen am stärksten mit den jeweils anderen. Auf der Seite des inhaltsbasierten Paradigmas steht die Eigenschaft: Entscheidungsverfahren. Dabei werden in einem SzP mittels der Daten Entscheidungen getroffen. Für das aktivitätsbasierte Paradigma wird die Ablaufplanung in den Mittelpunkt des Dreiecks gestellt. Dabei wird eine Aufgabenstellung definiert und die Reihenfolge von Aktivitäten vorgegeben, die notwendig sind, um das entsprechende Ziel zu erreichen. Beim Groupware Paradigma steht die Kommunikationssteuerung im Mittelpunkt. Bei jedem SzP ist Kommunikation notwendig, wie zum Beispiel die 1:1 Kommunikation zwischen Server und Client.

In dieser Arbeit werden Workflow-Ansätze betrachtet, die als Haupteigenschaft das inhaltsbasierte Paradigma anwenden.

Jeder inhaltsbasierte Prozessunterstützungsansatz besitzt sowohl eine gewisse Ausprägung des aktivitätsbasierten als auch des Groupware Paradigmas. Durch einordnen

¹Sozialer Treffpunkt in digitalen Medien

der Prozessunterstützungsansätze in das oben beschriebene Dreieck kann eine grobe Klassifikation stattfinden. Die Vergleichskriterien, die in Kapitel 7 beschrieben werden, beziehen sich auf Teilaspekte der unterschiedlichen Paradigmen. Eine eindeutige Einordnung kann nicht vorgenommen werden, weil die Vergleichskriterien mindestens in zwei, oft aber in allen drei Paradigmen wiederzufinden sind. Aus diesem Grund werden die Prozessunterstützungsansätze mittels der Vergleichsparameter in das Dreieck implementiert. Durch das vorgegebene Einsatzgebiet, medizinisches Umfeld, wird jeder Ansatz zur Prozessunterstützung mit sensiblen personenbezogenen Daten umgehen müssen. Daher werden sich die Ansätze im oder um das gelbe Dreieck einordnen (Abbildung 3.6).

4 α -Flow Kurzdarstellung und Einsatzumgebung

Im vorangegangenen Abschnitt 3 wurden die unterschiedlichen Paradigmen für den Einsatz eines Systems zur Prozessunterstützung beschrieben. Am Informatik 6 Lehrstuhl (Datenmanagement)¹ wird ein Ansatz zur Prozessunterstützung namens α -Flow entwickelt, der primär im medizinischen Umfeld eingesetzt wird und im Kapitel 5.1 als Referenz für den Vergleich mit verwandten Ansätzen zur Prozessunterstützung dient. In den nachfolgenden Abschnitten werden prägnant die Eigenschaften von α -Flow zusammengefasst. Im Abschnitt 4.1 wird beschrieben, welche Ziele α -Flow erreichen will. α -Flow orientiert sich an den Paradigmen, die in Kapitel 3 vorgestellt wurden. Deswegen wird im Abschnitt 4.2 beschrieben, wie die Paradigmen umgesetzt werden und wie sie im Kontext zum medizinischen Umfeld zu bewerten sind. Danach wird in Abschnitt 4.3 diskutiert, welche Bedingungen an die Paradigmen geknüpft sind.

4.1 Kurzdarstellung des Hintergrunds von α -Flow

Die elektronisch basierte Patientenverwaltung wird schon seit Jahren im medizinischen Umfeld diskutiert. Electronic Health Record (ERH) [PB05] ist ein Versuch mittels OpenERH eine Patientenverwaltung zu realisieren. OpenERH verfehlt aber dieses Ziel in Bezug auf lose gekoppelte Systeme im Langzeitszenario, wie beispielsweise die Integration von niedergelassenen Ärzten ohne zentrales Back-End.

Der medizinische Prozessablauf basiert heute auf verteilten Papier-Dokumenten (beispielsweise: Überweisungsformulare, Rezepte, Laboranalysen, Entlassungspapiere, ...) die von Abteilung zu Abteilung weitergereicht werden. Hier setzt α -Flow an. Die einzelnen Papier-Dokumente werden in digitale Dokumente umgesetzt, die dann wie die Papiervarianten als eigenständige Einheiten an die einzelnen Stationen weitergeleitet werden.

¹<http://www6.informatik.uni-erlangen.de/DE/>

Dafür wird ein Meta-Datenmodell erstellt, welches die notwendigen Informationen für die Durchführung eines Prozesses beinhaltet [NL09] [NL10] [NL11a].

Das Ziel von α -Flow ist es den Prozessablauf in einer verteilten und heterogenen Umgebung zu unterstützen, auch wenn die einzelnen Einheiten softwaretechnisch nur lose gekoppelt sind. Dabei wird der Prozess in einzelne Verantwortungsgebiete aufgeteilt, mit dem Ziel, dynamische Teams ad-hoc in den Prozessablauf einzubinden.

4.2 Realisierung der Paradigmen

Bei jeder Station eines diagnostischen und therapeutischen Zyklus fallen Ergebnisse einer Diagnose und/oder einer Therapie an. Diese Ergebnisse werden im Datendokument (α -Doc) gespeichert. Zusätzlich werden in den Datendokumenten die Verwaltungsinformationen hinzugefügt, wie beispielsweise: wer war der Autor der Daten, wer hat die Daten verändert und wer hat die Daten für andere freigegeben. Die Entscheidungen des Arztes über den weiteren Behandlungsverlauf, die er basierend auf den in dem Datendokument gespeicherten Informationen getroffen hat, werden ebenfalls in den Verwaltungsinformationen hinterlegt. Dabei handelt es sich beispielsweise um Informationen zu einer Therapie, die - idealerweise - zur Genesung des Patienten führt. Diese Vorgehensweise beschreibt das inhaltsbasierte Paradigma.

Eine Entscheidung des Arztes, wie beispielsweise „Arm röntgen“, wird in den Inhaltsinformationen abgespeichert. Allerdings wird zugleich in den Verwaltungsinformationen die Überweisung hinterlegt. Die Schritte von der Entscheidung einen Arm zu röntgen hin zur Erstellung der Verwaltungsinformation, eine Überweisung einzutragen, stellen einen Ablaufplan dar. Der Ablaufplan spiegelt das aktivitätsbasierte Paradigma wieder.

α -Flow kommt ohne eine Installation auf ein System aus, weil das Datendokument selbst ausführbar ist und mit dem Anwender interagiert. Ausführbare und mit dem Anwender interagierende Datendokumente werden als aktive Datendokumente bezeichnet.

In den aktiven Datendokumenten werden die Daten nach ‚Sichtbarkeit‘ (privat oder öffentlich) und ‚Gültigkeit‘ (gültig oder nicht gültig) getrennt. Abhängig von der ‚Sichtbarkeit‘ der Daten in einem Datendokument ist es möglich, gemeinsam auf die Daten zuzugreifen. Alternativ beschreibt die ‚Gültigkeit‘ den Zustand der Daten und definiert somit, ob sie in einem stabilen Zustand sind oder nicht. Anhand dieser Informationen können entweder alle Anwender auf die gleichen Daten zugreifen oder nur einzelne auf die ihnen zugeordneten und diese entsprechend ihrer jeweiligen Anforderungen verändern. Bei der Interaktion der Anwender mit den Daten zu einem bestimmten Fall sind die

Anwender nicht an einen Ort gebunden. Die Trennung der Sichtbarkeit und der Gültigkeit, sowie die verteilte gemeinsame Arbeit an einen Fall, beschreibt das Groupware Paradigma.

4.3 Paradigmabewertung

In diesem Kapitel wurde am Beispiel von α -Flow dargestellt, wie die unterschiedlichen Paradigmen in einem Ansatz der Prozessunterstützung partizipieren. α -Flow erfüllt zu einem gewissen Grad alle Paradigmen, die in Kapitel 3 vorgestellt wurden.

Das inhaltsbasierte Paradigma bildet das Grundgerüst von α -Flow und stellt somit das zentrale Paradigma dieses Ansatzes dar. Basierend auf den gespeicherten Informationen in einem Datendokument steuern diese die Abstimmung der an einer Behandlung beteiligten Institutionen. Die Anwender können auf der Basis der vorliegenden Informationen den Ablauf eines Prozesses effektiv mitgestalten.

Selbst ein Ansatz zur Prozessunterstützung, der basierend auf den Eigenschaften des inhaltsbasierten Paradigma entwickelt wurde, muss dahingehend weiter analysiert werden, wie viele Eigenschaften er von dem aktivitätsbasierten oder Groupware Paradigma enthält. α -Flow unterstützt die kooperative Arbeit mittels einer Versionierung, des Sichtbarkeits- und des Gültigkeitskonzeptes. Dieses ermöglicht das gleichzeitige, räumlich getrennte Arbeiten mit den gleichen Daten ohne das ein Informationsverlust zu befürchten ist.

Es existiert ein Ablaufplan, der die Reihenfolge der einzelnen Bearbeitungsschritte enthält. Diesem kann entnommen werden, ab wann Daten für einen Nutzer in einer Instanz sichtbar werden. Der Ablaufplan spiegelt das aktivitätsbasierte Paradigma wieder.

Dieses Beispiel veranschaulicht, dass ein Ansatz zur Prozessunterstützung, der nur ein Paradigma beinhaltet, nicht existiert. Infolgedessen muss die Einordnung der Ansätze entsprechend der Ausprägungen ihrer Eigenschaften in die unterschiedlichen Paradigmen erfolgen.

5 Kurzdarstellung der betrachteten Ansätze zur Prozessunterstützung

Im Kapitel 3 wurden die Unterschiede zwischen den einzelnen Paradigmen dargelegt. Die Ansätze der Prozessunterstützung sind auf der Basis eines spezifischen Paradigmas entwickelt worden. In dieser Arbeit werden unterschiedliche inhaltsbasierte (engl. *content oriented*) Ansätze zur Prozessunterstützung miteinander verglichen. Wie bereits im Kapitel 1 beschrieben wurde, bildet α -Flow die Basis des Vergleiches. Ausgehend von α -Flow wurden per Literaturrecherche fünf aktuelle Ansätze, welche die gleichen oder verwandte Eigenschaften besitzen, ausgewählt. Alle ausgewählten Ansätze realisieren zu einem gewissen Grad das inhaltsbasierte Paradigma. Dies gilt insbesondere für die Ansätze Artifact-Centric Business Process Models (ACBPM), Datenorientiertes Prozess-Management-System (DPMS) und Data-driven Process Structures (DDPS). Diese drei Ansätze repräsentieren aktuelle datengesteuerte Ansätze zur Prozessunterstützung. In Abbildung 1.3 in Kapitel 1 wurden die unterschiedlichen Modelle zur Prozessunterstützung beschrieben. Aus dem Bereich Case Handling (CH) wurde der Ansatz Adaptive Case Management (ACM) und aus dem Bereich Produktions-Workflows der Ansatz Business Process Modelling and Notation (BPMN) ausgewählt. ACM stellt einen inhaltsbasierten Ansatz dar, der auf dem Case Management (CM) basiert. ACM wurde ausgewählt, da er wie CM im medizinischen Bereich bereits zum Einsatz kommt [Löc03] und durch den adaptiven Ansatz inter-institutionell eingesetzt werden kann. BPMN wurde ausgewählt, weil er einen klassischen Ansatz zur Prozessunterstützung repräsentiert. An seinem Beispiel wird betrachtet, wie sich das inhaltsbasierte Paradigma auf diesen Typ von Prozessunterstützungsansatz auswirkt. Zusätzlich beschreibt jeder der Ansätze einen Lebenszyklus für die Daten des jeweiligen Ansatzes. In diesem Kapitel sind in den einzelnen Abschnitten die Ansätze kurz zusammengefasst.

Groupware stellt neben seinen Eigenschaften als Paradigma auch einen eigenen Ansatz dar. Bei den Ansätzen wird Groupware allerdings nicht betrachtet. Die Groupware-Ansätze der einzelnen Anbieter unterscheiden sich zu stark in den einzelnen Systemkomponenten und deren Funktionalitäten, und ermöglichen daher keine allgemeingültige

Bewertung von Groupware. Erschwerend kommt hinzu, dass die Hersteller nur bedingt detaillierte Informationen über ihre Entwicklungen preisgeben.

Die betrachteten Ansätze verwenden beispielsweise für die Datencontainer oder die Anwendungsfunktionen unterschiedliche Begriffe. Zur besseren Verständlichkeit werden im folgenden Datencontainer als *Datenobjekte* bezeichnet. Funktionen, die Aufgaben erfüllen, werden als *Anwendungsfunktionen* bezeichnet. *Anwendungsfunktion* setzen sich aus *Aktionen* zusammen. *Aktionen* sind atomar und können nicht mehr unterteilt werden. Genauere Spezifizierung der Begriffe befindet sich in Kapitel A.

5.1 α -Flow

Der α -Flow Ansatz stellt den Ausgangspunkt des Vergleiches dar. α -Flow wird von der ProMed-Forschungsgruppe des Lehrstuhls 6 für Informatik der Friedrich-Alexander-Universität Erlangen-Nürnberg entwickelt. Diese Gruppe hat bis dato folgende wissenschaftliche Artikel über α -Flow veröffentlicht:

- alpha-Flow: A Document-based Approach to Inter-Institutional Process Support in Healthcare. [NL09]
- The alpha-Flow Use-Case of Breast Cancer Treatment – Modeling Inter-Institutional Healthcare Workflows by Active Documents [NL10]
- The alpha-Flow Approach to Inter-Institutional Process Support Health [NL11b]

5.1.1 Kurzdarstellung von α -Flow

Im Zentrum des α -Flow Ansatzes steht ein aktives elektronisches Dokument (Datenobjekt) namens α -Doc. Ein α -Doc enthält neben den Domänen spezifischen Informationen, wie beispielsweise Befunde, Bilder, Anamnesen und Verordnungen, auch Verwaltungsinformation, die den technischen Ablauf steuern. Zu den Verwaltungsinformationen gehören die Informationen für das Regelsystem, wie die Benutzerverwaltung, das Zugriffssystem, die Protokolldateien, Anmerkungen und des weiteren Versions- und Koordinierungsinformationen. Diese Informationen werden von den gleichnamigen Dokumentfunktionalitäten verwendet, welche im α -Doc integriert sind. Dies entkoppelt ein α -Doc von Applikationen. Aus der Integration der Verwaltungssysteme in die Daten resultiert ein aktives Dokument. Das Ziel des aktiven Dokumentes ist es eine möglichst geringe Kopplung der verteilten, spezifischen Anwendersysteme, die am Prozessablauf beteiligt sind, zu erreichen.

Ein α -Doc kann eine oder mehrere α -Cards beinhalten. Die α -Cards stellen die kleinste Einheit der Validierung dar und beinhalten medizinische Daten, wie Diagnosen, Wirksamkeit von Therapien, Rezepte, usw. Neben den Domänen spezifischen Informationen enthalten α -Cards auch Verwaltungsinformationen, wie beispielsweise Zugriffsrechte.

Eine Patientenbehandlung selbst wird als α -Episode bezeichnet und durch eine α -Doc beschrieben. Ein α -Doc besitzt kein definiertes Ende. Es besteht die Möglichkeit ein α -Doc zu erweitern, wenn neue Informationen für diese Prozessinstanz vorliegen. Dies bedeutet, wenn ein Patient für eine eindeutige Diagnose eine weitere Untersuchung benötigt, werden die Informationen der neuen Untersuchung im selben α -Doc gespeichert. Bei der Erstellung eines α -Doc wird dieses mit Koordinierungs- α -Cards initialisiert. Eine Modifikation des Behandlungsverlaufs erzwingt die Eingabe einer neuen α -Card in das α -Doc.

α -Flow unterstützt verteilte und lose gekoppelte Systeme, in denen alle Informationen, sowie die Anwendungsfunktionen, wie beispielsweise die Benutzeroberfläche, in dem Dokument α -Doc, integriert sind. Mit diesen Eigenschaften lässt sich α -Flow in verschiedene Systeme integrieren ohne dass eine Installation notwendig wird. Zusätzlich ist in α -Flow ein Versionierungssystem integriert, welches α -Flow in die Lage versetzt, parallel auf verschiedenen Informationssystemen und dort auf jeweils einer eigenen Version der Daten zu arbeiten.

Eine α -Card beinhaltet mehrere Eigenschaften. Die zwei wichtigsten Eigenschaften sind die Sichtbarkeit und die Gültigkeit. Die Sichtbarkeit unterteilt sich in öffentlich oder privat. Ist eine α -Card öffentlich, dann haben alle Anwender der Instanz die Möglichkeit den Inhalt der entsprechenden α -Card für ihren Prozess zu lesen. Für Arbeitsschritte, die nicht der kooperativen Arbeit angehören, gibt es die Möglichkeit eine α -Card als privat zu definieren. Alternative kann der Inhalt einer α -Card in einem gültigen Zustand sein oder nicht. Im traditionellen DBS sind diese Eigenschaften eng mit einander verbunden. Wenn Daten für alle sichtbar sind, sind sie auch gültig. Bei α -Flow können dagegen Daten sichtbar sein, die aber noch nicht gültig sind.

Des Weiteren kann ein α -Doc parallel verarbeitet werden. Die mögliche Parallelverarbeitung von α -Cards bedarf am Ende der Parallelverarbeitung eine Zusammenführung der unterschiedlichen Versionen. Für diese Aufgabe ist eine Variantenverwaltung implementiert worden. Die Variantenverwaltung stellt immer eine global gültige Variante einer α -Card bereit. Für den Zugriff auf Daten sind weitere Autorisierungs- und Authentifizierung-Systeme implementiert.

Ein α -Doc ist durch weitere α -Cards unbestimmt erweiterbar. Zusätzlich zu den Daten auf einer α -Card können auf ihr auch Verweise auf andere α -Cards gespeichert werden. Somit kann ein Verlauf mittels Verweisen erweitert werden. Aus jedem α -Doc kann zur Laufzeit immer eine neue Instanz erzeugt werden. Die Funktionalitäten werden dabei vererbt. Des Weiteren unterstützt α -Flow eine Public Key Infrastructure (PKI), die zum einen für die Sicherheit bei der Ausführung und zum anderen für die Sicherheit der Daten sorgen soll.

5.1.2 Beispiel für die Anwendung von α -Flow

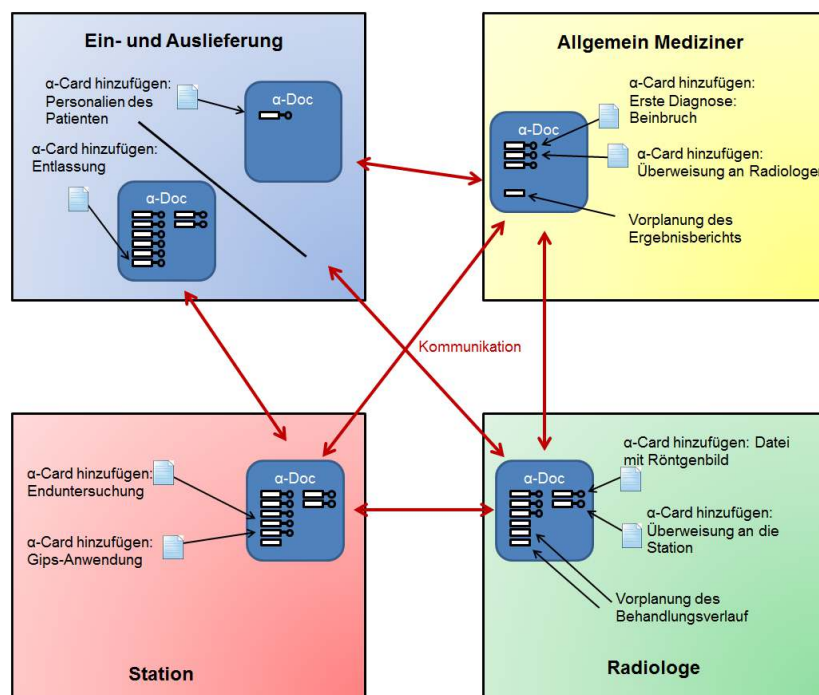


Bild 5.1: Skizze eines α -Flow-Anwendungsfalls

Die Abbildung 5.1 skizziert beispielhaft einen vereinfachten Anwendungsfall. Auf jeder Station liegt physisch ein α -Doc vor, auf das zugegriffen wird. Für die Verwendung eines α -Docs wird kein Softwareprodukt benötigt. Jeder Arzt oder sonstige an der Behandlung des Patienten beteiligten Institutionen fügen basierend auf den Ergebnissen der durchgeführten Diagnosen oder empfohlenen Therapien eine weitere α -Card hinzu. Während des Verlaufs der Behandlung durchläuft das α -Doc die gleichen Stationen wie der Patient selbst. Das α -Doc repräsentiert somit die Fallakte des Patienten. Die Abbildung verzichtet zwecks der besseren Übersichtlichkeit auf die Darstellung der Informationssysteme, auf denen ein α -Doc in einer Station verwendet wird.

5.2 Business Process Modelling and Notation

Die Kerneigenschaft des Business Process Modelling and Notation (BPMN) Ansatz zur Prozessunterstützung kann dem aktivitätsbasierten Paradigma zugeordnet werden. Hierdurch unterscheidet er sich von den anderen analysierten Ansätzen. BPMN ist standardisiert und wird zur allgemeinen Beschreibung von Prozessabläufen verwendet. Aus diesem Grund werden die Eigenschaften von BPMN in Bezug auf dessen möglichen Einsatz im Umfeld der Medizin bewertet.

Verantwortlich für die Koordinierung der Weiterentwicklung ist Object Management Group (OMG) [All08]. Über BPMN existieren viele wissenschaftliche Berichte und Bücher. Für die Beschreibung von BPMN wurden folgende Quellen betrachtet:

- BPMN Modeling and Reference Guide Understanding and Using BPMN [WM08]
- BPMN-Business Process Modeling Notation: Einführung in den Standard für die Geschäftsprozessmodellierung [All08]
- A bpmn extension for the modeling of security requirements in business processes [RFMP07]

5.2.1 Kurzdarstellung des Ansatz BPMN

Business Process Modelling and Notation (BPMN) repräsentiert eine graphische Spezifikationsprache für Geschäftsprozesse. Die Spezifikationsprache benutzt für die Modellierung Symbole, wie beispielsweise Anwendungsfunktionen, Gateways, Ereignisse und Start- und End-Symbole, die zur Erstellung eines Prozessverlaufes verwendet werden können. Weitere Elemente und deren Beschreibung befindet sich in Kapitel C. Das Ziel von BPMN ist eine Notation, die es den Benutzern ermöglicht, sowohl fachlich als auch technisch auf einem hohen Niveau Geschäftsprozesse zu modellieren. Ein BPMN-Modell kann selbst nicht ausgeführt werden. Hierfür wird eine Business Process Management-System (BPMS), die ein BPMN-Modell in einen ausführbaren Prozess umwandelt, benötigt. Die korrekte Umwandlung kann nur garantiert werden, wenn die Modellierer die Notationsrichtlinien strikt einhalten. Im Gegensatz zu Ereignisgesteuerten Prozessketten (EPK) [Leh08] ist BPMN standardisiert und wird von mehreren Herstellern verwendet.

BPMN ist für den Einsatz in unterschiedlichen Geschäftsbereichen entwickelt worden. Einen Prozess anhand seiner jeweiligen Aktivitäten zu beschreiben, steht bei BPMN im Vordergrund. Mittels Verbindungen und Gateways, die für Entscheidungen oder zur Parallelisierung verwendet werden, wird ein Sequenzfluss von Aktivitäten erzeugt. Ein

Dokumentenverlauf kann zusätzliche neben dem Sequenzfluss der Aktivitäten modelliert werden. Auf Basis der Informationen, die in den Datenobjekten gespeichert sind, können Gateways Entscheidungen treffen. Mittels der Gateways besteht die Möglichkeit auf Ereignisse zu reagieren oder einen Ablauf zu parallelisieren. Zusätzlich gibt es für die Aktivitäten noch die Möglichkeit beim Auftreten eines Fehlers oder eines unvorhergesehenen Ereignisses diese wieder zurückzusetzen. Dieser Mechanismus ist vergleichbar mit der Transaktionsverwaltung (TAV) von DBSen. Um solche kritischen Schritte zu markieren, können den Symbolen Anmerkungen hinzugefügt werden. Neben der Einteilung in kompensierbare Aktivitäten können diese in weitere unterschiedlichen Kategorien eingeteilt werden. Zu diesen Kategorien zählen beispielsweise mehrfachausgeführte, benutzergesteuerte, sequenzielle oder ad-hoc Anwendungsfunktionen.

BPMN unterstützt die inter-institutionelle Kommunikation. Dabei werden einzelne Institutionen als unterschiedliche Verlaufsbahnen dargestellt. Ein Prozessverlauf, der durch den Sequenzfluss dargestellt wird, umfasst dabei alle Verlaufsbahnen. Mittels des Versendens von Nachrichten können fremde Institutionen an einem Prozessablauf partizipieren.

Für BPMN existieren Erweiterungen, die ein Sicherheitskonzept beinhalten. Eine Standardisierung von PKI-Systemen in BPMN besteht nicht. Die Anwendbarkeit dieser Erweiterung im medizinischen Umfeld muss erst noch bewiesen werden. Das medizinische Umfeld arbeitet mit sensiblen persönlichen Daten. Aus diesem Grund wird eine Erweiterung betrachtet, die die Verschlüsselung von Datenobjekten beinhaltet

5.2.2 Beispiel für eine Anwendung von BPMN

Folgende Abbildung 5.2 skizziert eine Behandlung eines Beinbruchs. Zum Zwecke der besseren Übersichtlichkeit wurde der Prozess vereinfacht.

Die Abbildung 5.2 veranschaulicht am Beispiel eines Beinbruchs den Behandlungsverlauf mittels BPMN-Notation¹. Es wird der Sequenzfluss von Aktivitäten veranschaulicht, welche die diagnostischen und therapeutischen Zyklen der Behandlung beschreiben. Neben der Station ist auch die Radiologie an dem Prozess-Verlauf beteiligt. Die Radiologie wird mittels eines Ereignisses aufgerufen und entsprechend wird mittels eines Ereignisses der Sequenzfluss in der Radiologie beendet. Der Datenfluss, hier ein Formular, stellt einen separaten Sequenzfluss dar, der den Sequenzfluss der Aktivitäten nicht tangiert.

¹Eine genau Legende mit allen Symbolen und deren Bedeutung ist im Anhang C zu finden.

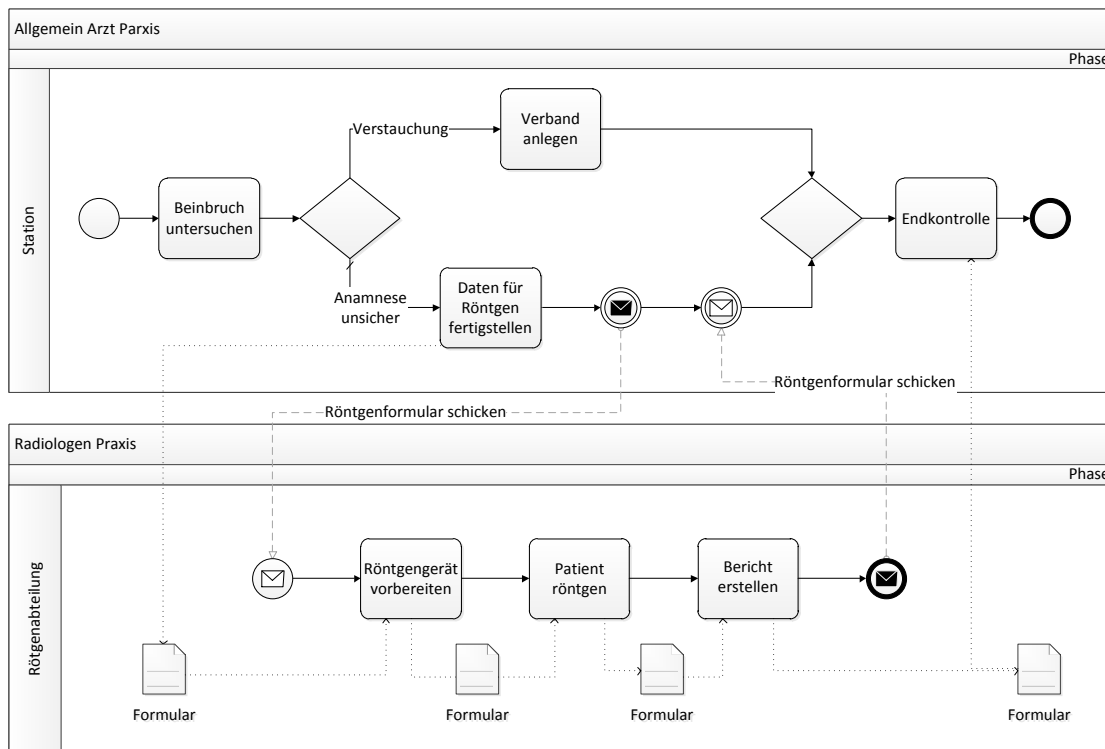


Bild 5.2: Schematische Darstellung einer Beinbruchbehandlung

5.3 Adaptive Case Management

Adaptive Case Management (ACM) stellt einen weiteren Ansatz zur Prozessunterstützung dar. ACM ist eine Entwicklung, die aus dem Case Management (CM) bzw. Case Handling (CH) hervorgeht und beschreibt einen Ansatz, der das Ergebnis eines Prozesses - das Produkt - in den Fokus seiner Ausführung stellt und nicht die Aktivitäten oder die Datenverwaltung. Das Produkt einer CH-Instanz zum Erreichen eines Geschäftsziels besitzt sowohl eine Struktur als auch Zustände, die mittels einer Ansammlung von Datenobjekten gespeichert sind. Zu den Datenobjekten existieren digitale Formulare, die verschiedene Betrachtungsmöglichkeiten der Daten in den Datenobjekten repräsentieren. Mit den digitalen Formularen können Aktivitäten verbunden sein [AWG05a]. Basierend auf dem Zustand des ausgeführten Prozesses werden die entsprechenden Aktivitäten ausgeführt. Die beteiligten Anwender können mittels Modifikation der Prozessdaten Veränderungen am Prozessverlauf vornehmen. An diesem Punkt greift ACM an. Bei diesem adaptiven CH-System werden die Benutzer in die Lage versetzt aktiver Modifi-

kation an dem Prozessverlauf vorzunehmen, als dies bei CH der Fall ist. Dadurch soll der Einsatz von IT-Fachkräften reduziert werden. Aufgrund der flexiblen Anpassung an komplexe Prozesse und des möglichen reduzierten Einsatzes von IT-Fachpersonal, die für die Erstellung eines Prozessablaufes benötigt werden, wurde dieser Ansatz zur Prozessunterstützung ausgewählt. Der Ansatz beschreibt viele wichtige Aspekte vom CM und erweitert sie. Allerdings fehlen für die Umsetzung von ACM die technischen Lösungsvorschläge. Als Grundlage für die Bewertung des Ansatzes dient das Buch, *Mastering the Unpredictable*, von Keith D.Swenson [Swe10].

5.3.1 Kurzdarstellung des Ansatzes ACM

Max j. Pucher beschreibt ACM in *Mastering the Unpredictable*[Swe10] als:

1. ACM is a productive system that deploys the organisation and process structure form defined architecture that through back-end interfaces becomes the system of record for the business data entities and content involved. All processes are completely transparent, as access authorization, and fully auditable.
2. ACM enables nontechnical business users in virtual organisations to seamlessly create/consolidate structured and unstructured processes from basic predefined business entities, Graphical User Interface (GUI) components, content, social interactions, and business rules.
3. ACM moves the knowledge-gathering process in the lifecycle from the template analysis phase to the process execution phase. The ACM system collects actionable knowledge – without an intermediate analyses phase - based on process patterns created by business users.

Die zentrale Eigenschaft dieses Ansatzes ist ein Master-Modell für das Datenschema. Das Master-Modell wird von einem Back-End, vermutlich einem DBS, zur Verfügung gestellt. Die Erstellung oder Veränderung der Daten oder Anwendungsfunktionen, die mit den Daten interagieren, werden mittels eines Regelsystems überwacht. Die Regeln können in natürlicher Sprache beschrieben werden. Der Systemzugriff wird durch ein Rollensystem mit Autorisierungs- und Authentifizierung geschützt. Alle Informationen werden zusätzlich verschlüsselt, um die Daten gegen den Zugriff von Dritten zu schützen.

Templates¹ für die Datenstruktur, die Regeln oder die Aktivitäten erleichtern das Anlegen von neuen Prozessinstanzen oder die Anpassung von alten. Alle Aktivitäten in einem Prozessverlauf sind von den verschiedenen Nutzern verfolgbar und erleichtern somit den Überblick über die Prozessschritte. Während des Verlaufs einer Instanz können Daten in den Prozessverlauf aufgenommen oder auch ausgegeben werden. ACM besitzt für die Ein- und Ausgabeverwaltung spezifische Anwendungsfunktionen, die diesen Prozess unterstützen. Die folgende Skizze 5.3 zeigt einen Ausschnitt der von ACM benutzten Systemkomponenten.

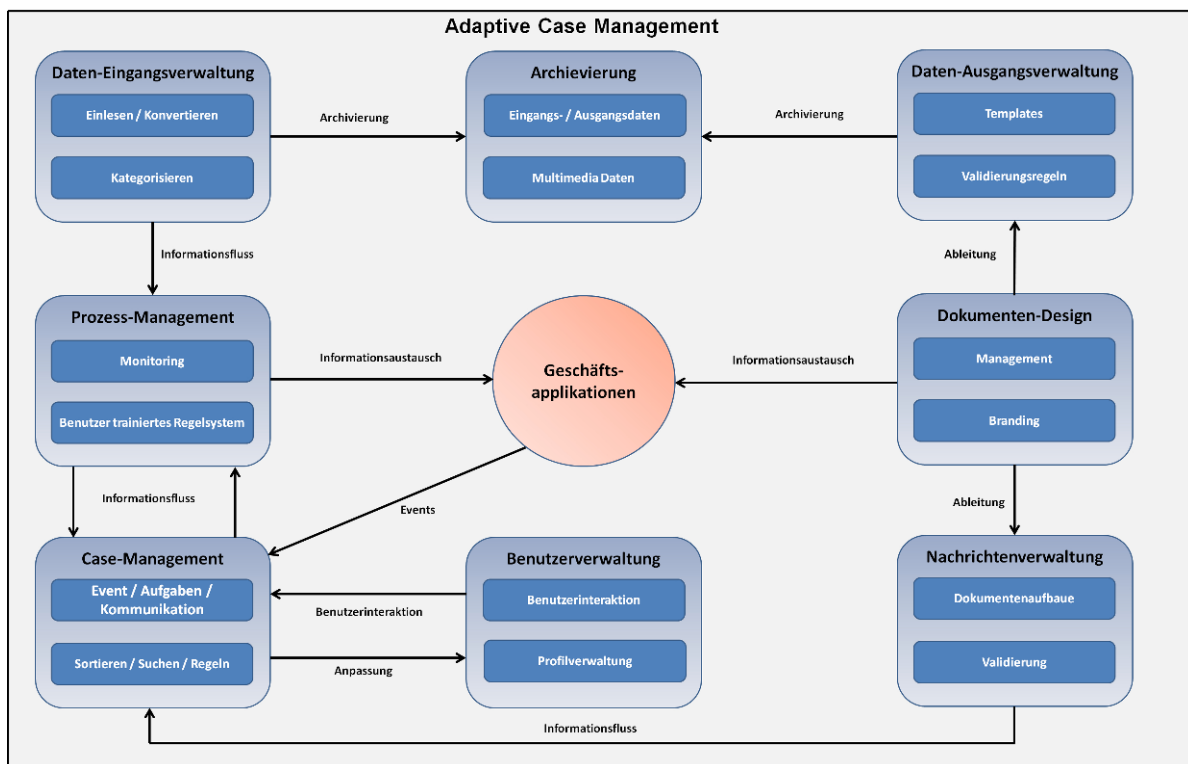


Bild 5.3: ACM-Systemkomponentenskizze

Templates von Prozesselementen sind die Kernkomponente dieses Ansatzes. Durch die Anpassung von alten, bereits verwendeten Prozesselementen und deren Transfer in einem neuen Verlauf ist es ACM möglich, sich in kurzer Zeit geänderten Rahmenbedingungen anzupassen. Jede neue Situation soll behandelbar sein und nicht im Voraus geplant

¹Prozesselementenvorlage, von der eine Instanz abgeleitet werden kann. Bei der Instanziierung eines Prozesses besteht die Möglichkeit alte Prozesselemente per Instanziierung an vorliegende Bedingungen anzupassen. Zu den Veränderungen gehören beispielsweise die Erweiterung des Datenschemata, der Anwendungsfunktionen oder der Regeln. Alle erstellten Prozesselemente werden zentral archiviert und können als Templates verwendet werden. [Swe10]

werden müssen. Die Planungsphase wird in die Ausführungsphase verschoben und bei einer entsprechenden Erfahrung der Anwender, können diese während der Abarbeitung die nächsten Schritte bestimmen. Zusätzlich wird die Möglichkeit geschaffen Daten, Benutzer oder Regeln während der Ausführung durch Prozessteilnehmer mittels natürlicher Sprache zu definieren und einzusetzen. Um die kooperative Arbeit mit ACM zu erleichtern, besteht die Möglichkeit Web 2.0 Anwendungen, wie beispielsweise Wikis, in den Prozess einzubinden.

ACM beinhaltet das Wort *adaptiv*. Adaptivität besitzt nach [Swe10] folgende Bedeutung:

Internal changes to an entity caused by outside conditions that become permanent and make the entity more fitting to those news conditions. It does imply that those changes are performed by means of the entity itself and not by some external force.

Die Bedeutung liegt darin, dass sich der Ansatz geänderten Anforderungen der Außenwelt anpasst, ohne dass eine komplexe Software für die neuen Anforderungen entwickelt werden muss. Damit wird die Anpassungszeit erheblich verringert, weil die Anpassung bzw. die Optimierung von den Mitarbeitern selbst durchgeführt werden kann ohne dass die Unterstützung von externen IT-Experten notwendig wird.

In diesem Ansatz wird beschrieben, welche Systemeigenschaften ACM erfüllen möchte. ACM selbst basiert im Wesentlichen auf CM, dessen Eigenschaften in [AWG05a] beschrieben werden. Neue Funktionalitäten wie die Integration von Web 2.0 – Web-Services werden beschrieben, aber wie sie in das restliche System integriert werden wird offengelassen. Es fehlt für die Erweiterungen ein technischer Lösungsansatz.

5.3.2 Beispiel für die Anwendung von ACM

Abbildung 5.4 skizziert den bekannten Anwendungsfall eines Beinbruchs. In der Mitte steht der zentrale Server, auf dem das ACM-System läuft. Jede Abteilung hat Zugriff auf ein System, auf dem ein ACM-Client installiert ist. Auf der Station ist ein Tablet-PC im Einsatz, da ACM mit unterschiedlichen Informationssystemen interagieren sollte, wenn die Clientsoftware dafür vorhanden ist. Die Abbildung verzichtet zwecks einer besseren Übersichtlichkeit auf die Darstellung der verwendeten Informationssysteme.

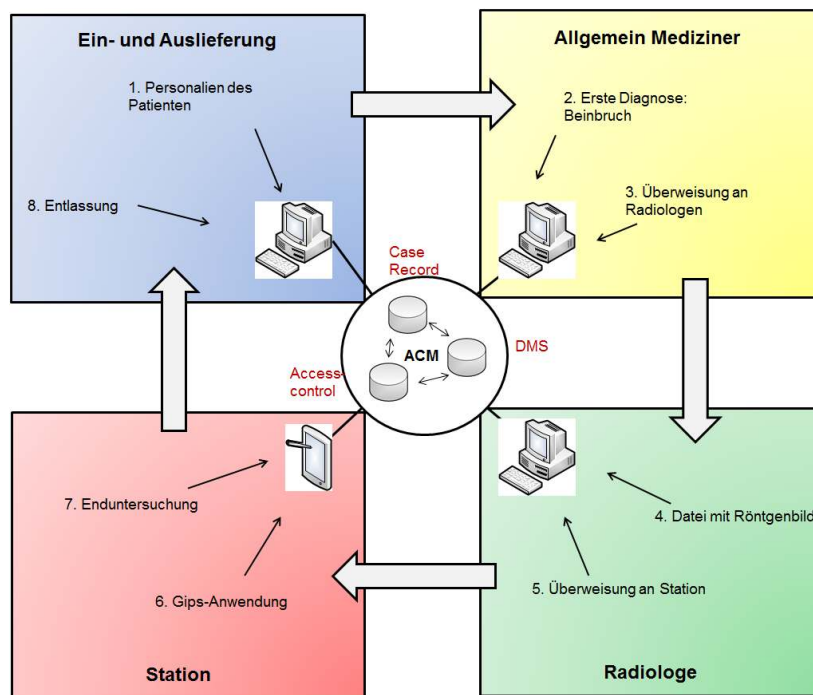


Bild 5.4: Skizze eines ACM-Anwendungsfalls

5.4 Artifact-Centric Business Process Models

Artifact-Centric Business Process Models (ACBPM) beschreibt einen Ansatz zur Prozessunterstützung der vom IBM T.J. Watson Research Center, Yorktown Heights, NY, USA mit der Unterstützung von mehreren Universitäten entwickelt wird. Für den ACBPM Ansatz existieren mehrere wissenschaftliche Veröffentlichungen. Zur Beschreibung von ACBPM wurden die folgenden Veröffentlichungen verwendet:

- Towards Formal Analysis of Artifact-Centric Business Process Models [BGH⁺07]
- Artifact-Centric Business Process Models: Brief Survey of Research Results and Challenges [Hul08]
- Automatic Construction of Simple Artifact-based Business Processes [FHS09]
- Artifact-Centric Workflow Dominance [CGHS09]

In diesem Ansatz werden vier Dimensionen beschrieben, die als Basis für einen Artefakt-zentrischen Ansatz¹ dienen.

5.4.1 4-Dimensionen

Der Artefakt-zentrische Ansatz beschreibt vier Komponenten: *Datenobjekte*, *Anwendungsfunktionen*, *Lebenszyklen* und *Assoziationen*. Diese vier Komponenten stellen die 4-Dimension dar. Abbild 5.5 skizziert das Modell.

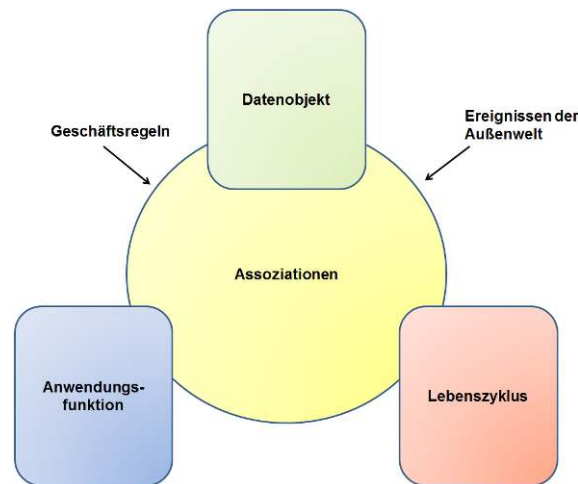


Bild 5.5: ACBPM-Systemkomponentenskizze

Ein *Datenobjekt* speichert die domänenspezifischen Nutzdaten in den Datenobjektattributen. Mittels der Datenobjektattribute wird der Lebenszyklus integriert, der den Prozess-Verlauf mitbestimmt. Der Lebenszyklus wird durch eine Konfiguration der Attribute im Datenobjekte bestimmt, auf welche die Anwendungsfunktionen zugreifen. Des Weiteren können Datenobjekte Attribute besitzen, die Verweise auf andere Datenobjekte beinhalten, welche die Beziehung der Informationen der beteiligten Datenobjekte untereinander beschreiben. Ein Datenobjekt stellt die Eigenschaften (Attribute) einer realen oder konzeptionellen Entität dar. Für das Erreichen eines Prozesszieles können mehrere Datenobjekte an einem Prozess beteiligt sein. Der Inhalt der Attribute oder die Attributanzahl eines Datenobjektes muss vor dem Einsatz nicht vollständig beschrieben sein. Basisinformationen, wie beispielsweise die Identifizierungsnummer (ID), müssen allerdings vorher definiert werden.

¹Artefakt-zentrischer Ansatz ist die Bezeichnung der Autoren, Richard Hull et. al. [Hul08] [BGH⁺07] für den Aufbau und die Umsetzung dieses Ansatzes zur Prozessunterstützung, der dem inhaltsbasierten Paradigma folgt.

Der *Lebenszyklus* oder auch Makro-Lebenszyklus beschreibt die Zustände, die von einem Datenobjekt im Prozessverlauf angenommen werden können. Mittels der Attributwerte zu einem bestimmten Zeitpunkt im Prozessverlauf werden die Knoten im Lebenszyklusgraph definiert. Ein Knoten stellt eine Konfiguration dar. Der Übergang von einer Konfiguration zur nächsten wird durch *Anwendungsfunktionen* vollzogen. Ein *Lebenszyklus* kann konzeptionell als endlicher Automat oder alternativ als Petri-Netz dargestellt werden.

Die *Anwendungsfunktionen* werden erstellt, um entweder ein Prozessziel zu erreichen, oder alternative ein Zwischenziel in Richtung eines Prozessziels zu beschreiben. Die *Anwendungsfunktionen* lesen und schreiben auf den Attributen des *Datenobjektes*. Aus einem *Datenobjekt* erhalten die *Anwendungsfunktionen* die Informationen, in welcher Reihenfolge sie auf den *Datenobjekten* arbeiten und welche Operationen sie ausführen sollen. Die *Anwendungsfunktionen* stellen die Übergangsfunktionen in einem endlichen Automaten dar.

Die *Assoziationen* stellen die Verbindung zwischen *Datenobjekt*, *Anwendungsfunktionen* und Ereignissen, wie Systemfehler, Erhalt einer Nachricht oder Warnungen vor dem Verletzen von Regeln, her. *Assoziationen* können als Abbildungsregelsystem angesehen werden. Die Regeln werden hierbei als Übergangsbedingungen im endlichen Automaten des Lebenszyklus dargestellt. Dabei beschreibt das Regelsystem, welche *Anwendungsfunktion* auf welchem *Datenobjekt* arbeiten darf. Zusätzlich werden die Regeln eingebunden, wie Ereignisse oder Daten in den Prozessverlauf aufgenommen werden dürfen.

Über die Ausprägung der unterschiedlichen Dimensionen ist es möglich, unterschiedliche Ansätze der Prozessunterstützung zu modellieren. Dabei kann ein Sequenzfluss von Anwendungsfunktionen modelliert werden oder ein Ansatz, der die Steuerung des Prozesses mit Hilfe der Daten durchführt.

5.4.2 Herausforderungen vom ACBPM-Ansatz

Der ACBPM Ansatz soll sich an *flexible* und an die *individuellen Änderungen der Rahmenbedingungen* anpassen können, um ein vorgegebenes Geschäftsziel zu erreichen. Die hierzu notwendige *Evolution des Prozessschemas* erfordert eine Anpassung der vier Dimensionen, was während der Laufzeit geschehen kann. Neben der Änderung der Rahmenbedingungen, die eine Veränderung des Prozessschemata notwendig machen, können mittels *Monitoring* Leistungsmerkmale in einem Prozessschemata erkannt und notfalls optimiert werden. Um die Verarbeitung von verschiedenen Prozessinstanzen zu erleichtern, wird ein *generischer/spezialisierter Ansatz* gewählt. Es werden Prozessschemata

erstellt, die für die Instanziierung entsprechend den Geschäftszielen, angepasst werden können. Zusätzlich werden die Elemente (Datenobjekte, Lebenszyklen, Anwendungsfunktionen, Assoziationen) als *Komponenten* entwickelt, um in einer neuen Prozessinstanz wiederverwenden zu können. Für die Benutzer werden unterschiedliche *Sichten* in den Prozess erstellt, mit dem Ziel die Übersichtlichkeit zu gewährleisten.

5.4.3 Beispiel für die Anwendung von ACBPM

Für die Betrachtung dieses Beispiels wird eine ACBPM Ausprägung betrachtet, die den Prozessverlauf mit den gespeicherten Daten steuert. In diesem Ansatz wird keine Aussage getätigt, ob die Datenobjekte auf einem zentralen Server oder nur auf lose gekoppelten Clients arbeiten. Im betrachteten Beispiel wird davon ausgegangen, dass die Datenobjekte von Client zu Client versendet werden.

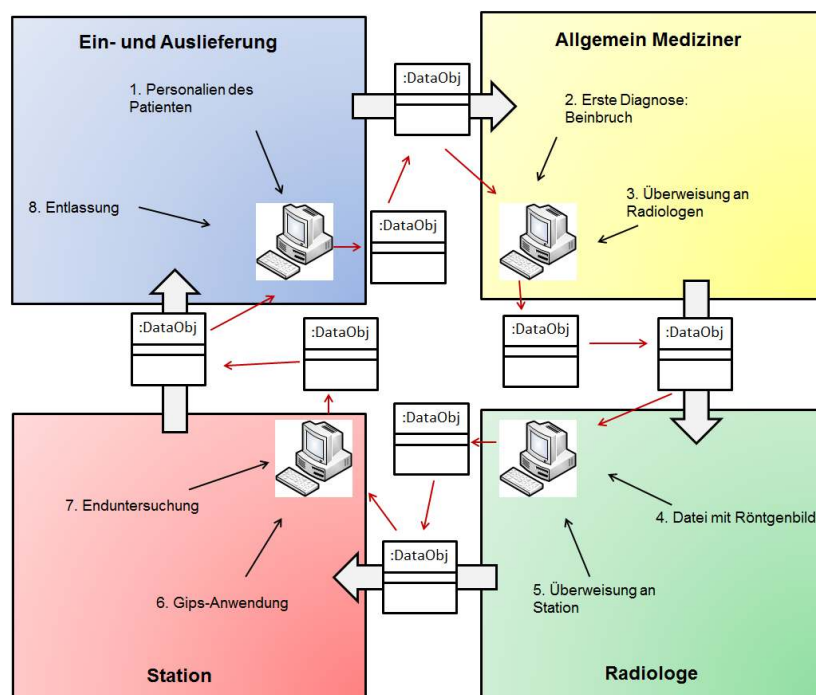


Bild 5.6: Skizze eines ACBPM-Anwendungsfalls

Die Abbildung 5.6 skizziert das bereits bekannte Beinbruchszenario. Dabei besitzt jetzt jede Abteilung einen Computer, auf dem die ACBPM-Software läuft. Das medizinische Personal fügt die Informationen per ACBPM-Software ein. Die ACBPM-Software persistiert die Daten in ein Datenobjekt und schickt dieses an die neue Abteilung. Die

neue Abteilung nimmt das Datenobjekt auf und liest die Daten aus. Danach können weitere Informationen hinzugefügt werden. Den Fluss der Datenobjekte visualisieren die Pfeile an den Datenobjekten in der Abbildung 5.6.

5.5 Datenorientiertes Prozess-Management-System

Der Ansatz Datenorientiertes Prozess-Management-System (DPMS) wird vom Institut für Datenbanken und Informationssysteme der Universität Ulm entwickelt. Über den Ansatz wurden viele wissenschaftliche Berichte veröffentlicht. Für die Betrachtung des Datenorientiertes Prozess-Management-System (DPMS)-Ansatzes wurden folgende Quellen herangezogen:

- Towards Object-aware Process Management Systems: Issues, Challenges, Benefits [KR09c]
- Herausforderungen auf dem Weg zu datenorientierten Prozess-Management-Systemen [KR09a]
- Integrating Users in Object-aware Process Management Systems: Issues and Challenges [KR10b]
- Herausforderungen bei der Integration von Benutzern in Datenorientierten Prozess-Management-Systemen. [KR10a]

DPMS beschreibt einen Ansatz zur Prozessunterstützung, bei dem die Daten zur Steuerung des Prozessverlaufs verwendet werden. Die Hauptelemente dieses Ansatzes sind die Daten, Aktivitäten, Prozesse und Benutzerinteraktion.

5.5.1 Kurzdarstellung des DPMS Ansatzes

Datenorientiertes Prozess-Management-System (DPMS) stellt einen datengesteuerten Ansatz zur Prozessunterstützung dar, mit dessen Hilfe versucht wird, die Nachteile von Aktivitätsfluss gesteuerten Prozessen zu lösen. Das Datenobjekt wird dabei als zentrales Prozesselement des Ansatzes angesehen, welches die ablaufenden Aktivitäten steuert. Zu einer Prozessinstanz können mehrere Datenobjekttypen gehören, die Informationen über eine Entität der Außenwelt beinhalten und diese Informationen in Attributen abspeichern. Ein Datenobjekt lässt sich durch seine ID eindeutig im Prozessverlauf identifizieren. Neben den Informationen über eine Entität der Außenwelt muss ein Datenobjekt die Daten,

welche zur Steuerung der Aktivitäten benötigt werden, speichern. In DPMS sind die Aktivitäten den Datenzuständen zugeordnet.

In den Attributen des Datenobjektes können auch Verweise auf weitere Datenobjekte beschrieben werden. Die Verweise strukturieren die Informationen einer Prozessinstanz. Von einem übergeordneten Datenobjekt gehen Verweise auf untergeordnete Datenobjekte aus. Die Struktur arbeitet nach dem Spezialisierungs- bzw. Generalisierungskonzept. Neben dem Spezialisierungs- bzw. Generalisierungskonzept können übergeordnete Datenobjekte noch aggregierte Ergebnisse der untergeordneten Datenobjekte beinhalten.

Während der Laufzeit ist es möglich dynamische Attribute oder Datenobjektinstanzen hinzuzufügen. Dies erlaubt die Anpassung des Prozesses auf der Basis von Prozessergebnissen und /oder geänderten Rahmenbedingungen. Anwender besitzen eine zentrale Sicht auf die Informationen in einem Prozess. Dabei werden die Grenzen eines Datenobjektes mittels deren Referenzen aufgelöst und die Informationen der Datenobjekte sind für die Benutzer transparent.

Mittels dieser Verweise können Objekthierarchien aufgebaut werden, welche die Informationen einer Prozessinstanz widerspiegeln. In Abbildung 5.7 werden die Informationen von drei Datenobjekte über die Behandlung eines Patienten dargestellt. Dabei stehen alle Objekte in einer 1:N Beziehung zueinander. Die Abbildung 5.7 skizziert die Unterschiede zwischen Datenobjekttyp und Datenobjektinstanz anhand eines fiktiven medizinischen Datenobjektes.[KR09b]

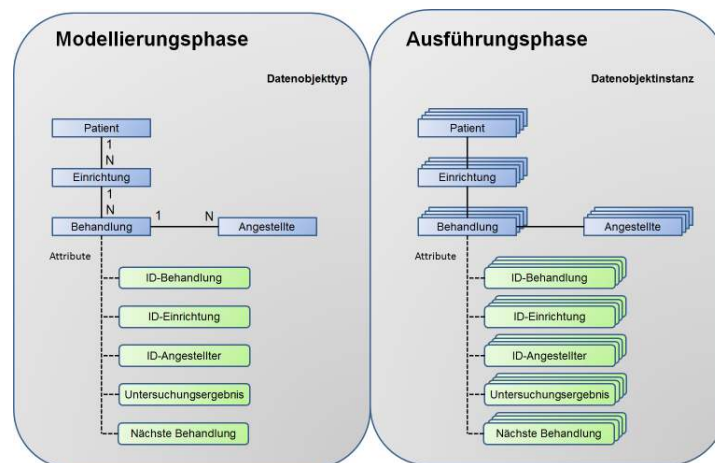


Bild 5.7: DPMS-Datenobjektskizze

In der Modellierungsphase werden zuerst die Datenobjekte entwickelt und danach die Aktivitäten, die auf den Datenobjekten arbeiten sollen. Innerhalb einer Prozessinstanz für ein spezifisches Datenobjekttyp beziehen sich Aktivitäten jeweils auf ein oder mehrere

Attribute eines Datenobjektes. Aktivitäten können eine sowohl horizontal ¹ als auch vertikal ² angeordnete Aktivitätshierarchie besitzen. Die untergeordnete Aktivitätsanzahl ist abhängig von der untergeordneten Objektanzahl. Die Abhängigkeit der Aktivitäten von dem Datenobjekt wird als Synchronität bezeichnet. Dabei kann die Bearbeitung von untergeordneten Aktivitäten asynchron zu den übergeordneten Aktivitäten durchgeführt werden.

Bei der Modellierung müssen die Datenobjekte im Vordergrund stehen und nicht die Aktivitäten. Veränderungen werden durch die Modifikation von Attributwerten oder des Datenobjekte realisiert. Nachgelagert müssen sich dann die Aktivitäten an die veränderten Datenobjekte anpassen. Abbildung 5.8 skizziert auf Typ-Ebene den Zugriff von Aktivitäten auf ein Datenobjekt anhand eines fiktiven Behandlungsprozesses.

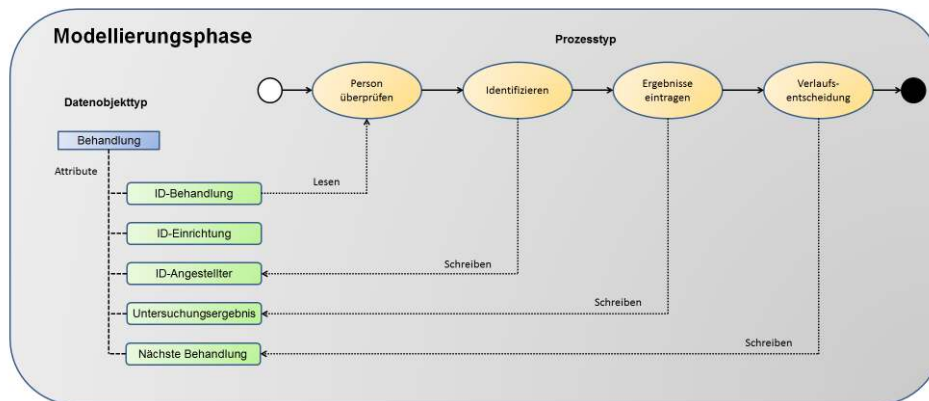


Bild 5.8: DPMS-Prozessablaufsskizze

5.5.2 Beispiel für die Anwendung von DPMS

In diesem Ansatz wird keine Aussage getätigt, ob die Datenobjekte auf einem zentralen Server oder nur auf lose gekoppelte Clients arbeiten. Allerdings werden im Zugriffsrechtmanagement Verweise von Benutzerkonten auf Prozesselemente beschrieben. Die Verweise lassen vermuten, dass es sich dabei um ein DBS handelt. Daher wird bei dem folgenden Beispiel von der Verwendung eines zentralen Servers ausgegangen.

Die Abbildung 5.9 skizziert ein beispielhaftes Beinbruchszenario. Bei diesem Beispiel werden die Computersysteme zum Zwecke der besseren Übersichtlichkeit nicht abgebildet.

¹Bearbeitung von mehreren verschiedenen Datenobjekttypen

²Bearbeitung von mehreren gleichen Datenobjekttypen

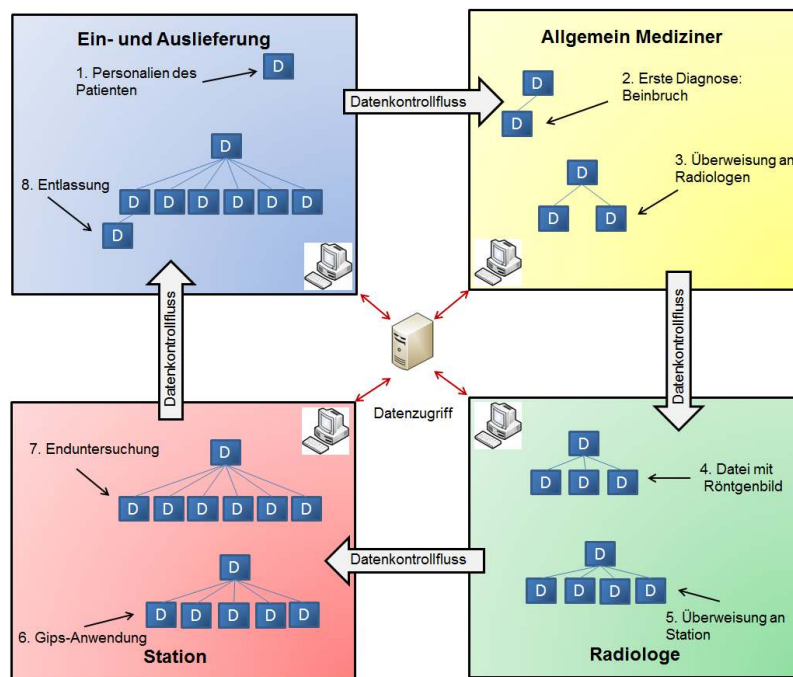


Bild 5.9: Skizze eines DPMS-Anwendungsfalls

Kommt der Patient zur Anmeldung, wird das übergeordnete Dokument erstellt. Bei jeder weiteren Station werden die Datenobjekte um die jeweils anfallenden Daten ergänzt. Nach der Abschlussuntersuchung werden zu dem Abschlussuntersuchungsdatenobjekt die Entlassungsdaten hinzugefügt.

5.6 Data-driven Process Structures

Der Ansatz Data-driven Process Structures (DDPS) wurde von Department Informatik der Universität Twente entwickelt. Für die Beschreibung dieses Ansatzes wurden folgende Quellen verwendet:

- Flexibility of Data-driven Process Structures [MRH06]
- Data-Driven Modeling and Coordination of Large Process Structures [MRH07b]

DDPS beschreibt einen Ansatz, bei dem der Prozessverlauf neben der Steuerung durch die Datenzustände auch durch die Informationen, die in den Anwendungsfunktionen hinterlegt sind, beeinflusst wird.

5.6.1 Kurzdarstellung des Ansatzes DDPS

Der DDPS Ansatz besteht aus fünf Hauptkomponenten, *Datenobjekte*, *Daten-Zustände*, *Lebenszyklus*, *Anwendungsfunktionen* und *Prozess-Konfiguration*.

Im Fokus des Ansatzes befindet sich das *Datenobjekt*. Die *Datenobjekte* beinhalten die Informationen über den domänenspezifischen Anwendungsfall. Allerdings enthalten sie keine Informationen über die Funktionsausführung einer bestimmten Anwendungsfunktion. Die *Datenobjekte* besitzen je nach Geschäftsziel eine oder mehrere *Daten-Zustände*. Ein Datenobjekt liefert die Information zu einer Entität der realen oder der konzeptionellen Außenwelt. Aus diesem Grund können mehrere *Datenobjekte* in einer Prozessinstanz existieren.

Ein *Daten-Zustand* beschreibt eine Konfiguration der Attribute in dem Datenobjekt. Eine Konfiguration beschreibt eine Momentaufnahme der Attributwerte in einem Datenobjekt. Während der Modellierungsphase werden alle *Daten-Zustände* beschrieben, die ein Datenobjekt während der Ausführungsphase einer Prozessinstanz annehmen kann. Der Übergang wird mittels eines *Lebenszyklus* beschrieben.

Der *Lebenszyklus* beschreibt die Übergänge von einem *Daten-Zustand* zum nächsten. Jedes *Datenobjekt* besitzt seinen eigenen *Lebenszyklus*. *Daten-Zustände* von einem *Datenobjekt* können von anderen *Datenobjekten* abhängig sein. Somit kann der *Lebenszyklus* eines einzelnen *Datenobjektes* nicht losgelöst von den anderen *Datenobjekten* betrachtet werden. Die Übergänge in einem *Lebenszyklus* werden mittels *Anwendungsfunktionen* realisiert. In der *Prozess-Konfiguration* werden Regeln beschrieben, die *Anwendungsfunktionen* mit den *Datenobjekten* verbinden. *Anwendungsfunktionen* werden in Aktionen untergliedert. Abhängig von dem Abarbeitungsstatus der Aktionen in einer *Anwendungsfunktion*, ergibt sich hieraus der Anwendungsfunktionsstatus. Die Erzeugung von *Anwendungsfunktionen* wird durch Templates gesteuert und kann mittels Regeln verschiedenen *Datenobjekten* zugeordnet werden.

Die *Anwendungsfunktionen* werden vom Prozessmodellierer entwickelt. Im Gegensatz dazu werden *Datenobjekte* und *Daten-Zustände* von Datenmodellierern entwickelt. Diesen Zusammenhang veranschaulicht folgende Abbildung 5.10. Veränderungen an *Datenobjekten* oder *Anwendungsfunktionen* können eingefügt werden. Allerdings kann dies zum Wiederholen von *Daten-Zuständen* führen, weil die Ausgangsbindungen durch die neuen Zustände verändert werden können.

Alle Prozess-Elemente werden auf Typ-Ebene beschreiben. Dies ermöglicht die Wiederverwendung von einzelnen Elementen für eine neue Prozessinstanz. Allerdings müssen

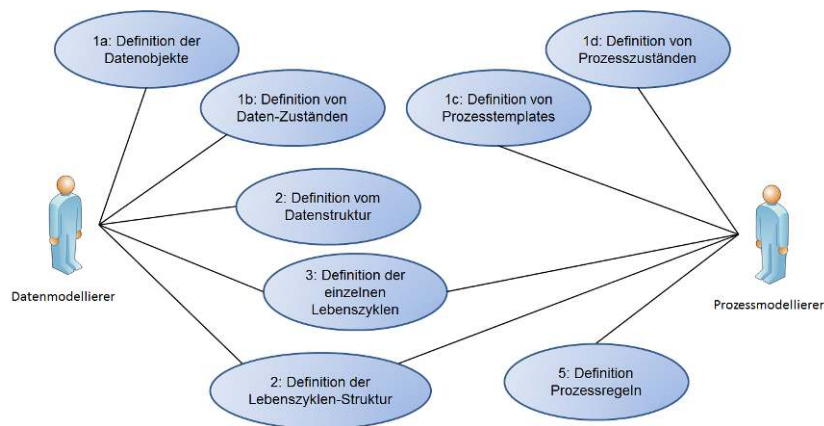


Bild 5.10: DDPS Modellierungsschritte

die Regeln zum Verbinden von *Datenobjekten* und *Anwendungsfunktionen* jeweils neu definiert werden.

5.6.2 Beispiel für die Anwendung von DDPS

In diesem Ansatz wird keine Aussage getätigt, ob die Datenobjekte auf einem zentralen Server oder nur auf lose gekoppelte Clients arbeiten. Allerdings wird ein allgemeines Regelsystem beschrieben, welches die Anwendungsfunktionen mit den Daten verbindet. Dabei kann angenommen werden, dass eine zentrale Verwaltung existiert. Für das nachfolgende Beispiel wird dies angenommen.

Als Anwendungsfall für DDPS kann das Beispiel eines Beinbruchs und dessen Behandlung benutzt werden. Für die Behandlung des Beinbruchs wird ein Datenobjekt erstellt, in dem die einzelnen Daten-Zustände, wie Einlieferung, Erst-Untersuchung, Röntgen und die aus der Anamnese resultierende Entscheidung über die Fortführung der Behandlung modelliert werden. Der Lebenszyklus beschreibt die Übergänge zwischen den einzelnen Schritten als Reihenfolge. Die Anwendungsfunktionen werden nur über die Bearbeitung der Informationen im Datenobjekt mittels der Prozess-Konfiguration nach Vorgabe des Lebenszyklus verbunden. Ein allgemeines Beispiel wird in der Abbildung 5.11 skizziert. Dabei werden die Informationssysteme nicht betrachtet. Datenobjekte werden durch spezifische Regeln mit den Anwendungsfunktionen verbunden. Dies wird durch ein Rechteck um die Anwendungsfunktionen dargestellt. Zu jedem Schritt dieser Behandlung existiert eine Abbildungsregel zwischen Datenobjekt und Anwendungsfunktion.

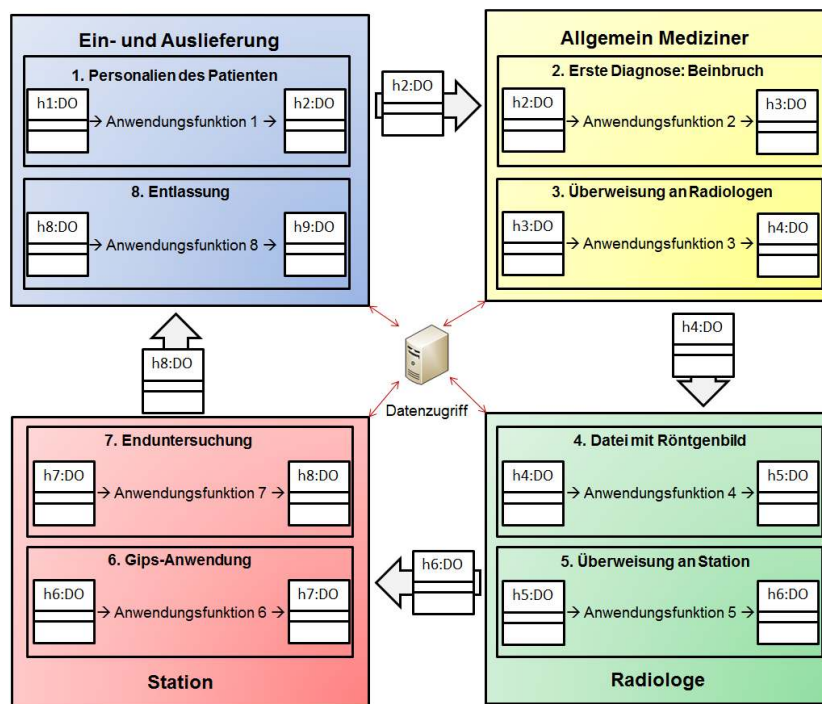


Bild 5.11: Skizze eines DDPS-Anwendungsfalls

5.7 Gemeinsame Eigenschaften der Ansätze zur Prozessunterstützung

In diesem Kapitel wurden die einzelnen Ansätze zur Prozessunterstützung, die im Verlauf dieser Arbeit evaluiert werden, beschrieben. Alle Ansätze orientieren sich zu einem gewissen Grad an dem inhaltsbasierten Paradigma. Außer BPMN unterstützt jeder betrachtete Ansatz die Steuerung eines Prozesses mittels der gespeicherten Daten.

α -Flow, ACM, ACBPM, DPMS und DDPS beschreiben einen Ansatz, bei dem ein Datenobjekt den Verlauf des Prozesses steuert. Bei α -Flow besitzt das Dokument aktive Eigenschaften, welche es einem Anwender erlauben mit den Daten zu interagieren. Die anderen Ansätze, ACBPM, DPMS und DDPS, beschreiben Passive Datenobjekte, die mittels der Konfiguration in den Attributwerten, die Ausführung des Prozess steuern. Der Ansatz ACM verwendet ein zentrales DBS, um die Daten zu speichern. Bei ACM wird der Verlauf mit Regeln gesteuert. Veränderungen und Anpassung von Prozesselementen müssen die vorabdefinierten Regeln befolgen. Die Regeln werden aber durch Datensätze definiert. DDPS entwickelt Daten und Aktivitäten getrennt. Bei der Entwicklung von Aktivitäten werden diesen unterschiedliche Zustände zugeordnet, die einen Subprozess in

einer Aktivität ermöglichen und somit eine zweite Steuerungsmöglichkeit des Prozesses darstellt. BPMN steuert den Verlauf eines Prozesses mit Hilfe des Sequenzflusses von Aktivitäten, die nicht von den Daten abhängig sind. Der Prozesslauf in α -Flow basiert nicht auf modellierten Aktivitäten, sondern im Verlauf einer α -Episode definieren neu angeforderte α -Cards, die als Platzhalter in der Koordinierungsliste angelegt werden, den Prozessfortschritt. Dieser wird als sukzessive Abarbeitung von angeforderten Inhalts- α -Cards interpretiert. Die Anpassungen und der Verlauf eines Prozesses werden durch die gespeicherten Daten in α -Flow beschrieben. ACBPM kann durch seine Dimensionen auch durch einen Sequenzfluss von Aktivitäten gesteuert werden. In dieser Arbeit wird bei dem ACBPM-Ansatz aber von einer Steuerung durch die Datenobjekte ausgegangen. Die beschriebenen Eigenschaften werden in der Tabelle 5.1 visualisiert.

| Ansatz | Art der Daten Aufbewahrung | Datensteuerung | Aktivitätssteuerung |
|----------------|----------------------------|----------------|---------------------|
| α -Flow | Dokument | JA | JA |
| BPMN | Datenobjekt | Unterstützend | JA |
| ACM | Datenbanksystem | JA | JA |
| ACBPM | Datenobjekt | JA | (JA) |
| DPMS | Datenobjekt | JA | NEIN |
| DDPS | Datenobjekt | JA | JA |

Tabelle 5.1: Zusammenfassung der Steuerung der Ansätze

Die Tabelle 5.1 veranschaulicht in vereinfachender Weise, wie stark sich allein die inhaltsbasierten Ansätze voneinander unterscheiden.

6 Vergleichsparameter

In dem letzten Kapitel 5 wurden die unterschiedlichen Ansätze zur Prozessunterstützung analysiert. Bei deren Beschreibung wurde besonderer Wert auf die Thematisierung der unterschiedlichen Eigenschaften der Ansätze gelegt. Die Eigenschaften der einzelnen Ansätze werden in dem nun folgenden Kapitel als Vergleichskriterien herangezogen.

Die Vergleichskriterien können in acht Aspekte von Systemen zur Prozessunterstützung eingeordnet werden. Zu den Aspekten gehören:

| | |
|--------------------------------------|--|
| die Gesamtmethodische Aspekte | die Integrative Aspekte |
| die Funktionale Aspekte | die Technisch-operative Aspekte |
| die Informative Aspekte | die Software-operative Aspekte |
| die Dynamische Aspekte | die Evolutionäre Aspekte |

Tabelle 6.1: Aufzählung der Aspekten

Zu erwähnen ist, dass ACM Eigenschaften beschreibt, ohne wirkliche technische Implementierungen zu nennen. Für den Vergleich wird angenommen, dass ACM diese Eigenschaften erfüllen kann

Diese acht Aspekte werden in dem Abschnitt 6.1 beschrieben. Im Anschluss folgt im Abschnitt 6.2 die Definition des nachfolgend verwendeten Vokabulars. Die Definition wurde notwendig, da in den betrachteten Ansätzen zur Prozessunterstützung unterschiedliche Begriffe für gleiche Prozesselemente in einem Ansatz verwendet werden. Diese sollen nun im Abschnitt 6.2 vereinheitlicht werden.

In den Abschnitten 6.3.1 bis 6.3.8 werden die einzelnen Vergleichskriterien unter Berücksichtigung der jeweiligen Aspekte beschrieben. Deren Zusammenarbeit wird abschließend in Kapitel 6.4 diskutiert.

6.1 Aspekte eines Ansatzes zur Prozessunterstützung

Die in der Einleitung beschriebenen Aspekte resultieren aus der Prozessmodellierung [MTJ⁺10] [CKO92] und dem Workflow-Metaschema-Modell nach [JBS97] [JB96] und

wurden für die Einordnung der Vergleichskriterien erweitert. Die Workflows sind Teil von Systemen zur Prozessunterstützung und können aus diesem Grund heran gezogen werden. Die Aspekte beschreiben diejenigen Komponenten, auf Basis derer ein Ansatz zur Prozessunterstützung aufgebaut sein kann. Im Folgenden werden die acht Aspekte beschrieben, die für den späteren Vergleich verwendet werden.

- Die **Gesamtmethodischen Aspekte** beschreiben die grundlegende Ausrichtung eines Ansatzes und repräsentieren die Basis für die weiteren Aspekte. Dazu zählt beispielsweise die Speicherungsart der Informationen oder die mögliche soziale Interaktion.
- Die **Funktionalen Aspekte** beschreiben, welche allgemeinen Funktionalitäten die Ansätze erfüllen. Dazu zählen beispielsweise, die Echtzeitfähigkeit oder die Langzeitfähigkeit.
- Die **Informativen Aspekte** beschreiben diejenigen Informationen (Daten), die von dem jeweiligen Ansatz benutzt werden. Dazu zählen u.a. die Struktur der Information, der Inhalt der gespeicherten Informationen und das Datenschema.
- Die **Dynamischen Aspekte** beschreiben das Verhalten eines Ansatzes im Normalbetrieb oder im Fehlerfall.
- Die **Integrativen Aspekte** beschreiben die Zusammenführung von unterschiedlichen Eigenschaften eines Ansatzes, um eine ganzheitliche Funktionalität zu garantieren.
- Die **Technisch-operativen Aspekte** beschreiben die technische Systemeigenschaften eines Ansatzes.
- Die **Software-operativen Aspekte** beschreiben den Umfang der Verwaltungsfunktionalitäten eines Ansatzes.
- Die **Evolutionären Aspekte** beschreiben die Anpassungsfähigkeit eines Ansatzes.

Die Aspekte bieten eine Struktur, in welche die Vergleichskriterien eingeordnet werden können, die später in diesem Kapitel beschrieben werden.

6.2 Begriffserklärung

In Kapitel 5 wurden die unterschiedlichen Ansätze zur Prozessunterstützung beschrieben. Die Autoren verwenden bei der Definition der jeweiligen Ansätze unterschiedliche Begriffe für die Datenobjekte und die Operationen, die mit einem Datenobjekt interagieren.

Für den Namen eines Datenobjektes werden zwei verschiedene Definitionen: Artefakt und Datenobjekt, benutzt. Im weiteren Verlauf dieser Arbeit werden Artefakte ebenfalls als Datenobjekte bezeichnet. Die Datenobjekte werden als Container für Daten (Datencontainer) definiert.

Des Weiteren besteht Uneinigkeit über die Bezeichnung von Ansammlungen einzelner Operationen auf Datenobjekten. Für den weiteren Verlauf der Arbeit werden daher atomare Operationen als Aktionen definiert. Eine Ansammlung von Aktionen beschreibt eine Anwendungsfunktion. Anwendungsfunktionen übernehmen spezifische Aufgaben in einem Prozess.

6.3 Vergleichskriterien der unterschiedlichen Aspekte

In den folgenden Unterabschnitten werden aufgeführt und erläutert. Die Vergleichskriterien wurden aus den unterschiedlichen Eigenschaften und Ideen der einzelnen Ansätze, die im Kapitel 5 beschrieben wurden, abgeleitet. Um alle Eigenschaften der Ansätze wertefrei vergleichen zu können, wird bei den Vergleichskriterien keine Vorauswahl getroffen.

6.3.1 Gesamtmethodische Aspekte

Modellierungsmethoden

Die Aufgabe eines Prozesses ist es ein definiertes Geschäftsziel zu erreichen. Um dieses Ziel zu erreichen, wird eine Abbildung der jeweiligen Institutionsstruktur benötigt, welche ein Prozessunterstützungssystem einsetzen will. In einer Institution arbeiten unterschiedliche Personen, die jeweils spezifische Aufgaben wahrnehmen können / sollen und unterschiedliche Fähigkeiten besitzen. Diese Eigenschaften der einzelnen Mitarbeiter oder auch der Geschäftspartner werden an bestimmten vordefinierten Positionen in einem Prozessverlauf benötigt.

Neben diesen personenbezogenen Informationen bildet die IT-Infrastruktur eine Kernkomponente für den Einsatz eines Prozessunterstützungssystems. Für die Analyse der vorhandenen Strukturen einer Institution können beispielsweise Organigramme, IT-Infrastrukturdiagramme, Personalhierarchiediagramme oder Fähigkeitsmodelle herangezogen werden. Die aufgeführten Diagramme sind Beispiele, die von den Ansätzen in einer eigenen Ausprägung verwendet werden können. Die Abbildung 6.1 skizziert eine Möglichkeit, eine Baumhierarchie einer Institution grafisch zu erfassen.

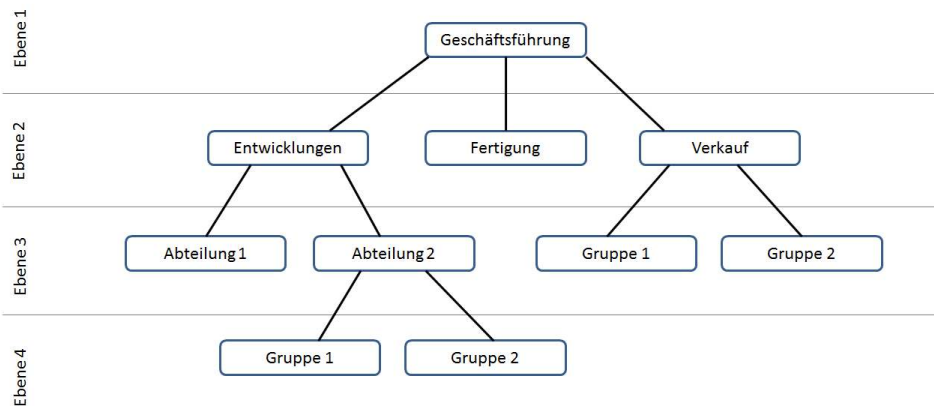


Bild 6.1: Abbildungsbeispiel der hierarchischen Struktur einer Institution

Des Weiteren müssen Analysen dahingehend durchgeführt werden, welche Daten und Anwendungsfunktionen für das Erreichen der angestrebten Geschäftsziele notwendig sind.

Explizite Definition von Lebensphasen in einem Prozess

Der Lebenszyklus eines Prozesses kann in mehrere Phasen unterteilt werden. Die Unterteilung ist beispielhaft in der folgenden Grafik wiedergegeben. Der dargestellte Zyklus 6.2 wird auch BPM-Lebenszyklus genannt. Der BPM-Lebenszyklus beschreibt eine Möglichkeit der Ausprägung von Phasen des Lebenszyklus eines Prozesses. Je nachdem welcher Ansatz verwendet wird, können die Phasen selbst und ihre Anzahl variieren.

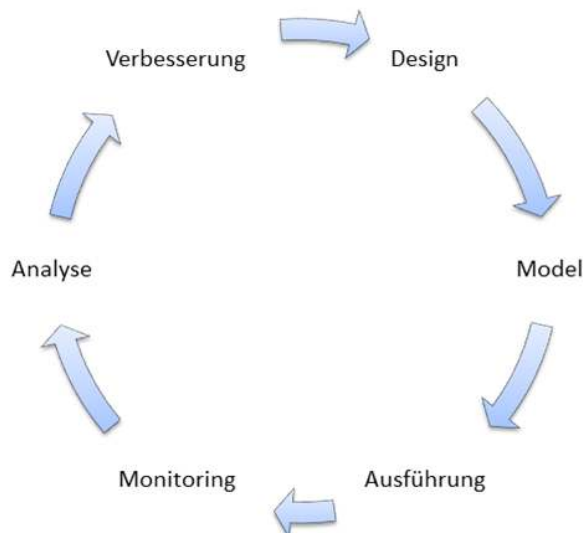


Bild 6.2: Lebensphasen von Prozessen nach [MBR05]

Analyse der Prozesse

Prozesse werden für unterschiedliche Zwecke analysiert. Ziel ist es hierbei einen Prozess effizienter und effektiver zu gestalten, um gegebenenfalls IT-Ressourcen, Energie und / oder Kosten zu sparen. Zur Optimierung eines Prozessablaufes können Sicherheits-, Anpassungs-, Zeit- und Systemanalysen angewandt werden. Letztendlich sollte auch die Zielorientiertheit des Prozesses betrachtet werden. Die Ergebnisse dieser Analysen sind der notwendige Input mit dessen Hilfe der Prozessablauf effizienter gestaltet werden kann.

Datenobjekt oder Rekord basierte Datenspeicherung

Für die Speicherung von Daten werden zwei unterschiedliche Ansätze von Quellen beschrieben. Der erste Ansatz beschreibt die Speicherung der Daten in einem Datenobjekt. Dabei wird ein Datenobjekt als eine Sammlung von Attributen beschrieben, welche die Informationen speichern. Der zweite Ansatz, die Rekord basierte Datenspeicherung, beschreibt die Speicherung von Informationen in einem DBS.

Zusammenführung von Nutzdaten, Verwaltungsdaten und Verwaltungssystem

Für die Steuerung eines Prozesses werden unterschiedliche Verwaltungssysteme benötigt. Die Verwaltungssysteme benötigen Informationen, welche Aufgabe sie auf bestimmten Nutzdaten¹ ausführen sollen. Verwaltungssysteme, Verwaltungsdaten und Nutzdaten können jeweils einzeln oder nicht trennbar implementiert werden. Weiter gibt es die Möglichkeit Verwaltungssystem und Verwaltungsdaten zu als ein Element zu implementieren und Nutzdaten separiert zu implementieren. Die vierte Möglichkeit ist die Nutzdaten mit den Verwaltungsdaten zusammen zu führen und die Verwaltungssysteme extra zu implementieren. Jeweils eins der vier Möglichkeiten wird von den betrachteten Ansätzen realisiert.

Kommunikationsmittel für soziale Interaktion

Prozessansätze werden von mehreren Anwendern benutzt. Somit lesen und schreiben Anwender auch auf die gleichen Daten. Dabei besteht das Problem der Koordinierung nicht nur auf der technischen Ebene, sondern auch auf der sozialen Ebene. Arbeiten

¹Daten, die zum Erreichen eines Geschäftsziels von Nöten sind, aber nicht zur Verwaltung verwendet werden

mehrere Personen auf den gleichen Daten entsteht eine Gruppendynamik. Die von der Personalhierarchie der beteiligten Institution abhängt. Die digitale stellt Probleme dar, weil eine direkte Kommunikation ausgeschlossen wird. Die Kommunikation ist eine wichtige Basis der Bürokommunikation [PR84]. Die Kommunikation kann auf digitaler Basis umgesetzt werden. Durch die Kommunikation sollen sich engere soziale Strukturen ergeben und das Einhalten von Ordnungs- und Arbeitsrahmen soll gewährleistet werden [Ger07]. Die betrachteten Ansätze können unterschiedliche Formen von Kommunikation etablieren. Beispielsweise können die Daten eines Prozesses, explizit für die Nachfolger lesbar gemacht werden

Natürliche Sprache als Prozess-Beschreibungssprache

Anwendungsfunktionen, Datenschemata und Regeln für das Regelsystem müssen für einen Prozess entwickelt und beschrieben werden. Für die Beschreibung der Prozesselemente kann gegebenenfalls eine strukturierte Sprache¹ verwendet werden, wie nach dem Semantics of Business Vocabulary and Business Rules (SBVR)-Konzept [Pit10] veranschaulicht wird. Dabei wird eine strukturierte Sprache in ausführbaren Code umgesetzt. Alternative können Ansätze zur Prozessunterstützung mittels Programmiercode beschrieben werden. Bei der Realisierung von unterschiedlichen Beschreibungssprachen können unterschiedliche Wege von den betrachteten Ansätzen gegangen werden. Es wäre auch möglich eine natürliche Sprache zu wählen, die keine Struktur enthält. Das Beispiel 6.3 skizziert die Umsetzung der natürlichen Sprache in Programmiercode.

Anpassbare Benutzeroberfläche

Prozessanwender unterliegen unterschiedlichen Aufgaben. Somit erfüllt jeder Prozessanwender unterschiedliche Funktionen in einem Prozessunterstützungssystem. Für jeden Prozessanwender sollte es möglich sein, die Benutzeroberfläche entsprechend seiner Wünsche und / oder Vorlieben anzupassen. Beispielsweise sollte es möglich sein einzelne Funktionen auf die oberste Ebene der Benutzeroberfläche zu ziehen.

¹Verwendung eines bestimmten Satzaufbaus oder bestimmter Schlüsselwörter zum Beschreiben von Prozesselementen

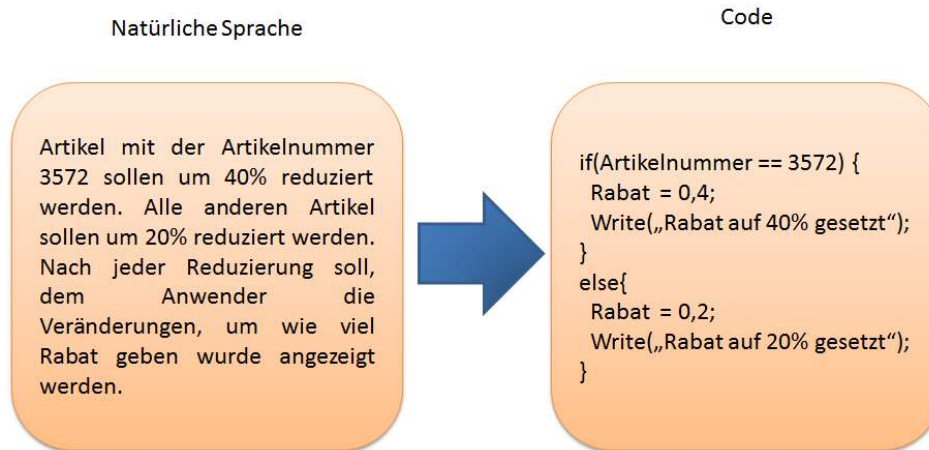


Bild 6.3: Beschreibungssprachen für Prozesselementen

6.3.2 Funktionale Aspekte

Benutzerinteraktion

Eine Anwendungsfunktion kann in zwei unterschiedliche Klassen aufgeteilt werden:

- Eine Anwendungsfunktion kann autark sein und kommt ohne Benutzerinteraktion aus. Dies kann mit einem Batchprozess verglichen werden. Nur bei Fehlersituationen wird eine Nachricht an den jeweiligen Anwendungsfunktionsbetreuer gesendet.
- Eine Anwendungsfunktion kann für ihre Abarbeitung Kontakt zum Benutzer aufnehmen. Dafür steht ihr eine Benutzeroberfläche zur Verfügung. Auf diese Art entsteht eine Kommunikation zwischen Benutzer und Anwendungsfunktionen.

Ereignisbehandlung

Eine Ereignisbehandlung analysiert die auftretenden Ereignisse, wie beispielsweise Warnungen, Meldungen oder Fehler, die während des Verlaufs eines Prozesses auftreten können. Folgende Fehlermeldungen könnten beispielsweise auftreten:

- Berechtigung fehlt
- Daten beschädigt. Daten können nicht gelesen werden
- Regelsystem verbietet eine Veränderung

Zudem können weiterhin noch beispielsweise die folgenden externen Ereignisse auftreten:

- Systemfehler (Hardware defekt)
- Eintreffen von Signalen von anderen Anwendungsfunktionen
- Eintreffen von Informationen, die in den Prozess integriert werden müssen

Die daraus resultierende notwendige Anpassung des Prozesses kann beispielsweise durch die Veränderung der Anwendungsfunktionsreihenfolge oder durch Hinzufügen oder entfernen von Anwendungsfunktionen erfolgen. Die Ansätze zur Prozessunterstützung besitzen unterschiedlich umfangreiche Ereignisbehandlungen. Sie können beispielsweise nur auf interne und / oder auf externe Ereignisse eingehen. Zugleich existieren unterschiedliche Vorgehensmöglichkeiten, wie beispielsweise das signalisieren von Ereignissen bis hin zur aktiven Korrektur von fehlerhaften Zuständen.

Online- / Offline- Modus

Wenn mobile Informationssysteme in einen Prozessverlauf eingebunden werden, dann können diese auch vom Kommunikationsnetz absichtlich oder unabsichtlich getrennt werden. Hierfür können Mechanismen etabliert werden, wie beispielsweise ein offline Modus, der es erlaubt ohne Verlust an Leistung oder gar Datenverlust Anwendungsfunktionen und / oder Datenobjekte vom Netz zu trennen und wieder zu verbinden. Gleichzeitig kann der Prozess über den Offlinestatus von Daten oder Anwendungsfunktionen informiert werden. Eine Trennung von anderen Anwendungsfunktionen kann auch durch die Unterbrechung des Kommunikationsnetzes erfolgen. Die spontane Unterbrechung, Abtrennung spezifischer Anwendungsfunktionen oder Daten sollte ein Prozess ermitteln können, um die Konsistenz der Ausführung zu gewährleisten.

Echtzeitfähigkeit von Anwendungsfunktionen

Die Anwendungsfunktionen sollten auf anfallende Ereignisse, wie Fehlermeldungen, Warnungen oder Meldungen in Echtzeit reagieren können. Eine verzögerte Behandlung von Ereignissen kann dazu führen, dass sich die Prozessausführung verlängert, falsche Ergebnisse produziert und der Prozess wiederholt werden muss oder im schlimmsten Fall der Prozessablauf abbricht.

Abhängigkeit von Anwendungsfunktionen

Es können Abhängigkeiten zwischen Anwendungsfunktionen existieren. Eine Anwendungsfunktion kann eine andere Anwendungsfunktion voraussetzen. Für die Ausführung einer bestimmten Anwendungsfunktion muss eine andere vorab ausgeführt werden.

Granularität von Anwendungsfunktionen

Eine Aktion kann eine atomare Operation auf Daten sein. Innerhalb einer Anwendungsfunktion können einzelne Aktionen hierarchisch angeordnet werden. Eine Beschränkung der Tiefe und Breite der hierarchischen Anordnung muss nicht vorgesehen sein. Eine derartige Struktur kann eine Parallel-Verarbeitung innerhalb einer Anwendungsfunktion darstellen. Die Abbildung 6.4 enthält ein Beispiel für eine Aktionshierarchie.

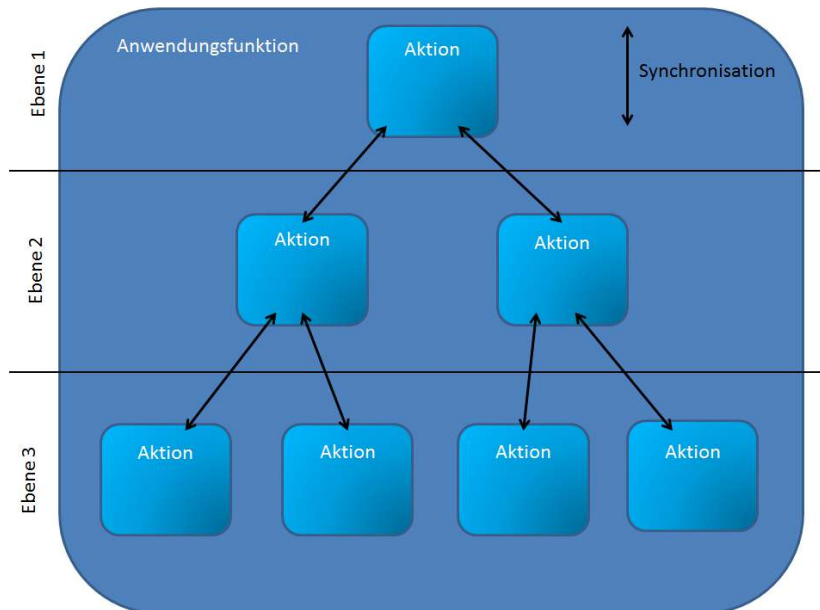


Bild 6.4: Hierarchie von Aktionen

Wiederverwendbarkeit von Prozesselementen

Für jede Prozessinstanz wird eine bestimmte Anzahl von Prozesselementen benötigt. Zu den Prozesselementen gehören beispielsweise Anwendungsfunktionen, Daten, Regeln usw. Die Prozesselemente können nach der Beendigung einer Prozessinstanz als Vorlagen für neue Prozesselemente archiviert werden. Für die Instanziierung einer neuen Prozessinstanz, können die Vorlagen verwendet und gegebenenfalls für die neue Instanz modifiziert werden.

Langzeiteinsatzfähigkeit

Differenzierung der Ausführungszeit einer Prozessinstanz nach Kurzzeit (Stunden, Tage) und Langzeit (Woche, Monate)

6.3.3 Informative Aspekte

Aktive Datenobjekte

Die Datenobjekte können in zwei verschiedene Kategorien eingeteilt werden; in aktive und passive Datenobjekte.

Aktive Datenobjekte können ausführbare Dateien sein, die aktiv den Zugang zu den Nutzdaten verwalten. Zusätzlich können aktive Datenobjekte Systemfunktionen beinhalten. Beispielsweise kann das Zugriffssystem, Regelsystem oder Protokollierungssystem integriert sein. In aktiven Datenobjekten wird nicht weiter auf die interne Speicherstruktur eingegangen, weil die Dokumente den Zugriff selbst verwalten und Anwendungsfunktionen eine Schnittstelle für die aktiven Dokumente besitzen.

Passive Datenobjekte stellen nur eine Speicherstruktur zu Verfügung. Als Speicherstruktur kommen beispielsweise DBS XML-ähnliche Dokumente in Frage. Folgende Grafik 6.5 veranschaulicht ein Datenobjekt welches in zwei Bereiche aufgeteilt werden kann, und zwar in die Verwaltungsinformationen und die Nutzinformationen. Verwaltungsinformationen beinhalten beispielsweise die Identifikationsnummer, Informationen zum Rollen- und Regelsystem.

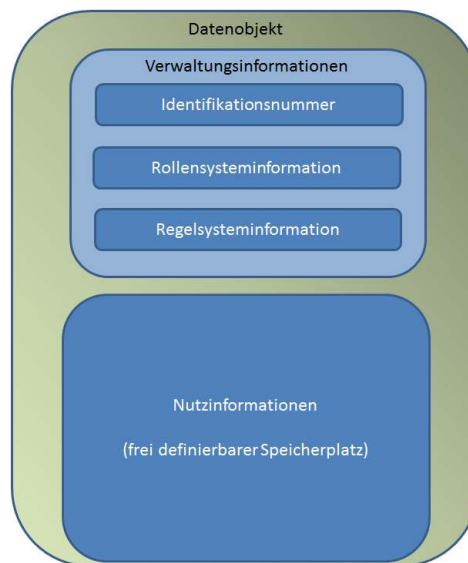


Bild 6.5: Aufbau eines Datenobjektes

Inhalt: Nutz- und Verwaltungsdaten

Datenobjekte können neben den reinen Nutzinformationen noch Verwaltungsinformationen beinhalten. Zu den Verwaltungsdaten zählen die Daten, welche für die Ausführung

eines Prozesses benötigt werden. Diese Daten werden nicht für das Erreichen eines Prozessziels benötigt, sondern für die Administration einer Prozessinstanz [PF88]. Zu den Verwaltungsinformationen können beispielsweise die Protokollierungsdatei, die Zugriffsrechte und die Versionierungsinformationen über Datenobjekte gehören. Abhängig von der Definition des Prozesses können unterschiedliche Verwaltungsdaten im Datenobjekt gespeichert werden.

Identifikation von Daten

Daten zu einer Entität der Außenwelt müssen innerhalb eines Prozesses unterscheidbar sein. Für unterschiedliche Entitäten und deren Instanzen können unterschiedliche administrative Bedingungen in einem Prozess existieren. Dazu zählen beispielsweise unterschiedliche Zugriffsrechte für Benutzer oder die differenzierte Betrachtung für Einträge in einer Protokolldatei (Log Files) ¹. Für die eindeutigen Identifizierung von einer Entität in einem Prozess kann beispielsweise in einem Attribut eine Identifizierungsnummer gespeichert werden oder ein eindeutiger Name.

Speicherung der Prozessverlaufsinformationen in den Daten

Daten können den weiteren Verlauf eines Prozesses, beispielsweise welche Anwendungsfunktionen auf die Daten als nächstes ausgeführt werden, bestimmen. Es existieren folgende Möglichkeiten:

- Es können spezifische Attribute in den Daten existieren, die den weiteren Verlauf einer Prozessausführung speichern. Darunter fallen, welche Anwendungsfunktionen auf die Nutzdaten ausgeführt werden sollen. Diese Daten stellen Verwaltungsdaten dar, die zusätzlich zu den Nutzdaten gespeichert werden müssen.
- Die Daten werden im System zur Prozessunterstützung als Attribute gespeichert. Die Attribute besitzen während der Ausführung immer bestimmte Werte. Diese Konfiguration der Attributwerte stellen Datenzustände dar. Anhand der Datenzustände werden die Anwendungsfunktionen zugeordnet. In diesem Fall, fallen keine spezifischen Verwaltungsdaten an, da die Nutzdaten für die Verwaltung des Verlaufs verwendet werden.

Die betrachteten Ansätze können eine der Varianten erfüllen oder auch eine Kombination der beiden Möglichkeiten realisieren.

¹Bsp: Wer hat welche Änderungen an Daten X durchgeführt

Anzahl der Datenobjekte für einen Anwendungsfall

Bezüglich der Anzahl der Datenobjekte können während der Ausführung mehrere unterschiedliche Konzepte existieren. Entweder werden alle Informationen, die für eine Prozessinstanz benötigt werden, in ein Datenobjekt gespeichert oder die Informationen werden auf mehrere Datenobjekte verteilt. Beim zweiten Fall kann durch eine Veränderung der Außenwelt die Anzahl der Datenobjekte während der Laufzeit variieren.

Einheit der Validierung

Im Verlauf eines Prozesses können die Werte von Attributen manuell durch Anwender oder automatisch durch Anwendungsfunktionen verändert werden. Bei vielen Ansätzen zur Prozessunterstützung arbeiten mehrere Personen und/oder Anwendungsfunktionen gleichzeitig auf den Daten. Bei einer Modifikation der Daten ist die Frage zu beantworten, wann der Inhalt der Daten für alle Prozessteilnehmer gültig wird und die Prozessteilnehmer auf den Daten arbeiten können. Alternative können Daten nur exklusiv von einem Prozessanwender verwendet gelesen und bearbeitet werden. Andere Anwender oder Anwendungsfunktionen können während der Bearbeitung nicht lesend auf den Daten zugreifen und müssen warten bis ein Veränderungsvorgang abgeschlossen ist.

Im Prozessablauf werden oft Daten verändert. Dabei stellt sich die Frage, wie feingranular diese Änderung sein kann. Entweder beziehen sich Veränderungen auf ein Datenobjekt oder auf ein Attribut. Bei der Attributlösung können die anderen Attribute im Datenobjekt, was bei der Datenobjektlösung nicht möglich ist.

6.3.4 Dynamische Aspekte

Daten- oder Aktivitätssteuerung

Die Steuerung des Prozesses kann auf mehreren Arten erfolgen. Anwendungsfunktionen können entweder mittels Daten (content-oriented), oder durch die Steuerung durch Sequenzfluss von Anwendungsfunktion (activity-oriented) verwaltet werden.

Wird der Prozessablauf mit Hilfe der Daten gesteuert, dann befinden sich die Informationen, welche Anwendungsfunktionen ausgeführt werden sollen, in den Daten, die beispielsweise per DBS oder Datenobjekt gespeichert werden. Bei der Steuerung des Prozesses durch den Sequenzfluss der Anwendungsfunktionen, beschreibt der Sequenzfluss die Reihenfolge der Anwendungsfunktionen im Prozessverlauf durchgeführt werden.

Datenlebenszyklus

Während der Partizipation an einem Prozess werden Informationen mittels mehrerer Attribute gespeichert. Enthalten diese Attribute Werte, die zum Zeitpunkt der Betrachtung stabil sind, dann stellen die momentanen Werte eine Konfiguration des Datenobjektes dar.

Der Lebenszyklus beschreibt den Verlauf von dem ersten Instanzieren der Datenschemata bis zum Zeitpunkt an dem das Schema nicht mehr verwendet wird. Der Datenlebenszyklus kann als endlicher Automat definiert werden, bei dem die Konfigurationen die Knoten darstellen. Um von einer Konfiguration zur nächsten zu gelangen, müssen Anwendungsfunktionen auf die Datenobjekte angewendet werden. Die Konfigurationen mit Übergängen werden als Datenschemalebenszyklus dargestellt.

Offenes Ende oder Terminierung einer Prozessinstanz

Eine Prozessinstanz kann terminieren, wenn beispielsweise keine weiteren Aktivitäten mehr ausgeführt werden müssen, um das Erreichen eines Geschäftszieles sicherzustellen. Ist dieser Zustand erreicht, werden die Ressourcen freigegeben und die Prozessinstanz kann von keinem Anwender mehr angesprochen werden. Alternativ können Prozessinstanzen, nach Beendigung der letzten Aktivität, inaktiv werden. Soll die Prozessinstanz wieder verwendet werden, kann sie reaktiviert werden und von Benutzern wieder angesprochen werden.

Echtzeitfähigkeit von Prozessen

Ein Prozess sollte auf alle auftretenden Ereignisse, wie Fehlermeldungen, Warnungen oder Meldungen, reagieren können. Zusätzlich kann ein Prozessverlauf gegebenenfalls parallele Anwendungsfunktionen ausführen. Daher sollte überprüft werden, in wie weit der Prozess, unter Berücksichtigung dieser Forderungen, die Echtzeit garantieren kann.

6.3.5 Integrative Aspekte

Transparenz

Während des Prozessverlaufs werden viele unterschiedliche Veränderungen an einer Prozessinstanz durchgeführt. Zu den Veränderungen gehört beispielsweise das Schreiben von neuen Attributwerten, Veränderungen an den Anwendungsfunktionen und Veränderung

des Prozessverlaufs Ein Prozessteilnehmer der seine Arbeitsschritte an einer Prozessinstanz durchführt und eine Veränderung durchführen will, benötigt den Überblick über den ganzen Prozess und die Kenntnis darüber, weshalb die Prozessinstanz verändert wurde, um mittels dieser Information seine Arbeitsschritte zu verrichten. Der Informationsfluss über die Veränderungen des Prozesses zur Laufzeit wird Transparenz genannt. Die betrachteten Ansätze können unterschiedliche Wege der Transparenz verfolgen. Entweder können nur Veränderung der Daten eingesehen werden oder auch wer die Daten verändert hat. Außerdem kann die Möglichkeit bestehen, zu sehen, wie und wer den Verlauf eines Prozesses modifiziert hat. Neben den genannten Varianten existieren noch Zwischenstufen.

Inter-institutionelle Prozessausführung

Um ein bestimmtes Geschäftsziels zu erreichen können neben mehreren Abteilungen einer Institution auch verschiedene Institutionen, die räumlich voneinander getrennt sein können, in den Prozessablauf eingebunden sein. Für dieses Einsatzgebiet können die Ansätze neben den entsprechenden Kommunikationsmitteln auch die Modellierungsmöglichkeit bereitstellen, die einen inter-institutionellen Prozessablauf unterstützen.

6.3.6 Technisch-operative Aspekte

Kommunikationsnetz

Prozesse arbeiten oft nicht nur auf einem Informationssystem. Werden mehrere Kommunikationssysteme verwendet, dann muss eine Kommunikationsverbindung zwischen den Informationssystemen etabliert werden, um Daten austauschen zu können. Um Daten transferieren zu können, werden Kommunikationsnetzwerke verwendet. Für das Kommunikationsnetzwerk können unterschiedliche Topologien¹ verwendet werden. Es besteht die Möglichkeit hierfür entweder geschlossene Firmennetzwerke oder das Internet zu nutzen. Des Weiteren existieren unterschiedliche Strukturen zwischen den einzelnen Netzwerkknoten [KR01] im Netzwerk. Informationssysteme können direkt miteinander verbunden sein oder durch Verteiler (beispielsweise: Hub, Switch), die eine bestimmte Verbindungskapazität besitzen. In den Ansätzen können unterschiedliche Topologien von privaten bis zum öffentliche Bereich verwendet werden.

¹Struktur der Verbindungen

Spezialgeräte (Pads, Mobiltelefone)

Neben den traditionellen Informationsverarbeitungssystemen, wie Arbeitsplatzcomputer oder Notebooks, besteht die Möglichkeit Spezialgeräte, wie Mobiltelefon oder Pads, in einen Prozessablauf zu integrieren. Diese Spezialgeräte verfügen oft nicht über die gleiche Hardwareausstattung und können Leistungsschwächer sein als das traditionelle Informationsverarbeitungssystem. Wenn die Prozesssoftware den Einsatz von Spezialgeräte zulässt, muss sie mit unterschiedlichen Hardwarevoraussetzungen umgehen können.

Kenntnis der Systemarchitektur

Systeme zur Prozessunterstützung besitzen eine bestimmte Systemarchitektur nach der sie aufgebaut sind. Die Systemarchitektur kann aus technischer Sicht Abhängigkeiten beschreiben, wie einzelne Systemkomponenten beispielsweise angesprochen werden müssen. Die gleiche Aussage gilt auch für den Fall, dass ein Prozess modelliert werden soll. Die Prozessmodellierer sollten entweder Kenntnis von den Abhängigkeiten besitzen, oder das System zur Prozessunterstützung bietet eine Modellierungsoberfläche, welche eine Modellierung auf einer höheren Ebene zulässt und bei der Abbildung die Abhängigkeiten automatisch setzt.

6.3.7 Software-operative Aspekte

Regelsystem

Ein Regelsystem beschreibt eine Kontrollinstanz, welche die Einhaltung von Regeln kontrolliert. Beispielhafte Anwendung eines Regelsystem: Vor dem Ablauf einer Prozessinstanz werden Regeln, wie beispielsweise: Daten müssen immer verschlüsselt sein, definiert. Das Regelsystem wahrt die Einhaltung der definierten Regeln. Alternative kann das Regelsystem Regeln für die Zuordnung der Daten zu den Anwendungsfunktionen bereitstellen. Wenn ein Bruch der Regeln vorliegt, können Regelsysteme eine Abweichung von der Norm signalisieren. Dies kann durch ein einfaches Signal erfolgen oder der Prozessablauf wird unterbrochen.

Die betrachteten Ansätze können für unterschiedliche Zwecke ein Regelsystem implementieren. Diese können entweder nur Teilbereiche der Ausführung abdecken, wie beispielsweise die Zuordnung von Anwendungsfunktionen zu den Daten, oder aber die Kontrolle eines kompletten Prozessablaufs, von der Modellierung bis zu Terminierung einer Prozessinstanz.

Sicherheitssysteme

Das Sicherheitssystem garantiert den Schutz der Daten und der Anwendungsfunktionen vor dem Lesen oder der Manipulation durch Dritte. Zu dem Sicherheitskonzept können beispielsweise Autorisations-, Authentifikations-, Verschlüsselungssysteme und digitale Signaturen gehören. Ein Sicherheitskonzept kann für alle Prozesselemente oder nur für einzelne Elemente des Prozesses eingesetzt werden. Die beschriebenen Systeme können auch einzeln implementiert werden. So können zum Beispiel nur Autorisations- und Authentifikationssystem implementiert sein. Eine Verschlüsselung der Daten kann aber nicht implementiert sein. Die betrachteten Ansätze enthalten in ihrem System zu Prozessunterstützung unterschiedliche Sicherheitssysteme.

Benutzerverwaltung

Eine Benutzerverwaltung erlaubt bestimmten Benutzern den Zugang auf einen Prozess. Für diesen Zweck wird ein Benutzer mit Privilegien und Zugriffsrechten, die seine Zugriffsmöglichkeiten auf den Prozess definieren, ausgestattet. Für die Verwaltung der Privilegien und Zugriffsrechte existieren unterschiedliche Möglichkeiten. Entweder können die Rechte, die ein Benutzer benötigt direkt zugewiesen werden oder einem Benutzer wird eine bestimmte Rolle zugewiesen. Einer Rolle können wiederum Privilegien und Zugriffsrechten zugeordnet.

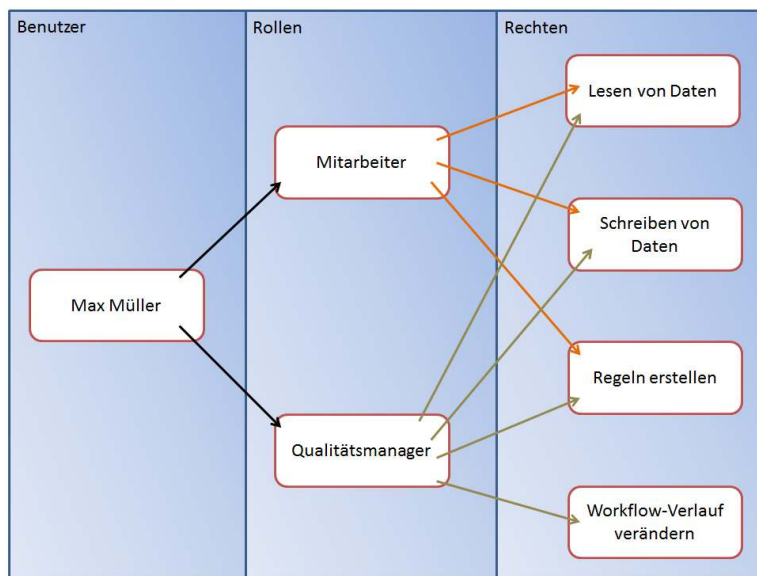


Bild 6.6: Benutzerverwaltung für ein Rollensystem

Die betrachteten Ansätze können eine einfache Benutzerverwaltung unterschiedlich implementieren. Zusätzlich können Hybridlösungen zwischen einfacher Benutzerverwaltung und Rollensystem implementiert werden.

Protokollierungssystem¹

Das Protokollierungssystem beschreibt ein System, das automatisch bestimmte Aktionen in einem Prozess protokolliert. Darunter fallen beispielsweise das An- und Abmelden eines Anwenders eines Prozessunterstützungssystems, Veränderungen an Elementen eines Prozessunterstützungssystems, Meldungen von Systemfehlern oder Ereignismeldungen resultierend aus dem Abschluss von definierten Aktivitäten.

Ein- und Ausgabeverwaltung

Ein Prozess kann Informationen aus unterschiedlichen Quellen für den Prozessablauf benötigen. Die Quellen können DBSe oder Dateien sein, die jeweils ihre eigenen Formate mitbringen können. Ein System zur Prozessunterstützung sollte in der Lage sein, die Daten der Quellen zu lesen und aufzunehmen. Für die Ausgabe von Informationen aus dem Prozessunterstützungssystem können unterschiedliche Forderungen definiert sein. Beispielsweise sollen die ausgegebenen Informationen ein bestimmtes Format besitzen oder spezielle Zusatzinformationen tragen, wie Firmenlogo, Wasserzeichen oder verschlüsselt sein. Für diese Zwecke wird ein Ausgabeverwaltungssystem benötigt, das diese vorgegeben Anforderungen realisiert.

Anmerkungssystem

Für die Dokumentation von Veränderungen kann die Möglichkeit bestehen Anmerkungen für andere Prozessanwender zu hinterlassen. Die Möglichkeit eine Anmerkung zu hinterlassen kann beispielsweise bei der Veränderung des Prozessverlaufes, bei Anwendungsfunktionsänderungen oder bei Änderungen der Struktur von Datenobjekten sinnvoll sein.

Lokalisieren von Anwendungsfunktionen

Anwendungsfunktionen eines Prozesses können auf unterschiedlichen Informationssystemen installiert sein und ihre Anzahl kann während der Laufzeit des Prozesses adaptive

¹engl. Logging

erweitert werden. Die Informationssysteme können mittels eines Kommunikationsnetzes verbunden sein. Durch die mögliche Lokalisierung von Anwendungsfunktionen müssen einem System zur Prozessunterstützung die beteiligten Anwendungsfunktionen eines Prozesses nicht bekannt sein. Nach der Beendigung einer Anwendungsfunktion müssen die Daten, je nach Prozessverlauf, an eine weitere Anwendungsfunktion weitergeleitet werden. Für die Weiterleitung der Daten werden Informationen benötigt, wo sich die nächste Anwendungsfunktion befindet. Als Alternative kann ein Anmeldungssystem implementiert sein, bei dem alle Anwendungsfunktionen registriert sind. Diese Möglichkeit bedingt allerdings eine zentrale Verwaltung, weil alle Anwendungsfunktionen wissen müssen, wo sich das Registrierungssystem befindet.

Koordinierung von Daten

Eine Koordinierung der Daten wird benötigt, wenn verschiedene Benutzer gleichzeitig auf die gleichen Daten zugreifen wollen. Es existieren unterschiedliche Möglichkeiten die parallele Arbeit auf den gleichen Daten zu organisieren und dabei die globale Konsistenz der Daten zu gewährleisten. Es können beispielsweise Duplikate der Daten erzeugt werden und das Koordinierungssystem integriert nach der Bearbeitung durch die Benutzer die Varianten zu einem global gültigen Datensatz. Dies wird als Variantensystem bezeichnet. Eine alternative Möglichkeit wäre die Anwendungsfunktionen zu informieren, welche Daten gerade geändert werden oder die Daten exklusiv zu sperren und die Anwendungsfunktionen vom Lesen und Schreiben der Daten auszuschließen. Die Koordinierungssysteme tragen dazu bei einen konsistenten Prozessablauf zu garantieren.

Die betrachteten Ansätze implementieren unterschiedliche Varianten die Koordinierung von Daten zu regeln. Die hier dargestellten Versionen von Koordinierungssystemen stellt ein Beispiel der Möglichkeiten der Koordinierung von parallelen Zugriffen auf Daten dar.

Die folgende Abbildung 6.7 veranschaulicht die Parallelverarbeitung eines Datenobjektes mit Hilfe von Duplikatnummern. Zusätzlich wird von einem System die globale Gültigkeit eines Datenobjektes gewährleistet.

Versionsverwaltung

Die Versionsverwaltung ist ein System zur Erfassung von Änderungen an Dokumenten, Daten oder anderen Prozesselemente. Die geänderten Prozesselement erhalten eine fortlaufende Versionsnummer. Wird beispielsweise eine Dateninstanz verändert, dann wird die Ursprungsdateninstanz nicht gelöscht, sondern mittels einer Versionsnummer archiviert.

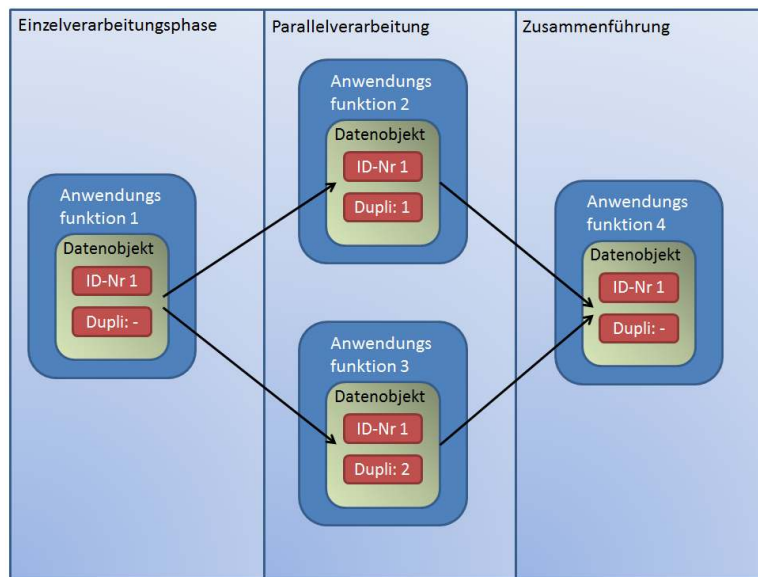


Bild 6.7: Koordination von Duplikaten

Fehlererkennung und Fehlerkorrektur

Datenfehler können durch Anwendungsfunktionen oder durch die Übertragung auf ein anderes System auftreten. Diese Fehler sollten erkannt und wenn möglich korrigiert werden. Dafür werden beispielsweise Daten redundant gespeichert oder Kontrollsummen angelegt. Beispiele für Fehler sind Bitkipper auf Datenträgern oder Übertragungsfehler beim Versenden zwischen Anwendungsfunktionen.

Verklemmungsvermeidungssystem

Eine Instanz eines Prozesses kann mehrere Datenobjekte für einen Fall besitzen. Für manche Aktionen können die Daten aus mehr als einem Datenobjekt benötigt werden. Wird zu dem noch die parallele Ausführung von Anwendungsfunktionen erlaubt, kann es zu Verklemmungen kommen. Durch Verklemmungen kann ein Prozess zum Stillstand kommen. Somit sollten Methoden zur Vermeidung von Verklemmungen, wie beispielsweise die Wait-and-Die-Strategie [RSL78], oder Auflösungsverfahren für Verklemmung, wie beispielsweise die Rollback-Strategie, in einem Workflow vorhanden sein. Die 1- und N-Datenobjektinstanz(en)-Lösung wird in der folgenden Grafik 6.8 visualisiert

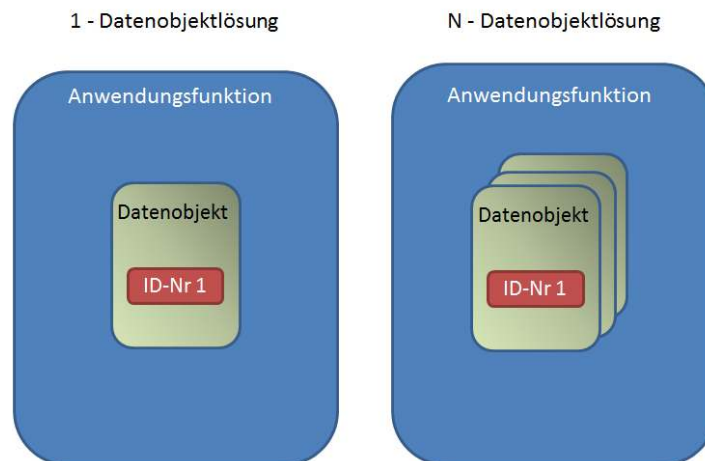


Bild 6.8: Paralleler Zugriff auf Datenobjekten

6.3.8 Evolutionäre Aspekte

Umgang mit der Unvorhersehbarkeit

In einem Prozess können entweder fest definierte Informationssysteme eingesetzt werden oder alternativ kann man mit einer dynamischen Anzahl von unterschiedlichen Informationssystemen arbeiten. Durch eine Veränderung der äußeren Rahmenbedingungen können sich aber die Geschäftsziele ändern, was eine Anpassung der Workflow-Instanz erfordert. Ein Beispiel hierfür ist die Nutzung eines neuen Lieferanten, der für einen Prozess verwendet wird (Stichwort: Software-Evolution [LK03]).

Beim Umgang mit der verteilten Unvorhersehbarkeit werden beispielsweise zwei verschiedene Arten von adaptiven Anpassungen an den Prozess betrachtet. Dem ersten Szenario (populär Szenario) liegt eine **fest** definierte Anzahl von Anwendungsfunktionen auf verteilten Informationssystemen zu Grunde, die je nach Notwendigkeit der momentan ausgeführten Prozessinstanz adaptiv partizipieren können. In diesem Szenario (populär Szenario) sind alle möglichen Anwendungsfunktionen vorab dem System zur Prozessunterstützung bekannt. Bei der zweiten Szenario (anonym Szenario) sind dem System zur Prozessunterstützung die Anzahl der möglichen Anwendungsfunktionen **nicht** bekannt. Ein Prozessanwender besitzt die Möglichkeit adaptiv, je nach Notwendigkeit, Anwendungsfunktionen zu definieren und diese in eine vorhandene Prozessinstanz während der Laufzeit zu integrieren. Des Weiteren können beispielsweise Datenobjekte, Regelsysteme, Rollen, Mitarbeiter, Rollensysteme, Zugriffssysteme oder das Routing während der Laufzeit angepasst werden. Die Abbildung 6.9 skizziert die Veränderung eines Datenobjektes und der Anwendungsfunktion während der Laufzeit.

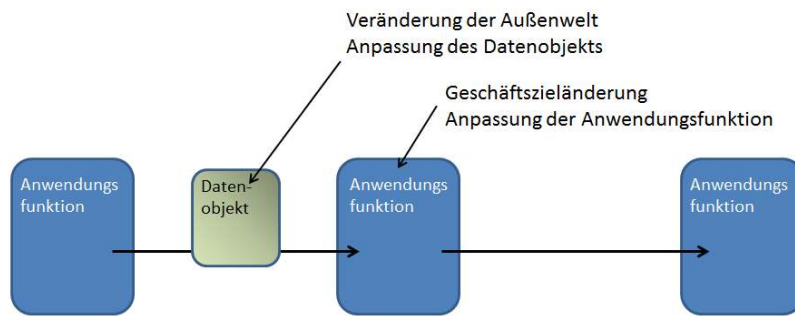


Bild 6.9: Neudefinition von Datenobjekt und Anwendungsfunktion

6.4 Aspekte der Systeme zur Prozessunterstützung und deren Zusammenhang

Die acht Aspekte beschreiben Eigenschaften auf Basis derer die Vergleichskriterien eingeordnet wurden. Die Vergleichskriterien können den einzelnen Aspekte zugeordnet werden.

Die Vergleichskriterien aus dem Abschnitt 6.3 beschreiben Eigenschaften der ausgewählten Ansätze zur Prozessunterstützung. Basierend auf den Aspekten sind diese eingeordnet worden, um Vergleichskriterien mit ähnlichen Eigenschaften definieren zu können.

7 Vergleich der unterschiedlichen Ansätze zur Prozessunterstützung

Im Kapitel 6 wurden die Vergleichskriterien, die den acht Aspekten zugeordnet wurden, beschrieben. Im Abschnitt 7.1 werden die Rahmenbedingungen und der Aufbau des Vergleichs beschrieben. Danach wird ein Vergleich der Ansätze zur Prozessunterstützung aus dem Kapitel 5 auf der Basis der Vergleichskriterien aus Kapitel 6, durchgeführt. Anhand der acht Aspekte (Abschnitte 7.2.1 bis 7.2.8) wird der Vergleich strukturiert. Zum besseren Verständnis werden Tabellen eingesetzt, die die Vergleichsergebnisse zusammenfassen. Abschließend folgt im Abschnitt 7.3 eine Bewertung der Stärken und der jeweiligen Schwächen der Ansätze. Die Ergebnisse des Vergleichs ermöglichen die Einordnung der Ansätze entsprechend den verschiedenen Paradigmen im Abschnitt 7.4.

Der Ansatz ACBPM kann sowohl als Steuerung eines Prozesses durch einen Sequenzfluss von Aktivitäten als auch durch Daten beschrieben werden. Für den weiteren Verlauf wird ACBPM als Ansatz betrachtet, der mit den Daten den Prozessablauf steuert.

Zu erwähnen ist, dass ACM Eigenschaften beschreibt, ohne wirkliche technische Implementierungen zu nennen. Für den weiteren Verlauf wird angenommen, dass ACM diese Eigenschaften erfüllen kann.

7.1 Aufbau des Vergleichs der Ansätze zur Prozessunterstützung

Um den Vergleich zu strukturieren wird dieser auf der Basis der acht Aspekte durchgeführt. Bei diesem Vergleich würden die Namen der Ansätze häufig verwendet. Zur besseren Übersichtlichkeit werden daher die Namen der Ansätze durch ihre Abkürzungen ersetzt. In der Tabelle 7.1 werden die Namen der Ansätze ihren jeweiligen Abkürzungen gegenübergestellt.

| Abkürzung | Voller Name |
|----------------|---|
| α -Flow | α -Flow |
| BPMN | Business Process Modelling and Notation |
| ACM | Adaptive Case Management |
| ACBPM | Artifact-Centric Business Process Models |
| DPMS | Datenorientiertes Prozess-Management-System |
| DDPS | Data-driven Process Structures |

Tabelle 7.1: Abkürzungen der Ansätze zur Prozessunterstützung

Die Ansätze werden mit Hilfe der acht Aspekte und der jeweiligen Vergleichskriterien miteinander verglichen. Zur Verdichtung des Argumentationsflusses und zur besseren Strukturierung werden nur diejenigen Vergleichskriterien textuell beschrieben, die von den Ansätzen explizit oder nur implizit erfüllt werden.

7.2 Vergleichsdurchführung

In den nachfolgenden Abschnitten werden die Ansätze mit Hilfe der acht Aspekte und der jeweiligen Vergleichskriterien miteinander verglichen. Es werden nur diejenigen Vergleichskriterien betrachtet, die von den Ansätzen explizit oder nur implizit erfüllt werden. Zur besseren Übersichtlichkeit werden die Ergebnisse in Tabellenform dargestellt.

7.2.1 Gesamtmethodische Aspekte

In diesem Abschnitt wird der Vergleich anhand der Vergleichskriterien der gesamtmethodischen Aspekte in folgender Reihenfolge in Tabellen dargestellt:

- Modellierungsmethoden
- Explizite Definition von Lebensphasen in einem Prozess
- Analyse der Prozesse
- Datenobjekt oder Rekord basierte Datenspeicherung
- Zusammenführung von Nutzdaten, Verwaltungsdaten, Verwaltungssystem
- Kommunikationsmittel für soziale Interaktion
- Natürliche Sprache als Prozess-Beschreibungssprache
- Anpassbare Benutzeroberfläche

| Ansatz | Vergleich: Modellierungsmethoden |
|----------------|--|
| α -Flow | Anwender, denen innerhalb einer Institution eine spezifische Rolle zugeordnet wird, können bei α -Flow an einer Prozessinstanz teilnehmen. Eine explizite Analysephase, bei der eine Fähigkeitsanalyse oder ein Organigramm erstellt werden, wird nicht beschrieben. Aufgrund seiner Eigenschaft ohne das eine Installation notwendig wird ausgeführt werden zu können, passt sich α -Flow den gegebenen System- und Hardwarebedingungen an. Aus diesem Grund ist eine Analyse der IT-Infrastruktur der eingesetzten Informationssysteme nicht notwendig. |
| BPMN | BPMN bildet beim Aufbau der Pools- bzw. Lanes- (Beschreibung der Begriffe im Anhang C) die Firmenstruktur ab. Eine Verantwortungshierarchie bzw. Organigramm wird in BPMN nicht unterstützt. Allerdings können durch Aktivitäten Datenobjekte und Lanes den Benutzergruppen direkt zugeordnet werden. Dadurch entsteht eine Hierarchie, wie sie sich im Verlauf eines bestimmten Falls in einem Workflow ergeben. Die IT-Infrastruktur wird von BPMN nicht abgebildet. Dies geschieht auf einer tieferen Ebene. Um ein Prozessziel zu erreichen, muss allerdings eine Analyse zur Erstellung des Sequenzflusses der Aktivitäten mit den Gateways durchgeführt werden. |
| ACM | ACM will die Bearbeiter eines Falles in die Lage versetzen, Änderungen an dem Arbeitsablauf durchführen. Dafür benötigt ACM Informationen auf der Basis derer es eine Analyse durchführen kann, mit dem Ziel ein Organigramm zu erstellen. Basierend auf den personenbezogenen Informationen zu den jeweiligen Anwendern und mit Hilfe des Organigramms, werden diese an definierten Stellen in den Ablauf mit eingebunden. Zusätzlich zu den Mitarbeitern müssen noch die Partner und die Kunden berücksichtigt werden. Sie werden hinzugefügt, um sicherzustellen, dass alle am Prozessablauf beteiligten in der Struktur abgebildet sind. Nach dem die Benutzer bekannt sind, werden ihnen entsprechend ihren Fähigkeiten, unterschiedliche Verantwortungsbereiche für die Prozesse übertragen. Neben den Personen in einem Prozess, spielt die IT-Infrastruktur eine entscheidende Rolle. Vorab wird die IT-Infrastruktur gesichtet und ein IT-Infrastrukturprogramm erstellt. |
| ACBPM | Eine explizite Methode zur Erstellung einer Organisationsstruktur, in der ein Prozess eingesetzt werden soll, wird von ACBPM nicht beschrieben. Ein Organigramm, Befähigungsdiagramm der Mitarbeiter oder IT-Strukturanalyse werden ebenfalls nicht beschrieben. Für unterschiedliche Anwender existieren unterschiedliche Zugriffsrechte auf die Daten. Dafür werden Analysen der Benutzer und ihrer jeweiligen Rechte benötigt. Zusätzlich müssen zum Erreichen eines Geschäftszieles Analysen durchgeführt werden, auf Basis derer die Datenobjekte und Aktivitäten bestimmten werden. Auf diesen Informationen basiert auch die Definition des Prozessablaufes, der entweder per Sequenzfluss, per Aktivitäten oder Datensteuerung erfolgt. |
| DPMS | Bei DPMS wird die Notwendigkeit einer Analyse der teilnehmenden Anwender, der Fähigkeiten der Benutzer, der Struktur der Institution oder des IT-Systems nicht explizit beschrieben. Allerdings werden die Geschäftsziele für die Modellierung der Datenobjekttypen benötigt. Anhand der Attribute der Datenobjekttypen werden dann die Geschäftsprozesse und -ziele explizit modelliert. Im DPMS-Ansatz werden verschiedenen Benutzerrechten bestimmte Rollen zugeordnet. Über die Definition der Rollen werden den Benutzern auch Zugriffsrechte auf bestimmte Prozesselemente zugewiesen. Um diese Rechte vergeben zu können, müssen die Benutzer und deren Rolle im Prozess analysiert werden. |
| DDPS | Bei DDPS werden keine Analysen genannt, die für die Modellierung einer Prozessinstanz durchgeführt werden müssen. Allerdings müssen mindestens die Geschäftsziele und die Schritte, um diese zu erreichen, analysiert werden. Ohne diese Analyse ist es nicht möglich die Datenobjektzustände zu modellieren. Es wird kein Organigramm und auch keine Analyse der Fähigkeiten von Benutzern erstellt. Auch die vorhandene IT-Struktur, auf der das System zur Prozessunterstützung eingesetzt werden soll, wird nicht analysiert. Für die Generierung der Datenobjekte und der Aktivitäten sowie der Abbildungsregeln muss sowohl eine Analyse des Geschäftsziels als auch wie dieses erreicht werden soll, durchgeführt werden. |

Tabelle 7.2: Vergleich: Modellierungsmethoden

7 Vergleich der unterschiedlichen Ansätze zur Prozessunterstützung

| Ansatz | Vergleich: Explizite Definition von Lebensphasen in einem Prozess |
|----------------|--|
| α -Flow | α -Flow beschreibt nur eine Ausführungsphase. Diese beginnt mit dem Anlegen eines neuen Falls und endet, wenn für ein bestimmtes Ziel keine neuen Informationen benötigt werden. |
| BPMN | BPMN beschreibt nur die Modellierungsphase eines Prozesses. Die darauf folgende Ausführungsphase, ist vom WfMS abhängig, dass die Umsetzung des Modells in eine laufende Instanz übernimmt. |
| ACM | Explizite Phasen werden von den ACM nicht erwähnt. Für die Erstellung des Organigramms oder des IT-Strukturdiagramms wird eine Modellierungsphase benötigt. Zusätzlich gibt es die Ausführungsphase und eine Terminierungsphase, während der die Prozessinstanz nicht mehr ausgeführt werden kann. Es existiert auch eine Monitoring-Phase, während der ein Prozess im Hinblick auf eine mögliche Optimierung analysiert wird. Nach Abschluss dieser Phasen werden die gewonnen Informationen in das System integriert. Alle beschriebenen Phasen, mit Ausnahme der Terminierungsphase, sollen in der Ausführungsphase durchgeführt werden können. |
| ACBPM | ACBPM beschreibt neben einer Phase in der die Prozesselemente erstellt werden, eine Ausführungsphase und eine Terminierungsphase. Während der zuletzt genannten Phase können die Prozesselemente archiviert werden, um sie für eine neue Instanz wiederzuverwenden. ACBPM besitzt auch eine Monitoring-Phase, während der ein Prozessverlauf analysiert wird, um basierend auf den Ergebnissen den Prozess zu optimieren. |
| DPMS | DPMS beschreibt eine Modellierungsphase und eine Ausführungsphase. |
| DDPS | DDPS beschreibt eine Modellierungsphase und eine Ausführungsphase. |

Tabelle 7.3: Vergleich: Explizite Definition von Lebensphasen in einem Prozess

| Ansatz | Vergleich: Analyse der Prozesse |
|----------------|---|
| α -Flow | Für α -Flow wird keine Analyse eines Prozesses beschrieben. |
| BPMN | Ebenfalls für BPMN wird keine Analyse eines Prozesses beschrieben. |
| ACM | ACM bietet die Möglichkeit einen laufenden Prozess zu analysieren. Der Einsatz der Analyse soll bezüglich der Prozesseffizienz durchgeführt werden. Wie diese Analyse im Detail durchgeführt wird, ist nicht beschrieben. Eine automatische Anpassung wird nicht angeboten. Damit liegt die Anpassung eines Prozessverlaufes in der Verantwortung der Benutzer. |
| ACBPM | Bei ACBPM können Analyse durchgeführt werden. Eine Beschreibung, welche Analyseverfahren angewendet werden und wie sie sich auf den Prozessablauf auswirken, liegt nicht vor. |
| DPMS | Für DPMS wird keine Analyse eines Prozesses beschrieben. |
| DDPS | Ebenfalls für DDPS wird keine Analyse eines Prozesses beschrieben. |

Tabelle 7.4: Vergleich: Analyse der Prozesse

| Ansatz | Vergleich: Datenobjekt oder Rekord basierte Datenspeicherung |
|----------------|--|
| α -Flow | α -Flow speichert die Daten, die für einen Prozessinstanz benötigt werden, in einem aktiven Dokument. Das aktive Dokument kann als eine Ansammlung von Datenobjekten angesehen werden, die die Daten speichern. |
| BPMN | In BPMN werden Datenobjekte eingesetzt, die mittels Attributen die Daten speichern. Wie diese Datenobjekte physisch gespeichert werden, wird nicht beschrieben. |
| ACM | ACM speichert die Daten in einem DBS ab. |
| ACBPM | ACBPM beschreibt Datenobjekte, die die Daten speichern. Wie die Datenobjekte physisch gespeichert werden, wird nicht beschrieben. |
| DPMS | DPMS beschreibt Datenobjekte, die die Daten speichern. Wie die Datenobjekte physisch gespeichert werden, wird nicht beschrieben. |
| DDPS | DDPS beschreibt Datenobjekte, die die Daten speichern. Wie die Datenobjekte physisch gespeichert werden, wird nicht beschrieben. |

Tabelle 7.5: Vergleich: Datenobjekt oder Rekord basierte Datenspeicherung

| Ansatz | Vergleich: Zusammenführung von Nutzdaten, Verwaltungsdaten, Verwaltungssystem |
|----------------|--|
| α -Flow | In α -Flow wird zwischen den Nutz- und den Verwaltungsdaten unterschieden. Zu den Verwaltungssystemen zählen Koordinations-, Authentifizierungs-, Versionierungs- und Variantensystem, während zu den Nutzdaten im Wesentlichen die Daten gehören, die den Therapieverlauf beschreiben. Diese Informationen werden auf spezifischen α -Cards abgelegt, die die kleinste Einheit der Validierung, Freigabe, gemeinsamer ‚Sichtbarkeit‘ und der Verschlüsselungssignaturen darstellen. Alle α -Cards zusammen repräsentieren ein α -Doc, in dem alle Informationen zusammengeführt werden, und das die kleinste Einheit des Informationsaustausches repräsentiert. |
| BPMN | BPMN beschreibt nur den Prozessablauf und nicht die Verwaltungssysteme, die einen Prozess steuern. Die Steuerung obliegt dem WfMS, der das Modell umsetzt. |
| ACM | ACM Nutzdaten, Verwaltungsdaten und Verwaltungssystem werden im Back-End des ACM-Systems zusammengefasst. Ob die Nutzdaten und Verwaltungsdaten getrennt oder zusammen in einem DBS gespeichert werden, wird nicht beschrieben. Die Verwaltungssysteme, wie beispielweise Benutzerverwaltung, Ein- und Ausgabesystem oder Regelsystem für die Prozesssteuerung, werden auch im ACM-System integriert. |
| ACBPM | In ACBPM stellen Verwaltungssysteme eigenständige Systeme dar, die nicht in den Datenobjekten implementiert sind. Eine Beschreibung der Speicherung der Informationen für die Verwaltung eines Prozesses, wie beispielsweise die Benutzerrechte, wird nicht beschrieben. Allerdings kann der Verlauf eines Prozess je nach Ausprägung entweder durch den Sequenzfluss der Aktivitäten oder durch den Zustand der Attributwerte in den Datenobjekten beschrieben werden. |
| DPMS | DPMS beschreibt eine Verbindung der Verwaltungsdaten mit den Nutzdaten und Prozesselementen. Es besteht die Möglichkeit in der Benutzerverwaltung Verweise auf Daten oder Prozesselemente zu hinterlegen. Dies kann ein mögliches Indiz sein, dass die Nutzdaten, Verwaltungsdaten und das Verwaltungssystem zusammen implementiert werden. Der Verlauf eines Prozesses wird durch die Zustände in den Datenobjekten gelenkt. |
| DDPS | In DDPS wird als einziges Verwaltungssystem ein Regelsystem beschrieben. Das Regelsystem ist unabhängig von den Nutzdaten. Informationen, wie die Datenbasis für die Regeln gespeichert wird, werden nicht beschrieben |

Tabelle 7.6: Vergleich: Zusammenführung von Nutzdaten, Verwaltungsdaten, Verwaltungssystem

| Ansatz | Vergleich: Kommunikationsmittel für soziale Interaktion |
|----------------|---|
| α -Flow | Als ein Dokument basierter Ansatz ist es das Ziel von α -Flow alle verfügbaren Informationen für die anderen Anwender ‚sichtbar‘ zu machen, sobald der Autor die Daten (α -Cards) entsprechend markiert. Veränderungen an den Daten werden sofort aktualisiert und jeder Benutzer kann entsprechend seinen Rechten auf die Daten zugreifen. Dies ist auch über Systemgrenzen hinweg möglich. Des Weiteren erlaubt α -Flow die parallele Arbeit und organisiert die Zusammenführung der verteilten Informationen |
| BPMN | Der Informationsaustausch der Benutzer erfolgt nur auf der Basis der Daten, die zum Erreichen eines Geschäftszieles (Nutzdaten) benötigt werden. Deren Fluss ist von dem Sequenzfluss vorgeben. Daher findet eine Kommunikation zwischen Nutzern faktisch nicht statt. |
| ACM | ACM unterstützt die soziale Kommunikation durch das Verantwortungsprinzip. Alle durchgeführten Veränderungen sind verfolgbar. Des Weiteren können die Veränderungen mit Kommentaren versehen werden. Durch eine entsprechende Vorausplanung können beispielsweise bestimmte Personen in den Prozessablauf eingebunden oder ein Arbeitsschritt kann an einen spezifischen Anwender weitergeleitet werden. Die Benutzer leiten während der Ausführung alle Informationen, die ein anderer Anwender für die Ausführung seiner Aktivität benötigt, weiter. Zusätzlich sollen Kalender und Kommunikationsmittel (Chat-Räume, Instant Messenger (IM), ...) eingebunden werden, um den Austausch von Informationen zu fördern. |
| ACBPM | Eine Benutzerinteraktion erfolgt nur mittels der Daten für das Erreichen eines Geschäftszieles. Weitere Interaktionen zwischen Benutzern werden nicht beschrieben. |
| DPMS | Eine Benutzerinteraktion erfolgt nur mittels der Daten für das Erreichen eines Geschäftszieles. Weitere Interaktionen zwischen Benutzern werden nicht beschrieben. |
| DDPS | Entsprechenden Informationen liegen nicht vor. Es existiert nur eine unidirektionale Informationsaustauschrichtung über die Daten im Prozess. |

Tabelle 7.7: Vergleich: Kommunikationsmittel für soziale Interaktion

| Ansatz | Vergleich: Natürliche Sprache als Prozess-Beschreibungssprache |
|----------------|---|
| α -Flow | α -Flow Die Verwendung der natürlichen Sprache wird nicht unterstützt. |
| BPMN | BPMN Die Elemente für eine Ausführung müssen in Programmiersprache angegeben werden. |
| ACM | ACM verwendet als Definitionssprache für Elemente der Prozessmodellierung die natürliche Sprache. |
| ACBPM | ACBPM beschreibt die Prozess-Beschreibungssprache nicht. |
| DPMS | DPMS beschreibt die Prozess-Beschreibungssprache nicht. |
| DDPS | DDPS beschreibt die Prozess-Beschreibungssprache nicht. |

Tabelle 7.8: Vergleich: Natürliche Sprache als Prozess-Beschreibungssprache

| Ansatz | Vergleich: Anpassbare Benutzeroberfläche |
|----------------|---|
| α -Flow | α -Flow beschreibt nicht, ob die verwendete Benutzeroberfläche anpassbar ist. |
| BPMN | BPMN Die Benutzeroberfläche ist abhängig vom jeweiligen Modellierungswerkzeug. |
| ACM | ACM Die Benutzeroberfläche kann durch ziehen und setzen nach den Wünschen des Benutzers angepasst werden. |
| ACBPM | ACBPM beschreibt keine Benutzeroberfläche. |
| DPMS | DPMS beschreibt keine Benutzeroberfläche. |
| DDPS | DDPS beschreibt keine Benutzeroberfläche. |

Tabelle 7.9: Vergleich: Anpassbare Benutzeroberfläche

7.2.2 Funktionale Aspekte

In diesem Abschnitt wird der Vergleich anhand der Vergleichskriterien der funktionalen Aspekte in folgender Reihenfolge in Tabellen dargestellt:

- Benutzerinteraktion
- Ereignisbehandlung
- Online- / Offline- Modus
- Echtzeitfähigkeit von Anwendungsfunktionen
- Abhängigkeit von Anwendungsfunktion
- Granularität von Anwendungsfunktionen
- Wiederverwendbarkeit von Prozesselementen
- Langzeiteinsatzfähigkeit

| Ansatz | Vergleich: Benutzerinteraktion |
|----------------|---|
| α -Flow | Bei α -Flow interagieren die Benutzer mit Hilfe der Benutzeroberfläche, die in dem α -Doc integriert ist. Über die Benutzeroberfläche können die Benutzer Daten eingeben, Daten verändern und sich einen Überblick über alle α -Cards verschaffen. Für welche Operation ein Benutzer Rechte besitzt oder wie viel ein Benutzer sehen darf, wird über das Rechtmanagement bestimmt. |
| BPMN | Die Benutzerinteraktion wird durch spezifische Aktivitäten beschrieben. Diese Aktivitäten werden als „User Task“ bezeichnet. |
| ACM | Benutzer interagieren mittels einer anpassbaren Benutzeroberfläche. Sie können damit den Verlauf eines Prozesses steuern oder Daten für bestimmte Aktivitäten eingeben oder auslesen. Der Zugriff auf die Daten wird mittels eines Rechtmanagements kontrolliert. |
| ACBPM | Es wird unterschieden zwischen Aktivitäten, die mit und ohne Benutzerinteraktion auskommen können. Die vorliegenden Veröffentlichungen enthalten weder hierzu Informationen noch wird beschrieben, wie Benutzer mit dem System interagieren können. |
| DPMS | Es wird zwischen automatisierten und manuellen Aktivitäten unterschieden. Weder wie im Fall der manuellen Aktivität die Benutzerinteraktion aussehen kann, noch auf welche Weise sie stattfindet, wird nicht beschrieben. Veränderungen sind durch die Benutzerverwaltung reguliert. |
| DDPS | Eine Benutzerinteraktion wird von diesem Ansatz nicht beschrieben. |

Tabelle 7.10: Vergleich: Benutzerinteraktion

7 Vergleich der unterschiedlichen Ansätze zur Prozessunterstützung

| Ansatz | Vergleich: Ereignisbehandlung |
|----------------|--|
| α -Flow | Während der Ausführung von α -Flow können verschiedene Modifikationen an den Inhalten einer α -Card durchgeführt werden. Beispielsweise kann die Sichtbarkeit von privat auf öffentlich gesetzt werden. Hierbei muss den verteilten α -Cards diese Änderung mitgeteilt werden. Diese Operation wird als Synchronisation verstanden. Des Weiteren können Zustandsänderungen vom α -Doc aufgenommen und verarbeitet werden. Dies wird durch die aktiven Eigenschaften eines α -Docs realisiert. Auch Fehler werden durch das System behandelt. |
| BPMN | In BPMN können Teilprozesse definiert werden. Somit kann beispielsweise eine Aktion nach einer bestimmten zeitlichen Länge unterbrechen. Zudem ist es BPMN möglich, auf ein Ereignis zu reagieren und den Prozessverlauf entsprechend anzupassen. Ein Ereignis kann beispielsweise benutzt werden, um Aktionen zu beenden oder auf Informationen zu warten. Zusätzlich können Aktionen auch zurückgesetzt werden, wie beispielsweise ein Rollback in einem DBS. Allerdings müssen Ereignisse extra modelliert werden. Sind keine entsprechenden Symbole für ein auftretendes Ereignissen im Modell vorhanden, kann nicht auf anfallende Ereignisse reagiert werden. Welche Ereignisse behandelt werden können ist im Abschnitt C beschrieben |
| ACM | ACM unterscheidet zwischen externen und internen Ereignisse. Externe Ereignisse, wie Eintreffen von Nachrichten, beispielsweise E-Mails, werden an die Ein- und Ausgabeverwaltung verwiesen und die Daten werden importiert. Eine Aussage, wie beispielsweise der Ansatz auf auftretende Systemfehler reagiert, wird nicht beschrieben. Interne Ereignisse werden vom Regelsystem nach den vorabdefinierten Geschäftsregeln verarbeitet. Kann ein Ereignis nicht verarbeitet werden, wird dieses Ereignis an den Benutzer weitergeleitet, der für die Ausführung der Aktivität verantwortlich ist, die das Ereignis ausgelöst hat. |
| ACBPM | Ereignisse können mittels Assoziationen in einen Prozessablauf eingebettet werden. Dabei müssen die auftretenden Ereignisse vorab formuliert werden. Das Warten auf vorabdefinierte Ereignisse, wie das Eintreffen von Daten, kann modelliert werden. Alle unerwarteten Ereignisse können nicht automatisch behandelt werden. Die unerwarteten Ereignisse werden an die Benutzer weitergeleitet |
| DPMS | Für den Umgang von Ereignissen liegen keine Informationen vor. |
| DDPS | Auf Ereignisse kann automatisch eingegangen werden, wenn sie mittels Lebenszyklus modelliert worden sind. Wenn dies nicht der Fall ist, kann nur der Benutzer den Ablauf entsprechend modifizieren. Wie die Ereignisse dem Benutzer dargestellt werden, wird nicht beschrieben. |

Tabelle 7.11: Vergleich: Ereignisbehandlung

| Ansatz | Vergleich: Online- / Offline- Modus |
|----------------|--|
| α -Flow | Informationen über einen Online-/Offline-Modus liegen nicht vor. Allerdings ist eine vom Kommunikationsnetz getrennte Ausführung des Ansatzes möglich. Ein Informationsaustausch ist in diesem Fall nicht möglich. |
| BPMN | Informationen über einen Online-/Offline-Modus liegen nicht vor. Auch eine Trennung von einzelnen Informationssystemen, die im Prozessablauf eingeplant sind, wird nicht beschrieben. |
| ACM | Eine Online- / Offline- Modus wird explizit unterstützt. Dabei wird sowohl die bewusste Trennung vom Netz beschrieben, als auch die spontane Trennung (Fehlerfall). Eine Umsetzung dieses Systems wird allerdings nicht beschrieben. |
| ACBPM | Informationen über einen Online-/Offline-Modus liegen nicht vor. Auch eine Trennung von einzelnen Informationssystemen, die im Prozessablauf eingeplant sind, wird nicht beschrieben. |
| DPMS | Informationen über einen Online-/Offline-Modus liegen nicht vor. Auch eine Trennung von einzelnen Informationssystemen, die im Prozessablauf eingeplant sind, wird nicht beschrieben. |
| DDPS | Informationen über einen Online-/Offline-Modus liegen nicht vor. Auch eine Trennung von einzelnen Informationssystemen, die im Prozessablauf eingeplant sind, wird nicht beschrieben. |

Tabelle 7.12: Vergleich: Online- / Offline- Modus

| Ansatz | Vergleich: Echtzeitfähigkeit von Anwendungsfunktionen |
|----------------|---|
| α -Flow | In α -Flow existieren keine Anwendungsfunktionen. |
| BPMN | Über die Echtzeitfähigkeit, dieses Ansatzes können keine Aussagen gemacht werden, da die Echtzeitfähigkeit vom ausführbaren WfMS abhängen, was im Zuge dieser Arbeit nicht betrachtet wurde. |
| ACM | Über die Echtzeitfähigkeit liegen keine Informationen vor. Da ein zentrales Back-End existiert und jeder Client per öffentliches Netz darauf zugreift, kann eine Echtzeitfähigkeit nicht garantiert werden. |
| ACBPM | Über die Echtzeitfähigkeit liegen keine Informationen vor. |
| DPMS | Über die Echtzeitfähigkeit liegen keine Informationen vor. |
| DDPS | Über die Echtzeitfähigkeit liegen keine Informationen vor. |

Tabelle 7.13: Vergleich: Echtzeitfähigkeit von Anwendungsfunktionen

| Ansatz | Vergleich: Abhängigkeit von Anwendungsfunktion |
|----------------|--|
| α -Flow | Es werden keine Anwendungsfunktionen implementiert, die auf die Daten von extern zugreifen. |
| BPMN | Eine direkte Abhängigkeit von Anwendungsfunktion untereinander wird nicht beschrieben. Nur durch die Vorgabe der Geschäftsziele, die am Ende ein fertiges Produkt ergeben sollen, ist eine definierte Reihenfolge von Anwendungsfunktionen vorgeben, weil im Regelfall nicht alle Schritte in einem Prozessablauf parallel ausgeführt werden können. Es existiert eine Hierarchie von Anwendungsfunktionen. Eine Anwendungsfunktion kann als ein untergeordneter Prozess dargestellt werden, der eigene Anwendungsfunktionen beinhaltet. Dabei ist die übergeordnete Anwendungsfunktion abhängig von den untergeordneten Anwendungsfunktionen. |
| ACM | Über die Abhängigkeit von Anwendungsfunktionen liegen keine Informationen vor. Nur durch die Vorgabe der Geschäftsziele, die am Ende ein fertiges Produkt ergeben sollen, ist eine definierte Reihenfolge von Anwendungsfunktionen vorgeben, weil im Regelfall nicht alle Schritte in einem Prozessablauf parallel ausgeführt werden können. |
| ACBPM | Über die Abhängigkeit von Anwendungsfunktionen liegen keine Informationen vor. Nur durch die Vorgabe der Geschäftsziele, die am Ende ein fertiges Produkt ergeben sollen, ist eine definierte Reihenfolge von Anwendungsfunktionen vorgeben, weil im Regelfall nicht alle Schritte in einem Prozessablauf parallel ausgeführt werden können. Die Anwendungsfunktionen werden durch die Zustände in den Datenobjekten gesteuert und die Zustandsübergänge werden durch den Lebenszyklus definiert. Somit kann nicht von einer direkten Abhängigkeit der Anwendungsfunktionen untereinander gesprochen werden, sondern von der Abhängigkeit der einzelnen Datenzustände. |
| DPMS | Über die Abhängigkeit von Anwendungsfunktionen liegen keine Informationen vor. Nur durch die Vorgabe der Geschäftsziele, die am Ende ein fertiges Produkt ergeben sollen, ist eine definierte Reihenfolge von Anwendungsfunktionen vorgeben, weil im Regelfall nicht alle Schritte in einem Prozessablauf parallel ausgeführt werden können. Die Anwendungsfunktionen werden durch die Zustände in den Datenobjekten gesteuert und die Zustandsübergänge werden durch den Lebenszyklus definiert. Somit kann nicht von einer direkten Abhängigkeit der Anwendungsfunktionen untereinander gesprochen werden, sondern von der Abhängigkeit der einzelnen Datenzustände. |
| DDPS | Über die Abhängigkeit von Anwendungsfunktionen liegen keine Informationen vor. Nur durch die Vorgabe der Geschäftsziele, die am Ende ein fertiges Produkt ergeben sollen, ist eine definierte Reihenfolge von Anwendungsfunktionen vorgeben, weil im Regelfall nicht alle Schritte in einem Prozessablauf parallel ausgeführt werden können. Die Anwendungsfunktionen werden durch die Zustände in den Datenobjekten gesteuert und die Zustandsübergänge werden durch den Lebenszyklus definiert. Somit kann nicht von einer direkten Abhängigkeit der Anwendungsfunktionen untereinander gesprochen werden, sondern von der Abhängigkeit der einzelnen Datenzustände. |

Tabelle 7.14: Vergleich: Abhängigkeit von Anwendungsfunktion

| Ansatz | Vergleich: Granularität von Anwendungsfunktionen |
|----------------|---|
| α -Flow | Es gibt keine Anwendungsfunktionen im α -Flow. |
| BPMN | Eine Aktivität kann durch einen Subprozess beschrieben werden. Dabei besitzt jeder Subprozess wieder Aktivitäten. Durch die Unterordnung von Aktivitäten kann eine Hierarchie von Aktivitäten aufgebaut werden, die beliebig breit als auch tief sein kann. Die Aktivitäten werden dann durch die Anwendungsfunktionen bei der Implementierung dargestellt. |
| ACM | Über die Granularität der Anwendungsfunktionen liegen keine Informationen vor. |
| ACBPM | Eine Anwendungsfunktion kann mehrere Aktivitäten besitzen. Eine eigene Prozessstruktur innerhalb einer Anwendungsfunktion wird nicht beschrieben. |
| DPMS | Die Anwendungsfunktionsstruktur ist an die Struktur der Datenobjekte gekoppelt. Werden die Datenobjekte beispielsweise als Baumhierarchie beschrieben, dann können für jede Ebene in der Hierarchie Anwendungsfunktionen existieren. |
| DDPS | Eine Anwendungsfunktion kann mehrere Aktivitäten besitzen. Eine eigene Prozessstruktur innerhalb einer Anwendungsfunktion wird nicht beschrieben. |

Tabelle 7.15: Vergleich: Granularität von Anwendungsfunktionen

| Ansatz | Vergleich: Wiederverwendbarkeit von Prozesselementen |
|----------------|---|
| α -Flow | Aus jedem α -Doc lässt sich eine weitere α -Doc erzeugen. So werden die Elemente eines Prozess auf eine neue Instanz per Kopie übertragen. |
| BPMN | Die Symbole in BPMN können immer wieder für die Modellierung verwendet werden. Sie liegen generalisiert vor und müssen für den Einsatzzweck durch Benutzer spezialisiert werden. |
| ACM | Nach der Modellierung der Prozesselemente werden alle Prozesselemente der Prozessinstanz archiviert. Bei einer neuen Prozessinstanz können die Prozesselemente direkt oder angepasst wieder verwendet werden. |
| ACBPM | Bereits verwendete Prozesselemente können als Vorlage für neue Prozesselemente dienen. Allerdings wird eine automatische Generierung von neuen Prozesselementen aus alten nicht unterstützt. |
| DPMS | Über die Wiederverwendbarkeit von Prozesselementen liegen keine Informationen vor. |
| DDPS | Bereits verwendete Prozesselemente können als Vorlage für neue Prozesselemente dienen. Allerdings wird eine automatische Generierung von neuen Prozesselementen aus alten nicht unterstützt. |

Tabelle 7.16: Vergleich: Wiederverwendbarkeit von Prozesselementen

| Ansatz | Vergleich: Langzeiteinsatzfähigkeit |
|----------------|--|
| α -Flow | Bezüglich der Langzeitausführungsfähigkeit gibt es keine Beschränkung. |
| BPMN | Darüber liegen keine Informationen vor. |
| ACM | Darüber liegen keine Informationen vor. |
| ACBPM | Darüber liegen keine Informationen vor. |
| DPMS | Darüber liegen keine Informationen vor. |
| DDPS | Darüber liegen keine Informationen vor. |

Tabelle 7.17: Vergleich: Langzeiteinsatzfähigkeit

7.2.3 Informative Aspekte

In diesem Abschnitt wird der Vergleich anhand der Vergleichskriterien der informativen Aspekte in folgender Reihenfolge in Tabellen dargestellt:

- Aktive Datenobjekte
- Inhalt: Nutz- und Verwaltungsdaten
- Identifikation von Daten
- Speicherung der Prozessverlaufsinformationen in den Daten
- Anzahl der Datenobjekte für einen Anwendungsfall
- Einheit der Validierung

| Ansatz | Vergleich: Aktive Datenobjekte |
|----------------|--|
| α -Flow | Die Voraussetzung für ein aktives Datendokument in α -Flow ist, dass zumindest eine α -Card aktive Eigenschaften enthält. Basierend auf diesen aktiven Eigenschaften wird der Prozessfortschritt gesteuert, worunter der erfolgreiche Abschluss einer angeforderten inhaltsbasierten α -Card verstanden wird. Aktive Eigenschaften dürfen nicht als direkte Aktivitäten verstanden werden, sondern sie unterstützen den Austausch von Datendokumenten und kreieren auf diese Weise implizit Aktivitäten derjenigen Organisationen, an die sie weitergeleitet wurden. Für die Anzeige der Daten stellt α -Doc eine eigene Benutzeroberfläche zur Verfügung. In dem Ansatz werden keine Schnittstellen zu weiteren Anwendungsfunktionen beschrieben. |
| BPMN | In BPMN sind die Datenobjekte passiv. Eine physische Abbildung der Datenobjekte liegt in der Verantwortung der Anwender |
| ACM | Verwendet keine Datenobjekte. |
| ACBPM | Das Datenobjekt ist passiv. Über die Struktur der Speicherung liegen keine Informationen vor. |
| DPMS | Die Datenobjekte sind keine aktiven Datenobjekte, denn von ihnen gehen keine direkten Aktionen aus. Aktivitäten sind nur durch Anwendungsfunktionen auf ihn möglich. Wie die Struktur, beispielsweise eine XML-Datei, aufgebaut ist, wird nicht beschrieben. |
| DDPS | Die Datenobjekte sind keine aktiven Datenobjekte, da von ihnen selbst keine Aktionen ausgehen. Aktivitäten sind nur durch Anwendungsfunktionen auf ihnen möglich. Wie die Struktur, beispielsweise eine XML-Datei, aufgebaut ist, wird nicht beschrieben. |

Tabelle 7.18: Vergleich: Aktive Datenobjekte

7 Vergleich der unterschiedlichen Ansätze zur Prozessunterstützung

| Ansatz | Vergleich: Inhalt: Nutz- und Verwaltungsdaten |
|----------------|--|
| α -Flow | Im α -Doc werden neben den Nutzinformationen auch die Verwaltungsinformationen gespeichert. Als Verwaltungsinformationen werden im α -Flow-Ansatz Informationen wie beispielsweise die Prozessteilnehmer, ihre Institutsbezogenen Informationen, das Behandlungsmodell, Topologie des Netzwerkes usw. gespeichert (CRA α -Card). |
| BPMN | In den Datenobjekten sind nur Nutzdaten gespeichert. Gateways können mittels Datenobjekten Entscheidungen treffen. Datenobjekte können den Sequenzfluss nicht aktiv steuern. |
| ACM | Hierrüber wird keine Aussage gemacht. Annahme: Beschrieben wird ein Back-End von dem die Informationen für die Verwaltungssysteme als auch für die Nutzdaten übertragen werden. Ein ACM-System kontrolliert mittels eines Regelsystems den kompletten Prozessablauf, wozu auch die Überwachung geänderter Daten gehört. Somit müssten auch die Informationen für das Regelsystem mitversendet werden. Ob im ACM-System die Daten auf unterschiedlichen DBS gespeichert werden, wird nicht beschrieben. |
| ACBPM | Datenobjekte beinhalten sowohl die Nutzdaten, als auch Verwaltungsdaten. Zu den Verwaltungsdaten zählen die Verlaufsdaten und die Information, welche Anwendungsfunktion als nächstes ausgeführt werden soll. Der Zustand des Datenobjektes wird mittels eines Attributs definiert. Zusätzlich verfügt das Datenobjekt über ein Attribut, welches seine ID beinhaltet. Wo weitere Verwaltungsdaten, wie Informationen über die Benutzerverwaltung oder die Protokollierungsdateien gespeichert werden, wird nicht beschrieben. |
| DPMS | Die Datenobjekte enthalten sowohl Nutz- als auch Verwaltungsdaten. Zu den Verwaltungsdaten zählen nur die Verlaufsdaten für den Prozess. Der Speicherort weiterer Verwaltungsdaten wird nicht beschrieben. Allerdings werden Verweise von Datenobjekten auf die Benutzerkonten beschrieben. Dies lässt vermuten, dass die Daten verbunden sind. Die Speicherung der Daten auf einem DBS kann nicht ausgeschlossen werden. Zusätzlich wird im Datenobjekt eine ID gespeichert für die eindeutige Identifikation vom Prozess. |
| DDPS | Die Datenobjekte enthalten nur Nutzinformationen. Die Nutzinformationen besitzen immer einen definierten Zustand. Anhand dieses Zustandes arbeiten die Anwendungsfunktionen auf dem Datenobjekt. Die Nutzinformationen stellen die Bedingungen für die Ausführung bereit. Somit kann der Zustand als Verwaltungsinformation deklariert werden. |

Tabelle 7.19: Vergleich: Inhalt: Nutz- und Verwaltungsdaten

| Ansatz | Vergleich: Identifikation von Daten |
|----------------|--|
| α -Flow | Ein α -Doc wird durch eine α -Episoden-ID eindeutig identifizierbar. Zusätzlich wird jeder α -Card eine eindeutige Card-ID. Somit kann jede Information im Prozessverlauf identifiziert werden. |
| BPMN | Datenobjekte erhalten nicht automatisch eine ID. Ein Benutzer kann diese aber einfügen. |
| ACM | Hierüber liegen keine genauen Informationen vor. Durch den Einsatz eines DBSs, kann davon ausgegangen werden, dass ein Tupel durch einen Primärschlüssel identifiziert wird. |
| ACBPM | Ein Datenobjekt besitzt eine ID, womit die Datenobjekte identifiziert werden können. |
| DPMS | Die Datenobjekte werden mit einer ID ausgestattet, die eindeutig in einer Prozessinstanz ist. DPMS geht vermutlich von einer DBS aus, bei dem jedes Datenobjekt ein Tupel ist und somit durch den Primärschlüssel identifizierbar ist. |
| DDPS | Hierüber liegen keine genauen Informationen vor. |

Tabelle 7.20: Vergleich: Identifikation von Daten

| Ansatz | Vergleich: Speicherung der Prozessverlaufsdaten in den Daten |
|----------------|---|
| α -Flow | Speichert den Prozessverlauf in der TSA α -Card ab. |
| BPMN | Der Verlauf wird mittels des Sequenzflusses beschrieben. Die Daten können dadurch keinen Verlauf speichern. |
| ACM | Erfüllt die Variante 2, die in den Vergleichskriterien beschrieben wurde. |
| ACBPM | Erfüllt die Variante 2, die in den Vergleichskriterien beschrieben wurde. |
| DPMS | Erfüllt die Variante 2, die in den Vergleichskriterien beschrieben wurde. |
| DDPS | Erfüllt die Variante 2, die in den Vergleichskriterien beschrieben wurde. |

Tabelle 7.21: Vergleich: Speicherung der Prozessverlaufsdaten in den Daten

| Ansatz | Vergleich: Anzahl der Datenobjekte für einen Anwendungsfall |
|----------------|--|
| α -Flow | Es existiert immer nur ein Dokument (α -Doc), das während des Ablaufs alle Informationen beinhaltet. Es existieren allerdings Replikate des (α -Doc) bei den Anwendern einem Prozessverlauf. |
| BPMN | Es können beliebig viele Datendokumente in einem Prozessverlauf existieren. Dies hängt davon ab, wie die Daten modelliert werden. Entweder in einem Datenobjekt oder in mehreren. |
| ACM | Besitzt keine Datenobjekte. Es wird allerdings eine Analyse der Geschäftsentitäten durchgeführt. Dies lässt vermuten, dass im Prozessverlauf mehrere Entitäten verwendet werden können. |
| ACBPM | In einer Prozessinstanz können mehrere Datenobjekte existieren. Dies ist abhängig von der Anzahl an einem Prozess beteiligten Entitäten. Pro Entität wird ein Datenobjekt angelegt. |
| DPMS | In einer Prozessinstanz können mehrere Datenobjekte existieren. Dies ist abhängig von der Anzahl an einem Prozess beteiligten Entitäten. Pro Entität wird ein Datenobjekt angelegt. |
| DDPS | Es können mehrere Datenobjekte beteiligt sein. Welche Informationen ein Datenobjekt trägt hängt von dem Entwickler der Datenobjekte ab. |

Tabelle 7.22: Vergleich: Anzahl der Datenobjekte für einen Anwendungsfall

| Ansatz | Vergleich: Einheit der Validierung |
|----------------|---|
| α -Flow | Eine α -Card ist die Einheit der Validierung. Sie kann gültig sein oder nicht. |
| BPMN | Zur Einheit der Validierung liegen keine Informationen vor. Die Datenobjekte stehen bei BPMN nicht im Vordergrund, aber es ist davon auszugehen, dass ein Datenobjekt gültig ist, wenn eine Anwendungsfunktion die Bearbeitung abgeschlossen hat. |
| ACM | Zur Einheit der Validierung liegen keine Informationen vor. |
| ACBPM | Zur Einheit der Validierung liegen keine Informationen vor. |
| DPMS | Zur Einheit der Validierung liegen keine Informationen vor. |
| DDPS | Zur Einheit der Validierung liegen keine Informationen vor. |

Tabelle 7.23: Vergleich: Einheit der Validierung

7.2.4 Dynamische Aspekte

In diesem Abschnitt wird der Vergleich anhand der Vergleichskriterien der dynamischen Aspekte in folgender Reihenfolge in Tabellen dargestellt:

- Daten- oder Aktivitätssteuerung
- Datenlebenszyklus
- Offenes Ende oder Terminierung einer Prozessinstanz
- Echtzeitfähigkeit von Prozessen

| Ansatz | Vergleich: Daten- oder Aktivitätssteuerung |
|----------------|--|
| α -Flow | In den PSA (Process Structure Artifact) α -Card ist das gesamte dynamische Prozessschema in der Form von α -Card Platzhaltern und der Beziehungen zwischen ihnen abgelegt. Somit ist in den PSA der Kontrollfluss eines α -Flow-Prozesses explizit festgelegt, während durch das Anlegen einer neuen α -Card implizit eine Aktivität angefragt wird, die durch den erfolgreichen Abschluss einer (inhaltsbasierten) α -Card abgeschlossen wird. Auf diese Art löst α -Flow die Dualität zwischen inhaltsbasiertem und aktivitätsbasiertem Paradigma. |
| BPMN | Durch den Sequenzfluss von Aktivitäten wird der Verlauf eines Prozesses gesteuert. Für die Ausführung des Prozessverlaufes spielen Daten keine Rolle. |
| ACM | Sowohl Daten als auch das Regelsystem für den Verlauf eines Prozesses steuern den Prozessverlauf. Das Regelsystem plant die Aktivitäten in einen Verlauf ein. Die Reihenfolge wird von dem zu erreichendem Geschäftsziel bestimmt. Der Prozessverlauf passt sich während des Verlaufes an die von den Benutzern eingegebenen Daten an. |
| ACBPM | Das Datenobjekt beschreibt anhand der Zustände der Daten in dem Datenobjekte den Prozessverlauf. Anwendungsfunktionen lesen die Konfiguration der Werte der Attribute aus und erkennen an diesen, ob sie auf dem Datenobjekte arbeiten dürfen und welche Funktionen sie ausführen sollen. |
| DPMS | Das Datenobjekt beschreibt anhand der Zustände der Daten in dem Datenobjekte den Prozessverlauf. Anwendungsfunktionen lesen die Konfiguration der Werte der Attribute aus und erkennen an diesen, ob sie auf dem Datenobjekte arbeiten dürfen und welche Funktionen sie ausführen sollen. Die Aktivitäten werden anhand der Datenobjekte und deren Attributstruktur entwickelt. |
| DDPS | Die Steuerung eines Prozesses wird mittels den Datenobjekten und deren Zuständen definiert. Zusätzlich werden Prozessschritte vorab definiert, die ein Datenobjekt erreichen muss. Die Prozessschritte werden den Datenzuständen zugeordnet und ergeben damit einen lauffähigen Prozess. Die Zustandsänderungen werden mittels Anwendungsfunktionen durchgeführt. Somit existieren hier eine Anwendungs- und eine Datensteuerung. |

Tabelle 7.24: Vergleich: Daten- oder Aktivitätssteuerung

| Ansatz | Vergleich: Datenlebenszyklus |
|----------------|---|
| α -Flow | Der Fortschritt des Prozesses definiert sich in α -Flow über die Änderung einer Ansammlung von α -Cards und wird als α -Episode bezeichnet. Das Ende einer α -Episode ist dann erreicht, wenn keine neuen Informationen benötigt werden, um ein bestimmtes Ziel zu erreichen. |
| BPMN | Die Daten sind für die Ausführung nicht relevant und besitzen damit keinen Lebenszyklus. Ein Lebenszyklus kann alternative über die einzelnen Aktivitäten im Sequenzfluss definiert werden. |
| ACM | Information über einen Lebenszyklus liegen nicht vor. |
| ACBPM | Der Lebenszyklus beschreibt von der Instanziierung bis zum Ende eines Datenobjektes die Zustände, die ein Datenobjekt annehmen kann. Dabei ist ein Zustand eine Konfiguration der Attributwerte in einem Datenobjekt. Ein Datenobjekt geht von einem Zustand in den nächsten über. Die Übergänge der Zustände werden durch Anwendungsfunktionen realisiert, die auf die Attribute eines Datenobjektes zugreifen und die Daten ändern. |
| DPMS | Der Lebenszyklus beschreibt von der Instanziierung bis zum Ende eines Datenobjektes die Zustände, die ein Datenobjekt annehmen kann. Dabei ist ein Zustand eine Konfiguration der Attributwerte in einem Datenobjekt. Ein Datenobjekt geht von einem Zustand in den nächsten über. Die Übergänge der Zustände werden durch Anwendungsfunktionen realisiert, die auf die Attribute eines Datenobjektes zugreifen und die Daten ändern. |
| DDPS | Der Lebenszyklus beschreibt von der Instanziierung bis zum Ende eines Datenobjektes die Zustände, die ein Datenobjekt annehmen kann. Dabei ist ein Zustand eine Konfiguration der Attributwerte in einem Datenobjekt. Ein Datenobjekt geht von einem Zustand in den nächsten über. Die Übergänge der Zustände werden durch Anwendungsfunktionen realisiert, die auf die Attribute eines Datenobjektes zugreifen und die Daten ändern. |

Tabelle 7.25: Vergleich: Datenlebenszyklus

| Ansatz | Vergleich:Offenes Ende oder Terminierung einer Prozessinstanz |
|----------------|---|
| α -Flow | Das Ende einer α -Episode ist dann erreicht, wenn keine neuen Informationen benötigt werden, um ein bestimmtes Ziel zu erreichen. |
| BPMN | Der Sequenzfluss terminiert nach der letzten Aktivität, die im Sequenzfluss eingeplant wurde. |
| ACM | Ein fest definiertes Ende wird nicht beschrieben. Das Ende wird von den Benutzern definiert, wenn das Geschäftsziel erreicht ist. Den Prozessverlauf können die Anwender immer an jeder Stelle des Prozesses erweitern. |
| ACBPM | Der Lebenszyklus schreibt das Ende des Prozesses vor. Daher wird beim Erreichen definierter Datenzustände der Prozess terminiert. Der Lebenszyklus kann zwar erweitert werden, der Prozesslauf wird dann entsprechend terminieren. |
| DPMS | Der Lebenszyklus schreibt das Ende des Prozesses vor. Daher wird beim Erreichen definierter Datenzustände der Prozess terminiert. Der Lebenszyklus kann zwar erweitert werden, der Prozesslauf wird dann entsprechend terminieren. |
| DDPS | Zusätzlich zu der Planung der Zustände der Datenobjekte werden, werden auch die Prozesszustände geplant. Somit muss vorab auch ein Ende des Prozesses eingeplant werden. Der Lebenszyklus kann zwar erweitert werden, der Prozesslauf wird dann entsprechend terminieren. |

Tabelle 7.26: Vergleich:Offenes Ende oder Terminierung einer Prozessinstanz

| Ansatz | Vergleich: Echtzeitfähigkeit von Prozessen |
|----------------|--|
| α -Flow | Da jeder Anwender auf ein Replikat der α -Docs zugreifen kann, ist jederzeit der direkte Zugriff möglich. Wegen der logischen Zentralisierung sind die Anwendungen (beispielsweise die Nachverfolgung von Modifikationen), die über ein Netzwerk ablaufen, von der aktuellen Auslastung des Netzes abhängig. Allgemein können alle Anwendungen, die Zugriffe über ein Netzwerk erlauben, die Ausführung von Aktivitäten in Echtzeit nicht garantieren, da diese von der Auslastung des Netzes und dem möglichen Ausfall eines Knotens abhängen. |
| BPMN | Die Echtzeitfähigkeit hängt von den eingesetzten WfMS ab. Da dieses im Zuge dieser Arbeit nicht betrachtet wurde, kann hierüber keine Aussage gemacht werden. |
| ACM | Eine genaue Aussage über die Echtzeitfähigkeit wird nicht gemacht. Annahme: Für die Verwaltung existiert ein zentraler Server auf dem das ACM-System läuft. Durch viele Clients und durch ein vermutlich öffentliches Netzwerk kann keine Echtzeitfähigkeit garantiert werden. |
| ACBPM | Es liegen keine Informationen über die Echtzeitfähigkeit vor. |
| DPMS | Es liegen keine Informationen über die Echtzeitfähigkeit vor. |
| DDPS | Es liegen keine Informationen über die Echtzeitfähigkeit vor. |

Tabelle 7.27: Vergleich: Echtzeitfähigkeit von Prozessen

7.2.5 Integrative Aspekte

In diesem Abschnitt wird der Vergleich anhand der Vergleichskriterien der integrativen Aspekte in folgender Reihenfolge in Tabellen dargestellt:

- Transparenz
- Inter-institutionelle Prozessausführung

| Ansatz | Vergleich: Transparenz |
|----------------|---|
| α -Flow | Während der Abarbeitung eines Prozesses kann jeder Prozessteilnehmer alle Informationen über einen Fall einsehen, die für seine Rolle freigegeben sind. Durch die Definition der Rollen in den α -Cards und den Veränderungseinträgen in der Log- Datei können alle Modifikationen während des Verlaufes nachvollzogen werden. |
| BPMN | Es liegen keine Informationen über die Transparenz vor. |
| ACM | In ACM sollen sowohl die Kunden und Geschäftspartner als auch die Mitarbeiter in den Prozessablauf integriert werden. Sie können den Prozessablauf einsehen und abhängig von ihrer Berechtigung auch beeinflussen. Für die Nachverfolgung von Entscheidungen können die Log-Informationen eingesehen werden. |
| ACBPM | Es liegen keine Informationen über die Transparenz vor. |
| DPMS | Ein Prozessteilnehmer kann die Daten, die er für einen Prozessverlauf benötigt, abhängig von seinen Benutzerrechten einsehen. Dabei werden die Grenzen der Datenobjekte aufgehoben. |
| DDPS | Es liegen keine Informationen über die Transparenz vor. |

Tabelle 7.28: Vergleich: Transparenz

| Ansatz | Vergleich:Inter-institutionelle Prozessausführung |
|----------------|--|
| α -Flow | Eine inter-institutionelle Prozessausführung wird durch lose-gekoppelte Systeme unterstützt. α -Docs können räumlich getrennt auf unterschiedlichen Informationssystemen ausgeführt werden. Die Kommunikation kann beispielsweise über das Internet erfolgen. |
| BPMN | Eine inter-institutionelle Prozessausführung wird unterstützt. Allerdings würde die inter-institutionelle Prozessausführung über ein WfMS durchgeführt werden. Somit liegen keine Informationen über die Ausführung vor. |
| ACM | Es liegen keine Informationen über inter-institutionelle Prozessausführung vor. |
| ACBPM | Es liegen keine Informationen über inter-institutionelle Prozessausführung vor. |
| DPMS | Es liegen keine Informationen über inter-institutionelle Prozessausführung vor. |
| DDPS | Es liegen keine Informationen über inter-institutionelle Prozessausführung vor. |

Tabelle 7.29: Vergleich:Inter-institutionelle Prozessausführung

7.2.6 Technischoperative Aspekte

In diesem Abschnitt wird der Vergleich anhand der Vergleichskriterien der technisch operativen Aspekte in folgender Reihenfolge in Tabellen dargestellt:

- Kommunikationsnetz
- Spezialgeräte (Pads, Mobiltelefone)
- Kenntnis der Systemarchitektur

| Ansatz | Vergleich: Kommunikationsnetz |
|----------------|---|
| α -Flow | Über das Kommunikationsnetz liegen keine Informationen vor. |
| BPMN | Über das Kommunikationsnetz liegen keine Informationen vor. |
| ACM | Über das Kommunikationsnetz liegen keine Informationen vor. |
| ACBPM | Über das Kommunikationsnetz liegen keine Informationen vor. |
| DPMS | Über das Kommunikationsnetz liegen keine Informationen vor. |
| DDPS | Über das Kommunikationsnetz liegen keine Informationen vor. |

Tabelle 7.30: Vergleich: Kommunikationsnetz

| Ansatz | Vergleich: Spezialgeräte (Pads, Mobiltelefone) |
|----------------|---|
| α -Flow | Über Spezialgeräte liegen keine Informationen vor. |
| BPMN | Über Spezialgeräte liegen keine Informationen vor. |
| ACM | In den Prozessverlauf können Spezialgeräte eingebunden werden. Zu den diesen gehören Pads, Mobiltelefon und PDAs. Wie die Einbindung durchgeführt wird, wird im Detail nicht beschrieben. |
| ACBPM | Über Spezialgeräte liegen keine Informationen vor. |
| DPMS | Über Spezialgeräte liegen keine Informationen vor. |
| DDPS | Über Spezialgeräte liegen keine Informationen vor. |

Tabelle 7.31: Vergleich: Spezialgeräte (Pads, Mobiltelefone)

| Ansatz | Vergleich: Kenntnis der Systemarchitektur |
|----------------|--|
| α -Flow | Ein Anwender muss keine Informationen über den Aufbau des Systems haben, um einen Prozessablauf durchzuführen. |
| BPMN | Informationen zur Systemarchitektur sind von dem verwendeten WfMS abhängig. |
| ACM | Ein Anwender soll das System bedienen können ohne Informationen über den Aufbau zu haben. |
| ACBPM | Ob Kenntnis über die Systemarchitektur benötigt werden, um einen Prozess zu beschreiben, wird nicht erwähnt. |
| DPMS | Ob Kenntnis über die Systemarchitektur benötigt werden, um einen Prozess zu beschreiben, wird nicht erwähnt. |
| DDPS | Ob Kenntnis über die Systemarchitektur benötigt werden, um einen Prozess zu beschreiben, wird nicht erwähnt. |

Tabelle 7.32: Vergleich: Kenntnis der Systemarchitektur

7.2.7 Softwareoperative Aspekte

In diesem Abschnitt wird der Vergleich anhand der Vergleichskriterien der software operativen Aspekte in folgender Reihenfolge in Tabellen dargestellt:

- Regelsystem
- Sicherheitssysteme
- Benutzerverwaltung
- Protokollierungssystem
- Ein- und Ausgabeverwaltung
- Anmerkungs-system
- Lokalisierung von Anwendungsfunktionen
- Koordinierung von Daten
- Versionsverwaltung
- Fehlererkennung
- Fehlerkorrektur
- Verklemmungsvermeidungssystem

| Ansatz | Vergleich: Regelsystem |
|----------------|---|
| α -Flow | In α -Flow müssen zwei Fälle unterschieden werden. a.) Die Entscheidung über den Prozessverlauf liegt komplett bei den Anwendern. b.) Es existiert ein Regelsystem, das die aktiven Eigenschaften einer α -Card überwacht. Wird von einem Benutzer eine Eigenschaft einer α -Card verändert, beispielsweise die Sichtbarkeit wird auf öffentlich gesetzt, löst dies eine Synchronisation der verteilten α -Docs aus. Das Regelsystem muss drei Gesichtspunkte unterstützen: 1. Aktivierungsphase: unter welchen Bedingung eine Regelausgelöst wird. 2. Ordnungsphase: es können mehrere Regeln für eine Änderung einer Eigenschaft aufgerufen werden. 3. Evaluationsphase: Überprüfung der Randbedingungen, Ausführung und Abschluss (Anzeige und eventuell Fehlerfallbehandlung) |
| BPMN | Ein Regelsystem, in dem Regeln von Benutzer angelegt werden können, wird nicht beschrieben. Es existieren Abbildungsregeln, wie beispielsweise eine BPMN-Grafik umgesetzt werden muss. Zusätzlich können in Programmiersprache ähnlicher Schreibweise Bedingung an die Aktionen oder Gateways ¹ definiert werden, die dann vom Übersetzer in einen ausführbaren Code umgesetzt werden. Ein Regelsystem, das die Ausführung kontrolliert, existiert nicht. Ein Regelsystem kann aber in dem Workflow-Managementssystem realisiert sein. |
| ACM | In ACM kontrolliert das Regelsystem die Ausführung des Prozesses. Das Regelsystem implementiert die Bedingungen, nach denen ein Prozess ablaufen soll. Alle Modifikationen des Prozessverlaufes müssen mit den definierten Regeln übereinstimmen. Ein Regelsystem kann in ACM nach Forward Chaining oder Backward Chaining aufgebaut sein. |
| ACBPM | Ein Regelsystem wird eingesetzt, um die Assoziationen dieses Ansatzes zu beschreiben. Beim Eintritt eines Ereignisses reagiert das Regelsystem und bearbeitet dieses entsprechend der implementierten Regeln. Auch die Zuordnung von Datenobjekten zu den Anwendungsfunktionen wird mittels eines Regelsystem durchgeführt. Der Datenzustand wird ebenfalls durch das Regelsystem kontrolliert und kann damit die Anwendungsfunktionen zuordnen. |
| DPMS | Ein Regelsystem wird nicht beschrieben. |
| DDPS | Ein Regelsystem beschreibt die Korrelation zwischen Datenobjekten und den Anwendungsfunktionen. Es wird beschrieben, welches Datenobjekt an einer Anwendungsfunktion teilnimmt und welcher Zustand am Ende vorliegen muss. Das Regelsystem kontrolliert die korrekte Ausführung eines Prozesses |

Tabelle 7.33: Vergleich: Regelsystem

| Ansatz | Vergleich: Sicherheitssysteme |
|----------------|---|
| α -Flow | Es wird ein Autorisierungs- und ein Authentifizierungssystem beschrieben. |
| BPMN | Es existieren Erweiterungen im Funktionsumfang von BPMN, die in [RFMP07] beschrieben werden. Diese erlauben die Einführung von PKI. Allerdings muss dann der Übersetzer an die neuen Symbole angepasst werden, weil diese nicht standardmäßig für die Übersetzung in ein lauffähiges System geeignet sind. Auch kann nicht im Voraus definiert werden, was beispielsweise verschlüsselt werden soll. Hier muss jeder Benutzer selbst definieren, ob und was genau verschlüsselt werden soll. Angaben zu Autorisierungs- oder Authentifizierungssystemen liegen vor. |
| ACM | ACM besitzt eine Sicherheitsstruktur. Dazu zählen: Authentifizierungs-, Autorisierungs-, Verschlüsselungs-, Protokollierungs- und Prüfsysteme. Zusätzlich können digitale Signaturen angewendet werden. |
| ACBPM | Angaben zu einem Sicherheitssystem liegen nicht vor. |
| DPMS | Authentifizierungs- und Autorisierungssysteme werden beschreiben. |
| DDPS | Angaben zu einem Sicherheitssystem liegen nicht vor. |

Tabelle 7.34: Vergleich: Sicherheitssysteme

7 Vergleich der unterschiedlichen Ansätze zur Prozessunterstützung

| Ansatz | Vergleich: Benutzerverwaltung |
|----------------|---|
| α -Flow | Eine Benutzerverwaltung wird bereitgestellt. Für einen Benutzer werden Informationen zu seiner Rolle und die ihm zugeordnete Institution benötigt. Mittels den zugeordneten Rechten wird der Zugang zu den α -Cards gewährt. Für den Systemzugriff wird ein Autorisierungs- und Authentifizierungssystem beschrieben. |
| BPMN | Eine Benutzerverwaltung wird erst durch das WfMS erstellt. |
| ACM | Die Benutzerverwaltung erfasst die Namen der Anwender und die Rollen, die diese in einem Prozess besitzen können. Zusätzlich wird die Hierarchie beschrieben. Für den Zugang wird ein Autorisierungs- und Authentifizierungssystem beschrieben, mit dessen Hilfe auch ein auf biometrischen Daten basierender Zugang möglich sein soll. |
| ACBPM | Über eine Benutzerverwaltung liegen keine Informationen vor. |
| DPMS | Es existiert eine Benutzerverwaltung, die auf der Basis eines Rollensystems arbeitet. Die Benutzerdaten sind durch Verweise mit den Daten und Anwendungsfunktionen verbunden. Dadurch kann festgelegt werden, wo im Prozessablauf welcher Zugriff durch welchen Anwender möglich ist. |
| DDPS | Über eine Benutzerverwaltung liegen keine Informationen vor |

Tabelle 7.35: Vergleich: Benutzerverwaltung

| Ansatz | Vergleich: Protokollierungssystem |
|----------------|---|
| α -Flow | Das Protokollierungssystem wird vom Ansatz nicht explizit thematisiert. Allerdings werden Zugangsversuche und durchgeführte Modifikationen dokumentiert. Zusätzlich muss im Varianten- und Versionsmanagement festgehalten werden, welche Version einer α -Card die global gültige ist. Daher kann von einem impliziten Protokollierungssystem ausgegangen werden. |
| BPMN | Es liegen keine Informationen zu einem Protokollierungssystem vor. |
| ACM | Ein Protokollierungssystem wird mittels Protokolldatei realisiert. Die Protokolldatei speichert sowohl die während des Prozessverlaufes durchgeführten Modifikationen als auch den Namen des Nutzers, der diese durchgeführt hat. |
| ACBPM | Bei dem Monitoring-System wird Data-Mining auf Protokolldateien durchgeführt. Allerdings wird nicht beschrieben, wie die Protokolldaten generiert werden und was genau in diesen enthalten ist. |
| DPMS | Es liegen keine Informationen über ein Protokollierungssystem vor |
| DDPS | Es liegen keine Informationen über ein Protokollierungssystem vor |

Tabelle 7.36: Vergleich: Protokollierungssystem

| Ansatz | Vergleich: Ein- und Ausgabeverwaltung |
|----------------|--|
| α -Flow | Ein Datenobjekt kann jede Art von Informationen aufnehmen. Unter anderem auch eigenständige Dateien. Zusätzlich besitzt ein Datenobjekt auch Formularfelder, die anfallende Informationen aufnehmen können. Die Ausgabe der Informationen wird über die Benutzeroberfläche gesteuert. Bei der Ausgabe von Formulardaten oder der aufgenommenen Dateien wird automatisch das Programm aufgerufen, das von dem Betriebssystem für den Datentyp vorgesehen ist. |
| BPMN | Der Datenfluss in oder aus einer Workflow-Instanz kann in BPMN als ein Informationsfluss mittels Datenobjekten modelliert werden. Nach welchem Prozess die Ein- oder Ausgabe von Daten abläuft, wird nicht genau beschrieben. Somit kann nicht spezifiziert werden, welche Eingangsdaten gelesen werden können oder, ob sich beispielsweise Firmenlogos in den Ausgabedaten befinden können. |
| ACM | ACM ermöglicht das Einlesen verschiedener Datenformate. Nach dem Einlesen der Daten erfolgt eine Kategorisierung, um die Informationen einordnen zu können. Danach stehen die Informationen für die Verarbeitung zur Verfügung. Ausgabedaten werden für den jeweiligen Einsatzzweck mittels Firmenlogo oder digitaler Signatur versehen. |
| ACBPM | Informationen können von der Außenwelt importiert werden. Wie diese Informationen in den Prozess einfließen, wird nicht beschrieben. Es liegen auch keine Informationen darüber vor, wie Daten ausgegeben werden können. |
| DPMS | Es liegen keine Informationen über ein Ein- und Ausgabesystem vor |
| DDPS | Es liegen keine Informationen über ein Ein- und Ausgabesystem vor |

Tabelle 7.37: Vergleich: Ein- und Ausgabeverwaltung

| Ansatz | Vergleich: Anmerkungs-system |
|----------------|--|
| α -Flow | Es liegen keine Informationen über ein Anmerkungs-system vor. |
| BPMN | Jeder Benutzer kann Anmerkungen neben den BPMN-Symbole platzieren. |
| ACM | Jeder Prozessteilnehmer soll den Prozesselementen Anmerkungen hinzufügen können. |
| ACBPM | Es liegen keine Informationen über ein Anmerkungs-system vor. |
| DPMS | Es liegen keine Informationen über ein Anmerkungs-system vor. |
| DDPS | Es liegen keine Informationen über ein Anmerkungs-system vor. |

Tabelle 7.38: Vergleich: Anmerkungs-system

| Ansatz | Vergleich: Lokalisierung von Anwendungsfunktionen |
|----------------|--|
| α -Flow | Es werden keine Anwendungsfunktionen benutzt. |
| BPMN | Es liegen keine Informationen über eine mögliche Lokalisierung von Anwendungsfunktionen vor. |
| ACM | Es liegen keine Informationen über eine mögliche Lokalisierung von Anwendungsfunktionen vor. |
| ACBPM | Es liegen keine Informationen über eine mögliche Lokalisierung von Anwendungsfunktionen vor. |
| DPMS | Es liegen keine Informationen über eine mögliche Lokalisierung von Anwendungsfunktionen vor. |
| DDPS | Es liegen keine Informationen über eine mögliche Lokalisierung von Anwendungsfunktionen vor. |

Tabelle 7.39: Vergleich: Lokalisierung von Anwendungsfunktionen

| Ansatz | Vergleich: Koordinierung von Daten |
|----------------|---|
| α -Flow | Die Koordinierung der unterschiedlichen Datenobjekte wird mittels eines Variantensystem durchgeführt. Wie das Variantensystem arbeitet, wird nicht beschrieben. |
| BPMN | Es liegen keine Informationen über eine Koordinierung von Daten vor. |
| ACM | Die Daten liegen zentral auf einen DBS |
| ACBPM | Es liegen keine Informationen über eine Koordinierung von Daten vor. |
| DPMS | Es liegen keine Informationen über eine Koordinierung von Daten vor. |
| DDPS | Es liegen keine Informationen über eine Koordinierung von Daten vor. |

Tabelle 7.40: Vergleich: Koordinierung von Daten

| Ansatz | Vergleich: Versionsverwaltung |
|----------------|---|
| α -Flow | Alle öffentlichen und gültigen α -Cards unterliegen einer Versionierung. Bei jeder Änderung einer öffentlichen und gültigen α -Card wird die Versionsnummer hochgezählt. Die Versionsnummer wird global hochgezählt, um die Verfolgbarkeit der durchgeführten Modifikationen zu gewährleisten |
| BPMN | Es liegen keine Informationen über eine Versionsverwaltung vor. |
| ACM | Alle modellierten Prozesselemente stellen eine eigene Version dar, um sie als Vorlagen für neue Prozesselemente verwenden zu können. |
| ACBPM | Es liegen keine Informationen über eine Versionsverwaltung vor. |
| DPMS | Es liegen keine Informationen über eine Versionsverwaltung vor. |
| DDPS | Es liegen keine Informationen über eine Versionsverwaltung vor. |

Tabelle 7.41: Vergleich: Versionsverwaltung

| Ansatz | Vergleich: Fehlererkennung |
|----------------|---|
| α -Flow | Es liegen keine Informationen zu einer Fehlererkennung vor. |
| BPMN | Fehler werden erkannt, wenn eine entsprechende Ereignisbehandlung modelliert wurde. |
| ACM | Es liegen keine Informationen zu einer Fehlererkennung vor. |
| ACBPM | Es liegen keine Informationen zu einer Fehlererkennung vor. |
| DPMS | Es liegen keine Informationen zu einer Fehlererkennung vor. |
| DDPS | Es liegen keine Informationen zu einer Fehlererkennung vor. |

Tabelle 7.42: Vergleich: Fehlererkennung

| Ansatz | Vergleich: Fehlerkorrektur |
|----------------|--|
| α -Flow | Es liegen keine spezifischen Informationen zu einer Fehlerkorrektur vor. |
| BPMN | Fehler können korrigiert werden, falls entsprechende Korrekturmechanismen vorgesehen sind. |
| ACM | Es liegen keine spezifischen Informationen zu einer Fehlerkorrektur vor. |
| ACBPM | Es liegen keine Informationen zu einer Fehlerkorrektur vor. |
| DPMS | Es liegen keine Informationen zu einer Fehlerkorrektur vor. |
| DDPS | Es liegen keine Informationen zu einer Fehlerkorrektur vor. |

Tabelle 7.43: Vergleich: Fehlerkorrektur

| Ansatz | Vergleich: Verklemmungsvermeidungssystem |
|----------------|---|
| α -Flow | Es existiert nur ein Dokument, somit können keine Verklemmungen auftreten. |
| BPMN | Verklemmungen können auftreten. Aber es liegen keine Informationen über Verklemmungsvermeidungssystem vor. |
| ACM | Durch ein zentrales DBS wird mittels einer Transaktionsverwaltung der Zugriff auf eine Datenbank beschrieben. |
| ACBPM | Der Zugriff einer Anwendungsfunktion auf zwei Datenobjekte wird nicht beschrieben. |
| DPMS | Der Zugriff einer Anwendungsfunktion auf zwei Datenobjekte wird nicht beschrieben. |
| DDPS | Der Zugriff einer Anwendungsfunktion auf zwei Datenobjekte wird nicht beschrieben. |

Tabelle 7.44: Vergleich: Verklemmungsvermeidungssystem

7.2.8 Evolutionäre Aspekte

In diesem Abschnitt wird der Vergleich anhand der Vergleichskriterien der evolutionären Aspekte in folgender Reihenfolge in Tabellen dargestellt:

- Anpassungsfähigkeit des Datenschemas zur Laufzeit
- Anpassungsfähigkeit der Anwendungsfunktionen zur Laufzeit
- Anpassungsfähigkeit der Anzahl von Anwendungsfunktionen zur Laufzeit
- Anpassungsfähigkeit der Benutzerverwaltung zur Laufzeit
- Anpassungsfähigkeit des Regelsystems zur Laufzeit
- Anpassungsfähigkeit des Routingsystems zur Laufzeit
- Anpassungsfähigkeit des Protokollierungssystems zur Laufzeit

| Ansatz | Vergleich: Anpassungsfähigkeit des Datenschemas zur Laufzeit |
|----------------|--|
| α -Flow | Durch das Hinzufügen von α -Cards kann das System beliebig viele Nutzinformationen aufnehmen. Eine definierte maximale Größe wird nicht beschrieben. |
| BPMN | Eine Veränderungsmöglichkeit des Datenschemas zur Laufzeit wird nicht beschrieben. |
| ACM | Neue Tupel für die Datenbank können erzeugt werden. Wie die dynamische Schemaänderung der Tabellen durchgeführt wird, ist nicht beschrieben. |
| ACBPM | Während der Laufzeit ist es möglich den Datenobjekten Attribute hinzufügen. Neue Datenobjekte können auch während des Verlaufes eingebunden werden. Sobald ein neues Attribute hinzugefügt wird, muss der Lebenszyklus modifiziert werden. Wird ein neues Datenobjekte hinzugefügt, dann muss ein neuer Lebenszyklus erstellt werden. Zusätzlich müssen die Assoziationen gepasst werden, welche die Zuordnung zwischen Datenobjekten und Anwendungsfunktionen definieren. Analog gilt das Gleiche für das Hinzufügen oder Entfernen von Anwendungsfunktionen. |
| DPMS | Es ist möglich während der Laufzeit neue Datenobjekte einzuführen. Dabei müssen die zu erreichenden Datenzustände angepasst werden und damit auch der Lebenszyklus. Ein besonderes Augenmerk muss hierbei auf die Datenobjekthierarchie gelegt werden. Durch das Einfügen eines neuen Datenobjektes wird die Struktur der Verweise zwischen den Datenobjekten verändert. Dies führt zu einer Anpassung der Anwendungsfunktionen und die damit verbundenen Zuordnung. |
| DDPS | Das Einführen von Datenobjektattributen oder deren Entfernung bedingt immer einer Änderung der Datenzustände. Daraus resultiert eine Veränderung des Lebenszyklus und der Abbildung der Zustände auf die Anwendungsfunktionen. Die Anwendungen müssen an die neue Anzahl an Attributen und/oder Datenobjekten angepasst werden. Zusätzlich muss das Regelsystem angepasst werden, das die Zuordnung zwischen Daten und Anwendungsfunktion durchführt. Gleiches gilt für neue Datenobjekte. |

Tabelle 7.45: Vergleich: Anpassungsfähigkeit des Datenschemas zur Laufzeit

7 Vergleich der unterschiedlichen Ansätze zur Prozessunterstützung

| Ansatz | Vergleich: Anpassungsfähigkeit der Anwendungsfunktionen zur Laufzeit |
|----------------|---|
| α -Flow | Dieser Ansatz besitzt keine Anwendungsfunktionen. |
| BPMN | Anpassungsfähige Anwendungsfunktionen während der Laufzeit können mittels Ad-Hoc-Anwendungsfunktionen beschrieben werden, die Sub-Prozessverläufe ermöglichen. Allerdings sind alle möglichen Prozessabläufe in einer ad-hoc-Funktion vorgegeben. |
| ACM | Anwendungsfunktionalitäten einer Anwendungsfunktion sollen während der Laufzeit verändert werden können. Dies erfordert aber eine Anpassung der Anwendungsfunktionen. Dabei muss darauf geachtet werden, dass das vorgegebene Geschäftsziel noch erreicht werden kann. Dies wird aber von dem Regelsystem, das die Ausführung kontrolliert überprüft. Es wird keine Aussage darüber gemacht, ob die Anwendungsfunktionen verändert werden können, während sie aktiv sind. |
| ACBPM | Anwendungsfunktionalitäten einer Anwendungsfunktion sollen während der Laufzeit verändert werden können. Es wird keine Aussage darüber gemacht, ob die Anwendungsfunktion während sie aktiv ist verändert werden kann. Bei signifikanten Modifikationen der Anwendungsfunktionalitäten müssen die Assoziationen angepasst werden, die eine Anwendungsfunktion einem Datenobjekt zuordnet. Durch die Änderungen an den Zuständen der Datenobjekte kann es notwendig sein auch den Lebenszyklus anzupassen. |
| DPMS | Modifikationen der Anwendungsfunktionen während der Laufzeit sind möglich. Allerdings setzt dies eine Veränderung der Datenobjekte voraus, da die Entwicklung eines Prozesses immer von den Daten ausgeht. |
| DDPS | Modifikationen der Anwendungsfunktionen während der Laufzeit sind möglich. Signifikant veränderte Funktionalitäten können eine Anpassung der Prozesskonfiguration erzwingen. |

Tabelle 7.46: Vergleich: Anpassungsfähigkeit der Anwendungsfunktionen zur Laufzeit

| Ansatz | Vergleich: Anpassungsfähigkeit der Anzahl von Anwendungsfunktionen zur Laufzeit |
|----------------|---|
| α -Flow | Dieser Ansatz besitzt keine Anwendungsfunktionen. |
| BPMN | Eine Veränderungsmöglichkeit der Anzahl der Anwendungsfunktionen zur Laufzeit wird nicht beschrieben. |
| ACM | Die Anzahl der Anwendungsfunktionen kann variieren. Wie die technische Umsetzung aussehen soll, wird nicht beschrieben. |
| ACBPM | Durch die Definition neuer Datenobjekte ist es möglich die Anzahl der Anwendungsfunktionen während der Laufzeit zu verändern Für die Veränderung müssen die Zustände der Datenobjekte angepasst werden, auf die Anwendungsfunktionen zugreifen. Danach kann mittels Assoziation die Zuordnung zwischen Datenobjektzustand und Anwendungsfunktion durchgeführt werden. |
| DPMS | Eine Modifikation der Anzahl der Anwendungsfunktionen während der Laufzeit ist möglich. Allerdings setzt diese Modifikation eine Anpassung der Datenobjekte voraus. Die Veränderungen der Datenobjekt bzw. Attribute müssen als erstes durchgeführt werden. |
| DDPS | Eine Modifikation der Anzahl der Anwendungsfunktionen ist nur dann möglich, wenn die Anwendungsfunktionen dem definierten Schema der Datenzustände entsprechen. Es besteht die Möglichkeit neue Regeln zu definieren, welche die Zuordnung von Datenzustand und den neuen Anwendungsfunktionen durchführt. |

Tabelle 7.47: Vergleich: Anpassungsfähigkeit der Anzahl von Anwendungsfunktionen zur Laufzeit

| Ansatz | Vergleich: Anpassungsfähigkeit der Benutzerverwaltung zur Laufzeit |
|----------------|---|
| α -Flow | Benutzer können zur Laufzeit hinzugefügt und / oder entfernt werden. |
| BPMN | Diese Möglichkeit wird von dem WfMS verwaltet und nicht von BPMN. Somit kann hierzu keine Aussage gemacht werden. |
| ACM | Benutzer können zur Laufzeit hinzugefügt und / oder entfernt werden. |
| ACBPM | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit der Benutzerverwaltung zur Laufzeit vor. |
| DPMS | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit der Benutzerverwaltung zur Laufzeit vor |
| DDPS | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit der Benutzerverwaltung zur Laufzeit vor. |

Tabelle 7.48: Vergleich: Anpassungsfähigkeit der Benutzerverwaltung zur Laufzeit

| Ansatz | Vergleich: Anpassungsfähigkeit des Regelsystems zur Laufzeit |
|----------------|---|
| α -Flow | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Regelsystems zur Laufzeit vor. |
| BPMN | Diese Möglichkeit wird von dem WfMS verwaltet und nicht von BPMN. Somit kann hierzu keine Aussage gemacht werden. |
| ACM | Die Geschäftsbedingungen können sich ändern. Somit muss das Regelsystem an die neuen Rahmenbedingungen der Außenwelt angepasst werden. Dafür können Regeln zur Laufzeit definiert und ausgeführt werden. |
| ACBPM | Das Regelsystem spiegelt die Assoziationen wieder, welche die Anwendungsfunktionen mit den einzelnen Datenobjekten verbinden. Um sich an veränderten Rahmenbedingungen der Außenwelt anzupassen, ist es möglich neue Regeln zu definieren, die dann ausgeführt werden können. |
| DPMS | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Regelsystems zur Laufzeit vor. |
| DDPS | Durch die Definition neuer Regeln kann das System zur Laufzeit Zuordnungen von Anwendungsfunktionen zu Datenobjekten vornehmen. |

Tabelle 7.49: Vergleich: Anpassungsfähigkeit des Regelsystems zur Laufzeit

| Ansatz | Vergleich: Anpassungsfähigkeit des Routingsystems zur Laufzeit |
|----------------|---|
| α -Flow | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Routingsystems zur Laufzeit vor. |
| BPMN | Diese Möglichkeit wird von dem WfMS verwaltet und nicht von BPMN. Somit kann hierzu keine Aussage gemacht werden. |
| ACM | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Routingsystems zur Laufzeit vor. |
| ACBPM | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Routingsystems zur Laufzeit vor. |
| DPMS | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Routingsystems zur Laufzeit vor. |
| DDPS | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Routingsystems zur Laufzeit vor. |

Tabelle 7.50: Vergleich: Anpassungsfähigkeit des Routingsystems zur Laufzeit

| Ansatz | Vergleich: Anpassungsfähigkeit des Protokollierungssystems zur Laufzeit |
|----------------|---|
| α -Flow | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Routingsystems zur Laufzeit vor. |
| BPMN | Diese Möglichkeit wird von dem WfMS verwaltet und nicht von BPMN. Somit kann hierzu keine Aussage gemacht werden. |
| ACM | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Routingsystems zur Laufzeit vor. |
| ACBPM | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Routingsystems zur Laufzeit vor. |
| DPMS | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Routingsystems zur Laufzeit vor. |
| DDPS | Es liegen keine Informationen bezüglich der Anpassungsfähigkeit des Routingsystems zur Laufzeit vor. |

Tabelle 7.51: Vergleich: Anpassungsfähigkeit des Protokollierungssystems zur Laufzeit

7.3 Bewertung der Ansätze zur Prozessunterstützung

In den letzten Abschnitten wurde der Vergleich anhand der acht Aspekte durchgeführt. Durch diesen Vergleich haben sich vier unterschiedliche Ausprägungen der inhaltsbasierten Ansätze ergeben. α -Flow beschreibt einen Ansatz, der auf der Basis eines Dokumentes arbeitet. Das α -Flow-Dokument implementiert neben den inhaltsbasierten Informationen zusätzlich noch Verwaltungsfunktionalitäten und wird dadurch zu einem aktiven Dokument.

BPMN stellt einen Ansatz dar, der den Sequenzfluss der Aktivitäten in den Mittelpunkt stellt. Die Datenobjekte können den Prozess beeinflussen, in dem Gateways auf die Daten in den Datenobjekten zugreifen. Den Sequenzfluss der Aktivitäten können Datenobjekte nicht verändern.

ACM basiert auf einem zentralen DBS, das als Back-End dient. Um die Daten zu manipulieren, wird auf das DBS mittels einer oder mehreren Anwendungsfunktionen zugegriffen. Die Steuerung des Ablaufes wird durch die Daten bestimmt, die die Entscheidungen der Benutzer widerspiegeln.

α -Flow, BPMN und ACM repräsentieren jeweils eine eigene Ausprägung des inhaltsbasierten Ansatz. Die Ansätze ACBPM, DPMS und DDPS verwenden mehrere Dateobjektinstanzen in einer Prozessinstanz. Der Aufbau und der Funktionsumfang der drei Ansätze sind zwar nicht identisch, aber der strukturelle Aufbau der Ansätze ist vergleichbar. Auf die technischen Ausprägungen gehen die Ansätze bis auf α -Flow fast nicht ein. Somit kann nur bedingt Aussagen über die Implementierung gemacht werden.

Bei der Bewertung wird ein besonderes Augenmerk auf die Beschreibung der jeweiligen, Vor- und Nachteile der Ansätze gelegt. Am Ende wird noch auf die Definition des Lebenszyklus eingegangen.

7.3.1 Was leisten die Ansätze zur Prozessunterstützung

Bewertung von α -Flow

Über die Institutionen verteilte Prozessanwender können gemeinsam auf die gleichen Daten zugreifen und mit ihnen arbeiten. Dies wird durch ein elektronisches Dokument unterstützt, das durch die Integration von Funktionalitäten mit den Prozessanwendern interagiert und auf diese Weise den Prozessablauf steuert. Das elektronische Dokument dient dabei als Austauschmedium zwischen den Informationssystemen.

Der Prozessablauf bei α -Flow ist Wissensbasiert, und die Prozessanwender steuern ihn über die Eingabe von Informationen. Ein Prozess repräsentiert einen Fall, der zu bearbeiten ist. Somit stellt ein elektronisches Dokument eine Fallakte dar, die alle Informationen über einen Fall beinhaltet und dann terminiert, wenn zum Erreichen eines Ziels keine weiteren Informationen benötigt werden.

α -Flow ist auf heterogenen Informationssystemen lauffähig, ohne dass eine Installation notwendig wird. Da es auf lose gekoppelten Informationssystemen eingesetzt werden kann, ist es Prozessteilnehmern möglich ad-hoc an einer Prozessinstanz zu partizipieren. Dabei koordiniert α -Flow den Datentransfer zwischen den einzelnen Prozessteilnehmern. Diesen steht ein Replikat eines α -Docs zur Verfügung. Dadurch sind die Daten physisch getrennt, aber logisch zentralisiert.

Die Koordinierung der Daten wird durch die aktiven Eigenschaften des elektronischen Dokuments gewährleistet. Ein Regelsystem registriert Zustandsänderungen der Daten,

beispielsweise wenn deren Sichtbarkeit auf öffentlich gesetzt wird, und aktualisiert sie bei den verteilten Prozessteilnehmern.

α -Flow kann unterschiedliche Dateien für den auszuführenden Fall aufnehmen. Dabei ist deren Dateiformat unerheblich. Die Prozessteilnehmer repräsentieren unterschiedliche Spezialisten, die durch die Institutionsübergreifende Fallbehandlung zu einem dynamischen Team zusammenwachsen können, in dem sie gemeinsam eine Lösung für den auszuführenden Fall suchen. Für den Einsatz von α -Flow ist die Organisationsstruktur nicht relevant. Zusätzlich kann aus einer laufenden Instanz eine neue abgeleitet werden.

Bewertung von Business Process Modelling and Notation

BPMN stellt ein Ansatz dar, der von dem Sequenzfluss von Aktivitäten dominiert ist. Dabei beschreibt der Sequenzfluss der Aktivitäten die einzelnen Aufgaben, die zum Erreichen eines Geschäftsziels notwendig sind. Die Datenobjekte sind nur für die Ausführung, aber nicht für die Steuerung des Prozessverlaufs relevant. Bei der Modellierung von BPMN steht der Sequenzfluss im Mittelpunkt und alle Aktivitäten müssen zum Zeitpunkt der Modellierung definiert sein. BPMN ist für den automatischen Ablauf von Aktivitäten geeignet. Die parallele Bearbeitung von Aufgaben wird unterstützt. Zugleich kann die institutionsübergreifende Kommunikation beschrieben werden.

Um die Kommunikation zu vereinfachen, werden bei der Modellierung einer BPMN Prozesselementen Symbole verwendet. Dies erleichtert die Benutzung, da durch einfaches ziehen und setzen von Prozesselementen der Prozessverlauf dargestellt werden kann. Für die Prozessanwender ist dies vorteilhaft, da sie nicht eigenhändig, den Programmiercode oder eine Beschreibung der Elemente erstellen müssen. Zusätzlich kann BPMN durch eine vereinfachte grafische Modellierung zur Präsentation bei Prozessteilnehmern verwendet werden. Das vereinfachte Modell kann dann um das technische Fachwissen erweitert und von den Übersetzern verarbeitet werden. Ein besonderes Merkmal ist die Unabhängigkeit von den WfMS, die ein BPMN-Modell abbilden. Für ein Modell können mehrere WfMS eingesetzt werden.

Für Anwendungsgebiete mit hohen Fehlerquoten besteht die Möglichkeit, Aktivitäten zurückzusetzen. Die Möglichkeit des Zurücksetzens einer Aktivität wird Transaktion genannt. Das Vorgehen bei einer Transaktion ist gleichzusetzen mit dem TAV in einem DBS. Durch Ereignisse werden kompensierende Aktivitäten gestartet. Die Erzeugung von kompensierenden Aktivitäten kann entweder automatisch oder manuell erfolgen.

BPMN bietet die Möglichkeit einen Prozessablauf für institutionsübergreifende Anwendungen zu erstellen. Somit kann ein Prozessablauf nicht nur die Aktivitäten eines Unternehmens abbilden, sondern es können auch externe Institution integriert werden.

Bewertung Adaptive Case Management

ACM verfolgt eine andere Richtung, als die bisher beschriebenen Ansätze. Bei ACM werden keine Datenobjekte verwendet, sondern ein zentrales DBS. Im Mittelpunkt steht die Anpassbarkeit der Ausführung. Diese Anpassbarkeit beruht auf der Möglichkeit der Prozessteilnehmer ihre Fähigkeiten und Erfahrungen in den Prozess zu integrieren. Dem Prozessteilnehmer ist es damit möglich, die Datenstruktur des DBSs, die Aktivitäten oder die Reihenfolge der Aktivitäten während der Laufzeit zu verändern. Zusätzlich wird ein Regelsystem etabliert, welches die Veränderungen kontrolliert. Damit wird sichergestellt, dass die durchgeführten Modifikationen den definierten Regeln entsprechen. Bei dieser Lösung liegt die Verantwortung bei den Benutzern des Prozesses.

ACM erlaubt die Einbindung von mobilen Geräten und bietet damit die Möglichkeit viele unterschiedliche, moderne, für den mobilen Einsatz entwickelte Informationssysteme, zu integrieren. Die Modellierung der Prozesselemente kann mittels natürlicher Sprache erfolgen. Dadurch kann jeder Prozessteilnehmer den Prozess modifizieren. Alle Vorgänge sind für die Prozessteilnehmer transparent. Jeder Prozessteilnehmer kann den vorliegenden Informationen entnehmen, wer welche Modifikationen an einem Prozessverlauf vorgenommen hat. Dies folgt dem Verantwortungsprinzip, das eine kooperative Arbeitsumgebung mittels Verantwortungskonzept und Transparenz unterstützt.

Der ACM Ansatz implementiert Sicherheitsmechanismen. Zu den Sicherheitssystemen zählen beispielsweise Authentifizierungssysteme, Autorisierungssysteme und Verschlüsselungssysteme. Zusätzlich gibt es Verschlüsselungsmechanismen, die die Verbindung zwischen zentralem DBS und den Clients schützen. Dies ist notwendig, um die Daten vor dem Zugriff Dritter zu schützen.

Die Prozessteilnehmer können mittels einer Benutzeroberfläche ihre Anwendungsfunktionen selber anordnen. Dies erleichtert die Arbeit, wenn z.B. Prozessteilnehmer ihre favorisierten Aktivitäten auf die erste Ebene der Benutzeroberfläche legen können.

Der Lebenszyklus von Elementen eines Prozess gliedert sich in drei Schritte. Prozesselemente werden modelliert, ausgeführt und archiviert. Die Modellierung kann auch während der Laufzeit stattfinden. Durch die Archivierung stehen, die Prozesselemente als Schablonen für neue Prozessinstanzen zu Verfügung.

Bewertung von Artifact-Centric Business Process Models

ACBPM arbeitet auf einem oder mehrere passiven Datenobjekte. In ihnen werden neben den Nutzdaten noch Informationen über den Prozessverlauf gespeichert. Der Inhalt der Datenobjekte kann mittels ER-Diagrammen oder einem XML-Schema erstellt werden können. Wie die Daten physisch gespeichert werden, wird offengelassen. Es könnte sein, dass die Datenobjekte in einer Datenbank gespeichert werden. ACBPM führt während des Prozessverlaufes Analysen durch, um den Effizienz eines Prozesses zu verbessern.

Das Regelsystem ist das zentrale Element bei der Zuordnung von Anwendungsfunktionen zu den Datenzuständen. Es wird nicht beschrieben, wie die Regeln definiert werden, und wie die Regeln zur Laufzeit angepasst werden können. Allerdings bietet ACBPM die Möglichkeit mit Hilfe des Regelsystems und den vier Dimensionen den Prozessablauf von datenorientierter auf aktivitätsbasierte Steuerung umzustellen. Somit können die Benutzer entscheiden, ob sie eher einen Wissensbasierten durch Daten repräsentierten Ablauf benötigen, oder einen automatisierten Sequenzfluss von Aktivitäten. Jede Änderung am System wird mittels eines Protokollierungssystems festgehalten. Zu den Änderungen gehören das Eintragen von Daten oder das Einfügen von neuen Prozesselementen. Zusätzlich wird ein Ein- und Ausgabesystem beschrieben, das für die Integration von Daten aus der Außenwelt verantwortlich ist.

Das Datenschema und das Regelsystem können sich während der Laufzeit an geänderte Rahmenbedingungen anpassen. Dafür ist es unerheblich, ob die Änderungen von der Außenwelt oder durch den Verlauf des Prozess hervorgerufen werden.

Bewertung von Datenorientiertes Prozess-Management-System

DPMS beschreibt einen Ansatz, der mit mehreren Datenobjekten arbeitet. Es besteht die Möglichkeit, die Datenobjekte in einem DBS zu speichern. In den Datenobjekten werden neben der Speicherung der Nutzdaten noch die Verlaufsdaten eines Prozesses gespeichert. Die Daten werden durch die Datenzustände kodiert. Dabei stellt ein Datenzustand die Werte der Attribute der Datenobjekte dar. Die Datenobjekte können bei DPMS in einer Beziehung, beispielsweise 1:N, stehen. Damit können die Verbindungen zwischen den Entitäten der Außenwelt dargestellt werden.

Bei der Entwicklung stehen die Datenobjekte im Vordergrund und die Anwendungsfunktionen werden an die Attribute der Datenobjekte angepasst. Bei der Modellierung der Datenzustände wird ein Prozessende festgelegt. Es kann zwar durch Veränderung der Datenzustände verändert werden, allerdings ist ein Ende im Design vorgesehen. Ein

Prozessteilnehmer kann die Datenobjekte entwickeln und die Datenzuständen modellieren. Nachträglich können dann die Anwendungsfunktionen erstellt werden.

DPMS ermöglicht die Anpassung des Datenschemas der Datenobjekte und deren Anzahl. Während der Laufzeit können Datenobjekte in eine Prozessinstanz hinzugefügt werden. Bei der dynamischen Erstellung neuer Datenobjekte, wird erst die Notwendigkeit der Einbeziehung der Daten betrachtet. Danach werden für die Datenzustände die Aktivitäten definiert. Somit kann auf Änderungen der Außenwelt eingegangen werden.

In DPMS werden die Hierarchiestufen noch entkoppelt. Untergeordnete Datenobjekte können asynchron zu den übergeordneten Objekten verarbeitet werden. Dies ermöglicht eine Optimierung der Parallelisierung. Zudem können die übergeordneten Datenobjekte die Ergebnisse der untergeordneten Datenobjekte aggregiert zusammenfassen. Somit muss nicht jedes untergeordnete Datenobjekt untersucht werden, um die Ergebnisse zusammen zu fassen. Dies bietet die Möglichkeit Benutzer auf unterschiedlichen Ebenen im Prozess einzusetzen.

DPMS ermöglicht während der Ausführung einen Überblick über alle Daten, die für eine Prozessinstanz vorliegen. Die Sichtbarkeit wird nur von den Benutzerrechten reglementiert.

Bewertung von Data-driven Process Structures

DDPS verwendet mehrere Datenobjekte für eine Prozessinstanz. Ein Datenobjekt enthält neben den Nutzdaten noch die Prozessverlaufsdaten. Die Prozessverlaufsdaten sind durch die Attributwerte der Nutzdaten beschrieben. Somit ergeben sich für ein Datenobjekt unterschiedliche Datenobjektzustände.

Die Steuerung eines Prozessverlaufs hängt bei DDPS nicht nur von den Datenzuständen, sondern auch von der Prozesskonfiguration ab. Anhand der Datenzustände wird der Prozessverlauf beschrieben. Die Datenzustände werden auch als Lebenszyklus dargestellt. Die Datenzustände werden am Ende der Entwicklungsphase mit der Prozesskonfiguration zusammengeführt. Die Datenobjekt- und die Prozessentwickler können aus diesem Grund getrennt voneinander Datenobjekte und Anwendungsfunktionen erstellen.

Die zentrale Einrichtung ist das Regelsystem oder auch Prozesskonfiguration genannt. Auf der Basis einfacher, von den Prozessanwendern definierten Regeln, ordnet diese die Anwendungsfunktionen den einzelnen Datenobjekten zu.

Ein weiterer Vorteil dieses Ansatzes ist die Anpassbarkeit der Anwendungen an geänderte Rahmenbedingungen. Dabei können neben den Anwendungsfunktionen auch

die Datenobjekte angepasst werden. Jede Anpassung erfordert eine Generierung neuer Regeln durch die Prozessanwender.

7.3.2 Wo liegen die Grenzen der Ansätze zur Prozessunterstützung

Bewertung von α -Flow

α -Flow verbindet lose gekoppelte Systeme über ein Kommunikationsnetz. Dieses kann aus unterschiedlichen Gründen unterbrochen werden, wie beispielsweise durch den Ausfall von Netzwerkknoten. Dies wirkt sich nachteilig aus, denn bei einer spontanen Trennung vom Kommunikationsnetz werden die Daten nicht mehr synchronisiert, und eine parallele Verarbeitung der Daten ist nur noch bedingt möglich. Durch ein solches Ereignis wird auch die Echtzeitfähigkeit beeinflusst.

Die Einbindung neuer digitaler Informationssysteme, wie beispielsweise Pads oder Personal Digital Assistant (PDA)s, die eine räumliche Flexibilität erlauben, wird nicht unterstützt

Auch kann die Benutzeroberfläche nicht angepasst werden. Soll α -Flow für andere Aufgabenbereiche angewendet werden, dann müsste die Benutzeroberfläche manuell von IT-Fachpersonal angepasst werden.

Bewertung von Business Process Modelling and Notation

BPMN analysiert nur die Geschäftsziele und nicht die Benutzer oder die Infrastruktur auf die das Prozessunterstützungssystem angewendet werden soll. Mögliche Inkompatibilitäten müssen in einem weiteren externen Prozess analysiert werden. Eine Benutzerinteraktion kann nur mit Hilfe der für explizite Benutzer definierten Anwendungsfunktionen realisiert werden. Während der Prozessausführung können keine Veränderungen an den Prozesselementen durchgeführt werden. Dies gilt auch für die Aktivitäten, die während der Laufzeit, durch den Prozessanwender ebenfalls nicht beeinflusst werden können. In der Modellierungsphase existiert die Möglichkeit Ad-Hoc-Anwendungsfunktionen zu modellieren. Dabei sind die Variationsmöglichkeiten der tatsächlichen Ausprägung der Eigenschaft des Prozesses durch eine feste Auswahl von Funktionalitäten begrenzt.

Übergangsbedingungen müssen mit Hilfe von Programmiercodes eingegeben werden. Die Möglichkeit hierzu die natürlicher Sprache zu benutzen ist nicht vorgesehen. Somit müssen Prozessanwender eine Programmiersprache erlernen, um Veränderungen an einem Prozess durchzuführen. Zusätzlich ist es nicht möglich, mehrere Prozessanwender gleichzeitig an einem bestimmten Arbeitsschritt teilnehmen zu lassen. Kommunikationsmittel

zwischen den Prozessanwendern existieren ebenfalls nicht, was die soziale Interaktion mit anderen Prozessanwendern erschwert. Zusätzlich haben die Prozessanwender auch nicht die Möglichkeit transparent auf Informationen im Prozess zu zugreifen. Aus diesem Grund können Prozessanwender nicht nachvollziehen, aus welchem Grund Änderungen vorgenommen wurden. Diese Eigenschaften könnten je nach Prozessumgebung, welche BPMN umsetzt, durch das Prozessmanagementsystem übernommen werden.

Ein Sicherheitskonzept bzw. Sicherheitsstrukturen existieren nur in einer Erweiterung von BPMN. Diese Erweiterung ist nicht standardisiert. Für eine Erweiterung des Umfangs von BPMN wird das Wissen von IT-Fachpersonal benötigt. Während der Ausführung können mehrere Anwendungsfunktionen parallel auf einem Datenobjekt lesend oder schreibend zugreifen. Eine Koordinierung von parallelen Schreibzugriffen wird nicht beschrieben. Dies könnte zu Inkonsistenzen in den Daten führen.

Aufgrund seiner grafische Benutzeroberfläche, die mittels ziehen und setzen von Symbolen arbeitet, können Benutzer innerhalb kurzer Zeit einen eigenen Prozessablauf modellieren. Für die Übersetzung von BPMN existieren unterschiedliche Übersetzer. Eine Möglichkeit wäre BPMN nach Business Process Execution Language (BPEL) zu übersetzen. Regel- oder Benutzerverwaltungssysteme werden erst nach der Übersetzung hinzugefügt und konnten aus diesem Grund nicht bewertet werden. Zusätzlich muss der Zugriff auf die Daten reglementiert werden, um zu verhindern, dass keine unbefugten Personen auf die Daten zugreifen können.

Der Ansatz terminiert, wenn im Zuge des Verlaufs die Aktivitäten abgearbeitet sind. Ist dieser Fall eingetreten, dann kann eine Prozessinstanz nicht wieder zum Leben erweckt werden. Der Lebenszyklus beschreibt den Sequenzfluss der Aktivitäten mit einem definierten Ende

Bewertung von Adaptive Case Management

ACM fordert viele funktionale Eigenschaften, wie beispielsweise die Verwendung der natürlichen Sprache, den Einsatz von IM oder eine globale Transparenz aller Informationen. Es wird allerdings nicht beschrieben wie diese Funktionalitäten technisch umgesetzt werden können. Viele realisierbare Eigenschaften von ACM werden aus den CM übernommen.

Das Regelsystem kontrolliert die Ausführung und Modifikationen der Prozesselemente in einem Prozessverlauf. Trotz der Kontrolle durch ein Regelsystem können Fehler bei der Modifikation eines Prozesses auftreten. Diese Fehler können allerdings durch entsprechende Eingriffe der Benutzer behoben werden. Eingriffe in den Prozessverlauf

sollten nur dann durchgeführt werden, wenn der jeweilige Prozessanwender den kompletten Prozessverlauf überblicken kann, da lokale Änderungen an anderer Stelle sich als Fehler erweisen können. Dies kann durch die Definition von Teilzielen vermieden werden. Die gleiche Sorgfalt ist auch bei der Verwendung der adaptiven Benutzeroberfläche notwendig. Jeder Prozessanwender kann die Benutzeroberfläche konfigurieren. Die Möglichkeit der persönlichen Adaption der Benutzeroberfläche kann aber auch zum Nachteil werden, nämlich dann, wenn andere Nutzer sich erst an die neue Oberfläche gewöhnen müssen,

Informationen über die Koordinierung und das Kommunikationsnetz mit dem Backend und den Clients liegen nicht vor. Allerdings ist das zentrale DBS der Flaschenhals des ganzen Systems. Alle Clients greifen darauf zu und müssen schnell eine Antwort erhalten. Eine Realtime-Verarbeitung ist damit fast ausgeschlossen. Fragen bezüglich der Verteilung eines zentralen DBS für ACM und bis zur welcher Anzahl an Clientsystemen das System noch effektiv arbeitet bleiben offen.

Ein paralleler Schreibzugriff auf die Daten wird genau so wenig beschrieben, wie die Möglichkeit, dass mehrere Clients auf verschiedene Daten gleichzeitig zugreifen können. Allerdings liegt ein DBS zur Grunde, welches eine Transaktionsverwaltung besitzt, die sich den parallelen Lese- und Schreib- Zugriff organisiert. Dies wird aber von den Quellen nicht beschrieben.

Es ist in ACM möglich das Datenschema anzupassen. Dies bei einer Verwendung eines DBSen während der Laufzeit zu realisieren, ist nicht einfach. Wie dies realisiert wird, wird nicht beschrieben.

Bewertung von Artifact-Centric Business Process Models

Ein Schachpunkt von ACBPM ist die fehlende Beschreibung, wie die Daten physisch gespeichert werden. Daher kann die Frage nicht beantwortet werden, ob ein zentraler Server für die Datenspeicherung verwendet oder mittels lose gekoppelten Systemen gearbeitet wird.

Es werden Benutzer beschrieben, die mit dem System interagieren können, aber wie sie an einem Prozess teilnehmen können, welche Änderung ein Benutzer durchführen kann, wie diese Veränderungen ersichtlich für andere Prozessteilnehmer werden, wird nicht ersichtlich.

Die Prozessteilnehmer müssen bei Modifikationen des Prozessverlaufes oder Prozesselemente wissen, welche Auswirkungen diese auf den Verlauf eines Prozesses haben. Eine Zugriffsmöglichkeit auf allen Daten wird nicht beschrieben.

Die Möglichkeit einer parallelen Bearbeitung von Daten wird ebenfalls nicht beschrieben. Prozesssteilnehmer können nicht an unterschiedlichen Standorten gleichzeitig auf die Daten zugreifen und gegebenenfalls Daten ändern. Daten können während des Prozessverlaufes Fehler aufweisen, aber eine Fehlerbehandlung, die beispielsweise Daten wiederherstellt, wird nicht beschrieben. Auch ein System der Fehlervermeidung wird nicht beschrieben.

Gleichzeitig werden keine Maßnahmen beschrieben, die die soziale Interaktion der Benutzer fördert. Somit sind die Benutzer auf sich gestellt und es kann sich keine Gruppendynamik bei einem Prozessablauf entwickeln. Auch wird nicht beschrieben, ob unterschiedliche Institutionen an einem Prozessablauf beteiligt werden können.

Bewertung von Datenorientiertes Prozess-Management-System

Prozessanwender werden von DPMS kaum unterstützt. Es wird weder die Verwendung einer Benutzeroberfläche noch die Möglichkeit der Verwendung der natürlichen Sprach beschrieben. Eine Unterstützung zur kooperativen Arbeit und der damit zusammenhängenden sozialen Interaktion der Prozessanwender wird ebenfalls nicht beschrieben. Auch müssen die Prozesselemente immer neu Modelliert werden. Eine Möglichkeit der Ableitung von bereits verwendeten Prozesselementen für neue Prozessinstanzen wird nicht beschrieben.

Eine Ereignisverwaltung wird nicht beschrieben. Auf Fehler oder Ereignisse kann DPMS nicht reagieren. Auch wird nicht beschrieben, ob mit DPMS ein Langzeiteinsatz realisiert oder ob es nur für kurze Prozessabläufe verwendet werden kann.

Informationen, wie andere Informationssysteme, in einen Prozessablauf technisch eingebunden werden können, liegen nicht vor. Auch werden keine Aussagen gemacht, welche Informationssysteme von DPMS unterstützt werden.

Bewertung von Data-driven Process Structures

Die in den DDPS Ansatz enthaltenen Informationen können nicht von allen Prozessanwendern eingesehen werden.

Änderungen an den Prozesselementen oder am Prozessverlauf können durch die Prozessanwender durchgeführt werden. Dabei müssen die Prozessanwender Programmcodes schreiben, was die Kenntnis von Programmiersprachen bzw. die Zugriffsmöglichkeit auf IT-Fachpersonal voraussetzt. Ein Kontrollsystem, das die Änderungen dahingehend überprüft, ob Prozessanwender Prozesselemente entsprechend dem Geschäftsziel ändern, existiert ebenfalls nicht. Daher müssen die Prozessanwender ihre Modifikationen sorgfältig

durchdenken, bevor diese realisiert werden. Eine automatische Erstellung von neuen Prozesselementinstanzen aus alten ist möglich. Das gilt aber nicht für die Datenobjekte.

Die Erweiterung der Informationen eines Datenobjektes mittels einer Datei, wird nicht beschrieben. Ebenfalls nicht beschrieben wird die Vorgehensweise beim importieren und / oder exportieren von Informationen. Ein Sicherheitssystem wird von keinem der Ansätze beschrieben. Ebenfalls fehlen Informationen zu Systemen zur Fehlererkennung und / oder Fehlerkorrektur. Ein paralleler Zugriff auf die Datenobjekte wird nicht beschrieben. Somit ist eine parallele Arbeit an den Informationen vermutlich nicht möglich. Anwendungsfunktionen oder Prozessanwender können nur sequentiell auf die Informationen zugreifen. Der Zugriff wird mittels Benutzeroberfläche durchgeführt. Eine anpassbare Benutzeroberfläche für die Prozessanwender liegt allerdings nicht vor.

7.3.3 Lebenszyklus

Ein besonderes Augenmerk bei dem Vergleich lag auf dem Lebenszyklus der Daten. Ein Lebenszyklus kann formal definiert werden, um beispielsweise die Konsistenz oder Leistung zu überprüfen [GS07] [RKG07b]. Alle Ansätze gehen zu einem gewissen Grad auf den Lebenszyklus ein.

Der Lebenszyklus wird von den Ansätzen ACBPM und DDPS als endlichen Automaten bzw. Petri-Netz dargestellt, formal beschrieben werden können. Diese Möglichkeit wird von den Ansätzen allerdings noch nicht realisiert. In den neuesten Veröffentlichungen zu ACBPM wird beschrieben, dass Aktivitäten den Lebenszyklus formal zu beschreiben in Angriff genommen wurden.

Die Ansätze ACM und DPMS beschreiben den Zustand einer Prozessinstanz mittels dem Zustand der gespeicherten Daten in den Attributen. DPMS beschreibt zwar, dass ein Lebenszyklus existiert, und verweist auf ein weiteres Forschungsvorhaben *Data Flow Correctness in Adaptive Workflow Systems* [RM09] am Institut für Datenbanksystemen und Informationssystemen der Universität Ulm. Durch die Zusammenführung der beiden wissenschaftlichen Veröffentlichungen kann für DPMS ein Lebenszyklus definiert werden.

Die betrachteten Ansätze würden grundsätzlich die Möglichkeit bieten einen Lebenszyklus formal zu beschreiben, wie nach [GS07] [RKG07b]. Allerdings muss bei BPMN der Lebenszyklus mittels des Sequenzflusses definiert werden, weil die Datenobjekte nicht den Prozessablauf steuern. BPMN selbst beschreibt keinen Lebenszyklus.

7.4 Einordnung der Ansätze ins Paradigmadreieck

In diesem Kapitel werden die einzelnen Ansätze basierend auf den Vergleichskriterien bewertet. Die Ergebnisse sind zum Zwecke der besseren, Übersicht in der Tabelle 7.53 bis 7.60 zusammengefasst. Die Bedeutung der Symbole werden in Tabelle 7.52 beschrieben. In Kapitel 3 wurden die unterschiedlichen Paradigmen beschrieben. Um die einzelnen Ansätze besser miteinander vergleichen zu können, wurde im Abschnitt 3.7 ein Spannungsdreieck für deren jeweilige Einordnung beschrieben. In dieses Spannungsdreieck werden nachfolgend die sechs Ansätze zur Prozessunterstützung eingetragen. Ausgangspunkt ist immer das inhaltsbasierte Paradigma.

| Symbol | Beschreibung |
|--------|--|
| + | Diese Eigenschaft wird vom Ansatz erfüllt. |
| - | Diese Eigenschaft wird vom Ansatz nicht erfüllt. |
| ! | Durch Umschreibung einer Eigenschaft wird angenommen, dass der Ansatz die Eigenschaft besitzt. (Implizite Annahme) |
| O | Es liegen zu dieser Eigenschaft keine Informationen vor. |

Tabelle 7.52: Symbolbeschreibung für den tabellarischen Vergleich

| Vergleichskriterium / Ansatz | α -Flow | BPMN | ACM | ACBPM | DPMS | DDPS |
|--|----------------|------|-----|-------|------|------|
| Modellierungsmethoden | O | + | + | O | O | O |
| Explizite Definition von Lebensphasen in einem Prozess | O | ! | ! | ! | + | + |
| Formale Beschreibung des Lebenszyklus | - | - | - | - | - | - |
| Analyse der Prozesse | O | O | + | + | O | O |
| Datenobjekt basierte Datenspeicherung | + | + | - | + | + | + |
| Rekord basierte Datenspeicherung | - | - | + | O | O | O |
| Zusammenführung von Nutzdaten, Verwaltungsdaten, Verwaltungssystem | + | - | + | O | O | O |
| Kommunikationsmittel für soziale Interaktion | + | O | + | O | O | O |
| Natürliche Sprache als Prozess-Beschreibungssprache | O | - | + | O | O | O |
| Anpassbare Benutzeroberfläche | O | O | + | O | O | O |

Tabelle 7.53: Zusammenfassung des gesamtmethodischen Vergleichs

| Vergleichskriterium / Ansatz | α -Flow | BPMN | ACM | ACBPM | DPMS | DDPS |
|--|----------------|------|-----|-------|------|------|
| Benutzerinteraktion | + | + | + | + | + | + |
| Ereignisbehandlung | ! | + | ! | + | O | O |
| Online / Offline Modus | O | O | + | O | O | O |
| Echtzeitfähigkeit von Anwendungsfunktionen | O | O | O | O | O | O |
| Abhängigkeiten zwischen Anwendungsfunktion | - | ! | ! | ! | ! | ! |
| Granularität von Anwendungsfunktionen | - | + | O | O | + | O |
| Wiederverwendbarkeit von Prozesselementen | + | + | + | + | O | + |
| Langzeiteinsatzfähigkeit | + | O | O | O | O | O |

Tabelle 7.54: Zusammenfassung des funktionalen Vergleichs

| Vergleichskriterium / Ansatz | α -Flow | BPMN | ACM | ACBPM | DPMS | DDPS |
|---|----------------|------|-----|-------|------|------|
| Aktive Datenobjekte | + | - | - | O | - | - |
| Inhalt: Nutz- und Verwaltungsdaten | + | - | + | + | + | + |
| Identifikation von Daten | ! | O | O | + | + | O |
| Speicherung der Prozessverlaufsinformationen in den Daten | + | - | + | + | + | + |
| Anzahl der Datenobjekte für einen Anwendungsfall | - | + | - | + | + | + |
| Einheit der Validierung | + | O | O | O | O | O |

Tabelle 7.55: Zusammenfassung des informativen Vergleichs

| Vergleichskriterium / Ansatz | α -Flow | BPMN | ACM | ACBPM | DPMS | DDPS |
|--|----------------|------|-----|-------|------|------|
| Steuerung des Prozess durch Daten | + | - | + | + | + | + |
| Steuerung des Prozesses durch Aktivitäten | + | + | + | - | - | + |
| Datenlebenszyklus | O | O | O | + | O | + |
| Beschreibung von Übergangsbedingungen für Anwendungsfunktion | + | + | + | + | + | + |
| Definiertes Ende einer Prozessinstanz | - | + | - | ! | ! | ! |
| Echtzeitfähigkeit von Prozessen | O | O | O | O | O | O |

Tabelle 7.56: Zusammenfassung des dynamischen Vergleichs

| Vergleichskriterium / Ansatz | α -Flow | BPMN | ACM | ACBPM | DPMS | DDPS |
|---|----------------|------|-----|-------|------|------|
| Transparenz | + | O | + | O | + | O |
| Inter-Institutionelle Prozessausführung | + | + | O | O | O | O |

Tabelle 7.57: Zusammenfassung des integrativen Vergleichs

| Vergleichskriterium / Ansatz | α -Flow | BPMN | ACM | ACBPM | DPMS | DDPS |
|------------------------------------|----------------|------|-----|-------|------|------|
| Kommunikationsnetz | O | O | O | O | O | O |
| Spezialgeräte | O | O | + | O | O | O |
| Kenntnis von der Systemarchitektur | - | O | - | O | O | O |

Tabelle 7.58: Zusammenfassung des technische-operativen Vergleichs

| Vergleichskriterium / Ansatz | α -Flow | BPMN | ACM | ACBPM | DPMS | DDPS |
|--|----------------|------|-----|-------|------|------|
| Regelsystem | + | - | + | + | O | + |
| Sicherheitssysteme | + | + | + | O | + | O |
| Benutzerverwaltung | + | - | ! | O | + | O |
| Protokollierungssystem | ! | O | + | + | O | O |
| Ein- und Ausgabeverwaltung | + | O | + | + | O | O |
| Anmerkungs-system | O | + | ! | O | O | O |
| Lokalisierung von Anwendungsfunktionen | - | O | O | O | O | O |
| Koordinierung von Daten | + | O | - | O | O | O |
| Versionsverwaltung | + | O | + | O | O | O |
| Fehlererkennung | O | O | O | O | O | O |
| Fehlerkorrektur | O | O | O | O | O | O |
| Verklemmungsvermeidungssystem | - | O | - | O | O | O |

Tabelle 7.59: Zusammenfassung des software-operativen Vergleichs

| Vergleichskriterium / Ansatz | α -Flow | BPMN | ACM | ACBPM | DPMS | DDPS |
|--|----------------|------|-----|-------|------|------|
| Anpassungsfähigkeit des Datenschemas zur Laufzeit | + | O | ! | + | + | + |
| Anpassungsfähigkeit der Anwendungsfunktionen zur Laufzeit | - | O | + | + | + | + |
| Anpassungsfähigkeit der Anzahl von Anwendungsfunktionen zur Laufzeit | - | O | + | + | + | + |
| Anpassungsfähigkeit der Benutzerverwaltung zur Laufzeit | + | O | + | O | O | O |
| Anpassungsfähigkeit des Regelsystems zur Laufzeit | O | O | + | + | O | + |
| Anpassungsfähigkeit des Routing-systems zur Laufzeit | O | O | O | O | O | O |
| Anpassungsfähigkeit des Protokollierungssystem zur Laufzeit | O | O | O | O | O | O |

Tabelle 7.60: Zusammenfassung des evolutionären Vergleichs

α -Flow

α -Flow lässt keine externen Anwendungsfunktionen auf seine Daten zugreifen. Der Prozesslauf in α -Flow basiert nicht auf modellierten Aktivitäten, sondern im Verlauf einer α -Episode definieren neu angeforderte α -Cards, die als Platzhalter in der Koordinierungsliste angelegt werden, den Prozessfortschritt. Dieser wird als sukzessive Abarbeitung von angeforderten Inhalts- α -Cards interpretiert. Daher kann der Verlauf eines Prozess in jedem Schritt neudefiniert werden. Aus diesem Grund wird α -Flow nicht weit in Richtung aktivitätsbasiertes Paradigma eingeordnet. α -Flow unterstützt allerdings kooperatives arbeiten auf lose gekoppelten Informationssystemen und ermöglicht über eine Kommunikationssteuerung das gemeinsame arbeiten. Durch die gemeinsame Arbeit an den Daten entsteht eine digitale Gruppe, die eine eigene soziale Interaktion entwickeln kann. Aus diesen Grund wird α -Flow in Richtung Groupware Paradigma verschoben.

Business Process Modelling and Notation

BPMN unterstützt ebenfalls die parallele Abarbeitung eines Prozess. Allerdings sind die Arbeitsschritte der Prozessteilnehmer voneinander getrennt. Stattdessen wird der Sequenzfluss der Aktivitäten in den Vordergrund gestellt. Somit bewegt sich BPMN in Richtung aktionsbasiertes Paradigma und weg vom Groupware basierten Paradigma.

Adaptive Case Management

ACM unterstützt wie α -Flow die kooperative Arbeit und die Kommunikation zwischen den einzelnen Nutzern eines Prozesses. Als Anforderung wird beschrieben Wikis in den Prozessverlauf einzuziehen. Mit der Hilfe von Wikis könnte gemeinsam auf den gleichen Daten gearbeitet werden. Auch wird die personenbezogene Adaption der Benutzeroberfläche, was die Benutzerfreundlichkeit erhöht, unterstützt. Bei ACM übernimmt das Regelsystem die Steuerung des Prozesses. Somit ist der Prozessablauf, solange kein Prozessanwender eingreift, festgelegt. Aus diesem Grund sind bei ACM die Eigenschaften des Groupware Paradigmas und des aktivitätsbasierten Paradigma als gleichwertig anzusehen.

Artifact-Centric Business Process Models, Datenorientiertes Prozess-Management-System und Data-driven Process Structures

Keiner der drei Ansätze bietet eine Unterstützung für die gemeinsame Arbeit auf den gleichen Daten an. Ein Austausch über eine Mehrwegkommunikation ist dadurch nicht mög-

lich. Aus diesem Grund entfernen sich die drei Ansätze von den Groupware Paradigmen-Eigenschaften. Sie unterscheiden sich allerdings in Bezug auf die Aktivitäten.

- In ACBPM werden Datenobjekte mit Anwendungsfunktionen mittels Assoziationen verbunden. Dies stellt einen Ablaufplan zwischen den Aktivitäten dar, der geänderten Rahmenbedingungen angepasst werden kann.
- DPMS synchronisiert die Anwendungsfunktionen, sowohl synchron als auch asynchron. Dies führt zu Aggregation von mehreren Unteranwendungsfunktionen durch eine Oberanwendungsfunktion. Der Ablauf wird somit von den übergeordneten Datenobjekten gesteuert.
- In DDPS werden Hierarchien von Anwendungsfunktionen aufgebaut, die mittels Regeln auf die Datenobjekte angewendet werden. Zusätzlich wird die Prozesssteuerung mit Zuständen ausgestattet, die wiederum eine eigene Steuerungslogik mitbringen. Somit erhalten die Anwendungsfunktionen eine weitere Steuerungslogik, zusätzlich zu denen in dem Datenobjekt. Dies ist damit begründet, dass die Entwicklung von Daten und Aktivitäten getrennt erfolgt.

DPMS und ACBPM ordnen Daten die Aktivitäten mittels der Datenzustände zu und besitzen deshalb ähnliche Eigenschaften in Bezug auf das aktivitätsbasierte Paradigma. Aus diesem Grund ist DDPS näher an dem aktivitätsbasierten Paradigma platziert als DPMS.

Inhaltsbasiertes Paradigma: Umsetzung bei den Ansätzen zur Prozessunterstützung

Jeder der Ansätze basiert mehr oder weniger auf dem inhaltsbasierten Paradigma. Am wenigsten ausgeprägt ist dieses bei BPMN. Bei BPMN wird keine Steuerungsinformation in den Datenobjekten gespeichert. Allerdings können die Gateways anhand der Information in den Datenobjekten Entscheidungen treffen. Der Sequenzfluss wird unabhängig von den Informationen definiert. ACBPM und DPMS steuern den Prozessablauf auf der Basis der Informationen aus den Datenobjekten. Alle Anwendungsfunktionen erfahren mittels der Datenobjektattribute, ob sie auf ein Datenobjekt zugreifen dürfen und welche Funktionen ausgeführt werden können. Somit beinhalten ACBPM und auch DPMS vor allem Eigenschaften des inhaltsbasierten Paradigmas. Allerdings erfolgt durch die Zuordnung von Anwendungsfunktionen zu den Datenzuständen eine Ablaufsteuerung durch die Aktivitäten eines Prozesses. Im Gegensatz dazu existiert in DDPS zusätzlich

zu der Steuerung des Prozessablaufes durch die Datenobjekten eine Steuerung durch die Aktivitäten. Dies stellt einen Mittelweg zwischen aktionsbasiertem und inhaltsbasiertem Paradigma dar. ACM benutzt ein DBS anstatt eines Datenobjektes. Allerdings wird bei ACM die Steuerung vom Regelsystem übernommen, welches die Reihenfolge der Abarbeitung zum Erreichen des vorgegebenen Geschäftsziels vorgibt. Die Regeln basieren auf den in den Daten abgespeicherten Informationen, die von Prozessanwender verändert werden können. Hierbei besteht nur ein indirekter Zusammenhang mit den Nutzinformationen. Somit existiert eine Ausprägung hin zum aktivitätsbasierten Paradigma. In α -Flow stellen die Informationen der Mittelpunkt der Ausführung dar. Für die Daten existieren nur vom Dokument ausgehende Funktionalitäten. Externe Anwendungsfunktionen existieren nicht. Somit erfüllen die Eigenschaften von α -Flow nicht nur die Ausprägung des inhaltsbasierten sondern auch die des aktivitätsbasierten Paradigmas.

Die Ausführung dieses Abschnittes sind in der Grafik 7.1 visualisiert.

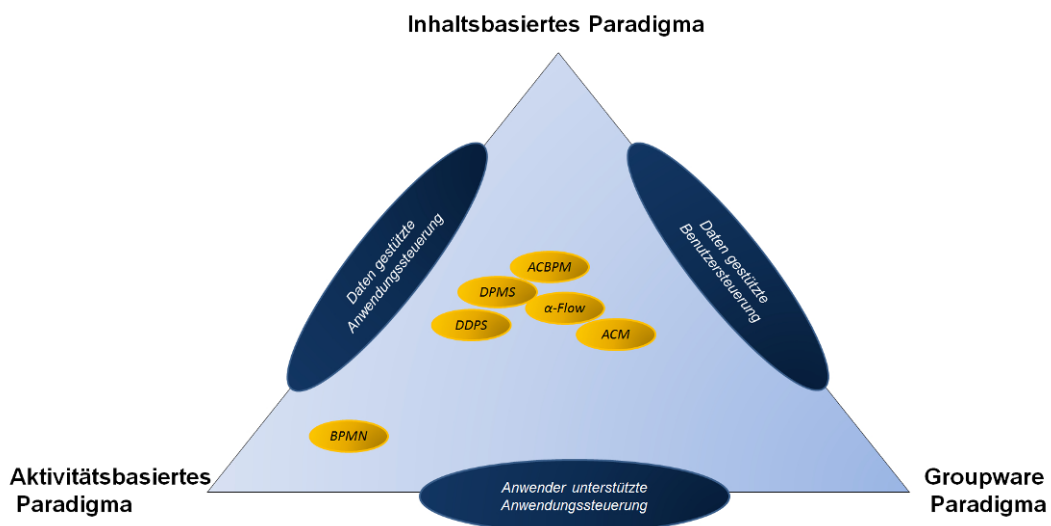


Bild 7.1: Einordnung der Ansätze in die Paradigmen

7.5 Erfahrungswerte aus dem Vergleich

In diesem Kapitel wurden die sechs ausgewählten Ansätze zur Prozessunterstützung miteinander verglichen. Bei dem Vergleich wurden die unterschiedlichen Ausprägungen des inhaltsbasierten Paradigma ersichtlich. α -Flow, ACM, ACBPM, DPMS und DDPS stellen dabei diejenigen Ansätze dar, die den Prozessverlauf durch ihre Daten steuern. Im Unterschied hierzu sind die Daten im Ansatz BPMN für die Steuerung des Prozesses zu vernachlässigen.

Für den allgemeinen Einsatz sind die Ansätze unterschiedliche geeignet. Durch die mögliche soziale Interaktion erlaubt α -Flow das parallele arbeiten in einem Prozess und fördert dadurch die Entwicklung einer Gruppendynamik. Ein weiterer Vorteil von α -Flow ist, dass alle zur Verfügung stehenden Informationen integriert und dem jeweiligen Prozessanwender zur Verfügung gestellt werden. Diese können, ohne dass eine vorherige Softwareinstallation notwendig wird, auf lose gekoppelten Informationssystemen spontan an einem Prozess teilnehmen. Alle Prozessabläufe werden dynamisch durch die Daten vorgeben. Die Datenstruktur kann sich während der Ausführung den Gegebenheiten der Außenwelt anpassen. α -Flow erlaubt auch die inter-institutionelle Kommunikation. Durch den einfachen Austausch des α -Docs, das alle Informationen über einen Patienten beinhaltet, kann ein Fall von einer Institution auf eine andere übertragen werden. Ein α -Doc stellt somit eine Fallakte dar.

Der BPMN Ansatz repräsentiert einen Produktions-Workflow-Ansatz. Der Workflow wird durch die einzelnen Arbeitsschritte gegliedert, die zum Erreichen eines Geschäftsziels benötigt werden. Dabei wird ein besonderes Augenmerk auf die Automatisierung der einzelnen Aktivitäten gelegt. Benutzerinteraktionen spielen nur eine untergeordnete Rolle, werden aber durch den Ansatz explizit unterstützt. Den Daten im Workflow-Verlauf wird bei dem BPMN Ansatz keine besondere Aufmerksamkeit geschenkt. Es kann ein Datenfluss neben dem Sequenzfluss der Anwendungsfunktionen modelliert werden. Allerdings existiert keine explizite Strukturierung des Datenflusses. Auch die Entscheidung der Gateways, die anhand der Daten durchgeführt werden können, wird nur oberflächlich von BPMN behandelt. BPMN ist ein Workflow-Ansatz, der nicht für eine bestimmte Umgebung entwickelt worden ist. Daher kann er in jedem Umfeld eingesetzt werden. BPMN unterstützt die inter-institutionelle Kommunikation durch die grafische Modellierungsoberfläche. Dabei können andere Institutionen Nachrichten oder Ereignisse erhalten.

Der ACM Ansatz unterscheidet sich von den anderen Ansätzen durch die Verwendung eines zentralen DBSs. Dies stellt auch gleichzeitig den Flaschenhals von ACM dar, wenn der Ansatz einmal global eingesetzt werden soll. Allerdings für den innerbetrieblichen Einsatz ist der ACM Ansatz ohne Client-Serverprobleme einsetzbar. ACM beschreibt eine Struktur, bei der sich die Clients ad-hoc verbinden können. Allerdings muss die Software und die Verbindung zum zentralen DBS bekannt sein. Die Einbindung unterschiedlicher, von dem Kommunikationsnetz trennbarer, mobiler Informationssysteme ist möglich. Die Steuerung des Prozessverlaufs wird mittels eines Regelsystems beschrieben. Die Regeln werden mittels Bedingungen, die in Daten vorliegen, definiert. Die soziale

Interaktionsmöglichkeit ist leicht durch Web 2.0-Anwendungen ausgeprägt. ACM stellt für einen innerbetrieblichen Ansatz eine gute Lösung dar. Alle Informationen gelangen zu den Prozessanwendern, die sie auch benötigen. Diese Eigenschaft wird durch die Anpassungsfähigkeit der einzelnen Prozesse unterstützt.

Die Ansätze ACBPM, DPMS und DDPS beschreiben alle einen Ansatz, der mittels einem oder mehrerer passiver Datenobjekte gesteuert wird. DDPS unterscheidet sich dabei von den anderen beiden Ansätzen, durch die zusätzliche Implementierung von Steuerungsinformation (Prozessstufen) in den Anwendungsfunktionen. Bei allen drei Ansätzen lässt sich sowohl der Prozessverlauf erweitern, als auch die Anzahl der Attribute der Datenobjekte. Die Möglichkeit der flexiblen Teilnahme am Prozessablauf wird durch die direkte Modellierung des Verlaufes eingeschränkt. Spontan kann keiner der drei Ansätze neue Anwendungsfunktionen integrieren. Dafür müssen vorab immer das Regelsystem oder die Datenobjekte angepasst werden.

Im Abschnitt 1.3, Abgrenzung, wurden die Grenzen der betrachteten Ansätze beschrieben. Nach dem Vergleich in diesem Kapitel können die einzelnen Ansätze nun in die in Abschnitt 1.3 beschriebene Grafik eingefügt werden. BPMN kann dabei den Produktions-Workflows gleichgesetzt werden, da der Sequenzfluss den explizit vorgeben Verlauf definiert. ACBPM, DPMS und DDPS ordnen sich bei den explizit strukturieren Ansätzen zur Prozessunterstützung ein. ACBPM und DPMS steuern den Prozess mittels der Daten. Zusätzlich ordnen sie den Datenzuständen Aktivitäten zu. Daraus ergibt sich eine von Daten beschriebene Ablaufsteuerung von Aktivitäten. ACM, der Ansatz, der sich von CH ableiten lässt, liegt an der Grenze von ad-hoc und impliziert definierten Ansätzen zur Prozessunterstützung. Benutzer können dynamisch, während der Laufzeit, den Prozessablauf definieren und eine Prozessinstanz mit leerem Verlauf beginnen. Ein Verlauf wird aber durch den ersten Benutzer implizit vorgeben. Die Daten in Zusammenhang mit dem Regelsystem definieren den Fortschritt eines Prozesses. α -Flow erfüllt eine ad-hoc Ausführung eines Prozesses. Der nächste Prozessschritt ist unbekannt. Zusätzlich wird der Prozessverlauf mittels der Daten im Datenobjekt gesteuert und die medizinischen Behandlungen (Aktivitäten) bestimmen die Daten. Die Einordnung der Ansätze wird in Abbildung 7.2 dargestellt.

Die Ansätze wurden in diesem Kapitel nur auf der Basis des durchgeführten Vergleiches und der Paradigmen beschrieben. Im nächsten Kapitel folgt eine Einordnung der Ansätze entsprechend deren möglichen Einsatzes im medizinischen Umfeld.

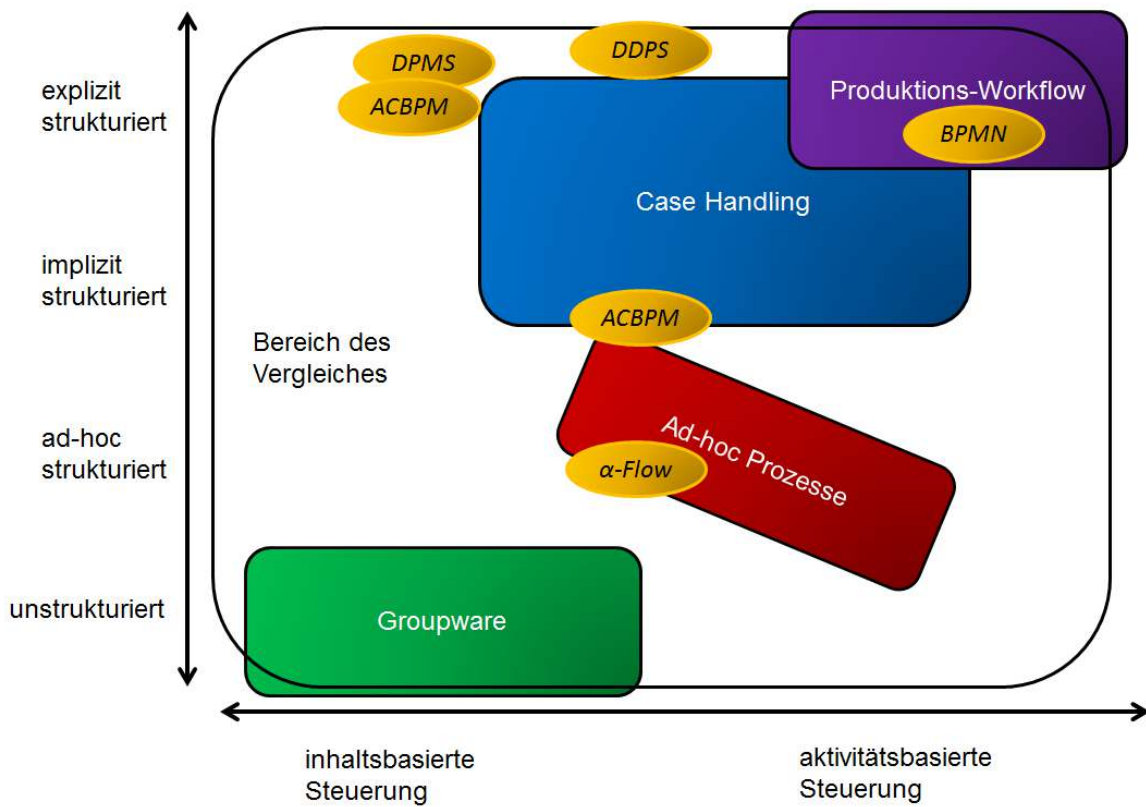


Bild 7.2: Steuerung und Struktur Positionierung der Ansätze nach [RRA03]

8 Evaluation

Im Kapitel 7 wurden die Ansätze zur Prozessunterstützung miteinander verglichen, und die Vor- und Nachteile der einzelnen Ansätze anschließend diskutiert. Im nachfolgenden Kapitel werden nun die Ergebnisse dieses Vergleichs bewertet und es wird beschrieben, in wie weit die Ansätze die Anforderungen im medizinischen Umfeld erfüllen. Hierbei wird auch thematisiert, wie der Lebenszyklus der einzelnen Ansätze definiert ist (Abschnitt 8). Abschließend werden im Abschnitt 8.2 aus der Bewertung der einzelnen Ansätze resultierende offene Fragen angesprochen und Lösungsvorschläge unterbreitet.

8.1 Evaluation der Anforderungen

In α -Flow wird der Behandlungsverlauf als eine Fallakte (α -Doc) interpretiert, die von den inter-institutionellen Anwendern sukzessiv ausgefüllt wird. Die Datenobjekte von α -Flow (α -Cards) können daher beliebig, auch mittels medizinischer Berichte (Dateien), erweitert werden. Alle relevanten Informationen über einen Patienten werden in einem Datenobjekt gespeichert. Somit kann eine Fragmentierung der Daten auf unterschiedliche Informationssysteme vermieden werden. Dies ist dann hilfreich, wenn beispielsweise durch Stress vergessen wurde, bestimmte Informationen weiterzuleiten.

Da der Kommunikationsverlauf nicht spezifiziert ist, beschränkt sich der Informationsaustausch nicht nur auf eine Institution, sondern kann auch auf räumlich getrennte ausgedehnt werden. Durch einen installationslosen Betrieb von α -Flow kann jeder Anwender an einen Fall partizipieren.

Der mögliche Einsatz mobiler Datenverarbeitungssysteme vermeidet das Zurückgreifen auf andere Medien (z.B. Papier) und reduziert den Aufwand dahingehend, die gesammelten Informationen später in ein Informationssystem zu integrieren. Ein weiterer nicht unerheblicher Vorteil ist die Vermeidung von Übertragungsfehlern

Eine α -Episode besitzt kein definiertes Ende. Ein Patient kann, basierend auf einer bereits existierenden Prozessinstanz, wiederholt behandelt werden, denn seine Vorgeschichte ist im α -Doc gespeichert. Als Ende einer α -Episode kann der Zustand definiert werden, wenn keine weiteren Informationen mehr zur Zielerfüllung benötigt werden

BPMN stellt nur eine Modellierungssprache dar, deren Umsetzung von dem jeweiligen Übersetzungstool abhängig ist. Dadurch werden Systemeigenschaften auf tieferen Ebenen implementiert, die von dem verwendeten Tool abhängig sind. Es können verschiedene WfMS verwendet werden, wodurch sich BPMN an neue medizinische Systeme anpassen lässt. Zusätzlich muss beachtet werden, dass der Prozess von den Aktivitäten gesteuert wird.

Im medizinischen Umfeld lässt sich der Ablauf in den meisten Fällen nicht vorab planen, da nicht alle Eventualitäten vorausgesehen werden können. Somit müsste, abhängig vom Behandlungsverlauf, dieser gegebenenfalls immer wieder neu definiert werden. Die Fähigkeiten von Ad-Hoc-Prozessen in BPMN zu implementieren, werden diesen Anforderungen nicht gerecht. Dies schränkt die Verwendung von BPMN im medizinischen Bereich erheblich ein. Vorteilhaft ist die mögliche grafische Modellierung der Behandlung, was die Planungsphase für das medizinische Personal erleichtert.

ACM stellt einen weiteren flexiblen und adaptiven Ansatz zur Prozessunterstützung dar. Statt eines Datenobjektes besitzt ACM ein zentrales DBS, welches alle Informationen speichert. Zusätzlich besteht die Möglichkeit, mobile Geräte wie Pads und Laptops in den Prozessverlauf zu integrieren, wenn sie mit den Betriebssystemen Microsoft Windows, Linux oder MacOS ausgestattet sind. Eine Einbindung dieser Informationssysteme ist trotz der möglichen Trennung vom Kommunikationsnetz realisierbar. Die mobilen Informationssysteme können bei der Visite eingesetzt werden, bei der normalerweise keine lokalen Datenverarbeitungssysteme zur Verfügung stehen.

Bei der Aufnahme von Daten werden mittels Optical Character Recognition (OCR) Dokumente eingelesen und klassifiziert, um die Relevanz für den vorliegenden Fall zu definieren. Der Nachteil von ACM ist das zentrale DBS, welches sich als Flaschenhals in der Verbindungsstruktur erweist. Die Anwendung zielt primär auf den inter-institutionellen Bereich und ist weniger gut für den Einsatz auf Landes- bzw. Bundes-Ebene geeignet.

Auf der anderen Seite setzt ein DBS bekannte Praktiken bei der Modellierung ein, durch die Transaktionsverwaltung gewährleistet es das parallele Lesen und Schreiben auf den Daten. Dabei garantiert das Atomicity, Consistency, Isolation, Durability (ACID)-Konzept die Sicherheit der gespeicherten medizinischen Daten. Wie allerdings die dynamische Veränderung bei dem Datenschema oder bei der Transaktionsverwaltung umzusetzen ist, wird nicht beschrieben. Die mögliche Anpassung der Benutzeroberfläche repräsentiert einen Vorteil dieses Systems.

Der Lebenszyklus von ACM endet nur indirekt. Alle Elemente einer Prozessinstanz werden archiviert und können als Vorlage dem Archiv entnommen und für eine neue

Prozessinstanz verwendet werden. Das Archiv und alle weiteren Elemente einer Prozessinstanz können von einem Prozessteilnehmer eingesehen werden. Dazu zählen auch die Verwaltungsinformation; beispielsweise wer Information hinzugefügt hat und welche Informationen wann und wie verändert wurden. Dies spiegelt die tägliche Arbeitsweise im medizinischen Umfeld wieder.

Die Ansätze ACBPM, DPMS und DDPS stellen vergleichbare Ansätze dar. Ihre Prozessinstanzen besitzen mehrere Datenobjektinstanzen, die die Aktivitäten steuern. Bis auf DDPS befinden sich die Steuerungsinformationen für eine Prozessinstanz in den Datenobjekten. Bei DDPS existiert noch eine zusätzliche Logik in den Anwendungsfunktionen.

Für die oben erwähnten Ansätze beschreibt der Lebenszyklus einen Graphen, bei dem die Knoten die Zustände der Datenobjekte darstellen. Dabei repräsentiert jeder Knoten einen stabilen Datenzustand, an dem Prozessteilnehmer ihre Entscheidung treffen. Die Übergänge (Kanten) werden durch eine Diagnose oder Therapie beschrieben. Ein solcher Lebenszyklus besitzt kein definiertes Ende. Daher müssen die Ansätze explizit erweitert werden, wenn neue Informationen anfallen, die eine Fortführung des Prozesses erfordern.

Während der laufenden Prozessinstanz kann eine Behandlung immer weiter geführt werden, da die Möglichkeit besteht, die Datenobjekte zu erweitern. Des Weiteren können mittels eines Regelsystems die Behandlungsreihenfolge bestimmt oder die Attributwerte überwacht werden.

Ein Sicherheitskonzept zum Schutz der persönlichen Daten ist für BPMN, ACBPM, DPMS und DDPS nicht vorhanden. Das parallele Arbeiten auf den gleichen Daten wird von BPMN, ACBPM, DPMS und DDPS ebenfalls nicht unterstützt. Somit kann nur eine medizinische Institution auf den Prozess und dessen Daten zugreifen. Des Weiteren wird von den drei Ansätzen auch keine Unterstützung der Prozessteilnehmer angeboten, wie beispielsweise eine modifizierbare Benutzeroberfläche oder die Möglichkeit, die Transparenz einer Prozessinstanz mittels natürlicher Sprache zu optimieren.

α -Flow und ACM sind zwei Ansätze, die für den medizinischen Einsatz als geeignet erscheinen. Sie ermöglichen ein paralleles Arbeiten, passen sich mittels Erweiterungen der Datenobjekte geänderten Rahmenbedingungen an und ermöglichen den sicheren Zugriff auf Daten. Allerdings beschreiben sie genau so wenig wie die anderen Ansätze nicht,

wie die Informationen über das Kommunikationsnetz transportiert werden, wenn ein Netzwerksplit¹ vorliegt.

α -Flow und BPMN bieten die Möglichkeit Institutionsübergreifend Informationen über einen Patienten auszutauschen. Bei BPMN wird eine eigene Infrastruktur beim Empfänger mittels eines installierten Clients benötigt. Währenddessen kann bei α -Flow auf die Installation einer Software (lose gekoppelte Informationssysteme) verzichtet werden. Die sofortige Ausführbarkeit eines α -Docs gewährleistet eine schnelle Adaption an unterschiedliche medizinische Einrichtungen.

Schwachpunkt von ACM ist das zentrale DBS, das noch einer Lösung bedarf. BPMN ist, bedingt durch seinen Aufbau, nur nach spezifischen Anpassungen im medizinischen Bereich einsetzbar.

ACBPM, DPMS und DDPS sind Ansätze, die in ihrer Grundfunktionalität für den Einsatz geeignet erscheinen, allerdings noch einiger Weiterentwicklungen bedürfen. Dies gilt vor allem für die folgenden Bereiche: Benutzerinteraktion, Sicherheit und Adaption an geänderte Rahmenbedingungen und nicht terminierende Prozessinstanzen.

BPMN kann die medizinischen Informationen nicht adäquat verarbeiten. Die Modellierung der Behandlungsschritte und die nur spärlich vorhandene Anpassung an neue Situationen erschwert einen Einsatz von BPMN im medizinischen Umfeld.

8.2 Ausblick

In dem vorigen Abschnitt 8 wurden die unterschiedlichen Ansätze zur Prozessunterstützung im Hinblick auf ihre mögliche Einsatzfähigkeit im medizinischen Umfeld hin evaluiert. Dabei zeigte sich, dass sowohl α -Flow als auch - bedingt - ACM Ansätze sind, die für den Einsatz im medizinischen Umfeld als geeignet erscheinen. Ein Schwachpunkt von ACM ist die Verwendung eines zentralen DBS. Dies umgeht α -Flow mittels eines ausführbaren Datendokuments. Ausführbare Datendokumente werden auch *aktive Dokumente* genannt. Sie erlauben eine flexible Ausführung, sind nicht an ein Informationssystem gebunden und interagieren mit den Benutzern [HM00], [KRW90].

Es existieren auch weitere Ansätze [DEH⁺00], [DEL⁺00], bei denen jedes Dokument mit *aktiven Eigenschaften* ausgestattet wird. Dadurch kann jedes Text-Dokument in ein aktives Datenobjekt verwandelt werden. Dies ermöglicht dessen Integration in ein

¹Trennung von einem Informationssystem vom Rest. Begründet durch Netzwerktrennung durch Fehler

bestehendes Softwareprodukt, das bereits von einer Institution verwendet wird. Die Methode der Umwandlung von Text-Dokumenten in ein aktives Datenobjekt und deren Kompatibilität mit bereits existierenden Datei-Formaten müsste aber überprüft werden. Ein weiterer Ansatz [WK05] beschreibt einen Dokument basierten Ansatz, der mit aktiven Formularen mittels einer DBS arbeitet.

Sowohl bei der Betrachtung als auch beim Vergleich der Lebenszyklen der Datenobjekte in den Ansätzen, wurde der Lebenszyklus selbst nicht formal definiert. Der Lebenszyklus wird während der Ausführung eines Prozesses für die Konsistenzsicherung der Übergänge der Zustände¹ eines Datenobjektes definiert. Die Zustandsänderungen werden mittels Anwendungsfunktionen vollzogen. [GS10] und [RKG07a] beschreiben Ansätze, die den Lebenszyklus formalisieren und die Konsistenz der Datenobjekte sichern. Für die hier betrachteten Ansätze können endliche Automaten oder Petri-Netze für die Einsatzszenarien erzeugt werden [UKM03], [WS02]. Die Möglichkeit der Erzeugung von endlichen Automaten behandelt weder die alternative Pfadmöglichkeiten noch die Anpassung des endlichen Automaten während der Laufzeit. Diese Eigenschaften müssen vor einer formalen Beschreibung weiter untersucht werden. Denn ohne die Möglichkeit alternative Pfade oder die Anpassung des Lebenszyklus in die Konsistenzprüfung eines Prozess einzubeziehen, führt die formale Definition eines Lebenszyklus zu keinem Mehrwert.

Eine weitere Möglichkeit ist es die Prozessabläufe, die nach einem bestimmten Hauptparadigma beschrieben werden, zu formalisieren, um daraus einen Lebenszyklus abzuleiten. Für den ACBPM Ansatz wurde ein Ansatz entwickelt, der es ermöglicht die Artefakte zu formalisieren [GBS07]. Dabei wird eine Ausprägung vom inhaltsbasierten Paradigma verwendet. In dieser Arbeit wurden drei Paradigmen vorgestellt. Für weitergehende Vergleiche wäre eine feinere Abstufung der verwendeten Paradigmen vorteilhaft. [BRNT97] beschreibt die Unterschiede zwischen einem Kontrollfluss und einem Datenfluss. In dieser Veröffentlichung thematisiert er auch, wie beide Flüsse vereint in einem adaptiven Ansatz beschrieben werden können.

Des Weiteren muss das Spannungsdreieck der drei Paradigmen untersucht werden. α -Flow und ACM beschreiben die Möglichkeit des Prozessteilnehmers entsprechend seinen Erfahrungen und seines Wissens den Prozessverlauf verändern zu können. Zusätzlich beschreiben ACBPM und DPMS, dass die Prozessanwender die Anpassung des Prozessverlaufs durchführen sollen, wenn spezifische Ereignisse eintreten. Damit soll

¹Der Zustand eines Datenobjektes ist eine Momentaufnahme der Werte der Attribute eines Datenobjektes

untersucht werden, wie sich das Wissen der Prozessteilnehmer auf den Prozessablauf auswirkt. Grafik 8.1 visualisiert diesen Aspekt und erweitert das Spannungsdreieck zu einem Viereck.

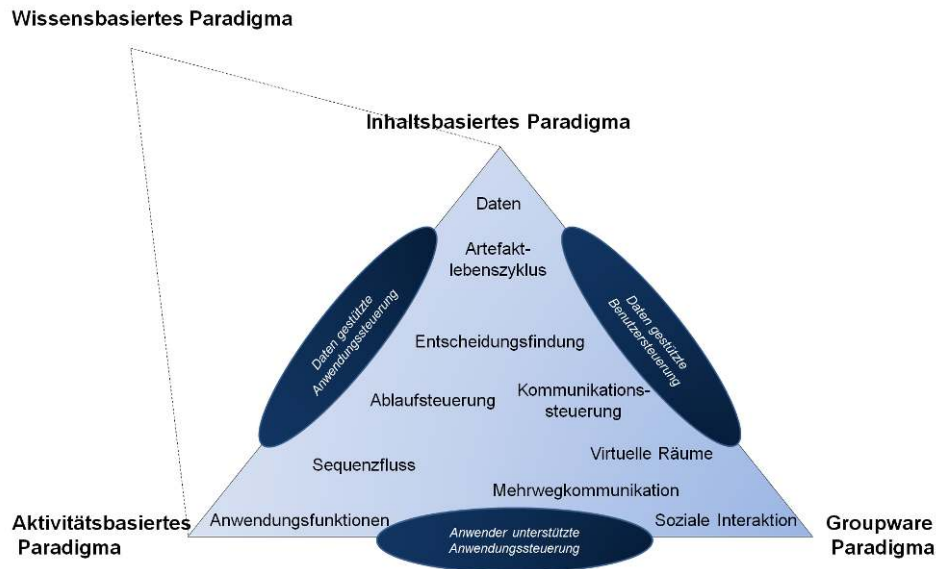


Bild 8.1: Erweiterung des Spannungsdreiecks zum Viereck

CSCW-Ansätze und Groupware-Ansätze sind Vorreiter in Bezug auf die Optimierung der kooperativen Arbeit und der Kommunikation zwischen den Prozessteilnehmern [EGR91]. Die Groupware-Systeme sind bereits mehrere Jahre im Einsatz und ihre Entwicklung befindet sich auf einem hohen Niveau. Dennoch, die Schwächen dieser Ansätze sind mit demjenigen von ACM, nämlich die Verwendung eines zentralen DBS, vergleichbar. Trotzdem sind die Eigenschaften der Groupware-Systeme im Kommunikationsbereich und der Gruppendynamik weiter entwickelt, als bei den hier betrachteten Ansätzen. Um deren Eigenschaften für andere Ansätze adaptieren zu können, sollten die existierenden Groupware-Ansätze klassifiziert und evaluiert [Ell00] werden.

Für den Einsatz im medizinischen Umfeld steht die Möglichkeit der dynamischen Anpassung an neue Prozessteilnehmer und Informationssysteme im Vordergrund. Diese Anforderungen unterstützen die Ad-Hoc Prozessunterstützungssysteme [SAMS01] [GHS95].

Mit ACM wurde die Kategorie von Case Handling (CH) betrachtet. Bei CH steht immer das Objekt im Vordergrund [AWG05b]. Im medizinischen Umfeld steht der Patient im Mittelpunkt. Aus diesem Grund sind die Eigenschaften von CH von Interesse, vor allem die Frage, wie die Informationen bezüglich des Objektes verarbeitet werden.

α -Flow und ACM beschreiben Sicherheitsmechanismen, die sie vor den Zugriffen von Dritten schützen. Bei BPMN ist dies nur mittels einer Erweiterung möglich, die nicht standardisiert ist und damit die automatische Abbildung eines Modells nicht mehr garantiert. Die anderen drei Ansätze ACBPM, DPMS und DDPS betrachten diesen Fakt gar nicht. Bei weiteren Recherchen sollte darauf Wert gelegt werden, dass Sicherheitssysteme in die Ansätze integriert sind, um die Datensicherheit zu gewährleisten. [AW08] beschreibt Mechanismen, die die Sicherheit in den Anwendungen garantieren, soweit das heutzutage möglich ist.

Neben den Zugriffsversuchen Dritter können Systemfehler die Ausführung eines Prozessunterstützungssystems stoppen. Um Systemfehler zu erkennen und gegebenenfalls zu vermeiden, werden Anwendungsfunktionen benötigt, die die Konsistenz der Prozesselemente sowohl auf der Anwendungsebene als auch auf der physischen Ebene gewährleisten.

Die betrachteten Ansätze beschreiben entweder die Terminierung einer Prozessinstanz nach der Ausführung oder sie werden einfach inaktiv. Bei zukünftigen Analysen sollte darauf Wert gelegt werden, dass Ansätze kein festgelegtes Ende besitzen, als auch die Möglichkeit offerieren, die gewonnenen Informationen zu archivieren.

Die Vergleichskriterien müssen feingranularer werden. Die in dieser Arbeit verwendeten Vergleichskriterien geben einen Überblick über den Aufbau, über die Kernfunktionalitäten und über das Verhalten eines Prozessunterstützungssystems. Viele der betrachteten Vergleichskriterien können feingranularer betrachtet werden. Im Vergleich wird beschrieben, welche Funktionalitäten die Ansätze unterstützen. Eine genaue Aussage, wie die Funktionalitäten technisch umgesetzt werden, wird offengelesen. Bei zukünftigen Analysen sollte von verfeinerten Vergleichskriterien ausgegangen werden, mit dem Ziel, die einzelnen Ansätze zur Prozessunterstützung technisch detaillierter zu bewerten. Werden die Vergleichskriterien feiner, dann muss die Einordnung der Prozessunterstützungssysteme 7.1 mit weiteren Merkmalen versehen werden, damit die Unterschiede zwischen den einzelnen Ansätzen detaillierter beschrieben werden können.

Zusätzlich sollte das inhaltsbasierte Paradigma verfeinert werden. In dieser Arbeit wurde das inhaltsbasierte Paradigma gleichgesetzt mit der Eigenschaft: Steuerung eines Prozessunterstützungssystems, dass durch seine gespeicherten Daten den Prozessverlauf beeinflusst. In der Literatur existieren detaillierte Beschreibungen, die sich unter dem Begriff inhaltsbasiertes Paradigma zusammenfassen lassen. Die Abbildung 8.2 beschreibt nach [KR10c] die unterschiedlichen Darstellungen bezüglich inhaltsbasierter Prozessunterstützungssysteme und stellt deren Zusammenhang dar.

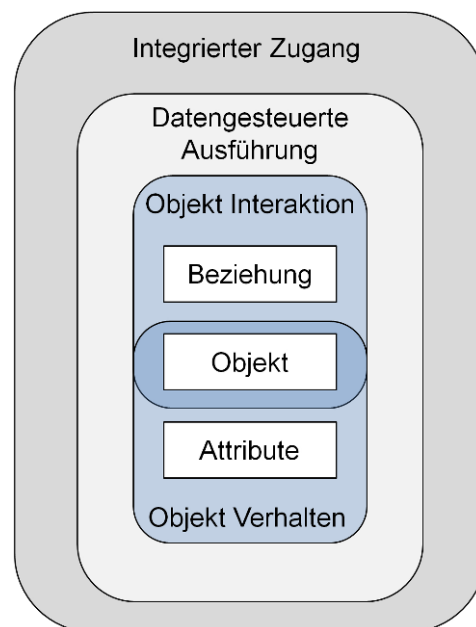


Bild 8.2: Hauptcharakteristika von inhaltsgesteuerten Prozesssystemen nach [KR10c]

9 Zusammenfassung

Der aktuelle, papierbasierte Informationsaustausch im medizinischen Umfeld mutet in der heutigen Zeit anachronistisch an. Aufgrund der in der Medizin gemachten Fortschritte sind heute oft verschiedene, teilweise unabhängige, Institutionen oder multi-organisatorische Teams an der Behandlung¹ eines Patienten beteiligt. Die hierbei anfallenden Informationen müssen effizient zwischen den Beteiligten ausgetauscht werden. Diese Aufgabe übernehmen Prozessunterstützungssysteme.

Ansätze zur Prozessunterstützung, wie Business Process Modelling and Notation (BPMN), basieren auf dem aktivitätsbasierten Paradigma. Dieses Paradigma stellt die Ausführungsreihenfolge der Aktivitäten in den Vordergrund. Die medizinische Behandlung bestehend aus den Diagnosen und den daraus abgeleiteten Therapien sind aber nicht im Voraus planbar.

Das Ziel einer Patientenbehandlung ist es dessen Gesundheit wieder herzustellen oder diese zu gewährleisten. Somit muss der Patient für die Informationssysteme abgebildet werden. Ein Patient definiert sich sowohl über seine personenbezogenen Daten als auch über Daten, die die Schritte der medizinischen Behandlung beschreiben. Somit sind die Informationen der Mittelpunkt eines Prozessunterstützungssystems. Wenn die Daten im Mittelpunkt der Ausführung stehen, stellt dies das inhaltsbasierte Paradigma dar.

Ein Prozessunterstützungssystem besteht aus Daten und Aktivitäten auf die mehrere Benutzer zugreifen und / oder diese veranlassen. Die Kommunikation und die kooperative Arbeit mit einer eigenen Gruppendynamik beschreiben Eigenschaften des Groupware Paradigmas. Die drei hier beschriebenen Paradigmen stehen nicht orthogonal zueinander, sondern gehen fließend ineinander über.

Wie mehrere Paradigmen parallel in einem Ansatz zur Prozessunterstützung interagieren können zeigt α -Flow². Im Mittelpunkt von α -Flow steht ein aktives Dokument. Jedes dieser Dokumente beinhaltet neben den medizinisch relevanten Daten zusätzlich

¹Behandlung: bestehend aus einem oder mehreren diagnostischen und therapeutischen Zyklen

²Vom Informatik Lehrstuhl 6 der Friedrich-Alexander-Universität Erlangen-Nürnberg entwickelter Prozessunterstützungssystem

Informationen zur Steuerung des Prozessablaufes. Bei der parallelen Arbeit an einer Prozessinstanz können die Prozessteilnehmer gleichzeitig auf die Daten zugreifen. Zusätzlich kann den Daten entnommen werden, wer diese generiert und / oder geändert hat. Mittels der eingegebenen Daten wird der nächste Schritt im Verlauf eines Prozesses gesteuert. Infolgedessen realisiert α -Flow signifikant das inhaltsbasierte Paradigma, aber zu gewissen Teilen auch das aktivitätsbasierte und das Groupware Paradigma.

Mit Hilfe einer Literaturrecherche wurden sechs aktuelle Ansätze zur Prozessunterstützung ausgewählt, die mit α -Flow vergleichbare Eigenschaften aufweisen und auf der Basis von Informationen gesteuert werden. Die Eigenschaften der Ansätze orientieren sich an dem anwendungsbasierten und dem Groupware Paradigma. Die Informationen stehen immer im Mittelpunkt von Datenobjekten oder von einem Datenbanksystem (DBS).

Jeder der betrachteten sechs Ansätze, α -Flow, Business Process Modelling and Notation (BPMN), Adaptive Case Management (ACM), Artifact-Centric Business Process Models (ACBPM), Datenorientiertes Prozess-Management-System (DPMS) Data-driven Process Structures (DDPS), beinhaltet Eigenschaften, die mehr oder weniger stark ausgeprägt auch von den anderen Ansätzen umgesetzt werden. Basierend auf den Eigenschaften der Ansätze wurden acht Aspekte definiert, die für einen Vergleich der unterschiedlichen Ansätze verwendet wurden. Diese beschreiben das grundlegende Verhalten, die Struktur, die Informationen, die Funktion, das Verhalten, die operativen Eigenschaften (Technisch und Software) und die Evolutionsfähigkeit der Ansätze.

Bei dem Vergleich der Ansätze zeigten sich deren unterschiedliche Ausprägungen nach den Paradigmen. Besonders im Bereich der Anpassungsfähigkeit an geänderte Rahmenbedingungen, der Gruppendynamik und der Ausrichtung der Ansätze unterschieden sich diese. In Bezug auf den Schutz transferierter Daten besitzen nur zwei Ansätze entsprechende Sicherheitsmechanismen. Eine lose gekoppelte Verbindung unterstützt nur α -Flow. Bei der Bewertung der Ansätze wurde auf den Lebenszyklus, der den Ablauf eines Prozesses und damit den Verlauf der Informationen beschreibt, ein besonderes Augenmerk gelegt. Mit dem Ziel die Konsistenz eines Prozesses vorab zu überprüfen, kann der Lebenszyklus formal beschrieben werden. Diese Möglichkeit wird aber von keinem Ansatz genutzt.

Für den Einsatz im medizinischen Umfeld eignen sich primär α -Flow und nur bedingt ACM. α -Flow stellt einen lose gekoppelten Ansatz dar, bei dem die benötigten Funktionalitäten in einem Datenobjekt integriert sind und kann dadurch auf unterschiedlichsten Informationssystemen ohne Installation verwendet werden. ACM verwendet ein zentrales DBS. Ein zentrales DBS skaliert nicht, wenn ACM überregional angewendet werden soll.

Die restlichen Ansätze besitzen Schwächen vor allem in dem Bereich der Anpassbarkeit an geänderte Rahmenbedingungen, eine Eigenschaft die vor allem im medizinischen Umfeld benötigt wird. Daher muss sich ein Ansatz zur Prozessunterstützung, das in diesem Bereich eingesetzt werden soll, an die aus der Patientenbehandlung resultierenden Daten adaptiv anpassen können. Aufgrund der inhärenten Annahme von α -Flow, dass Entscheidungen als eine Forderung nach weiteren Informationen betrachtet werden können, erfüllt er diese Forderung am besten.

A Begriffsdefinition

In dieser Arbeit werden verschiedene Begriffe, die vorab erst einmal definiert werden müssen, verwendet. Dieses erfolgt in diesem Kapitel:

A.1 Prozess

Summe der Tätigkeiten und Bearbeitungsschritte im Laufe der Leistungserbringung einer Organisation.

A.2 Workflow

Ein **Workflow** ist eine automatisierte, in Algorithmen gegossene, ablaufende Gesamtheit von Aktivitäten, die sich auf Teile eines gesamten Anwendungsprozesses bezieht [Ste98]

A.3 Organisatorische Sicht

Die organisatorische Sicht beschreibt die in den Prozessen arbeitenden Akteure und die von den Akteuren eingesetzten Ressourcen.

A.4 IT Sicht

Die IT Sicht beschreibt die Ausführung von Prozessen mit Hilfe von Informationssystemen.

A.5 Aktivität

Eine Aktivität ist die konkrete Durchführung von definierten Aktionen innerhalb eines Prozesses. Dabei kann die Durchführung digital umgesetzt werden

A.6 Anwendungsfunktion

IT-Sicht: Technische Umsetzung einer Aktivität. Eine Anwendungsfunktion stellt eine Funktion dar, die die eigentliche Objektbearbeitung innerhalb der Anwendung durchführt.

A.7 Objektlebenszyklus

Geben ist ein Objekttyp dessen Lebenszyklus aus einer begrenzten Menge an Zuständen besteht. Zu der Vielzahl von Zuständen zählen sowohl der Start- als auch der Endzustand. Um von einem Zustand zum anderen zu kommen, werden Übergänge benötigt. Ein Übergang beschreibt den Übergang von einem Zustand zum nächsten. Dabei können die Zustände als Knoten in einem endlichen Automaten angesehen werden, und die Übergänge als Kanten zwischen den Knoten.

A.8 Konfiguration

Prozessdaten besitzen eine endliche Menge an Attributen. Während der Ausführung sind die Attribute mit Werten belegt. Wird die Ausführung zu einem bestimmten Zeitpunkt angehalten, liegen bestimmte Werte in den Attributen der Daten eines Prozesses vor. Die Gesamtheit der Werte der Attribute der Daten zu einem bestimmten Zeitpunkt der Ausführung wird als Konfiguration bezeichnet.

B Internet

Kommunikationsmittelverteilung

Das Internet und die dort zur Verfügung stehenden Anwendungen werden von immer mehr Menschen genutzt. Bereits im Jahre 2001 waren in Deutschland mehr als ein Viertel der Bevölkerung im World Wide Web unterwegs (Abbildung B.1). In den folgenden Jahren stieg die Nutzung immer weiter an, so das im Frühjahr 2010 in Deutschland schon mehr 49 Millionen Bundesbürger online waren [AZ10].

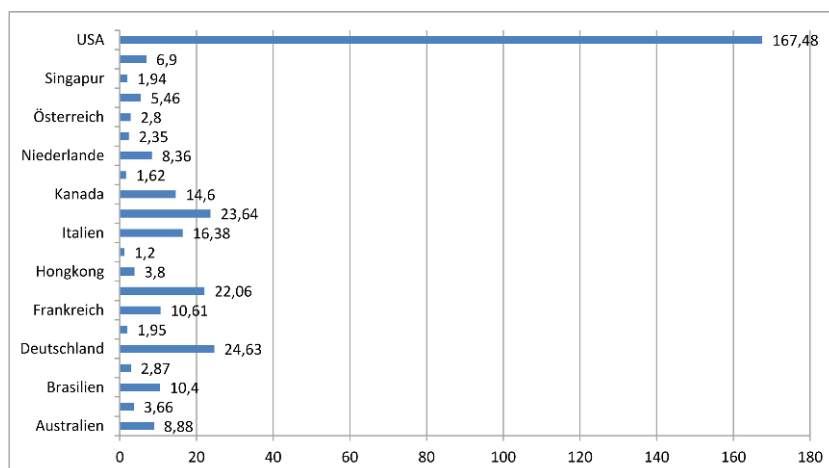


Bild B.1: Wie viele Menschen sind in einem Land online nach [Onl10]. Stand: 2001

In der Grafik B.2 wird veranschaulicht, wie viele Stunden pro Monat die Anwender in den unterschiedlichen Ländern online sind. Die wiedergegebenen Zahlen stammen aus dem Jahr 2001.

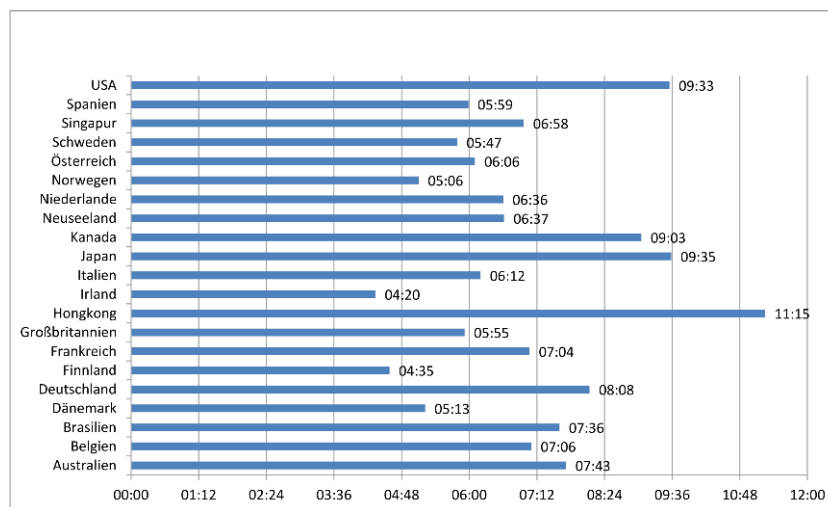


Bild B.2: Wie viele Stunden sind Internetnutzer online nach [Onl10] Stand: 2001

Grafik B.3 beschreibt aus dem Jahre 2010. Wie viel Zeit Jugendliche (Alter 21-25 mit Internetzugang) in der Woche im Internet verbracht haben [Sch10].

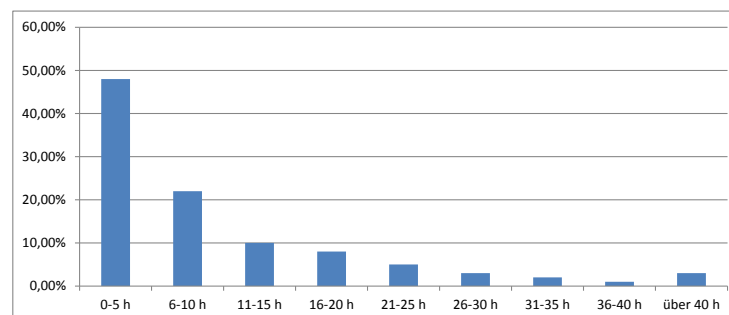


Bild B.3: Wie viel Stunden sind Jugendliche pro Woche alles in allem im Internet nach [Sch10] Stand: 2010

In der Zeit, in der Personen im Internet sind, verwenden diese unterschiedliche Kommunikationsapplikationen, die das Internet bereitstellt. Die Grafik B.4 veranschaulicht die Nutzung von Kommunikationsmittel der deutschen Internetnutzer.

Die genannten Kommunikationsmittel beschreiben Systemkomponenten von Groupware-System, die die soziale Interaktion fördern. Über die Jahre wurden diese

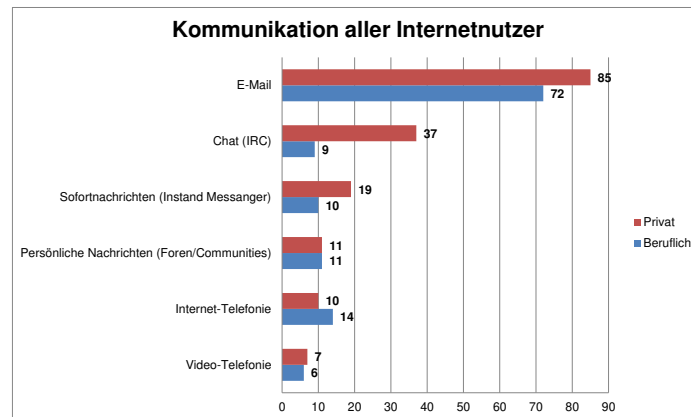
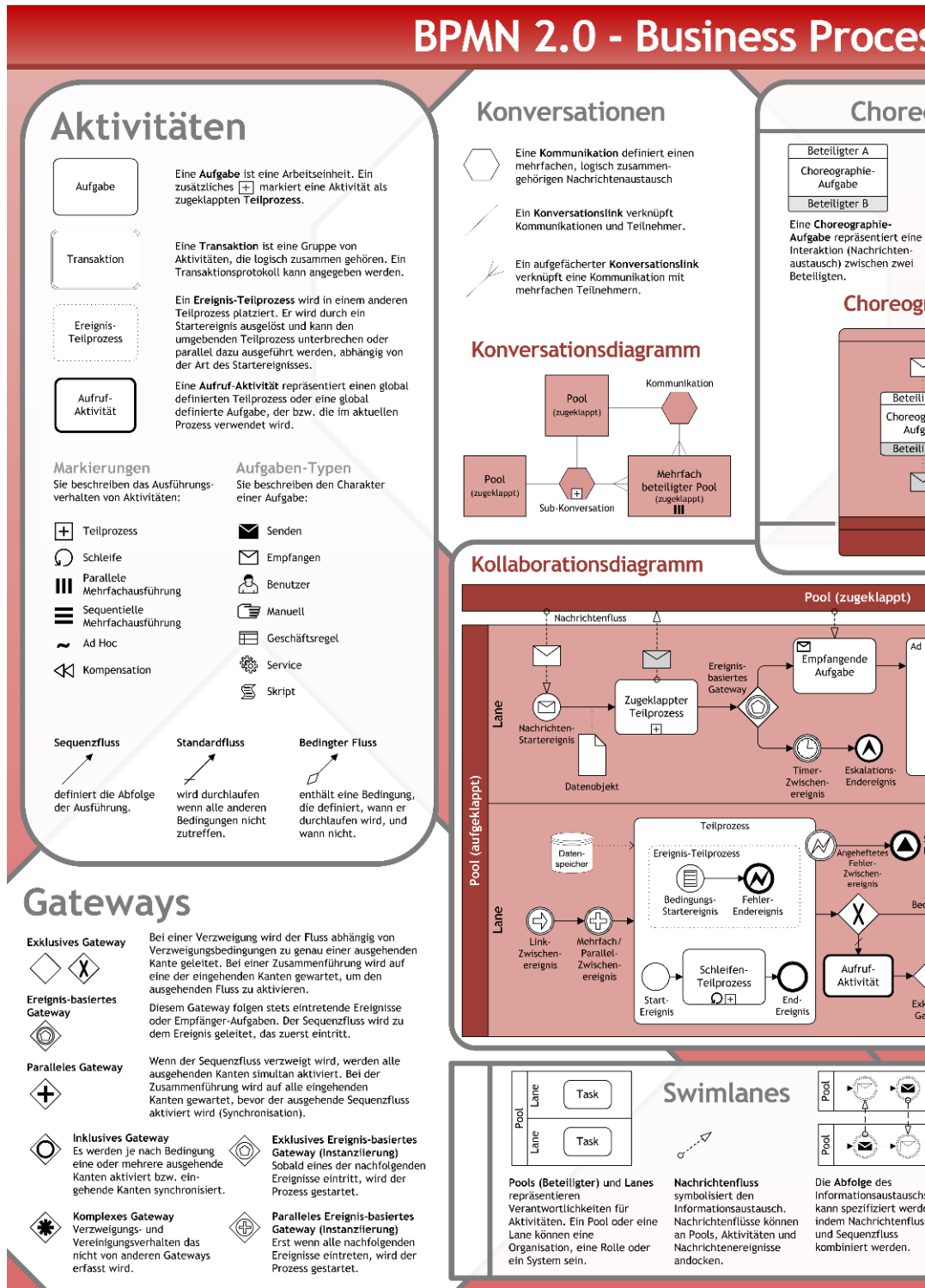


Bild B.4: Welche Kommunikationsmittel werden von allen Internetnutzern in Prozent verwendet nach [Bit10] Stand: 2010

Systeme vermehrt im privaten Bereich etabliert, und die Nutzung ist den Anwendern vertraut.

C BPMN-Poster



Gateways

Exklusives Gateway

Bei einer Verzweigung wird der Fluss abhängig von Verzweigungsbedingungen zu genau einer ausgehenden Kante geleitet. Bei einer Zusammenführung wird auf eine der eingehenden Kanten gewartet, um den ausgehenden Fluss zu aktivieren.

Ereignis-basiertes Gateway

Diesem Gateway folgen stets eintretende Ereignisse oder Empfänger-Aufgaben. Der Sequenzfluss wird zu dem Ereignis geleitet, das zuerst eintritt.

Paralleles Gateway

Wenn der Sequenzfluss verzweigt wird, werden alle ausgehenden Kanten simultan aktiviert. Bei der Zusammenführung wird auf alle eingehenden Kanten gewartet, bevor der ausgehende Sequenzfluss aktiviert wird (Synchronisation).

Inklusives Gateway

Es werden je nach Bedingung eine oder mehrere ausgehende Kanten aktiviert bzw. eingehende Kanten synchronisiert.

Exklusives Ereignis-basiertes Gateway (Instanzisierung)

Sobald eines der nachfolgenden Ereignisse eintritt, wird der Prozess gestartet.

Komplexes Gateway

Verzweigungs- und Vereinigungsverhalten das nicht von anderen Gateways erfasst wird.

Paralleles Ereignis-basiertes Gateway (Instanzisierung)

Erst wenn alle nachfolgenden Ereignisse eintreten, wird der Prozess gestartet.

Swimlanes

Pool

Lane

Task

Lane

Task

Pools (Beteiligter) und Lanes repräsentieren Verantwortlichkeiten für Aktivitäten. Ein Pool oder eine Lane können eine Organisation, eine Rolle oder ein System sein.

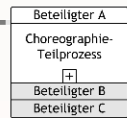
Nachrichtenfluss symbolisiert den Informationsaustausch. Nachrichtenflüsse können an Pools, Aktivitäten und Nachrichtenergebnisse andocken.

Die **Abfolge des Informationsaustauschs** kann spezifiziert werden indem Nachrichtenfluss- und Sequenzfluss kombiniert werden.

Bild C.1: BPMN Symbolposter Teil 1. Quelle: BPMP[BO10]

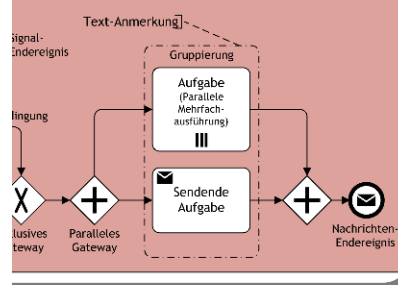
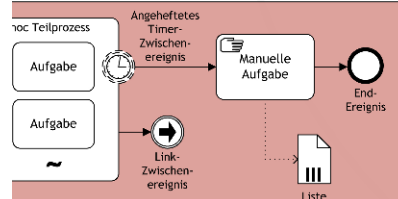
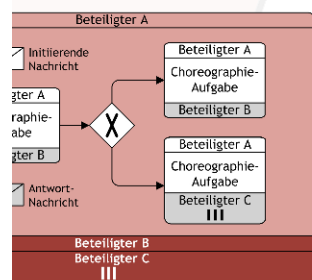
Choreographien

III
Eine **Mehrfach-Beteiligter Markierung** beschreibt eine Reihe von Beteiligten derselben Sorte.



Ein **Choreographie-Teilprozess** enthält eine verfeinerte Choreographie mit mehreren Interaktionen.

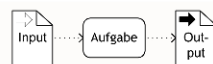
Choreographie-Diagramm



Ereignisse

| | Start | Zwischen | Ende |
|---|-------|----------|------|
| Top-Level | | | |
| Ereignis-Teilprozess Unterbrechend | | | |
| Ereignis-Teilprozess Nicht-unterbrechend | | | |
| Eingetreten | | | |
| Angeheftet unterbrechend | | | |
| Angeheftet Nicht-unterbrechend | | | |
| Ausgelöst | | | |
| Blanko: Untypisierte Ereignisse, i. d. R. am Start oder Ende eines Prozesses. | | | |
| Nachricht: Empfang und Versand von Nachrichten. | | | |
| Timer: Periodische zeitliche Ereignisse, Zeitpunkte oder Zeitspannen. | | | |
| Eskalation: Meldung an den nächsthöheren Verantwortlichen. | | | |
| Bedingung: Reaktion auf veränderte Bedingungen und Bezug auf Geschäftsregeln. | | | |
| Link: Zwei zusammengehörige Link-Ereignisse repräsentieren einen Sequenzfluss. | | | |
| Fehler: Auslösen und behandeln von definierten Fehlern. | | | |
| Abbruch: Reaktion auf abgebrochene Transaktionen oder Auslösen von Abbrüchen. | | | |
| Kompensation: Behandeln oder Auslösen einer Kompensation | | | |
| Signal: Signal über mehrere Prozesse. Auf ein Signal kann mehrfach reagiert werden. | | | |
| Mehrfach: Eintreten eines von mehreren Ereignissen. Auslösen aller Ereignisse. | | | |
| Mehrfach/Parallel: Eintreten aller Ereignisse. | | | |
| Terminierung: Löst die sofortige Beendigung des Prozesses aus. | | | |

Daten



Ein **Dateninput** ist ein externer Input für den ganzen Prozess. Er kann von einer Aktivität gelesen werden.
Ein **Datenoutput** ist eine Variable, die als Ergebnis eines ganzen Prozesses erzeugt wird.



Ein **Datenobjekt** repräsentiert Informationen, die durch den Prozess fließen, wie z.B. Dokumente, Emails oder Briefe.



Ein **Listen-Datenobjekt** repräsentiert eine Gruppe von Informationen, z.B. eine Liste mit Bestellpositionen.



Ein **Datenspeicher** ist ein Ort wo der Prozess Daten lesen oder schreiben kann, z.B. eine Datenbank oder ein Aktschrank. Er existiert unabhängig von der Lebensdauer der Prozessinstanz.



Eine **Nachricht** weist auf den Inhalt einer Kommunikation zwischen zwei Teilnehmern hin.



Bild C.2: BPMN Symbolposter Teil 2. Quelle: BPMP[BO10]

Literaturverzeichnis

- [ABN⁺04] AMMENWERTH, E. ; BRENDER, J. ; NYKÄNEN, P. ; PROKOSCH, H.U. ; RIGBY, M. ; TALMON, J. u. a.: Visions and strategies to improve evaluation of health information systems:: Reflections and lessons based on the HIS-EVAL workshop in Innsbruck. In: *International Journal of Medical Informatics* 73 (2004), Nr. 6, S. 479–491. – ISSN 1386–5056
- [ADLP⁺06] ARDISSONO, L. ; DI LEVA, A. ; PETRONE, G. ; SEGNAN, M. ; SONNESSA, M.: Adaptive medical workflow management for a context-dependent home healthcare assistance service. In: *Electronic Notes in Theoretical Computer Science* 146 (2006), Nr. 1, S. 59–68. – ISSN 1571–0661
- [AGH⁺03] AMMENWERTH, E. ; GRÄBER, S. ; HERRMANN, G. ; BÜRKLE, T. ; KÖNIG, J.: Evaluation of health information systems—problems and challenges. In: *International Journal of Medical Informatics* 71 (2003), Nr. 2-3, S. 125–135. – ISSN 1386–5056
- [AK10] AAGESEN, G. ; KROGSTIE, J.: Analysis and design of business processes using BPMN. In: *Handbook on Business Process Management 1* (2010), S. 213–235
- [All08] ALLWEYER, T.: *BPMN-Business Process Modeling Notation: Einführung in den Standard für die Geschäftsprozessmodellierung*. Books on Demand, 2008. – ISBN 3837070042
- [AW08] ATLURI, V. ; WARNER, J.: Security for workflow systems. In: *Handbook of Database Security* (2008), S. 213–230
- [AWG05a] AALST, W. M. P. d. ; WESKE, M. ; GRÜNBAUER, D.: Case handling: a new paradigm for business process support. In: *Data & Knowledge Engineering* 53 (2005), Nr. 2, S. 129–162

- [AWG05b] AALST, W.M.P. Van d. ; WESKE, M. ; GRÜNBAUER, D.: Case handling: a new paradigm for business process support. In: *Data & Knowledge Engineering* 53 (2005), Nr. 2, S. 129–162. – ISSN 0169–023X
- [AWS01] AUGUST-WILHELM SCHEER, Alexander K.: *Consulting: Wissen für die Strategie-, Prozess- und IT-BeratungSprin.* Springer, 2001
- [AZ10] ARD-ZDF: *Fast 50 Millionen Deutsche sind online.* ard-zdf-onlinestudie. <http://www.ard-zdf-onlinestudie.de/index.php?id=264&L=0&type=1>. Version: September 2010
- [BCM⁺10] BOTTRIGHI, A. ; CHESANI, F. ; MELLO, P. ; MONTALI, M. ; MONTANI, S. ; STORARI, S. ; TEREZIANI, P.: Analysis of the GLARE and GPROVE Approaches to Clinical Guidelines. In: *Knowledge Representation for Health-Care. Data, Processes and Guidelines* (2010), S. 76–87
- [BGH⁺07] BHATTACHARYA, Kamal ; GEREDE, Cagdas E. ; HULL, Richard ; LIU, Rong ; SU, Jianwen: Towards Formal Analysis of Artifact-Centric Business Process Models. In: *BPM, 2007*, S. 288–304
- [Bit10] BITCOM: *Internetnutzerverhalten bzgl. der Nutzung von Internetkommunikationsmitteln.* Bitcom. http://www.bitkom.org/de/presse/62013_58801.aspx. Version: September 2010
- [BJ88] BULLEN, C.V. ; JOHANSEN, R.: Groupware, a key to managing business teams? In: *Working paper (Sloan School of Management); 2013-88.* (1988)
- [BO10] BPM-OFFENSIVE, Berliner: BPMN-Poster. In: *Berliner BPM-Offensive Webseite* 1 (10.01.2010), 1. <http://www.bpmb.de/index.php/BPMNPoster>
- [BRNT97] BERGAMASCHI, R.A. ; RAJE, S. ; NAIR, I. ; TREVILLYAN, L.: Control-flow versus data-flow-based scheduling: combining both approaches in an adaptive scheduling system. In: *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 5 (1997), Nr. 1, S. 82–100. – ISSN 1063–8210
- [CGHS09] CALVANESE, Diego ; GIACOMO, Giuseppe D. ; HULL, Richard ; SU, Jianwen: Artifact-Centric Workflow Dominance. In: *ICSOC/ServiceWave, 2009*, S. 130–143

- [CH09] COHN, D. ; HULL, R.: Business artifacts: A data-centric approach to modeling business operations and processes. In: *IEEE Data Eng. Bull* 32 (2009), Nr. 3, S. 3–9
- [CKO92] CURTIS, Bill ; KELLNER, Marc I. ; OVER, Jim: Process modeling. In: *Commun. ACM* 35 (1992), September, 75–90. <http://dx.doi.org/http://doi.acm.org/10.1145/130994.130998>. – DOI <http://doi.acm.org/10.1145/130994.130998>. – ISSN 0001–0782
- [Coa99] COALITION, Workflow M.: Workflow Management Coalition: Terminology & Glossary. In: *Document Number WFMC-TC-1011* 1 (1999)
- [CWW⁺06] CHAUDHRY, B. ; WANG, J. ; WU, S. ; MAGLIONE, M. ; MOJICA, W. ; ROTH, E. ; MORTON, S.C. ; SHEKELLE, P.G.: Systematic review: impact of health information technology on quality, efficiency, and costs of medical care. In: *Annals of internal medicine* 144 (2006), Nr. 10, S. 742
- [DEH⁺00] DOURISH, P. ; EDWARDS, W.K. ; HOWELL, J. ; LAMARCA, A. ; LAMPING, J. ; PETERSEN, K. ; SALISBURY, M. ; TERRY, D. ; THORNTON, J.: A programming model for active documents. In: *Proceedings of the 13th annual ACM symposium on User interface software and technology* ACM, 2000. – ISBN 1581132123, S. 41–50
- [DEL⁺00] DOURISH, P. ; EDWARDS, W.K. ; LAMARCA, A. ; LAMPING, J. ; PETERSEN, K. ; SALISBURY, M. ; TERRY, D.B. ; THORNTON, J.: Extending document management systems with user-specific active properties. In: *ACM Transactions on Information Systems (TOIS)* 18 (2000), Nr. 2, S. 140–170. – ISSN 1046–8188
- [EFGK03] EUGSTER, P.T. ; FELBER, P.A. ; GUERRAOU, R. ; KERMARREC, A.M.: The many faces of publish/subscribe. In: *ACM Computing Surveys (CSUR)* 35 (2003), Nr. 2, S. 114–131. – ISSN 0360–0300
- [EGR91] ELLIS, C.A. ; GIBBS, S.J. ; REIN, G.: Groupware: some issues and experiences. In: *Communications of the ACM* 34 (1991), Nr. 1, S. 39–58. – ISSN 0001–0782
- [Ell00] ELLIS, CA: An evaluation framework for collaborative systems. In: *University of Colorado at Boulder Technical Report CU-CS-9001-00* 1 (2000), S. 1–20

- [FHS09] FRITZ, Christian ; HULL, Richard ; SU, Jianwen: Automatic construction of simple artifact-based business processes. In: *ICDT*, 2009, S. 225–238
- [GBS07] GEREDE, Cagdas E. ; BHATTACHARYA, Kamal ; SU, Jianwen: Static Analysis of Business Artifact-centric Operational Models. In: *SOCA*, 2007, S. 133–140
- [Ger07] GERYBADZE, A.: Gruppendynamik und Verstehen in Innovation Communities. In: *Management der frühen Innovationsphasen: Grundlagen-Methoden-neue Ansätze* (2007), S. 199. ISBN 3834903752
- [GHS95] GEORGAKOPOULOS, D. ; HORNICK, M. ; SHETH, A.: An overview of workflow management: From process modeling to workflow automation infrastructure. In: *Distributed and parallel Databases 3* (1995), Nr. 2, S. 119–153. – ISSN 0926–8782
- [Gru88] GRUDIN, J.: Why CSCW applications fail: problems in the design and evaluation of organizational interfaces. In: *Proceedings of the 1988 ACM conference on Computer-supported cooperative work* ACM, 1988. – ISBN 0897912829, S. 85–93
- [Gru02] GRUDIN, J.: Computer-supported cooperative work: History and focus. In: *Computer* 27 (2002), Nr. 5, S. 19–26. – ISSN 0018–9162
- [GS07] GEREDE, C. E. ; SU, J.: Specification and verification of artifact behaviors in business process models. In: *Lecture Notes in Computer Science* 4749 (2007), S. 181
- [GS10] GEREDE, C. ; SU, J.: Specification and verification of artifact behaviors in business process models. In: *Service-Oriented Computing–ICSOC 2007* (2010), S. 181–192
- [HM00] HEINRICH, E. ; MAURER, H.A.: Active documents: Concept, implementation and applications. In: *Journal of Universal Computer Science* 6 (2000), Nr. 12, S. 1197–1202
- [HPA05] HEINIS, T. ; PAUTASSO, C. ; ALONSO, G.: Design and evaluation of an autonomic workflow engine. In: *Autonomic Computing, 2005. ICAC 2005. Proceedings. Second International Conference on IEEE*, 2005, S. 27–38

- [Hul08] HULL, R.: Artifact-centric business process models: Brief survey of research results and challenges. In: *On the Move to Meaningful Internet Systems: OTM 2008* 1 (2008), S. 1152–1163
- [Hut04] HUTH, C.: *Groupware-basiertes Ad-hoc-Workflow-Management: das GroupProcess-System; Konzeption und prototypische Implementierung einer "Collaboration on Demand"-Lösung zur Unterstützung von schwach strukturierten Prozessen in Unternehmen*, Universität Paderborn, Diss., 2004
- [Jab95] JABLONSKI, S.: Workflow-Management-Systeme: Motivation, Modellierung, Architektur. In: *Informatik Spektrum* 18 (1995), Nr. 1, S. 13–24
- [JB96] JABLONSKI, S. ; BUSSLER, C.: Workflow management: modeling concepts, architecture and implementation. (1996)
- [JBS97] JABLONSKI, S. ; BÖHM, M. ; SCHULZE, W.: *Workflow-Management: Entwicklung von Anwendungen und Systemen: Facetten einer neuen Technologie*. dpunkt Verlag für digitale Technologie, 1997. – ISBN 392099373X
- [Joh88] JOHANSEN, R.: *Groupware: Computer support for business teams*. The Free Press New York, NY, USA, 1988. – ISBN 0029164915
- [KP04] KUSHNIRUK, A.W. ; PATEL, V.L.: Cognitive and usability engineering methods for the evaluation of clinical information systems. In: *Journal of Biomedical Informatics* 37 (2004), Nr. 1, S. 56–76. – ISSN 1532–0464
- [KR01] KUROSE, J.F. ; ROSS, K.W.: *Computer networking*. Bd. 75. Addison-Wesley Boston, 2001
- [KR09a] KÜNZLE, V. ; REICHERT, M.: Herausforderungen auf dem Weg zu datenorientierten Prozess-Management-Systemen. In: *EMISA Forum* Bd. 29, 2009. – ISSN 1610–3351, S. 9–24
- [KR09b] KÜNZLE, V. ; REICHERT, M.: Towards Object-Aware Process Management Systems: Issues, Challenges, Benefits. In: *Lecture Notes in Business Information Processing* 29 (2009), S. 197–210
- [KR09c] KÜNZLE, V. ; REICHERT, M.: Towards object-aware process management systems: issues, challenges, benefits. In: *Enterprise, Business-Process and Information Systems Modeling* 1 (2009), S. 197–210

- [KR10a] KÜNZLE, V. ; REICHERT, M.: Herausforderungen bei der Integration von Benutzern in Datenorientierten Prozess-Management-Systemen. In: *EMISA Forum* Bd. 30 Koellen-Verlag, 2010. – ISSN 1610–3351, S. 11–28
- [KR10b] KÜNZLE, V. ; REICHERT, M.: Integrating users in object-aware process management systems: Issues and challenges. In: *Business Process Management Workshops* Springer, 2010, S. 29–41
- [KR10c] KÜNZLE, V. ; REICHERT, M.: PHILharmonicFlows: towards a framework for object-aware process management. In: *Journal of Software Maintenance and Evolution: Research and Practice* 1 (2010), S. 1532–0618. – ISSN 1532–0618
- [KRW90] KARBE, B. ; RAMSPERGER, N. ; WEISS, P.: Support of cooperative work by electronic circulation folders. In: *ACM SIGOIS Bulletin* 11 (1990), Nr. 2-3, S. 109–117. – ISSN 0894–0819
- [Löc03] LÖCHERBACH, P.: Einsatz der methode case management in Deutschland: "Übersicht zur Praxis im Sozial-und Gesundheitswesen. In: *Porz F., Erhardt, H.: Kongressbericht* 3 (2003)
- [Leh08] LEHMANN, F.R.: *Integrierte Prozessmodellierung mit ARIS*. dpunkt Verl., 2008. – ISBN 3898644979
- [Len09] LENZ, Richard: Information Systems in Healthcare-State and Steps towards Sustainability. In: *Yearbook of medical informatics* (2009), S. 63
- [LHHM05] LENZ, Richard (Hrsg.) ; HASENKAMP, Ulrich (Hrsg.) ; HASSELBRING, Wilhelm (Hrsg.) ; MANFRED, Reichert (Hrsg.): *Enterprise Application Integration 2005, Proceedings of the 2nd GI-Workshop on Enterprise Application Integration (EAI-05)*. Bd. 141. Marburg, Germany : CEUR-WS.org, July 2005 (CEUR Workshop Proceedings)
- [LK03] LENZ, R. ; KUHN, K. A.: Towards a continuous evolution and adaptation of information systems in healthcare. In: *Int J Med Inf* 73 (2003), July, Nr. 1, S. 75–89
- [MBR05] MUTSCHLER, B. ; BUMILLER, J. ; REICHERT, M.: Towards an Evaluation Framework for Business Process Integration and Management. In: *Proceedings 2nd Int'l Workshop on Interoperability Research for Networked Enterprises Applications and Software*, IEEE Digital Library, 2005, S. 1

- [Min82] MINKER, J.: On indefinite databases and the closed world assumption. In: *6th Conference on Automated Deduction* Springer, 1982, S. 292–308
- [MRH06] MÜLLER, Dominic ; REICHERT, Manfred ; HERBST, Joachim: Flexibility of Data-Driven Process Structures. In: *Business Process Management Workshops*, 2006, S. 181–192
- [MRH07a] MÜLLER, D. ; REICHERT, M. ; HERBST, J.: Data-driven modeling and coordination of large process structures. In: *Lecture Notes in Computer Science* 4803 (2007), S. 131
- [MRH07b] MÜLLER, Dominic ; REICHERT, Manfred ; HERBST, Joachim: Data-Driven Modeling and Coordination of Large Process Structures. In: *OTM Conferences (1)*, 2007, S. 131–149
- [MTJ⁺10] MILI, Hafedh ; TREMBLAY, Guy ; JAOUDE, Guitta B. ; LEFEBVRE, Éric ; ELABED, Lamia ; BOUSSAIDI, Ghizlane E.: Business process modeling languages: Sorting through the alphabet soup. In: *ACM Comput. Surv.* 43 (2010), December, 4:1–4:56. <http://dx.doi.org/http://doi.acm.org/10.1145/1824795.1824799>. – DOI <http://doi.acm.org/10.1145/1824795.1824799>. – ISSN 0360–0300
- [NH99] NASTANSKY, L. ; HABERSTOCK, P.: Groupware-basiertes Knowledge Management im Controlling. In: *GCC-Webseite (27-12-2010)* 1 (1999), S. 11–13
- [NL09] NEUMANN, Christoph P. ; LENZ, Richard: alpha-Flow: A Document-based Approach to Inter-Institutional Process Support in Healthcare. In: *Proc of the 3rd Int'l Workshop on Process-oriented Information Systems in Healthcare (ProHealth'09) in conjunction with the 7th Int'l Conf on Business Process Management (BPM'09)*. Ulm, Germany : 7th Int'l Conf on Business Process Management, September 2009, S. 1
- [NL10] NEUMANN, Christoph P. ; LENZ, Richard: The alpha-Flow Use-Case of Breast Cancer Treatment – Modeling Inter-Institutional Healthcare Workflows by Active Documents. In: *Proc of the 8th Int'l Workshop on Agent-based Computing for Enterprise Collaboration (ACEC) at the 19th Int'l Workshops*

- on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE 2010)*. Larissa, Greece : Infrastructures for Collaborative Enterprises (WETICE 2010), Juni 2010, 1
- [NL11a] NEUMANN, Christoph P. ; LENZ, Richard: The alpha-Flow Approach to Inter-Institutional Process Support Health. In: *International Journal of Knowledge-Based Organizations (IJKBO)* (2011)
- [NL11b] NEUMANN, Christoph P. ; LENZ, Richard: The alpha-Flow Approach to Inter-Institutional Process Support Health. In: ? ? (2011), S. ?
- [OBS97] ORTNER, S. E: J. E: Jablonski ; BÖHM, M. ; SCHULZE, W. (.: *Workflow-Management: Entwicklung von Anwendungen und Systemen: Facetten einer neuen Technologie*. dpunkt.Verlag für digitale Technologie, 1997. – ISBN 392099373X
- [Onl10] ONLINE, Focus: *Internet-Nutzer. Wie viele Menschen sind online*. Focus Online. <http://www.focus.de/D/DD/DD36/DD36A/dd36a.htm>. Version: September 2010
- [PB05] POWELL, J. ; BUCHAN, I.: Electronic health records should support clinical research. In: *Journal of Medical Internet Research* 7 (2005), Nr. 1
- [PF88] PICOT, A. ; FRANCK, E.: Die Planung der Unternehmensressource" Information". In: *WISU* 10 (1988), S. 544–549
- [Pit10] PITSCHKE, J.: *Unternehmensmodellierung für die Praxis: Eine Einführung in die Darstellung von Unternehmensmodellen*. BoD–Books on Demand, 2010. – ISBN 3842325762
- [PR84] PICOT, A. ; REICHWALD, R.: *Bürokommunikation*. Ludwig-Maximilians-Universität München, 1984
- [RFMP07] RODRÍGUEZ, A. ; FERNÁNDEZ-MEDINA, E. ; PIATTINI, M.: A bpmn extension for the modeling of security requirements in business processes. In: *IEICE Transactions on Information and Systems* 90 (2007), Nr. 4, S. 745. – ISSN 0916–8532
- [RKG07a] RYNDINA, K. ; KÜSTER, J. ; GALL, H.: Consistency of business process models and object life cycles. In: *Models in Software Engineering* (2007), S. 80–90

- [RKG07b] RYNDINA, K. ; KUSTER, J.M. ; GALL, H.: Consistency of business process models and object life cycles. In: *Lecture Notes in Computer Science* 4364 (2007), S. 80
- [RM98] ROSEMAN, M. ; MUEHLEN, M.: Evaluation of workflow management systems-a meta model approach. In: *Australian Journal of Information Systems* 6 (1998), S. 103–116. – ISSN 1039–7841
- [RM09] RINDERLE-MA, S.: Data Flow Correctness in Adaptive Workflow Systems. In: *Emisa Forum* Bd. 29, 2009, S. 25–35
- [RRA03] REIJERS, H.A. ; RIGTER, JHM ; AALST, W.M.P. van d.: The case handling case. In: *International Journal of Cooperative Information Systems* 12 (2003), Nr. 3, S. 365–392. – ISSN 0218–8430
- [RRD03] RINDERLE, S. ; REICHERT, M. ; DADAM, P.: Evaluation of correctness criteria for dynamic workflow changes. In: *Business Process Management* 1 (2003), S. 1021–1021
- [RSL78] ROSENKRANTZ, Daniel J. ; STEARNS, Richard E. ; LEWIS, Philip M. II: System level concurrency control for distributed database systems. In: *ACM Trans. Database Syst.* 3 (1978), June, 178–198. <http://dx.doi.org/http://doi.acm.org/10.1145/320251.320260>. – DOI <http://doi.acm.org/10.1145/320251.320260>. – ISSN 0362–5915
- [SAMS01] SCHWARZ, S. ; ABECKER, A. ; MAUS, H. ; SINTEK, M.: Anforderungen an die Workflow-Unterstützung für wissensintensive Geschäftsprozesse. In: *Proc. WM* 1 (2001)
- [SB92] SCHMIDT, K. ; BANNON, L.: Taking CSCW seriously: Supporting Articulation Work. In: *Computer Supported Cooperative Work (CSCW)* 1 (1992), Nr. 1, S. 7–40
- [Sch10] SCHELL, Schell J.: *Wie viel Stunden sind Sie pro Woche alles in allem im Internet (privat, in der Ausbildung, im Beruf)?* Statista. <http://de.statista.com/statistik/daten/studie/177010/umfrage/zeit-pro-woche-die-im-internet-verbracht-wird/>.
Version: September 2010

- [SRKM01] SCHLICHTER, J. ; REICHWALD, R. ; KOCH, M. ; MÖSLEIN, K.s: Rechnergestützte Gruppenarbeit (CSCW)(Computer Supported Cooperative Work (CSCW). In: *i-com* 1 (2001), Nr. 0/2001, S. 5. – ISSN 1618–162X
- [ST01] STAUCH, M. ; TOLKSDORF, R.: Design and implementation of an XSL-T and XML-based workflow system. In: *Proceedings of XML Europe 2001* (2001), S. 229–241
- [Ste98] STEIN, Katrin: *Integration von Anwendungsprozeßmodellierung und Workflow-Management*, Friedrich-Alexander-Universität Erlangen-Nürnberg, Diss., 1998
- [Swe10] SWENSON, Keith D.: *Mastering the Unpredictable*. Bd. 1. Meghan-Kiffer Press, 2010
- [UKM03] UCHITEL, S. ; KRAMER, J. ; MAGEE, J.: Synthesis of behavioral models from scenarios. In: *IEEE Transactions on Software Engineering* (2003), S. 99–115. – ISSN 0098–5589
- [Wil91] WILSON, P.: *Computer supported cooperative work: an introduction*. Springer, 1991. – ISBN 0792314468
- [WK05] WANG, Jianrui ; KUMAR, Akhil: A Framework for Document-Driven Workflow Systems. In: *Business Process Management*, 2005, S. 285–301
- [WM08] WHITE, S.A. ; MIERS, D.: *BPMN Modeling and Reference Guide: Understanding and Using BPMN*. Future Strategies Inc., 2008. – ISBN 0977752720
- [WS02] WHITTLE, J. ; SCHUMANN, J.: Generating statechart designs from scenarios. In: *Software Engineering, 2000. Proceedings of the 2000 International Conference on IEEE*, 2002. – ISBN 1581132069, S. 314–323

Lebenslauf

Name: André Kreuter
Geburtsdatum: 22.12.1984
Geburtsort: Castrop-Rauxel
Staatsangehörigkeit: Deutsch
Familienstand: ledig

1995–2004 **Schulbildung und Abitur**
an der Oswald von Nell Breuning Schule, Rödermark
seit 2004 **Studium der Informatik**
an der Friedrich-Alexander-Universität Erlangen-Nürnberg

