



Universität Ulm
Fakultät für Informatik

Abteilung Datenbanken
und Informationssysteme

Diplomarbeit

Visualisierungskonzepte für Prozessinformationen

vorgelegt von

Max Moldmann

Februar 2006

Gutachter: Prof. Dr. Peter Dadam, Universität Ulm
Dr. Manfred Reichert, Assoc. Prof., Universität Twente, NL
Betreuer: Dipl. Inf. Ralph Bobrik, Universität Ulm

Kurzfassung

Die bisherigen und gegenwärtigen Forschungs- und Entwicklungsarbeiten im Rahmen von Prozess-Management-Systemen (PMS) konzentrieren sich bisher auf theoretische, technische und implementierungsspezifische Aspekte. Diese Arbeit beschäftigt sich dagegen mit der Visualisierung von Prozessinformationen. Am Markt werden fortgeschrittene Werkzeuge für Prozess-Monitoring und Analyse angeboten. Die bis zum heutigen Stand verfügbaren PMS bieten allerdings nur recht eingeschränkte Möglichkeiten, die vom System verwalteten Prozessinformationen (Modell- und Laufzeitdaten) darzustellen.

Die Zielsetzung dieser Arbeit ist es zu untersuchen, welchen Anforderungen eine leistungsfähige Prozessvisualisierungskomponente genügen muss. Es gilt alle Aspekte von Prozessinformationen zugänglich zu machen, wie z. B. den Datenfluss oder temporale Aspekte. Durch die Darstellungen sollen Zusammenhänge sichtbar werden. Weiterhin gilt es die Visualisierungsbedürfnisse unterschiedlicher Benutzer zu erfüllen. Am Lifecycle eines Prozesses beteiligen sich u. a. Prozess-Modellierer und -Verantwortliche, Management, Anwender. Dies hat ein großes Spektrum an unterschiedlichen Aufgabentypen, etwa entwerfen, informieren, entscheiden und die Forderung nach unterschiedlichen Abstraktionsniveaus zur Folge, denen eine Darstellung gerecht werden muss.

Diese Anforderungen lassen sich durch unterschiedliche Ansätze erfüllen. Verschiedene Darstellungsformen wie Prozessgraphdarstellung, Swimlane-Darstellung, Kalenderdarstellung, Tabellendarstellung, Interaktionsdiagramm, Datenflussdiagramm und Matrixdarstellung werden als Werkzeuge zur Lösung unterschiedlichster Aufgaben entwickelt. Verschiedene Darstellungsarten wie Schema-, Instanz-, Multi-Instanz- und Multi-Schema-Darstellungen ermöglichen Detail- bis hin zu Übersichtsdarstellungen. Weiterhin werden Funktionen zur dynamischen Einblendung von Prozessaspekten vorgeschlagen und die View-Bildung, zur Erzeugung unterschiedlich komplexer Sichten auf Prozesse diskutiert.

Umfangreiche Interaktions- und Navigationsmöglichkeiten auf Prozessdaten werden vorgeschlagen. Die vorgestellten Lösungen beziehen Forschungsergebnisse zu Usability und Interface-Design mit ein, um auch Gebrauchstauglichkeit sicherzustellen.

Inhaltsverzeichnis

1. Einleitung	1
1.1. Einordnung	1
1.2. Das Ziel	3
1.3. Motivation	5
2. Grundlagen und Einführung	9
2.1. Visualisierung von Prozessdaten	9
2.1.1. Prozessinformationen	10
2.1.2. Darstellungsarten	16
2.1.3. Organisationsmodell	18
2.2. Ansätze zur Prozessvisualisierung	19
2.2.1. Veränderungen an der Grafik	20
2.2.2. Veränderungen an der Struktur	20
2.3. Zusammenfassung	22
3. Anforderungsanalyse	23
3.1. Goal-Directed-Design	24
3.2. Ziele nach Benutzergruppen	26
3.2.1. Bearbeiter / Anwender	30
3.2.2. Prozessverantwortlicher / Abteilungsleiter	30
3.2.3. Management	31
3.2.4. IT	31
3.2.5. Externer Mitarbeiter / Kunde	31
3.2.6. Prozessmodellierer	31
3.3. Zusammenfassung	31
4. Verwandte Arbeiten	33
4.1. Informationsvisualisierung	33
4.1.1. Kognitions- und Wahrnehmungsforschung	34
4.1.2. Usability	37
4.1.3. Interface-Design	38
4.1.4. Styleguides und Normen	41
4.1.5. Zusammenfassung	43
4.2. Existierende Visualisierungslösungen	44
4.2.1. Business Process Management (BPM) Werkzeuge	45
4.2.2. Projektplanung	52
4.2.3. Prozessneutrale Visualisierungssoftware	53
4.2.4. Zusammenfassung	55

5. Grafische Aspekte der Prozessdarstellung	57
5.1. Farben	57
5.1.1. Kontraste	58
5.2. Ebenen	59
5.3. Notationen	60
5.3.1. Knoten	61
5.3.2. Kanten	63
5.3.3. Ausführungsmarkierungen	69
5.4. Orientierung	70
5.5. Zusammenfassung	70
6. Darstellungsformen für Prozessdaten	73
6.1. Prozessgraphdarstellung	75
6.1.1. Blockstruktur	79
6.1.2. Zeitleiste	80
6.1.3. Implementierungsaspekte	81
6.2. Swimlane-Darstellung	82
6.2.1. Implementierungsaspekte	88
6.3. Kalenderdarstellung	88
6.3.1. Dynamische Zeitachse	94
6.4. Tabellendarstellung	94
6.4.1. Implementierungsaspekte	100
6.5. Interaktionsdiagramm	100
6.5.1. Notwendige Voraussetzungen	103
6.6. Datenflussdiagramm	103
6.7. Matrixdarstellung	105
6.7.1. Notwendige Voraussetzungen	117
6.7.2. Implementierungsaspekte	117
6.7.3. Bedingungen im Kontrollfluss	117
6.8. Zusammenfassung	118
7. Interaktionen auf Prozessinformationen	123
7.1. View-Bildung	123
7.1.1. Aggregation	125
7.1.2. Reduktion	126
7.2. Aktivitäten markieren	126
7.2.1. Wichtige einfache Markierungsregeln	127
7.2.2. Regelbaukastensystem	127
7.3. Detailansicht & Tooltips	130
7.4. Aufgabenbereich	130
7.5. Navigation	134
7.6. Prozesshistorie	138
7.6.1. On-Line Verlaufsinformationen	138
7.7. Zusammenfassung	139

8. Ausblick	141
8.1. Sicherheitsaspekte	141
8.2. Partitionierbarkeit von Prozessinformationen	142
8.3. Anpassung des Nutzer-Interface an Prozessmodelle	143
8.4. Mitbestimmung des Betriebsrates	144
8.5. Fazit	144
A. Glossar	147
B. Literaturverzeichnis	151

Tabellenverzeichnis

2.1. Übersicht über die Prozessinformationen	14
3.1. Benutzergruppen mit Beschreibung	27
3.2. Übersicht über die Fragestellungen und Aufgaben der Benutzergruppen – Teil 1 (Bearbeiter/Anwender, Prozessverantwortlicher/Abteilungsleiter, Management)	29
3.3. Übersicht über die Fragestellungen und Aufgaben der Benutzergruppen – Teil 2 (IT, Externer Mitarbeiter/Kunde, Prozessmodellierer)	30
6.1. Varianten der Swimlane-Darstellung	85
6.2. Varianten der Matrixdarstellung	106
6.3. Darstellungsarten mit Darstellungszielen und Einordnung der Werkzeuge nach Benutzerzielen	119
6.4. Mögliche Werkzeuge zum Lösen der Aufgaben der Benutzergruppen – Teil 1 (Bearbeiter/Anwender, Prozessverantwortlicher/Abteilungsleiter, Management)	120
6.5. Mögliche Werkzeuge zum Lösen der Aufgaben der Benutzergruppen – Teil 2 (IT, Externer Mitarbeiter/Kunde, Prozessmodellierer)	121
7.1. Beispiele für die Markierungsfunktion	129
7.2. Aufgaben im Aufgabenbereich	133
8.1. Kurzkritik für Zugriffsschutz durch Metamodell-Unterstützung	141
8.2. Kurzkritik für Zugriffsschutz durch Views	142

Abbildungsverzeichnis

1.1.	Architekturmodell Process-Warehouse	4
1.2.	Beispielprozess Change Management – Ausschnitt alte Version	6
1.3.	Beispielprozess Change Management – Ausschnitt neue Version	7
2.1.	Schematischer Ablauf der Prozessvisualisierung	10
2.2.	Möglichkeiten der Zuordnung von Kontrollfluss-Informationen zu Aktivitäten	15
2.3.	Darstellungsarten für die Visualisierung	16
2.4.	Konzepte zur Visualisierung	20
2.5.	Ausschnitt aus dem Netzplan des Donau-Iller-Nahverkehrsverbundes	21
3.1.	Ontologisches Designdiagramm	23
3.2.	Schema des Goal-Directed-Design Entwurfskonzeptes – Teil 1	24
3.3.	Schema des Goal-Directed-Design Entwurfskonzeptes – Teil 2	24
3.4.	Photoshop Variationenübersicht	25
3.5.	Schema für die Einordnung der Benutzergruppen	28
4.1.	Kategorisierung von Prozessorientierter Visualisierungssoftware	45
5.1.	Wirkung von Kontrast in Tabellen	58
5.2.	Symbolreduktion durch Integration	62
5.3.	Nutzung von Metaphern	62
5.4.	Knoten und Feldmarkierungen	63
5.5.	Variationen von Kanten	64
5.6.	Semantik Darstellungsmöglichkeiten	66
5.7.	Verdeutlichen der Blockstruktur	66
5.8.	Darstellung von Prozessschnittstellen	68
5.9.	Ausführungszustände von Aktivitäten (bereit, laufend, abgebrochen, fehl- geschlagen, abgeschlossen)	70
5.10.	Knoten bei horizontaler oder vertikaler Prozessdarstellung	70
5.11.	Darstellung von Aktivitätenknoten mit unterschiedlich vielen Attributen	71
6.1.	Übersicht über die Darstellungsformen mit dargestellten primären Prozess- aspekten	74
6.2.	Prozessgraphdarstellung	76
6.3.	Ausschnitt aus Change Management Prozess – Wirkung perspektivischer Darstellung	79
6.4.	Prozessgraph mit Blockstruktur	80
6.5.	Prozessgraph mit Blockstruktur und geöffnetem Unterprozess	80
6.6.	Orthogonales Layout	82

6.7. Hierarchisches Layout	82
6.8. Prozessgraph in Swimlane-Darstellung	83
6.9. Prozessgraph in geschachtelter Swimlane-Darstellung	84
6.10. Beispiel einer horizontalen Platzersparnis bei der Swimlane- gegenüber der Prozessgraphdarstellung	85
6.11. Kalenderdarstellung	89
6.12. Kalenderdarstellung in Multi-Instanz Variante	92
6.13. Planungszeiten in der Kalenderdarstellung	93
6.14. Mögliche Balkenkonfigurationen in der Kalenderdarstellung	93
6.15. Tabellendarstellung des Change-Management-Prozesses – Einfache Tabelle (ohne Hierarchie)	95
6.16. Tabellendarstellung des Change-Management-Prozesses – Normale Tabelle .	96
6.17. Interaktionsdiagramm – Schnittstellen zwischen Abteilungen	102
6.18. Datenflussdiagramm – Dokument: change request	104
6.19. Matrixdarstellung - Ausführungseinheiten und Systeme	107
6.20. Matrixdarstellung - Ausführungseinheiten und Dokumente	108
6.21. Matrixdarstellung - Ausführungseinheiten und Dokumente mit Tooltipp . .	108
7.1. Ablauf Prozessvisualisierung mit Views	124
7.2. Selektionsdialog in Apples iTunes	128
7.3. Spotlight im MacOS X System-Einstellungs-Dialog	128
7.4. Beispiel für Markierungsregeln	129
7.5. Aufgabenbereich in Microsofts Powerpoint	131
7.6. Konfiguration der Anzeige der Prozessdetails	132
7.7. Vertikale und horizontale Navigation in Prozesshierarchie	133
7.8. Zurück-Funktion mit benannten Aktionen im Internet Explorer	135
7.9. Übersichtsfenster aus Adobe Photoshop	136
7.10. Tabs im Hauptfenster mit verschiedenen Darstellungen	136
7.11. Navigation im Amazon Shop	137

1

Einleitung

1.1. Einordnung

Prozess-Management-Systeme (PMS) optimieren Arbeitsabläufe in Unternehmen. Hierfür werden Geschäftsprozesse in ihre einzelnen Arbeitsschritte zerlegt und für die Verwendung in einem PMS modelliert. PMS bieten u. a. folgende Funktionalität im Umgang mit Prozessen: Ausführung, Automatisierung, Steuerung, Monitoring und Analyse.

Für den Umgang mit Prozessen werden außer einem meist auf Kennzahlen basierenden Monitoring auch leistungsfähige Visualisierungsmöglichkeiten von Prozessinformationen benötigt. Marktübliche PMS zeigen allerdings genau auf diesem Gebiet große Einschränkungen, was die Mächtigkeit der Darstellung, ihre Konfigurierbarkeit, die Interaktionsmöglichkeiten, View-Bildung und allgemein ihre Gebrauchstauglichkeit angeht.

WARUM PROZESS-MANAGEMENT?

Heute gibt es viele Gründe [RD00], die vorhandenen Strukturen und Prozesse in Unternehmen zu erfassen, zu analysieren und zu optimieren. Einige sind die Verbesserung der Produktqualität oder der Wirtschaftlichkeit und die Reduzierung der Zeit zwischen Produktzyklen. Hierbei auf die Unterstützung durch rechnergestützte Werkzeuge zu setzen, liegt auf der Hand. Diese helfen bei der Modellierung und Analyse von Arbeitsabläufen.

Eine hochgradige Arbeitsteilung impliziert naturgemäß die Gefahr von Reibungsverlusten. Daher ist die Optimierung der Abläufe unumgänglich. Dieser Optimierungsprozess kann durch Rechnersysteme gestützt werden. PMS sorgen dafür, dass die nötige Koordinierungsarbeit, die einen Overhead darstellt, nicht so groß wird, dass die Synergieeffekte durch die Arbeitsteilung nicht in großen Teilen wieder zunichte gemacht werden. Arbeitsteilung hat also nicht nur positive Effekte, denn für den Einzelnen ist der Zusammenhang seiner Teilaufgabe mit dem Gesamtprozess oft nicht mehr erkennbar. Die Optimierung von Geschäftsprozessen endet beispielsweise oft an Abteilungsgrenzen. Es besteht auch die Gefahr, dass eine Teilaufgabe zum reinen Selbstzweck wird. Nach dem britischen Historiker

Cyril Northcote Parkinson [Park05], der sich mit der Bürokratie der britischen Admiralität beschäftigte, blähen sich Aufgaben selbstständig soweit auf, bis sie die ganze verfügbare Zeit ausfüllen.

NOTWENDIGKEIT VON MÄCHTIGEN VISUALISIERUNGSKONZEPTEN

Aufgeblähte Abläufe machen Unternehmen unbeweglich. Der hohe Wettbewerbsdruck erfordert jedoch rasche Anpassung an veränderte Rahmenbedingungen. D. h. es ist erwünschtes Ziel, dass ein PMS ein Unternehmen in die Lage versetzt, Prozesse zu verschlanken, zu optimieren, und Geschäftsprozesse ständig anzupassen und zu ändern (siehe Schema-Evolution [RD98]). Optimal wäre, wenn sämtliche betriebliche Abläufe durchgängig rechnergestützt abgebildet werden. Ein positiver Nebeneffekt liegt im dadurch verbesserten Wissensmanagement [Jabl01]. Mit dem Ausscheiden von Mitarbeitern verlieren Unternehmen das oftmals nur implizite Wissen über Abläufe. Ein Großteil dieses Wissens kann in PMS gespeichert und verwaltet werden.

Ein PMS kann als Informationssystem aufgefasst werden. Die Benutzerschnittstelle eines solchen Systems muss in der Lage sein, seinen Anwendern mächtige Hilfsmittel zur Seite zu stellen, die es trotz großer Informationsmengen erlauben, schnell und effizient die gesuchten Informationen zu finden. Die Lösung hierfür sind leistungsfähige, an die jeweilige Aufgabe angepasste Visualisierungen. Die Herausforderung liegt darin, trotz den zwischen komplexen Ablaufplänen bestehenden logischen Abhängigkeiten und komplizierten Datenflüssen, dem Anwender zu ermöglichen, den nötigen Überblick zu verschaffen.

ZUM DETAILGRAD DER ABBILDUNG VON ABLÄUFEN IN PROZESSEN

Die Entwicklung von neuen Arbeitsprozessen ist sehr aufwendig. Es gilt, die richtige Granularität zu finden, in der Arbeitsabläufe im PMS abgebildet werden. Dabei gibt es zwei Extremfälle. Eine sehr grobkörnige Abbildung zerlegt einen Prozess nur in die notwendigsten Schritte. Das andere Extrem ist eine feinkörnige Abbildung, die für jeden noch so kleinen Arbeitsschritt eine separate Aktivität vorsieht.

Der Vorteil einer hohen Auflösung liegt darin, dass sehr viel Wissen über die Abläufe im System gespeichert werden kann. Eine feinkörnige Einteilung aber erhöht den Aufwand für die Erstellung und spätere Anpassungen. Außerdem wächst mit höherem Detaillierungsgrad die Anzahl der Aktivitäten eines Prozesses und damit die Anzahl der Einträge in den Arbeitslisten der Bearbeiter, was sie dann auch für kleine Routineaufgaben an den Bildschirm zwingt.

Relativ grobkörnige Auflösungen haben den Vorteil, dass die resultierenden Darstellungen eine geringere Komplexität aufweisen. Eine übersichtliche Visualisierung gelingt bei weniger komplexen Prozessen natürlich leichter. Für spätere Prozessänderungen reduziert sich der Aufwand.

Der Zielkonflikt zwischen Granularität und Anpassbarkeit ließe sich zumindest für den Anwender abmildern, indem zwischen verschiedenen manuellen Prozessschritten unterschieden wird: Zwischen solchen, die in den Arbeitslisten auftauchen und solchen, die nur der detaillierteren Beschreibung des Arbeitsablaufs dienen. Geeignete Ansichten verbergen dann bei Bedarf die Details.

1.2. Das Ziel

Diese Arbeit setzt sich mit der Frage auseinander, wie Prozessdaten aus PMS visualisiert werden können. Es wird aufgezeigt, welche Möglichkeiten bestehen, um:

- Prozessinformationen grafisch aufzubereiten
- Prozessinformationen mit veränderter Strukturierung darzustellen

Außer Optimierungen der grafischen Repräsentation kann also versucht werden, die Prozessinformationen aus einer anderen Sichtweise zu betrachten. Damit ist die Darstellung eines Prozesses nicht mehr auf die Struktur festgelegt, in der er modelliert wurde.

Weiterhin wird gezeigt, wie Prozessinformationen mit organisatorischen Strukturdaten (Organisationsmodell) angereichert werden. Ein weiteres Ziel besteht darin, möglichst viele zusammenhängende Informationen aus den verfügbaren Prozessdaten zu extrahieren. Dabei bildet die Anzeige von Prozessdaten also die reine Visualisierung und nicht deren Editieren, den Fokus. Natürlich sind adäquate Visualisierungsformen von Prozessen die Grundvoraussetzung, um diese effektiv bearbeiten/editieren zu können.

Durch die in den letzten Jahren steigende Bedeutung von Benutzerschnittstellen als Ansatzpunkt für Verbesserungen bei Produktivität und Nutzerakzeptanz gewinnen Usability-Betrachtungen mehr und mehr Gewicht. Entsprechende Aspekte sowie Erkenntnisse der Kognitionsforschung können als Basis dienen, um an die Bedürfnisse der Benutzer angepasste Visualisierungskonzepte zu entwickeln.

Ausgangspunkt der nachfolgenden Betrachtungen ist ein Architekturmodell wie es Abbildung 1.1 zeigt. Eine Prozessvisualisierungskomponente greift auf eine globale Datenbasis mit Prozessinformationen aus einem heterogenen Verbund prozessorientierter Informationssysteme zu. Das in der Abbildung gezeigte Process-Warehouse integriert Prozessdaten, die teilweise auf unterschiedlichen Prozess-Metamodellen basieren. Im Idealfall speichert diese Datenbank auch die passenden Organisationsmodelle.

Diese Arbeit leistet Vorarbeiten für die Implementierung einer Prozessvisualisierungskomponente. Eine solche Komponente basiert auf einem Prozess-Metamodell. Dies wird hier nicht explizit festgelegt, denn im praktischen Einsatz ist es nicht sinnvoll, auf ein starres Modell zu setzen. Andernfalls wäre die Darstellung von Prozessen, die auf komplexeren Prozessmodellen basieren, eingeschränkt. Die hier vorgestellten Visualisierungskonzepte sind daher weitgehend unabhängig von den Ausdrucksmöglichkeiten des zugrunde gelegten Prozess-Metamodells. Erst für die tatsächliche Implementierung der Visualisierungskomponente ist es notwendig, genauere Modelldetails, wie darzustellende Kantentypen oder unterstützte Verzweigungen festzulegen.

Vision dieser Diplomarbeit ist es aus (Geschäfts-) Prozessen das enthaltene Wissen zu extrahieren und falls möglich, mit Wissen aus einem Organisationsmodell angereichert auf vielfältige Weisen darzustellen.

Eine Visualisierungskomponente ist im übrigen keine Monitoring-Komponente, wenngleich sich sicherlich einige Funktionen überschneiden können. Diese Abgrenzung ist notwendig, da diese beiden Anwendungstypen teilweise gegensätzliche Zielsetzungen haben. Ziel von Monitoring ist in erster Linie die Analyse laufender Prozessinstanzen. Hierzu dient die

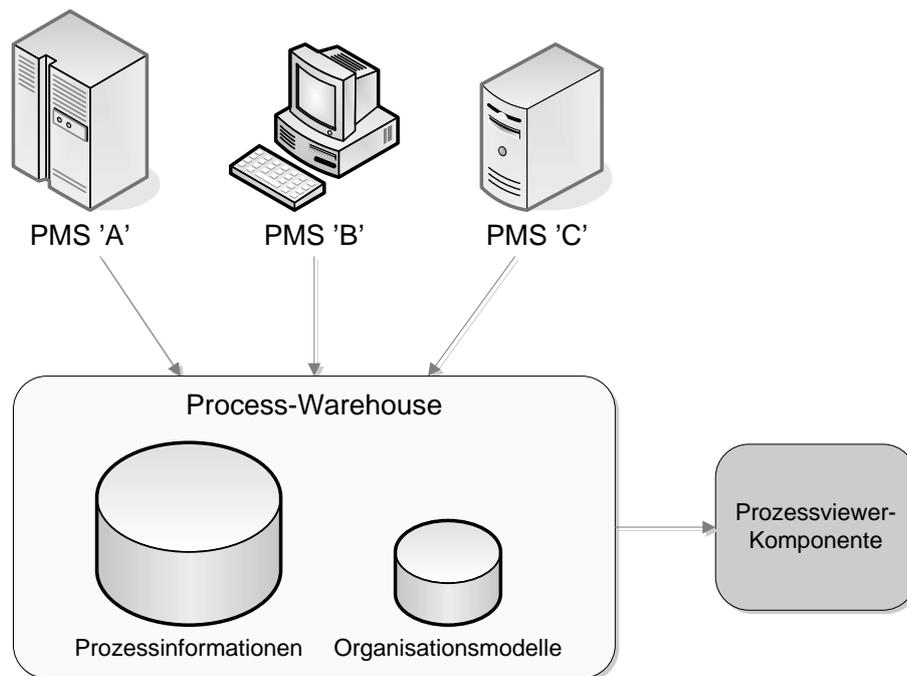


Abbildung 1.1.: Architekturmodell Process-Warehouse

(Echtzeit-)Erfassung von aussagekräftigen Kennzahlen über Durchsatzzeiten, Kostenentwicklungen und Ähnliches. Dadurch lassen sich frühzeitig Flaschenhälse bei der Prozessausführung erkennen. Das Ziel von Visualisierung ist die ansprechende und übersichtliche Darstellung einzelner Prozess-Modelle oder -Instanzen. Überschneidungen zum Monitoring ergeben sich, wenn Übersichtsdarstellungen vieler Prozesse erzeugt werden.

GLIEDERUNG

Hauptgegenstand dieser Arbeit bilden Visualisierungskonzepte für Prozessdaten. Diese bilden das Herz einer künftigen Prozessvisualisierungskomponente. In Kapitel 2 wird zunächst eine Übersicht dazu gegeben, auf welche Arten Prozesse visualisiert werden können und welche Prozessinformationen auftreten. Daran schließt sich eine Einführung über Ansätze zu Visualisierungskonzepten von Prozessdaten an. Es folgt eine Anforderungsanalyse in Kapitel 3. Hier werden mögliche Benutzer einer Prozessvisualisierungskomponente kategorisiert. Die jeweilige Benutzergruppe ist durch spezifische Motivationen oder Benutzerziele charakterisiert. Die Entwicklung und Systematisierung von Visualisierungskonzepten beginnt also mit der Analyse der Anforderungen der typischen Benutzer in einem Prozessumfeld. Kapitel 4 wirft einen Blick auf bereits am Markt erhältliche Programme und die dort anzutreffenden Visualisierungslösungen sowie auf den Stand der Forschung in den Bereichen Visualisierung, Usability, Wahrnehmungsforschung und Interface-Design. Den Kern der Arbeit, die Visualisierungskonzepte, bilden die zwei folgenden Kapitel. Es geht einerseits um grafische Aspekte und andererseits um Darstellungsformen. Beide Kapitel verfolgen unterschiedliche Ansätze zur Verbesserung der Prozessvisualisierung, deren Ergebnisse sich für die endgültige Darstellung kombinieren lassen. Kapitel 5 widmet sich der Frage, wie man durch Änderungen an der grafischen Darstellung, also durch Veränderung

von Farben, Formen und Symbolen Ansichten erzeugen kann, die auf die Belange der Benutzer zugeschnitten sind und ihnen übersichtlich das präsentieren, was in der aktuellen Situation sinnvoll ist. Zudem kommen die in Kapitel 4 zusammengeführten Erkenntnisse aus der Kognitions-, Wahrnehmungs- und Usability-Forschung zur Anwendung, um Informationsdarstellungen zu finden, die der menschlichen Wahrnehmung möglichst gut entsprechen. Die besprochenen grafischen Aspekte werden im Kapitel 6 wieder aufgegriffen. Eine Visualisierungskomponente für Prozessinformationen sollte eine möglichst breite Palette an Darstellungsmöglichkeiten der Daten bieten. In Kapitel 6 werden verschiedene Ansichten für Prozessdaten behandelt. Bei jeder Darstellungsform bildet jeweils ein Aspekt aus den verfügbaren Prozessinformationen den Fokus der Darstellung. Insbesondere werden die entwickelten Ansichten hinsichtlich Restriktionen und spezifischer Probleme (z. B. bezüglich Layout) untersucht. Kapitel 7 beschreibt mögliche Benutzerinteraktionen auf Prozessdarstellungen, z. B. Historiendarstellung oder Navigation in Prozessen. Auch auf die View-Bildung wird eingegangen, die ein mächtiges Konzept darstellt unterschiedliche Sichten auf einen Prozess zu ermöglichen. Den Abschluss bildet Kapitel 8 mit Anmerkungen über offene Fragen und einem Fazit. Als roter Faden und Bewertungskriterium dient die Frage nach der Gebrauchstauglichkeit und die Anwendung von Erkenntnissen aus der Wahrnehmungsforschung.

1.3. Motivation

Ein Ziel dieser Arbeit ist es, die üblichen Prozessgraphdarstellungen zu verbessern. Abbildung 1.2 zeigt einen Ausschnitt aus einem Change-Management-Prozess in Business Process Management Notation (*BPMN*) [Bpmi03a]. Die Darstellung zeigt verschiedene untereinander verbundene Objekte: Aktivitäten, Ausführungseinheiten, Systeme und Dokumente. Jeder Objekttyp hat eine eindeutige Form und Farbe. Die schwarzen Kanten zwischen den Aktivitäten und die hellgrauen Kanten für Ausnahmebehandlungen bilden den Kontrollfluss ab. Die Aktivitäten am oberen Rand zeigen den Elternprozess. Schon in diesem kleinen Ausschnitt erscheint die Darstellung sehr unübersichtlich.

Abbildung 1.3 zeigt den gleichen Prozess, es wurden jedoch viele Verbesserungsvorschläge eingearbeitet, die in dieser Arbeit gemacht werden. Durch die Integration der verschiedenen Objekte zu einem Aktivitätenknoten reduziert sich die Anzahl von Symbolen und Farben. Die Struktur des Prozesses ist leichter erfassbar. Um die Übersichtlichkeit weiter zu steigern, wurde der Elternprozess ausgeblendet und die Dokumente an den oberen Rand verschoben. Die gesamte Darstellung wirkt ruhiger und aufgeräumter.

Üblicherweise werden Informationen über Prozesse als Prozessgraphen dargestellt. Diese stellen aber nur eine mögliche Darstellungsform von Prozessinformationen dar. Denn außer der Optimierung der grafischen Repräsentation ist die Zielsetzung dieser Arbeit, die in den Prozessdaten enthaltenen Informationen zu nutzen, um dem Benutzer auch alternative Darstellungsformen anzubieten. So lassen sich Zeitinformationen nutzen, um einen Prozess in Form einer Kalenderansicht darzustellen.

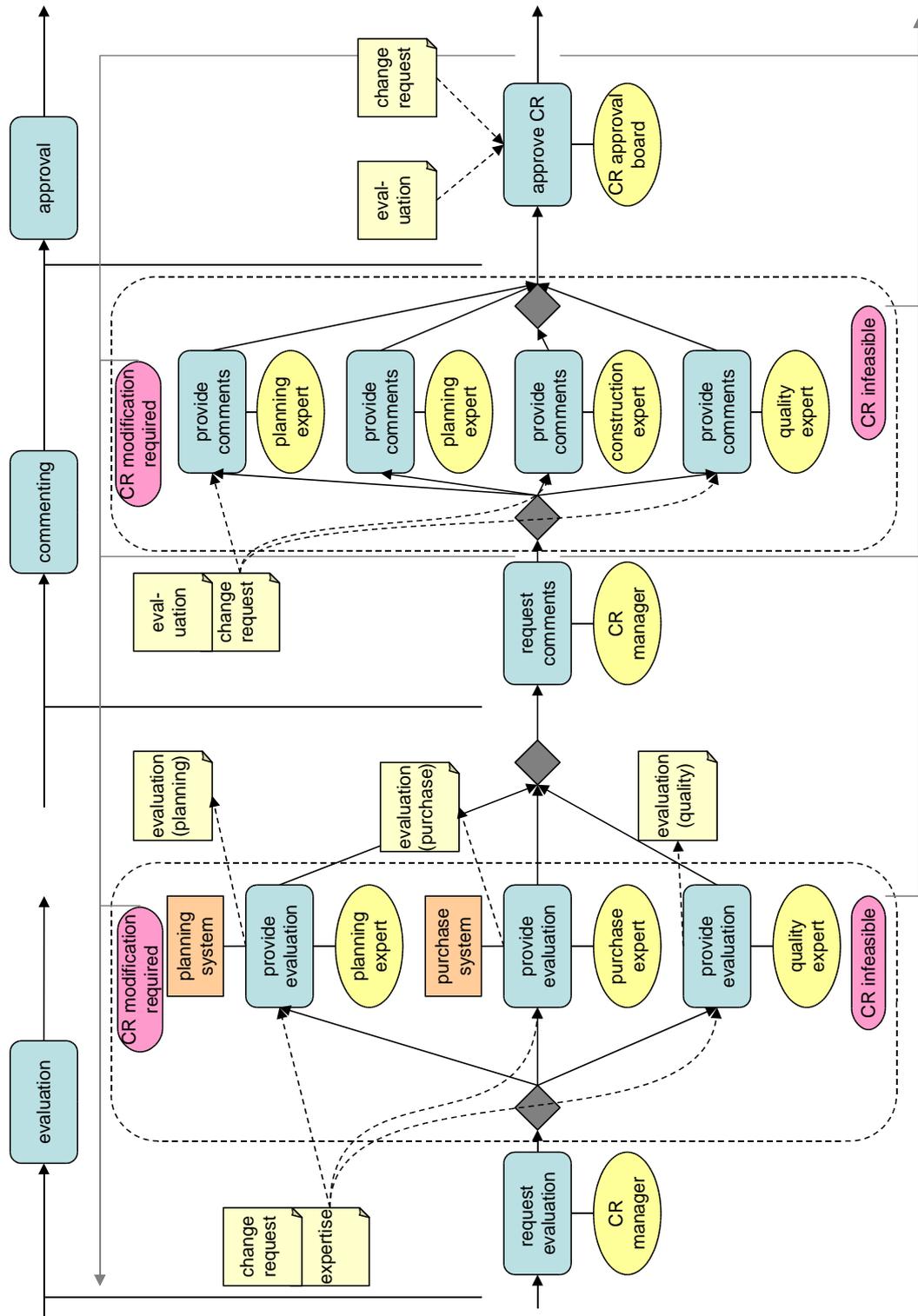


Abbildung 1.2.: Beispielprozess Change Management – Ausschnitt alte Version

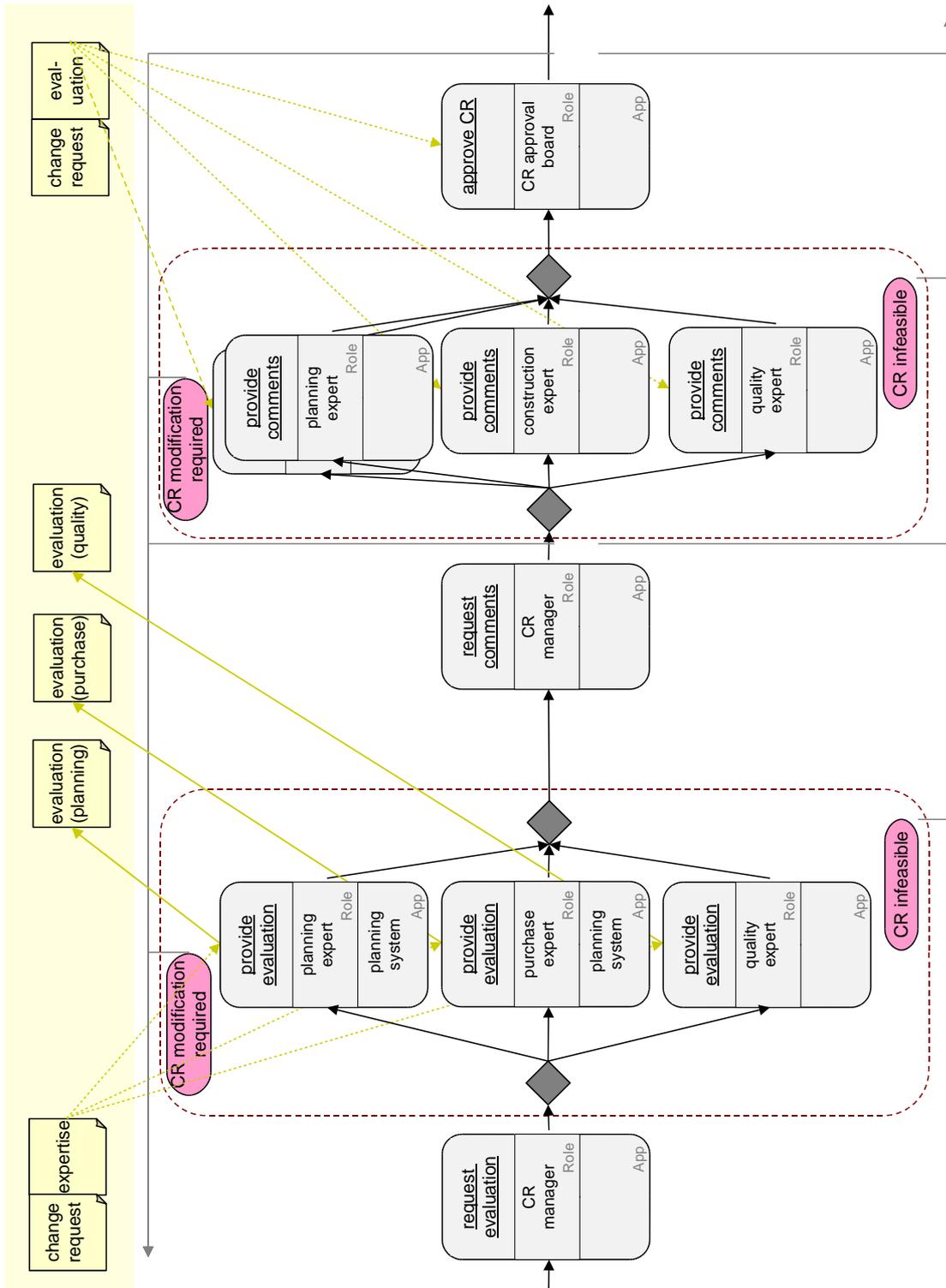


Abbildung 1.3.: Beispielprozess Change Management – Ausschnitt neue Version

2

Grundlagen und Einführung

Dieses Kapitel führt in das Thema Prozessvisualisierung ein. Daher wird zunächst darauf eingegangen, aus welchen Einzelinformationen Prozessdaten bestehen und wie sie strukturiert sind. Anschließend wird darauf eingegangen, welche Darstellungsarten für Prozesse benötigt werden. Prozesse können beispielsweise zur Laufzeit mit Ausführungszuständen oder nur als schematisches Modell dargestellt werden. Einzelne Prozesse können dargestellt werden oder Gruppen von Prozessen. Am Ende des Kapitels wird die grundsätzliche Idee vorgestellt, wie sich die Prozessvisualisierung verbessern lässt und auf welchem Weg ein umfassendes Konzept für die Aufbereitung von Prozessdaten gefunden werden kann.

2.1. Visualisierung von Prozessdaten

Für die Prozessvisualisierung erscheint es unerheblich, wie die im Prozesskontext vorkommenden Informationen zusammenhängen oder in welcher Form sie gespeichert sind.

Aus dem Szenario einer unternehmensweiten Prozesshierarchie, mit Prozessen, die von unterschiedlichen PMS verwaltet werden, ergeben sich für eine Visualisierungskomponente wichtige Punkte. Für den Benutzer soll es transparent sein, auf welchem PMS die Prozesse laufen. Weiterhin soll der Benutzer in großen Prozessen navigieren können.

Am Anfang der Visualisierung steht die Frage, aus welchen Daten Prozessinformationen bestehen und welches explizit darin kodierte Wissen wie dargestellt werden soll.

Mindestanforderung für die Visualisierung von Prozessen sind:

- Prozessmodell
- Datenmodell
- Visualisierungsmodell

Optionale Informationen sind Strukturdaten wie Organisationsmodelle. Ein Process-Warehouse kann die Prozessinformationen zu Prozess-, Daten- und Organisationsmodell sammeln und für die Visualisierung bereitstellen (siehe Abbildung 2.1).

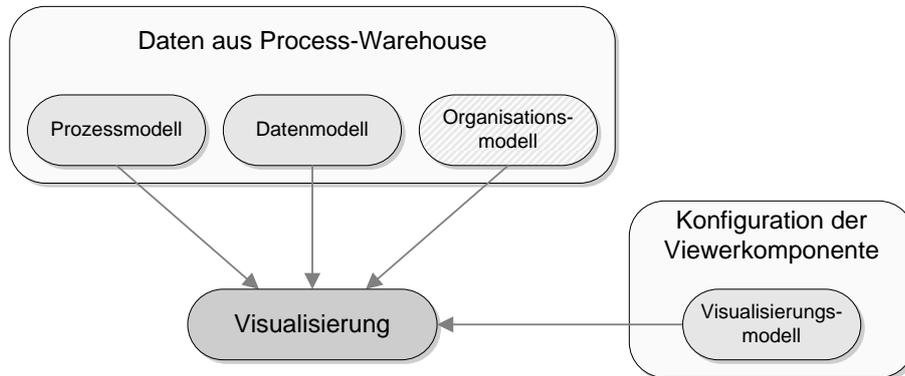


Abbildung 2.1.: Schematischer Ablauf der Prozessvisualisierung

Das Prozess-(Meta)Modell gibt den Rahmen vor, wie Prozesse modelliert werden können. Es legt fest, welche Arten von Verknüpfungen es zwischen Aktivitäten gibt. Es definiert, welche Ausführungszustände es gibt und legt fest, wie ein- und ausgehende Datenflüsse modelliert werden. Das Datenmodell legt die Attribute fest, die in Aktivitäten hinterlegt werden.

Das Visualisierungsmodell gehört nicht zu den Prozessdaten, sondern es legt das Aussehen ihrer Visualisierungsdaten fest. Ein wesentliches Merkmal des Visualisierungsmodells besteht darin, dass es nicht statisch ist. Vielmehr liegt das Augenmerk auf der Anpassbarkeit für die jeweiligen Benutzer. Es ist auch nicht statisch, sondern lässt sich an die Bedürfnisse der Benutzer anpassen.

Einem Prozess liegt ein Schema zugrunde. Zur Definition eines solchen Schemas gehört die Beschreibung eines Netzplanes, der aus Knoten (Aktivitäten) und Kanten (Weiterschaltungslogik) besteht. Weiterhin besitzt ein Schema verschiedene Attribute (z. B. Prozessverantwortlicher, Start-/Endzeit oder zugeordnetes Produkt/Objekt).

Das Datenmodell legt fest, welche zugeordneten Attribute ein Schema enthält. Einigen dieser Attribute werden erst dann Werte zugeordnet, wenn ein Prozess ausgeführt bzw. instanziiert wird.

Instanziierte Prozesse werden Instanzen genannt. Das PMS übernimmt die Koordination über die Ausführung der Instanzen.

Nach dieser kurzen Einführung in die Struktur der Prozesse ist noch anzumerken, dass die Aktivitäten in den Prozessen wiederum ganze Prozesse kapseln können, auf diese Art und Weise und auch durch Sprünge im Kontrollfluss zu anderen Prozessen entstehen *hierarchische Workflows*.

2.1.1. Prozessinformationen

Viele Informationen eines Prozesses stecken in seinen Aktivitäten. Und diese Aktivitäten sind letztlich Datenstrukturen, die Attribute haben wie Name der Aktivität, Beschrei-

bungstext, Start-/Endzeit, Frist (engl. *Deadline*), Kosten und Ähnliches. Weiterhin werden jeder Aktivität Objekte mit Dokumenten und Daten, Ressourcen und eine oder in Sonderfällen auch mehrere Ausführungseinheiten zugeordnet (z. B. bei Abstimmungen oder dem Vieraugenprinzip).

Der Kontrollfluss bildet die Ausführungsabfolge der einzelnen Aktivitäten ab. Er ist explizit durch die Struktur des Netzplanes definiert. In den Kanten sind (durch die Kantentypen) alle weiteren Informationen gespeichert, um den Rest des Kontrollflusses abzubilden, zum Beispiel bedingte Verzweigungen, Synchronisierungskanten, Schleifen und Ausnahmebehandlungen.

Die Informationen über die Dokumente und Daten bilden den Datenfluss. Er ist je nach verwendetem Metamodell implizit oder explizit modelliert. Wenn der Datenfluss implizit gegeben ist, werden die ein- und ausgehenden Dokumente der Aktivitäten im Hinblick auf den Kontrollfluss ausgewertet.

Die Aktivitäten werden selten komplett mit allen dazugehörigen Informationen dargestellt, da dies zu unübersichtlich wäre. Dem Benutzer wird nur das angezeigt, was er aktuell zur Erreichung seines Zieles benötigt. Dies impliziert eine flexible Konfigurierbarkeit der Darstellung. Der Name einer Aktivität wird nahezu immer angezeigt, alle anderen Prozessinformationen sind mehr oder weniger optional.

Die Prozessinformationen setzen sich also aus folgenden Einzelteilen zusammen, die über die Visualisierung letztendlich zugänglich sein sollen:

➤ **Aktivitäten**

Der Benutzer soll erkennen können, aus welchen Aktivitäten sich ein Prozess zusammensetzt.

➤ **Objekt Ausführungseinheit**

Jeder Aktivität ist eine ausführende Einheit zugeordnet (in Spezialfällen, wie dem 4-Augenprinzip oder Abstimmungen, auch mehrere). Dieser Bearbeiter der Aktivität kann ein Individuum, ein automatisches System, eine Rolle/Funktion und/oder eine Organisationseinheit sein.

➤ **Objekt Ressourcen**

Jeder Aktivität sind Ressourcen zugeordnet, z. B. Applikationen und Systeme, die während der Ausführung der Aktivität verwendet werden.

➤ **Objekt Dokumente/Daten**

Jeder Aktivität sind ein- und ausgehende Datenverbindungen zu Datenbanken oder Dokumenten zugeordnet, auf die während der Ausführung der Aktivität zugegriffen wird. Es ist Aufgabe des Metamodells dabei Unterscheidbarkeit herzustellen zwischen Daten die für jede Instanz extra anfallen (lokale Dokumente oder Daten) und globale Daten auf die jede Instanz zugreift.

➤ **Kontrollfluss**

Abfolge der Aktivitäten, Informationen über bedingte Ausführung, Schleifen, Ausnahmen (engl. *exceptions*), Aufrufe von anderen Prozessen in hierarchischen Prozessen (Verweise/Sprünge zu anderen Prozessen) und Synchronisationsinformationen (Synchronisationskanten).

➤ **Datenfluss**

Der Datenfluss ist meist durch den Kontrollfluss implizit definiert. Er zeigt den Weg der Daten und Dokumente durch den Prozess.

➤ **Aktivitätenattribute aus dem Datenmodell**

Den Aktivitäten werden verschiedene Attribute, z. B. Zeitangaben, Priorität, Kosten, usw. zugeordnet.

➤ **Prozessattribute aus dem Datenmodell**

Auch Prozesse haben diverse Attribute, wie z. B. Zeitangaben, Prozessverantwortlicher, ein zugeordnetes Produkt oder Objekt, u. a.

➤ **Zusammenhänge**

Zwischen allen genannten Informationen bestehen auch Zusammenhänge, die nicht explizit kodiert sind, die jedoch implizit mit in den Prozessinformationen stecken. Hierfür gilt es Werkzeuge zu entwickeln, die diese Zusammenhänge sichtbar machen können.

Hinter der Information zur Ausführungseinheit muss nicht zwangsläufig ein menschlicher Bearbeiter stecken. Prozessschritte können sich auf manuelle Tätigkeiten oder auf automatische Abläufe beziehen. Es existieren die folgenden vier Arten von Ausführungseinheiten [WfMC99]:

➤ **System**

Ein automatisches System ist für den Prozessschritt verantwortlich, dies kann ein Computersystem oder eine Maschine sein.

➤ **Person**

Meist sind die Ausführungseinheiten Mitarbeiter. Selten wird einer Aktivität direkt eine bestimmte Person zugeordnet. Oft wird der einer Aktivität zugeordnete Bearbeiter indirekt adressiert, z. B. über seine Rolle oder Funktion und/oder die Organisationseinheit, der er angehört. Im Folgenden wird nicht mehr zwischen System und Person unterschieden, der Einfachheit halber wird nur von Bearbeitern gesprochen.

➤ **Rolle (Funktion)**

Eine Ausführungseinheit kann über eine im Organisationsmodell enthaltene Rolle beschrieben werden. Alternativ können auch Fertigkeiten oder Attribute aufgelistet sein, die die Person erfüllen muss, damit das PMS dementsprechend eine Person auswählt, die diese Aktivität ausführen soll. Eine Rolle ist eine Funktion, die eine Person innerhalb einer Organisation innehat. Hier geschieht bei der Beschreibung der Ausführungseinheit einer Aktivität keine konkrete Zuordnung zu einer Person. Es ist die Aufgabe des PMS, zur Laufzeit eine konkrete Zuordnung zu bestimmen. Es kann auch ein Koordinator definiert werden, der benachrichtigt wird, falls die Zuordnung zu einer konkreten Person durch das PMS scheitern sollte, weil z. B. alle möglichen Bearbeiter momentan belegt sind. Meist geschieht die Zuordnung zur Laufzeit durch das PMS nicht konkret, sondern alle zur Ausführung in Frage kommenden Personen erhalten den Eintrag zur Ausführung dieser Aktivität in ihrer Arbeitsliste. Der Erste, der diese Aktivität dann tatsächlich über-

nimmt, wird Bearbeiter. Bei allen anderen wird der Eintrag wieder aus der Arbeitsliste entfernt.

➤ **Organisationseinheit**

Organisatorischen Einheiten wie Abteilungen können Prozessschritte zugeordnet sein, in diesem Fall bekommen alle Mitglieder dieser Organisationseinheit die Tätigkeit in ihre Arbeitsliste, bis der Erste den Auftrag annimmt. Alternativ kann auch der Abteilungsleiter die Aktivität zugeteilt bekommen, um sie an eine bestimmte Person zu delegieren.

Die Angabe einer Rolle als Ausführungseinheit hat den Vorteil, dass die Zuweisung von Aufgaben effizienter geschehen kann als bei einer konkreten Personenzuordnung. Denn Rollen sind in einer Organisation meist nicht eindeutig einer Person, sondern einer Gruppe von Personen mit denselben Fähigkeiten zugeordnet. Somit kann z. B. die Person mit der geringsten Arbeitsbelastung die Aktivität oder Aufgabe zugeordnet bekommen. Bei Krankheit oder anders bedingten Ausfällen gerät die Ausführung des Prozesses so nicht ins Stocken, da nicht nur eine Person zur Ausführung in Frage kommt. In solchen Fällen ist es aber trotzdem sinnvoll, wenn das Organisationsmodell auch Vertreter für Personen oder Rollen definiert. Erwähnenswert ist in diesem Zusammenhang der Worst-Case, wenn alle Vertreterregelungen fehlschlagen. Dann muss das Schema des Prozesses mit Techniken der Schema-Evolution [Rind04] im laufenden Betrieb geändert werden. Auch so etwas sollte langfristig direkt in einer Prozessvisualisierungskomponente ermöglicht werden.

Außer einer expliziten Rollenangabe ist es in vielen Systemen auch möglich, den späteren Bearbeiter durch einen Satz von Fähigkeiten zu beschreiben, welche für die Bearbeitung der Aufgabe notwendig sind. Es ist dann Aufgabe des PMS, die Zuordnung zu einer Rolle durchzuführen.

Bei der Visualisierung eines laufenden Prozesses kann dem Benutzer die Ausführungseinheit auf drei unterschiedliche Weisen repräsentiert werden. Je nach Erfordernis: als konkrete Personenzuordnung, als Rolle oder als Zugehörigkeit des Bearbeiters zu einer Organisationseinheit. Der letzte Fall ist besonders dann interessant, wenn von Interesse ist, welche Teile eines Prozesses, also welche Aktivitäten, welchen unterschiedlichen Organisationseinheiten zugeordnet sind.

Wo der Benutzer die konkrete Personenzuordnung als Ausgabe wünscht, kann es den Fall geben, dass bei noch nicht gestarteten Aktivitäten als Ausführungseinheit eine Rolle oder eine Organisationseinheit angegeben ist. Falls nun diese Vorgaben durch das Organisationsmodell nicht eindeutig einer Person zuordenbar sind, muss auf die Ausgabe der Rolle zurückgegriffen werden, bis das PMS eine konkrete Zuordnung vorgenommen hat.

Da eine Visualisierungskomponente nicht nur Instanzen darstellen können soll, sondern auch Prozessschemata, muss klar sein, welche Informationen schon vor der Instanziierung zur Anzeige zur Verfügung stehen. Tabelle 2.1 (→ S. 14) zeigt eine Auflistung der wichtigsten Prozessinformationen. Die Tabelle ist unterteilt in Informationen, die schon in einem Prozessschema gespeichert sein können und in Informationen die erst zur Laufzeit bestimmt werden. Letztlich sind aber erst nach Beendigung eines Prozesses alle instanzspezifischen Informationen komplett verfügbar. Die Attribute und die Objekte von Aktivitäten sind in der Tabelle zur besseren Unterscheidung noch einmal besonders mit 'ATT' oder 'OBJ' markiert.

Informationsaspekte	Beschreibung und Beispiele
Verfügbare Informationen eines Schemas	
<i>ATT</i> Aktivitätenname	(kurzer) Name, der die Funktion der Aktivität beschreibt.
<i>OBJ</i> Ausführungseinheiten	hier typischerweise nur Rolle, Funktion oder Organisationseinheit, keine Personen oder Systeme
<i>OBJ</i> Systeme	Für die Ausführung der Aktivität benötigte Systeme und Applikationen
<i>OBJ</i> Dokumente und Daten	lesende und schreibende Datenbankzugriffe oder beteiligte Dokumente
<i>ATT</i> Aktivitätenbeschreibungstext	Nähere textuelle Beschreibung der Aktivität
<i>ATT</i> Zeiten (aktivitätsbezogene)	typische Dauer, Fristen, spätester Startzeitpunkt
<i>ATT</i> Kategorie	Einteilung der Aktivitäten in semantische Klassen wie z. B. Ablagetätigkeit oder konzeptionelle Tätigkeit
<i>ATT</i> andere	weitere vom Datenmodell abhängige Aktivitätenattribute, z. B. Fixkosten
Kontrollfluss ^a	z. B. Ausführungsbedingungen bei alternativen Verzweigungen
Verfügbare Informationen einer Instanz	
<i>OBJ</i> Zugeordnetes Produkt oder Objekt	Produkte, z. B. verschiedene Fahrzeugmodelle ^b
<i>OBJ</i> Zugeordneter Kunde	Verschiedene Unternehmenskunden ^c
<i>OBJ</i> Ausführungseinheiten	typischerweise Personen oder auch (automatische) Systeme oder Organisationseinheit [Team, Projekt, (Arbeits-)Gruppe, Abteilung, Unternehmen, Organisation]
<i>OBJ</i> Prozessverantwortlicher	Person, die die Verantwortung über die Prozessinstanz hat
<i>ATT</i> Zeiten (aktivitätsspezifische)	tatsächliche Start-/Endzeit, Bereitzeit
<i>ATT</i> Zeiten (instanzspezifische)	Start-/Endzeit
<i>ATT</i> andere	weitere vom Datenmodell abhängige Aktivitätenattribute, z. B. aktuelle Kosten

^aDer Kontrollfluss ist weder Objekt, noch Attribut von Aktivitäten. Aktivitäten werden durch den Kontrollfluss verbunden

^bDiese Information ist für Multi-Instanz-Darstellungen wichtig, um die Instanzen nach Produkt gruppieren zu können

^cDiese Information ist für Multi-Instanz-Darstellungen wichtig, um die Instanzen nach Kunden gruppieren zu können

Tabelle 2.1.: Übersicht über die Prozessinformationen

Wenn in dieser Arbeit von Aktivität die Rede ist, ist meist nur der Name der Aktivität gemeint – also der Aktivitätsbezeichner – nicht das gesamte Datenobjekt mit zugeordneten Attributen. Ein anderweitiger Gebrauch des Wortes ist aus dem Kontext ersichtlich.

Eine sehr wichtige Feststellung ist, dass der eine Teil der Prozessinformationen aktivitäten- und der andere instanzspezifisch ist. Aktivitätsspezifische Informationen sind immer mit Aktivitäten verknüpft. Die instanzspezifischen Informationen sind dagegen mit einer In-

stanz verknüpft. Nur die Prozessattribute aus dem Datenmodell aus der obigen Auflistung sind instanzspezifische Informationen, alle anderen sind aktivitätsspezifisch. Erstaunlicherweise lassen sich diese beiden verschiedenen Arten von Prozessinformationen nun in den verschiedenen Darstellungskonzepten nicht mischen. Entweder ist eine Darstellung auf die Aktivitäten hin ausgerichtet oder auf die Instanzinformationen. Eine Prozessgraphdarstellung ist beispielsweise auf die Aktivitäten ausgerichtet und in der resultierenden Darstellung ist eigentlich kein Platz für den Prozessverantwortlichen, das zugeordnete Produkt oder Objekt oder die Startzeit.

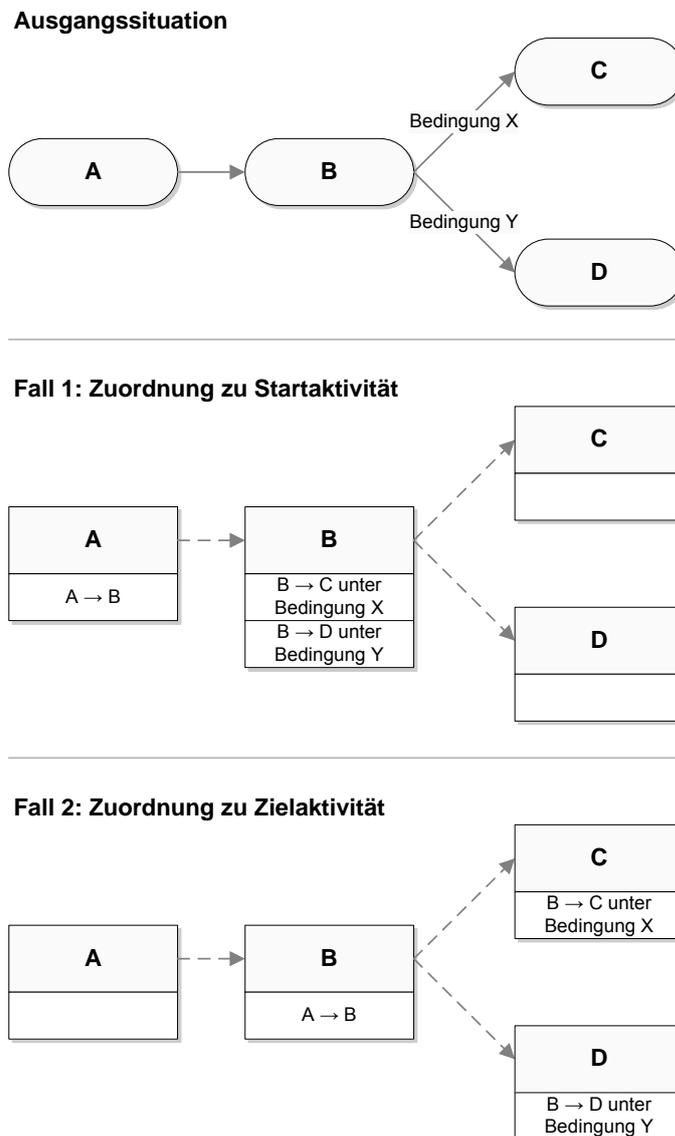


Abbildung 2.2.: Möglichkeiten der Zuordnung von Kontrollfluss-Informationen zu Aktivitäten

Die Darstellung des Kontrollflusses ist ein Problem bei Visualisierungsformen, die den Kontrollfluss nicht in Form von (beschrifteten) Kanten darstellen können, beispielsweise bei Tabellen, die nur Aktivitäten auflisten. Wie Abbildung 2.2 zeigt, können diese

Kontrollfluss-Informationen den Aktivitäten zugeordnet werden. Dadurch kann bei tabellenbasierten Darstellungen zumindest textuell auf den Kontrollfluss hingewiesen werden. Dafür müssen die Netzkanten, die zwei Aktivitäten miteinander verbinden, jedoch eindeutig einer der beiden Aktivitäten zugeordnet werden.

Die Kontrollfluss-Informationen lassen sich besser auf die Aktivitäten verteilen, wenn Bedingungen als Ausführungsbedingungen begriffen werden und sie der Aktivität am Ende der Kante zugeordnet werden. Bei einer großen Verzweigung im Prozessablauf müssen all die vielen Bedingungen nun nicht im Zusammenhang mit der einzelnen Aktivität, die am Beginn der Verzweigung steht, angezeigt werden. Auf diese Weise lassen sich auch leicht alle Aktivitäten in einer Tabelle markieren, an die Ausführungsbedingungen geknüpft sind, da nicht überprüft werden muss, ob in der Vorgängeraktivität eine Bedingung hinterlegt ist. Und im Fall von Verknüpfungen entscheidet ohnehin die Zielaktivität darüber, ob eine 'Und' oder 'Oder' Weiterschaltung erfolgt.

2.1.2. Darstellungsarten

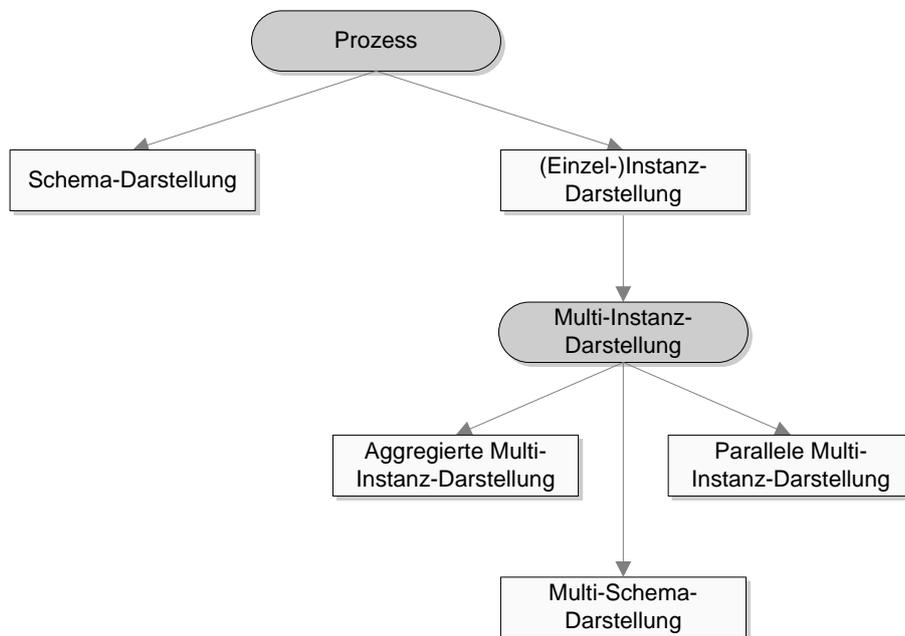


Abbildung 2.3.: Darstellungsarten für die Visualisierung

Abbildung 2.3 zeigt mit welchen Darstellungsarten die Prozessdaten aus einem PMS dargestellt werden können.

SCHEMA-DARSTELLUNG

Prozessschemata legen den ganzen Prozessablauf fest, der Kontrollfluss ist gemäß dem Prozess-Metamodell modelliert und dem Prozess und allen Aktivitäten sind Attribute aus dem Datenmodell zugeordnet. Die Attribute enthalten schon die notwendigen Werte, damit Instanzen die dem Schema zugeordnet sind sofort gestartet werden können. Unter Prozessvisualisierung wird meist die Darstellung bereits laufender Prozesse (also Instanzen) ver-

standen. Was ein Schema von einer Instanz unterscheidet, sind hauptsächlich die fehlenden Laufzeitinformationen. Eine Schema-Darstellung eignet sich dazu, Prozesse zu entwerfen, zu verändern oder zu analysieren. Sie dient dazu, Zusammenhänge zu verdeutlichen und sich einen Überblick über den logischen Ablauf zu verschaffen. Damit ist eine Schema-Darstellung bereits sehr aussagekräftig. Sie zeigt Aktivitäten sowie deren Eigenschaften (oder Attribute), zugeordnete Datenflüsse und Ressourcen (z. B. zur Ausführung von Aktivitäten benötigte Applikationen und IT-Systeme). Informationen zu Ausführungseinheiten umfassen üblicherweise Rollenangaben (aus dem Organisationsmodell); eine Zuordnung von konkreten Bearbeitern erfolgt üblicherweise erst zur Laufzeit. Prozessschemata können hierarchisch organisiert sein. In diesem Fall hat ein Schema Verbindungen in andere Teile der Prozesshierarchie. Einzelne oder alle Aktivitäten eines Schemas sind dann Platzhalter für Aufrufe von Teilprozessen. Für erfahrene Anwender sollten daher, aggregierte Aktivitäten von normalen Aktivitäten in der Darstellung unterscheidbar sein.

INSTANZ-DARSTELLUNG

Auf Instanzebene werden eventuell angegebene Rollen bei den Ausführungseinheiten zu konkreten Bearbeiterinformationen aufgelöst. Nicht ausgeführte Pfade im Workflow können zur besseren Übersichtlichkeit ausgeblendet werden. Dadurch ist auch der Datenfluss genauer bestimmt, da er vom Kontrollfluss abgeleitet wird. Die Aktivitäten erhalten zusätzlich Zustandsmarkierungen (z. B. gestartet, abgeschlossen) und die Prozesshistorie kann dargestellt werden.

MULTI-INSTANZ-DARSTELLUNG

Falls mehrere Instanzen eines Prozessschemas vorliegen, kann der Benutzer eine Multi-Instanz-Darstellung wünschen. Hier sollen Übersichtsdarstellungen für den nötigen Überblick sorgen. Oftmals ist auch die Rede von Prozess-Cockpit oder -Dashboard. Die einzelnen Prozessinstanzen können (je nach Prozess-Metamodell) folgende Zustände haben:

- gestartet
- nicht gestartet
- pausiert
- abgeschlossen

Diese Zustände und der jeweilige (Ausführungs-)Fortschritt der Instanzen können visualisiert werden. Es wird zwischen zwei Darstellungsvarianten unterschieden. Instanzen können entweder 'aggregiert' oder 'parallel' dargestellt werden. Eine aggregierte Darstellung zeigt das (Prozess-)Schema dieser Instanzen, angereichert um Informationen zum jeweiligen Ausführungsfortschritt der Instanzen. Details einzelner Instanzen können dabei nicht angezeigt werden, nur aggregierte Werte wie beispielsweise Durchschnittskosten. Eine parallele Darstellung listet alle Instanzen (parallel) untereinander auf. Die Informationen zu jeder einzelnen Instanz werden somit separat visualisiert. Einzelne Prozessinformationen müssen somit nicht aggregiert werden. Allerdings müssen die einzelnen Instanzen in platzsparender komprimierter Form für diese Art der Darstellung aufbereitet werden (z. B. als Meilenstein-Ansicht). Der Benutzer hat die Möglichkeit, sich aus einer solchen Multi-Instanz-Darstellung gezielt eine einzelne Instanz zu wählen, um sich diese mit allen Details anzeigen zu lassen.

MULTI-SCHEMA-DARSTELLUNG

Einen Spezialfall und eine Schwierigkeit für die Visualisierung stellt die Anforderung dar, Multi-Schema-Darstellungen zu unterstützen. Denkbare Anwendungsfälle ergeben sich überall dort, wo an einem Produkt oder Objekt mehrere unterschiedliche Prozesse beteiligt sind. Eine einzige Darstellung ermöglicht so dem Benutzer eine Übersicht über diese Prozesse.

Ein Beispiel wären verschiedene Change-Management-Prozesse, die für verschiedene Teile eines Gesamtprodukts initiiert wurden. Die einzelnen Change-Management-Prozesse können dabei für jedes Bauteil unterschiedlich aussehen. Eine Übersichtsdarstellung zeigt dann alle laufenden Change-Management-Prozesse dieses Produkts. Es können dabei natürlich mehrere Instanzen eines Prozessschemas laufen.

Nativ lässt sich so etwas nur in einer Kombination von aggregierter und paralleler Multi-Instanz-Darstellung visualisieren (siehe letzter Abschnitt). Eine besonders übersichtliche Darstellung ergibt sich, wenn die verschiedenen Prozessschemata auf eine gemeinsame View abgebildet werden. Verschiedene Change-Management-Prozesse lassen sich so beispielsweise auf die Grundelemente Initialisierung, Evaluation, Realisierung und Abschluss zurückführen. Damit wird eine Multi-Schema-Darstellung auf eine Multi-Instanz-Darstellung zurückgeführt. Damit können Multi-Schema-Darstellungen auch mit Darstellungskonzepten angezeigt werden, die nur den Multi-Instanz Fall beherrschen.

SCHEMA-VERSIONEN-DARSTELLUNG

Ganz am Rande der Betrachtungen dieser verschiedenen Darstellungsformen bleibt noch eine ganz andere Visualisierungsanforderung zu erwähnen: Die Gegenüberstellung von modifizierten Prozessschemata erfordert eine gesonderte Herangehensweise. Unter anderem durch die Schema-Evolution entstehen neue Versionen von einzelnen Schemata. Um die Änderungen von einer Version zur Nächsten nachzuvollziehen, wird eine besondere Darstellung benötigt. Hierfür bietet sich eine Prozessgraphdarstellung an. Die Schema-Version mit den meisten Knoten bietet hierfür die Grundlage, die restlichen Versionen werden dazu in Beziehung gesetzt. Farbcodierungen der Knoten und Kanten zeigen die Versionsänderungen.

2.1.3. Organisationsmodell

PMS referenzieren im Allgemeinen ein Organisationsmodell des Unternehmens. Üblicherweise ist damit nur das Organisationsmodell gemeint, das die Personalstruktur abbildet. Für die Prozessvisualisierung sind jedoch alle Informationen hilfreich, die es erlauben, Ressourcen zu kategorisieren/organisieren. Wenn also Modelle existieren, die z. B. die Struktur von IT-Landschaften erfassen, sind dies ebenfalls Organisationsinformationen, die im Kontext dieser Arbeit als Organisationsmodell aufgefasst werden.

Eventuell existiert auch für die Dokumente und Daten eines Unternehmens eine verwaltete organisierte Struktur, wie ein Dokumenten-Management-System. Mithilfe solcher Informationen ist es z. B. in Darstellungen möglich, Dokumente nach Kategorien zu gruppieren.

Wie viele solcher Modelle in der Realität vorliegen, ist nicht entscheidend. Es geht hier um die Idee, dieses Wissen über die Umgebung in der die Prozesse laufen für die Visua-

lisierung zu nutzen. Deswegen werden in dieser Arbeit alle Modelle unter dem Namen Organisationsmodell gebündelt.

Das Organisationsmodell lässt sich als hierarchisch gegliedertes Ressourcenverzeichnis beschreiben. Personen aber evtl. auch IT-Systeme, Maschinen und Dokumente sind dabei in einer '1:n'-Beziehung den verschiedenen Organisationsstrukturen des Unternehmens oder auch zu externen Partnern zugeordnet. Innerhalb des Unternehmens sind das üblicherweise Zuordnungen zu Abteilungen, (Projekt-)Teams und (Arbeits-)Gruppen. Das Modell enthält auch Rollen, sie spezifizieren die verschiedenen Funktionen innerhalb der Organisation. In einem Labor gibt es z. B. die Rollen Arzt, MTA und Chemiker. Einer Rolle können einzelne oder mehrere Personen zugeordnet werden und je nach Komplexität des Modells sogar Prioritäten und Zeitbedingungen. Auch Stellvertreterrollen oder -Regelungen können definiert sein. Den Aktivitäten eines Prozessschemas sind solche Rollen zugeordnet. Anhand der Informationen aus dem Organisationsmodell kann dann das PMS zur Laufzeit den Aktivitäten konkrete Bearbeiter zuordnen.

Für die Prozessvisualisierung sind Organisationsmodelle relevant, weil es durch die in den Modellen gespeicherten Informationen möglich ist, Prozessdaten damit anzureichern. Damit wird es möglich, alle Aktivitäten, die von Mitarbeitern einer Abteilung bearbeitet werden, gesondert zu markieren. Dies kommt der Übersichtlichkeit zugute, wenn der Benutzer des Systems beispielsweise nur an den Teilen des Prozesses interessiert ist, an denen nur diese Abteilung beteiligt ist.

2.2. Ansätze zur Prozessvisualisierung

Der vorangehende Abschnitt hat eine Übersicht der zu visualisierenden Prozessinformationen gegeben. Ausgangspunkt der nachfolgenden Betrachtungen bildet das Konzept Prozessgraphdarstellung. Dieses wird analysiert, um weitere Darstellungsformen zu finden, die dessen Nachteile nicht teilen. Auch sollen für neue Darstellungsformen andere Prozessaspekte als die Prozessstruktur im Vordergrund stehen.

Abbildung 2.4 illustriert die Herangehensweise dieser Arbeit, um zu einer besseren Visualisierung von Prozessinformationen zu gelangen. Sie zeigt, zwei grundlegende Ansätze, die Prozessvisualisierung, ausgehend von der Prozessgraphdarstellung, zu optimieren. Der erste Ansatz beruht darauf, die grafische Darstellung zu optimieren. Der zweite Ansatz besteht darin, die bestehende Struktur der Prozessdaten zu ändern und sie anders darzustellen. Wichtig hierbei ist insbesondere die Feststellung, dass es sich hier um orthogonale Konzepte handelt. Das heißt, auch veränderte Strukturen profitieren von der Optimierung der grafischen Repräsentation. Beide Ansätze werden daher im Folgenden kombiniert.

Diese Arbeit widmet sich Verfahren der automatischen Generierung von Prozessgrafiken. Diese Übersicht über Ansätze zur Prozessvisualisierung wäre allerdings ohne die Erwähnung von dynamischen Darstellungen nicht vollständig.

Dynamische Darstellungen sind Grafiken mit dynamischen Inhalten denen Prozessdaten zugeordnet sind. Auf diese Weise können Arbeitsabläufe eines Prozesses besonders realitätsnah dargestellt werden. Der Benutzer muss eine Darstellung nicht mehr interpretieren, wenn sie genau seinem Gedankenmodell entspricht. Prozesse eines Verkehrsleitstandes lassen sich beispielsweise auf der Grundlage eines Liniennetzplanes wie in Abbildung 2.5

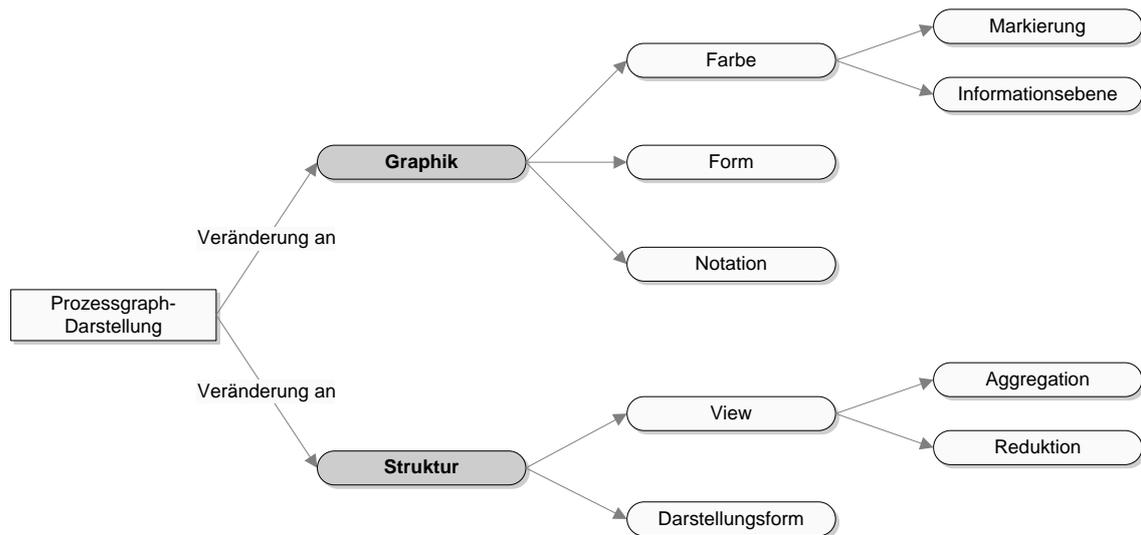


Abbildung 2.4.: Konzepte zur Visualisierung

visualisieren. Produktionsabläufe können anhand einer schematischen Darstellung der Fabrik dargestellt werden. Allerdings erfordern dynamische Darstellungen eine manuelle Zuordnung von Prozessdaten auf dynamische Objekte einer Darstellung. Dennoch kann sich der Aufwand zur Erstellung einer dynamischen Darstellung lohnen.

2.2.1. Veränderungen an der Grafik

In diesem und dem folgenden Abschnitt werden die beiden wichtigsten Herangehensweisen an die Aufgabenstellung beschrieben. Gleichzeitig wird in die weitere Strukturierung der Arbeit eingeführt:

Kapitel 5 widmet sich der Frage, für welche Komponente einer grafischen Darstellung Verbesserungen erzielbar sind und was beachtet werden sollte, um benutzerfreundliche Darstellungen zu erzeugen. Dazu wird auf die in Kapitel 4 zusammengestellten Forschungsergebnisse zurückgegriffen. Dieses Kapitel behandelt Teilaspekte wie den Einsatz von Farben und deren Anwendung in Form von Kontrasten. Beides kann verwendet werden, um verschiedene Informationsebenen in Darstellungen einzuziehen. Weiterhin wird für eine Prozessgraphdarstellung eine Notation benötigt, die die Konventionen festlegt, in welcher Form die Informationen aus den Prozessdaten in eine grafische Darstellung umgesetzt werden sollen.

2.2.2. Veränderungen an der Struktur

Der Ansatz, die grafische Darstellung von Prozessansichten zu optimieren, lässt sich mit dem nachfolgend beschriebenen Ansatz, die Struktur der Prozessdaten zu verändern, kombinieren. Daher fließen die Erkenntnisse aus Kapitel 5 in die verschiedenen in Kapitel 6 präsentierten Darstellungsformen mit ein. Dabei lässt sich zwar nicht alles auf jede Dar-

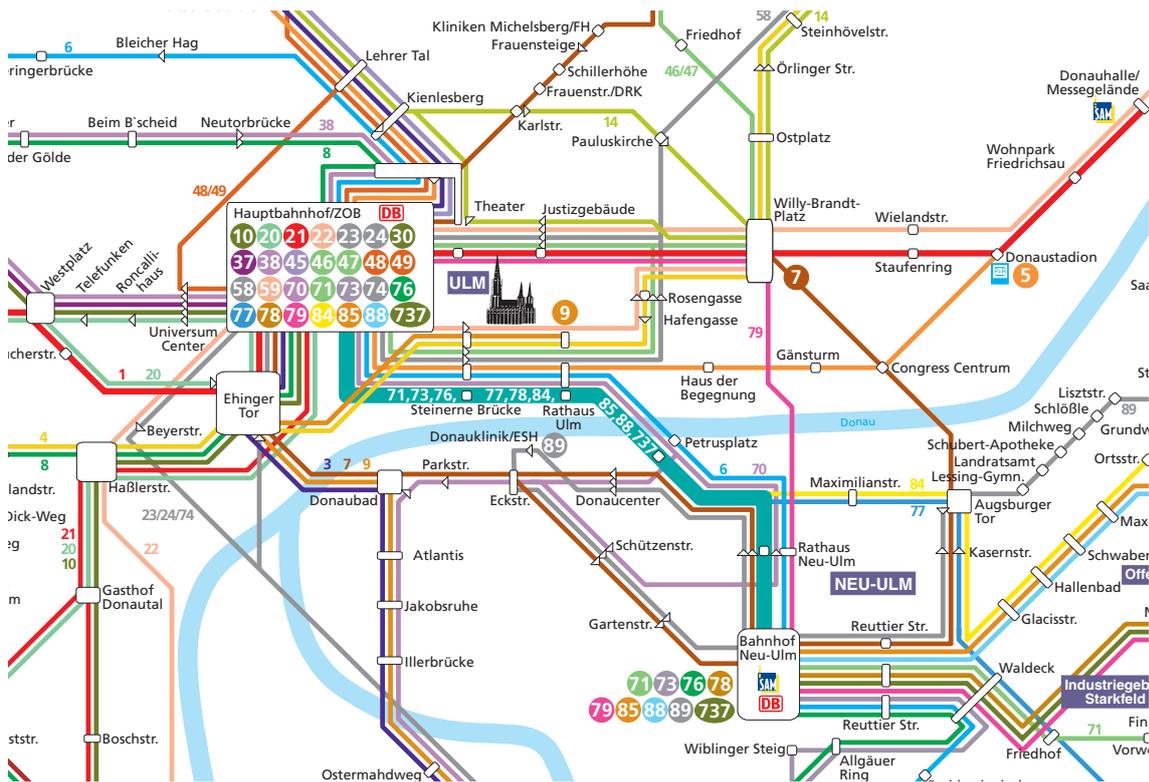


Abbildung 2.5.: Ausschnitt aus dem Netzplan des Donau-Iller-Nahverkehrsverbundes

stellungsform anwenden, aber vor allem die Ansichten, die eng mit der Prozessgraphdarstellung verwandt sind, orientieren sich stark daran.

Prozessdaten beschreiben einen gerichteten Graphen mit definiertem Start und möglicherweise mehreren Endpunkten. Die Prozessgraphdarstellung ist somit die natürlichste Form, der Prozessvisualisierung. Die Knoten eines Prozessgraphen stellen die Aktivitäten des Prozesses dar; jede Aktivität umfasst eine Menge von Attributen und zugeordneten Objekten, (z. B. die Ausführungseinheit). Dies entspricht der Sicht und dem (Gedanken-)Modell eines Prozessmodellierers.

An der Graphstruktur muss bei der Prozessdarstellung nicht starr festgehalten werden. Einerseits besteht die Möglichkeit Views (siehe Abschnitt 7.1) auf den Prozessdaten zu erzeugen, andererseits kann eine ganz andere Sichtweise auf die Prozessinformationen, zu einer Neu-Anordnung der Prozessdaten führen. Ein anderes Ziel als die Darstellung des Kontrollflusses, z. B. das Ziel Zeitaspekte zu visualisieren, führt so zu einer Kalenderdarstellung – die Struktur der Daten wird verändert. Die verschiedenen Benutzerziele, auf die im Kapitel 3 eingegangen wird, führen so zu neuen Darstellungsformen (siehe Goal-Directed-Design Abschnitt 3.1).

Kapitel 6 gliedert sich in Abschnitte zu den verschiedenen Konzepten. Diese Abschnitte sind jeweils nach dem gleichen Muster strukturiert. Nach einer kurzen Einführung folgt ein Beispiel. Danach wird die Mächtigkeit des Konzepts diskutiert. Es werden Fragen geklärt, wie gut sich Prozesse verschiedener Komplexität darstellen lassen, beispielsweise Prozesse mit Schleifen, hierarchische Prozesse oder mehrere Prozesse gleichzeitig, als Multi-

Instanz-Ansicht in aggregierter oder paralleler Form. Es wird behandelt, welche Informationsaspekte die Darstellung abdeckt, etwa in Bezug auf Aktivitätendetails, Kontrollfluss und damit zusammenhängende (Ausführungs-)Abhängigkeiten, Datenfluss und zeitliche Aspekte. Weiterhin wird diskutiert, aus welchen Gründen sich der Kontrollfluss mehr oder weniger gut darstellen lässt, mitsamt seinen Bestandteilen wie Verzweigungen und Schleifen, Sprüngen oder Ausnahmen. Weiterhin wird geklärt, welchen Stellenwert das Organisationsmodell in der Darstellung hat, oder wo die Unterschiede zwischen Schema-, Instanz- und Multi-Instanz-Darstellungen liegen. Je nach Mächtigkeit des Prozess-Metamodells ließen sich noch weit mehr Spezialfälle betrachten, die genannten Aspekte sollen aber zur Einschätzung und Bewertung der Mächtigkeit der einzelnen Darstellungsformen genügen. Es schließt sich eine Untersuchung der spezifischen Vorteile und Nachteile der jeweiligen Darstellungsform an. Wenn sich von einzelnen Darstellungsformen konzeptbedingt noch verschiedene mögliche Varianten ergeben, werden diese ebenfalls vorgestellt. Auch Multi-Instanz-Ansichten werden berücksichtigt. Des Weiteren kann die Anpassung an verschiedene Benutzerprofile (z. B. Anfänger oder Experte) zu weiteren Varianten führen. Die Darstellungen unterscheiden sich dann nur in der Komplexität. Schließlich werden noch Konfigurationsmöglichkeiten aufgelistet, die dem Benutzer speziell für die jeweilige Darstellungsform angeboten werden können. Auch auf besondere notwendige Voraussetzungen für die Darstellung oder besondere Implementierungsaspekte wird in entsprechenden Unterkapiteln eingegangen.

2.3. Zusammenfassung

In diesem Kapitel wurde eine Übersicht über die Struktur und den Inhalt von Prozessinformationen gegeben. Um flexible Informationsmöglichkeiten für den Benutzer zu schaffen, braucht es Schema-, Instanz- und Multi-Instanz-Darstellungen. Um mächtige Darstellungen zu realisieren, ist die Integration von organisatorischem Wissen über die Struktur von Ressourcen wie Personal, Daten & Dokumenten und Systemen in die Prozessinformationen wichtig. Dieses Strukturwissen wird hier unter dem Begriff Organisationsmodell zusammengefasst.

Visualisierungskonzepte bilden eine Einheit aus Farben, Formen, Notationen, Strukturierung und Nutzerinteraktionen. Ausgehend vom Konzept Prozessgraphdarstellung sollen über Veränderungen an der grafischen Repräsentation und Änderungen an der Struktur der Prozessdaten, eine verbesserte Prozessgraphdarstellung und neue Konzepte entwickelt werden.

Im folgenden Kapitel werden die Anforderungen an eine Prozessvisualisierungskomponente, vor allem wie sie sich aus der Sicht ihrer Benutzer stellen, untersucht.

3

Anforderungsanalyse

Am Anfang dieses Kapitels steht die Frage, welchen Anforderungen ein Informationssystem genügen muss, das Prozessinformationen darstellen soll und wie die Rahmenbedingungen dafür aussehen. Der Design-Theoretiker Gui Bonsiepe [Bons96, S.15] beschreibt ein ontologisches Designdiagramm (Abbildung 3.1) dessen Mittelpunkt das Interface darstellt. Es verbindet die drei Bereiche

- Benutzer
- Zu bewältigende Aufgabe
- Werkzeug, für die Bearbeitung dieser Aufgabe

Im Abschnitt 3.2 werden die Benutzer und ihre Aufgaben diskutiert. Das Werkzeug sind die später im Kapitel 6 beschriebenen Visualisierungskonzepte.

“Durch das [...] Interface wird der Handlungsraum des Nutzers von Produkten gegliedert. Das Interface erschließt den Werkzeugcharakter von Objekten und den Informationsgehalt von Daten. Interface macht Gegenstände zu Produkten. Interface macht aus Daten verständliche Informationen.” (Gui Bonsiepe)

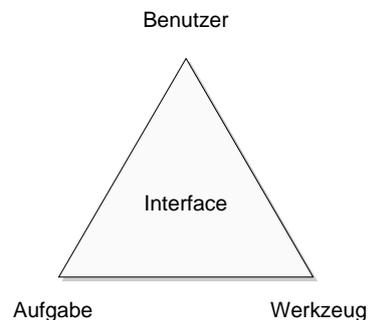


Abbildung 3.1.: Ontologisches Designdiagramm

Vor dem Kontext der Entwicklung einer Prozessvisualisierungskomponente ist auch das Entwicklungsparadigma interessant. Im Allgemeinen geht der Entwicklung einer Software eine Anforderungsanalyse voraus. Ein häufiger Fehler ist ein funktionszentrierter Ansatz (engl. *Form follows function*), bei dem eine Liste der gewünschten Funktionen als Grundlage für die Entwicklung eines Systems dient. Programmierer denken bei Produkten häufig in Kategorien wie Funktionen und Features. Das ist nachvollziehbar, denn das entspricht im Allgemeinen der Art und Weise, wie Entwickler Software produzieren [CR03, S.19].

3.1. Goal-Directed-Design

Anliegen dieser Arbeit ist es, die Grundlagen für eine Prozessvisualisierungs-Engine zu schaffen, die es den Nutzern erlaubt ihre Ziele zu erreichen und die nicht einfach nur möglichst viele Funktionen anbietet. Alan Cooper [CR03, S.18] war ein Pionier von auf das (Benutzer-)Ziel ausgerichteten Softwareentwicklungsprozessen. Die Abbildungen zeigen die Struktur seines Goal-Directed-Design Ansatzes. 3.2 & 3.3.

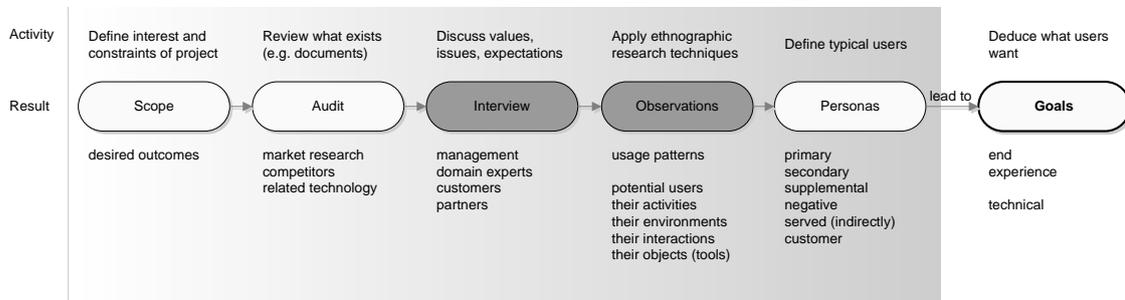


Abbildung 3.2.: Schema des Goal-Directed-Design Entwurfskonzeptes – Teil 1

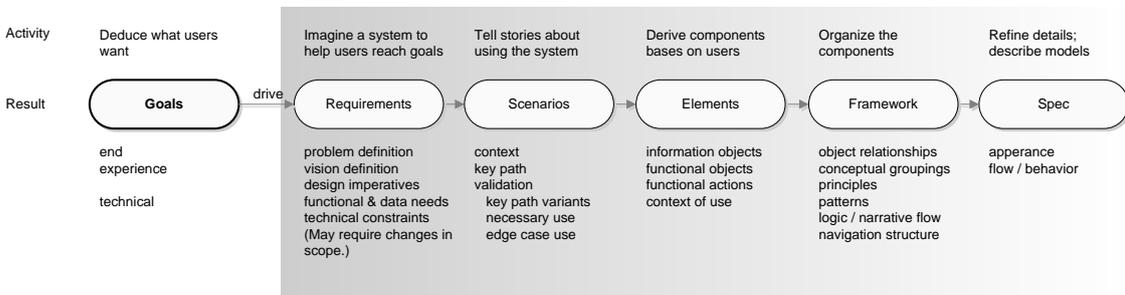


Abbildung 3.3.: Schema des Goal-Directed-Design Entwurfskonzeptes – Teil 2

Die Schritte dieses Entwicklungsprozesses dienen als Rahmen dieser Arbeit. Sie legen die Grundlage für eine am Bedarf der potenziellen Anwender ausgerichteten Komponente zur Prozessvisualisierung. Goal-Directed-Design ist zielorientiert und nicht funktions- oder aufgabenorientiert. Zuerst wird der Benutzer mit seinen Zielen gesehen. Um sein Ziel zu erreichen, wird er nacheinander verschiedene Einzelaufgaben abarbeiten, indem er die vom Programm bereitgestellten Funktionen nutzt. Wenn die Software mit dem Ziel entwickelt wurde, dass sie in erster Linie einen bestimmten Satz von Funktionen bieten soll, wird

der Anwender gezwungen, sich nach dem Programm zu richten. Effizientes Arbeiten wird dann möglich, wenn die Funktionen der Software auf die möglichen Ziele der Benutzer hin ausgerichtet sind.

Ein gutes Beispiel hierfür liefert die Photoshop Variationsübersicht (siehe Abbildung 3.4). Wie in einem Farbkreis sind Variationen des Bildes mit mehr Grün, Gelb, Rot, Magenta, Blau oder Cyan um die Originalversion in der Mitte angeordnet. Wenn eine Variante gewählt wird, kann ausgehend von dieser weiter gewählt werden, bis das gewünschte Ergebnis erreicht ist. Auf diese Weise kann ein Bild auch schrittweise erhellt oder abgedunkelt werden. Dieses Interface orientiert sich an dem mentalen Modell eines Grafikers, der sich für das Bild interessiert, nicht für abstrakte numerische Werte.

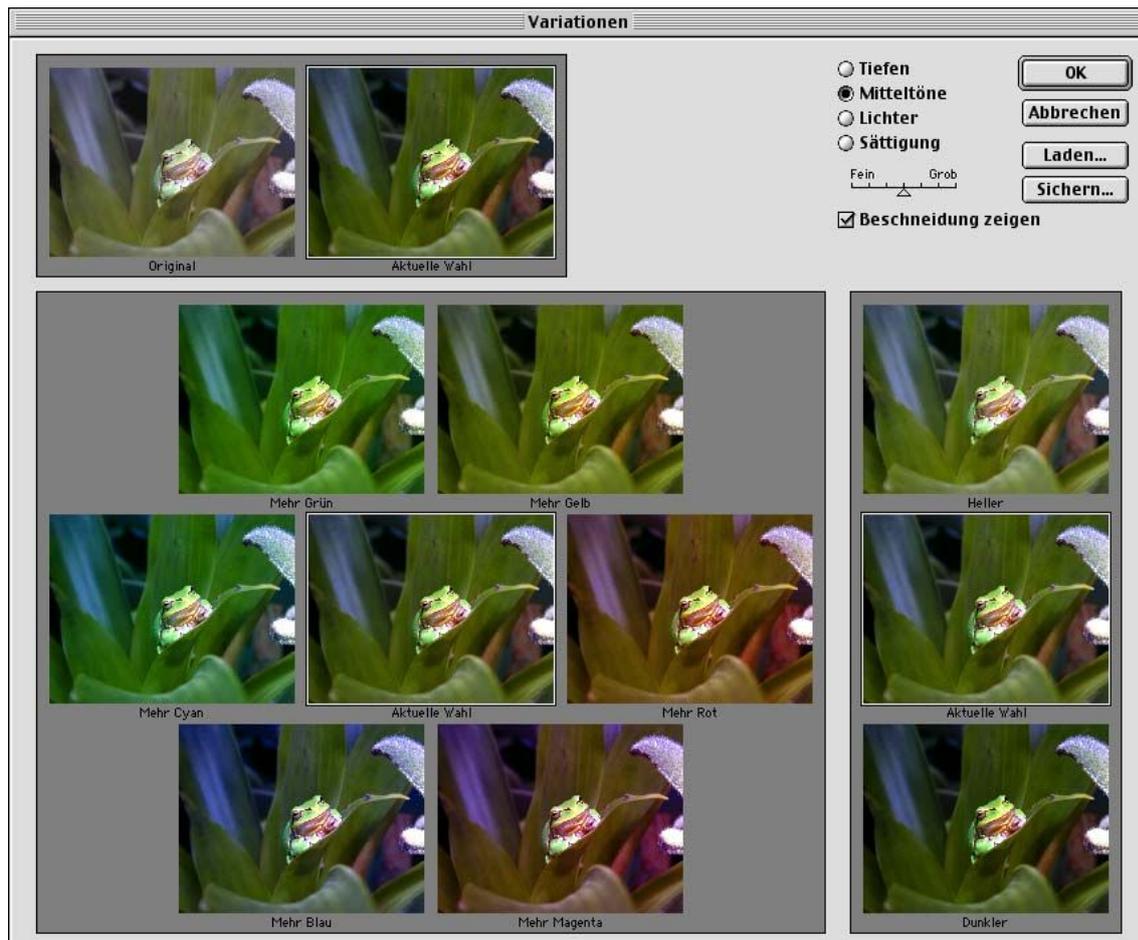


Abbildung 3.4.: Photoshop Variationsübersicht

Am Anfang eines Entwicklungsprozesses steht die Formulierung der Zielsetzung und der Randbedingungen. Die Zielsetzung geschah bereits in Kapitel 1.

Im Kapitel 4, schließt sich die Auseinandersetzung mit bereits existierenden Lösungen und deren Einschränkungen an (engl. *Audit*). Auf ein Gespräch mit potenziellen Nutzern (engl. *Interview*) und die Beobachtung ihrer typischen Arbeitsweisen (engl. *Observation*) muss im Rahmen dieser theoretischen Arbeit jedoch verzichtet werden. Abbildung 3.2 zeigt diese

beiden Schritte daher grau markiert. Bei der Entwicklung einer konkreten Visualisierungslösung wäre dies jedoch unerlässlich. Eine spätere Arbeit könnte Visualisierungskonzepte für ein konkretes Projekt untersuchen und dafür auch eine ausführliche Analyse vornehmen, beispielsweise für das Change Management in der Automobilindustrie oder für ein Krankenhausinformationssystem.

Der Abschnitt 3.2 widmet sich der Definition der typischen Benutzer eines solchen Systems (*Personas*) [CR03, S.55]. Von diesen lassen sich dann die Ziele ableiten (engl. *User Goals*), die mithilfe der Software erfüllt werden sollen. Wo in anderen Projekten oft schon Benutzer und Werkzeuge feststehen, stellt sich hier nicht nur die Frage nach der Gestaltung des Interface, sondern vor allem die Suche nach geeigneten Werkzeugen, also Visualisierungskonzepten, um diese Ziele möglichst ohne Umwege zu erfüllen.

3.2. Ziele nach Benutzergruppen

Die Benutzer der zukünftigen Anwendung sollen dem von Cooper definierten Satz von Nutzermodellen zugeordnet werden. Der erste Schritt besteht darin, den Anwenderkreis zu interviewen und ihre typischen Arbeitsweisen zu beobachten. Ausgehend von diesen Informationen muss dann versucht werden, den Typus des Primärnutzers und dann den Typus der anderen Anwender zu charakterisieren. Die folgende Auflistung dient der Abgrenzung der verschiedenen Typen voneinander. Die Zuordnung der Anwender zu den Nutzermodellen ist dann korrekt, wenn alle folgenden Anforderungen erfüllt sind:

➤ **primary**

Für diesen Primärnutzer wird das Interface entwickelt. Alle anderen Nutzer werden im Großen und Ganzen mit der Ausrichtung des Interface auf den Primärnutzer zufrieden gestellt. Eine Ausrichtung des Interface auf andere Nutzer würde den Primärnutzer nicht zufrieden stellen. Aber wenn die Aufgabenunterstützung der Primärnutzer das Ziel ist, dann sind die anderen Nutzer zumindestens nicht unzufrieden. Wichtig zum Verständnis der Nutzermodelle ist: Wenn es mehr als einen Primärnutzer gibt, dann muss ein Produkt auch mehrere Interfaces anbieten, da sich die unterschiedlichen Zielsetzungen der Primärnutzer nicht unter einem Interface erfüllen lassen!

➤ **secondary**

Sekundärnutzer werden mit dem Interface des/eines Primärnutzers zufrieden gestellt, sie fordern nur einige wenige Zusatzanforderungen.

➤ **supplemental**

Zu diesem Nutzerkreis gehören Personen, die weder Primär- noch Sekundärnutzer sind. Ihre Bedürfnisse werden komplett von einem der Interfaces erfüllt.

➤ **customer**

Hier werden die Bedürfnisse von Auftraggebern erfasst, nicht die von Endnutzern. (Werden wie Sekundärnutzer behandelt)

➤ **served**

Diese Personengruppe zählt nicht zu den Nutzern des Produkts, sie sind nur indirekt Beteiligte. Hierdurch kann beispielsweise überprüft werden, ob der

Einsatz einer neuen Software auch positive Auswirkungen auf Kunden hat.
(Werden wie Sekundärnutzer behandelt)

➤ **negative**

Für diese Personengruppe soll das Produkt nicht entworfen werden. Dies dient zur Abgrenzung und Konkretisierung des eigentlichen Ziels. So könnte z. B. festgelegt werden, dass Power-User nicht die angepeilte Zielgruppe sind.

Diese sechs Nutzermodelle repräsentieren jeweils unterschiedliche Gruppen von Handlungsweisen, Zielen und Motivationen. Diese Einteilung garantiert Vollständigkeit, da sich jede Person mindestens einer dieser Gruppen zuordnen lässt.

Ziel ist es einen Satz markanter Personas mit ihren Zielen, Motivationen und Bedürfnissen zu erstellen. Bei der Anwendungsentwicklung dienen dann diese Personas als Messlatte. Sie helfen den Entwicklern, den von den Benutzerzielen abgeleiteten gewünschten Features, Prioritäten zuzuordnen.

Personas repräsentieren Verhaltensweisen. Sie sind keine Job-Beschreibungen, weil diese nicht zwangsläufig übereinstimmen müssen. An dieser Stelle soll jedoch eine Liste von wichtigen Zielpersonengruppen für eine Visualisierungskomponente genügen, wie sie Tabelle 3.1 zeigt.

Benutzergruppe	Bedürfnisse, Verhaltensweisen
Bearbeiter / Anwender	Wünscht einen guten Überblick über kommende und laufende Aufgaben. Nutzt die gegebenen Werkzeuge ohne große Anpassungen
Prozessverantwortlicher / Abteilungsleiter	Wünscht umfassende Visualisierungsmöglichkeiten, um einen Prozess aus vielen Perspektiven untersuchen zu können. Wünscht Anpassbarkeit der Werkzeuge an spezielle Aufgabestellungen
Management / Geschäftsführung	Wünscht eine kompakte Übersicht über viele Prozessinstanzen. Details sind nicht von Interesse
IT	Wünscht Informationen über den Datenfluss und beteiligte Systeme und Anwendungen. Interessiert sich viel für Zusammenhänge, weniger für konkrete Prozessabläufe
Externer Mitarbeiter / Kunde	Wünscht Einblick in die Einbindung des Kunden in den Prozess, also teilweise Details, teilweise nur Übersicht

Tabelle 3.1.: Benutzergruppen mit Beschreibung

Das heißt, wir beschränken uns im Folgenden auf primary, secondary und supplemental Personas. Als negative Persona wäre der Geschäftsprozessentwickler (oder Prozessmodellierer) zu nennen, der neue Prozessschemata entwirft. Der Fokus dieser Arbeit ist eine Visualisierungskomponente. Wenn die Software auch an Prozess-Entwickler gerichtet wäre, würden diese eine weitere primary Persona ergeben, die dann auch ein extra Interface erfordert.

Einzelne Aufgaben oder Benutzerziele, aber auch ganze Benutzergruppen, lassen sich auch nach dem in Abbildung 3.5 gezeigten Schema voneinander abgrenzen. Dies geschieht auf zwei unabhängigen Ebenen. Aufgaben haben einen bestimmten Aufgabentyp (Entwerfen, Entscheiden, Informieren) und Detailgrad (Übersicht, Zusammenhang, Details). Ein

Halbkreis symbolisiert die Gesamtheit aller Aufgaben. Die eine Ebene der Aufgaben, beispielsweise des Managements, umfassen davon nur einen Teilbereich: Sie lassen sich im Allgemeinen den Aufgabentypen 'Informieren' und 'Entscheiden' zuordnen, innerhalb dieser Aufgabentypen aber nur dem Detailgrad 'Übersicht'. Jede Benutzergruppe lässt sich auf diese Weise einem definiertem Teilbereich der Gesamtheit aller Aufgaben zuordnen.

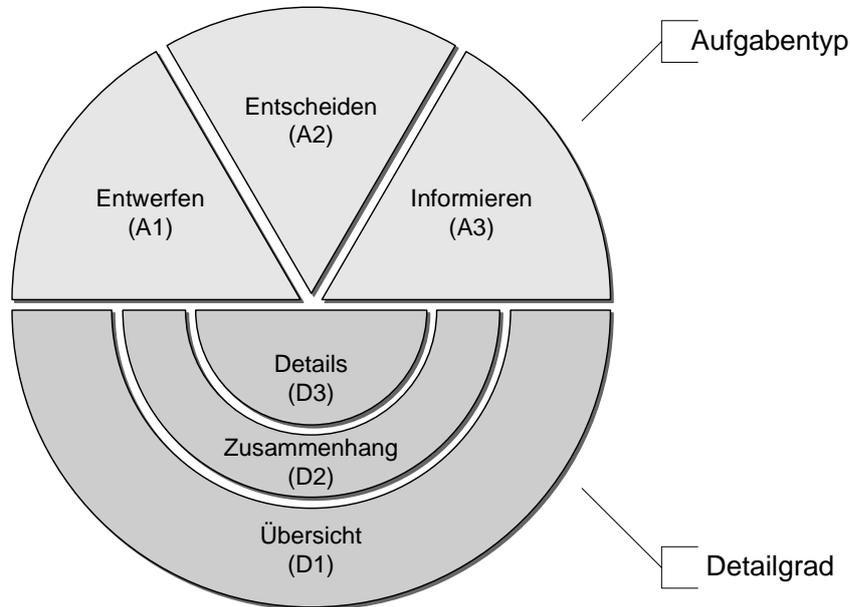


Abbildung 3.5.: Schema für die Einordnung der Benutzergruppen

Manager wünschen ihren Aufgaben entsprechend daher eine stark vereinfachte (aggregierte) Übersicht auf Prozesse, während Prozessbeteiligten eher ein Ausschnitt des Prozesses, mit den für sie relevanten Aktivitäten, in hohem Detaillierungsgrad angezeigt wird [BRB05].

Die Tabellen 3.2 und 3.3 auf den beiden folgenden Seiten listen beispielhafte Aufgaben und Fragestellungen der verschiedenen Benutzergruppen auf:

3.2. Ziele nach Benutzergruppen

Aufgabe (Fragestellung)	Aufgabentyp			Detailgrad		
	Entwerfen ^a	Entscheiden	Informieren	Übersicht	Zusammenhang	Details
Bearbeiter/Anwender						
Liste aktueller Aufgaben		✓				✓
Welche Aufgabe hat Priorität?		✓				✓
Wie ordnet sich eine Aufgabe ins Umfeld ein?			✓			✓
Welche anderen Anwender sind noch an dem Prozess beteiligt?			✓			✓
Welche Aufgaben stehen in naher Zukunft an?			✓		✓	
Was ist schon alles erledigt? (Motivation)			✓	✓		
Prozessverantwortlicher/Abteilungsleiter						
View-Bildung	✓	✓		✓		
Anpassung des Prozessschemas an veränderte Bedingungen	✓			✓		
Zeit-Kontrolle (Plan/Ist Vergleich)		✓		✓	✓	
Wie groß sind die Pufferzeiten?		✓		✓	✓	
Was hat sich in der letzten Zeit getan?			✓	✓		
Welche Meilensteine wurden erreicht?			✓	✓		
Überblick gewinnen			✓	✓		
Wie ist der aktuelle Stand?			✓		✓	
Wie sieht der Daten-/Dokumentfluss aus?			✓		✓	
Wer kommuniziert mit wem?			✓		✓	
Können alle Fristen eingehalten werden?			✓		✓	
Wie sind die Mitarbeiter ausgelastet?			✓		✓	
Wie ist die Kostenentwicklung?			✓		✓	✓
Management						
Wie ist der Status aller Instanzen?		✓		✓		
Wie groß ist die Gesamtbearbeitungsdauer der Instanzen? (Durchsatz)			✓	✓	✓	
Wie sind die Mitarbeiter ausgelastet?			✓	✓	✓	
Wie ist die Kostenentwicklung?			✓	✓	✓	
Können alle Fristen eingehalten werden?			✓	✓		
Zeit-Kontrolle (Plan/Ist Vergleich)			✓	✓		
Übersicht über Instanzen: Welche Bearbeiter sind den Aktivitäten zugeordnet?			✓		✓	

^aSiehe Abbildung 3.5

Tabelle 3.2.: Übersicht über die Fragestellungen und Aufgaben der Benutzergruppen – Teil 1 (Bearbeiter/Anwender, Prozessverantwortlicher/Abteilungsleiter, Management)

Aufgabe (Fragestellung)	Aufgabentyp			Detailgrad		
	Entwerfen ^a	Entscheiden	Informieren	Übersicht	Zusammenhang	Details
IT						
Welche Daten fallen im Laufe eines Prozesses an?			✓	✓	✓	
Welche Anwendungen sind an welchen Prozessen beteiligt?			✓		✓	
Zu welchen Personen wandern die Daten?			✓		✓	
Zu welchen Systemen wandern die Daten?			✓		✓	
Externer Mitarbeiter/Kunde						
Welche Meilensteine wurden in den verschiedenen Instanzen erreicht?			✓	✓		
Überblick über wichtige Abschnitte des Prozesses (geeignete View)			✓	✓	✓	
Wo findet im Prozess Kommunikation zwischen den Firmen statt?			✓	✓	✓	
Wo im Prozess ist der Mitarbeiter/Kunde beteiligt?			✓	✓	✓	✓
Prozessmodellierer						
Prozessschema entwerfen	✓			✓		
View-Bildung	✓	✓		✓		
Fristen und typische Ausführungsdauer festlegen	✓			✓	✓	
Schemaänderungen durchführen	✓				✓	

^aSiehe Abbildung 3.5

Tabelle 3.3.: Übersicht über die Fragestellungen und Aufgaben der Benutzergruppen – Teil 2 (IT, Externer Mitarbeiter/Kunde, Prozessmodellierer)

3.2.1. Bearbeiter / Anwender

Die Bearbeiter benützen die Prozessvisualisierung hauptsächlich als Informationssystem. Es werden auch anhand der gelieferten Informationen Entscheidungen getroffen, in welcher Reihenfolge Aufgaben bearbeitet werden. Meist sind die Nutzer an Detailinformationen von einzelnen Aktivitäten interessiert oder am Zusammenhang ihrer Aufgaben mit denen anderer Bearbeiter in der Prozesskette.

3.2.2. Prozessverantwortlicher / Abteilungsleiter

Die Aufgaben dieser Nutzergruppe lassen sich grob in zwei Bereiche aufteilen, die normalen administrativen Aufgaben eines Abteilungsleiters und Arbeiten am System, wie die

Bildung von Views für andere Bearbeiter oder Änderungen am Prozessschema. Administrative Aufgaben sind Überprüfungen ob Fristen oder Kostengrenzen eingehalten werden können, welche Fortschritte zu verzeichnen sind oder wie die Arbeit unter den Mitarbeitern aufgeteilt ist.

3.2.3. Management

Das Management wird wenig Interesse an ausführlichen Details von Prozess-Spezifikationen haben. Um einen Gesamtüberblick zu ermöglichen, braucht ein Manager Werkzeuge, um Abläufe von den Details abstrahierend zusammenzufassen (Aggregation). Ideal sind Multi-Instanz-Ansichten, die die wichtigsten Informationen zum Fortschritt der Prozessabläufe liefern. Die wichtigsten Prozessaspekte für das Management sind Fortschritt, Zeitplanung, Mitarbeiter und Kosten.

3.2.4. IT

Aufgaben, die sich durch Prozessvisualisierung erledigen lassen, sind sicher nur ein kleiner Teilausschnitt des Aufgabenspektrums von IT-Verantwortlichen. Prozess-Monitoring, Data-Mining und spezialisierte Datenbankabfragen dürften wesentlich wichtigere Informationsquellen darstellen. Daher listet die Tabelle 3.3 auch nur wenige beispielhafte Aufgaben auf, die sich per Prozessvisualisierung lösen lassen.

3.2.5. Externer Mitarbeiter / Kunde

Prozessdetails spielen in diesem Kontext nur selten eine Rolle; das Informieren über den Auftragsstatus und Interaktionen zwischen dem Kunden und der Firma stehen im Vordergrund des Interesses. Geeignete Views werden benötigt, die für die Aufgabe nur die wichtigsten Abschnitte im Prozessablauf detailliert zeigen.

3.2.6. Prozessmodellierer

Auch wenn der Prozessmodellierer hier nicht im eigentlichen Interesse liegt, so profitiert er doch automatisch von guten Visualisierungsmöglichkeiten. Wenn das PMS Schemaänderungen gut unterstützt, können Prozesse im laufenden Betrieb ständig den Erfordernissen angepasst und verbessert werden.

3.3. Zusammenfassung

In diesem Kapitel wurde mit dem Goal-Directed-Design ein Software-Entwicklungskonzept vorgestellt, das keinen funktionszentrierten Ansatz hat, sondern den Fokus auf die Benutzerziele legt. Für dieses Entwicklungskonzept wurden Benutzerkategorien einer Prozessvisualisierungskomponente aufgestellt. Die Ziele dieser Benutzergruppen werden untersucht und dienen als Motivation zur Entwicklung von entsprechenden Visualisierungskonzepten.

Im folgenden Kapitel werden einerseits existierende Visualisierungslösungen untersucht und andererseits werden Arbeiten aus den Bereichen Wahrnehmungsforschung, Usability und Interface-Design ausgewertet. Dies bildet die Grundlage für die Konzeption dieser Arbeit.

4

Verwandte Arbeiten

Das Thema Visualisierungskonzepte für Prozessinformationen berührt vor allem zwei Bereiche: Einerseits die Theorie der Informationsvisualisierung und andererseits bereits existierende Visualisierungslösungen aus dem Marktumfeld von Workflow Management Systemen (*WfMS*). Eine Auseinandersetzung mit diesen Themenbereichen bildet die Grundlage für die Konzeption dieser Arbeit.

Den Anfang macht ein Überblick über die Informationsvisualisierung, indem Arbeiten aus daran beteiligten und angrenzenden Forschungsbereichen im Hinblick auf Prozessdatenvisualisierung ausgewertet werden. Dazu gehören u. a. die Kognitions- und Wahrnehmungsforschung, der Themenkomplex Usability-Engineering und das Interface-Design.

Im zweiten Teil dieses Kapitels werden existierende Visualisierungslösungen ausgewertet. Dabei wird zwischen verschiedenen Software-Kategorien unterschieden. Es werden Softwarepakete aus dem Bereich Business Process Management (*BPM*) daraufhin untersucht, welche Möglichkeiten sie dem Anwender bieten, Prozesse zu visualisieren. Insbesondere wird dabei auf ihre Prozessmodellierungs- und Monitoring-Komponenten eingegangen. Weiterhin werden die Prozessmodelle der BPM Werkzeuge studiert, da sie die Grenzen der Prozessvisualisierung festlegen. Die zweite Produktkategorie bildet Prozessneutrale Visualisierungssoftware. Diese Werkzeuge sind nicht auf den Umgang mit Prozessen spezialisiert oder festgelegt. Es gibt also kein Prozess-Modell, das die Visualisierung begrenzt. Im Hinblick auf ihre Eignung für die Prozessvisualisierung werden Produkte aus dem Bereich Leitstand-Visualisierung und Entwicklungstools für die Eigenentwicklung von Prozess-Visualisierungslösungen untersucht.

4.1. Informationsvisualisierung

Usability und Interface-Design sind Begriffe die sich alle unter dem Schlagwort Mensch-Maschine-Interaktion *HCI* einordnen lassen. HCI ist ein interdisziplinärer Forschungsweig

von Psychologen, Informatikern, Grafikern, Ergonomieexperten und Soziologen. Eine gute Einführung zum Themengebiet HCI bietet Ben Shneiderman in seinem Buch 'Designing the User Interface' [SP05].

Die Gestaltung der Benutzerschnittstelle entscheidet letztlich über die Produktivität der Nutzer und der Akzeptanz des Gesamtsystems. Usability oder Softwareergonomie erhalten bei der Softwareentwicklung mehr und mehr Gewicht.

Methoden des Usability-Engineering werden herangezogen, um die Qualität einer Software zu messen. Fast jedes Produkt, vielleicht mit Ausnahme von Produkten, die nur für die interne Nutzung eines sehr begrenzten Nutzerkreises vorgesehen sind und die nicht nur einen proprietären Werkzeugcharakter haben, muss es sich gefallen lassen, auf Benutzbarkeit hin untersucht zu werden. Auch eine Visualisierungskomponente zur Darstellung von Prozessen sollte im Hinblick auf gute Usability entwickelt werden. Denn die perfekte Visualisierung ist nicht die, die einen Prozess in jeder nur denkbaren Art und Weise darstellen kann, sondern die, mit der die Benutzer ihre Ziele am besten erreichen können.

Die Produktivität eines Unternehmens kann besonders dann gesteigert werden, wenn die Produktivität jedes Einzelnen gesteigert wird.

4.1.1. Kognitions- und Wahrnehmungsforschung

Alles Wissen im Bereich Mensch-Maschine-Interaktion baut letztlich auf der Kognitions- und Wahrnehmungsforschung auf. Wesentliche Erkenntnisse der Kognitionsforschung, die in den 50er und 60er Jahren gemacht wurden, sind nach wie vor aktuell. Die wichtigsten Prinzipien der Wahrnehmungsforschung sind seit Anfang des 19. Jahrhunderts bekannt. Beide Forschungsfelder sind aber nach wie vor aktiv, denn immerhin sind ca. 60% unseres Gehirns mit der Wahrnehmung, die natürlich auch eng mit der Kognition gekoppelt ist, beschäftigt.

Diese Forschungsfelder liefern Hinweise, welchen Prinzipien die Diagrammgestaltung, die für die Prozessvisualisierung eine wichtige Rolle spielt, folgen sollte, um gute Lesbarkeit (engl. *readability*) zu gewährleisten. Gute Kenntnis der menschlichen Wahrnehmung ermöglicht es, darauf abgestimmte grafische Darstellungen zu erzeugen.

“Bilder sagen mehr als tausend Worte”

(alte Volksweisheit)

Die Gestaltpsychologie (Gestalttheorie) begründete die heutige Wahrnehmungsforschung. Ihre Anfänge nahm sie 1912 mit der ersten wichtigen Arbeit von Max Wertheimer [Wert12], der mit seinen Arbeiten die grundlegenden bis heute wichtigen Ideen zur Gestaltwahrnehmung lieferte. Gestaltwahrnehmung bezeichnet die Muster- und Objekterkennung der menschlichen Wahrnehmung.

Aus der Gestaltpsychologie kennen wir heute die grundlegenden Prinzipien der Wahrnehmungsorganisation (engl. *perceptual organising principles*) [Ware00, S.203-212] [DOL02, Dodd02]. Diese Prinzipien beschreiben, wie die Objektwahrnehmung beim Menschen funktioniert – also wie der Mensch Elemente eines Bildes gruppiert:

- **Gesetz der Prägnanz/guten Gestalt/Einfachheit**
(engl. *principle of prägnanz*¹)
Objekte werden am ehesten erkannt, wenn sie regelmäßig, systematisch, einfach oder symmetrisch sind. Dies ist das wichtigste Gestaltgesetz, es ist die Grundlage für alle anderen Prinzipien.
- **Gesetz der Nähe** (engl. *principle of proximity*)
Elemente, die sich in relativer räumlicher Nähe zueinander befinden, werden als zusammengehörig erkannt.
- **Gesetz der Ähnlichkeit** (engl. *principle of similarity*)
Je ähnlicher sich Elemente in Bezug auf Form, Farbe, Helligkeit, Größe oder Orientierung sind, desto eher werden sie auch als zusammengehörig erkannt.
- **Gesetz der Kontinuität/guten Fortsetzung**
(engl. *principle of continuity/continuation*)
Strukturen, die aus kontinuierlichen Konturen (also ohne abrupte Richtungsänderungen) gebildet werden, kann das Auge eher als Objekte wahrnehmen.
- **Gesetz der Geschlossenheit** (engl. *principle of closure*)
Geschlossene Konturen werden als Objekte erkannt, Menschen können aber auch unvollständige Formen als Ganzes erkennen, indem sie die fehlenden Teile durch frühere Erfahrungen ergänzen.
- **Gesetz der Symmetrie** (engl. *principle of symmetry*)
Symmetrische Strukturen werden sehr viel leichter als andere erkannt.
- **Gesetz der Relativen Größe** (engl. *principle of relative size*)
Kleinere Elemente eines Musters werden eher als Objekte erkannt als die größeren. Senkrechte und waagrechte Ausrichtung erleichtert die Erkennung von Objekten.
- **Gesetz des gemeinsamen Schicksals**
(engl. *principle of common fate*)
Elemente die sich gleichförmig bewegen (im Gegensatz zu anderen Elementen der Darstellung) werden als eine Einheit wahrgenommen.
- **Gesetz der Verbundenheit** (engl. *principle of connectedness*)
Zur Ergänzung wird dieses Prinzip in die Liste mit aufgenommen. Die klassische Gestaltpsychologie kennt nur die Obengenannten. Es handelt sich wohl um ein von der Gestaltpsychologie übersehenes Gesetz. Palmer und Rock [PR94] zeigen, dass die Verbundenheit ein stärkeres Organisationsprinzip ist als Nähe, Farbe, Größe und Form. D. h. wenn Elemente durch eine Linie verbunden sind, werden sie als zusammengehörig erkannt, auch wenn sie sonst keinerlei Ähnlichkeit aufweisen.

Diese Gestaltgesetze sollten nicht verwechselt werden mit ähnlichen Prinzipien bezüglich Tiefen- und Entfernungswahrnehmung. Dort gibt es beispielsweise auch ein 'principle of relative size'. Es bleibt auch anzumerken, dass die Zusammenstellung dieser Gestaltgesetze nicht als vollständig angesehen werden kann, da je nach Quelle andere Zusammenstellungen

¹Der von den Deutschen Wertheimer and Koffka geprägte Begriff, wurde in der englischen Literatur so übernommen.

der Prinzipien zu finden sind. Vollständigkeit ist aber auch nicht notwendig, da es nur darum geht, sich die wesentlichen Prinzipien zunutze zu machen, die sich eignen, um gute Darstellungen für Informationen zu finden.

Auch Studien, in denen Augenbewegungen verfolgt wurden, haben zu einem besseren Verständnis der grundlegenden Prinzipien von menschlicher Wahrnehmung und Erkennen geführt. Auch aus diesen Studien folgt, dass Designer verwandte Informationen durch örtliche Nähe, grafische Begrenzungen oder Ähnlichkeit von Helligkeit, Farbe oder Orientierung gruppieren sollten. Generell sollte darauf geachtet werden, dass die grafische Aufbereitung aller Informationselemente auf dem Bildschirm konsistent und vorhersagbar ist [Will00].

Das zweite Forschungsgebiet, die Kognitionsforschung, liefert viele Hinweise für das Interface-Design und damit auch für die Diagrammerstellung. Die engen Grenzen des Kurzzeitgedächtnisses [Mill56, Mill68] sind beispielsweise schon lange bekannt. Beim Interface-Design sollte berücksichtigt werden, dass im Kurzzeitgedächtnis nur etwa sieben Informationseinheiten (engl. *chunks*) abgelegt werden können. Die Wahrnehmung ist sehr eng an das Kurzzeitgedächtnis gekoppelt, d. h. es sollten beispielsweise nicht zu viele Farben gleichzeitig verwendet werden. Denn wenn das Kurzzeitgedächtnis, das sich merkt wofür die Farben stehen, überlastet wird, verringert sich die Arbeitsgeschwindigkeit beträchtlich, da nun das Langzeitgedächtnis mitbenutzt werden muss. Dies gilt natürlich nicht nur für Farben, sondern für alle Arten von dargestellter Information.

Die konsistente Nutzung von Farben und Formen sorgt für Erfahrungswerte, die sich das Gehirn merkt. Erst dadurch wird ein schnelles Erfassen von Informationen ermöglicht. Unser Wahrnehmungsapparat ermöglicht uns durch eine Fähigkeit, deren Auswirkungen das Prinzip/Gesetz der Verbundenheit beschreibt, selbst durch kurzes oberflächliches Anschauen auf den wahrscheinlichen Inhalt zu schließen [Mill68]. Das Sichern von Konsistenz jedweder Art ist im Allgemeinen das wichtigste Konzept für gute Benutzeroberflächen.

Wenn man versteht, wie die menschliche Objektwahrnehmung funktioniert, kann man auch gut wahrnehmbare Informationsobjekte gestalten. Daher lassen sich aus diesen Informationen Gestaltungshinweise für Netzplan-Diagramme (engl. *node-link diagram*) ableiten [Ware00, S.225]. Geschlossene Konturen repräsentieren darin Objekte oder Entitäten:

1. Die Form einer geschlossenen Kontur kann den Typ einer Entität repräsentieren.
2. Die Farbe einer umschlossenen Region kann den Typ einer Entität repräsentieren oder ein Attribut der Entität.
3. Die Größe einer umschlossenen Region kann den Stellenwert der Entität repräsentieren oder ein einzelnes skalares Attribut der Entität.
4. Linien, die eine geschlossene Kontur in Regionen partitionieren, können Einzelaspekte einer Entität voneinander abgrenzen.
5. Regionen von geschlossenen Konturen können aggregiert werden, dies wird als Verbundstruktur wahrgenommen.
6. Wo sich einzelne geschlossene Konturen innerhalb einer größeren geschlossenen Kontur befinden, kann dies konzeptuelle Zugehörigkeit repräsentieren.
7. Geschlossene Konturen, die geordnet im Raum platziert sind, können eine konzeptuelle Ordnung widerspiegeln.

8. Verbindungslinien zwischen Entitäten repräsentieren irgendeine Art von Beziehung.
9. Verbindungslinien können verschiedene Farben oder andere grafische Aspekte, wie z. B. Linienstärke oder Muster haben, um die Art einer Beziehung zu repräsentieren.
10. Die Dicke einer Verbindungslinie kann den Stellenwert einer Beziehung ausdrücken oder ein skalares Attribut wie z. B. einen prozentualen Beitrag oder die Höhe eines Geldtransfers.
11. Die Form von Konturen kann andeuten, dass Entitäten in einer bestimmten Verwandtschaft stehen, z. B. eine halbkreisförmige Einbuchtung einer Kontur und eine dazu passende Kontur mit einer halbkreisförmigen Ausbuchtung
12. Die Nähe von Entitäten kann Gruppen repräsentieren.

Es gibt jedoch auch Kritik an der Gestalttheorie [Huss84], denn sie hat hauptsächlich eine beschreibende Gestalt, keine geschlossene Theorie. Sie hat nur eine vage Terminologie, denn elementare analytische Kategorien werden nicht exakt definiert, wie z. B. Einfachheit und Ähnlichkeit. Die Bewertung von Experimenten erfolgt mit offenen Wertungskategorien wie z. B. Vertrautheit, Ähnlichkeit oder gute Figur. Die Gestalttheorie rechtfertigt sich hauptsächlich durch die Anwendung von Ex-Post-Analysen.

Studien weisen aber immer wieder darauf hin, dass die Gestalttheorie die Realität gut beschreibt. Nach Ryan und Schwartz [RS56] scheinen beispielsweise Konturen eine der wichtigsten Aspekte unserer Wahrnehmung zu sein. Sie zeigen, dass Objekte nur anhand ihrer Konturen deutlich schneller erkannt werden als echte fotografische Abbildungen.

4.1.2. Usability

Eine Software hat eine gute Usability, wenn sie für den Benutzer gut benutzbar, benutzerfreundlich, leicht zu bedienen, zugänglich, nachvollziehbar, klar verständlich, jederzeit verfügbar und bereit ist. Dieser Versuch, Usability im Allgemeinen zu beschreiben, klingt sehr vage und subjektiv. Nur wenn es um die Entwicklung einer bestimmten Software, für einen bestimmten Nutzerkreis, innerhalb eines bestimmten Kontextes geht, dann kann systematisch vorgegangen werden.

Die ISO-Norm 9241 definiert Usability folgendermaßen [ISO 98]:

“Usability ist das Ausmaß, in dem ein Produkt durch einen bestimmten Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und mit Zufriedenheit zu erreichen.”

Um Benutzbarkeit zu erreichen, werden Messinstrumente (Usability-Metriken) benötigt. Klassische Messinstrumente aus dem Bereich des Usability-Engineering sind z. B. Protokollierungssoftware, Protokollierungsvideos, Vergleichs-/Benchmark-Aufgaben oder Fragebögen [Schu05].

Diese Messinstrumente liefern Daten und mit Hilfe von geeigneten Messgrößen wird Usability dann messbar gemacht. Alle Messgrößen lassen sich in die drei schon in der Definition von Usability genannten Bereiche unterteilen:

➤ **Effektivität**

Wie genau und komplett ist das Ergebnis, das erreicht wurde?

➤ **Effizienz**

Stehen Aufwand und Ergebnis in einem angemessenen Verhältnis?

➤ **Zufriedenheit**

Ist der Nutzer mit dem Produkt zufrieden und wird nicht frustriert?

Diese Aspekte spielen für den Maßstab der Gebrauchstauglichkeit die entscheidende Rolle. Wichtig für die Festlegung von Messkriterien ist primär ihre Messbarkeit [MR04]. Beispiele für Messgrößen sind benötigte Zeit (für die Erledigung einer bestimmten Aufgabe), Fehlerraten, Flexibilität oder Erlernbarkeit.

Die Messinstrumente liefern subjektive und objektive Daten. Subjektive Daten sind die Zufriedenheit eines Benutzers oder dessen Einstellung zur untersuchten Software. Die objektiven Daten unterteilen sich weiter in quantitativ messbare Größen wie Fehlerhäufigkeit und in qualitative Daten z. B. Problembeschreibungen.

Die Analyse der Usability einer Mensch-Maschine-Interaktion beginnt mit der Eingrenzung und Untersuchung des Nutzerkreises und der Bestimmung des Kontextes, für den die Software geschaffen wird. Die Definition der Benutzer geschieht über relevante Eigenschaften, die die Benutzer untereinander abgrenzen. In erster Linie sind dies die unterschiedlichen Aufgaben und Ziele. Aber auch Alter, Benutzungshäufigkeit oder Erfahrung (Anfänger und Experten) sind relevant. Im Kapitel 3 geschah dies durch Einteilung in Benutzergruppen, die sich voneinander durch die Ziele ihrer Interaktionen unterscheiden. An die Definition dieser Interaktionsziele schließt sich die Definition der Aufgaben an, deren Erfüllung zum Erreichen der Benutzerziele notwendig sind. Und schließlich gehört zum Kontext noch die Beschreibung der Arbeitsaustattung, der Arbeitsbedingungen und die Definition der Messinstrumente. Diese letzten drei Schritte, können im Rahmen dieser grundlegenden Entwurfsarbeiten für eine Visualisierungskomponente entfallen. In den Entwurf fließen jedoch allgemeine Usability und Interface-Design Erkenntnisse ein.

4.1.3. Interface-Design

Im Umfeld von Usability und *GUI* Design hat sich Ben Shneiderman einen Namen gemacht. Seine acht goldenen Regeln des 'Dialogue Design' werden sehr häufig zitiert [SP05]. Gleich die erste Regel lautet: Strebe nach Konsistenz. So breit das Spektrum an Schlagwörtern im Kontext des Interface-Design auch ist, Konsistenz ist der Schlüsselbegriff, um gut funktionierende Mensch-Maschine-Schnittstellen zu entwickeln.

Konsistenz von Benutzerschnittstellen ist schwer zu definieren. Innerhalb einer Anwendung können zwei Arten unterschieden werden:

➤ **Interne Konsistenz** (auch engl. *orthogonality*)

Interne Konsistenz bezeichnet die Anforderung, dass Unterschiede und Ähnlichkeiten der verarbeiteten Objekte ihre Entsprechung in ihrer grafischen Repräsentation haben. Ähnliche Konzepte sollen auch ein visuelles Attribut (z. B. Form oder Farbe) gemeinsam haben. Einzelne Objekte können durch verschiedene solcher (orthogonaler) Attribute ausgezeichnet sein. Das ermöglicht dem Anwender die damit assoziierten Konzepte richtig zu rekombinieren.

➤ **Externe Konsistenz**

Konsistenz im Sinne von Übereinstimmungen zwischen realer Welt und der

fiktiven Darstellung. Da die Prozessvisualisierung kaum Berührungspunkte mit der realen Welt hat, ist die externe Konsistenz hier in diesem Zusammenhang nicht von so großer Bedeutung. In einer Darstellung, in der manuelle und automatische Prozessschritte dargestellt werden, erhält man jedoch eine intuitive, weil für den Mensch konsistente Darstellung wenn ein stilisiertes Hand-Symbol, ein Kopf oder ein Mensch manuelle Aktivitäten repräsentieren. Automatisch oder maschinell ausgeführte Aktivitäten können dagegen durch ein 'Bildschirm und Tastatur'-Symbol oder ein Zahnrad kenntlich gemacht werden.

In diese Kategorie fällt auch die Konformität zu üblichen Konventionen der verwendeten Plattform (z. B. Windows). Die Lernkurve ist umso niedriger, je eher der Benutzer bekannte Konzepte wiederfindet. Die externe Konsistenz gibt also auch den Grad der Ähnlichkeit zu Software aus demselben Kontext an.

Viele Prinzipien des Interface-Design lassen sich letztlich auf die Schaffung von Konsistenz zurückführen:

“Die wichtigste Art von Konsistenz, ist die Konsistenz des Interface mit den Erwartungen des Benutzers.”
(Bruce Tognazzini)

Der sicherste Weg, die Benutzererwartungen an ein konkretes Interface herauszufinden, ist Benutzertests (engl. *user testing*) durchzuführen [Togn04]. Es gibt dennoch wichtige allgemein gültige Designprinzipien. Anwender sollen beispielsweise das Interface verstehen können, ihre Aktionen sollen für sie vorhersehbare Folgen haben und sie müssen das Gefühl haben, dass sie das Interface kontrollieren und nicht das Interface sie kontrolliert [SP05].

Die folgende Liste führt wichtige Punkte auf, die für das Interface-Design berücksichtigt werden sollten. Auf die einzelnen Punkte wird anschließend kurz eingegangen:

- Vorhersagbarkeit
- Zurück und Undo
- Angemessene Reaktionszeit (Antwortzeit)
- Aufwand minimieren
- Eindeutigkeit (engl. *unambiguity*)
- Einfachheit (engl. *economy*)
- Anzeige des Systemstatus
- Fehlervorbeugung
- Flexibilität
- Ästhetisches und minimalistisches Design

Vorhersagbarkeit

Alle angebotenen Funktionen sollten für den Benutzer zu vorhersehbaren Ergebnissen führen. Das fängt damit an, dass die Namen von Funktionen angemessen gewählt sind [SP05].

Zurück und Undo

Es sollte immer einen Weg zurück geben, sei es in einer Dialogfolge oder wenn es darum geht, die Auswirkungen der letzten Operation(en) rückgängig zu machen. Auch bei normalen Bedienschritten würde das Nichtvorhandensein einer solchen Funktion für den Anwender einen unnötig hohen Aufwand darstellen. Selbst wenn zu jeder Funktion eine komplementäre Funktion existiert, müssten diese einzeln aufgerufen werden. Ein 'Zurück'-Button reduziert den Aufwand auf einen Mausklick pro Schritt. Auch eine 'Zurück' oder 'Undo' Funktion braucht Vorhersagbarkeit für den Benutzer, das impliziert, dass das Programm auch textuell anzeigen können sollte, welche Aktion nun rückgängig gemacht wird. Alan Cooper weist darauf hin, dass die Undo-Funktion weit unterschätzt wird. Eine mächtige Undo-Funktion könnte es z. B. möglich machen, nur die drittletzte Operation rückgängig zu machen, die letzten beiden aber beizubehalten. Traditionelles Undo und Redo ist dazu nicht in der Lage [CR03, S.161].

Angemessene Reaktionszeit

Erstaunlicherweise hängt die allgemeine Produktivität nicht nur von der Reaktionszeit, also der Geschwindigkeit des Interface ab, sondern auch von der Fehlerrate des Menschen und wie leicht es ist, die Auswirkungen dieser Fehler wieder zu beseitigen [Mill68]. Untersuchungen haben gezeigt, dass Antwortzeiten von mehr als 15 Sekunden sich sehr negativ auf die Produktivität auswirken. Die Fehlerrate steigt an und die Zufriedenheit nimmt ab. Interaktionen die weniger als eine Sekunde dauern, können die Produktivität dagegen steigern, allerdings führen sie bei komplexen Aufgabenstellungen auch zu höheren Fehlerraten [SP05, S.457].

Beim Beginn von rechenintensiven Operationen, die den Benutzer zum Warten zwingen, sollte dies signalisiert werden, wenn möglich sogar mit Einblendung der erwarteten Dauer der Operation. Das Interface sollte jederzeit reaktionsfähig bleiben und auch deutlich signalisieren, dass etwas geschieht. Es sollte hierbei auch eine Möglichkeit für den Benutzer geben die Operation abubrechen, sobald zu erwarten ist, dass sie einige Sekunden dauern könnte. Ideal ist es, wenn Teilergebnisse schon vorab auf dem Schirm erscheinen, beim Layouting eines Graphen dieser z. B. Stück für Stück vor den Augen des Benutzers aufgebaut wird. Dieser Punkt betrifft auch die Vorhersagbarkeit. Der Benutzer erwartet, dass das System reagiert. Die Geduld wird maßgeblich durch Zeiträume strapaziert, in denen sich für den Benutzer nicht sichtbar etwas tut. Auch beim Aufbau einer langen Tabelle erscheint dem Benutzer nur die Zeit bis zum Aufbau der ersten Bildschirmseite als Reaktionszeit des Systems. Die Zeit bis zur endgültigen Fertigstellung der Tabelle ist für den Benutzer dann nur von sekundärer Bedeutung. Sobald das Interface eine erste Reaktion zeigt, kann der Benutzer anfangen die weiteren Schritte zu planen, die Zeit vorher muss der Benutzer erdulden.

Aufwand minimieren

Ständiges Design-Ziel sollte sein, den Aufwand (engl. *excise*) für den Anwender zum Erreichen seines Zieles zu minimieren [CR03, S.135]. Seine Ziele erreicht der Benutzer dadurch, dass er Teilaufgaben erledigt. Hier lassen sich zwei verschiedene Arten von Teilaufgaben unterscheiden. Einige dienen direkt der Erreichung des Ziels, andere sind nur Aufwand. Die Fenster einer Anwendung umzupositionieren ist ein typisches Beispiel für unnötigen Aufwand, wenn dies vermeidbar oder automatisierbar wäre.

Aufwand lässt sich auch reduzieren, indem sich Funktionen zuletzt benutzte Eingabeparameter merken, oder Listen häufig benutzter Eingaben (engl. *most recently used*) vorhalten. Ganz allgemein sollen Anwendungen nicht Funktionen anbieten, sondern Benutzern helfen ihre Ziele zu erreichen (Stichwort: Goal-Directed-Design siehe Abschnitt 3.1).

Eindeutigkeit

Unzweideutigkeit der grafischen Symbole ist notwendig, um eine eindeutige Interpretation der grafischen Repräsentation zu gewährleisten [LLW⁺01].

Einfachheit

Wenig unterschiedliche und einfache Symbole reduzieren die grafische Komplexität und verbessern dadurch die Zugänglichkeit [LLW⁺01].

Anzeige des Systemstatus

Die Anwendung sollte den Anwender jederzeit darüber informiert halten, was das System tut. Dies sollte innerhalb angemessener Zeit durch ein geeignetes Feedback geschehen [NM94, Heuristic evaluation].

Fehlervorbeugung

Besser als gute Fehlermeldungen, ist es fehlerträchtige Aktionen im vornherein zu entschärfen oder sie durch den Benutzer bestätigen zu lassen [NM94, Heuristic evaluation].

Flexibilität

Erfahrene Nutzer sollten die Möglichkeit haben häufig benutzte Funktionen an ihre Bedürfnisse anzupassen [NM94, Heuristic evaluation].

Ästhetisches und minimalistisches Design

Dialoge sollten keine irrelevanten oder selten gebrauchten Informationen zeigen. Jede zusätzliche Information mindert die relative Sichtbarkeit der wichtigen Informationen [NM94, Heuristic evaluation].

4.1.4. Styleguides und Normen

Im Abschnitt 4.1.3 wurde zuvor die Wichtigkeit von Konsistenz für die Gestaltung von Benutzeroberflächen geschildert. Einen maßgeblichen Beitrag dazu leistet die Einhaltung von Normen und die Orientierung an Styleguides.

Styleguides sind Richtlinien, die für eine konsistente Gestaltung, ergonomische Bedienung und eine einheitliche Optik sorgen sollen. Sie bestehen häufig aus Zusammenstellungen von Regeln und Empfehlungen zur Gestaltung von Applikationen und Dialogen. Styleguides basieren auf Normen und Erkenntnissen aus Benutzertests [Heid02].

Gründe für den Einsatz von Styleguides sind:

- Sicherstellen von Qualität und Konsistenz einer Applikation
- Konsistenz zwischen verschiedenen Applikationen
- Reduzierte Entwicklungszeiten

- Zufriedenheit und Bediensicherheit beim Anwender

Einige Beispiele für Styleguides und Normen:

- **ISO 9241 Norm** [ISO 98]

Diese Norm zur Qualitätssicherung von Software und Webangeboten beschreibt in insgesamt 17 Teilen die Anforderungen an ergonomische Benutzerschnittstellen. Besondere Bekanntheit hat dabei der Teil 10 'ISO 9241-10' erlangt, der Software-ergonomische Grundsätze der Dialoggestaltung beschreibt. Das Entwurfskonzept des *Goal-Directed-Design* entspricht einer praktischen Umsetzung dieses Teils der Norm.

- **W3C Web Content Accessibility Guidelines 2.0** [W3C 06b]

Das Ziel dieser Richtlinie ist es die Zugänglichkeit von Webinhalten zu erhöhen. Vor allem für Ältere und Menschen mit Behinderungen wie Blindheit oder Farbenblindheit sollen Inhalte benutzbarer werden. Weiteres Ziel ist es, die Inhalte unabhängig vom verwendeten Anzeigemedium zu machen. Damit sind Webseiten nicht nur von Browsern, sondern auch für Screenreader, Suchmaschinen, PDA, Drucker, etc. abrufbar/darstellbar. Die Grundprinzipien dieser Norm sind: Wahrnehmbarkeit, Bedienbarkeit, Verständlichkeit und Kompatibilität zu früheren und künftigen Standards.

Bundesbehörden sind im Rahmen der 'Barrierefreie Informationstechnik-Verordnung' (BITV) [BMI 02] verpflichtet, Webinhalte besonders zugänglich zu gestalten, was der Umsetzung dieser W3C Norm in Version 1.0 vom Mai 1999 entspricht.

- **Apple Human Interface Guidelines** [Appl87]

Umfassende Richtlinien für die Entwickler von MacOS-Software.

Die International Organization for Standardization (ISO) [ISO 06] hat noch viele weitere Normen entwickelt, die sich mit Usability-Themen auseinander setzen (z. B. ISO/TR 16982, ISO 14915, ISO 23973). Eine Übersicht über einige weitere der für Usability relevanten ISO-Normen liefert [Jaco03].

Die Firma Apple begann sehr früh, faktische Standards für die Interface-Gestaltung zu setzen. Schon 1987 erschienen die Apple Human Interface Guidelines [Appl87]. Der Apple Human Interface Evangelist Bruce Tognazzini [Togn92] hatte damals die Aufgabe für Konsistenz im Interface-Design zu sorgen. Er sollte die Entwickler überzeugen, keine eigenen Interfaces zu strukturieren, sondern sich an die Apple Richtlinien zu halten. Damals wie heute ist diese Konsistenz die wichtigste Richtschnur, um zu möglichst intuitiv bedienbaren Benutzeroberflächen zu kommen. Diese Richtlinien werden seither aktuell gehalten.

Die Apple Guideline enthält beispielsweise auch Hinweise, wie viele Funktionen eine Software anbieten sollte. Wenn bei der Gestaltung des Interface Probleme auftreten, empfiehlt sie nicht alle möglichen Funktionen anzubieten, sondern nur diejenigen mit denen 80% der typischen Nutzer ihre Ziele erreichen können. Diese Vorgehensweise führe oftmals zu deutlich einfacheren und eleganteren Oberflächen. Wo versucht wird, auch die Wünsche von den restlichen 20%, den Power-Usern, zu erfüllen, kann das Ergebnis möglicherweise für die anderen 80% unbrauchbar werden.

4.1.5. Zusammenfassung

Gute Visualisierungslösungen beachten die Stärken und Schwächen des menschlichen Wahrnehmungssystems, um möglichst viele Informationen in einer Darstellung unterzubringen. Sie minimieren jedoch auch den Aufwand für das Erfassen und für das Gedächtnis [Will94]. Usability-Betrachtungen liefern die Grundlage für gelungenes Interface-Design. Viele Design-Richtlinien lassen sich daher direkt aus Studien zur menschlichen Wahrnehmung und Kognition ableiten. Ein Schlüsselbegriff für das Interface-Design ist Konsistenz.

Die wichtigsten Prinzipien bei der Visualisierung sind:

➤ **Kontrast**

Unterschiedliche Prozessaspekte auch unterschiedlich darstellen

➤ **Wiederholung**

Die durchgängige Verwendung der gleichen Konzepte führt zu Konsistenz

➤ **Ausrichtung der Elemente**

Die richtige Anordnung entscheidet darüber, ob die Informationen in einem Fluss gelesen werden können

Weitere wichtige Prinzipien sind:

- Klare Darstellung von Beziehungen zwischen Prozessinformationen
- Klare Gliederung der dargestellten Informationen
- Klare Navigationsmöglichkeiten
- Gute Lesbarkeit
- Gebräuchliche Ausdrucksweisen

Der Kontrast lässt sich auch durch das Prinzip der Eindeutigkeit (siehe 4.1.3) beschreiben. Alle Symbole der verwendeten grafischen Notation sollten unterschiedliche Formen (Umrisse) haben. Prinzipiell wäre auch Farbe als Unterscheidungsmerkmal denkbar. Formen sind aber bessere Bedeutungsträger. Farbe ist dennoch wichtig für die Visualisierung, aber sie eignet sich nicht gut, um unterschiedliche Symbol-Notationen zu kennzeichnen. Sie kann hier aber unterstützend wirken. Beispielsweise könnten die verschiedenen Symbolformen als redundante Visualisierung zusätzlich noch mit unterschiedlichen Farben hinterlegt werden, was den Wiedererkennungswert erhöht. Dies sollte jedoch mit Vorsicht geschehen, da Farben leicht zu viel Aufmerksamkeit auf sich ziehen.

Die Größe und Ausrichtung von Symbolen kann genutzt werden, um eine Bedeutung zu transportieren. Die Ersatzrepräsentation für mehrere Aktivitäten, die zusammengefasst wurden, könnte beispielsweise etwas größer ausfallen, um damit anzudeuten, dass das Symbol mehr Information in sich trägt als eine einfache Aktivität. Es gilt allerdings auch, dass Graphen im allgemeinen ruhiger und strukturierter wirken, wenn alle Knoten von gleicher Größe sind.

Zusammenhänge zwischen Symbolen, können auf unterschiedliche Arten deutlich gemacht werden [LLW⁺01].

- Symbole können andere Symbole enthalten
- Symbole berühren oder überlappen sich

- Symbole werden miteinander verbunden
- Symbole werden nahe beieinander positioniert

Ein Symbol, das ein anderes enthält, kann hierarchische Beziehungen repräsentieren. Ein in den aktuellen Prozess eingebundener Unterprozess sollte daher umrahmt werden. Beschriftungen von Symbolen werden direkt im Symbol platziert, bei anderen Beschriftungsmethoden steigt die grafische Dichte der Darstellung unnötig an.

Das Überlappen, Verbinden und in der Nähe Platzieren wird nur für nicht hierarchische Beziehungen verwendet. Nähe ist im Vergleich zu Verbundenheit und Überlappen/Berühren kein präzises Konzept. Es könnte beispielsweise dennoch verwendet werden, um Dokumente einer Aktivität zuzuordnen.

Die Anordnung der Aktivitäten sollte der zeitlichen Aktivierungsreihenfolge entsprechen. Ein orthogonales Layout für einen Graphen eignet sich daher nicht gut, da die Aktivitäten in rechteckigen Strukturen angeordnet werden, die zwar sehr kompakt sind, aber die Ausführungsreihenfolge nicht widerspiegeln.

Einen empfehlenswerten Kurzüberblick über alle Aspekte, die die Lesbarkeit von Darstellungen betreffen, liefert van Vliet et al. [KDV02]. Insgesamt sind Usability und Interface-Design gut untersuchte Forschungsbereiche, sie liefern sehr viele direkt verwertbare Hinweise zur Entwicklung von Prozessvisualisierungslösungen.

4.2. Existierende Visualisierungslösungen

Im Folgenden soll untersucht werden, was existierende Visualisierungslösungen bieten, welche Einschränkungen sie haben und inwiefern sie für eine umfassende Prozessvisualisierung geeignet sind.

Grundsätzlich lassen sich zwei Kategorien von Software zur Prozessvisualisierung unterscheiden. Zum einen prozessorientierte Werkzeuge und prozessneutrale Software.

➤ Prozessorientierte Visualisierungssoftware

besitzt ein vordefiniertes Prozess-Metamodell das einen festen Satz an Beschreibungskonstrukten enthält. Sie dient speziell dem Umgang mit Geschäftsprozessen. Nachteile sind die durch das Metamodell begrenzte Ausdrucksmächtigkeit und häufig nur eingeschränkte Visualisierungsmöglichkeiten.

➤ Prozessneutrale Visualisierungssoftware

Hierbei handelt es sich um Software zur Visualisierung von beliebigen Inhalten. Einige Werkzeuge unterstützen explizit das Anzeigen von aus Knoten und Kanten bestehenden Graphen. Software dieser Kategorie besitzt kein zugrunde liegendes Prozess-Metamodell. Damit sind beliebige Prozesse darstellbar. Visualisierung ist teilweise nur eine von mehreren unterstützten Funktionen. Nachteile sind ein hoher Initialaufwand für das Mapping zwischen Prozessdaten und Darstellung und das Fehlen von spezialisierten Editoren. Die sich ergebenden Vorteile sind große Flexibilität bei der Visualisierung von Prozessen und Herstellerunabhängigkeit bei Verwendung offener Standards wie Scalable Vector Graphics (SVG) [W3C 06a].

4.2.1. Business Process Management (BPM) Werkzeuge

Abbildung 4.1 zeigt die Kategorien für prozessorientierte Visualisierungssoftware. Die meisten Werkzeuge fallen unter die Bezeichnung *BPM Werkzeuge*. Am Markt existieren größere Suites wie z. B. WebSphere von IBM, die die ganze Bandbreite von Modellierung, Analyse, Monitoring und Steuerung abdecken und spezialisierte Tools für einzelne dieser Bereiche. Zu einer ganz anderen Kategorie gehört Projektplanungssoftware wie MS Project.

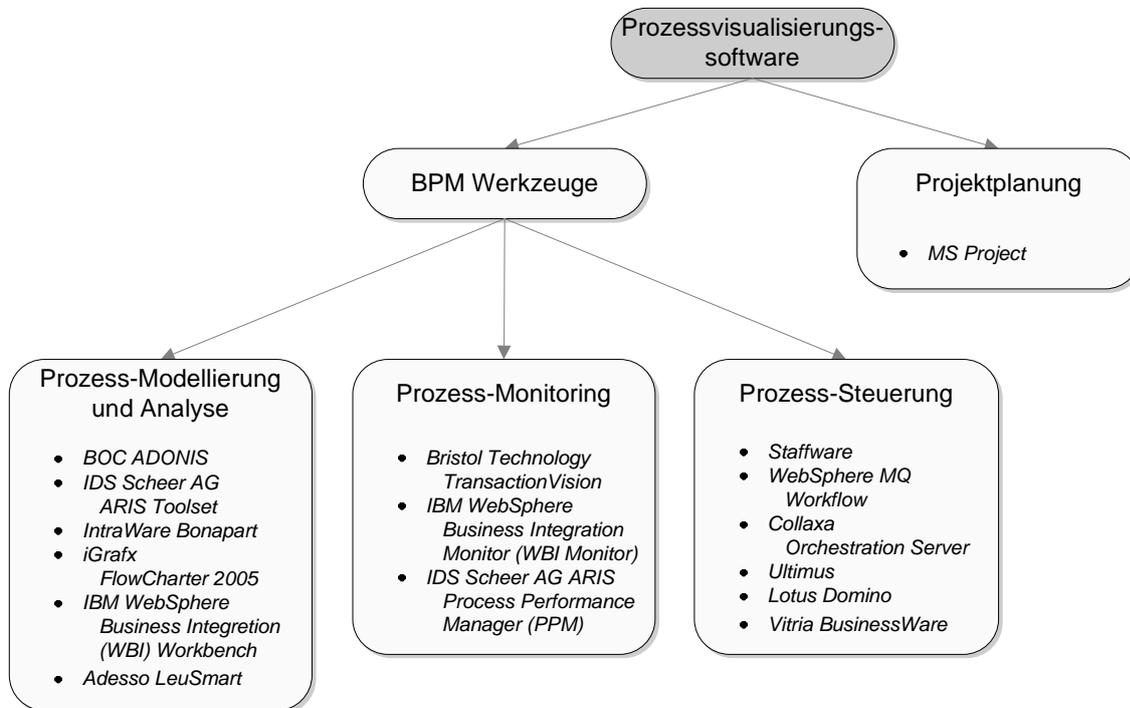


Abbildung 4.1.: Kategorisierung von Prozessorientierter Visualisierungssoftware

In den folgenden Abschnitten werden einige der am Markt befindlichen Visualisierungswerkzeuge kurz vorgestellt, wobei der Fokus auf der jeweiligen (Prozess-)Visualisierungskomponente und deren Möglichkeiten, Stärken und Schwächen liegt. Eine ausführliche Analyse vieler Werkzeuge findet sich in [RR03].

PROZESSMODELLIERUNG & ANALYSE

Werkzeuge zur Prozessmodellierung unterstützen die Designer von Geschäftsprozessen beim Entwurf und der Verifikation auf der Grundlage eines vorgegebenen Prozess-Metamodells. Analysemöglichkeiten bilden die Grundlage für eine Optimierung vorhandener Prozesse.

IntraWare Bonapart

Bonapart ist Teil der IntraWare CRMSuite [Intr01]. Das in Bonapart verwendete Metamodell ist objektorientiert, unterstützt aber auch die klassische Zusammensetzung aus Funktions-, Daten-, Organisations- und Prozess-Sicht. Die Modellierung der Prozesse erfolgt graphbasiert. Der Datenfluss wird implizit durch den Kontrollfluss modelliert (Me-

tapher: 'Weiterreichen von Dokumenten'). Prozesse können mittels statischer Analysen (z. B. Konsistenzchecks) und dynamischer Analysen (in Verbindung mit Simulationen) untersucht werden.

Visualisierung ist mit dem Prozess-Editor möglich. Bonapart kann nur Prozessschemata visualisieren, keine Instanzen. Kanten verbinden die verschiedenen Symbole für Aktivitäten, Daten, Systeme und Rollen. Es gibt keine expliziten Konstrukte zur Modellierung von Verzweigungen und Schleifen. Die Art eines Split- oder Join-Knotens wird implizit durch Ein-/Ausgangsbedingungen festgelegt. Es lassen sich Verzweigungswahrscheinlichkeiten anzeigen. Parallele Verzweigungen sind implizit daran erkennbar, dass alle ausgehenden Kanten die Übergangswahrscheinlichkeit 100% besitzen. Daten, Systeme und Rollen lassen sich für die Ausgabe wahlweise ausblenden. Die Darstellung von Symbolen lässt sich flexibel konfigurieren. Benutzerdefinierte (Aktivitäten-)Attribute sind möglich und können in bestimmtem Umfang auch visualisiert werden. Das Layout geschieht manuell durch den Benutzer.

IDS Scheer AG ARIS Toolset

Das ARIS Toolset [SH99, Sche01, Sche02] und seine Zusatzkomponenten unterstützen das klassische Sichtenmodell (vgl. Bonapart) und bietet zusätzlich vielfältige Erweiterungen. Das Prozess-Metamodell basiert auf Ereignis-Prozessketten (EPK). Aktivitäten die als Funktionen dargestellt werden, müssen daher immer alternierend mit Ereignissen angeordnet sein. Kontroll- und Datenfluss können getrennt voneinander modelliert werden. Verschiedene Verzweigungstypen können explizit modelliert werden. Prozesse können animiert werden und statisch oder dynamisch analysiert werden. Die Möglichkeit zur Simulation besteht über eine Zusatzkomponente.

Die Prozessvisualisierung erfolgt in ARIS mit dem Prozess-Editor. Das Aussehen der Symbole kann verändert werden, es steht aber nur eine begrenzte Auswahl zur Verfügung. Es ist möglich die EPK-Ereignisse auszublenden, dadurch wird die Prozessdarstellung übersichtlicher. Die Anzeige von benutzerdefinierten Attributen ist möglich. In der animierten Simulation werden in begrenztem Umfang Ausführungszustände gezeigt. Es kann farblich unterschieden werden, ob eine Aktivität bereits aktiviert/abgeschlossen oder ob sie noch nicht aktiviert worden ist.

Adesso LeuSmart

Bei LeuSmart [ades01] dienen FunSoft-Netze [DGS95] als Grundlage des Prozess-Metamodells. FunSoft-Netze sind abstrakte Petrinetze. Das Prozessmodell unterstützt damit Prozess-, Daten und Organisationssicht. Der Datenfluss wird implizit durch den Kontrollfluss modelliert. Eine explizite Modellierung von Kontrollflussverzweigungen ist nicht möglich. LeuSmart bietet Prozessanimation, statische und (durch Simulation) dynamische Analysen.

Die Visualisierung von Prozessen erfolgt in LeuSmart mit dem Prozess-Editor. Die konkrete Semantik von Verzweigungen wird aus der Visualisierung nicht ersichtlich. Die fehlende Trennung zwischen Kontroll- und Datenfluss trägt zu unübersichtlichen Prozessdarstellungen bei. Die Konfiguration der Darstellung der Symbole beschränkt sich auf eine kleine Auswahl. Benutzerdefinierte Attribute werden nicht unterstützt. Welche vordefinierten Attribute angezeigt werden sollen, ist nicht konfigurierbar. Einzelne Modellelemente können nicht ausgeblendet werden.

IBM WebSphere Business Integration Workbench (WBI Workbench)

Die WBI Workbench von IBM dient der Modellierung und Analyse von Geschäftsprozessen [Cava03, Peis05]. Das Prozessmodell basiert auf azyklischen Aktivitätennetzen. Schleifen lassen sich aber über Spezialkonstrukte integrieren. Auch bedingte Verzweigungen werden vom Metamodell berücksichtigt. Kontroll- und Datenfluss lassen sich jeweils explizit modellieren. Benutzerdefinierte Aktivitätenattribute sind nicht vorgesehen. Zeitliche Aspekte können nur begrenzt modelliert werden. Es können nur einfache Deadlines in Form relativer Zeitangaben angegeben werden.

Organisationsmodell und Prozessmodell können in Repositories verwaltet werden, somit können größere Teams Prozessmodelle gemeinsam entwickeln. Die WBI Workbench enthält ein UML-Modellierungswerkzeug und ein Werkzeug zur Erstellung von Benutzermasken (Xform-Designer). Diese Masken können dann einzelnen Aktivitäten zugeordnet werden. Für die Optimierung von Geschäftsprozessen bietet WBI Workbench statische und dynamische Analysen basierend auf Simulationen an.

Zu IBM Websphere gehört auch das WfMS MQ Workflow. Die vom Modellierungswerkzeug WBI Workbench erstellten Prozessmodelle lassen sich dort verwenden. In MQ Workflow laufende Instanzen lassen sich mit einer auf den Visualisierungsmöglichkeiten von WBI Workbench aufsetzenden Komponente namens IBM WebSphere Business Integration Monitor visualisieren (siehe Abschnitt auf S. 50).

Bei der Visualisierung sind benutzerdefinierte Symbole möglich. Unterprozesse lassen sich 'aufklappen', diese werden dann unterhalb des Elternprozesses dargestellt. Bei den Aktivitäten wird ein Zähler sichtbar, sobald sie wegen einer Schleife mehrmals ausgeführt wurden. Kontroll- und Datenfluss werden konsequent getrennt durch die Verwendung unterschiedlicher Kantenarten. Zusätzlich kann der Datenfluss auch ausgeblendet werden. Aktivitätenattribute sind per Mausklick zugreifbar.

BOC ADONIS

ADONIS [JK01, KJS96, KK01] bietet mit EPKs und einem proprietären Metamodell zwei unterschiedliche Metamodell-Typen zur Modellierung von Geschäftsprozessen an. Des Weiteren zählt ADONIS zu den Computer-Aided-Software-Engineering (CASE) Tools und bietet damit eine ganze Reihe weiterer Modellierungswerkzeuge z. B. für UML Aktivitätsdiagramme und Organisationsmodelle an.

Für das EPK-Metamodell gelten die Aussagen zu ARIS (siehe Abschnitt auf S. 46). Dort können Kontroll- und Datenfluss getrennt voneinander modelliert werden. Beim proprietären Metamodell von ADONIS lässt sich dagegen kein Datenfluss modellieren. Der Kontrollfluss kann hinreichend modelliert werden. Benutzerdefinierte Attribute gibt es nicht. ADONIS bietet Möglichkeiten zu statischen Analysen und Simulationen an. Die Prozessdarstellung kann nicht konfiguriert werden.

iGrafx FlowCharter 2006

Die iGrafx Produktfamilie [iGra06] enthält FlowCharter 2006 für die Prozessmodellierung und SixSigma für die Analyse und Simulation. Prozesse werden mit *BPMN* modelliert. Benutzerdefinierte Aktivitätsattribute sind möglich. Die Visualisierung erfolgt als BPMN-Swimlane-Darstellung, wobei die Aktivitäten je nach zugeordneter Rolle einer horizontalen

Partition zugeordnet werden. Alle Aktivitäten, die einer Rolle zugeordnet sind, werden somit übersichtlich horizontal angeordnet.

Zusammenfassung

Die in diesen Abschnitten vorgestellten Werkzeuge erlauben Prozessmodellierung und Analyse auf semantisch hoher Ebene. Vom zugrunde liegenden Modellierungsparadigma ist die Ausdrucksmächtigkeit, die Visualisierung und deren Verständlichkeit sehr stark abhängig. Petrinetz und EPK basierte Metamodelle tragen nicht zu einer übersichtlichen Visualisierung bei. Benutzerdefinierte Aktivitätenattribute sind leider nicht überall möglich. Der gravierendste Nachteil bei allen Modellierungswerkzeugen ist die fehlende Möglichkeit zur Visualisierung von Prozessinstanzen. Übliche Darstellungsform für die Prozesse ist die Graphdarstellung, FlowCharter 2006 dagegen setzt auf eine sehr übersichtliche Swimlane-Darstellung.

Die vorgestellten Modellierungswerkzeuge bieten teilweise Exportschnittstellen für WfMS an. Einige WfMS werden im Abschnitt Prozess-Steuerung (siehe Abschnitt auf S. 49) vorgestellt.

Im folgenden Abschnitt werden Prozess-Monitoring Werkzeuge vorgestellt. Sie sind im Gegensatz zu den vorgestellten Prozessmodellierungswerkzeugen in der Lage, laufende Prozesse zu visualisieren. Der IDS Process Performance Manager (PPM) und der IBM WebSphere Business Integration Monitor (WBI Monitor) sind eng mit den Modellierungstools aus dem jeweils gleichen Haus gekoppelt. Sie verwenden für die Visualisierung der Prozessinstanzen dieselbe Darstellungsform wie auf Modellierungsebene. Beide Werkzeuge importieren die Daten zu den Prozessinstanzen aus dem jeweiligen WfMS.

PROZESS-MONITORING

Process Performance Management (PPM) Werkzeuge lassen sich zum Themengebiet Prozess-Monitoring einordnen. Sie erlauben eine Analyse laufender Geschäftsprozesse anhand von Echtzeitdaten. Datenquelle sind häufig Repositories, die Informationen über Prozesse und Geschäftsdaten aus verschiedenen WfMS und aus Anwendungssystemen integrieren. Der Schwerpunkt liegt dabei auf der gemeinsamen Aufbereitung von vielen Geschäftsprozessen mithilfe von zeit-, kosten- und qualitätsbezogener Kennzahlen. Im Rahmen dieser Untersuchung ist vor allem interessant, dass PPM-Systeme oftmals die Visualisierung einzelner Prozessinstanzen ermöglichen.

IDS Scheer AG ARIS Process Performance Manager (PPM)

Der ARIS Process Performance Manager (ARIS PPM) [IDS 03] ermöglicht es Leistungsdaten zu laufenden Geschäftsprozessen automatisiert zu ermitteln. Die gelieferten Informationen ermöglichen systemübergreifende Visualisierung und Auswertung von Prozessinstanzen. Schwachstellen in den Prozessen bei Kosten und Bearbeitungszeiten lassen sich so aufdecken. Verschiedene Datenquellen wie Historiendaten und Belege werden vom ARIS PPM Repository nach ihrer zeitlichen Abfolge sortiert und zu EPK-Prozessinstanzen zusammengesetzt. Diese (virtuellen) Prozessinstanzen lassen sich visualisieren. Dazu reichert ARIS PPM die Prozessketten mit Attributen, wie z. B. Bearbeitern, aus den hinterlegten Prozessmodellen an. Auch einzelne Prozessinstanzen können visualisiert werden, allerdings

werden nur die bisherigen Prozessverläufe wiedergegeben, zukünftige Prozessschritte fehlen in der Darstellung. Diese Einschränkung stellt allerdings eine sehr starke Restriktion dar.

Geschäftsfälle aus dem PPM Repository, können an das ARIS Toolset übergeben werden und dort zum Re-Design von Geschäftsprozessen verwendet werden.

IBM WebSphere Business Integration Monitor (WBI Monitor)

Die WBI Monitor Komponente gehört zu MQ Workflow (siehe Abschnitt auf S. 50).

Sie erlaubt das gleichzeitige Monitoring vieler Geschäftsprozesse mittels statistischer Analysen (z. B. zu Kosten- und Zeiteigenschaften). Vor dem Kontext dieser Arbeit ist aber eher die Möglichkeit zur Visualisierung einzelner Prozessinstanzen interessant. Dabei wird dieselbe grafische Darstellungsform wie bei WBI Workbench gewählt (vgl. Abschnitt auf S. 47), jedoch um farbcodierte Zustandsinformationen angereichert. Die Datengrundlage des WBI Monitors wird einerseits durch den Zugriff auf die Modelldaten der WBI Workbench und andererseits durch den Zugriff auf die Ausführungs-/Instanzdaten (z. B. Audit Trails) aus MQ Workflow gebildet.

Ausgewählte Mengen von Prozessinstanzen können in tabellarischer Form dargestellt werden. Jede Zeile repräsentiert dabei eine Prozessinstanz. Die in den Spalten angezeigten Attribute (z. B. Startzeit, Bearbeitungsdauer, Kosten und Prozessverantwortlicher) sind konfigurierbar, jedoch werden benutzerdefinierte Attribute nicht unterstützt.

Bristol Technology TransactionVision

Vom Hersteller wird TransactionVision den Bereichen BPM und Business Activity Monitoring zugeordnet [Bris05]. Das Werkzeug ermöglicht Monitoring und Analyse laufender Geschäftsprozesse. Es lassen sich diverse Kenngrößen errechnen u. a. Transaktionszeiten. Schwachstellen in den Prozessen wie Überlastsituationen lassen sich gut erkennen. TransactionVision erlaubt das Überwachen von Service-Leveln über komplette Geschäftstransaktionen. Benutzer können sich jederzeit über den Status ihrer Transaktionen informieren. Alle Daten können in einer Datenbank gespeichert werden. Daraus lassen sich Berichte über Geschäftsprozesse generieren, die Auskunft über das Zeitverhalten und über Veränderungen über verschiedene Zeiträume geben. Im Gegensatz zu ARIS PPM liegt der Fokus auf kurz laufenden Geschäftstransaktionen und damit verbundenen Echtzeitanforderungen.

PROZESS-STEUERUNG

In diesem Abschnitt werden die Visualisierungsfähigkeiten verschiedener marktgängiger WfMS untersucht. Außerdem wird kurz auf das jeweilige Metamodell eingegangen. Die Hauptaufgabe von WfMS ist die Prozessausführung, jedoch bieten die Systeme meist auch Modellierungs- und Monitoring-Komponenten, welche zur Prozessvisualisierung zum Einsatz kommen können. WfMS bieten typischerweise keine Möglichkeiten für den Import von produktfremden Prozess- und Laufzeitdaten, daher gestatten sie ausschließlich das Monitoring und die Visualisierung der von ihnen ausgeführten Prozessinstanzen.

Staffware Staffware

Bei Staffware können Prozesse grafisch modelliert werden. Es gibt Editor für den Entwurf von Bildschirmmasken und umfangreiche Schnittstellen für den Export von Laufzeitdaten in externe Visualisierungskomponenten. Allerdings werden die Prozessmodelle nicht exportiert.

Das Prozess-Metamodell ist sehr mächtig. Es ermöglicht nicht nur bedingte und parallele Verzweigungen und Schleifen. Interessant sind Möglichkeiten bei Fristüberschreitungen Alternativaktivitäten zu starten. Datenflussbeziehungen zwischen Aktivitäten werden nur implizit festgelegt. Temporale Aspekte werden nicht ausreichend unterstützt.

Die Visualisierung laufender Prozessinstanzen ist eine Minimallösung. Der Grafik ist lediglich der Kontrollfluss und die aktuell ausgeführte Aktivität zu entnehmen. Bereits ausgeführte und nicht ausgeführte Aktivitäten sind nicht unterscheidbar. Aktivitätenattribute sind nicht darstellbar.

Von der Firma e-FACT GmbH wird für Staffware eine externe Visualisierungskomponente namens ProcessReports angeboten [e-FA00]. Diese basiert auf der Visualisierungskomponente von iGrafx (siehe Abschnitt auf S. 47). Da Staffware keinen externen Zugriff auf das Prozessmodell zulässt, muss mittels eines iGrafx Tools ein zum Staffware-Prozessmodell korrespondierender Prozessgraph erstellt und mit den Staffware-Daten verknüpft werden. Daraus resultiert eine ansprechende Swimlane-Darstellung, die nun verschiedene Zeitattribute und die jeweilige Zielanwendung anzeigt. Ausführungszustände werden durch verschieden eingefärbte Aktivitäten repräsentiert. Der relative Zeitaufwand von abgeschlossenen Aktivitäten wird intelligent über unterschiedlich intensive Farbtöne visualisiert. Die Darstellung ist weitgehend konfigurierbar.

Eine Prozess-Monitoring Software namens Staffware Process Monitor ermöglicht die Analyse vieler Prozessinstanzen gleichzeitig. Einzelinstanzen lassen sich als EPKs darstellen. Es handelt sich nicht um eine Eigenentwicklung, sondern um eine angepasste Version des Process Performance Managers (PPM) von IDS Scheer (siehe Abschnitt auf S. 48).

IBM WebSphere MQ Workflow

IBM WebSphere [IBM 03b, IBM 03a] bietet vielfältige Komponenten für die Modellierung (WBI Workbench siehe Abschnitt auf S. 47), das Monitoring (WBI Monitor siehe Abschnitt auf S. 49) und die Ausführung von Workflows [LR99]. MQ Workflow selbst bietet auch eine Oberfläche für das Monitoring von Prozessinstanzen. Weitaus mächtiger ist jedoch der WBI Monitor. Diese und andere Zusatzkomponenten werden durch sehr viele (offen gelegte) Import/Export Schnittstellen (siehe [IBM 03b]) für Modell- und Instanzdaten angebunden. Das Metamodell wurde bereits im Abschnitt über die WBI Workbench beschrieben.

Vorbildlich ist die durchgängig durch alle Komponenten konsistente Nutzung derselben Modellierungs- und Visualisierungsansätze. Die Visualisierung von Prozessinstanzen ist intuitiv und konfigurierbar. Die Ausdrucksmächtigkeit des Prozess-Metamodells ist eingeschränkt.

Eine relativ neue Entwicklung ist die Unterstützung des BPEL4WS Standards [IBM,03] zur Beschreibung von Geschäftsprozessen mit Webservices. Die IBM WebSphere Business

Integration Server Foundation unterstützt dies seit Version 5.1 [Spal05]. Die erste Implementierung fand sich beim Collaxa BPEL Orchestration Server (siehe Abschnitt auf S. 51).

Vitria BusinessWare

Bei Vitria BusinessWare handelt es sich um ein typisches Enterprise Application Integration (EAI) Werkzeug, das allerdings Workflow-Funktionen anbietet. Es sorgt für die Integration von Anwendungen und Prozessen (z.B. SAP R3). Für Geschäftsprozesse ist die Ankopplung an Anwendungssysteme wichtig, BusinessWare bietet hierfür viele Konnektoren zur Anbindung an Standardsoftware (SAP R/3) und Datenbanken. Zudem gibt es umfangreiche Exportschnittstellen für den Process Performance Manager von IDS (siehe Abschnitt auf S. 48).

Geschäftsprozesse werden durch UML Statecharts abgebildet, deren Zustandsübergänge mittels Java codiert werden. Kontrollflüsse werden also mittels Statecharts modelliert. Diese kann BusinessWare auch aus dem CASE-Tools Rational Rose importieren. Die Komponente Automator stellt den Prozess-Editor dar. Zustandsübergänge werden auf der Grundlage von Event-Condition-Action (ECA) Regeln in Java codiert. Externe Ereignisse (von Anwendungen) können gut integriert werden. Hierarchische Prozessmodelle werden unterstützt. Der Datenfluss ist lediglich implizit definiert. Als Ausführungseinheiten von Aktivitäten können nur die beiden Entitätstypen Benutzer und Gruppe genutzt werden. Semantisch höherwertige Beschreibungskonzepte sind nicht vorgesehen.

Insgesamt bietet BusinessWare ein ausdrucksstarkes Metamodell. Jedoch sind Prozesse nicht vollständig grafisch modellierbar und der eingebettete Java-Code führt zu nicht analysierbaren Modellen.

Zum Prozess-Monitoring dient die Komponente Cockpit. Alternativ können Prozessdaten in ARIS PPM übernommen werden. In Cockpit lassen sich einzelne oder mehrere Prozessinstanzen auch mit komplexen Anfragen auswerten. Prozessgraphdarstellungen lassen sich nicht darstellen (das verwendete komplexe Metamodell eignet sich auch nicht gut dafür). Benutzer können sich Arbeitslisten anzeigen lassen.

Alternativ können Instanzdaten (Audit Trails) nach ARIS PPM exportiert werden (siehe Abschnitt auf S. 48).

Collaxa BPEL Orchestration Server

Collaxa bietet eine mächtige Engine für die Definition und Steuerung von Webservice-Flows. Technische Grundlage bildet die Geschäftsprozess-Beschreibungssprache BPEL4WS (auch WS-BPEL) [IBM,03]. Collaxa bietet hierzu noch eigene Erweiterungen wie manuelle Schritte und Kontrollflusskonstrukte.

Im Rahmen dieser Arbeit sind vor allem die angebotenen Möglichkeiten zu Visualisierung und Monitoring laufender Prozessinstanzen interessant. Die BPEL Console bietet u. a. auch ansprechende visuelle Darstellungen einzelner Prozessinstanzen [Coll03]. Der Status von Aktivitäten wird sowohl farblich als auch symbolisch hervorgehoben. Temporale Aspekte werden kaum unterstützt, da dies in BPEL4WS bisher nicht vorgesehen ist. Insgesamt sind die gebotenen Visualisierungsmöglichkeiten eher an Experten gerichtet.

Ultimus Ultimus Workflow Suite 4

Ultimus ist ein Workflow-System das speziell für die Definition und Steuerung von Formular- und Dokumentflüssen konzipiert wurde [Ulti02]. Prozessmodelle können mithilfe des Workflow Designer modelliert und angezeigt werden. Den Aktivitäten können entweder Windows-Anwendungen oder mit dem Ultimus Form Designer erstellte Formularvorlagen zugeordnet werden. Mithilfe des Werkzeugs Administrator ist die Visualisierung von Prozessinstanzen und umfassendes Prozess-Monitoring möglich. Daten zu Prozessinstanzen können exportiert werden, Prozessmodelle dagegen nicht.

Der Ultimus Administrator ermöglicht das Monitoring laufender Prozessinstanzen. Über ein Query-Interface kann die Menge der gerade relevanten Prozessinstanzen eingegrenzt werden. Diese können über Reports ausgewertet und analysiert werden. Einzelne Instanzen lassen sich als Prozessgraph darstellen. Die Aktivitätszustände sind farbcodiert. Die Art einer Verzweigung kann der Grafik nicht entnommen werden, das Symbol für parallele und alternative Verzweigungen ist identisch. Die Darstellung kann mit unterschiedlichen Zoomstufen erfolgen. Unterprozesse können ausgeblendet werden, die Attribute der stellvertretenden Aktivitäten zeigen dann aggregierte Werte ihrer Sub-Workflows an (z. B. Kosten). Interessant ist, dass sich der Anwender über die Tabellen informieren kann, welche Daten von einer bestimmten Aktivität gelesen bzw. geschrieben worden sind.

Ultimus bietet den Benutzern Arbeitslisten, in denen anstehende Aktivitäten, entsprechend ihrer Dringlichkeit kategorisiert, angezeigt werden können.

IBM Lotus Workflow

Auch Lotus Workflow (LWF) zielt auf formularbasierte Workflows ab. LWF ist keine Workflow-Engine, die Ausführung wird von Lotus Notes verwaltet. Daher ist die Funktionalität im Vergleich zu klassischen WfMS eingeschränkt. Anstelle von beliebigen Anwendungen können nur Notes Dokumente und Datenbanken angekoppelt werden. Mithilfe eines grafischen Editors werden Prozesse modelliert. Diese können in einem XML-Format exportiert werden und enthalten eine vollwertige Beschreibung des Prozesses.

Die Mächtigkeit des Metamodells ist für einfache formularbasierte Workflows ausreichend. Der Datenfluss wird implizit definiert. Zeitaspekte werden nicht unterstützt.

LWF bietet eine Visualisierungskomponente für das Monitoring laufender Prozessinstanzen. Die verwendeten Symbole sind konfigurierbar. Unterstützte Aktivitätszustände sind lediglich 'nicht gestartet' und 'gestartet' (farblich codiert). Pop-up-Fenster zeigen zugehörige Attribute an. Terminüberschreitungen sind nicht darstellbar. Insgesamt bietet der Viewer interessante Ansätze, was Konfigurations- und Visualisierungsmöglichkeiten angeht.

4.2.2. Projektplanung

Die behandelten Prozessmodellierungswerkzeuge und WfMS unterstützen Zeitaspekte und deren Visualisierung nicht adäquat. Daher wird an dieser Stelle exemplarisch das Projekt-Management-Werkzeug MS Project betrachtet.

Microsoft MS Project

MS Project ist ein Projektplanungs- und Verwaltungswerkzeug [Schw03]. Projekte werden durch sequenzielle oder parallele Vorgänge (Aktivitäten) modelliert, denen eine Zeitdauer, Ressourcen und ein Bearbeiter zugeordnet werden können. Das Projekt wird als Gantt-Diagramm (Balkendiagramm) dargestellt, d. h. jede Aktivität wird durch einen Balken repräsentiert, dessen Länge der geplanten Dauer der Aktivität entspricht. Datenflüsse können nicht erfasst werden. Die Zuordnung von Bearbeitern und Ressourcen ist jedoch möglich. Bedingte Verzweigungen und Schleifen können nicht modelliert werden. Laufzeitdaten werden nicht verwaltet, Projektfortschritte werden manuell erfasst. Sie erscheinen als schwarze Balken innerhalb der Aktivitätenbalken. Temporale Aspekte werden gut unterstützt, z. B. lassen sich Pufferzeiten und Fristen darstellen. Die Aktivitäten können hierarchisch angeordnet werden, 'Unterprozesse' lassen sich ausblenden.

4.2.3. Prozessneutrale Visualisierungssoftware

Software-Tools dieser Kategorie haben kein zugrunde liegendes Prozess-Metamodell. Sie sind daher nicht direkt vergleichbar mit den zuvor genannten Tools. Da es in diesem Dokument aber um die Visualisierung von Prozessinformationen geht, denen ein Prozess-Metamodell zugrunde liegt, beschränkt sich die Betrachtung dieser Werkzeuge einzig auf die Art und Weise wie hier Prozesse visualisiert werden und darauf, ob die verwendeten Visualisierungskonzepte auch auf die Darstellung von modellierter Prozessinformation übertragbar sind. Die Verwendung prozessneutraler Visualisierungssoftware verursacht einen höheren Initialaufwand als bei BPM-Werkzeugen, bietet gegenüber solchen prozessorientierten Ansätzen mit 'starrem' Metamodell und eingeschränkter Visualisierungskomponente aber auch zahlreiche Vorteile. Insbesondere lässt sich eine an die Bedürfnisse der jeweiligen Anwendung angepasste Prozessvisualisierung erzielen.

LEITSTAND-VISUALISIERUNG

In Abschnitt 2.2 wurden verschiedene Ansätze zur Prozessvisualisierung diskutiert. Dabei wurden auch Dynamische Darstellungen erwähnt. Die folgenden Werkzeuge unterstützen die Entwicklung solcher Darstellungen.

Natürlich können mithilfe solcher Systeme aber auch 'normale' Prozessdarstellungen erzeugt werden, die sich für das Echtzeit-Monitoring von Prozessinstanzen eignen.

in GmbH sphinx open

Sphinx open (in GmbH) [in -03] erlaubt die Echtzeit-Visualisierung von 2D-Grafiken für das Web und die dynamische Anbindung dieser Grafiken an beliebige Datenbestände. Bisherige Einsatzgebiete von sphinx open sind die Visualisierung von Prozessleitsystemen, Produktionsabläufen, Verkehrsleitsystemen sowie von Steuerungs- und Überwachungssystemen in Energieversorgungsnetzen. Aufgrund seiner generischen Architektur und Anwendungsneutralität eignet sich sphinx open prinzipiell aber auch für die Visualisierung komplexer Entwicklungsprozesse. Mit dem leistungsfähigen Grafikeditor von sphinx open können SVG Grafiken importiert oder aber eigene anspruchsvolle 2D-Vektorgrafiken erzeugt werden. Prinzipiell können somit auch Prozessdiagramme realisiert werden. Die einzelnen Objekte der Darstellung können dynamisch über das sphinx open API manipuliert

werden. Objekte können gruppiert und kombiniert werden und als Schablonen abgelegt werden. Dies ist auch für die Erstellung von Prozessdiagrammen nützlich. Den Objekten lassen sich über das API andere Farbattribute zuweisen, sie lassen sich drehen und verschieben. Das spätere Verhalten lässt sich auch im Editor simulieren. Sphinx open API ist als Java-Klassenbibliothek, als JavaBean und als C/C++-Bibliothek verfügbar. Damit lassen sich, die mit dem Editor erstellten Grafiken in eigene Anwendungen integrieren. Mit den ca. 300 angebotenen Funktionen dürfte nahezu jede Visualisierungsanforderung erfüllt werden. Beispielsweise können neue Grafikobjekte zur Laufzeit erstellt werden und es ist stufenloses Zoomen in den Grafiken möglich. Für Sphinx open existieren Erweiterungsmodule, wie DConnect zur komplexen Datenanbindung (JDBC, Socket, Bussysteme etc.) oder Controls zur Erstellung interaktiver Steuerelemente. sphinx EMP ist ein Enterprise Monitoring Portal. Es erfasst über Standardschnittstellen und Adaptoren für proprietäre Systeme Prozessdaten, Daten aus Produktivsystemen, Leitsystemen und Data-Warehouses in einer Datenbank. Diese Daten werden zentral aufbereitet und über das Inter-/Intranet auch personalisiert zur Verfügung gestellt.

COPA-DATA GmbH zenOn

Die Software zenOn [COPA06] wurde speziell für den Bereich der Industrieautomation entwickelt. Anwendungsbereiche gehen von Maschinensteuerungen bis hin zu Leitstandvisualisierungen. Die Darstellung kann anstatt über die Visualisierungskomponente auch über Webseiten erfolgen. Die Programmier-Schnittstellen sind insgesamt weniger mächtig als die Sphinx open Lösung.

Gefasoft GraphPic

GraphPic [Gefa04] ist ein modulares System für das Monitoring technischer Anlagen, die Messwerterfassung (MDE) und für Leitstände in der Fertigungsumgebung. Grundlage des Systems sind ein Grafikeditor und ein Modul zur dynamischen Darstellung der Grafiken. Messwerte können zur Erfassung und Archivierung in einer Datenbank abgelegt werden. Diese Daten können angezeigt und über Diagramme analysiert werden.

Inosoft VisiWinNET

VisiWinNET [Inos03] zielt, ebenso wie die beiden vorangehend vorgestellten Systeme, auf die Realisierung einfacher Bedien- und Monitoring-Applikationen bis hin zu komplexen Leitstandanwendungen. Auch ansonsten bestehen viele Gemeinsamkeiten zu zenOn und GraphPic. Die Visualisierung basiert auf Visual Basic oder C#, ActiveX und diversen Standardschnittstellen (u.a. COM / DCOM, ODBC).

VISUALISIERUNGSKOMPONENTEN

Im Folgenden werden einige kommerzielle Komponenten vorgestellt, die sich in Eigenentwicklungen für die Prozessvisualisierung integrieren lassen, um als Visualisierungsgrundlage zu dienen.

ILOG Views

Die Firma ILOG bietet Grafikbibliotheken mit Programmierschnittstellen für C++ und Java an [ILOG06]. Diverse Diagramme (z.B. Kuchen- und Liniendiagrammen) und die

für die Prozessvisualisierung wichtigen Graphdarstellungen (mitsamt Layouting), BPMN-Swimlane-Darstellungen und Gantt-Diagramme lassen sich erzeugen. Die erzeugten Grafiken lassen sich für die webbasierte Anzeige nach SVG exportieren. Die Firma bietet außerdem einen frei verfügbaren BPMN-Modeller zum Erstellen von Prozessen in BPMN-Notation an. Für die Visualisierung von Prozessdaten und Organisationsstrukturen ist das angebotene hierarchische Layout interessant. Auch inkrementelles Layout und die Animation von Veränderungen am Layout werden unterstützt.

yWorks yFiles

yFiles ist eine umfangreiche Java Klassenbibliothek [yWor06b], die Layout-Algorithmen und Komponenten bereitstellt für die Visualisierung von Graphen. Weiterhin enthält sie auch eine große Sammlung von Graphenalgorithmien. Unter den unterstützten Layout-Algorithmen sind hierarchisches und orthogonales Layout für die Graphdarstellung interessant. Auch zyklische Graphen werden unterstützt. Das hierarchische Layout unterstützt auch die Anordnung in Swimlanes. Inkrementelles Layout und Animation werden ebenfalls unterstützt. Weiterhin sind die in der Bibliothek enthaltenen Routing-Algorithmen weitestgehend konfigurierbar. Es lassen sich auch nachträglich weitere Kanten zu existierenden Graphen hinzufügen. Zusatzmodule erweitern die Funktionalität z. B. um die Möglichkeit der Ausgabe als SVG-Grafik. Eine Dokumentation der API findet sich in [yWor06a].

oreas GoVisual

GoVisual ist eine plattformunabhängige Grafikbibliothek zur Visualisierung von Graphen mit Programmierschnittstellen für C++, Java, .NET sowie COM [orea04]. Es werden verschiedene Layout-Algorithmen unterstützt, beispielsweise hierarchisches Layout und orthogonales Layout. Allerdings werden Graphen mit Untergruppierungen (engl. *nested graph*) nur für das orthogonale Layout unterstützt.

AbsInt aiSee

aiSee der Firma AbsInt ist das kommerzielle Nachfolgeprodukt [AbsI06] des an der Universität des Saarlandes entwickelten VCG-Projektes (Visualisierung von Compiler Graphen). Ein hierarchisches Layout für die Graphdarstellung wird unterstützt, ebenso wie zyklische Graphen, Untergruppierungen und animierte Layout-Veränderungen. Die Entwickler betonen die schnellen Algorithmen und die Eignung für große Graphen (bis zu 1 Million Knoten). Ein Export nach SVG ist möglich. Die enthaltene Visualisierungskomponente ermöglicht eine Fish-Eye Ansicht. Inkrementelles Layout wird nicht unterstützt. Knoten können nur einen Bezeichner tragen. Beschriftete Kanten werden nicht unterstützt.

4.2.4. Zusammenfassung

Aus der Untersuchung diverser Prozessmodellierungswerkzeuge wurde deutlich, dass diese ausschließlich Prozessschemata visualisieren können, Instanzdarstellungen sind nicht möglich. Die Modellierung erfolgt meist anhand von Prozessgraphdarstellungen. Die Modellierungskomponente von iGrafx (siehe Abschnitt 4.2.1) setzt auf eine Swimlane-Darstellung.

MS Project eine Software aus dem Bereich Projektplanung liefert die Erkenntnis, dass auch Gantt-Diagramme sehr mächtige Werkzeuge darstellen, um Zeitaspekte von Prozessen zu visualisieren.

Werkzeuge zum Prozess-Monitoring unterstützen die Visualisierung von einzelnen Prozessinstanzen, jedoch bestehen für die Darstellung nur sehr wenige Freiheitsgrade. Die gebotenen Visualisierungs- und Interaktionsmöglichkeiten sind im Allgemeinen nicht besonders ansprechend und nur begrenzt konfigurierbar. Diese Ergebnisse sind allerdings nicht verwunderlich, die Stärke solcher Monitoring-Software ist eher die prozessübergreifende Analyse. Der WBI Monitor erlaubt es, Informationen zu Prozessinstanzen in einer konfigurierbaren Tabelle übersichtlich darzustellen. Ultimus zeigt den Einsatz von Tabellen als sortierte Arbeitslisten. Ein weiterer interessanter Aspekt ist, dass in den Werkzeugen Views auf virtuelle Prozessinstanzen hergestellt werden. Das heißt, Einzelprozesse werden zu einem virtuellen Verbundprozess zusammengeschlossen und in dieser Form untersucht. Weiterhin ist der Aufbau von Process-Warehouses ein guter Ansatz für die Prozessdatenintegration [Miha05]. Das Problem der meisten existierenden Lösungen ist denn auch, dass sie meist nur die eigenen Prozessmodelle visualisieren können, das Ziel aber ist eine systemübergreifende Visualisierung.

Die betrachteten Architekturen zur Prozess-Steuerung ermöglichen durch offene Schnittstellen den Zugriff auf Prozessdaten. Dadurch wird es prinzipiell möglich, externe Visualisierungskomponenten für die Darstellung der Prozessinstanzen zu verwenden. Leider ist nur der Import von Prozessmodellen möglich, jedoch nicht der Import von Laufzeitdaten. Damit kommen die integrierten Visualisierungskomponenten nicht für die systemübergreifende Visualisierung in Frage.

Keines der vorgestellten Systeme erfüllt Anforderungen an die Visualisierung von Prozessinstanzen wie:

- Flexible Konfigurierbarkeit der Darstellungen
- Dynamisches Ein-/Ausblenden einzelner Prozessaspekte (z. B. Datenfluss, Aktivitätsattribute)
- Bildung von Views
- Einbeziehung zeitlicher Aspekte (z. B. Visualisierung von Fristüberschreitungen)
- Hohe Ausdrucksmächtigkeit des zugrunde liegenden Metamodells

Die vorgestellten Leitstandvisualisierungslösungen dagegen, zeichnen sich gerade durch eine hohe Konfigurierbarkeit aus, sie eignen sich allerdings von ihrer Funktionalität her besser für das Monitoring industrieller Fertigungsprozesse als für die Visualisierung von Prozessinstanzen. Die in Abschnitt 2.2 vorgestellten dynamischen Darstellungen sind ihre Domäne.

Als Basis für die Darstellung von Prozessgraphen, in einer Prozessvisualisierungskomponente, sind Visualisierungslösungen wie z. B. yFiles, sehr gut geeignet. Sie ermöglichen eine flexible Konfigurierung der Darstellung. Benutzer können zwischen hierarchischem oder orthogonalem Layout wählen. Aspekte der Darstellung können dynamisch eingeblendet werden, inkrementelle Layout-Algorithmen spielen hier ihre Stärke aus. Jedoch muss im Einzelfall überprüft werden, ob die Implementierung der Visualisierungskomponenten auch performant genug ist, für Darstellungen mit vielen Aktivitäten. Anwender tolerieren spürbare Wartezeiten bei Veränderungen an der Darstellung im Allgemeinen nur sehr wenig. Grafische Darstellungen sollten sich ohne Verzögerungen flüssig verschieben und zoomen lassen.

5

Grafische Aspekte der Prozessdarstellung

In Kapitel 2 wurde vorgeschlagen, grafische und strukturelle Aspekte der Prozessvisualisierung zu untersuchen. Diese beiden Ansätze sollen jeweils unabhängig voneinander zu einer verbesserten Prozessvisualisierung beitragen.

In diesem Kapitel werden die grafischen Aspekte diskutiert. Die grafische Repräsentation von Prozessgraphdarstellungen soll optimiert werden. Die wichtigsten Einzelaspekte von grafischen Darstellungen sind Farben und Formen. Das Aussehen von Prozessgraphen wird durch Notationen, die u. a. die Form von Knoten und Kanten beschreiben festgelegt. Auf der Grundlage der in Abschnitt 4.1 vorgestellten Konzepte für die Informationsvisualisierung, wird eine grafische Notation für die Graphdarstellung entwickelt. Diese soll der menschlichen Wahrnehmung entgegenkommen, aber auch rein praktischen Gesichtspunkten Rechnung tragen, um etwa kompakte Darstellungen zu ermöglichen.

5.1. Farben

Farben sind nicht nur Gestaltungsmittel. Wenn Farben konsistent immer wieder für dieselben Dinge stehen, erhöhen sie den Wiedererkennungswert. Bei der Verwendung von Farben gilt es einiges zu beachten. Beispielsweise sollte die direkte Kombination von Vordergrund- und Hintergrundfarbe aus den drei Grundfarben Rot, Grün und Blau unbedingt vermieden werden. Wegen der stark unterschiedlichen Wellenlängen hat das Auge Schwierigkeiten, jeweils beide Farben zu fokussieren und scharf zu sehen [This00, S.93] (Chromostereopsis-Effekt).

Die Farbe Rot sollte nicht für normale Inhalte in den Darstellungen verwendet werden. Sie bietet sich an, um wichtige Dinge zu markieren, denn das menschliche Auge ist nicht, wie es zu erwarten wäre, den Verhältnissen der Natur angepasst, wo verschiedenste Grüntöne vorherrschen. Vielmehr hat der Mensch zu 64% Sinneszellen für Rot, zu 34% für Grün und nur zu 2% für Blau. Und das, obwohl in der Natur die Farbe Rot am seltensten vorkommt, z. B. bei der Glut des Feuers und bei Blut [This00, S.125].

Der *HSL-Farbraum* bietet sich an, um aufeinander abgestimmte Farben für die Darstellung zu finden. Stark gesättigte Farben wirken schwer und haben die Tendenz sich in den Vordergrund zu drängen. Weniger gesättigte Farben lassen sich besser als Hintergrundfarbe einsetzen.

“Ein buntes Bild wirkt unruhig, ein Bild das aufeinander abgestimmte Farben verwendet wirkt harmonisch und ruhig.” (Frank Thissen) [This00, S.126]

Eine Einführung, wie man solche Farbkombinationen generiert, liefert John Deceber's Webseite [Dece05]. Eine weitere Webseite beschäftigt sich auch mit der Harmonielehre, sie bietet zum interaktiven Ausprobieren auch einen Farbmischer [RH00].

Farben sind ein wichtiges Gestaltungsmittel. Farben ziehen die Aufmerksamkeit auf sich, das ist ein großer Vorteil. Sie können dazu eingesetzt werden, dass Benutzer sich schneller orientieren und schneller zum Ziel navigieren können. Es ist jedoch wichtig, darauf zu achten, dass sich die Farbe gut in das Zusammenspiel mit dem anderen Gestaltungsmitteln, also Symbolen und Text integriert. Zu viele Farben machen Darstellungen unübersichtlich. Zum Markieren sollten, wegen der Limitierungen des menschlichen Kurzzeitgedächtnisses, maximal sieben Farben gleichzeitig verwendet werden (vgl. Abschnitt 4.1.1). Es ist allerdings ohnehin so, dass beim Menschen die Farbwahrnehmung auf 6 bis 12 gleichzeitig dargestellte und dabei noch sicher unterschiedene Farben begrenzt ist. Wo dennoch zwingend eine eindeutige Farbkodierung vieler unterschiedlicher Zusammenhänge gefordert ist, ist die Kombination von Farbtönen, Helligkeiten und Mustern eine gute Lösung, eine Vielzahl eindeutiger Kodierungen zu erzeugen.

5.1.1. Kontraste

Häufig geht es aber nicht darum Dingen explizit eine Farbe zuzuweisen, sondern dem Auge Kontraste anzubieten, an denen es Strukturen erkennen kann. Leichte Kontraste dienen dazu, die Struktur einer Informationsdarstellung zu betonen. Auf hohe Kontraste sollte überall dort geachtet werden, wo Text in einer Darstellung erscheint oder wo Teile der Darstellung die Aufmerksamkeit auf sich ziehen sollen. Hohe Kontraste erleichtern das Lesen und Betrachten und helfen Ermüdungserscheinungen entgegenzuwirken. Ihr Einsatz sollte aber durchaus gezielt erfolgen, da hohe Kontraste dazu führen von anderem abzulenken, wie Abbildung 5.1 zeigt.

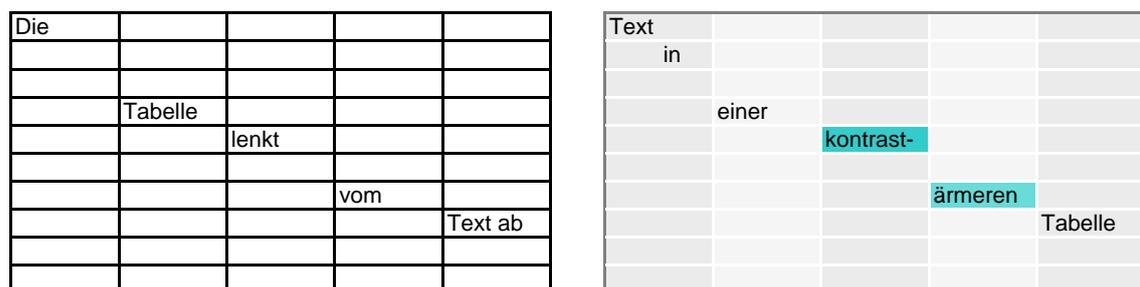


Abbildung 5.1.: Wirkung von Kontrast in Tabellen

Eine Tabelle lässt sich beispielsweise am besten lesen, wenn Text möglichst kontrastreich erscheint und auf einen leichten Kontrast zwischen den Zeilen und Spalten geachtet wird.

Der Betrachter profitiert von diesen Führungslinien, da sie helfen, beim Betrachten einer Zeile nicht zu verrutschen. In hellen Farbtönen hinterlegte Zellen können zusammengehörige Informationen markieren (siehe Abbildung 5.1).

Durch Farben und Kontraste lassen sich Informationsgruppen optisch voneinander trennen oder absichtlich einander zuordnen [CR03, S.242]. Unterschiedlich starke Kontrastabstufungen können für unterschiedlich starke Zusammenhänge zwischen Informationen stehen. Besonders große Kontraste dienen als bewusste Trenner zwischen Informationen (siehe Abschnitt 5.2).

Zur Unterscheidung einander ähnlicher Informationen können Farben dienen, die im HSL-Farbraum sehr nah beieinander liegen und sich nur in der Farbhelligkeit leicht unterscheiden. Das Modifizieren der Helligkeit erzeugt einen Effekt der Nähe für das menschliche Auge. Farben für einander unähnliche oder gar konträre Informationen lassen sich finden, indem man sie so wählt, dass sie im HSL-Farbkreis möglichst weit voneinander entfernt liegen. Genau gegenüber liegt immer die Komplementärfarbe mit dem größtmöglichen Kontrast. Der optische Effekt von Ferne kommt durch starkes Ändern des Farbtones zustande. Diese optische Nähe (Ferne) assoziiert das Gehirn gleich mit der logischen Nähe (Ferne) der dargestellten Informationen.

5.2. Ebenen

Ziel dieser Arbeit ist es, möglichst viel nützliche Information aus Prozessinformationen zu extrahieren und derart aufzubereiten, dass der Anwender es möglichst einfach hat, die gesuchten Informationen zu finden. Die Prozessinformationen bestehen jedoch nicht nur einfach aus Einzelinformationen. Aus dem Zusammenhang lässt sich weitere Information extrahieren, da zwischen den Einzelinformationen Verbindungen auf inhaltlicher Ebene bestehen.

Letztlich ist ein Anzeigemodul für Prozessinformationen ein Informationssystem. Die verschiedenen Informationssorten, die ein Informationssystem darstellen soll, stehen in einer Art Verwandtschaftsbeziehung (Relation) zueinander, wobei eine Informationssorte zur anderen, entweder eine große Nähe oder Distanz aufweisen kann.

Ein Beispiel hierfür sind Relationale Datenbanken, ihre Tabellen sind durch Primär- und Fremdschlüssel untereinander verknüpft. Zwei Tabellen können so nun direkt miteinander verbunden sein oder aber entfernt über mehrere andere Tabellen.

Wenn nun verschiedene Informationssorten in einer Darstellung gemischt werden, kann man den Benutzer darin unterstützen, diese verschiedenen Informationskategorien auseinander zu halten. Üblicherweise geschieht dies durch unterschiedliche Notationen der grafischen Repräsentationsformen. Das menschliche Gehirn profitiert jedoch davon, wenn möglichst viele Unterschiede erkennbar sind. Jedes erkennbare Merkmal trägt dazu bei, dass das Gehirn Informationen kategorisiert.

Was bedeutet nun Nähe und Distanz bei Prozessinformationen? Letztlich geht es um Objekte und um ihre Beziehungen untereinander. Folgende Objekte werden in den Darstellungsformen unterschieden:

- Aktivitäten

- Ausführungseinheit
- Systeme
- Daten/Dokumente

Diese Objekte bilden die Knoten des darzustellenden Graphen¹ (siehe dazu auch 5.3). Der Kontrollfluss, der die Prozessstruktur abbildet und der Daten-/Dokumentfluss bilden die Kanten des Graphen, d. h. die Beziehungen zwischen den Objekten ab.

In einer Prozessgraphendarstellung gilt es beispielsweise die Einzelinformationen wie (Aktivitäten-)Knoten, Kontrollfluss und Daten-/Dokumentfluss optisch voneinander zu trennen. Die Aktivitäten und der Kontrollfluss haben dabei viel miteinander zu tun, sie stehen sich nah.

Diese verschiedenen Informationsebenen können durch unterschiedliche Farbgebung dargestellt werden. Dies bietet sich an, da Farben sich gut eignen, um Objekte in Kategorien einzuordnen [KDV02]. Abbildung 1.3 (→ S. 7) zeigt, dass Gruppen von ähnlichen Farben (Farbton) für sich jeweils nah verwandte Informationen stehen können. Für Dokumente wird hier beispielsweise ein helles Gelb, für den Dokumentfluss ein sattes Gelb verwendet. Aktivitäten und Kontrollfluss dagegen, sind in Schwarz und Grautönen gehalten. Auf diese Weise kann der Mensch leicht zwischen Kontroll- und Datenfluss unterscheiden, aber gleichzeitig auch wahrnehmen, dass Dokumente und Dokumentfluss zusammengehören. Wo mehrere solcher Farbgruppen gleichzeitig verwendet werden, sollten sie untereinander hohe Kontraste aufweisen, um dem Auge die klare Trennung zwischen den Farbgruppen leichter zu machen (siehe Abschnitt 5.1.1).

Van Vliet et al. [KDV02] weisen darauf hin, das bei mehreren solcher visuellen Ebenen, eine Ebene eindeutig als Hauptebene identifizierbar sein sollte. Dies kann beispielsweise durch leuchtende Farben, starke Kontraste, dicke Linien oder platzieren in der Darstellungsmitte geschehen.

Unter Anwendung von Transparenz können sich Informationsebenen auch lokal überlappen. Damit können Verbindungslinien auch durch Objekte einer anderen Ebene geführt werden, ohne dass die Lesbarkeit sonderlich darunter leidet. Die Linienführung (engl. *Routing*) unterliegt damit weniger Bedingungen. Linienzüge können so kürzer ausfallen und reduzieren die Komplexität der Darstellung. Colin Ware weist jedoch auch daraufhin, das Transparenzen auch die Wahrnehmung stören können [Ware00, S.222].

5.3. Notationen

Um Geschäftsprozesse darzustellen, bietet es sich an, grafische Notationen von Standards wie *BPMN* [Bpmi03a] von der BPMI [Bpmi03b] oder UML 2.0 [Omg 05b] von der OMG [Omg 05a] zu verwenden. Die erste Version des UML Standards wurde 1997 veröffentlicht. Der erste Entwurf zu BPMN erschien 2003. Die älteste Spezifikation sind die noch heute genutzten IDEF-Sprachen [NIST93a, NIST93b]. Sie entstanden in den 80er Jahren und wurden zuerst vom amerikanischen Verteidigungsministerium genutzt. Die beiden neueren Standards haben den Vorteil, das die resultierenden Diagramme nicht nur von Fachleuten gelesen werden können. Darum dienen sie dieser Arbeit ebenfalls als Grundlage der

¹Es können auch zwecks Übersichtlichkeit mehrere Objekte zu einem Knoten gebündelt werden

weiteren Betrachtungen. Notationen, die sich daran anlehnen, profitieren vom Wiedererkennungswert², da UML und BPMN inzwischen eine recht hohe Verbreitung gefunden haben und sich relativ ähnlich sind. Ein Vergleich dieser beiden Notationen findet sich in [Whit04].

Die beste Notation für eine Prozessvisualisierungskomponente ist aber nicht notwendigerweise BPMN oder UML, denn bei beiden liegt der Fokus auf Genauigkeit und Unmissverständlichkeit. Das Ziel dieser Arbeit gebietet jedoch als oberste Prämisse, die Übersichtlichkeit beim Darstellen komplexer Prozesse zu gewährleisten. Daher sind die folgenden Betrachtungen für eine geeignete Notation nur von Usability-Maßstäben geleitet.

5.3.1. Knoten

Es gilt folgende Objekte in der Darstellung unterzubringen: Aktivitäten, Ausführungseinheit, Systeme und Daten/Dokumente. Die erste Maßnahme zu mehr Übersichtlichkeit, beispielsweise gegenüber BPMN, ist, alle diese Objekte in einem Aktivitätenobjekt zu kombinieren. Das bringt folgende Vorteile mit sich:

- Eine Prozessdarstellung besteht aus weniger Objekten (Knoten)
- Eine Prozessdarstellung besteht aus weniger verschiedenen Symbolen (äußerlich unterscheidbare Knoten)
- Die Darstellung ist weitgehend unabhängig von der Anzahl eingeblendeter Prozessaspekte
- Es werden in der Darstellung keine Verbindungskanten mehr benötigt, die die verschiedenen Objekte mit dem Aktivitätenobjekt verbinden

In Netzplänen werden von der Wahrnehmung alle umschlossenen Konturen als Knoten erkannt. Es geht darum, an die Wahrnehmung möglichst gut angepasste Knotenformen zu verwenden. Ein Knoten hat eine bestimmte Form, eine bestimmte Größe, eine bestimmte Farbe und praktische Aspekte, wie z. B. eine gute Unterteilbarkeit für die Darstellung von Unterobjekten.

Eine naheliegende Form sind Rechtecke. Denn unter den Gesichtspunkten effektive Platzausnutzung, guter Beschriftbarkeit und guter Unterteilbarkeit sind sie gegenüber allen anderen Formen im Vorteil.

Eckige Formen, wie Rechtecke oder sonstige Vielecke, sind letztlich Konturen, die aus Linien mit abrupten Richtungsänderungen bestehen. Es ist aber für das Auge viel einfacher, weichen Rundungen zu folgen [Ware00, S.207]. Da nun aber andere runde Formen wie Kreis und Ellipse sich nicht so praktisch verwenden lassen, liegt es nahe, als bevorzugte Knotenform Rechtecke mit abgerundeten Ecken zu verwenden.

Abbildung 5.2 zeigt das Ergebnis dieser Symbolintegration.

Natürlich können auf diese Weise auch diverse Attribute wie Kosten oder Bearbeitungsdauer direkt in der Darstellung angezeigt werden. Wenn alle Knoten die gleiche Größe aufweisen, führt das zu einer strukturierten übersichtlichen Darstellung. Das Knoteninnere ist leicht eingefärbt, um das Objekt mehr vom Hintergrund abzuheben.

²siehe Usability Kapitel 4.1

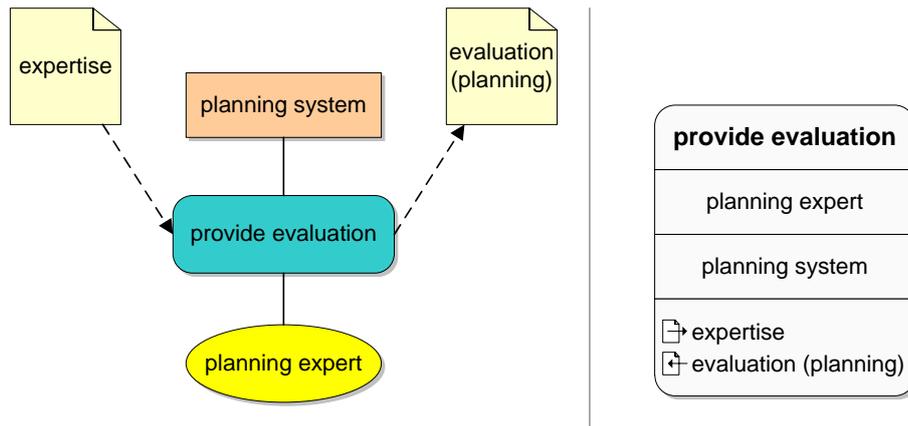


Abbildung 5.2.: Symbolreduktion durch Integration

METAPHERN

Den Benutzern kann bei der Orientierung durch den gezielten Einsatz von Metaphern geholfen werden. Der Vorteil von Metaphern ist, dass der Umgang mit ihnen nicht erlernt werden muss. Mit Metaphern ausgezeichnete Informationen sprechen für sich (siehe Abbildung 5.3). Es ist im Allgemeinen nicht möglich, nur Metaphern zu verwenden, sobald das der Visualisierung zugrunde liegende Metamodell benutzerdefinierte Attribute zulässt. Hier wäre es zu aufwendig, solchen Attributen manuell passende Metaphern zuzuordnen. Wenn Metaphern verwendet werden sollen, bietet es sich daher an, einen Satz einfach einprägsamer Metaphern für die häufigsten Aktivitätsattribute bereitzuhalten und alle anderen Attribute schriftlich zu benennen.

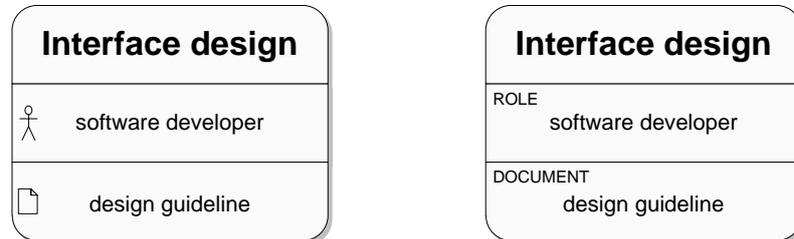


Abbildung 5.3.: Nutzung von Metaphern

MARKIERUNGEN

Häufig ist es sinnvoll, einer Darstellung mehr Aussagekraft zu verleihen, indem einzelne Elemente besonders hervorgehoben werden. Üblicherweise geschieht dies durch besondere Farbgebung (siehe Abschnitt 5.1) [CR03, S.242]. Ein bestimmter Informationsaspekt wird dazu ausgewertet und je nach Ergebnis werden die Elemente in der Darstellung markiert. Bekannt ist dieses Prinzip von Tabellenkalkulationen, diese bieten Funktionen an, um Geldbeträge grün oder rot hinterlegt darzustellen, je nachdem ob es sich um positive oder negative Beträge handelt.

Dieses Prinzip kann auch bei der Prozessvisualisierung angewendet werden. Denn für den Benutzer sind nicht alle Informationen gleich wichtig, was aber in einer Darstellung, in der

alles gleichmäßig dargestellt ist, suggeriert wird [CR03]. Abbildung 5.3 zeigt die folgenden Markierungsmöglichkeiten:

- > Farbhinterlegung
- > Umrahmungen
- > Aktivitäten Ampel (Licht: rot, gelb oder grün)
- > Aktivitäten Symbol

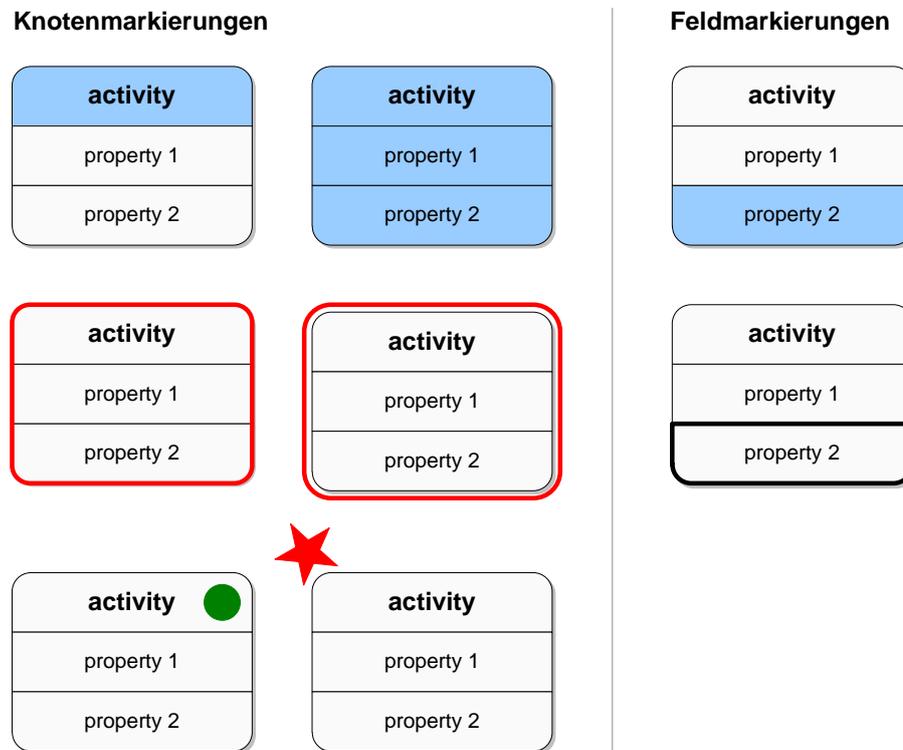


Abbildung 5.4.: Knoten und Feldmarkierungen

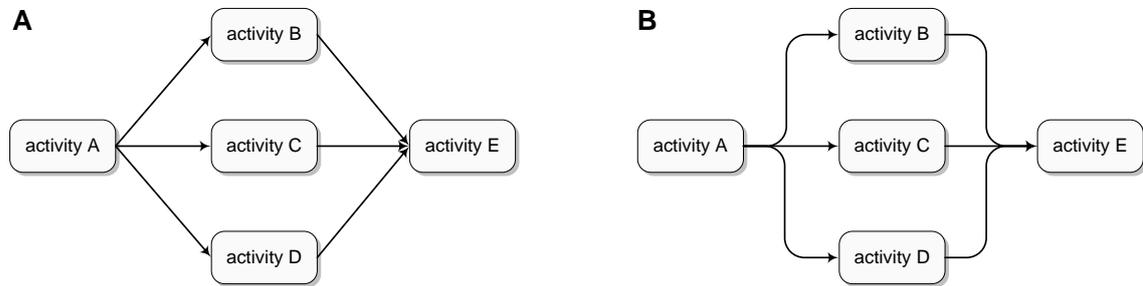
Die Ampel kann beispielsweise dazu dienen unterschiedliche Kosten- beziehungsweise Zeitspannen zu visualisieren. Während die übrigen Markierungsarten eher einzelne Sachverhalte herausstellen.

5.3.2. Kanten

Die Aufgabe von Verbindungskanten ist es, Verbundenheit zwischen Knoten auszudrücken. Die menschliche Wahrnehmung versucht mit dem Auge Strukturen zu erfassen, dabei orientiert es sich stark an eckigen Kanten. Eckige Kanten lenken die Aufmerksamkeit auf sich. Da die Kanten eines Prozessgraphen aber in sich keine Information tragen, außer durch eine mögliche Beschriftung, sollte es vermieden werden, dass die Darstellung der Kanten viel Aufmerksamkeit auf sich lenkt. Bei der Darstellung von Kanten sollten daher abrupte Richtungsänderungen vermieden werden [Ware00, S.207]. Dies geht auf das Gestaltgesetz der 'Guten Fortsetzung' zurück. D. h. es bleibt die Wahl zwischen geraden Direktverbindungen und rechtwinkliger Linienführung mit gebogenen Kanten. Prinzipiell möglich wäre

damit auch eine organische Linienführung, aber dem Auge fällt es bei geordneterer Linienführung einfacher Strukturen zu erkennen. Abbildung 5.5 zeigt Möglichkeiten, auf welche Art und Weise sich zwei Objekte miteinander verbinden lassen.

Wahrnehmungspsychologisch günstige Linienführung



Ungünstige Linienführung

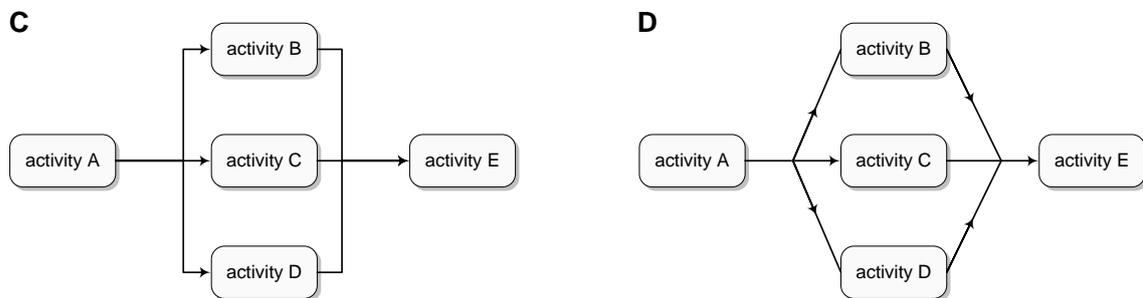


Abbildung 5.5.: Variationen von Kanten

Die grafische Repräsentation einer Pfeilrichtung sollte möglichst wenig Aufmerksamkeit auf sich ziehen. Eventuell kann sogar darauf verzichtet werden, eine Pfeilrichtung abzubilden. Die räumliche Anordnung der Knoten und ein expliziter Startknoten können die konzeptuelle Ordnung auch ohne Pfeilspitzen repräsentieren (siehe Gestaltungshinweis Nr. 7 in Abschnitt 4.1.1).

Für die folgenden Kantentypen sollten unterschiedliche grafische Repräsentationen verwendet werden:

➤ **Kontrollflusskante**

Es existieren noch weitere Untertypen, wie Schleifen (engl. *Loop*) und Ausnahmen (engl. *exception*), die wiederum unterscheidbar sein sollen.

➤ **Datenflusskante**

Auch hier existieren noch weitere Untertypen, wenn zwischen Zugriffstypen wie Lesen, Schreiben unterschieden werden soll.

Die im Abschnitt 5.2 über Ebenen schon angesprochenen Farbebenen kommen hier wieder zum Zuge. Die beiden Obertypen von Flusskanten können über unterschiedliche Farbgebung verschiedenen Ebenen zugeordnet werden, sodass das Auge sich auf einzelne Informationsebenen der Darstellung konzentrieren kann. Zur weiteren Abgrenzung können auch unterschiedliche Linienstärken, Farbhelligkeiten oder Muster zur Anwendung kommen. Es

bietet sich allerdings an, diese letztgenannten Diversifikationsmöglichkeiten den jeweiligen Untertypen vorzubehalten. Eine Besonderheit stellen Kanten für die Ausnahmebehandlung dar, diese gehören zwar zum Kontrollfluss, sollten aber dennoch auch farblich von diesem getrennt werden, beispielsweise durch die Signalfarbe Rot.

SEMANTIK

Ein sequenzieller Kontrollfluss ist leicht nachzuvollziehen. Für den Fall, dass mehrere Kontrollflusskanten eine Aktivität zum Ziel haben oder eine Aktivität verlassen, werden üblicherweise grafische Repräsentationen für die semantischen Verzweigungs- und Verknüpfungsoperationen verwendet, um es den Anwendern zu erleichtern den Kontrollfluss einfach nachzuvollziehen. Die möglichen Verzweigungstypen gibt das zugrunde liegende Metamodell vor. Im Folgenden sind die Standardoperationen aufgelistet:

➤ **Oder-Verzweigung**

Ein oder mehrere Pfade werden aufgrund von erfüllten Bedingungen gewählt.

➤ **Und-Verzweigung** (Parallelisierung)

Alle ausgehenden Pfade werden gewählt (→Alle Bedingungen sind identisch).

➤ **Exklusiv-Oder-Verzweigung**

Genau ein Pfad wird gewählt. Dies lässt sich erreichen, indem sich die Übergangsbedingungen gegenseitig ausschließen.

➤ **Und-Verbindung**

Zur Weiterschaltung müssen alle eingehenden Pfade abgearbeitet sein (→Die Aktivität bildet Rendezvous-Punkt).

➤ **Oder-Verbindung**

Zur Weiterschaltung muss nur ein eingehender Pfad abgearbeitet sein.

Dies sind die üblichen Join- und Split-Fälle, komplexere Metamodelle können weitere Routing-Regeln anbieten, etwa eine 'N-aus-M'-Join Regel.

Um die grafische Komplexität gering zu halten, sollten möglichst wenige Symbole eingeführt werden. Es ist möglich, komplett auf grafische Symbole für die Standard Join/Split-Fälle zu verzichten. Für die Und-Verzweigung und für die Und-Verbindung muss kein grafisches Symbol verwendet werden, sie sind beide ohnehin nicht an Bedingungen geknüpft, die dargestellt werden müssten. Auch die anderen Verknüpfungsarten können ohne Symbol auskommen und trotzdem eindeutig sein. Bei (Exklusiv-)Oder-Verzweigungen werden hierfür die jeweiligen Weiterschaltungsbedingungen als Kantenbeschriftungen angezeigt. Oder-Verbindungen können mittels 'Oder' als Kantenbeschriftung eindeutig gemacht werden.

Weitere (komplexere) Split/Join Konstrukte können bei Bedarf zusätzlich eingeführt werden. Zur Unterscheidung von den bereits angesprochenen Semantik-Konstrukten werden sie durch ein kleines Verzweigungssymbol (mit entsprechender Beschriftung) eingeleitet (Split) oder abgeschlossen (Join). BPMN nutzt einen Diamanten als Entscheidungssymbol, wie es aus Flussdiagrammen bekannt ist und UML einen Diamanten oder einen kleinen Balken, je nachdem ob es sich um eine 'Oder' oder 'Und' Verzweigung handelt.

Abbildung 5.6 zeigt einige Variationen der Darstellungsmöglichkeiten. Der Vorteil eines dedizierten Symbols für Oder-Verzweigungen ist, das auf die Einblendung der Bedingungen verzichtet werden kann. Bei Bedarf sind die Bedingungen via Tooltipp abrufbar.

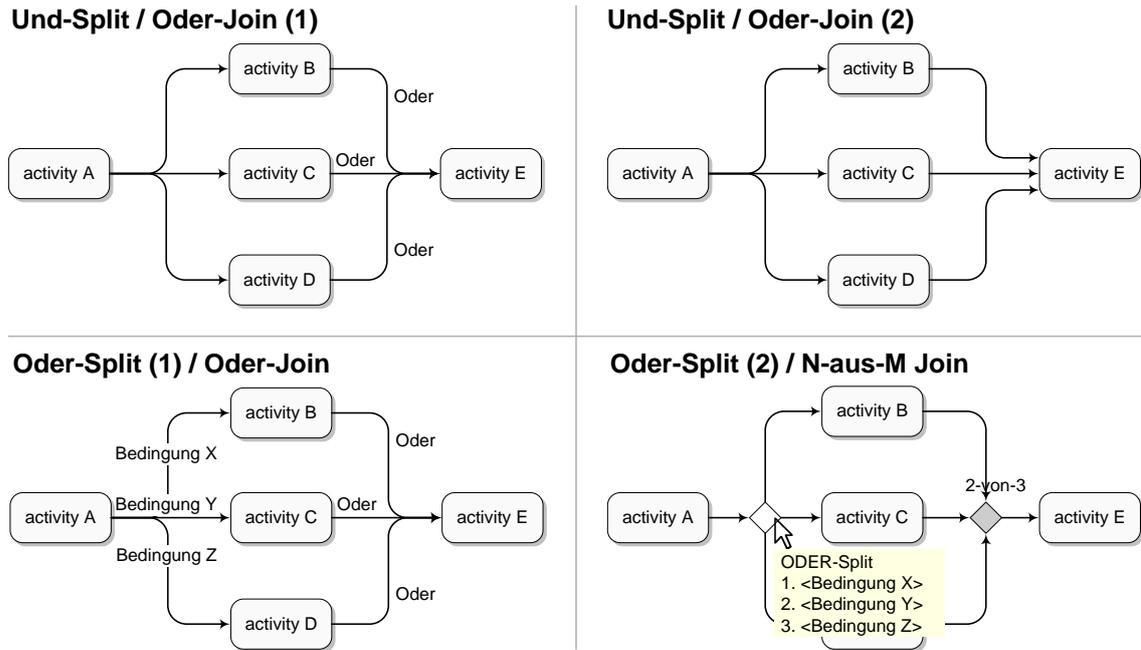


Abbildung 5.6.: Semantik Darstellungsmöglichkeiten

Blockstruktur

Bei Prozessen, die symmetrische Blöcke aufweisen, kann dem Betrachter die resultierende Blockstruktur sehr einfach wie in Abbildung 5.7 grafisch anschaulich dargestellt werden. Dies hilft bei komplexen Kontrollflüssen den Überblick über die Semantik des Prozessgraphen zu behalten bzw. ihn in kurzer Zeit zu durchschauen. Die Abbildung zeigt auch, dass diese Blöcke auch zu Interaktionszwecken dienen können. Der linke Block ist ausgewählt und stellt dem Benutzer eine 'Minimieren'-Schaltfläche bereit, um die Darstellung der Verzweigung einzuklappen.

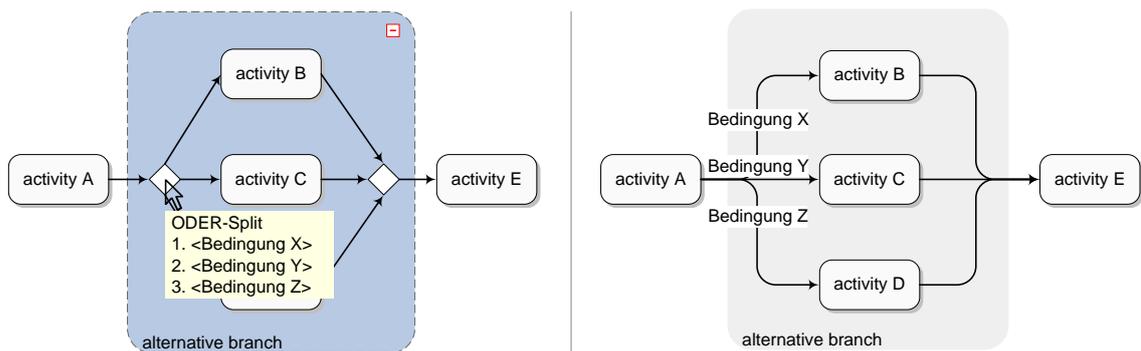


Abbildung 5.7.: Verdeutlichen der Blockstruktur

KONTROLLFLUSSKANTE

Das Prozess-Metamodell kann verschiedene Kantentypen unterscheiden. Eine Unterstützung verschiedener Typen hat den Vorteil, dass bei der Visualisierung selektiv einzelne Kantentypen dynamisch ein- und ausgeblendet werden können.

Die folgende Liste zeigt verschiedene Kantentypen:

- Normale Kontrollflusskante
- Sprungkante
- Ausnahmekante, wird bei Ausnahmebehandlungen aktiviert
- Schleifenkante
- Synchronisationskante (Sync-Kante)

Bei allen Kantentypen, mit Ausnahme der Sync-Kante, handelt es sich um Ausführungskanten auf denen die Weiterschaltung der Prozessausführung stattfindet. Sync-Kanten werden eingesetzt, wenn z. B. in parallelen Zweigen gewisse Reihenfolgebeziehungen eingehalten werden müssen.

Sprünge, Verweise und Prozessschnittstellen

Sprungkanten repräsentieren in erster Linie ebenfalls Kontrollflüsse, diese Ähnlichkeit sollte auch in der grafischen Darstellung einer Sprungkante wiederzufinden sein. Es ist aber denkbar, ihnen eine leicht abweichende Farbe zuzuweisen, die sofort verdeutlicht, dass es sich hier nicht um eine normale Kontrollflusskante handelt. Es ist wichtig, dem Benutzer anzudeuten, ob es sich um einen weiten oder einen nahen Sprung handelt. Eine Analogie stellen Links auf Webseiten dar, der Benutzer schätzt es, wenn durch die Auszeichnung eines Links für ihn vorhersehbar ist, ob er mit einem Klick darauf, nur innerhalb der Seite navigiert, oder ob er damit den aktuellen Bereich verlässt und auf eine andere Seite wechselt.

Üblicherweise wird immer nur ein einzelner Prozess aus einer Prozesshierarchie dargestellt. Prozesse können aber Sprungkanten enthalten, die von einem Prozess zu einem anderen verweisen. Diese Verweise (Sprünge) benötigen eindeutige Zielangaben. Navigation in hierarchischen Prozessen erfordert daher einen eindeutigen Namensraum.

Zwischen drei Arten von Sprüngen kann unterschieden werden³:

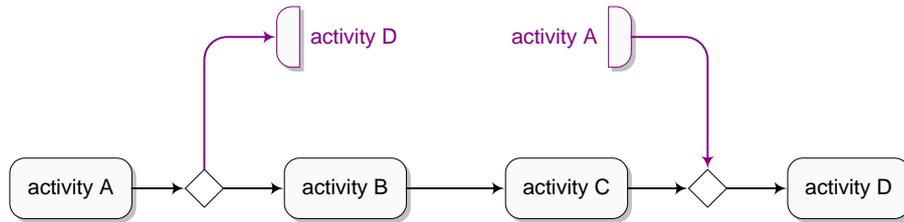
1. Sprünge zu anderen Aktivitäten innerhalb des Prozesses.
2. Sprünge in benachbarte Prozesse derselben Schicht
3. Sonstige Sprünge zu anderen Schichten eines hierarchischen Workflows

Fall 1 wird nur bei Bedarf eingesetzt, um innerhalb eines Prozesses lange Verbindungskanten zu umgehen, die das Lesen des Prozesses erschweren. In den Fällen 2 und 3 kann auch von Prozessschnittstellen gesprochen werden, da hier nicht Aktivitäten das Ziel sind, sondern andere Prozesse. Statt eines Verweises auf einen anderen Prozess, können die beteiligten Prozesse auch zu einer Gesamtdarstellung 'verwebt' werden. Dies erscheint aber nur bei wenig komplexen Prozessen sinnvoll. Bei Fall 2 reicht eine relative Zielangabe, bei

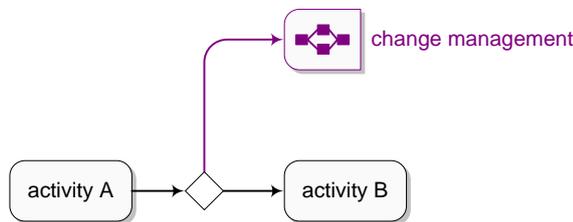
³Abbildung 7.7 (→ S.133) zeigt zur Verdeutlichung einen Ausschnitt aus einer Prozesshierarchie. Ein Sprung zu einem benachbarten Prozess wäre hier beispielsweise von 'bewerten' zu 'kommentieren'.

Fall 3 ist eine absolute Zielangabe aus dem Namensraum notwendig. Eine schriftliche Beschreibung einer absoluten Zielangabe wäre z. B. der Prozess <production> der obersten Ebene besitzt einen Unterprozess <quality management>, dort soll wiederum der Unterprozess <change management> aufgerufen werden. Die grafischen Repräsentationen der Fälle 1 bis 3 zeigt Abbildung 5.8.

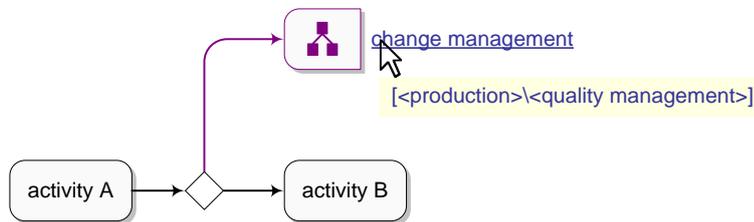
Sprung zu anderer Aktivität



Naher Sprung zu benachbartem Prozess (Zielangabe relativ)



Weiter Sprung in Prozesshierarchie (Zielangabe absolut)



Sprung in Prozesshierarchie mit Rückkehr

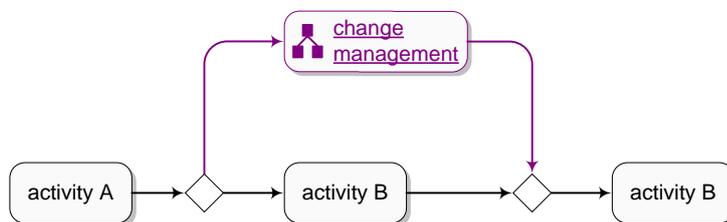


Abbildung 5.8.: Darstellung von Prozessschnittstellen

DATENFLUSSKANTE

Da es Lese- und Schreibkanten gibt, sollte dies auch eine Darstellung klar ersichtlich wiedergeben. Zwei verschiedene, aber dennoch genügend Ähnlichkeit aufweisende Repräsentationen werden also benötigt. Zwei Möglichkeiten wären beispielsweise:

- Dünne Lesekante und dicke Schreibkante
- Heller Farbton der Lesekante und satter Farbton der Schreibkante

Die Symbolik soll dabei ein Überschreiben eines Wertes andeuten, die auffallendere Linie überschreibt die Andere.

5.3.3. Ausführungsmarkierungen

Sowohl Aktivitäten, als auch Kontrollflusskanten können Informationen über den Ausführungszustand transportieren.

Die möglichen Kantenzustände sind:

- nicht aktiviert
- aktiviert
- deaktiviert (diese Kante wird nie durchlaufen)

Kanten kommen in den Zustand 'deaktiviert', wenn ihr Pfad bei einer Verzweigung nicht ausgewählt wurde.

Die Ausführungszustände von Aktivitäten können vielfältiger sein. Hier nur einige Beispiele:

- nicht bereit
- bereit
- laufend
- abgebrochen
- fehlgeschlagen
- abgeschlossen

Eine Prozessvisualisierungskomponente muss dem Anwender bei der Darstellung von laufenden Instanzen die Aktivitätsausführungszustände anzeigen. Auch bei abgeschlossenen Instanzen ist dies hilfreich, um gewählte und abgewählte Verzweigungspfade im Nachhinein nachvollziehen zu können. Die Kantenmarkierungen sind hingegen weniger wichtig, schließlich kann durch die Zustände der Aktivitäten auch auf sie geschlossen werden.

Mögliche grafische Repräsentationen sind Knotenfarben oder speziell eingblendete Symbole. Gegen Knotenfarben spricht, dass diese die Markierungsfunktion einschränken (siehe Abschnitt 5.3.1). Abbildung 5.9 zeigt einige Symbole, diese können z. B. links oben über den Aktivitätenknoten eingblendet werden oder innerhalb des Knotens, links des Aktivitätsnamens.



Abbildung 5.9.: Ausführungszustände von Aktivitäten (bereit, laufend, abgebrochen, fehlgeschlagen, abgeschlossen)

5.4. Orientierung

Darstellungen sollten vom Anwender so konfiguriert werden können, dass die resultierenden Graphen von links nach rechts, oder von oben nach unten ausgerichtet angezeigt werden können. Diese im westlichen Kulturkreis üblichen logischen Anordnungen zeigen die Abbildungen Abbildung 5.11 und 5.10. Eine internationale Software sollte jedoch auch bei Bedarf an das asiatische Denkmodell anpassbar sein und von rechts nach links laufende Prozessdarstellungen anbieten.

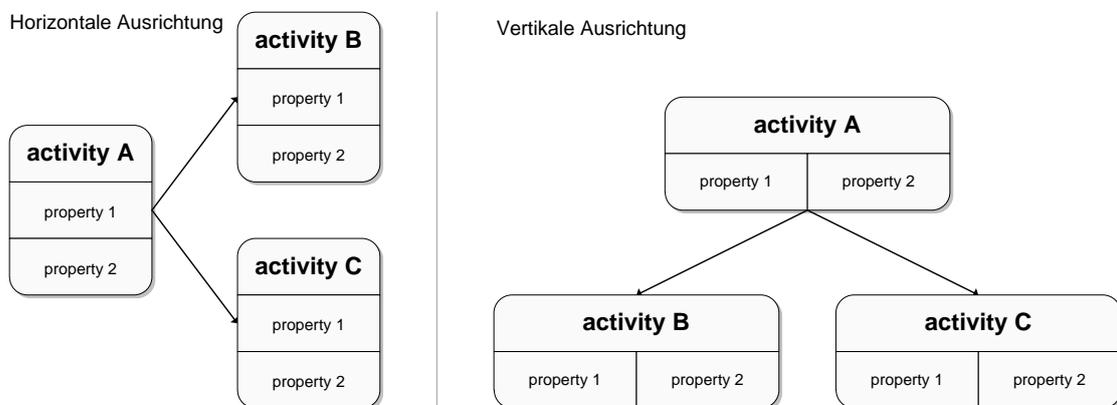


Abbildung 5.10.: Knoten bei horizontaler oder vertikaler Prozessdarstellung

Abbildung 5.11 zeigt Knotenkonfigurationen für die vertikale und horizontale Graphdarstellung. Je nachdem wie viele zusätzliche Attribute direkt in der Darstellung erscheinen sollen, wird eine entsprechende Knotenrepräsentation verwendet. Das Ziel der, für die vertikale und horizontale Darstellung, jeweils angepassten Knotenkonfigurationen ist eine effiziente Platzausnutzung, um möglichst viele Aktivitäten auf einer Bildschirmseite darstellen zu können.

5.5. Zusammenfassung

In diesem Kapitel wurde eine praktische grafische Notation für Graphdarstellungen entwickelt. Unter anderem wurde hierfür der effiziente Einsatz von Farben und Kontrasten angesprochen sowie getrennte Informationsebenen beispielsweise für Kontroll- und Datenfluss.

Den Ausschnitt eines Change-Management-Prozesses, sowohl in BPMN als auch im Vergleich dazu die hier entwickelte Notation, zeigten schon die Abbildungen 1.2 und 1.3 (→S. 6), auf die an dieser Stelle noch einmal verwiesen wird.

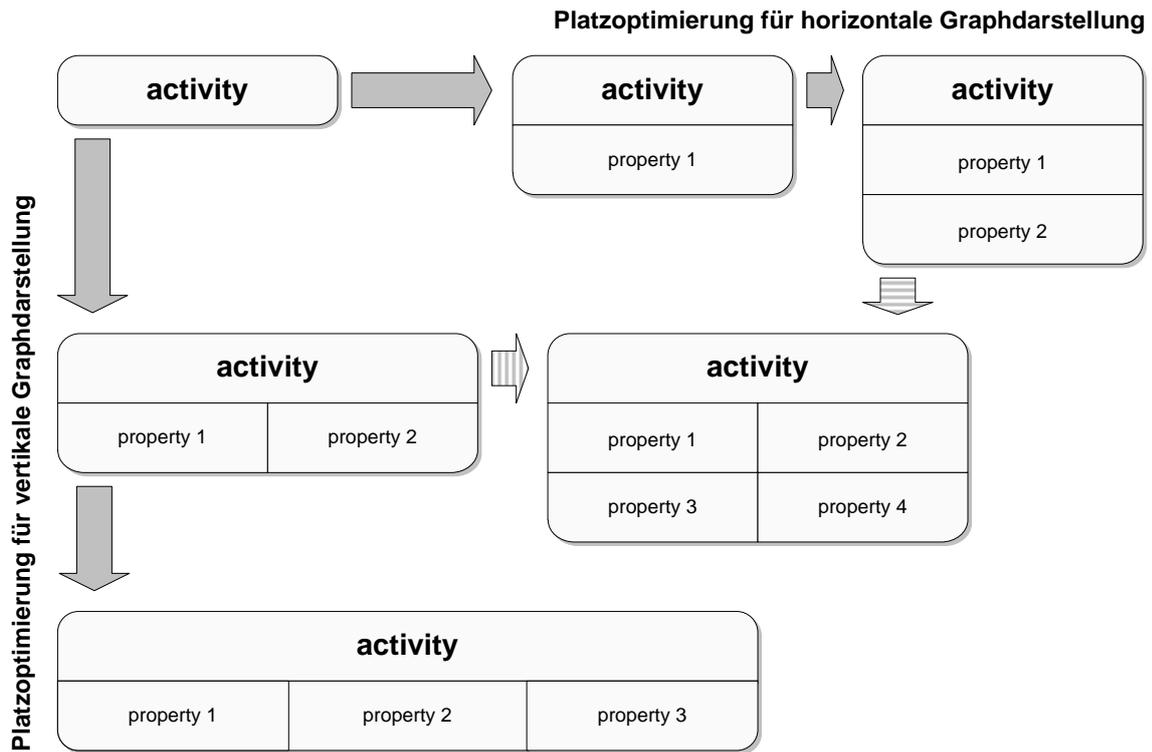


Abbildung 5.11.: Darstellung von Aktivitätenknoten mit unterschiedlich vielen Attributen

Im folgenden Kapitel werden nun alternative Darstellungsformen für Prozessinformationen vorgestellt.

6

Darstellungsformen für Prozessdaten

Die Konzepte aus dem Kapitel 5 über grafische Aspekte stehen orthogonal zu den in diesem Kapitel behandelten Darstellungskonzepten¹. Daher lassen sich all diese Konzepte für die Prozessvisualisierung kombinieren.

Welche Darstellungen sind für Prozessinformationen außer den üblichen Prozessgraphen denkbar? In Kapitel 3 wurden Benutzeranforderungen herausgearbeitet. Hier soll aber eine andere Herangehensweise gewählt werden, um herauszufinden, welche Darstellungstypen sinnvoll sind. Die Betrachtung der Tabelle 2.1 mit der Auflistung der Prozessinformationen zeigt, welche Informationstypen in Prozessinformationen stecken. Daraus ergibt sich der Ansatz, Darstellungen zu entwickeln, die jeweils einen dieser Informationsaspekte aufgreifen und diesen gut visualisieren können. Das Ergebnis ist Abbildung 6.1. Sie zeigt die verschiedenen Darstellungsformen und die Prozessinformationen, die jeweils im Fokus stehen.

In den folgenden Abschnitten werden diese Darstellungsformen vorgestellt und ihre Mächtigkeit diskutiert. D. h. es wird darauf eingegangen, welche Prozessinformationen überhaupt darstellbar sind und welche spezifischen Vorteile und Restriktionen die einzelnen Darstellungsformen haben. Weiterhin existieren für einige der Konzepte auch Variationen, die diskutiert werden.

Generell ist wichtig, dass der Benutzer beim Arbeiten mit verschiedenen Darstellungen nahtlose Übergänge beim Wechsel zwischen zwei Darstellungen vorfindet.

¹Der Begriff Darstellungskonzept bezieht sich auf den strukturellen Aspekt. Von einem Visualisierungskonzept wird erst durch das Zusammenspiel mit vielen anderen Konzepten, wie grafischen Konzepten, Bedienkonzepten oder dem View-Konzept gesprochen.

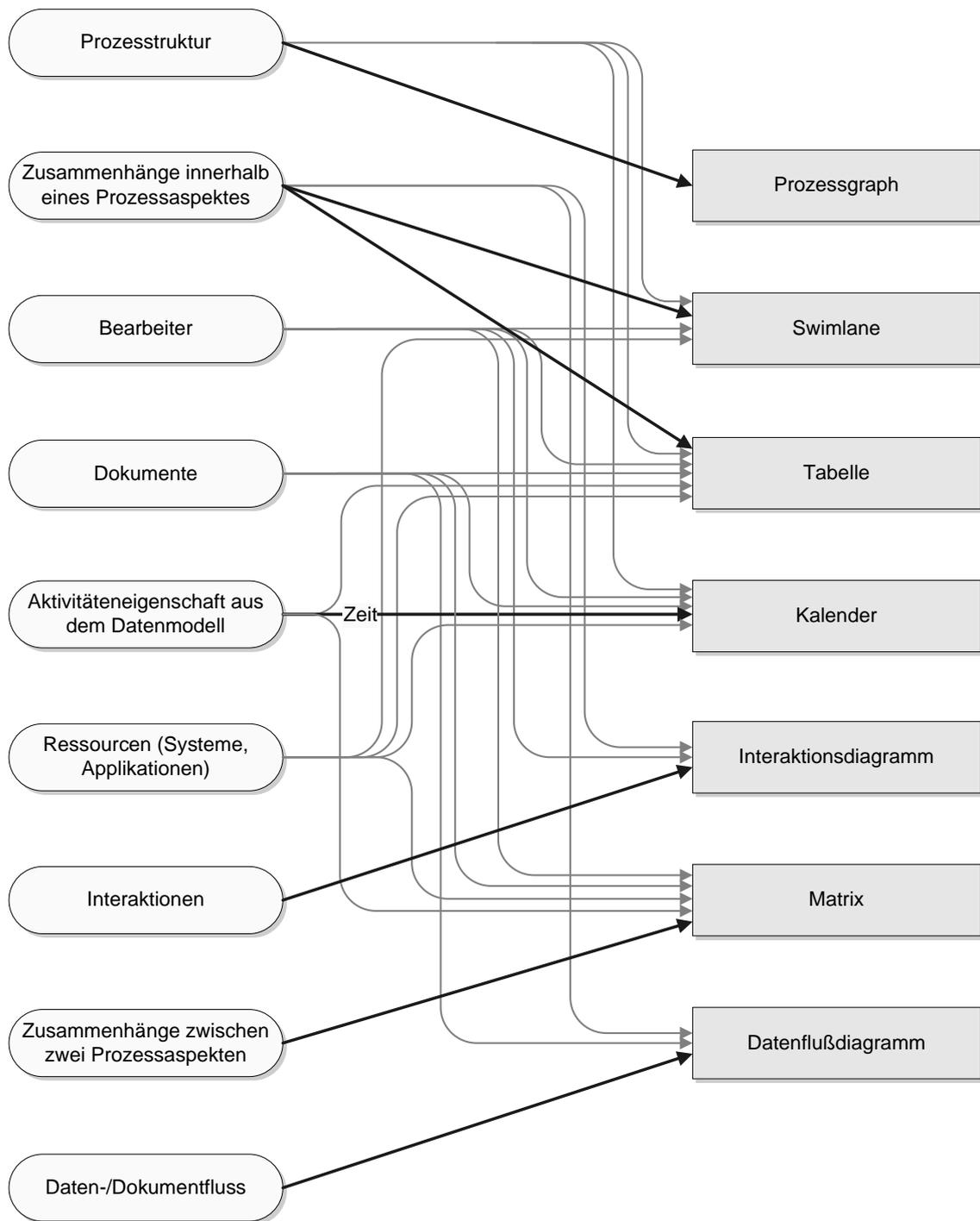


Abbildung 6.1.: Übersicht über die Darstellungsformen mit dargestellten primären Prozessaspekten

Aspekte, die beim Umschalten von einem Darstellungskonzept zum anderen möglichst erhalten bleiben sollen, sind:

➤ **Gezeigter Bildschirmausschnitt**

Der Benutzer arbeitet meist auf einem Ausschnitt des aktuell dargestellten Prozesses. Wird auf eine andere Darstellung gewechselt, sollte wieder, soweit möglich, derselbe Prozessausschnitt dargestellt werden. Dadurch lässt sich eine komplette Neuorientierung vermeiden.

➤ **Merkmale der aktuellen View**

Falls möglich sollten die Darstellungsmerkmale übernommen werden, etwa ausgeblendete Prozessaspekte (z. B. Daten & Dokumente).

➤ **Markierte Elemente** (vgl. Abschnitt 5.3.1)

Vom Benutzer markierte Elemente bleiben erhalten.

Durch einen solchen Ansatz ist der Benutzer nicht auf eine einmal gewählte Darstellungsform festgelegt.

6.1. Prozessgraphdarstellung

Die Stärke dieser Darstellungsform liegt darin, die logische Struktur eines Prozesses zu verdeutlichen. Die Darstellung als Netzplan ist die klassische Visualisierungsform für Prozesse. Sie stellt einen Prozess so dar, wie er meist vom Prozessdesigner auch entworfen wurde. Die Prozessgraphdarstellung ist die natürliche Umsetzung des Gedankenmodells, das Benutzer im allgemeinen, vor allem aber Prozessdesigner haben.

BEISPIEL

Abbildung 6.2 zeigt ein Beispiel für eine Prozessgraphdarstellung. Die drei Aktivitäten 'design', 'marketing planning' und 'production planning' laufen parallel ab, 'design' und 'mass-production development' dagegen sequenziell. Bei einer solchen Darstellung ist konfigurierbar, welche und wie viele Attribute bei der Darstellung eines Aktivitätenknotens mit angezeigt werden (siehe Abbildung 5.11). Hier sind es die Bearbeiterzuordnung und die beteiligten Dokumente. Je nachdem, mit welcher Zielsetzung der Benutzer die Graphdarstellung verwendet, gibt es dafür eine passende Konfiguration, die eben gerade die Informationen zeigt, die der Benutzer in diesem Moment benötigt.

Das Benutzerinterface muss eine Möglichkeit bieten, die Knotenkonfiguration schnell umzuschalten. Dies kann in einer seitlichen Funktionsleiste im 'Aufgabenbereich' geschehen (siehe Abschnitt 7.4).

Eine weitere Funktionsleiste könnte weitere (Detail-)Informationen zum gerade gewählten Knoten anzeigen. Wenn der Mauszeiger über einem Knoten verharret, zeigen eingeblendete Kurzinformationen (engl. *tooltips*) zusätzlich weitere Informationen an.

MÄCHTIGKEIT DES KONZEPTES

Die Graphdarstellung ist sehr mächtig. Sie ist nicht umsonst die native Darstellungsform für Prozesse. Vor allem der Kontrollfluss ist sehr gut nachvollziehbar. Verzweigungen im Ablauf mitsamt Verzweigungsbedingungen lassen sich sehr gut darstellen, auch Schleifen,

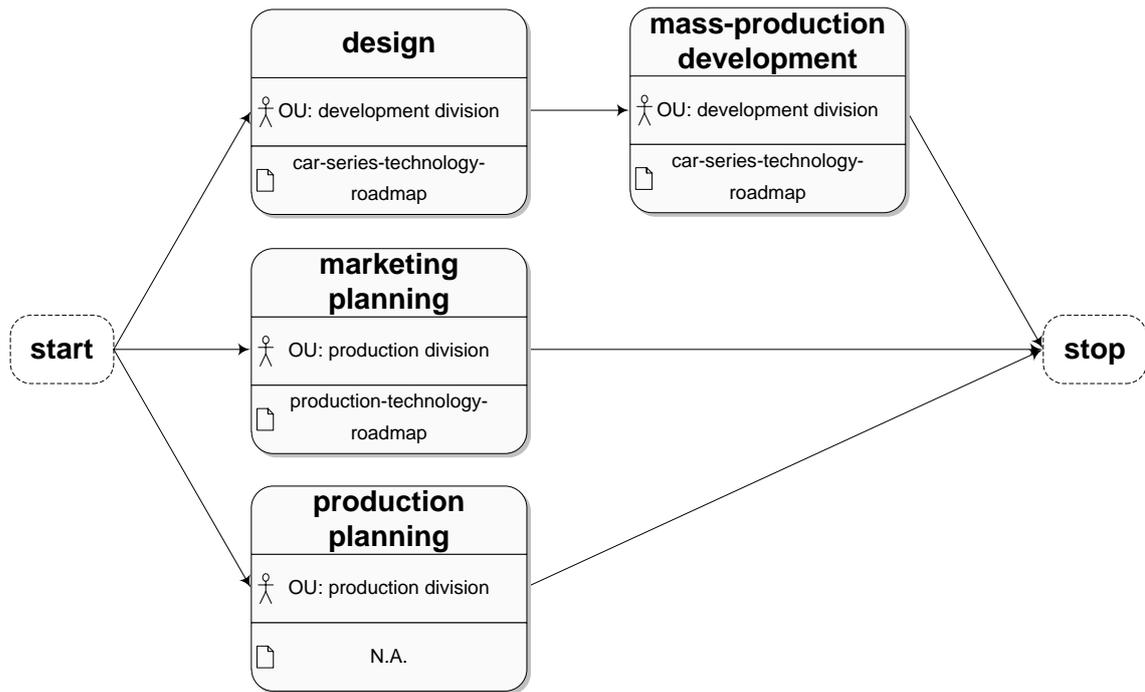


Abbildung 6.2.: Prozessgraphdarstellung

Ausnahmen (engl. *exceptions*) und sonstige Abhängigkeiten (z. B. Sync-Kanten) passen sehr natürlich in diese Darstellung. Nur wenn sich der Benutzer für Einzelheiten aus dem Prozessablauf interessiert, kann die Swimlane-Darstellung (siehe Abschnitt 6.2) Ansichten erzeugen, die an den anstehenden Arbeitsschritt speziell angepasst sind.

Alle Aktivitätenattribute und -Objekte lassen sich bei Bedarf darstellen. Die Visualisierung des Datenflusses lässt sich unabhängig von der restlichen Darstellung zuschalten.

Das Organisationsmodell spielt in diesem Konzept praktisch keine Rolle, es können zwar Informationen daraus dargestellt werden, wie z. B. die Zuordnung einer Aktivität zu einer Abteilung über die Auswertung des zugehörigen Objektes 'Ausführungseinheit', aber die eigentliche Darstellung erfolgt unabhängig vom Organisationsmodell. Es ist dennoch hilfreich, wenn die Markierungsfunktion (siehe Abschnitt 7.2) verwendet wird, um z. B. alle Aktivitäten zu markieren, die auf ein bestimmtes Dokument lesend zugreifen.

Das Konzept eignet sich sowohl für Schema- und Instanzdarstellung als auch, mit leichten Modifikationen für Multi-Instanz-Darstellungen. Sogar hierarchische Prozesse lassen sich darstellen, dies ist der Übersichtlichkeit allerdings nicht immer zuträglich. Einzelne Unterprozesse lassen sich jedoch noch gut in die Darstellung eines Prozesses integrieren. Wenn eine Aktivität andere Aktivitäten kapselt, kann der Benutzer entweder diesen gekapselten Subprozess in einer extra Darstellung visualisieren lassen oder die bisher versteckten Aktivitäten in die Darstellung integrieren lassen. Der Übersichtlichkeit halber werden solche Blöcke gesondert markiert, oder farblich hinterlegt (siehe Abbildung 6.4).

VOR- UND NACHTEILE

Ein großer Vorteil der Prozessgraphdarstellung ist die leichte Zugänglichkeit der erstellten Prozessabbildungen. Sie sind für viele Benutzergruppen intuitiv erfassbar. Und mithilfe der weiten Konfigurierbarkeit der Darstellung lassen sich viele Problemstellungen mit dieser Darstellungsform lösen.

Die Graphdarstellung hat allerdings auch Nachteile, etwa dass entsprechende Darstellungen sehr schnell einen großen Platzbedarf erfordern. Dies hat auch zu den Überlegungen für diese Arbeit geführt und der Suche nach alternativen Darstellungsformen.

Die Darstellung stellt den Kontrollfluss und die Aktivitäten in den Mittelpunkt der Betrachtungen, das führt dazu, dass einige Bereiche der Prozessinformationen nicht optimal dargestellt werden können. Der Datenfluss lässt sich nur aus einer Gesamtperspektive betrachten, es fällt schwer, den Bearbeitungsweg einzelner Dokumente zu verfolgen. Auch die Interaktionen zwischen den verschiedenen Bearbeitern stecken zwar letztlich in der Darstellung, ließen sich aber sehr viel deutlicher darstellen. Temporale Aspekte lassen sich nur als textuelles Datum visualisieren, die Art der Darstellung sagt sonst nichts über die Dauer von Aktivitäten aus. Aus der Darstellung lassen sich nur sehr schlecht Zusammenhänge zwischen verschiedenen Prozessaspekten ableiten. Es ist beispielsweise schwer herauszufinden, welche Bearbeiter mit welchen Dokumenten zu tun haben, da die Dokumente in der Darstellung primär nur mit den Aktivitäten zusammen dargestellt werden. Und schließlich lässt sich die Prozessgraphdarstellung nicht gut für den Zweck einer Multi-Instanz-Übersichtsdarstellung verwenden. Es ist keine parallele Darstellung der Instanzen möglich, nur eine aggregierte Darstellung. Diese ist aufgrund ihrer Größe für eine Übersichtsdarstellung relativ unübersichtlich.

MÖGLICHE VARIATIONEN

Varianten der Schema- und Instanzdarstellung existieren nicht, jedoch lässt sich das Aussehen der Darstellung über die Knotenkonfiguration variieren, aber die dargestellten Informationen ändern sich hierdurch nicht. Prinzipiell zeigt die Prozessgraphdarstellung immer alle Aktivitäten an. Durch geeignete Views (7.1) lässt sich dies ändern. Damit lassen sich beispielsweise Ansichten erzeugen, bei denen nur noch die Aktivitäten einzelner Mitarbeiter angezeigt werden. Bei Instanzdarstellungen lassen sich zusätzlich alle nicht gewählten Ausführungspfade ausblenden (siehe Abschnitt Konfigurationsmöglichkeiten 6.1).

Multi-Instanz-Darstellung

Eine aggregierte Multi-Instanz-Darstellung lässt sich realisieren, indem eine Status-Information über jedem Knoten eingeblendet wird. Dies kann ein segmentierter Balken sein, der in jedem Segment farbcodiert über den Aktivitätenstatus (abgeschlossen, laufend, bereit, nicht bereit, etc.) Auskunft gibt. Alternativ könnten auch mehrere gestapelte Balken jeweils die Anzahl der Aktivitäten anzeigen, die einen bestimmten Status haben. Die Detailansicht (siehe 7.3) kann dann aufschlüsseln, in welchen Instanzen eine Aktivität einen bestimmten Ausführungsstatus hat.

In den Knoten lassen sich bei dieser Darstellung nur aggregierte Informationen zu den Aktivitäten darstellen. Also beispielsweise nur Rollenangaben statt konkreter Mitarbei-

terzuordnungen und Minimum-, Maximum- und Durchschnittsangaben zu (Bearbeitungs-)Zeiten oder Kosten.

KONFIGURATIONSMÖGLICHKEITEN

Die folgende Liste zeigt wichtige Einstellungen für die Prozessgraphdarstellung. Auf die fett markierten Einträge wird anschließend kurz eingegangen.

- Darstellungsvariante wählen
- Graph-Ausrichtung horizontal oder vertikal
- **Vertikal-Perspektive an/aus**
- **Knotenkonfiguration**
- View-Konfiguration: Daten-/Dokumentfluss an/aus
- View-Konfiguration: Schleifen an/aus
- View-Konfiguration: Ausnahmen an/aus
- Darstellung der Blockstruktur²
- Zeitleiste an/aus
- **Tatsächlichen Ausführungspfad markieren**
- **Noch nicht aktivierte Aktivitäten transparent darstellen**
- **Deaktivierte Aktivitäten ausblenden**

Vertikal-Perspektive an/aus

Routenplaner in Autos und im Internet [Maps06] stellen mehr und mehr auf eine perspektivische Ansicht um, da so mehr Raum auf die Darstellungsfläche projiziert werden kann. Einen Vergleich der Graphdarstellung mit und ohne perspektivische Verzerrung zeigt Abbildung 6.3. Das erste Bild benötigt, durch, dass nach hinten kippen, weniger Höhe, somit können bei gleicher Darstellungshöhe tiefere Verzweigungen dargestellt werden. Es bietet sich dabei an, falls technisch möglich, die gelbe Dokumentationsebene nicht zu verzerren.

Knotenkonfiguration

Der Anwender kann wählen, wie die Aktivitätenknoten in dieser graphbasierten Darstellungsform aussehen sollen (siehe 5.3.1). Damit für den Benutzer vorhersehbar ist, wie sich die Darstellung ändern wird, erfolgt die Auswahl visuell anhand von kleinen Thumbnails der zur Auswahl stehenden grafischen Knotenrepräsentationen.

Tatsächlichen Ausführungspfad markieren

Bei einer Instanzdarstellung werden hiermit die gewählten Ausführungskanten fett markiert oder alternativ werden die nicht gewählten grau gefärbt und auch die nicht aktivierten Knoten werden grau gefärbt. Mithilfe der nächsten Funktion können deaktivierte Knoten komplett aus der Darstellung entfernt werden.

²siehe Abschnitt 6.1.1

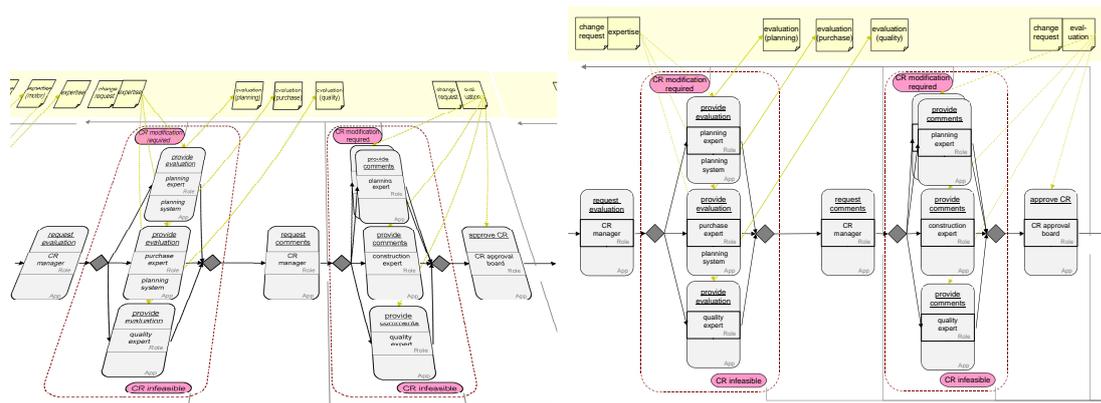


Abbildung 6.3.: Ausschnitt aus Change Management Prozess – Wirkung perspektivischer Darstellung

Noch nicht aktivierte Aktivitäten transparent darstellen

Noch nicht aktivierte Aktivitäten einer Instanz werden transparent dargestellt, um den Ausführungsfortschritt zu verdeutlichen.

Deaktivierte Aktivitäten ausblenden

Bei Instanz-Darstellungen werden hiermit alle Pfade des Prozesses ausgeblendet, die nicht durchlaufen wurden, um die Darstellung übersichtlicher werden zu lassen.

6.1.1. Blockstruktur

Ein Ansatz zur Verbesserung der Übersichtlichkeit einer Prozessgraphdarstellung besteht darin, zusammenhängende Strukturen im Graphen (z. B. Parallel-Verzweigungen oder Schleifenblöcke) besonders zu markieren. Dies hilft dem Benutzer einen schnellen Überblick über den Kontrollfluss zu erhalten.

Um dies zu ermöglichen, sollte der dargestellte Prozess ein zugrunde liegendes Prozessmodell haben, das dem Konzept der regelmäßigen Blockstrukturierung genügt (BPMN nennt dies Full-Blocked), also einen blockstrukturierten Kontrollfluss aufweisen, in dem Kontrollstrukturen wie Schleifen, Sequenzen und Verzweigungen auf Blöcke mit eindeutiger Start-/Endaktivität abgebildet werden (siehe dazu [Rein93]). Die einzelnen Blockstrukturen dürfen sich nicht überschneiden, können aber beliebig ineinander verschachtelt sein. Dies führt zu einer symmetrischen Strukturierung des Kontrollflussgraphen (engl. *symmetrical control structures*), da eine Aufspaltung an einem Knoten beginnt und es auch genau einen Knoten gibt, an dem die Zweige wieder zusammengeführt werden.

Das an der Universität Ulm entwickelte ADEPT Prozess-Metamodell [DR03] ist blockstrukturiert. Dies bringt noch weitere Vorteile mit sich, etwa bei Schemaänderungen, wenn diese im laufenden Betrieb möglich sein sollen. Dies erfordert zur Laufzeit automatische Korrektheitsanalysen, die bei symmetrischen Kontrollflüssen besonders effizient möglich werden. Aber auch die Prozessmodellierung lässt sich vereinfachen, weil syntaxgesteuerte Editoren eingesetzt werden können.

Aber selbst wenn Prozesse keinen komplett symmetrisch strukturierten Kontrollflussgraphen haben, lassen sich über eine Graphanalyse blockstrukturierte Graphenteile ermitteln, diese können dann in der Visualisierung mit farbigen Blöcken hinterlegt werden, um dem Benutzer das Erfassen des Kontrollflusses zu erleichtern. Einen Beispielprozess mit hinterlegter Blockstruktur zeigt Abbildung 6.4. Die Aktivität 'B-I' kapselt einen Unterprozess, darauf weist das Plus-Symbol im Knoten hin. Abbildung 6.5 zeigt denselben Prozess, wobei der Unterprozess nun geöffnet wurde, und damit in die Darstellung integriert wurde.

Das zweite Bild zeigt einen hellblau hervorgehobenen Block, der beispielsweise durch Anklicken oder längeres Verweilen derart eingefärbt worden ist. Dieser selektierte Block lässt sich über das nun eingblendete Minus-Symbol zusammenfallen.

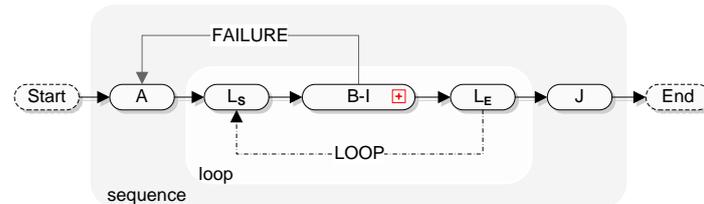


Abbildung 6.4.: Prozessgraph mit Blockstruktur

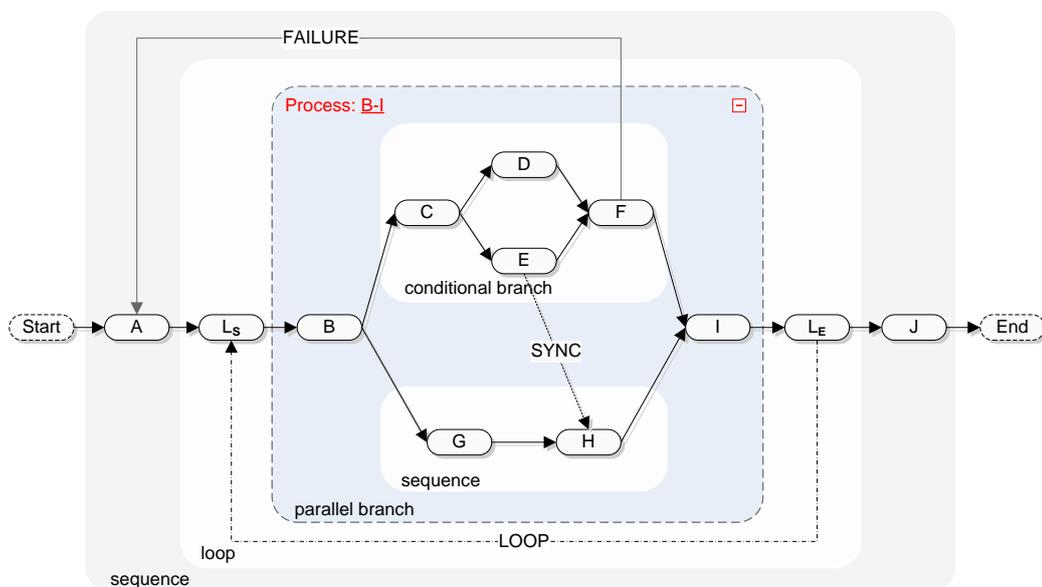


Abbildung 6.5.: Prozessgraph mit Blockstruktur und geöffnetem Unterprozess

Die Farben dieser Blöcke weisen dabei keinen hohen Kontrast auf, um von der eigentlichen Information nicht abzulenken (siehe Abschnitt 5.1.1).

6.1.2. Zeitleiste

Eine unten am Darstellungsfenster angebrachte Zeitleiste dient dazu, die Historiendarstellung zu steuern und grobe Informationen über die zeitliche Einordnung der Aktivitäten zu

liefern. Die Zeitleiste besteht aus zwei unterschiedlich gefärbten Bereichen, Vergangenheit und Zukunft. Die Skala verläuft linear, es werden nur die Startzeiten der Aktivitäten angezeigt. Wenn der Mauszeiger über einer Aktivität verharrt, kann als Zusatzinformation in der Zeitleiste ein Zeitbalken eingeblendet werden, der Auskunft über Dauer und Endzeit der Aktivität gibt. Weiterhin können Zeitpunkt der Aktivierung, Fristen und damit der Zeitspielraum für die Ausführung der Aktivität visualisiert werden.

Sobald der Schieberegler, der den Zeitpunkt der Prozessdarstellung festlegt, bewegt wird, verändern sich die Ausführungszustände entsprechend. Falls die Historie in einem Fenster angezeigt wird, zeigt dieses die Einträge bis zum gewählten Zeitpunkt an oder markiert deutlich den Bereich, ab dem die Zukunft beginnt (d. h. durch graues Einfärben der in der Zukunft liegenden Einträge).

Auch der Umgang mit On-Line-Verlaufsinformationen kann über diese Zeitleiste erfolgen (siehe dazu Abschnitt 7.6.1). Beim Eintreffen neuer Informationen können zwei Möglichkeiten für die Reaktion des Interface eingestellt werden. Entweder löst der Anwender die Aktualisierung der Darstellung manuell aus oder sie wird automatisch aktualisiert.

Die Zeitleiste stellt eine hilfreiche Ergänzung für die Prozessgraphdarstellung dar, die temporale Aspekte nicht gut darstellen kann. Das Konzept der Zeitleiste ist von der Kalenderdarstellung abgeleitet (siehe Text und Abbildungen in Abschnitt 6.3).

6.1.3. Implementierungsaspekte

Grundlegend für Graphdarstellungen sind Layout-Algorithmen [RBRB06]. Die Aufgabe dieser Algorithmen ist es, die Kanten und Knoten des Netzplans so zu platzieren, dass die Darstellung gut strukturiert erscheint und somit schnell erfassbar ist. Gutes Layouting trägt zu Usability bei. In Abschnitt 4.1 erwähnte Usability-Prinzipien sind daher auch Anforderungen an einen Layout-Algorithmus:

- Ausrichtung der Elemente
- Nähe
- Erzeugte Graphen sollen möglichst überschneidungsfrei sein

Erwähnenswert sind u. a. die folgenden Layout Algorithmen:

➤ **Orthogonales Layout** (siehe Abbildung 6.6)

Ein relativ einfacher Layout Algorithmus, der äußerst kompakte Darstellungen ermöglicht. Objekte werden rechtwinklig ohne Überschneidungen angeordnet. Jedoch fällt es schwer die Strukturierung zu erfassen.

➤ **Hierarchisches Layout** (siehe Abbildung 6.7)

Ein sehr komplexer Layout Algorithmus, er erzeugt 'stromorientierte' Graphen. Dies kommt der menschlichen Wahrnehmung entgegen und erleichtert so das Erfassen der logischen Zusammenhänge. Aktivitäten, die weiter rechts platziert sind, sind logisch abhängig von weiter links platzierten.

➤ **Inkrementelles Layout**

Dieser Algorithmus stellt einen Aufsatz auf einen der herkömmlichen Layout-Algorithmen dar, d. h. es gibt z. B. einen inkrementellen hierarchischen Layout-Algorithmus. Ausgehend von einem einmal erzeugten Layout kann das Ein-

/ Ausblenden von Graphteilen möglichst darstellungsneutral erfolgen, wenn Techniken des inkrementellen Layouts verwendet werden.

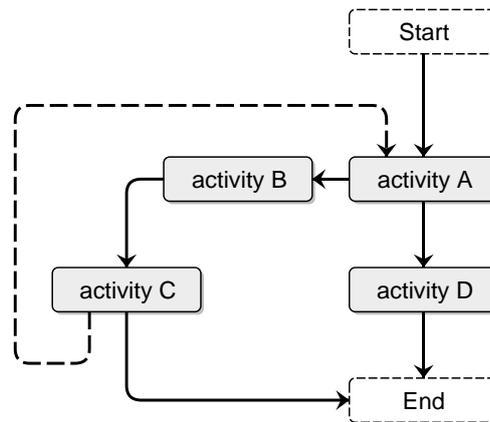


Abbildung 6.6.: Orthogonales Layout

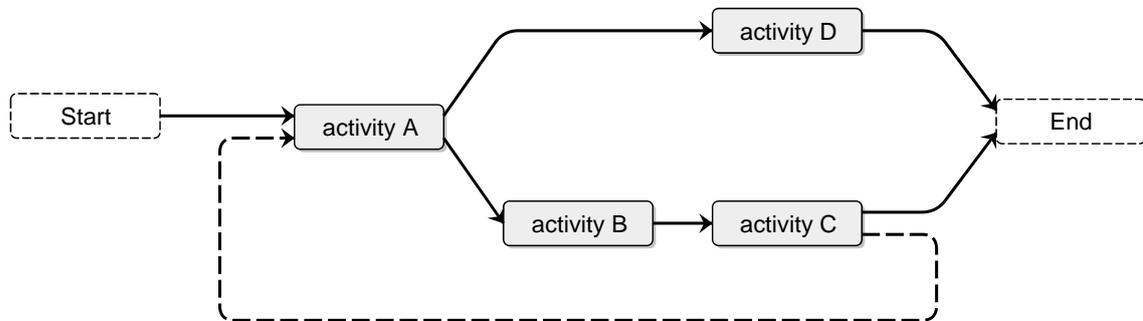


Abbildung 6.7.: Hierarchisches Layout

Die Vorteile eines 'Inkrementellen Layouters' kommen in Verbindung mit einer Animationskomponente besonders zur Geltung. Bei Änderungen am Graphen, z. B. beim Ausklappen bisher zusammengefasster Aktivitäten, behält der Benutzer leichter den Überblick, da Änderungen am Graphen vor den Augen des Benutzers geschehen und er sich nicht an eine plötzlich eingblendete, neue Darstellung gewöhnen muss.

6.2. Swimlane-Darstellung

Die Swimlane-Darstellung ist eine Ableitung aus der Prozessgraphdarstellung, bei der die Aktivitäten in Partitionen aufgeteilt dargestellt werden. Der Netzplan wird also bezüglich eines Ressourcentyps in horizontale 'Pools' (Partitionen) aufgeteilt. Wenn ein Ressourcentyp n unterschiedliche Ressourcen hat, enthält der resultierende Netzplan für jede Ressource diesen Typs einen Pool. Im einfachsten Fall, wenn es keine zwei Ressourcen desselben Typs gibt, ist die Swimlane-Darstellung 'identisch' mit der Prozessgraphdarstellung.

Die Aktivitäten des ursprünglichen Netzplanes werden den verschiedenen Pools zugeordnet. Alle Aktivitäten, die sich nicht zuordnen lassen, weil sie die entsprechende Ressource

nicht verwenden, werden einem weiteren Ersatzpool zugeordnet. Jede Darstellung enthält also insgesamt n oder $n+1$ Pools. UML bezeichnet diese Pools als Partitionen.

BEISPIEL

Abbildung 6.8 zeigt ein einfaches Beispiel einer Swimlane-Darstellung. Der Benutzer kann, wie bei der Prozessgraphdarstellung, frei konfigurieren, welche Aktivitätendetails in der Darstellung mit angezeigt werden sollen. Nähere Details kann eine Tooltipp-Information oder eine Aktivitätendetailansicht im Aufgabenbereich anzeigen (siehe Abschnitt 7.4).

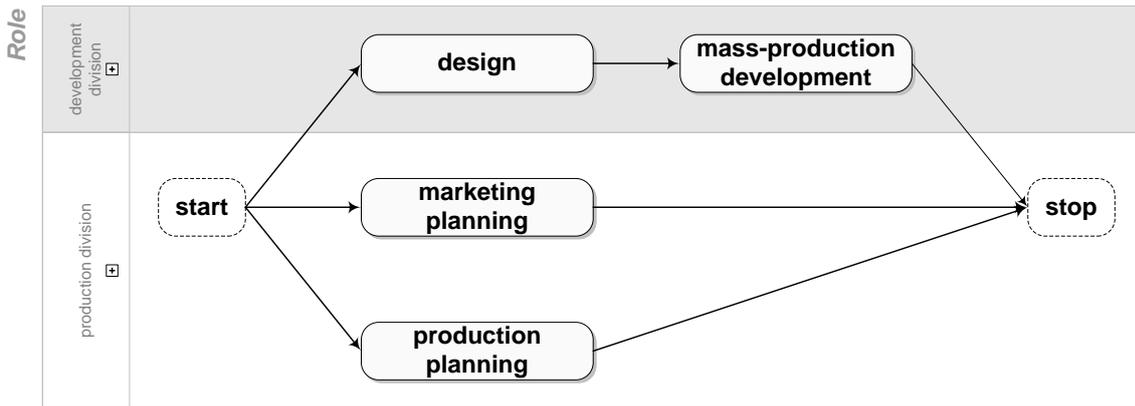


Abbildung 6.8.: Prozessgraph in Swimlane-Darstellung

Besitzt ein Ressourcentyp einen Obertyp, ein Beispiel wäre die Ressource 'Mitarbeiter' mit Obertyp 'Abteilung', kann der Netzplan in Pools nach Abteilungen kategorisiert oder gruppiert werden. Jeder Mitarbeiter muss dabei genau einer Abteilung zugeordnet werden können. Jeder einzelne Abteilungs-Pool ist dann weiter in (Swim-)Lanes unterteilt, für jeden Mitarbeiter eine. Dies erzeugt dann eine geschachtelte Darstellung. Ein Beispiel zeigt Abbildung 6.9.

MÄCHTIGKEIT DES KONZEPTES

Da die Swimlane-Darstellung eine Erweiterung der Prozessgraphdarstellung ist, treffen alle Aussagen über deren Mächtigkeit hier genauso zu. Jedoch ist es nun möglich, Zusammenhänge zwischen einzelnen Prozessressourcen und Aktivitäten besser darzustellen. Auch das Interaktionsdiagramm und die Matrixdarstellung dienen dazu, Zusammenhänge in den Prozessinformationen herauszuarbeiten, hierbei handelt es sich aber um sehr spezialisierte Darstellungsformen für spezielle Aufgaben.

In der Swimlane-Darstellung kann der Kontrollfluss nicht mehr so übersichtlich dargestellt werden und die Darstellungskomplexität ist daher konzeptbedingt gegenüber der Prozessgraphdarstellung erhöht. Deshalb wird hier auf die Darstellung der Blockstruktur verzichtet (vgl. Abschnitt 6.1.1), es kann ohnehin nicht garantiert werden, dass parallele Aktivitäten lokal beieinander platziert sind. Schleifen, Sprünge, Ausnahmen und Sync-Kanten lassen sich wie bei der Prozessgraphdarstellung anzeigen, jedoch erschwert die parallele Darstellung des Prozesses das Erfassen.

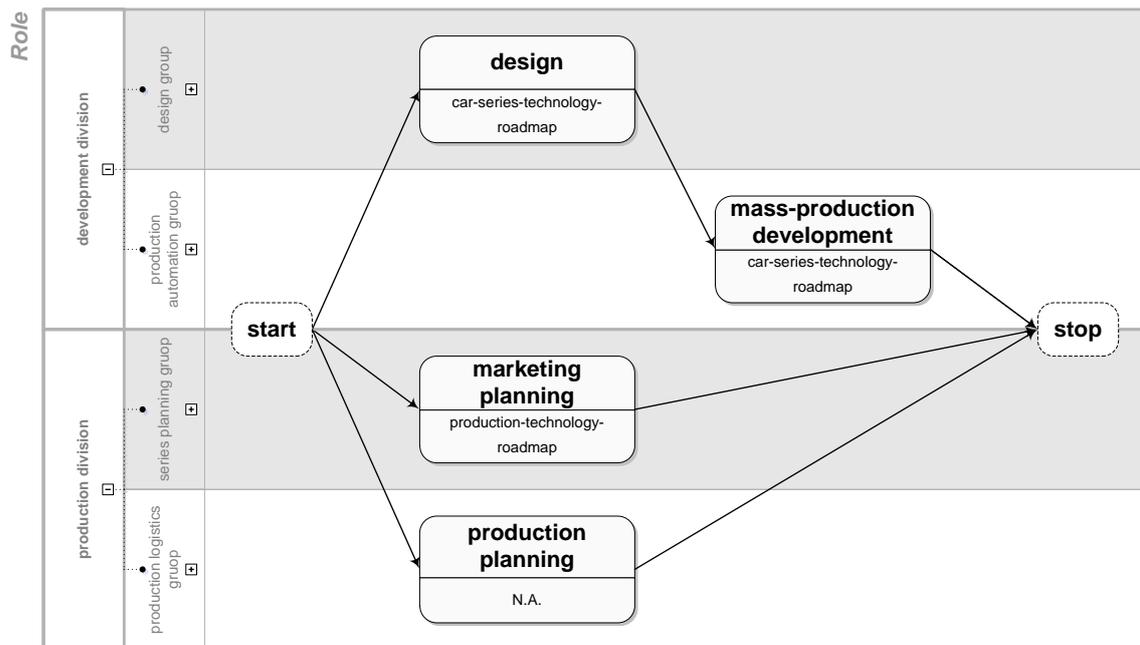


Abbildung 6.9.: Prozessgraph in geschachtelter Swimlane-Darstellung

Das Organisationsmodell spielt in diesem Darstellungskonzept eine wesentlich größere Rolle als bei der Prozessgraphdarstellung. Die Unterteilung ist zwar, unabhängig vom Organisationsmodell, nach den Prozessressourcen möglich. Aber oft verhilft erst das Organisationsmodell zur gewünschten Übersichtlichkeit, indem Ressourcen zusammengefasst werden. Beispielsweise können die Mitarbeiter statt in vielen getrennten Lanes, innerhalb von Abteilungs-Lanes dargestellt werden.

Die Swimlane-Darstellung ist ein mächtiges Werkzeug für die Bildung von Views, anhand der Informationen aus dem Organisationsmodell. Durch geeignetes Zusammenfassen kann es auch seine Stärken zum Untersuchen von Kommunikationsstrukturen, zwischen verschiedenen Organisationseinheiten innerhalb eines Prozesses, ausspielen.

Wie bei der Prozessgraphdarstellung lassen sich Schema-, Instanz- und Multi-Instanz-Darstellungen erzeugen. Auch hierarchische Prozesse lassen sich darstellen, indem Aktivitäten aus Unterprozessen in die Darstellung mit eingebettet werden. In der resultierenden Darstellung kann dann allerdings nicht mehr so leicht zwischen Aktivitäten des Eltern- und des Kindprozesses unterschieden werden. Mit den Möglichkeiten der Markierungsfunktion (siehe 7.2.1), lässt sich aber beispielsweise die Prozessherkunft farblich kodieren.

VOR- UND NACHTEILE

Prinzipiell weist die Swimlane-Darstellung ähnliche Vor- und Nachteile wie die Prozessgraphdarstellung auf. Bei Swimlane-Darstellungen werden jedoch Zusammenhänge zwischen einer Ressource des Prozesses und den Aktivitäten deutlich besser sichtbar. Dieser Vorteil wird mit einer geringeren Übersichtlichkeit des Kontrollflusses 'erkauft'. Vor allem Verzweigungen im Kontrollfluss können nicht mehr so übersichtlich dargestellt werden. Dies fällt jedoch nicht sehr ins Gewicht, da die Prozessgraphdarstellung weiterhin als

Darstellungsform zur Verfügung steht. Je nach Prozess und gewählter Darstellungsvariante (siehe nächster Abschnitt) können einige Prozesse aber sehr viel kompakter als bei der Prozessgraphdarstellung dargestellt werden (siehe Abbildung 6.10), wenn der Nutzer nicht darauf besteht, dass die ein- und ausgehenden Kontrollflusskanten mittig in den Knoten ein- bzw. austreten.

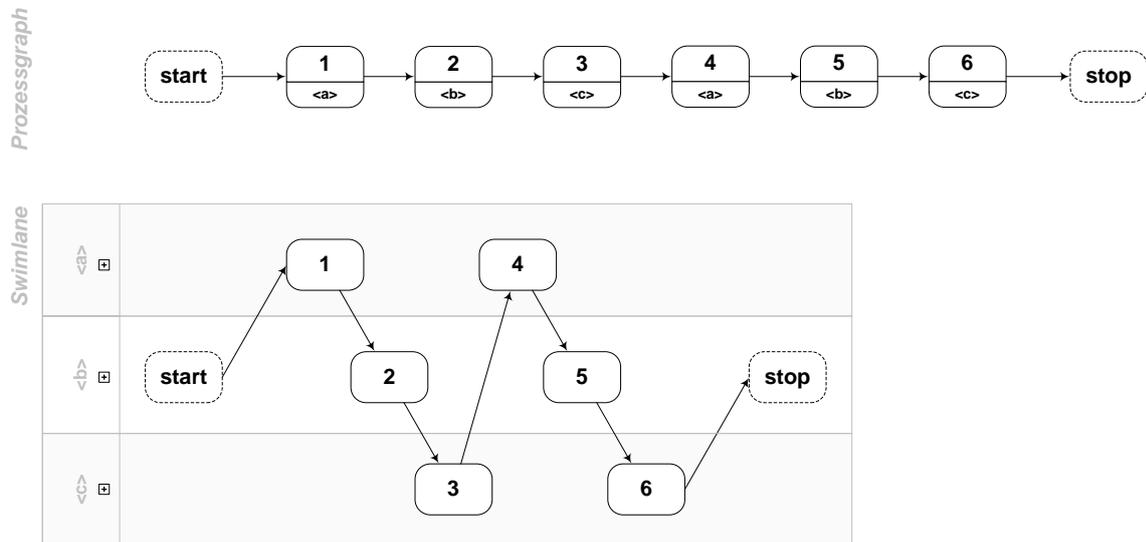


Abbildung 6.10.: Beispiel einer horizontalen Platzersparnis bei der Swimlane- gegenüber der Prozessgraphdarstellung

Weitere algorithmische Probleme, das Layout betreffend, sind im Abschnitt Implementierungsaspekte 6.2.1 beschrieben.

MÖGLICHE VARIATIONEN

Die folgende Tabelle listet die Prozessressourcen auf, mit denen der ursprüngliche Netzplan in Pools unterteilt werden kann.

Ressource	Beschreibung
Ausführungseinheit	Die gebräuchlichste Nutzung von Swimlane-Darstellungen sammelt jeweils alle Aktivitäten einer Rolle, eines Bearbeiters oder einer Organisationseinheit in einem Pool
System	Alle Aktivitäten, die auf dem gleichen System laufen oder mit der gleichen Applikation arbeiten, sind in einem Pool angeordnet
Dokument	Alle Aktivitäten, denen dasselbe Dokument zugeordnet ist, sind in einem Pool angeordnet

Tabelle 6.1.: Varianten der Swimlane-Darstellung

Weitere spezielle Varianten von Swimlane-Darstellungen sind je nach Anwendungsfall denkbar. Beispielsweise eine Partitionierung der Aktivitäten in reine Kostenstellen und in wertschöpfende Aktivitäten.

Die Voraussetzung für Swimlane-Darstellungen ist die eindeutige Partitionierbarkeit der Prozessinformationen (siehe 8.2).

Schema- & Instanz-Darstellung

Variante 1: Swimlane nach Ausführungseinheiten

In dieser Darstellung lassen sich im Besonderen die Interaktionen der verschiedenen Ausführungseinheiten miteinander visualisieren. Durch die Informationen im Organisationsmodell entsteht eine geschachtelte Darstellung, die der Benutzer bei Bedarf auseinander falten kann.

Variante 2: Swimlane nach Systemen

Diese Variante kann dazu dienen, zu veranschaulichen, welche Prozessteile auf welchen Systemen oder Applikationen abgewickelt werden. Voraussetzung für eine eindeutige Darstellung ist, dass jeder Aktivität maximal ein System oder eine Applikation zugeordnet ist.

Variante 3: Swimlane nach Dokumenten

Aus dieser Darstellung ist sehr gut ersichtlich, welche Aktivitäten auf ein Dokument zugreifen und in welcher Reihenfolge, die am Prozess beteiligten Dokumente bearbeitet werden. Voraussetzung für eine eindeutige Darstellung ist, dass jeder Aktivität maximal ein Dokument oder ein Datenbankeintrag zugeordnet ist.

Multi-Instanz-Darstellung

Auch bei der Swimlane-Darstellung ist eine Multi-Instanz-Darstellung möglich, umgesetzt wird dies analog zur Prozessgraphdarstellung (siehe 6.1). Das Ziel einer Multi-Instanz-Darstellung ist, den Fortschritt der einzelnen Instanzen einander gegenüberzustellen. Das Ziel der Swimlane-Darstellung ist es jedoch Zusammenhänge aufzuzeigen. Die Visualisierung des Multi-Instanz-Falles kombiniert diese beiden Ziele. Notwendige Voraussetzung für die Darstellung als Swimlane ist lediglich, dass sich alle Instanzen gleichermaßen in die gleichen Lanes einordnen lassen.

Variante 1: Swimlane nach Ausführungseinheiten

Der Vorteil dieser Darstellung ist, dass die Prozess-Fortschritte der verschiedenen Organisationseinheiten miteinander verglichen werden können. Auf diese Weise können Engpässe erkannt werden.

Bei dieser Darstellung werden in den Lanes nicht die Mitarbeiter angegeben, sondern ausschließlich Rollen- oder Funktionsangaben, da diese nicht von Instanz zu Instanz abweichen.

Variante 2: Swimlane nach Systemen

Diese Variante dürfte vor allem bei der Fertigung interessant sein, um beispielsweise zu untersuchen, ob alle Systeme gleichmäßig ausgelastet werden.

Variante 3: Swimlane nach Dokumenten

Wo der Fortschritt bei der Bearbeitung von Dokumenten von Interesse ist, kann diese Darstellung sinnvoll sein. Voraussetzung hierfür ist, dass jede Aktivität genau einem Dokument zuordenbar ist.

KONFIGURATIONSMÖGLICHKEITEN

- Darstellungsvariante wählen
- Graph-Ausrichtung horizontal oder vertikal
- Vertikal-Perspektive an/aus
- Knotenkonfiguration
- Graph neu zeichnen
- Historiendarstellung (mit Timeline)
- Übersichtstreifen einblenden
- Deaktivierte Aktivitäten ausblenden
- Lanes zusammenfassen

Vertikal-Perspektive an/aus

Diese Konfigurationsmöglichkeit wurde schon bei der Prozessgraphdarstellung angesprochen (siehe 6.1). Da Swimlane-Darstellungen sehr viel Höhe benötigen, kann die Vertikal-Perspektive hier besonders zu einer besseren Übersichtlichkeit beitragen.

Knotenkonfiguration

Der Anwender kann hier wie bei der Prozessgraphdarstellung (siehe Abschnitt 6.1) die grafische Knotenrepräsentation wählen.

Graph neu zeichnen

Wie im nächsten Abschnitt (6.2.1) beschrieben, wird eine Implementierung Näherungsalgorithmen zur Layout-Gestaltung verwendet. Nach Ablauf einer kurzen Zeitspanne (ca. fünf Sekunden) wird die bisher beste Lösung Grundlage des Layouts. Dadurch kann die Reaktionszeit des Interface für den Benutzer erträglich gehalten werden. Die Berechnung von besseren Layout-Lösungen kann aber im Hintergrund weiterlaufen. Somit bekommt der Benutzer die Möglichkeit, bei einer nicht zufriedenstellenden Darstellung, auf ein verbessertes Layout zu wechseln.

Übersichtstreifen einblenden

Bei Visualisierungen mit vielen Lanes verteilen sich die Aktivitäten des dargestellten Prozesses sehr stark. Das führt dazu, dass innerhalb einer Lane große Abstände zwischen den Aktivitäten auftreten können. Zu einer besseren Übersicht für den Benutzer verhilft hier die gleichzeitige Darstellung eines Übersichtsfensters (siehe Abschnitt 7.5). Eine Alternative dazu bietet die Implementierung eines schmalen Übersichtstreifens unterhalb jedes Pools oder jeder Lane. Dieser Streifen zeigt miniaturisiert auf der verfügbaren Breite alle Aktivitätenknoten und markiert den derzeit dargestellten Bereich.

Deaktivierte Aktivitäten ausblenden

Bei Instanzdarstellungen werden alle Pfade des Prozesses ausgeblendet, die nicht durchlaufen wurden, um die Darstellung übersichtlicher werden zu lassen.

Lanes zusammenfassen

Um die Übersichtlichkeit durch eine reduzierte Anzahl von Swimlanes zu erhöhen, kann der Benutzer gezielt manuell einzelne Lanes zu einer zusammenfassen. Alle Aktivitäten der gewählten Lanes werden danach in nur einer Lane dargestellt. Dies dürfte vor allem bei Prozessen mit vielen Lanes sinnvoll sein, wo aber nur wenige Aktivitäten den einzelnen Lanes zugeordnet sind. Auch können Lanes die im Moment nicht von größerem Interesse sind, auf diese Art und Weise zusammengefasst werden.

6.2.1. Implementierungsaspekte

Eine gute übersichtliche Swimlane-Darstellung zu erzeugen ist keine triviale Aufgabe. Dies ist somit auch eines der größten Probleme dieses Darstellungskonzeptes, falls die benötigte Rechenzeit für das Layouting sich für die Anwender störend bemerkbar macht.

Für eine Implementierung ist wichtig, dass für diese Darstellung, wie auch schon für die Prozessgraphdarstellung, prinzipiell dieselben Layout-Algorithmen zum Einsatz kommen. Allerdings wird die Komplexität des Layout-Problems noch dadurch erhöht, dass aufgrund der zusätzlichen Nebenbedingungen bestimmte Aktivitäten jeweils im gleichen Pool positioniert werden müssen. Dazu kommt ein Optimierungsproblem, denn es existiert mindestens eine Anordnung der Pools, für die die Summe über alle horizontalen Weganteile der Kontrollflusskanten minimal ist. Diese Anordnung sollte in der Darstellung angestrebt werden, da dann die Übersichtlichkeit maximal ist, weil die Kontrollflusskanten am wenigsten von der restlichen Darstellung ablenken. Außerdem ergibt sich der Zusatznutzen, dass Pools, die viel miteinander zu tun haben, räumlich nah beieinander liegen.

Bei einer Poolanzahl von n liegt allerdings die untere Schranke der Komplexität zur Bestimmung dieser optimalen Anordnung bei $n!$, da es entsprechend viele Möglichkeiten gibt, die Pools anzuordnen. Das heißt, es müssten $n!$ Layouts berechnet werden, um das beste Layout zu finden. Da aber schon das Berechnen eines einzigen Layouts viel Rechenzeit beansprucht, erscheint dies nicht als gangbarer Weg. Für das Layout wird keine exakte Lösung benötigt, als Ziel genügt eine Näherungslösung. Die entscheidende Vorgabe bleibt, die Reaktionszeit des Interface gering zu halten. Es gilt also, heuristische Verfahren, genetische Algorithmen, Ameisenalgorithmen oder ähnliche Näherungsverfahren zur Lösung des Problems heranzuziehen. Zum Zeitpunkt der Drucklegung existierte keine, dem Autor bekannte, Literatur zu diesem Swimlane Layout Problem. Alle existierenden Algorithmen berechnen Layouts auf der Grundlage einer vorher festgelegten Pool-Reihenfolge.

Solange keine Algorithmen zur Lösung dieses Problems existieren, dürfte es ein gangbarer Weg sein, die Pools mit den meisten Knoten in die Mitte der Darstellung zu sortieren, um die Wegstrecken statistisch zu reduzieren.

6.3. Kalenderdarstellung

Die Stärke dieser Darstellungsform liegt darin, zeitliche Projektierungsaspekte zu visualisieren. Die Kalenderdarstellung, auch Balkenplan oder Gantt-Diagramm genannt, entsteht, wenn ein Prozessgraph auf die Zeitachse projiziert wird. Ein einfaches Diagramm kann als Variante eines Netzplanes gesehen werden, bei dem die Aktivitäten auf der X-Achse nach

MÄCHTIGKEIT DES KONZEPTES

Die Kalenderdarstellung ist auch die übliche Darstellungsform von Projektplanungssoftware (siehe Abschnitt 4.2.2), da hier der Zeitaspekt am besten berücksichtigt ist. Die Darstellung kann sowohl Instanzen als auch Schemata visualisieren, bei einem Schema allerdings nur die Planungszeiten. Bei Instanzen können mithilfe der tatsächlichen Zeiten auch Soll-Ist Vergleiche angestellt werden. Weiterhin können der kritische Pfad und Pufferzeiten visualisiert werden.

Der Kontrollfluss lässt sich bei der Prozessgraphdarstellung deutlich besser darstellen. In Gantt-Diagrammen werden üblicherweise nur einfache Kontrollflüsse mit parallelen Verzweigungen, aber ohne bedingte Verzweigungen, Schleifen und Ausnahmen visualisiert. Es lassen sich zwar Lösungen für die Darstellung dieser Kontrollflussaspekte finden, jedoch gewinnen die resultierenden Prozessansichten dadurch nicht an Übersichtlichkeit. Um bedingte Verzweigungen darstellen zu können, wird ein zweiter Kantentyp benötigt. Schleifen zeigen bis zur ersten Ausführung auf eine früher eingeplante Aktivität. Sobald sie aktiviert wird, werden die in ihr enthaltenen Aktivitäten wiederholt ausgeführt und hintereinander in den Projektplan eingefügt, erst danach folgt der restliche Prozessablauf. Daher ist in solchen Fällen der ursprüngliche Kontrollfluss nicht mehr gut sichtbar. Ausnahmebehandlungen werden im Projekt mit eingeplant, Aktivitäten die Ausnahmen auslösen, erhalten eine besondere Markierung. Der Datenfluss bleibt bei Kalenderdarstellungen ganz außen vor.

Das Organisationsmodell ist bei der Ressourcen-orientierten Variante dieser Darstellung sehr hilfreich. Hierarchische Prozesse können dargestellt werden, wobei die Aktivitäten von Eltern- und Kindprozess gut unterscheidbar bleiben. Eine besondere Stärke sind sehr übersichtliche Multi-Instanz und Multi-Schema-Darstellungen.

VOR- UND NACHTEILE

Die Übersichtlichkeit der Darstellung des Kontrollflusses ist eingeschränkt. Vor allem zeitliche Abhängigkeiten stellt ein solches Diagramm aber gut dar. Außer Zeitplanungen lassen sich auch die Personalplanung und die Auslastung von Maschinen/Systemen übersichtlich visualisieren.

MÖGLICHE VARIATIONEN

Schema- & Instanz-Darstellung

Variante 1: Sortiert nach Startzeit / Aktivitäten-orientiert

Das ist die übliche Variante einer Kalenderdarstellung. Sie wirkt übersichtlich, da die einzelnen Aktivitätenbalken von links nach rechts geordnet erscheinen. Wenn eine noch laufende Instanz dargestellt wird und einzelne Startzeiten noch nicht verfügbar sind, wird stattdessen topologisch sortiert. Auf der linken Seite kann eine Tabelle dargestellt werden, die Informationen zu jeder einzelnen Zeile – also zu den einzelnen Aktivitäten – zeigt. Die erste Spalte zeigt dabei den Namen der jeweiligen Aktivität, ansonsten können wie bei der Tabellendarstellung (siehe Abschnitt 6.4) weitere Aktivitäteneigenschaften eingeblendet werden. Wenn man auf diese Tabelle verzichtet, gewinnt man mehr Raum für

die eigentliche Darstellung, es muss dann allerdings jeder Balken mit dem Aktivitätensnamen beschriftet werden. Bei Anzeige der Tabelle kann diese Beschriftung beliebige andere Informationen anzeigen, wie etwa den Namen des Bearbeiters.

Bei hierarchischen Prozessen wird unterhalb einer Aktivität, die einen Unterprozess kapselt, dieser eingefügt dargestellt. Nach der letzten Aktivität des Unterprozesses folgt die nächste Aktivität des Elternprozesses.

Variante 2: Ressourcen-orientiert

Auf der y-Achse werden (alle) Ressourcen eines Typs aufgetragen. Alle Aktivitäten, die sich eine Ressource teilen, bilden dabei eine Gruppe. Die Gruppe besteht dann aus so vielen Zeilen, wie es Aktivitäten gibt, die diese Ressource nutzen. Es zeigt, welche Aktivitäten sich eine Ressource teilen. Auch die Auslastung oder Überlastung einer Ressource kann so visualisiert werden. Einen Auszug aus den wählbaren Ressourcen bietet Tabelle 2.1. Die einzelnen Gruppen lassen sich zu einer Zeile zusammenfassen, sodass beispielsweise alle Aktivitäten eines Mitarbeiters gesammelt in einer Zeile erscheinen.

Die Voraussetzung für eine solche Darstellung ist die eindeutige Partitionierbarkeit der Prozessinformationen (siehe 8.2).

Multi-Instanz-Darstellung

Eine aggregierte Darstellung ist hier wegen der differierenden Ausführungszeiten der Instanzen nicht möglich. Auch eine parallele Darstellung würde nur in einer sehr unübersichtlichen Darstellung resultieren. Mehrere Instanzen lassen sich nur darstellen, indem Aggregation und parallele Darstellung kombiniert werden. Jede Instanz wird auf eine Kalenderzeile reduziert. Die Instanzen werden untereinander dargestellt. Für die nötige Aggregation gibt es mehrere Möglichkeiten. Entweder existiert eine spezielle View (siehe Abschnitt 7.1) oder die Prozessinformationen enthalten eine Phasen- oder Meilenstein-Einteilung.

Variante 1: Ressourcen-orientiert

Analog zur Ressourcen-orientierten Variante der Schema- & Instanz Darstellung, lässt sich dieses Prinzip auf alle laufenden Instanzen eines PMS ausweiten. Der Anwender wählt beispielsweise eine Bearbeiteransicht und zur Eingrenzung eine Abteilung. Die resultierende Kalenderdarstellung zeigt dann die Auslastung jedes Mitarbeiters (alle Aktivitäten).

Variante 2: Mehrere Instanzen

Abbildung 6.12 zeigt eine solche Darstellung. Einige Change Management Prozesse werden hier reduziert auf die Bestandteile Evaluationsphase, Realisierungsphase und Abschlussphase dargestellt. Das Ergebnis ist eine sehr übersichtliche informative Übersichtsdarstellung. Die Farbgebung, Balken im Inneren der Phasen oder Symbole wie das Ausrufezeichen in der Abbildung weisen beispielsweise auf Verzögerungen beim Ablauf hin.

Variante 3: Mehrere Instanzen aus verschiedenen Schemata

Auf die besondere Problematik von Multi-Schema-Darstellungen weist Abschnitt 2.1.2 hin. Analog zu Variante 1 lässt sich die Visualisierung von Instanzen verschiedener Schemata realisieren. Alle Instanzen eines Schemas werden dabei wie in Variante 1 dargestellt. Es ist aber auch möglich, alle Schemata auf ein Grundschema zurückzuführen. Damit können völlig unterschiedliche Prozesse sehr übersichtlich miteinander verglichen werden.

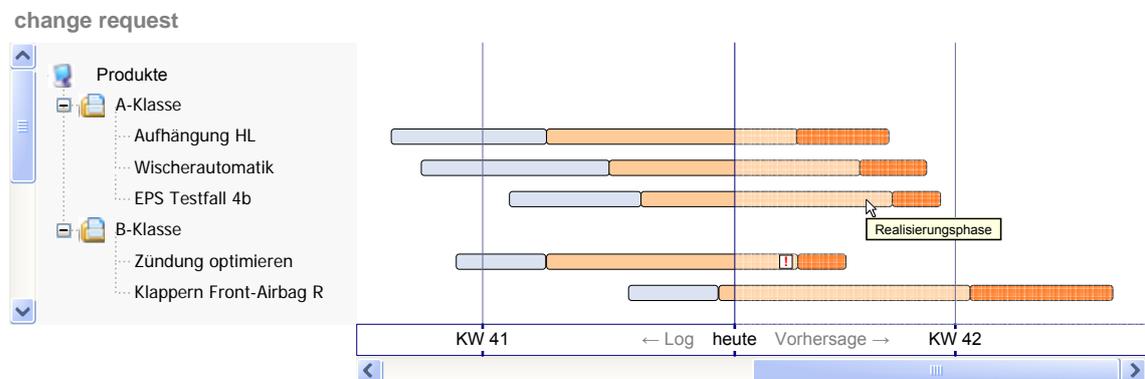


Abbildung 6.12.: Kalenderdarstellung in Multi-Instanz Variante

KONFIGURATIONSMÖGLICHKEITEN

Konfigurationsmöglichkeiten sind:

- Darstellungsvariante wählen
- Zeiten einblenden
- Balkenkonfiguration
- Ausnahmen einblenden
- Schleifen einblenden
- Deaktivierte Aktivitäten ausblenden
- Aktivitäten aller alternativer Ausführungspfade markieren

Zeiten einblenden

Da Zeiten wie die Fristen nicht immer von Interesse sind, lassen sie sich ein- und ausblenden. Eingebildet erscheinen sie beispielsweise als rotes Kreuz. Welche Zeiten angezeigt werden können, zeigt Abbildung 6.13.

Balkenkonfiguration

Hier kann das Aussehen der Aktivitätenbalken geändert werden. Dies geschieht analog zur Prozessgraphdarstellung (→ 6.1). Beispiele hierfür zeigt Abbildung 6.14. Hier sind zu einer Variante jeweils drei Fälle abgebildet:

- Normale Ausführung der Aktivität (grüner Balken).
- Die Aktivität braucht schon länger als die geplante Endzeit, aber die gesetzte Frist wurde noch nicht überschritten (gelber Balken).

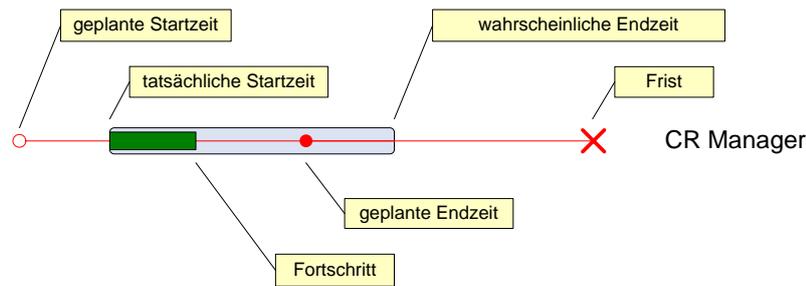


Abbildung 6.13.: Planungszeiten in der Kalenderdarstellung

- Die Aktivität konnte nicht innerhalb der gesetzten Frist abgeschlossen werden (roter Balken).

Der linke Balkentyp ermöglicht sehr übersichtliche Darstellungen, wenn die Markierungen für geplante Start- & Endzeiten und Fristen ausgeblendet werden. Bei Verwendung des rechten Balkentyps zeigen sich die verfügbaren Pufferzeiten sehr deutlich. Wenn die Aktivitäten bald nach der geplanten Startzeit beginnen, zeigen sich viele grüne Zonen, sonst Rote.

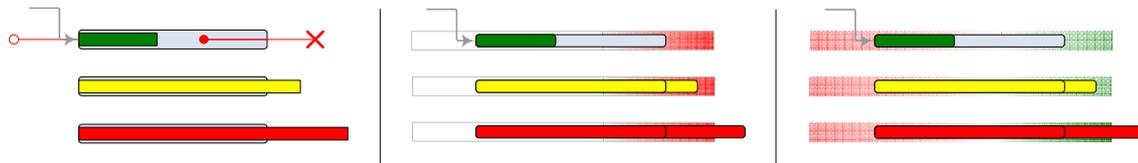


Abbildung 6.14.: Mögliche Balkenkonfigurationen in der Kalenderdarstellung

Im Inneren der Aktivitäten-Balken kann entweder ein prozentualer Fortschrittsbalken angezeigt werden oder ein Zeitbalken, der die tatsächlich seit Start der Aktivität vergangene Zeit visualisiert. Sobald eine Aktivität länger als die geplante Dauer braucht, verlängert sich der Balken entsprechend. Prozentuale Fortschrittsbalken sind vor allem dann sinnvoll, wenn die dargestellten Aktivitäten Unterprozesse haben, dann zeigen die Balken deren Fortschritt an.

Auf der rechten Seite der Balken lassen sich auch Texte (Aktivitätenattribute) einblenden, ohne die Lesbarkeit der Darstellung signifikant zu verschlechtern.

Deaktivierte Aktivitäten ausblenden

Bei Instanz-Darstellungen werden alle nicht durchlaufenen Pfade des Prozesses ausgeblendet, um die Darstellung übersichtlich zu gestalten.

Aktivitäten aller alternativen Ausführungspfade markieren

Für die Zeit und Ressourcenplanung ist es wichtig, Aktivitäten, die in jedem Fall ausgeführt werden, von Aktivitäten zu unterscheiden, die nicht in jedem Fall ausgeführt werden, d. h. Aktivitäten, an deren Ausführung Bedingungen geknüpft sind. Mithilfe dieser Funktion können alle Aktivitäten markiert werden, die sich in jeweils alternativen Ausführungspfaden befinden. Dies unterstützt die im Abschnitt 7.2.1 über Markierungsregeln

angesprochene Funktion, die bei Selektion einer einzelnen Aktivität alle Aktivitäten markiert, die sich in alternativen Ausführungspfaden befinden.

6.3.1. Dynamische Zeitachse

Die Unterteilung der Zeitachse entscheidet maßgeblich darüber, wie gut eine Kalendardarstellung nutzbar ist. Microsoft Project bietet drei Ansichten, eine Tages-, eine Wochen- und eine Monatsansicht. Auf den ersten Blick scheint eine solche Einteilung zu genügen. Auf diese Weise werden zwar Übersichten über verschiedene Zeiträume funktional ermöglicht, den Anwendern genügt diese Einteilung jedoch nicht. Wichtiges Nutzerziel ist, sich eine Übersicht über Projekte und Teilprojekte zu verschaffen. Das bedeutet, dass die Zeiteinteilung der Darstellung sich nach der jeweiligen (Teil-)Projektdauer richten soll. Wenn der Nutzer die Übersicht über ein z. B. 6-wöchiges oder 24-monatiges Projekt haben möchte, dann ist eines seiner Ziele, eine Darstellung die genau diesen Projektzeiträumen umfasst. Vom Programm gesetzte Einschränkungen wie Wochen- und Monatsansichten wird der Nutzer eher als hinderlich ansehen.

Letztlich handelt es sich nur um eine 1:1-Funktionsübernahme von Kalendern auf Papier. Jedoch kann eine Visualisierung auf dem Bildschirm deutlich mehr leisten, hier ist auch eine dynamische Darstellung möglich. Beispielsweise ein Zeit-Scrollbalken, der nicht an Monatsgrenzen halt macht. Oder eine Art Fish-Eye-View Ansicht, bei der nicht für alle Einträge gleichviel Platz vorgesehen ist, sondern ein Einzelner in der Mitte der Darstellung größer ist und mehr Details anzeigt [Coop05].

6.4. Tabellendarstellung

Eine Tabelle kann sehr viele Informationen kompakt darstellen, in der Beispieltabelle 6.16 wird 3-mal mehr Information wiedergegeben, wie bei einer Prozessgraphdarstellung auf gleichem Raum (vgl. Abbildung 1.3 → S. 7).

BEISPIEL

Das Beispiel aus Abbildung 6.15 zeigt eine Tabelle, die man auch als Aufgaben- oder Aktivitätenliste bezeichnen könnte, da die Zeilen topologisch nach der Aktivitätsstruktur geordnet sind. Die semantische Struktur ist zur besseren Lesbarkeit durch farbige Blöcke untermalt. Dasselbe Prinzip ist auch schon bei der Prozessgraphdarstellung zum Tragen gekommen (siehe Abbildung 6.4).

Die Beispieltabelle in Abbildung 6.16 deutet die Möglichkeit an, Blöcke von Aktivitäten zur besseren Übersicht zusammenzuklappen, um die Darstellung auf das momentan Wesentliche zu begrenzen. Dafür bietet sich eine Baumstruktur an, wie sie die Benutzer beispielsweise vom Windows Explorer her kennen. In einer solchen Baumstruktur lässt sich ein hierarchischer Prozess natürlich abbilden. Die Baumstruktur setzt allerdings eine ganz bestimmte Sortierung voraus. Eine solche Tabelle lässt sich dann nicht mehr nach anderen Kriterien sortieren.

Außer einer an den Aktivitäten orientierten Auflistung sind aber auch andere Auflistungen denkbar. Solche Tabellen können nach allen Ressourcentypen sortiert sein, z. B. nach Ausführungseinheiten, Systemen oder Dokumenten, wobei wieder jede Zeile für eine Aktivität steht. Auch bei diesen anderen Sortierungen kann die Übersichtlichkeit der Darstellung durch Untermahlung mit farbigen Blöcken verbessert werden. Bei einer Sortierung etwa nach Organisationseinheit bilden alle der gleichen Einheit zugeordnete Aktivitäten-Zeilen einen Block. Dies kann geschachtelt erfolgen, ähnlich wie in Abbildung 6.16 dargestellt. Jedoch färbt das Interface, bei vielen Hierarchiestufen, höchstens zwei oder drei ein (z. B. Arbeitsgruppe, Abteilung, Ressort), da die Hinterlegung sonst zu viel Aufmerksamkeit auf sich zieht.

function	role	system	documents in	documents out
initiate change request	CR initiator			change request
determine CR manager	contact person			
instruct expertise	CR manager			
modify CR				
AND block				
generate expertise	car body engineer	CAD system	change request	expertise->car body
generate expertise	electronic engineer	CAD system	change request	expertise->electronic
generate expertise	motor engineer	CAD system	change request	expertise->motor
generate expertise	development chief	CAD system	change request	expertise
request evaluation	CR manager			
EXCEPTION block (CR modification required => modify CR, CR infeasible => conclude CR)				
AND block				
provide evaluation	planning expert	planning system	change request	expertise
provide evaluation	purchase expert	planning system	change request	expertise
provide evaluation	quality expert	planning system	change request	expertise
request comments	CR manager			
EXCEPTION block (CR modification required => modify CR, CR infeasible => conclude CR)				
AND block				
provide comments	planning expert		evaluation	change request
provide comments	planning expert		evaluation	change request
provide comments	construction expert		evaluation	change request
provide comments	quality expert		evaluation	change request
approve CR	CR approval board		evaluation	change request
start realization	CR manager		change request	
CR modification required => modify CR				
realize CR (construction)	construction engineer	CAD system		
CR modification required => modify CR				
start realization	CR manager			
realize CR (production)	production engineer	production planning		
conclude CR	CR manager			

Abbildung 6.15.: Tabellendarstellung des Change-Management-Prozesses – Einfache Tabelle (ohne Hierarchie)

MÄCHTIGKEIT DES KONZEPTES

Die Tabellendarstellung eignet sich auch für sehr große Prozesse, denn kein anderes Konzept bietet derart kompakte Darstellungen. Allerdings kann der Kontrollfluss praktisch nicht dargestellt werden. In einer Tabellenvariante, in der die Aktivitäten in den Zeilen so sortiert sind, dass sequenziell ausgeführte Aktivitäten immer einen Block bilden, lässt sich der Kontrollfluss rudimentär abbilden. Bei größeren Schachtelungstiefen bringt aber auch dies keine genügende Übersicht mehr über den Kontrollfluss. Hinweise auf Verzweigungen, Schleifen, Ausnahmen und Sprünge (siehe Abschnitt 5.3.2) lassen sich in den Tabellenspalten unterbringen, praktikabler dürften jedoch Symbole sein, über denen Tooltips die Details zu Verzweigungsbedingungen und Ähnlichem anzeigen. Um die mangelhafte Darstel-

6. Darstellungsformen für Prozessdaten

id	function	role	system	documents in	documents out	jump destination
1	main process (level 1)					
1	initiation					
1.1	initiate change request	CR initiator			change request	
1.2	determine CR manager	contact person				
2	expertise					
2.1	instruct expertise	CR manager				
2.2	modify CR					
2.3	AND block					
2.3	generate expertise	car body engineer	CAD system	change request	expertise->car body	
2.4	generate expertise	electronic engineer	CAD system	change request	expertise->electronic	
2.5	generate expertise	motor engineer	CAD system	change request	expertise->motor	
2.6	generate expertise	development chief	CAD system	change request	expertise	
3	evaluation					
3.1	request evaluation	CR manager				
3.1	EXCEPTION block					
3.1	AND block					
3.2	provide evaluation	planning expert	planning system	change request	expertise	
3.3	provide evaluation	purchase expert	planning system	change request	expertise	
3.4	provide evaluation	quality expert	planning system	change request	expertise	
	CR modification required					2.2 modify CR
	CR infeasible					8.1 conclude CR
4	commenting					
4.1	request comments	CR manager				
4.1	EXCEPTION block					
4.1	AND block					
4.2	provide comments	planning expert		evaluation	change request	
4.3	provide comments	planning expert		evaluation	change request	
4.4	provide comments	construction expert		evaluation	change request	
4.5	provide comments	quality expert		evaluation	change request	
	CR modification required					2.2 modify CR
	CR infeasible					8.1 conclude CR
5	approval					
5.1	approve CR	CR approval board		evaluation	change request	
6	realization in construction					
6.1	start realization	CR manager		change request		
	CR modification required					2.2 modify CR
6.2	realize CR (construction)	construction engineer	CAD system			2.2 modify CR
	CR modification required					2.2 modify CR
7	realization in production					
7.1	start realization	CR manager				
7.2	realize CR (production)	production engineer	production planning			
8	conclusion					
8.1	conclude CR	CR manager				

Abbildung 6.16.: Tabellendarstellung des Change-Management-Prozesses – Normale Tabelle

lung des Kontrollflusses zu kompensieren, kann die Darstellung, wenn eine Zeile selektiert ist, mit kleinen Markierungen auf Folgeaktivitäten aufmerksam machen.

Das Organisationsmodell hat bei dieser Darstellung keine besondere Bedeutung, jedoch können dadurch beispielsweise in einer Spalte die Abteilungen aufgelistet werden, zu denen der Bearbeiter der Aktivität gehört. Dies erfolgt, indem die Abteilungsinformation aus dem Organisationsmodell von der Bearbeiterinformation abgeleitet wird. Dadurch können zusammengehörige Aktivitäten, die alle einer Abteilung zuzurechnen sind, durch farbig markierte Blöcke hervorgehoben werden.

DARGESTELLTE PROZESSINFORMATIONEN

Die Verzweigungsstruktur eines Prozesses, der durch den Kontrollfluss festgelegt wird, wird aus einer Tabelle nur dann ersichtlich, wenn sie topologisch sortiert ist. In diesem Fall können Aktivitäten die logische Blöcke bilden, in der Tabelle gruppiert dargestellt werden. Sobald eine Tabelle nach anderen Kriterien sortiert wird, kann der Informationsaspekt des Kontrollflusses nur noch insoweit wiedergegeben werden, indem eine Tabellenspalte, die topologische ID jeder Aktivität anzeigt. Das Benutzer-Interface kann jedoch Funktionen anbieten, die jeweilige(n) Folgeaktivität(en) zur aktuell selektierten Aktivität optisch in der Tabelle zu markieren. Die auch zum Kontrollfluss gehörigen Informationen zu Verzweigungsbedingungen, Ausnahmen, Schleifen und Sprüngen lassen sich in zusätzlichen Spalten darstellen oder in Tooltips unterbringen, jedoch bleibt bei einer Tabellendarstellung die Wiedergabe des Kontrollflusses prinzipbedingt mangelhaft. Der Datenfluss lässt sich nur

implizit darstellen, indem in den Aktivitätenzeilen ein- und ausgehende Dokumente angezeigt werden. Den Mittelpunkt der Darstellung bilden immer die Aktivitäten mit den ihnen zugeordneten Informationen. Diese können nach jedem Kriterium sortiert werden. Eine Multi-Instanz Variante ermöglicht auch den Zugriff auf den instanzbezogenen Teil der Prozessinformationen (z. B. zugeordnetes Produkt).

VOR- UND NACHTEILE

Tabellen bieten eine sehr kompakte Darstellung von Prozessabläufen. In keiner anderen Darstellungsform lassen sich so viele Aktivitäten auf begrenztem Bildschirmplatz unterbringen. Und selbst bei großen Prozessen ist nur vertikales Scrollen nötig, um den Prozessverlauf zu verfolgen. Tabellen können eine 'unbegrenzte Menge' von Aktivitätendetails in ihren Spalten anzeigen.

Durch Multi-Instanz-Ansichten können viele Instanzen desselben Schemas effektiv miteinander verglichen werden; auf Wunsch werden vorhandene Informationen aggregiert oder Details von mehreren Instanzen untereinander dargestellt.

Prozessgraphen zeigen die Prozessstruktur sehr viel intuitiver an, als das bei Tabellen möglich ist. Bei großen Prozessen mit großen Schachtelungstiefen kann die Übersicht sehr schnell verloren gehen. Bei nicht blockstrukturierten Prozessen lässt sich die Struktur besonders schlecht darstellen, da in solchen Fällen Sprungziele an andere Stellen der Tabelle angegeben sind und die Tabelle, selbst bei einer Sortierung nach Ausführungsreihenfolge, nicht mehr sequenziell gelesen werden kann. Jedoch dürfte für Benutzer dieser Darstellungsform die Prozessstruktur meist von untergeordneter Bedeutung sein.

MÖGLICHE VARIATIONEN

Variation 1: Arbeitsliste

Um sich schnell einen Überblick über die aktuellen und anstehenden Tätigkeiten innerhalb eines Prozesses zu verschaffen, reicht eine einfache Arbeits- oder Aktivitätenliste aus. Sie zeigt nur Aktivitäten und die ihnen zugeordneten Bearbeiter an. Dabei könnten aktuell in der Ausführung befindliche Aktivitäten und/oder ausführungsbereite Aktivitäten dargestellt werden. Die Darstellung macht auch deutlich, ob das PMS ausführungsbereiten Aktivitäten jeweils einen konkreten Bearbeiter zugeordnet hat. Bereit-Zeiten³ Prioritäten oder Deadlines lassen sich ebenfalls einblenden. Außer nach Bearbeitern, kann diese Liste auch nach allen anderen genannten Kriterien sortiert werden.

Das Konzept einer Liste für einen Grund-Prozess lässt sich leicht für den Multi-Instanz-Fall erweitern. Dann wird nur eine weitere Spalte notwendig, die den Prozess kennzeichnet, dem die jeweilige Aktivität zugeordnet ist. Das ermöglicht dann auch, die Tätigkeiten nach Prozessen zusammenzufassen. Falls ein Kundenattribut gegeben ist, können die Aufgaben auch nach Kunden zusammengefasst werden. Optional ist es möglich, die Liste nach Aktivitäten zu sortieren, sodass gleiche Aktivitäten untereinander erscheinen. Somit kann der Bearbeiter effizienter arbeiten, indem er gleiche Arbeitsschritte am Stück bearbeitet.

Eine Arbeitsliste, die speziell für einen konkreten Mitarbeiter dargestellt wird, zeigt natürlich nur die ihm zugeordneten, ausführungsbereiten Aktivitäten beziehungsweise die vom

³Ein Zeitstempel, der anzeigt, wann eine Aktivität in den Zustand 'bereit zur Ausführung' wechselte. Eine Abarbeitung nach Bereit-Zeit entspricht dem FIFO-Prinzip.

PMS noch nicht zugeordneten Aktivitäten, für die er als möglicher Bearbeiter in Frage kommt. Diese beiden Sorten von Aktivitäten können auch gruppiert dargestellt werden. Zur Motivation sollten auch bereits erledigte Aktivitäten aufgelistet werden können. Als weitere Anwendungsmöglichkeit lassen sich in naher Zukunft anstehende Aktivitäten auflisten. Hierfür werden aktuell laufende Instanzen entsprechend ausgewertet.

Variation 2: Einfache Tabelle

Abbildung 6.15 zeigt den gesamten Change Management Prozess, die ausgegrauten Einträge am Anfang der Tabelle repräsentieren hier beendete Aktivitäten. Sie deuten an, wie weit die Prozessausführung fortgeschritten ist. Fett formatierte Einträge sind aktuell laufende Aktivitäten. Die Tabelle lässt sich nicht umsortieren, sie zeigt die Aktivitäten immer in einer Reihenfolge an, in der die Blockstruktur soweit möglich visualisiert wird. Aktivitäten, die sequenziell ausgeführt werden, stehen dadurch immer untereinander und sind durch eine Linie voneinander abgetrennt. Parallel ausgeführte Aktivitäten werden dagegen durch keine Linie getrennt. Ausnahmebehandlungsaktivitäten (hier: 'modify CR') erscheinen ebenfalls als Einträge. Die Stellen, an denen solche Ausnahmen ausgelöst werden können, zeigen die Ausnahmebedingungen und die jeweiligen Sprungziele.

Variation 3: Normale Tabelle

Jede Zeile hat hier eine eindeutige Nummerierung, die von der topologischen Sortierung abgeleitet ist. Dies erlaubt es, die Tabelle nach jeder beliebigen Spalte zu sortieren und dennoch die Ausführungsreihenfolge im Blick zu behalten.

Durch das Sortieren nach einer Spalte, entstehen in dieser Spalte Gruppen mit gleichen Elementen. Wenn z. B. nach Bearbeitern sortiert wird, bilden alle Zeilen (die jeweils für eine Aktivität stehen), die einem bestimmten Bearbeiter zugeordnet sind, eine Gruppe. Diese Gruppen werden besonders gekennzeichnet, z. B. durch farbige Hinterlegung, analog zur Hervorhebung der Blockstruktur bei der 'Einfachen Tabelle'.

Auch eine Sortierung nach mehreren Kriterien (Spalten) ist möglich, beispielsweise primär nach Dokumenten und sekundär nach Bearbeitern. Dadurch kann sich der Anwender einen Überblick verschaffen, an wie vielen Stellen im Prozess ein und derselbe Bearbeiter an einem Dokument arbeitet.

Die Details, die in den Spalten angezeigt werden, sind beliebig konfigurierbar. Details, die besonders häufig von Interesse sind, können vom Benutzer in den am weitesten links stehenden Spalten angeordnet werden, um häufiges horizontales Scrollen zu vermeiden. Ein komfortables System bietet dem Benutzer das Speichern und Wiederherstellen von Spaltenkonfigurationen an, sodass die benötigten Informationen zu Aktivitäten schnell zur Verfügung stehen. Dies kann in Analogie, zur Prozessgraphdarstellung erfolgen. Ebenso kann die Knotenkonfiguration über ein Fenster im Aufgabenbereich schnell umgeschaltet werden.

Variation 4: Hierarchische Tabelle

Im Gegensatz zu den bisher diskutierten Tabellentypen zeigt Abbildung 6.16 einen hierarchischen Prozess über zwei Ebenen. Da Aktivitäten eines Prozesses wiederum andere Prozesse kapseln können, ist es gegebenenfalls sinnvoll solche integrierten Darstellungen

erzeugen zu können. Innerhalb der ersten Prozess-Ebene, die aus acht Schritten besteht, sind ihre Kindprozesse eingebettet.

In der dargestellten Tabelle sind Aktivitäten, die bereits in der Vergangenheit ausgeführt worden sind, grau hinterlegt. Blockstrukturen sind nicht nur farbig hinterlegt, sondern lassen sich auch zusammenfalten, um die Darstellung bei Bedarf auf das Wesentliche reduzieren zu können.

Multi-Instanz Tabelle

Es sind Tabellenansichten denkbar und auch sinnvoll, die eine Übersicht zu allen Instanzen eines Prozessschemas verschaffen.

Was die Navigation anbelangt, ermöglicht das Benutzer-Interface von den Multi-Instanz-Ansichten aus, beispielsweise per Kontextmenü wieder zu (Einzel-)Instanz-Darstellungen zurückzuwechseln.

Variation 1: Multi-Instanz Übersicht

Jede Zeile der Tabelle stellt eine Instanz dar. In den Spalten stehen Informationen zur jeweiligen Instanz, prinzipbedingt jedoch keine Informationen zu den einzelnen Aktivitäten. Angezeigt werden kann in Tabellenspalten beispielsweise der Prozessstatus oder die aggregierten Gesamtkosten, geschätzte Kosten, geschätzter Abschlusszeitpunkt und ähnliche Instanzbezogene Informationen. Eine sinnvolle Angabe ist auch das zugeordnete Produkt (oder Objekt). Weitere wichtige Informationen über die Instanzen liefern beispielsweise das Instanzierungsdatum, die zuletzt aktivierte Aktivität und dessen Bearbeiter oder das Datum der letzten Änderung. Nach allen Spalten kann sortiert werden.

Der Prozessstatus kann durch einen einfachen grafischen Prozentbalken visualisiert werden, der angibt, wie viele seiner Aktivitäten bereits durchlaufen sind. Auch eine Angabe des erwarteten frühesten Abschlusszeitpunktes ist bei gegebenen durchschnittlichen Bearbeitungszeiten möglich.

Hier ist auch eine Baumstruktur nach Art des Windows Explorer möglich. Die Einteilung in Ordner kann via Produktangabe oder z. B. Kalenderwoche zur Instanzierungszeit erfolgen. Auf diese Weise ist sogar eine Multi-Schema-Übersichtsdarstellung möglich, wenn die Prozesshierarchie als Baumstruktur wiedergegeben wird.

Variation 2: Multi-Instanz-Ansicht

Bei diesem Ansatz werden die Aktivitäten des zugrunde liegenden Prozessschemas aufgelistet und in den Spalten mögliche aggregierte Werte angezeigt. Interessant dürften vor allem Minimal-, Maximal- und Durchschnittswerte sein (d. h. Kosten). Werte die sich nur als Mengen darstellen lassen, wie beispielsweise die verschiedenen den Aktivitäten zugeordneten Ausführungseinheiten, lassen sich nur über eingeblendete Tooltips oder über eine Detailansicht im Aufgabenbereich (siehe Abschnitt 7.4) darstellen. Das zugehörige Tabellenfeld kann aber zumindest über die Mengenkardinalität Auskunft geben. Eine Tabellenspalte zeigt einen segmentierten Balken mit einem Segment pro Instanz. Die Segmente geben über eine Farbcodierung entsprechend über Aktivitätenstatus (abgeschlossen, laufend, bereit, nicht bereit, etc.) Auskunft; alternativ kann dies über eine einfache Prozentangabe geschehen, die für die Anzahl der bereits abgeschlossenen Aktivitäten steht.

Variation 3: Multi-Instanz Aktivitätenprojektion

Im Zusammenhang mit Variante 2 könnte schnell der Wunsch aufkommen, einzelne Aktivitäten verschiedener Instanzen direkt miteinander zu vergleichen. Zu diesem Zweck kann eine Ansicht generiert werden, die genau eine ausgewählte Aktivität für alle Instanzen in eine Tabelle projiziert. Das heißt, für jede Instanz in einer Zeile jeweils die Informationen über die entsprechende Aktivität dieser Instanz anzeigt. Auf diese Art und Weise lassen sich parallel ablaufende Instanzen detailliert überwachen und die Werte in den Tabelleneinträgen müssen nicht mehr aggregiert werden, da jede Zeile für eine Instanz steht.

Der Benutzer kann direkt in der Reihenfolge der logischen Blockstruktur (oder alternativ in topologischer Reihenfolge) vorwärts oder rückwärts zur nächsten Aktivität im Prozess schalten. Denkbar wäre auch die Möglichkeit bei eingeblendetem Übersichtsfenster (siehe Abschnitt 7.5) per Doppelklick direkt zu einer anderen Aktivität zu wechseln. Dieses Übersichtsfenster zeigt immer an, welche Aktivität im Moment dargestellt wird.

KONFIGURATIONSMÖGLICHKEITEN

Konfigurationsmöglichkeiten sind:

- Darstellungsvariante wählen
- Spalten konfigurieren
- Blockmarkierungen an/aus
- Zeilen mit deaktivierten Aktivitäten ausblenden

Blockmarkierungen an/aus

Der Benutzer kann festlegen, ob mehrere untereinander liegende Zeilen als Block markiert werden sollen, falls sie sich gruppieren lassen; etwa alle Aktivitäten, deren Bearbeiter der gleichen Abteilung angehören.

Zeilen mit deaktivierten Aktivitäten ausblenden

Bei Instanz-Darstellungen werden alle Aktivitäten-Zeilen ausgeblendet, die nicht durchlaufen wurden, so dass eine übersichtliche Darstellung resultiert.

6.4.1. Implementierungsaspekte

Große Tabellen brauchen schnell mehr als eine Bildschirmseite Platz. Daher sollte das Scrolling so implementiert werden, dass die Spaltenbeschriftung immer am oberen Rand sichtbar bleibt. Je nach Tabelle bleibt so eventuell auch der linke Rand immer sichtbar, beispielsweise wenn die erste Spalte grafisch die Gruppierung der einzelnen Zeilen zueinander markiert.

6.5. Interaktionsdiagramm

Das Interaktionsdiagramm und die Matrixdarstellung (siehe Abschnitt 6.7), nehmen eine Sonderstellung unter den bisher erörterten Darstellungsformen ein. Wie Abbildung 6.1

(→ S. 74) zeigt, lehnen sich Prozessgraph, Swimlane, Kalender und Tabelle, was die Strukturierung der Darstellung angeht, hauptsächlich an die Prozessstruktur an. Die Matrixdarstellung dagegen zeigt die Zusammenhänge zwischen jeweils zwei Prozess-Aspekten. Die Aufgabe des Interaktionsdiagramms ist es wiederum, Informationsflüsse adäquat zu visualisieren. Sie stellen eine Abstraktion aus dem Prozessablauf dar.

Die hier gewählte Form ist an Sequenzdiagramme, bekannt z. B. aus UML 2.0 [Omg 05b], angelehnt. Sequenzdiagramme sind auch unter dem Namen Message Sequence Charts (*MSC*) bekannt. Die Aufgabe von Sequenzdiagrammen ist es, das Interobjektverhalten zu beschreiben, d. h. den Austausch von Nachrichten zwischen einzelnen Objekten – ihre Interaktion. Weil der Begriff Interaktion besser als der Begriff Sequenz beschreibt, dass es hier um die Darstellung des Informationsflusses geht, wird hier der Name Interaktionsdiagramm gewählt.

Die primäre Zielsetzung besteht darin, die Interaktion zwischen zwei Abteilungen zu visualisieren. Damit lassen sich auch die beiden weniger komplexen Interaktions-Szenarien 'Bearbeiter - Bearbeiter' und 'Bearbeiter - Abteilung' darstellen. Letztlich lässt sich die Kommunikation zwischen verschiedensten Organisationseinheiten, etwa von Arbeitsgruppen visualisieren. Wenn Zugriff auf unternehmensübergreifende Prozesse besteht, können auch die Kommunikationsschnittstellen zwischen zwei Unternehmen ausgewertet werden.

Sobald der Anwender zwei Organisationseinheiten ausgewählt hat – das können im einfachsten Fall auch zwei Rollenangaben oder Personen sein – besteht der erste Schritt darin, Interaktionen aus den vorhandenen Prozessdaten zu extrahieren. Hierfür werden die Ausführungseinheiten, der am Prozess beteiligten Aktivitäten, untersucht, ob sie einer der beiden ausgewählten Organisationseinheiten zugeordnet sind. Mithilfe der beiden folgenden Regeln, lassen sich dann die Stellen eines Prozesses herausfiltern, wo Interaktionen zwischen den beiden ausgewählten Organisationseinheiten stattfinden können:

1. Der Zugriff auf ein Dokument erfolgt nacheinander von beiden Ausführungseinheiten.
2. Zwei unmittelbar aufeinanderfolgende Aktivitäten, die den beiden Ausführungseinheiten zugeordnet sind.

Dies sind zwei notwendige Bedingungen für eine Interaktion, hinreichende Bedingungen können nur definiert werden, wenn das verwendete Metamodell Interaktionen explizit deklariert.

BEISPIEL

Abbildung 6.17 zeigt ein Interaktionsdiagramm zwischen zwei Abteilungen. In der Prozessablaufreihenfolge werden nacheinander alle Schnittstellen innerhalb des Prozesses zwischen den beteiligten Ausführungseinheiten dargestellt. Die an der Schnittstelle beteiligten Aktivitäten werden dargestellt, hier mit Nennung der jeweiligen Bearbeiter. Schnittstellen können aus mehreren in Reihe oder parallel ablaufenden Interaktionen bestehen. Die einzelnen Schnittstellen sind optisch durch einen Trennstrich voneinander getrennt. Die Art einer Interaktion ist sofort aus der Kantenfarbe ersichtlich.

Durch die oben genannten Regeln, lassen sich zwei Arten des Informationsflusses unterscheiden: Kontrollflussinteraktion und Datenflussinteraktion. Diese können, wie im Beispiel, auch kombiniert auftreten. Überall wo Daten ausgetauscht werden, sind die beteiligten Dokumente (oder Daten) entlang der Interaktionskante aufgelistet.

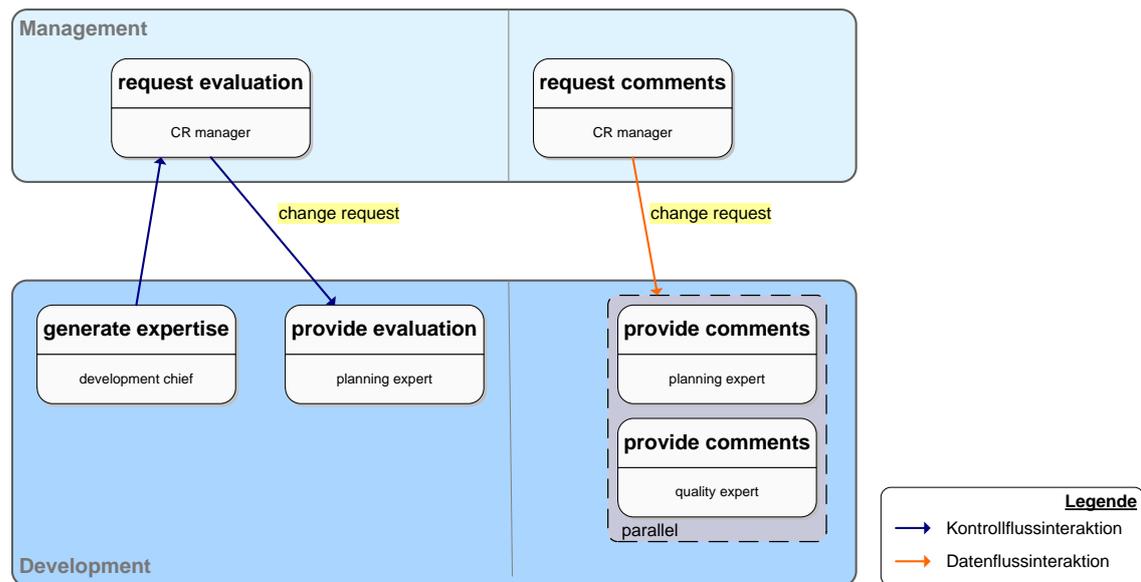


Abbildung 6.17.: Interaktionsdiagramm – Schnittstellen zwischen Abteilungen

MÄCHTIGKEIT DES KONZEPTE

Das Interaktionsdiagramm in der hier beschriebenen Form ist von *MSC-Darstellungen* abgeleitet, eine Verwandtschaft zur Swimlane-Darstellung besteht jedoch auch. In Swimlanes können mit Hilfe geeigneter Zusammenfassungen der Lanes ähnliche Darstellungen erzeugt werden. Dort sind sie jedoch nicht auf zwei Organisationseinheiten begrenzt. Interaktionsdiagramme sind jedoch im Allgemeinen übersichtlicher.

Die Anwendungsmöglichkeiten zur Darstellung von Interaktionen beschränken sich nicht auf die Auswahl zweier beliebiger Organisationseinheiten. Allgemein können zwei beliebige disjunkte Mengen aus der Gesamtheit aller Organisationseinheiten für die Visualisierung herangezogen werden. Als Datenbasis können einzelne Schemata oder Instanzen dienen. Es können aber auch viele Prozessschemata gleichzeitig auf Interaktionen untersucht werden. Es ist somit auch möglich, dass sich ein Anwender (zumindest auf Rollenbasis) alle Kommunikationsschnittstellen zu Mitarbeitern aus anderen Abteilungen oder zu allen anderen Mitarbeitern visualisieren lassen kann. Für konkrete Bearbeiter gelingt dies jedoch nicht, da die Bearbeiterinformationen mehrerer Instanzen desselben Schemas beim Aggregieren verloren gehen beziehungsweise durch Rollenangaben ersetzt würden.

Die Darstellung berücksichtigt die Ablaufreihenfolge und es wird zwischen sequenziell und parallel ablaufenden Interaktionen unterschieden. Oder- & Und-Verzweigungen lassen sich über beschriftete Gruppierungen (siehe Abbildung 6.17) unterscheiden. Verzweigungsbedingungen lassen sich über Tooltips in Erfahrung bringen. Schleifen lassen sich durch Symbole an den einzelnen Prozessschnittstellen kenntlich machen. Die Darstellung von Interaktionen lässt sich auf einzelne Schemata und Instanzen begrenzen. Es können aber auch Sprünge mit ausgewertet werden oder ganze Teile aus der Prozesshierarchie auf Interaktionen hin untersucht werden.

Synchronisationskanten können auch für Interaktionen stehen. Bei Bedarf könnte so noch eine dritte Interaktionsregel (siehe oben) eingeführt werden: „Eine Synchronisationskante verbindet zwei Aktivitäten, die den beiden Ausführungseinheiten zugeordnet sind.“

VOR- UND NACHTEILE

Interaktionsdiagramme stellen ein äußerst effizientes Visualisierungskonzept dar. Die erzeugten Darstellungen sind sehr übersichtlich, auch weil konzeptbedingt maximal eine Verzweigungstiefe angezeigt wird. Mithilfe von Interaktionsdiagrammen erhält der Anwender sehr schnell eine kompakte Sicht auf den Prozess mit den gewünschten Informationen. Auch Prozessverantwortliche oder andere Personen, die für andere Anwender Views erstellen, finden hierin ein geeignetes Werkzeug um personalisierte Views für bestimmte Nutzerkreise, beispielsweise Unternehmenskunden zu erstellen.

MÖGLICHE VARIATIONEN

Es existieren keine besonderen Varianten. Durch die vielen möglichen Kombinationen von Organisationseinheiten erhält die Darstellung jedoch auch eine starke Variabilität. Bei der Darstellung einer Instanz werden Oder-Verzweigungspfade, die nicht gewählt wurden grau markiert. Nicht ausgeführte Aktivitäten erhalten eine entsprechende Zustandsmarkierung (siehe Abschnitt 5.3.3). Wenn die Kommunikationsschnittstellen mehrerer Schemata dargestellt werden, wird der jeweilige Herkunftsprozesses angezeigt.

KONFIGURATIONSMÖGLICHKEITEN

Als graphbasierte Darstellungsart, ähneln die Konfigurationsmöglichkeiten denen der Prozessgraphdarstellung. Weitere Konfigurationsmöglichkeiten sind:

- Organisationseinheiten wählen
- Sprüngen (Verweisen) zu anderen Prozessen folgen

Sprüngen (Verweisen) zu anderen Prozessen folgen

Der Benutzer kann festlegen, ob auch weitere mit dem Schema verbundene Schemata für die Suche nach Interaktionen mit herangezogen werden sollen.

6.5.1. Notwendige Voraussetzungen

Der Einsatz von Interaktionsdiagrammen wird lediglich durch die, im Organisationsmodell beschriebenen, Organisationseinheiten begrenzt.

6.6. Datenflussdiagramm

Datenflussdiagramme (DFD) sind vor allem als Modellierungssicht und Teil von UML bekannt. Entwickler erstellen im Rahmen der Analyse diese Diagramme. Das Ziel der Prozessvisualisierung geht jedoch in die andere Richtung, Diagramme sollen hierfür automatisch erzeugt werden.

Ähnlich wie bei den UML-Datenflussdiagrammen ist das Ziel dieser Darstellungsform den Daten- und Dokumentfluss darzustellen. Jedoch wird immer nur ein Dokument oder eine Datenstruktur für die Visualisierung ausgewählt. Die Darstellung erfolgt anhand einer abstrakten Graphdarstellung.

BEISPIEL

Abbildung 6.18 zeigt ein Datenflussdiagramm des Dokuments 'change request' aus dem Change-Management-Prozess. In der gezeigten Prozessansicht erscheinen alle an der Bearbeitung des Dokuments beteiligten Aktivitäten in Ausführungsreihenfolge. Verzweigungen werden dabei normal dargestellt. Die Kanten sind je nach Zugriffstyp farblich markiert. Die Zuordnung ist aus der Legende ersichtlich (siehe Abschnitt 7.4).

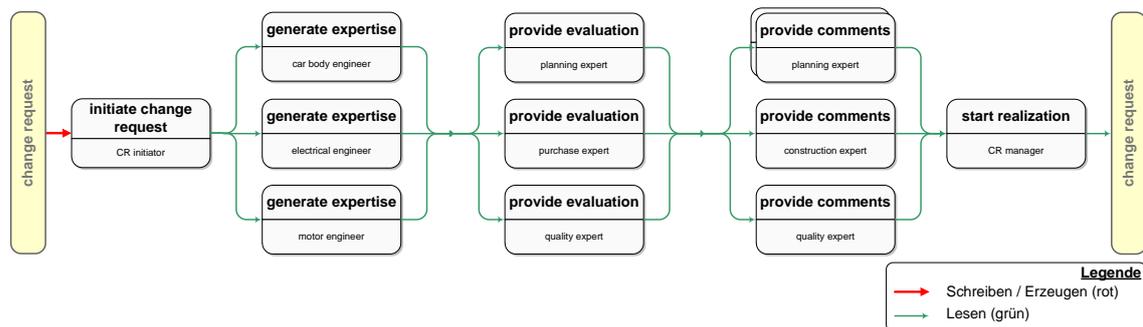


Abbildung 6.18.: Datenflussdiagramm – Dokument: change request

MÄCHTIGKEIT DES KONZEPTES

Die Darstellung erlaubt es den Ablauf eines Prozesses aus der Sicht eines Dokumentes zu verfolgen. Der Kontrollfluss lässt sich analog zur Prozessgraphdarstellung anzeigen, bleibt aber auf die jeweilige Sicht begrenzt. Außer prozess-lokalen Dokumenten, lässt sich auch der Dokumentfluss (eines global zugreifbaren Dokumentes) in Prozesshierarchien verfolgen. Auch eine aggregierte Multi-Instanz Darstellung ist möglich.

VOR- UND NACHTEILE

Durch die sehr begrenzte View ist der Datenfluss selbst bei großen Prozessen gut nachvollziehbar. Wenn zwingend mehr als ein Dokumentfluss dargestellt werden soll, muss beispielsweise auf eine Swimlane-Darstellung ausgewichen werden, die die Aktivitäten zu gewünschten Dokumenten/Daten in jeweils verschiedenen Lanes führt.

MÖGLICHE VARIATIONEN

Außer der vorgestellten Variante, die den Datenfluss eines Dokumentes innerhalb eines Schemas oder einer Instanz anzeigt, lässt sich auch eine prozessübergreifende Variante und eine Multi-Instanz Dokumentenübersicht realisieren.

Prozessübergreifendes Datenflussdiagramm

Diese Variante ermöglicht, dass Dokumente, die innerhalb einer Prozesshierarchie an verschiedenen Stellen (Prozessen) verwendet werden, ebenso in einer Datenflussansicht dargestellt werden können, wie lokale Dokumente eines Prozesses. Hierfür wird die Ausführungsreihenfolge innerhalb der Prozesshierarchie bestimmt. Daraus wird ein Datenflussgraph erzeugt. Die Ausführungsreihenfolge muss nicht in eine Sequenz zerlegbar sein: unbestimmte, bedingte oder parallele Ausführungsfolgen, werden in entsprechende Verzweigungsstrukturen umgesetzt. Der Graph stellt initial nur einen Knoten pro beteiligtem Prozess dar. Prozesse, die mehrere Aktivitäten enthalten, die auf das Dokument zugreifen, lassen sich analog zur Darstellung von Subprozessen öffnen (vgl. Abbildung 5.7 → S. 66).

Multi-Instanz Variante

Mehrere Instanzen eines Schemas werden aggregiert dargestellt, indem eine Status-Information über jedem Knoten eingeblendet wird. Dies kann ein segmentierter Balken sein, der in jedem Segment farbcodiert über den Aktivitätenstatus (abgeschlossen, laufend, bereit, nicht bereit, etc.) der aggregierten Aktivitäten Auskunft gibt. Diese Variante kann dazu verwendet werden, sich schnell einen Überblick über den Fortschritt/Zustand, beispielsweise jedes 'change request' Dokumentes der angezeigten Instanzen zu verschaffen.

KONFIGURATIONSMÖGLICHKEITEN

Als graphbasierte Darstellungsart, ähneln die Konfigurationsmöglichkeiten denen der Prozessgraphdarstellung.

6.7. Matrixdarstellung

Die bisher vorgestellten Darstellungsformen erlauben die Visualisierung aller möglichen Aspekte von Prozessinformationen. Es fehlt noch eine Darstellung, die es erlaubt, Zusammenhänge zwischen all diesen Aspekten darzustellen. Daraus ergibt sich ganz natürlich die Darstellung von je zwei dieser Aspekte als Matrix, die solche Verbindungen explizit machen. Von dieser Darstellungsform existieren verschiedene Varianten, es gibt keine explizite Hauptform. Die möglichen Variationen der Matrixdarstellung zeigt Tabelle 6.2.

Die Felder der Matrix sind Kreuzungspunkte, definiert durch jeweils ein Element aus zwei Informationsaspekten. Jedes Feld steht für einen Ausschnitt aus den Prozessinformationen. Bei diesen Prozess-Ausschnitten handelt es sich immer um eine ganze (oder auch mehrere) Aktivitäten. Da es aber zu unübersichtlich wäre, alle Informationen zu einer Aktivität direkt in der Matrix anzuzeigen, wählt der Benutzer einzelne Attribute aus, die dann in der Darstellung erscheinen. Weitere Informationen können durch Tooltips und durch seitlich eingeblendete Detailinformationen (z. B. im Aufgabenbereich 7.4) dargestellt werden. In Fällen, in denen die Information dazu ausreicht, ob überhaupt ein Zusammenhang zwischen zwei Informationsaspekten besteht, kann das Matrixfeld auch einfach nur ein Häkchen zeigen. Diese Abstraktion sollte als Voreinstellung dienen, da sie zu den einfachsten und übersichtlichsten Darstellungen führt.

Dimension 1 (vertikal)	Dimension 2 (horizontal)	besondere Information
Schema & Instanz Darstellung		
Aktivität	Ausführungseinheit ^a	Status der Aktivität (z. B. beendet, gestartet, abgebrochen)
Aktivität	Dokument	lesen/schreiben, Reihenfolge der Zugriffe, Zeit des letzten Zugriffs
Aktivität	System	-
Ausführungseinheit	Dokument	lesen/schreiben, Reihenfolge der Zugriffe, Zeit des letzten Zugriffs, zugeordnete Aktivität
Ausführungseinheit	System	zugeordnete Aktivität
Dokument	System	zugeordnete Aktivität
Multi-Instanz-Darstellung		
Prozess ^b	Ausführungseinheit	-
Prozess	Dokument	-
Prozess	System	-
Prozess	Aktivität	-

^aAusführungseinheit steht hier für Bearbeiter, Rolle oder Organisationseinheit

^bDie Begriffe Prozess und Instanz werden hier in diesem Kontext synonym verwendet

Tabelle 6.2.: Varianten der Matrixdarstellung

Wenn es für einen Kreuzungspunkt mindestens eine Aktivität in den Prozessinformationen gibt, diese also die beiden durch die Elemente definierten Bedingungen erfüllt, ist das Matrixfeld nicht leer. Das impliziert auch, dass es Fälle gibt, bei denen ein Kreuzungspunkt für mehrere Aktivitäten steht. Dies muss dann, dem Benutzer entsprechend signalisiert werden. Entweder werden dann direkt die Attribute aus mehreren Aktivitäten angezeigt oder eine Zahl gibt an, wie viele Aktivitäten sich hinter dem jeweiligen Eintrag verstecken.

BEISPIEL

Ein einfaches Beispiel, wie eine solche Matrixdarstellung aussehen könnte, liefert Abbildung 6.19. Die Darstellung zeigt in einer Dimension alle an einem Prozess beteiligten Ausführungseinheiten an und in der anderen Dimension die beteiligten Systeme.

In den Schnittpunkten der beiden Informationsdimensionen lässt sich im einfachsten Fall, wie in der Abbildung gezeigt, darstellen, ob überhaupt ein Zusammenhang zwischen diesen beiden Objekten besteht. Oft bietet es sich auch an, an dieser Stelle weitere Information über einen bestehenden Zusammenhang einzublenden.

Abbildung 6.20 zeigt ein weiteres Beispiel einer Matrixdarstellung. Es spiegelt das genaue Gegenteil des einfachen Beispiels wieder. Es werden die Ausführungseinheiten und Dokumente eines Prozesses gegenübergestellt. Jeder Eintrag zeigt an, ob das betreffende Dokument durch den jeweiligen Bearbeiter gelesen (grüne Markierung) und/oder geschrieben wird (rote Markierung).

Die Zahl hinter jedem Eintrag weist darauf hin, in welcher wahrscheinlichen Reihenfolge die Aktivitäten mit Lese- und Schreibzugriffen im Prozesskontext angeordnet sind. Diese Zusatzinformation ist dann von Interesse, wenn ein Benutzer nicht nur wissen will, welche

Division	Role	System		
		CAD system	planning system	production planning
Management	CR Manager			
Construction	car body engineer	✓		
	electronic engineer	✓		
	motor engineer	✓		
	construction expert			
	construction engineer	✓		
	production engineer			✓
	quality expert			
Development	development chief	✓		
	planning expert		✓	
	purchase expert		✓	
Accounting	contact person			
others	CR initiator			
	CR approval board			

Abbildung 6.19.: Matrixdarstellung - Ausführungseinheiten und Systeme

Aktivität oder welcher Bearbeiter auf ein Dokument zugreift, sondern wenn für ihn die logische Abfolge, in der Aktivitäten auf das jeweilige Dokument zugreifen ebenfalls relevant ist. Dies ist notwendig, da die Matrixdarstellung ohne diesen Zusatz keine Rückschlüsse auf die Prozessstruktur erlaubt. Jede Zahl ist dabei eindeutig einer Aktivität zugeordnet, vorausgesetzt, die Reihenfolge ist eindeutig bestimmbar. Bei parallelen Aktivitäten ist dies nur möglich, wenn die Start- und Endzeiten bekannt sind. Diese Zeiten müssen allerdings überlappungsfrei oder die Aktivitäten durch Synchronisierungskanten geordnet sein.

Die Matrixdarstellung zeigt immer nur einen kleinen Anteil der insgesamt verfügbaren Prozessinformationen an. Prinzipbedingt lassen sich nur wenige Zusatzinformationen in den Matrixfeldern unterbringen, ohne dass die Übersichtlichkeit verloren geht. Eine Prozessgraphdarstellungen bleibt dagegen auch bei vielen angezeigten Aktivitätendetails noch 'einigermaßen' lesbar. Abbildung 6.21 zeigt auch hier eine Möglichkeit, durch eingeblen-dete Tooltips, mehr mit einem Matrixfeld in Zusammenhang stehende Information zur Anzeige zu bringen.

Tooltips lassen sich in jeder Darstellungsform einsetzen. Im Kontext der Matrixdarstellung müssen allerdings für jede Darstellungsvariante andere Details in den Tooltips dargestellt werden, denn jede Kombination von Prozessaspekten in den beiden Matrixdimensionen zeigt jeweils einen anderen Anteil der verfügbaren Prozessinformationen an, und damit sind es auch jeweils andere Zusatzinformationen die per Tooltip sinnvoll eingeblen-det werden können.

Wenn in den Matrixfeldern den Aktivitäten zugeordnete Dokumente angezeigt werden, erscheinen diese gleich als anklickbarer, verknüpfter Link zum jeweiligen Dokument.

Falls dargestellte Informationen in Listenform nicht mehr in ein Matrixfeld passen, wie im Fall einer Vielzahl von mit der Aktivität verknüpften Dokumenten, können sie in einer aufklappbaren Liste dargestellt werden.

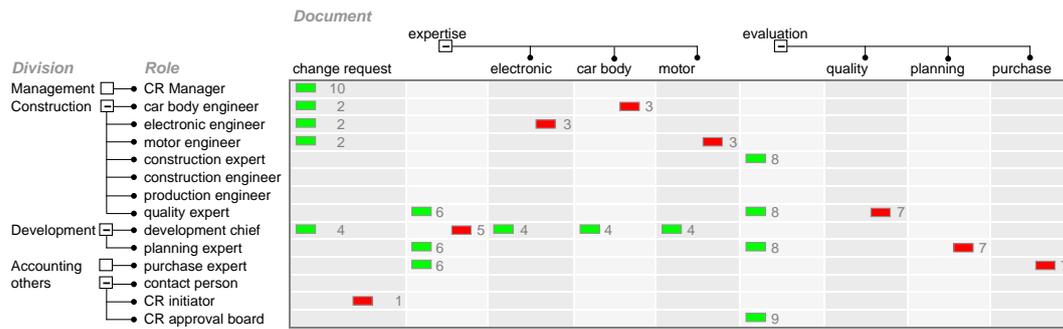


Abbildung 6.20.: Matrixdarstellung - Ausführungseinheiten und Dokumente

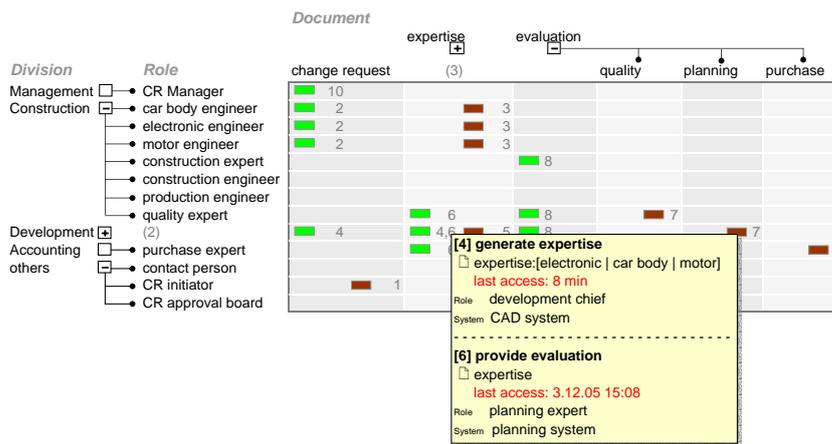


Abbildung 6.21.: Matrixdarstellung - Ausführungseinheiten und Dokumente mit Tooltip

MÄCHTIGKEIT DES KONZEPTE

Das Matrixkonzept eignet sich nicht für die Darstellung des Kontrollflusses. Verzweigungen, Schleifen und Ausnahmen lassen sich daher schlecht abbilden. Verzweigungsbedingungen des Kontrollflusses ließen sich zwar als Eintrag in den Matrixfeldern prinzipiell anzeigen. Tooltips eignen sich hierfür aber besser. Diese können auch weitere Informationen anzeigen etwa zu Ausnahmen oder zu einem Schleifenzähler.

Der Datenfluss lässt sich nur bedingt verfolgen. Zwar lassen sich sehr viele Informationen zu Dokumenten darstellen etwa wo und durch wen diese verwendet werden. Die Reihenfolge der Zugriffe lässt sich dagegen nicht exakt bestimmen. Dies ist wiederum eine Konsequenz daraus, dass der Kontrollfluss in einer Matrix nicht abgebildet werden kann.

Dieses Konzept profitiert sehr vom Vorhandensein von umfangreichen Organisationsmodellen, denn damit können die für die Matrixdimensionen gewählten Prozessaspekte, wie Aktivität, Ausführungseinheit, Dokument und System in einer hierarchischen Ordnung dargestellt werden und zu Oberkategorien zusammengefasst werden. Dies macht die Anzeige von großen Prozessen deutlich übersichtlicher.

Die Matrixdarstellung eignet sich sowohl für die Darstellung von Schemata, als auch für Instanzen. Die Zielsetzung, Zusammenhänge aufzuzeigen, wird auch schon bei der Darstellung eines Schemas gut erreicht. Wesentliche Vorteile hat eine Instanzdarstellung nicht

mehr, es sei denn der Benutzer interessiert sich für Laufzeitdetails, wie z. B. die konkreten Bearbeiter der Aktivitäten eines Prozesses.

In den Schema- und Instanz-Darstellungen lassen sich aktivitätszentrierten Prozessinformationen besonders gut darstellen; instanzbezogenen Prozessinformationen wie zugeordnetes Produkt, Prozessverantwortlicher oder aggregierte Gesamtkosten machen hier wenig Sinn. Die Multi-Instanz-Darstellungen gleichen das aus, hier lassen sich aktivitätszentrierte Daten nicht direkt darstellen, sondern lediglich in aggregierter Form über alle Aktivitäten eines Prozesses. Eine Ausnahme bildet die Multi-Instanz Variante 'Prozess & Aktivität' (→ S. 116), die von den Darstellungsmöglichkeiten her eher zu den Schema- und Instanz-Darstellungen zu rechnen ist.

Auch Multi-Instanz-Darstellungen sind möglich. Mehrere Varianten sind hier möglich, beispielsweise kann eine Prozessliste als neue Matrixdimension eingeblendet werden. Optional kann eine Prozessliste in eine hierarchische Ordnung gebracht werden, indem eine Gruppierungsebene nach Produkt (beziehungsweise Objekt) eingezogen wird. Damit ist es möglich nur, die Instanzen zu beobachten, die einem bestimmten Produkt zugeordnet sind.

Hierarchische Prozesse lassen sich prinzipiell darstellen. Dies wirft aber erhebliche Probleme auf, die aber lösbar sind. Die Darstellung hierarchischer Prozesse ist eine Konfigurationsmöglichkeit (siehe unten). Es dürfte nicht immer gewünscht sein, dass nicht nur der aktuelle Prozess ausgewertet wird, sondern auch all seine Unterprozesse oder Verweise (Sprünge) an andere Stellen eines hierarchischen Workflows. Details dieser Probleme werden im folgenden Abschnitt diskutiert.

VOR- UND NACHTEILE

Die Vorteile der Matrix-Darstellung liegen darin, dass allgemeine Zusammenhänge zwischen den Prozess-Aspekten Aktivität, Ausführungseinheit, Dokument und System gut darstellbar sind. Der Fokus liegt auf der Gegenüberstellung von Aspekten, im Gegensatz zu den anderen Darstellungen, wo die Aspekte zwar angezeigt werden, aber Beziehungen unter ihnen nicht leicht ersichtlich sind.

Ein wichtiger Vorteil der Matrixdarstellung ist, dass sich Informationen gut zusammenfassen lassen, wenn ein Organisationsmodell die nötigen Hinweise über die Zusammenhänge liefert. So zeigt beispielsweise Abbildung 6.20, dass in einer Aktivität auf den kompletten Datensatz (z. B. 'expertise') zugegriffen wird, in anderen dagegen nur auf einzelne Bereiche (z. B. 'expertise->electronic'). Abbildung 6.21 zeigt nur noch den übergeordneten Datensatz (hier 'expertise'). Des Weiteren lassen sich mit den Daten aus dem Organisationsmodell beispielsweise alle Bearbeiter oder Rollen, die einer Abteilung zugeordnet sind, eben dieser zuordnen (hier 'development').

Dieses Zusammenfassen fördert das Problem zutage, wie nun die Informationen aus mehreren Matrixfeldern zu einem Matrixfeld aggregiert werden können. Je nach Art der dargestellten Daten, können dies z. B. Durchschnittswerte oder Listen sein, wie in Abbildung 6.21 (→ S. 108) dargestellt. Die Daten können auch auf andere Weise aggregiert werden. Mit der Aggregation beschäftigt sich Abschnitt 7.1.1.

Weitere Nachteile dieser Darstellung liegen darin, dass sich die logische und zeitliche Aktivitätenabfolge nicht gut visualisieren lässt. Nur wenn eine Matrixdimension die Aktivitäten in topologischer Sortierung auflistet, sind Rückschlüsse auf die Ausführungsreihenfolge

möglich, ansonsten gibt es nur noch die Möglichkeit einer expliziten Nummerierung (wie in Abbildung 6.20 vorgenommen). Nur eine optische Markierung kann den Benutzer in die Lage versetzen, auf den ersten Blick Aktivitäten in der Matrix zu erkennen, welche zu einer Schleife gehören. Auch Aktivitäten die zu externen Prozessen gehören und bei der Darstellung von hierarchischen Prozessen durch Sprungkanten erreicht wurden, können durch ein Symbol gekennzeichnet werden. Vor allem aber ist wichtig, dass Aktivitäten markiert werden, an deren Ausführung Bedingungen geknüpft sind, also Aktivitäten nach bedingten Verzweigungen oder Ausnahmebehandlungen (siehe dazu 6.7.3).

Des Weiteren lassen sich mit der Matrixdarstellung immer nur zwei Aspekte gegenüberstellen. Zudem lassen sich aus Gründen der besseren Übersichtlichkeit, nur recht wenige Details in den Matrixfeldern unterbringen.

Wenn es hierarchische Prozesse gibt und der Benutzer bei der Auswertung eines Prozesses also auch dessen Unterprozesse in der Matrixdarstellung mit berücksichtigen möchte, wirft dies viele Fragen auf, die im Folgenden diskutiert werden:

Problem 1: Änderungen an den Einträgen der Matrixdimensionen

Prinzipiell gibt es zwei Fälle die Matrixdimensionen betreffend, die im Zusammenhang mit hierarchischen Prozessen betrachtet werden müssen:

Fall 1: Aktivitäten als Matrixdimension gewählt

Wo eine Aktivität einen Unterprozess kapselt, kann dieser aufgeklappt werden. Die Auswertung erscheint für den Anwender konsistent, wenn Unterprozesse erst mit dem Ausklappen berücksichtigt werden. Gegebenenfalls sollte auch eine Maximaltiefe für die Auswertung vom Benutzer angegeben werden können. Wo jedoch Unterprozesse in der ersten Matrixdimension aufgeklappt werden, kann es notwendig werden auch die zweite Matrixdimension anzupassen, wenn dadurch neue Bearbeiter, Systeme oder Dokumente der zweiten Dimension an die bestehenden Einträge der Spalten hinten angefügt werden müssen – 'Hinten anfügen', weil sich dadurch die Änderung des Aufbaus der Tabelle für den Benutzer am wenigsten störend bemerkbar macht. Der Übersichtlichkeit halber werden immer nur die Bearbeiter, Systeme und Dokumente dargestellt, die auch von den eingeblendeten Aktivitäten verwendet werden.

Fall 2: Aktivitäten sind nicht als Matrixdimension gewählt

Auch hier können sich die Einträge in den Matrixdimensionen ändern, wenn die Unterprozesse des zu untersuchenden Prozesses mit ausgewertet werden. Da hier jedoch nicht, wie in Fall 1, der Benutzer direkt steuern kann bis zu welcher Tiefe Unterprozesse mit ausgewertet werden sollen, ist hier eine explizite Angabe (Konfigurationsmöglichkeit) nötig.

Problem 2: Einträge in den Matrixfeldern

Was aber zeigen nun die Matrixfelder an, wenn die Prozesse hierarchisch ausgewertet werden? Zuerst einmal ist es sinnvoll, die Matrixfelder farblich zu kodieren, je nachdem, zu welchem Prozess die dahinterliegende Aktivität gehört. In einer Legende im Aufgabenbereich könnten die verwendeten Farben einzeln aufgeschlüsselt werden (siehe 7.4). Weiterhin gibt es hier auch wieder verschiedene Fälle zu betrachten, je nachdem welche Inhalte ein Matrixfeld zeigt.

Fall 1: Anzeige von aggregierbaren Attributen (z. B. Kosten oder Bearbeitungszeit)

Wenn eine Aktivität des dargestellten Prozesses einen Kindprozess kapselt und von diesem auch Aktivitäten in der Darstellung erscheinen, dann zeigt das der kapselnden Aktivität zugeordnete Matrixfeld die aggregierten Gesamtkosten des Kindprozesses an und einige weitere Felder zeigen einzelne Kosten von Aktivitäten dieses Kindprozesses an. Aus der Darstellung muss ersichtlich sein, dass Einzelemente und aggregiertes Element logisch zusammengehören. Auch dies kann durch geschickte Farbgebung erfolgen. Beispielsweise kann das aggregierte Element eine gesättigte Hintergrundfarbe erhalten, die anderen Elemente eine helle Hintergrundfarbe des selben Farbtons (siehe Abschnitt 5.1); oder sie erhalten alle dieselben Farben und der Text des aggregierten Elementes wird fett ausgezeichnet.

Fall 2: Anzeige aller anderen Attribute oder auch Anzeige eines Häkchens zur Zuordnung oder Anzeige von Prozess- oder Aktivitätenlisten.

In diesem Fall ergeben sich keine Besonderheiten.

Problem 3: Bedingungen im Kontrollfluss

Aber auch damit sind noch nicht alle Besonderheiten aufgezählt. Denn bei der Darstellung von aggregierten Werten gibt es zwei besondere Fälle zu beachten. Ist der Kontrollfluss, wegen bedingten Ausführungspfaden, die zum Darstellungszeitpunkt noch nicht aufgelöst sind, noch nicht vorherbestimmt, dann gilt:

Fall 1: Bedingte Verweise zu anderen Prozessen.

Wo Verweise (Sprünge) zu anderen Prozessen vorkommen, kann es sein, dass diese durch eine davor liegende Oder-Verzweigung nur bedingt ausgeführt werden. Wenn Verweise zu anderen Prozessen nicht berücksichtigt werden sollen, ergibt sich kein Problem. Im Falle einer Berücksichtigung, können sich neue Einträge in den Matrixdimensionen ergeben (z. B. durch weitere Bearbeiter). Es kommt durch die Berücksichtigung des Verweises aber in jedem Fall zu neuen Matrixeinträgen. Diese müssen den Benutzer optisch darauf hinweisen, dass es sich bei den dahinter liegenden Aktivitäten um nur bedingt Ausgeführte handelt. Wo ein Matrixfeld einen aggregierten Wert anzeigt, kann dann wegen der unbestimmten Ausführung die Ausgabe als 'von-bis' Wert erfolgen oder der bedingte Wert in Klammern.

Fall 2: Bedingte Ausführungspfade im Kontrollfluss von Unterprozessen

Es liegt der Fall vor, dass ein Unterprozess mit eingeblendet wird. Das Matrixfeld, dessen dahinter liegende Aktivität diesen Unterprozess kapselt, zeigt dann einen aggregierten Wert für den (gesamten) Unterprozess. Wenn dieser einen unbestimmten (weil bedingten) Kontrollfluss enthält, erfolgt die Ausgabe in diesem Matrixfeld als 'von-bis' Wert.

Konsequenzen, die sich aus bedingten Kontrollflüssen für die Matrixdarstellung ergeben, behandelt Abschnitt 6.7.3.

MÖGLICHE VARIATIONEN

Andere Varianten als die in Tabelle 6.2 aufgelisteten sind nicht sinnvoll, denn nur von Aktivität, Ausführungseinheit, Dokument und System existiert eine abgeschlossene Menge von Elementen, die alle mindestens einmal in den Prozessinformationen vorkommen.

Alle anderen Attribute, wie z. B. Kosten oder Prozessverantwortliche, bieten für eine Matrixdarstellung entweder nicht genügend Diversifizierung oder sie lassen sich besser in einer Tabellendarstellung darstellen. Dort lassen sich dann beispielsweise auch Zusammenhänge wie Kosten & Systeme untersuchen, indem die Tabelle primär nach Systemen sortiert wird und sekundär nach Kosten (siehe Abschnitt 6.4). Bei Multi-Instanz Matrixdarstellungen werden dagegen auch Prozesslisten, Produkte (oder Objekte) und Kunden zu sinnvollen Matrixdimensionen, da auch hier jeweils eine endliche abgeschlossene Menge von Elementen existiert.

In den Matrixfeldern lassen sich in jeder Variante beliebige Aktivitätenattribute und Objekte anzeigen (aktivitätsbezogene Informationen). Zusätzlich gibt es bei einigen Varianten besondere Informationen, die nur dort besonders passend sind. Diese listet die Tabelle auf. Außer den aktivitätsbezogenen Prozessinformationen, gibt es noch instanzbezogenen (wie z. B. der Prozessverantwortliche). In einer Matrixdarstellung ist beides nicht kombinierbar. Die meisten Multi-Instanz Varianten sind jedoch instanzbezogen, sie können explizit nur diesen Teil der Prozessinformationen darstellen.

Zunächst werden die Varianten für die Schema- & Instanz-Darstellung betrachtet, anschließend die Varianten für die Multi-Instanz-Darstellung. Der erstgenannte Informationsaspekt stellt im Folgenden immer die erste Matrixdimension dar, d. h. die Zeilen sind der ersten Dimension zugeordnet, die Spalten der zweiten.

Generell ließen sich die beiden beteiligten Aspekte jeweils beliebig als Dimension 1 und 2 einteilen, aber da 'Nicht alles was möglich ist, gut ist', stehen die Aktivitäten stets in der vertikalen Dimension. Dadurch werden mehr Einheitlichkeit erreicht, sowie die Übersichtlichkeit und Wiedererkennbarkeit gefördert. In der Tabellendarstellung sind Aktivitäten stets vertikal angeordnet. Wenn in einer Matrix die Aktivitäten nicht dargestellt werden, stehen stattdessen die Ausführungseinheiten in der vertikalen Dimension, was auch bei der Kalenderdarstellung und bei der Swimlane-Darstellung häufig der Fall ist.

Schema- & Instanz-Darstellung

Variante 1: Aktivität & Ausführungseinheit

Den Aktivitäten eines Prozesses werden die ihnen zugeordneten Ausführungseinheiten gegenübergestellt. Im nicht zusammengefalteten Grundzustand, enthält jede Zeile genau ein gefülltes Matrixfeld, da jeder Aktivität ein Bearbeiter zugeordnet ist. Jedem Bearbeiter können aber mehrere Aktivitäten zugeordnet sein.

Durch ein Organisationsmodell ist es möglich die Ausführungseinheiten zu abstrahieren und je nach Wunsch einzelne Einträge in der Darstellung zusammenzufassen, also beispielsweise zu Organisation, Abteilung, (Projekt-)Team oder (Arbeits-)Gruppe. Um die Darstellung nicht zu überfrachten, bietet das Interface außerhalb der Darstellung, also beispielsweise im Aufgabenbereich (vgl. Abschnitt 7.4) eine Konfigurationsseite, auf der einstellbar ist, welche dieser Unterteilungen angezeigt werden sollen. Dort kann der Benutzer auch wählen, ob die hinterlegten Informationen zur Ausführungseinheit (wo möglich) in konkrete Bearbeiter aufgelöst werden sollen oder ob die Ausführungseinheiten, wie im Prozess modelliert, angezeigt werden sollen, z. B. als Rollenangabe (siehe Sorten von Ausführungseinheiten 2.1). Dies kann zur Folge haben, dass Einträge in der Matrixdimension

'Ausführungseinheit' mit konkreten Bearbeiterangaben und unkonkreten Bearbeiterinformationen, wie z. B. Rollenangaben gemischt erscheinen. Aber auch diese gemischten Einträge lassen sich durch das Organisationsmodell meist problemlos zusammenfassen, wenn es nur eine mögliche Zuordnung gibt. Nur wenn die Bearbeiterinformation sehr unkonkret als Menge von Fertigkeiten oder Attributen erscheint, kann es Fälle geben, die nicht mehr automatisch vom Programm einer bestimmten Organisationseinheit zugeordnet werden können.

Aktivitäten dagegen lassen sich zunächst einmal nicht weiter aggregieren. Nur im Rahmen von hierarchischen Prozessen lassen sich Aktivitäten, die alle zu einem übergeordneten Prozess gehören, zusammenfassen. Um ein reibungsloses Zusammenspiel der verschiedenen Darstellungsformen zu ermöglichen, wäre es für den Benutzer aber sinnvoll, mehrere Aktivitäten, die er in einer Prozessgraphdarstellung zusammengefasst hat, auch nach dem Umschalten zur Matrixdarstellung genauso zusammengefasst wieder vorzufinden. Die Darstellung zusammengefasster Aktivitäten orientiert sich am Beispiel von zu Abteilungen zusammengefassten Ausführungsinformationen (siehe Abbildung 6.21).

Wenn besonders viele Informationen zu einzelnen Aktivitäten oder Bearbeitern gewünscht sind, macht es Sinn, zur Tabellendarstellung (siehe Abschnitt 6.4) zu wechseln und diese, je nach Bedarf, nach Aktivitäten oder nach Bearbeitern zu sortieren. Denn, wie schon im Abschnitt 'Vorteile und Nachteile' beschrieben, bietet die Matrixdarstellung nur wenig Platz zur Anzeige von Details.

Variante 2: Aktivität & Dokument

In dieser Kombination sind die Zusammenhänge ersichtlich, welche Dokumente oder Daten an einer Aktivität beteiligt sind. Auch wird sofort deutlich an wie vielen Aktivitäten ein Dokument beteiligt ist und, bei entsprechender Konfiguration, ob Daten gelesen oder geschrieben werden. Je nach Prozessmodell können aber auch noch andere Arten des Dokumentenzugriffs unterschieden werden, wie Erzeugen, Anfügen oder Löschen. Wenn wie in Abbildung 6.20 eine Farbkodierung dafür verwendet wird, schlüsselt eine Legende (siehe Legende im Aufgabenbereich 7.4) die Farbzusordnungen auf. Zusätzlich geben Tooltips über den Einträgen Auskunft über den jeweiligen Zugriffstyp.

Außer dem Dokumentenzugriff ist für die Einträge der Matrixfelder häufig auch die Bearbeiterinformation interessant, da Dokumente und Bearbeiter sehr nah verwandte Prozessaspekte darstellen.

Da ein Dokument während seiner Erstellung meist an mehreren Aktivitäten beteiligt ist, wird es hier häufig eine im Vergleich zu den anderen Varianten relativ dicht besetzte Matrix ergeben.

Außer Aktivitäten (bei hierarchischen Prozessen) können auch Dokumente hierarchisch strukturiert sein, wenn dazu ein entsprechendes Organisationsmodell durch ein Dokumenten-Management-System oder eine Datenbank gegeben ist. Damit lassen sich dann auch die Dokumente zu Gruppen zusammenfassen, wie in Abbildung 6.20 zu sehen ist.

Variante 3: Aktivität & System

Diese Variante zeigt welche Systeme oder Applikationen von den einzelnen Aktivitäten verwendet werden. Jede Zeile enthält höchstens ein gefülltes Matrixfeld, da jeder Aktivität

(normalerweise) maximal ein System (oder eine Applikation) zugeordnet ist. Jedem System sind allerdings häufig mehrere Aktivitäten zugeordnet.

Mit entsprechenden Informationen aus einem (IT-)Organisationsmodell, lassen sich auch hier die Systeme hierarchisch darstellen. Damit lassen sich Subsysteme zu einem Eintrag zusammenfassen und können somit helfen, eine große Matrix übersichtlicher zu machen.

Variante 4: Ausführungseinheit & Dokument

Diese Variante lässt es zu, sich etwas von den Aktivitäten zu lösen. Hier werden die Dokumente und ihre Bearbeiter gegenübergestellt. Aber auch hier zeigen die Matrixfelder wieder Aktivitätenattribute an, denn in den Matrixfeldern werden wieder einzelne oder mehrere Aktivitäten selektiert. Dies gilt auch für die beiden folgenden Varianten.

Diese Variante dürfte häufig eine kompakte Matrix ergeben, da die Zahl der am Prozess beteiligten Bearbeiter kleiner oder gleich der Anzahl der Aktivitäten ist. Auch die Zahl der Dokumente dürfte selten größer als die Zahl der Aktivitäten sein. Dafür wird ein Dokument während seiner Erstellung meist von mehreren Bearbeitern verändert, ergänzt oder begutachtet, was dazu führt, dass die Matrix, im Vergleich zu den anderen Kombinationen, relativ dicht besetzt sein wird.

Es ist möglich, dass hinter einem Matrixfeld mehr als eine Aktivität steckt, wenn ein und derselbe Bearbeiter mit demselben Dokument in mehreren Aktivitäten arbeitet. Dass ein Matrixfeld mehrere Aktivitäten abdeckt, geschieht aber auch dann, wenn Dokumente und Ausführungseinheiten, entsprechend den Organisationsmodell-Informationen, zusammengefasst werden. [Abbildung 6.21](#) zeigt genau dies. In der Zeile, in der mehrere Bearbeiter zu einer Abteilung zusammengefasst sind, informiert die Zahl in Klammern hinter dem Abteilungsnamen über die Anzahl der Bearbeiter, die hier zusammengefasst wurden.

Variante 5: Ausführungseinheit & System

Ähnlich wie bei der Variante Ausführungseinheit & Dokument, werden hier die in den Aktivitäten verwendeten Systeme (oder Applikationen) und die jeweiligen Bearbeiter gegenübergestellt. Es kann auch vorkommen, dass einzelne Kombinationen von Ausführungseinheit und System in mehreren Aktivitäten gleichermaßen auftreten. Und wie in den Varianten zuvor, lassen sich auch hier einzelne Einträge in beiden Matrixdimensionen zusammenfassen. Also müssen auch hier die Matrixfelder dem Benutzer signalisieren, wenn mehr als eine Aktivität von einem Eintrag zusammengefasst wird.

Variante 6: System & Dokument

Variante 6 ist die letzte fehlende Kombination der Aspekte Aktivität, Ausführungseinheit, Dokument und System. Diese Variante ist eigentlich nicht unbedingt notwendig, denn die genannten vier Elemente können die gleichen Informationen zu Tage fördern. Der Vollständigkeit halber sei diese Variante hier aber aufgeführt. Man kann aus ihr ersehen, an wie vielen Stellen im Prozess ein und dasselbe Dokument in Verbindung mit unterschiedlichen Systemen verarbeitet wird.

Diese Gegenüberstellung dürfte die kleinste Matrix ergeben, sie kann zudem recht dünn besetzt sein, denn hinter einem Eintrag können mehrere Aktivitäten stecken, auf die die Kombination von Dokument und System zutrifft.

Multi-Instanz-Darstellung

Auch die Matrixdarstellung eignet sich dafür, Informationen zu mehreren Instanzen eines Schemas gleichzeitig anzuzeigen. Hierzu werden als neue Matrixdimension die Prozesse herangezogen, die jeweils in der ersten Dimension, anstelle der Aktivitäten (wie bei der Schema- und Instanzdarstellung) dargestellt werden. Diese können nach Produkt/Objekt oder Kunde gruppiert werden. Dies ermöglicht z. B. , gezielt Prozesse in der Matrix zu untersuchen, die alle dem gleichen Kunden zugeordnet sind. Das Interface bietet dem Benutzer alternativ die Option, die Instanzen nach anderen Kriterien zu sortieren, was dann allerdings die Gruppierung z. B. in Produktgruppen verhindert. Sinnvoll ist beispielsweise die Sortierung nach Startzeit der Instanz, um so schnell zu erkennen, ob einzelne Instanzen im Vergleich zu den anderen lange Bearbeitungszeiten aufweisen.

Variante 1: Prozess & Ausführungseinheit

Diese und die folgenden drei Varianten stellen das Äquivalent zur Multi-Instanz Tabellenübersicht dar (siehe Abschnitt 6.4), in der jede Zeile für einen Prozess (oder eine Instanz) steht.

Diese Variante dient dazu, einen Überblick zu erhalten, welche Bearbeiter mit welchen Prozessen beschäftigt sind. Der Benutzer kann so Listen der Instanzen erhalten, mit denen einzelne Bearbeiter zu tun haben. Darüber hinaus können auch in Multi-Instanz-Darstellungen weitere Informationen in den Matrixfeldern dargestellt werden.

Die Matrixfelder selektieren in dieser und in den beiden folgenden Varianten jedoch nicht mehr einzelne oder mehrere Aktivitäten eines Prozesses, sondern immer alle Aktivitäten eines Prozesses gleichzeitig. Daher lassen sich die aktivitätsbezogenen Informationen, wie z. B. Fristen, nur in aggregierter Form darstellen. Allerdings können hier nun die instanzspezifischen Informationen, wie z. B. die Prozessverantwortlichen, angezeigt werden.

Variante 2: Prozess & Dokument

Diese eher spezielle Variante sei der Vollständigkeit halber aufgelistet. Sie macht nur Sinn, wenn der Anwender nach Dokumenten forschen möchte, die in mehreren Prozessen verwendet werden. Ansonsten verhält sich diese Variante analog zur Variante 'Prozess & Ausführungseinheit'.

Variante 3: Prozess & System

Wo der Wunsch besteht, die Prozesse daraufhin zu untersuchen, inwieweit sie auf die gleichen Systeme und Applikationen zugreifen, ist diese Variante interessant. Ansonsten verhält sie sich analog zur Variante 'Prozess & Ausführungseinheit'.

Variante 4: Prozess & Produkt (Objekt)

Die Elemente in der ersten Matrixdimension können nur dann zu Produktgruppen zusammengefasst werden, wenn die dort gelisteten Instanzen nach den ihnen zugeordneten Produkten sortiert werden. Wenn der Benutzer eine andere Sortierung wünscht, bekommt er durch diese Variante die Möglichkeit, die Prozessinformationen dennoch nach Produktgruppen zu gliedern.

Variante 5: Prozess & Aktivität

Bei der Schema- und Instanzdarstellung werden die Aktivitäten nie in der zweiten Matrixdimension angeordnet, diese Darstellung ergänzt die drei vorangegangenen Varianten. Im Gegensatz zu den anderen Multi-Instanz Varianten müssen die dargestellten Aktivitätsattribute hier nicht aggregiert werden, da in dieser Darstellung jede einzelne Aktivität jedes Prozesses genau einem Matrixfeld zugeordnet wird. Auf diese Weise können die Instanzen effektiv miteinander verglichen werden, solange jeweils nur ein einzelner Aspekt Gegenstand der Untersuchung ist. Ein typisches Beispiel wäre die Untersuchung der Kosten.

KONFIGURATIONSMÖGLICHKEITEN

Dieser Abschnitt klärt, welche Konfigurationsmöglichkeiten sich für den Benutzer beim Umgang mit der Matrixdarstellung ergeben. Hierbei werden nur Punkte genannt, die speziell für die Matrixdarstellung gelten. Das (farbige) Markieren von Aktivitäten gilt beispielsweise generell in allen Darstellungsformen (siehe dazu Abschnitt 7.2).

Konfigurationsmöglichkeiten sind:

- Darstellungsvariante wählen
- Matrixfeld-Einträge konfigurieren
- Tooltip-Anzeige für jede Variante extra konfigurieren
- Auswahl von Organisationsmodell-Informationen
- Hierarchische Prozesse berücksichtigen
- Übersetzung von Rollenangaben in Bearbeiter

Darstellungsvariante wählen

Es gibt keine Standard-Darstellungsform für Matrizen, der Benutzer wählt zwischen verschiedenen Schema- und Instanz-Darstellungen oder zwischen Multi-Instanz-Darstellungen

Matrixfeld-Einträge konfigurieren

In jeder Variante steht ein Satz von anzeigbaren Attributen zur Verfügung aus denen der Benutzer eine Teilmenge für die Darstellung auswählen kann. Wird kein Attribut für die Anzeige gewählt, zeigt die Darstellung nur ein Häkchen, statt der ausgewählten Attribute (siehe Abbildung 6.19).

Tooltip Anzeige für jede Variante extra konfigurieren

Anders als bei den anderen Darstellungskonzepten, kann die Tooltip-Anzeige hier nicht global für jede Variante gleichsam konfiguriert werden, sondern für jede Variante separat. Denn bei jeder Darstellungsvariante stehen für den Benutzer andere Prozessaspekte im Mittelpunkt des Interesses und die jeweils naheliegenden Zusatzinformationen, die ein Tooltip anzeigen könnte, ändern sich damit auch.

Auswahl von Organisationsmodell-Informationen

Der Benutzer kann wählen, in welchem Umfang die Daten aus dem Organisationsmodell für die Darstellung herangezogen werden. Bei einer Darstellung mit der Ausführungseinheit als Matrixdimension kann beispielsweise eine Auswahl erfolgen, welche Firmenstrukturen, also z. B. Team, Projekt, (Arbeits-)Gruppe, Abteilung, Unternehmen und Organisation mit in die Strukturierungshierarchie der Matrixdimension aufgenommen werden. Diese Auswahl muss auf Konsistenz überprüft werden, da Arbeitsgruppen beispielsweise abteilungsübergreifend zusammengesetzt sein können, was eine Baumstruktur bei gleichzeitiger Anzeige der Abteilungen verhindert.

Hierarchische Prozesse berücksichtigen

Standardeinstellung für die Matrixdarstellung ist, dass nur die aktuelle Prozessinstanz (oder Schema) ausgewertet wird und Sprünge in andere Teilprozesse oder Unterprozesse nicht verfolgt werden. Bei Bedarf ist es möglich, Unterprozesse bis zu einer vom Benutzer gewählten Tiefe mit zu berücksichtigen, was aber die Komplexität der Darstellung erhöht.

Übersetzung von Rollenangaben in Bearbeiter

Bei der Instanzdarstellung kann der Benutzer wählen, ob das Programm – soweit möglich – konkrete Bearbeiter anzeigt, anstatt der auf Schemaebene getroffenen Rollenangaben. Analog dazu, ist auch die Rückrichtung möglich, also die Übersetzung von Bearbeiter- in Rollenangaben.

6.7.1. Notwendige Voraussetzungen

Zusammenfaltbare Matrixdimensionen sind nur mit entsprechenden Zusatzinformationen über das Organisationsmodell möglich (siehe Abschnitt 2.1.3).

6.7.2. Implementierungsaspekte

Eine Matrixdarstellung kann leicht mehr als eine Bildschirmseite benötigen. Daher wird das Scrolling so implementiert, dass dem Benutzer beim Betrachten des Inhaltes ständig die Spalten- und Zeilenbeschriftungen angezeigt werden. Wenn Informationen aus dem Organisationsmodell dazu herangezogen werden, um die Einträge in den Matrixdimensionen in eine hierarchische Ordnung zu bringen, die auseinander- und zusammengefasst werden kann, dann verbrauchen die Spalten- und Zeilenbeschriftungen selbst reichlich Platz. Daher wird auch von diesen immer nur ein scrollbarer Ausschnitt gezeigt, um dem eigentlichen Inhalt der Darstellung nicht zu viel Bildschirmplatz zu nehmen.

6.7.3. Bedingungen im Kontrollfluss

An einigen Stellen wurde bei der Vorstellung der Matrixdarstellung schon deutlich, dass Bedingungen im Kontrollfluss Probleme mit sich bringen. Der Grund dafür ist, dass der Kontrollfluss bei dieser Darstellungsform konzeptbedingt nicht dargestellt werden kann. Dies hat Konsequenzen für die Visualisierung.

Die Grundidee der Matrixdarstellung ist, dass Zusammenhänge aus der Gesamtmenge aller Aktivitäten eines Prozesses dargestellt werden sollen. Die Matrixdarstellung orientiert sich dafür nicht am Kontrollfluss. Wenn alle Matrixeinträge optisch identisch dargestellt würden, würde dies suggerieren, dass alle Aktivitäten während des Prozessverlaufs ausgeführt werden. Die Einträge einer Matrix lassen jedoch keinerlei Rückschlüsse darüber zu, wie wahrscheinlich es ist, dass eine mit einem Matrixfeld assoziierte Aktivität ausgeführt wird. Daher ist es notwendig, dem Benutzer Matrixeinträge, die nicht zwingend zutreffen, zu signalisieren. Dies betrifft nicht nur Aktivitäten hinter Oder-Verzweigungen, sondern auch Aktivitäten die aufgrund von Ausnahmebehandlungen ausgeführt werden. Nur bei bereits abgelaufenen Instanzen und Prozessen mit deterministischer Aktivitätenabfolge sind von dem Problem nicht betroffen. Jedoch sollten, wenn schon Laufzeitdaten (auch für Teilbereiche) vorliegen, Einträge von Aktivitäten in abgewählten Verzweigungspfaden nicht entfernt werden, sondern beispielsweise grau dargestellt werden.

Matrixfelder mit bedingten Aktivitäten werden daher farbig markiert. Ideal ist es, wenn der dargestellte Prozess automatisch analysiert werden kann und die Farbhelligkeit die jeweilige Ausführungswahrscheinlichkeit widerspiegelt. Aktivitäten deren Ausführung an zwei Bedingungen geknüpft sind, können so etwa dunkler dargestellt werden. Dies lässt sich noch ausbauen, indem Historiendaten (engl. *audit trails*) vergangener Prozessabläufe statistisch ausgewertet werden.

Erst durch die Hinweise auf Kontrollflussbedingungen, können Anwender die von einem Prozess genutzten Ressourcen richtig auswerten. In einer Matrix, die beispielsweise Dokumente und Bearbeiter gegenüberstellt, wird dadurch sichtbar, welche Dokument-bearbeitenden Aktivitäten nur bedingt ausgeführt werden.

6.8. Zusammenfassung

Das Kapitel stellt die Darstellungskonzepte Prozessgraph, Swimlane, Kalender, Tabelle, Interaktionsdiagramm, Datenflussdiagramm und Matrix vor und diskutiert deren spezifische Vor- und Nachteile, Variationen der Darstellung und die Grenzen ihrer Einsetzbarkeit.

Nach der Vorstellung der Darstellungskonzepte ist es interessant zu untersuchen, inwieweit sich diese als Werkzeuge zur Bearbeitung der Fragestellungen der Benutzer eignen. Dafür werden sie ebenso wie schon die Benutzer in das in Kapitel 4 vorgestellte Schema (siehe Abbildung 3.5) eingeordnet. Das Ergebnis zeigt Tabelle 6.3.

Damit lassen sich nun den Fragestellungen aus dem Abschnitt 3.2 ('Ziele nach Benutzergruppen') Werkzeuge zuordnen, die geeignet sind, um die Aufgaben zu erfüllen. Dies zeigen Tabelle 6.4 und Tabelle 6.5.

Konzept	Ziel	Aufgabentyp			Detailgrad		
		Entwerfen ^a	Entscheiden	Informieren	Übersicht	Zusammenhang	Details
Prozessgraphdarstellung	Kontrollfluss	✓	✓	✓	✓		
Swimlane-Darstellung	Zusammenhänge innerhalb eines Prozessaspektes		✓	✓		✓	
Kalenderdarstellung	Zeitliche Aspekte aus den Prozessinformationen		✓	✓		✓	
Tabellendarstellung	Zusammenhänge innerhalb eines Prozessaspektes		✓	✓		✓	✓
Interaktionsdiagramm	Interaktionen zwischen verschiedenen Organisationseinheiten		✓	✓		✓	
Datenflussdiagramm	Dokument-/Datenfluss		✓	✓		✓	
Matrixdarstellung	Zusammenhänge zwischen zwei Prozessaspekten		✓	✓		✓	✓

^aSiehe Abbildung 3.5

Tabelle 6.3.: Darstellungsarten mit Darstellungszielen und Einordnung der Werkzeuge nach Benutzerzielen

Aufgabe (Fragestellung)	Werkzeug
Bearbeiter/Anwender	
Liste aktueller Aufgaben	Tabelle (Arbeitsliste)
Welche Aufgabe hat Priorität?	Tabelle (Arbeitsliste)
Wie ordnet sich eine Aufgabe ins Umfeld ein?	Prozessgraph, Swimlane
Wer hat außer mir mit dem Prozess zu tun?	Matrix, Swimlane, Tabelle, Interaktionsdiagramm
Welche Aufgaben stehen in naher Zukunft an?	Tabelle (Arbeitsliste), Prozessgraph
Was ist schon alles erledigt? (Motivation)	Tabelle (Arbeitsliste), Prozessgraph
Prozessverantwortlicher/Abteilungsleiter	
View-Bildung	Swimlane, Prozessgraph
Anpassung des Prozessschemas	Prozessgraph, Swimlane
Zeit-Kontrolle (Plan/Ist Vergleich)	Kalender
Pufferzeiten bestimmen	Kalender
Was hat sich in der letzten Zeit getan?	Kalender (Instanz & Multi-Instanz Varianten), Prozessgraph
Welche Meilensteine wurden erreicht?	Kalender (Instanz & Multi-Instanz Varianten), Tabelle (Instanz & Multi-Instanz Übersicht)
Überblick gewinnen	Prozessgraph, Swimlane, Kalender
Wie ist der aktuelle Stand?	Kalender (Instanz & Multi-Instanz Varianten), Tabelle (Instanz & Multi-Instanz Übersicht), Prozessgraph
Wie sieht der Daten-/Dokumentfluss aus?	Datenflussdiagramm, Swimlane, Prozessgraph
Wer kommuniziert mit wem?	Interaktionsdiagramm, Swimlane
Können alle Fristen eingehalten werden?	Kalender
Übersicht über Mitarbeiterauslastung	Kalender
Überblick über Kosten	Tabelle, Matrix
Management	
Status aller Instanzen nach Meilensteinen / nach Fristen / in Prozent	Kalender (Multi-Instanz Varianten 2 & 3), Tabelle (Multi-Instanz Übersicht)
Übersicht über Instanzen: Gesamtbearbeitungsdauer (Durchsatz)	Tabelle (Multi-Instanz Übersicht)
Übersicht über Mitarbeiterauslastung	Kalender (Ressourcen-orientierte Multi-Instanz Variante)
Überblick über Kosten	Tabelle, Matrix
Können alle Fristen eingehalten werden?	Kalender (Multi-Instanz Varianten 2 & 3)
Zeit-Kontrolle (Plan/Ist Vergleich)	Kalender
Übersicht über Instanzen: Bearbeiterzuordnung zu Aktivitäten	Matrix, Kalender (Ressourcen-orientierte Multi-Instanz Variante)

Tabelle 6.4.: Mögliche Werkzeuge zum Lösen der Aufgaben der Benutzergruppen – Teil 1 (Bearbeiter/Anwender, Prozessverantwortlicher/Abteilungsleiter, Management)

Aufgabe (Fragestellung)	Werkzeug
IT	
Welche Daten fallen im Laufe eines Prozesses an?	Datenflussdiagramm, Matrix, Tabelle, Swimlane, Prozessgraph
Welche Anwendungen sind an welchen Prozessen beteiligt?	Matrix
Zu welchen Personen wandern die Daten?	Datenflussdiagramm, Matrix
Zu welchen Systemen wandern die Daten?	Datenflussdiagramm (mit eingblendeten Systemen), Matrix
Externer Mitarbeiter/Kunde	
Status aller Instanzen nach Meilenstein	Kalenderdarstellung (Instanz & Multi-Instanz Varianten)
Überblick über wichtige Abschnitte des Prozesses (geeignete View)	Prozessgraph, Swimlane
Übersicht über die Kommunikation mit der Firma	Interaktionsdiagramm, Swimlane
Übersicht über eigene Prozessbeteiligung	Prozessgraph, Interaktionsdiagramm
Prozessmodellierer	
WF-Schema entwerfen	Prozessgraph, Swimlane, Kalender
View-Bildung	Swimlane, Prozessgraph
Fristen und typische Ausführungsdauer festlegen	Kalender
Schemaänderungen durchführen	Prozessgraph, Swimlane

Tabelle 6.5.: Mögliche Werkzeuge zum Lösen der Aufgaben der Benutzergruppen – Teil 2 (IT, Externer Mitarbeiter/Kunde, Prozessmodellierer)

7

Interaktionen auf Prozessinformationen

Dieses Kapitel ergänzt die beiden vorangegangenen Kapitel, welche den verschiedenen Konzepten der Prozessvisualisierung gewidmet sind, mit den Interaktionsmöglichkeiten, die sich aus der Bereitstellung dieser Konzepte für die Benutzer einer Prozessvisualisierungskomponente ergeben.

Die verschiedenen Darstellungsmöglichkeiten sind Werkzeuge (siehe Abbildung 3.1 → S. 23), die von einer Software zur Nutzung angeboten werden. Erst wenn die Werkzeuge sich an die Bedürfnisse der Benutzer anpassen lassen und sie damit in die Lage versetzen ihre Ziele zu erreichen, erzeugen sie einen Mehrwert. Benutzerinteraktionen betreffen aber nicht nur die Anpassung von Werkzeugen, sondern auch die Integration dieser Werkzeuge in eine Benutzeroberfläche. In diesem Kapitel werden verschiedene Aspekte der Benutzeroberfläche beschrieben, die mit der Nutzung der Darstellungskonzepte im Zusammenhang stehen. Dies sind z. B. Navigationsaspekte und die Prozesshistorie.

Benutzerinteraktionen, auf den Darstellungen, betreffen zum einen die Konfigurationsmöglichkeiten der grafischen Darstellung, zum anderen Hilfestellungen, um gesuchte Informationen zu finden. Hinzu kommen Möglichkeiten für das Ein- und Ausblenden von Teilen der Prozessinformationen (z. B. Ausblenden des Datenflusses). Diese Interaktion ist aber letztlich eine Operation auf Views.

Views, Bildung von Views und Operationen auf Views werden im folgenden Abschnitt diskutiert.

7.1. View-Bildung

Eine View (oder auch Sicht) ist eine Abbildung eines Prozesses auf eine Untermenge der vorliegenden Informationen. Eine View kann darüber hinaus um Informationen aus einem Organisationsmodell angereichert werden. Darstellungen, die auf Views basieren, lassen sich den Bedürfnissen der Nutzer anpassen.

Views stellen einen mächtigen Aspekt der Prozessvisualisierung dar. Wichtig hierbei ist, dass sie unabhängig sind, von den bisher besprochenen Aspekten der Visualisierung: Kapitel 5 behandelte den Aspekt der grafischen Repräsentation, Kapitel 6 die Darstellungsformen also die strukturelle Repräsentation. Wichtigste Eigenschaft des View-Konzeptes ist, dass Views auf alle Darstellungsformen gleich wirken. Die Konzepte View, Grafik und Darstellung stehen also für die Visualisierung orthogonal zueinander. Die einzige Einschränkung besteht darin, dass nicht alle Darstellungsformen die gleichen Prozessaspekte visualisieren können.

Sinnvollerweise arbeiten Visualisierungskomponenten für ein PMS von vornherein nur auf Views. Dadurch kann auch gesteuert werden, welche Prozessinformationen bestimmte Benutzer betrachten können. Außer diesen Sicherheitsaspekten (siehe Abschnitt 8.1) haben View-Konzepte noch ganz praktische Vorteile:

Views können, analog zu den Views auf Tabellen in Datenbanksystemen, kombinierte Ansichten von mehreren Prozessen erzeugen. Beispielsweise können bei hierarchischen Prozessen einzelne Aktivitäten ganze Subprozesse kapseln. Kapselt ein Prozess mit seinen Aktivitäten viele Subprozesse, kann es sinnvoll sein, eine View zu erzeugen, die einen virtuellen Prozess darstellt, in dem alle Subprozesse aus dem übergeordneten Prozess expandiert dargestellt werden. Eine solche View würde dann nur die Aktivitäten in der zweiten Ebene aus Abbildung 6.16 enthalten.

Den Ablauf der Prozessvisualisierung mit einem solchen Konzept zeigt Abbildung 7.1. Das Bild zeigt die beiden wesentlichen Operationen, die zur View-Bildung durchgeführt werden können, Aggregation und Reduktion. Weitergehende Untersuchungen zu Views und mögliche Operationen zur Bildung von Views auf Prozessdaten werden in [Klot04] diskutiert.

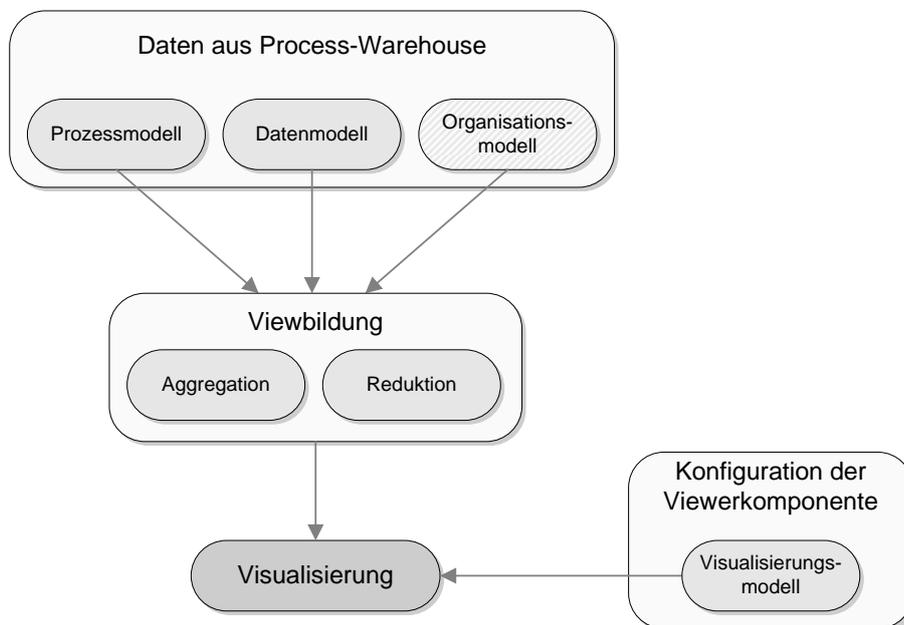


Abbildung 7.1.: Ablauf Prozessvisualisierung mit Views

Aggregation und Reduktion dienen in erster Linie dazu, die Komplexität einer Darstellung zu vermindern, sie können aber auch dazu verwendet werden Informationen zu verbergen.

7.1.1. Aggregation

Mit Aggregation ist im Prozesskontext in erster Linie Graphaggregation gemeint, also das Zusammenfassen von zusammenhängenden Teilgraphen zu einer virtuellen Ersatzaktivität, die die damit ersetzten Aktivitäten zur Förderung der Übersichtlichkeit bündelt.

Arten der Aggregation:

- > Zusammenfassen einer Abfolge von Aktivitäten zu einer Aktivität
- > Zusammenfassen mehrerer Prozess-/Aktivitäts-Attribute
- > Gruppierung von Ressourcen eines Typs unter eine Obermenge

Aggregation beschränkt sich also nicht auf das Zusammenfassen von Aktivitäten. Im Allgemeinen steht Aggregation für das Verdichten von Informationen, der Begriff ist wie beim *Data-Warehousing* zu verstehen. Dort gibt es Fakten und darauf angewendete Aggregatfunktionen, um Fakten zusammenzufassen. Es ist jedoch zu beachten, dass nicht alle Fakten aggregierbar sind. Es ist hierfür vor allem die Typverträglichkeit von Fakt und Aggregatfunktion wichtig. Die Fakten von Prozessinformationen sind in erster Linie die Aktivitäten mit ihren Eigenschaften (siehe Tabelle 2.1). Es lassen sich beispielsweise nur Fakten, die alle vom Typ Kosten sind, zu einer Gesamtsumme aggregieren. In der Darstellung muss dann deutlich erkennbar sein, wo solche aggregierten Daten angezeigt werden.

Im Rahmen einer View kann einzelnen Fakten, wie z. B. den eben genannten Kosten, eine Standardaggregatfunktion zugewiesen werden. Der Aufgabenbereich (siehe Abschnitt 7.4) bietet dazu ein Fenster, in dem der Benutzer die Zuordnungen von Fakten und Aggregatfunktion ändern kann und auch sonstige Einstellungen zur gegenwärtig angezeigten View vornehmen kann.

Mögliche Aggregatfunktionen sind z. B. Summe, Anzahl, Maximum, Minimum oder Mittelwert, aber auch statistische Funktionen wie Standardabweichung oder Besonderheiten wie eine Top-N Angabe. Nicht jede Funktion macht in jedem Zusammenhang Sinn, trotzdem kann die Wahl der Aggregation dem Benutzer überlassen werden. Es könnten aber auch in einer eingeschränkten View bereits aggregierte Daten vordefiniert sein.

Andere Fakten, z. B. Informationen zu Ausführungseinheiten, lassen sich über Mengenaufbildung zusammenfassen. Wenn z. B. zwei Aktivitäten mit den Bearbeitern A und B zusammengefasst werden, trägt die neue stellvertretende Aktivität die Information zur Ausführungseinheit 'Bearbeiter A, B'. Durch Informationen aus dem Organisationsmodell lässt sich dafür eventuell auch eine Darstellung finden wie 'Abteilung X (2)' – zwei Bearbeiter aus Abteilung X.

Bei Fakten mit endlichem Wertebereich, z. B. der Aktivitätenzustand, lassen sich auch Ordnungen finden. Sollen z. B. Aktivitäten zusammengefasst werden, welche beispielsweise die Zustände 'beendet' und 'laufend' haben, wird der Wert des aggregierten Fakts auf 'laufend' gesetzt. Nur, wenn alle aggregierten Aktivitäten den Zustand 'beendet' haben, kann dies der gemeinsame Zustand sein (vergleiche [Klot04, S.27]).

Eine Besonderheit stellt das Zusammenfassen von Prozessinformationen unter Einbeziehung eines Organisationsmodells dar. Es können beispielsweise zusammengehörige Informationen logisch gruppiert werden. Beispielsweise können mehrere Bearbeiter zu einer oder mehreren Abteilungen gruppiert/aggregiert werden, was der Übersichtlichkeit der Darstellung zugute kommt. Detailinformationen, die die ursprünglichen (nicht zu Abteilungen gruppierten) Bearbeiter auflisten, können an anderer Stelle der Oberfläche dargestellt werden (siehe dazu Abschnitt 7.4).

Auch die Informationen zum Ausführungszustand können vom Anwender zur Aggregation herangezogen werden. Bei einer laufenden Prozessinstanz können schon beendete Aktivitäten oder noch weit in der Zukunft liegende nicht aktivierte Aktivitäten zusammengefasst werden, sodass eine Übersicht über den aktuell anstehenden Prozessbereich entsteht.

7.1.2. Reduktion

Bei der Graphreduktion werden Informationsaspekte entfernt, dies können einzelne Aktivitäten oder ganze Teilgraphen sein oder auch Aktivitätenattribute, die für die Erledigung der Aufgabe ohnehin keine Rolle spielen.

Die Auswahl auszuschneidender Aktivitäten erfolgt meist anhand von:

- Aktivitätsattributen
- Zeitausschnitten
- Rollen (Ausführungseinheiten)
- Dokumente/Daten
- Systeme/Anwendungen

Mögliche Anwendungen wären beispielsweise: Nur Aktivitäten darzustellen, die einer bestimmten Rolle zugeordnet sind. Aktivitäten entfernen, deren Bearbeiter zu einer bestimmten Abteilung gehören. Teilgraphen verbergen, die notwendige Schritte im Falle eines Fehlers enthalten. Manche Attribute wie z. B. Kosten oder Dauer sind sensible Informationen, die durch eine View ausgeblendet werden können.

Die Reduktion kann auch sehr gut als Filterfunktion beispielsweise in großen Tabellen(darstellungen) verwendet werden.

7.2. Aktivitäten markieren

Meist zeigt eine Darstellung viel zu viele Informationen auf einmal an und es werden alle Elemente gleich stark betont. Aber in der Praxis sind dem Benutzer einige Elemente wichtiger als andere [CR03]. Dem Benutzer muss daher ein Werkzeug an die Hand gegeben werden, mit dem er festlegen kann, welche Elemente in der Darstellung besonders hervorgehoben oder markiert bzw. mit mehr Details dargestellt werden sollen. Üblicherweise geschieht das Hervorheben meist durch Farbe und Form (siehe Abschnitt 5.3.1).

Diese Funktion dient dazu, bestimmte Aktivitäten in den Darstellungen zu markieren. Dies erfolgt mehr im Sinne von 'Hervorheben' und nicht 'Selektieren'. Dies ist nützlich,

um in großen Prozessen schnell einen Überblick zu gewinnen, indem die für die aktuelle Aufgabe relevanten Aktivitäten hervorgehoben dargestellt werden.

Wie aber erfolgt nun die Auswahl der zu markierenden Informationen? Für das System erfolgt eine solche Auswahl ganz allgemein über Regeln – über Ausdrücke, die unter bestimmten Bedingungen wahr werden. Hierfür werden die verfügbaren Einzelemente der Prozessinformationen mit Booleschen Funktionen zu Ausdrücken verknüpft (siehe Tabelle 7.1).

Sinnvolle, häufig gebrauchte Auswahlkriterien für das Markieren von Aktivitäten stellt das Interface dem Benutzer zum sofortigen Zugriff jederzeit bereit. Andere speziellere Regelsätze kann der Benutzer in einem Regelbaukasten erstellen.

7.2.1. Wichtige einfache Markierungsregeln

Alle folgenden Regeln beziehen sich immer auf eine bereits gewählte Ausgangsaktivität, da diese Regeln Beziehungsaspekte zwischen Aktivitäten beschreiben. Häufig gebrauchte naheliegende Markierungsfunktionen sind:

- Vorgängeraktivität(en)
- Nachfolgeraktivität(en)
- Aktivitäten in alternativen Ausführungspfaden

Nicht in jedem Kontext der verschiedenen Darstellungsformen machen alle Punkte dieser Liste Sinn. In der Prozessgraphdarstellung ist die Bestimmung der Vorgängeraktivität trivial, nicht jedoch in einer Tabellen- oder Matrixdarstellung.

Die Funktion 'Aktivitäten in alternativen Ausführungspfaden' markiert alle Aktivitäten, die sich in den anderen Zweigen einer Oder-Verzweigung befinden; d. h. es werden Aktivitäten markiert, die sich relativ zur gewählten Aktivität in alternativen Ausführungspfaden befinden.

7.2.2. Regelbaukastensystem

Die Regeln, die die Markierungsfunktion steuern, müssen dem Programm bekannt gemacht werden. Eine benutzerfreundliche Lösung für einen Regelbaukasten zeigt Abbildung 7.2. Dort dienen die Regeln zur Erstellung einer Abspielliste. Weil der Benutzbarkeit hier mehr Priorität eingeräumt wird als der Funktionsvielfalt, können hier allerdings keine *Und* & *Oder* Regeln miteinander kombiniert werden.

Einfacher, aber weniger mächtig, wird das Markieren über ein Kontextmenü, das je nach Situation sinnvolle Möglichkeiten anbietet. Befindet sich der Mauszeiger beispielsweise in einer Graphdarstellung über einer Aktivität, der die Rolle Kundenbetreuer zugeordnet ist, erscheint im Kontextmenü '*Markiere Aktivitäten von Kundenbetreuer*' zur Auswahl. Durch die Auswahl dieser Option werden alle Aktivitäten in der Darstellung, auf die die Regel Rolle=Kundenbetreuer zutrifft, optisch hervorgehoben.

Eine Hervorhebung kann optisch auch signalisiert werden, indem alles andere, was nicht markiert werden soll, mehr in den Hintergrund gerückt wird. Apple zeigt dies wiederum eindrucksvoll in MacOS X (siehe Abbildung 7.3).



Abbildung 7.2.: Selektionsdialog in Apples iTunes



Abbildung 7.3.: Spotlight im MacOS X System-Einstellungs-Dialog

Alle gerade aktiven Regeln sollte das Programm anzeigen können. Neue Regeln können aufgenommen werden, Bestehende verändert oder gelöscht werden. Jeder Regel ist eine bestimmte Markierungsart zugeordnet. Nur, wenn diese Markierungsarten sich kombinieren lassen (z. B. Linienstärke und Farbe, vgl. 5.3.1), kann ein einzelnes Element auch durch verschiedene Regeln mehrfach markiert werden, ansonsten kommt nur die Markierung nach der ersten Regel zum Zuge. Abbildung 7.4 zeigt alle Aktivitäten mit der Rolle 'Kundenbetreuer' mit einem fetten Rahmen und zusätzlich bei allen Aktivitäten, die auf

das Dokument 'Kostenvoranschlag' zugreifen, das entsprechende Element farbig markiert. Durch die unterschiedlichen Markierungsarten lassen sich mehrere Regeln auf eine Aktivität anwenden.



Abbildung 7.4.: Beispiel für Markierungsregeln

Tabelle 7.1 zeigt beispielhaft mögliche Markierungsregeln. In Abschnitt 5.3.1 werden mögliche Markierungen für Elemente aufgelistet.

Markierungsregel	Beschreibung
Bearbeiter = Herr Maier	Einfache Regel. Alle Aktivitäten, die einer bestimmten Person zugeordnet sind
Bearbeiter = Prozessverantwortlicher	Regel mit Variable. Alle Aktivitäten, die der Prozessverantwortliche selbst durchführt
Rolle = Kundenbetreuer	Alle Aktivitäten, denen ein Kundenbetreuer zugeordnet ist
Abteilung = Vertrieb	Alle Aktivitäten, die dem Vertrieb zugeordnet sind
Dokument = Kostenvoranschlag	Alle Aktivitäten, die auf das Dokument Kostenvoranschlag zugreifen
Dokument.lesend = Kostenvoranschlag	Objektorientierte Regel. Alle Aktivitäten, die auf das Dokument Kostenvoranschlag lesend zugreifen
Kosten >= 500	Alle Aktivitäten, deren Kosten 500 EUR übersteigen
Endzeit > Datum - 3 Tage	Alle Aktivitäten, die in spätestens drei Tagen abgeschlossen sein müssen
Frist <= 1 Tag	Alle Aktivitäten, für deren Ausführung eine Frist von maximal einem Tag vorgesehen ist
Abteilung = Abteilung(Prozessverantwortlicher) UND Nächste Aktivität. Abteilung != Abteilung(Prozessverantwortlicher)	Markierung unter Anwendung mehrerer Regeln. Alle Aktivitäten, die eine Folgeaktivität einer anderen Abteilung haben
Kategorie = Konzeptionelle Tätigkeit	Alle Aktivitäten markieren, die semantisch als Konzeptionelle Tätigkeit klassifiziert sind. Andere Tätigkeitstypen könnten gleichzeitig auch markiert werden (in anderer Farbe)
identisch(System) && überlappend(Zeit)	Alle Aktivitäten markieren, die zum gleichen Zeitpunkt auf das gleiche System zugreifen
Abteilungen farbkodieren	Alle Aktivitäten des Prozesses werden (jeweils unterschiedlich) farblich markiert
Prozesse farbkodieren	Bei hierarchischen Prozessen: Die Aktivitäten werden farbkodiert, je nachdem, zu welchem (Unter-)prozess sie gehören

Tabelle 7.1.: Beispiele für die Markierungsfunktion

Auswahlmenüs für häufig benutzte Regeln und für die im aktuellen Kontext zur Wahl stehenden Parameter können den Benutzer beim Erstellen der Regeln unterstützen. Je einfacher die Verwendung dieser Funktion, desto eher wird sie verwendet. Einige Markierungen könnten auch als Standardmarkierung für bestimmte Darstellungsformen z. B. Prozessgraph eingestellt werden, so beispielsweise die farbliche Hinterlegung der Aktivitäten entsprechend des Aktivitätentyps.

Weiter soll auf die regelbasierte Markierung von Aktivitäten nicht eingegangen werden, sie stellt nur einen Randaspekt im Rahmen der Prozessvisualisierung dar.

7.3. Detailansicht & Tooltips

Es ist nicht sinnvoll alle verfügbaren Prozessdetails im Hauptfenster anzuzeigen. Um eine Darstellung übersichtlich zu halten, werden dort nur die notwendigsten Informationen visualisiert:

- Alle Informationen, die notwendig sind, damit sich der Benutzer in der Darstellung zurechtfindet
- Die eigentlich gesuchte Information

Alle weiteren Details, die für den Benutzer weiterhin noch von Bedeutung sind, werden in zusätzlichen Informationsfenstern untergebracht. Hierfür selektiert der Benutzer eine Aktivität (also einen Knoten, eine Zeile oder Zelle – je nach gewählter Darstellung) woraufhin Detailinformationen über die Aktivität in diesem Fenster angezeigt werden. Wenn keine Aktivität ausgewählt ist, zeigt dieses Fenster allgemeine Informationen über den Prozess, also beispielsweise instanzbasierte Informationen, wie den Prozessverantwortlichen oder Start- und Endzeiten der Prozessinstanz (siehe Tabelle 2.1).

Um dem Benutzer sehr schnell zusätzliche Informationen zu einzelnen Elementen der Darstellung zugänglich zu machen, stellen Tooltips eine geeignete Möglichkeit dar. Sie können außer Detailinformationen zu Aktivitäten auch weitergehende Informationen zu Dokumenten, Systemen, Organisationsmodell oder zu den Historiendaten anzeigen.

7.4. Aufgabenbereich

Mit dem Aufgabenbereich ist eine Werkzeugleiste gemeint, in der Eclipse Rich Client Platform [Ecli06] werden solche Zusatzfenster 'View' genannt.

Bildschirmplatz ist bei komplexer Software Mangelware. Eine Anwendung die Prozesse visualisieren kann, braucht außer der eigentlichen Darstellung, ein Übersichtsfenster, eine Historienansicht, eine Detailansicht und Möglichkeiten zum schnellen Ändern von Konfigurationsparametern. Viele der genannten Ansichten brauchen aber nicht gleichzeitig auf dem Schirm dargestellt zu werden, sie können sich häufig auch eine umschaltbare Werkzeugleiste teilen. Daher bietet sich eine Oberfläche an, die schnell an das jeweilige Bedürfnis des Benutzers angepasst werden kann.

Ein Beispiel für eine gute Lösung, um eine aufgeräumte, Platz sparende Benutzeroberfläche zu schaffen, findet sich z. B. bei Microsofts PowerPoint. Abbildung 7.5 zeigt, wie

dort eine seitliche Leiste eingeblendet werden kann, über die viele verschiedene Folien-Layouts schnell übernommen werden können. Solche seitlichen Funktionsleisten werden meist Aufgabenbereich genannt. Die Abbildung zeigt auch, dass dieser Aufgabenbereich viele verschiedene Aufgaben erfüllt, zwischen denen umgeschaltet werden kann. In einer direkten Umsetzung könnte in den graphbasierten Darstellungsformen Graph und Swimlane das Aussehen der Aktivitätenknoten schnell geändert werden.

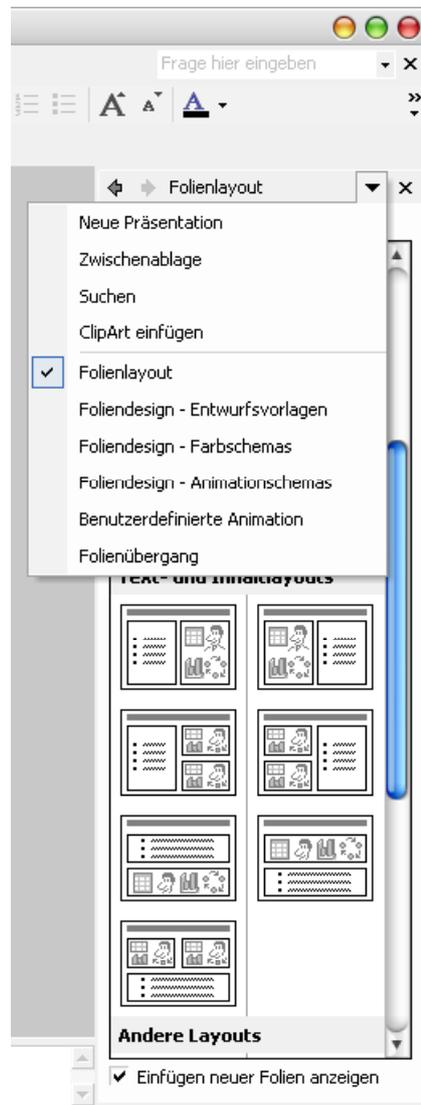


Abbildung 7.5.: Aufgabenbereich in Microsofts Powerpoint

Ein Aufgabenfenster kann also beispielsweise die (Darstellungs-)Konfiguration für die momentan aktive Darstellungsform enthalten. Je nach gewählter Darstellung gibt es unterschiedlichen Konfigurationsbedarf. In einer Graphdarstellung gibt es z. B. mehr Einstellungsmöglichkeiten als für die Matrixdarstellung. Dort könnten die verschiedenen zu konfigurierenden Einstellungen auch auf mehrere Aufgabenfenster verteilt werden.

Wo der Platz einer seitlichen Funktionsliste nicht ausreicht, kann auch auf andere Lösungen wie einen modalen Einstellungsdialog zurückgegriffen werden. Es bietet sich beispielsweise an, die Einstellungen, welche Prozessinformationen direkt in der Darstellung angezeigt werden, welche in einer Detailansicht und welche als Tooltipp-Information, in einem gemeinsamen Konfigurationsdialog, wie ihn Abbildung 7.6 zeigt, unterzubringen. Da diese drei Einstellungen nicht unabhängig voneinander sind.

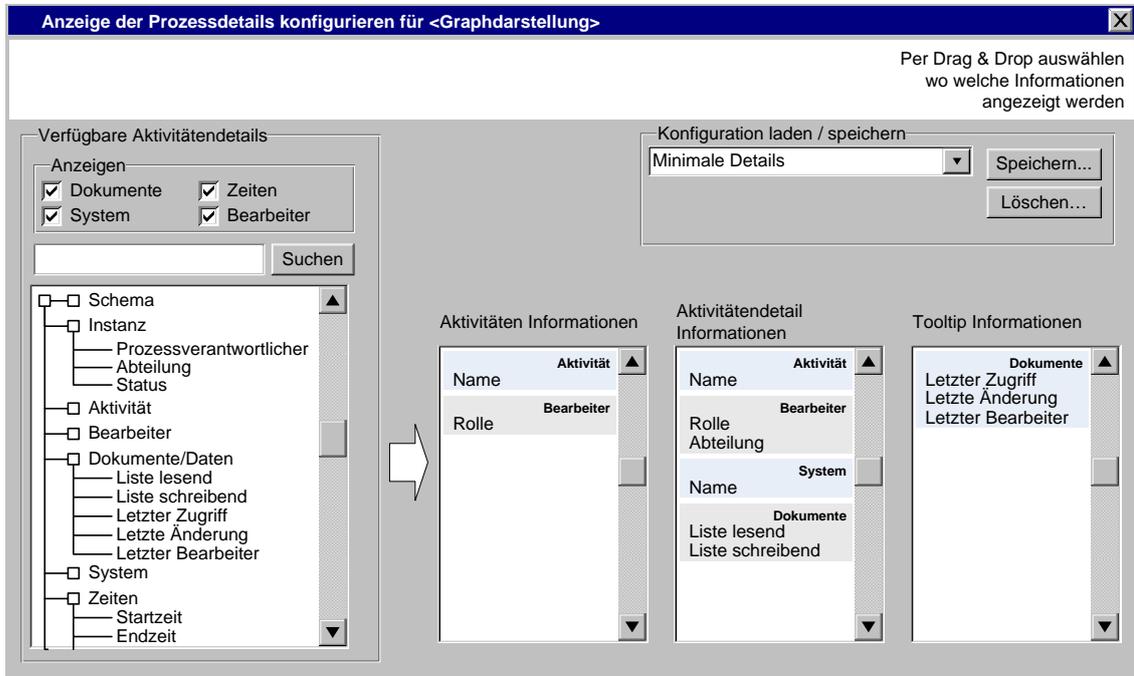


Abbildung 7.6.: Konfiguration der Anzeige der Prozessdetails

Die Tabelle 7.2 zeigt eine Übersicht, welche Aufgaben ein Aufgabenbereich erfüllen könnte.

Navigation

Der Aufgabenbereich Navigation, wird nur in einer Umgebung mit Zugriff auf eine Geschäftsprozesshierarchie benötigt, denn dann benötigt der Benutzer umfassendere Navigationsmöglichkeiten. Hier besteht die Möglichkeit in der direkten Umgebung des aktuellen Prozesses zu navigieren, alle Prozesse, mit denen der Aktuelle in irgendeiner Verbindung steht, sind direkt anwählbar, also Elternprozess, Kindprozesse und andere Prozesse an beliebigen Stellen in der Prozesshierarchie, zu denen ein Prozess Verknüpfungen im Kontrollfluss aufweist. Abbildung 7.7 zeigt die Prozesshierarchie ausgehend vom Prozess 'Änderungen durchführen', hier wird deutlich, dass hier in vertikaler Richtung im Prozessbaum hoch- und runternavigiert werden kann und in horizontaler Richtung links und rechts zu benachbarten Prozessen.

Details in Aktivitäten / Tooltipp / Detailfenster

Die Einstellung welche Aktivitätendetails direkt an der Aktivität in Tooltips oder im Detailfenster angezeigt werden, kann hier im Aufgabenbereich erfolgen, oder z. B. gemeinsam

Aufgabe	Beschreibung
Navigation	Navigation in hierarchischen Workflows. Vertikale Links zu Elternprozess und Kindprozessen, horizontale Links zu Vorgänger- und Nachfolgerprozess und Verknüpfungen zu anderen Prozessen
Detailinformation	Zeigt detailliertere Informationen über eine selektierte Aktivität ^a
Darstellungskonfiguration	Einstellungen von Farben, Formen, Komplexität (Datenkanten, Schleifen, etc.)
Details in Aktivitäten/ Tooltip/Detailfenster	Einstellung welche Aktivitätenattribute wo dargestellt werden
View-Konfiguration	Einstellungen zur aktuell dargestellten View
Historie	Anzeige von Historieninformationen
Legende	Aufschlüsseln der Bedeutungen von Farben, speziellen Markierungen, Zuordnung von Nummern in der Darstellung zu Aktivitäten (siehe Abbildung 6.20), etc.

^asiehe Detailansicht 7.3

Tabelle 7.2.: Aufgaben im Aufgabenbereich

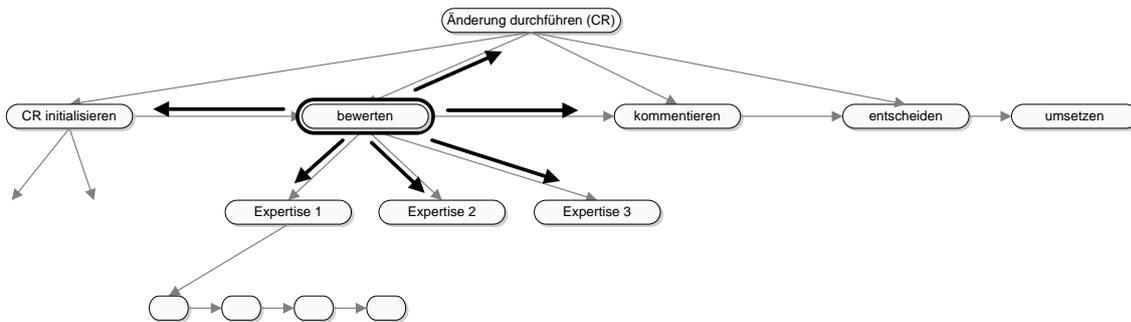


Abbildung 7.7.: Vertikale und horizontale Navigation in Prozesshierarchie

in einem Konfigurationsdialog, wie ihn Abbildung 7.6 zeigt. Kontextmenüs über den Aktivitäten erlauben es direkt zu diesen Einstellungsfenstern zu springen, denn das Interface sollte die Funktionen dort anbieten, wo der Benutzer sie benötigt, also beim Betrachten der Aktivitäten.

View-Konfiguration

Die View-Konfiguration listet vorgenommene Reduktionen auf und die für die Aggregation wichtigen Zuordnungen von Fakten zu Aggregatfunktionen (siehe dazu Abschnitt 7.1).

Historianzeige

Die Historianzeige ist an eine Zeitleiste (siehe Abschnitt 6.1.2) gekoppelt, über die ausgewählt wird, bis zu welchem Datum die Informationen über den Prozessablauf angezeigt werden. Es könnte auch die Möglichkeit angeboten werden, nur bestimmte Aspekte aus der Historie anzuzeigen, beispielsweise nur Änderungen an Daten oder Dokumenten.

Die Auswahl einer Aktivität könnte die Anzeige der Historie auf die Einträge diese Aktivität betreffend einschränken. Analog würde die Auswahl eines Dokuments alle Zugriffe auf das Dokument auflisten.

Legende

Die Legende kann beim ersten Aufruf einer Darstellung automatisch im Aufgabenbereich angezeigt werden. Was die Aufschlüsselung von farblichen Kodierungen in einer Darstellung betrifft, muss ganz allgemein beim Einsatz von Farben darauf geachtet werden, dass die Zahl der Farben sieben möglichst nicht übersteigt. Weniger ist hier sinnvoller. Falls diese Farbzahl nicht ausreicht, können einzelne Farben auch für zwei oder mehr unterschiedliche Dinge stehen. Diese Farbkodierung soll das Erfassen des Inhaltes einer Darstellung beschleunigen, also die Perzeption verbessern. Eine exakte Zuordnung ist nicht notwendig, wäre aber als Konfigurationsoption für den besonderen Bedarf denkbar. Allerdings kann sich das menschliche Gehirn im Mittel nur sieben verschiedene Dinge auf einmal merken, darunter fallen auch Farbzuordnungen. Zudem bekommen Farbenblinde schnell Probleme beim Auseinanderhalten von Farben, wenn zu viele gleichzeitig verwendet werden.

7.5. Navigation

Innerhalb einer Software, die Geschäftsprozesse anzeigt, gibt es drei Arten von Navigation ([CR03]):

- Navigation innerhalb eines Prozesses
- Navigation zwischen verschiedenen Darstellungen eines Prozesses
- Navigation zwischen verschiedenen Instanzen eines Prozesses
- Navigation in hierarchischen Workflows

Außer dieser prozesszentrierten Navigation ist auch die Navigation auf dem Benutzer-Interface einer Visualisierungskomponente wichtig. Die meisten Funktionen werden traditionell in Menüs oder in Toolbars untergebracht. Da es für den Benutzer mehr Navigationsaufwand bedeutet, mit Menüs zu arbeiten, weil ihre Inhalte vor dem Anklicken nicht sichtbar sind, werden häufig benutzte Funktionen besser in Symbolleisten, Paletten oder Ähnlichem untergebracht. Üblicherweise gibt es in Applikationen zwei Funktionsblöcke. Einige Funktionen stehen jederzeit zur Verfügung, andere sind nur bei bestimmten Darstellungen verfügbar. Dies gilt auch für die hier diskutierte Art von Software. Um die Bedienoberfläche nicht mit Schaltflächen zu überfrachten, bietet es sich an, die darstellungsspezifischen Funktionen auf eine gesonderte Symbolleiste auszulagern. Je nach aktiver Darstellung werden in dieser nur die gerade hier möglichen Funktionen angeboten. Tastenkombinationen zum schnellen Erreichen der meisten Funktionen erleichtern die Bedienung zusätzlich.

Die Visualisierungskomponente soll eine Vorwärts/Rückwärts Aktionshistorie ähnlich einem Internet Browser bieten. Dies ergibt sich aus dem im Usability Kapitel angesprochenen Designprinzipien Zurück und Undo (siehe Abschnitt 4.1.3). Für den Benutzer ist es bequem, wenn über ein Auswahlmenü gezielt eine der letzten Ansichten gewählt werden kann. Dazu ist es notwendig den Aktionen Namen zuzuweisen wie 'Switch to Processgraph',

'Switch to Calendar' oder 'Hide Documents'. Die Abbildung 7.8 zeigt diese Funktion am Beispiel eines Browsers, wo die Namensgebung allerdings kein Problem darstellt, denn die Namen der besuchten Webseiten sprechen für sich.



Abbildung 7.8.: Zurück-Funktion mit benannten Aktionen im Internet Explorer

Navigation innerhalb eines Prozesses

Selbst einzelne Prozesse können eine enorme Komplexität aufweisen. Daher gilt es den Benutzer bei der Navigation innerhalb eines Prozesses, durch geeignete Hilfsmittel zu unterstützen.

Das üblichste Navigationswerkzeug sind Scrollbars. Genau diese Form der Navigation gilt es allerdings soweit wie möglich zu vermeiden, denn es ist für den Nutzer die schwierigste Art und Weise zu navigieren. Leider lassen sich selbst kleinere Prozesse kaum auf einer Bildschirmseite unterbringen. Daher muss dem Nutzer möglichst viel Hilfestellung gegeben werden.

Eine kompakte Darstellung reduziert den Navigationsbedarf. Je nach anzuzeigendem Prozess kann eine horizontale oder eine vertikale Ausrichtung der Prozessdarstellung den vorhandenen Bildschirmplatz besser ausnützen. Der Benutzer kann zwischen diesen beiden Orientierungen wechseln, wenngleich die meisten Nutzer aus dem westlichen Kulturkreis, die Informationsaufnahme von links nach rechts bevorzugen.

Um eine kompaktere Darstellung zu erzeugen, bietet das Interface eine einfache Möglichkeit, schnell zwischen verschiedenen Schemata oder Aktivitätenknoten-Konfigurationen (engl. *Styles*) hin- und herzuwechseln. So können einerseits kleinere Knotenrepräsentationen gewählt werden. Andererseits kann die Anzahl der in einem Knoten angezeigten Attribute verkleinert werden (siehe dazu Abbildung 5.11 → S. 71).

Je kleiner die Darstellung der Aktivitätenknoten, desto mehr Überblick über den Prozess gewinnt der Nutzer.

Ein Übersichtsfenster, das den Gesamtprozess als Thumbnail enthält, sorgt für den nötigen Überblick. Ein ideales Beispiel stellt das Übersichtsfenster von Adobe Photoshop dar (Abbildung 7.9). Der im Hauptfenster sichtbare Ausschnitt ist markiert und kann verschoben werden, wodurch sich der angezeigte Ausschnitt im Hauptfenster ändert. In einigen Programmen wird der gerade nicht sichtbare Bereich leicht grau hinterlegt, sodass

gleich zwei optische Merkmale darauf hinweisen, wo sich der gerade sichtbare Bereich befindet. Zusätzlich kann in dem Photoshop Übersichtsfenster stufenlos oder in Schritten heraus- und hineingezoomt werden. Für die Verwendung des Konzeptes Übersichtsfenster im Prozessumfeld sollte darauf geachtet werden, dass im Übersichtsfenster die momentan selektierte Aktivität sowie markierte Aktivitäten deutlich erkennbar sind.

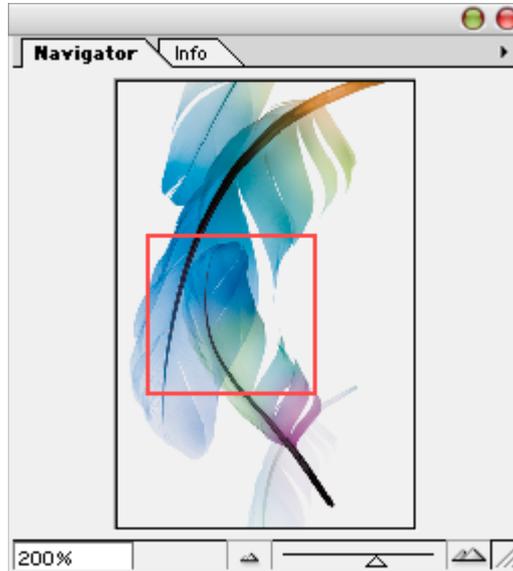


Abbildung 7.9.: Übersichtsfenster aus Adobe Photoshop

Für den Anwender bedeutet es viel Navigationsaufwand, wenn oft zwischen Überblick gewinnen und Details begutachten gewechselt werden muss. Eine praktikable Lösung hierfür ist, das Konzept des Übersichtsfensters auszuweiten und bei Bedarf zwei Fenster, die sich den Bildschirm teilen, anzuzeigen. Das erste Fenster zeigt den Prozess im Überblick (mit einem minimalistischen Schema), das andere zeigt einen Ausschnitt mit den notwendigen Details. Im ersten Fenster ist jederzeit der Ausschnitt markiert, der im zweiten Fenster angezeigt wird.

Navigation zwischen verschiedenen Darstellungen eines Prozesses

Die Standard-Darstellungsform ist die Prozessgraphdarstellung. Via Kontextmenü oder Symbolleiste kann zu anderen Darstellungsformen gewechselt werden. Darstellungen können in Tabs gespeichert werden¹.



Abbildung 7.10.: Tabs im Hauptfenster mit verschiedenen Darstellungen

¹Dieses Bedienkonzept ist den meisten Benutzern geläufig, sei es von Tabellenkalkulationsprogrammen oder vom Tabbed Browsing bei Internet Browsern.

Navigation zwischen verschiedenen Instanzen eines Prozesses

Die umfassendste Navigationsmöglichkeit bietet eine explorerartige Auflistung aller Prozessinstanzen, wie sie in Variante 1 der Tabellendarstellung beschrieben ist (6.4). Natürlich können auch alle Multi-Instanz-Darstellungen dazu genutzt werden eine bestimmte Instanz eines Prozesses zu wählen.

Navigation in hierarchischen Workflows

Als globale Navigationswerkzeuge eignen sich baumartige Darstellungen der ganzen Prozesshierarchie. Dies kann grafisch erfolgen, wie bei einem Organigramm oder in einer explorerartigen Auflistung. Letztgenannte wird auch zur Navigation zwischen Instanzen eines Prozesses verwendet (siehe oben). In ihrer Multi-Schema-Ausprägung (siehe Abschnitt 6.4), kann sie Prozesse, ihre Unterprozesse und Instanzen auflisten. Zusätzlich können in der Tabellenstruktur interessante Informationen wie der Prozessfortschritt oder Ähnliches angezeigt werden. Auf diese Weise lässt sich auch in großen Prozesshierarchien navigieren. Die Prozessauswahl über grafische Darstellungen wird indes bei großen Hierarchien schnell unübersichtlich. Die Vorteile beider Konzepte lassen sich allerdings kombinieren, indem neben der Explorer-Liste eine verkleinerte grafische Darstellung der Prozesshierarchie in einem Übersichtsfenster (vgl. Abschnitt 7.5) angezeigt wird. Weiterhin kann zur Unterstützung der Aufgabenbereich 'Navigation' eingeblendet werden (siehe Abschnitt 7.4).

Für die lokale Navigation bietet sich das bewährte und einfache Konzept einer Navigationsspur (engl. *breadcrumb trail*) an. Sie zeigt, wo sich der Benutzer in der Prozesshierarchie befindet. Umsetzungen finden sich beispielsweise bei den Kategorien von Ebay oder Amazon (siehe Abbildung 7.11). Auf ähnliche Weise soll dem Benutzer stets eingeblendet werden, wo er sich in einem großen hierarchischen Prozess befindet.

Es ist wichtig, dass diese Navigationsinformation jederzeit sichtbar bleibt, im Gegensatz beispielsweise zu Hinweisen auf Detailinformationen, die nur in einem eingegrenzten Kontext von Interesse sind. Weiterhin muss für den Benutzer ersichtlich sein, dass es sich um eine hierarchische Ordnung handelt und nicht um gleichberechtigte Links zu anderen Prozessen. Der Schlüssel hierzu ist die Verwendung eines eindeutigen Trenners wie '>' und '→' anstatt z. B. ':' [Niel01].

Weiterhin steht dem Nutzer der Aufgabenbereich Navigation (siehe Abschnitt 7.4) zur Verfügung, um zu benachbarten Prozessen (des aktuellen Prozesses) zu springen.



Abbildung 7.11.: Navigation im Amazon Shop

7.6. Prozesshistorie

Oftmals interessiert nicht nur der aktuelle Zustand eines Prozesses, sondern auch seine Historie. Diese Prozesshistorie kann in der Historienansicht im Aufgabenbereich (siehe Abschnitt 7.4) eingeblendet werden.

Animation (Replay)

Die Informationen aus der Ablaufhistorie können dazu genutzt werden, um in den Darstellungsformen Prozessgraph und Swimlane, die Ausführungsfolge der Aktivitäten einer Prozess-Instanz vom Start des Prozesses bis zur Gegenwart animiert darzustellen. Der Benutzer kann somit verfolgen, wie an den einzelnen Aktivitäten, während des Ablaufs des Prozesses, die verschiedenen Aktivitätszustände (z. B. aktiviert, laufend, beendet) durchlaufen werden.

7.6.1. On-Line Verlaufsinformationen

Informationen über aktuelle Zustandsänderung an laufenden Prozessen sind nicht nur für eine Visualisierungskomponente interessant. Auch normale Workflow-Clients sind mit ihrem Server verbunden, damit ihre Arbeitslisten aktualisiert werden können. Eine Visualisierungskomponente bucht in ähnlicher Weise Updates über die Prozessausführung bei einem Server. In einem großen System ist zu erwarten, dass sich am Informationsstand des Systems laufend etwas ändert. Aber nicht alle Informationen sind für den Benutzer interessant. Filterfunktionen sollten angeboten werden, damit selektiert werden kann, welche Updates der Server liefern soll. Die verschiedenen angebotenen Darstellungsformen zeigen unterschiedliche Informationen. Für den Benutzer ergeben sich damit drei Fälle:

- **Fall A** Update an der eingeblendeten Darstellungsform
- **Fall B** Update, das nur in einer anderen Darstellungsform sichtbar ist
- **Fall C** Update an Prozessen, die momentan nicht angezeigt werden oder an Prozessteilen, die nicht zur aktuellen Sicht der Visualisierungskomponente gehören²

Nur in den Fällen A und B soll der Benutzer informiert werden. Neu verfügbare Verlaufsinformationen werden im Normalfall nicht sofort ohne Hinweis zur Anzeige gebracht, da dies vom Benutzer übersehen werden könnte oder den Arbeitsablauf stören könnte. Neue Verlaufsinformationen werden daher solange gesammelt, bis der Benutzer die Darstellung aktualisiert. Ein dezent kurz am Bildschirmrand eingeblendetes Popup-Fenster weist den Benutzer auf Updates hin. Ein Klick darauf bringt die neuen Änderungen zur Anzeige und wenn nötig, wird der dargestellte Bildschirmbereich an die richtige Stelle verschoben. Außer dem kurz eingeblendeten Popup-Fenster zeigt das Interface eine Markierung, die über neue Verlaufsinformationen informiert.

²mehr zum Sichtenkonzept siehe Abschnitt 7.1

7.7. Zusammenfassung

Die View-Bildung ist ein mächtiges Werkzeug um Prozessinformationen auf das Wesentliche zusammenzufassen und zu reduzieren. Sie lässt sich parallel zu den in Kapitel 5 und 6 vorgestellten grafischen und strukturellen Aspekten anwenden, um die Prozessvisualisierung signifikant zu verbessern.

In diesem Kapitel wurden außerdem sinnvolle Funktionen für den Umgang mit Prozessinformationen vorgestellt. Werkzeugeisten erschließen dem Anwender einer Visualisierungskomponente diese Funktionsvielfalt. Um die Übersichtlichkeit zu verbessern und um gewünschte Informationen schneller zu finden kann der Benutzer Markierungsregeln verwenden und anlegen. Die Detailansicht dient dazu, die eigentliche Prozessansicht von Details zu befreien, um die Darstellungskomplexität zu verringern. Navigationslösungen, sowohl innerhalb von Prozessen, als auch in Prozesshierarchien wurden vorgestellt.

8

Ausblick

In den folgenden Unterkapiteln wird kurz auf einige in dieser Arbeit nicht näher betrachtete Aspekte eingegangen. So werden Sicherheitsaspekte angesprochen, da Prozessinformationen in Unternehmen äußerst wertvolle Ressourcen darstellen. Weiterhin geht es um das Problem, der nicht eindeutigen Partitionierbarkeit von Prozessinformationen. Den Abschluss bildet ein Fazit der erzielten Ergebnisse.

8.1. Sicherheitsaspekte

Nicht jeder Anwender eine Prozessvisualisierungskomponente soll auf den kompletten Prozess-Pool eines Unternehmens zugreifen können. Insbesondere wenn Kunden, zwecks Kooperation oder Information, Einblick in Arbeitsprozesse erhalten sollen, gilt es ihnen nur ausgewählte Teilbereiche zugänglich zu machen. Es gibt nun zwei naheliegende Ansätze für den Aufbau von Mechanismen für den Zugriffsschutz.

Im Metamodell verankerter Zugriffsschutz

Das Metamodell kann Erweiterungen für den Zugriffsschutz erhalten. So können Attributen beispielsweise Sicherheitsschichten zugeordnet werden, um damit die Sichtbarkeit von Attributen zu regeln und Zugriffsmodi wie nur lesen oder lesen & schreiben. Tabelle 8.1 listet Vor- und Nachteile dieser Methode auf.

Kurzkritik	
+	Aufwand beim Erstellen, Ändern des Metamodells, aber schnelles Zuteilen von Zugriffsrechten möglich
-	Ausdrucksmächtigkeit stark begrenzt, Sonderfälle erfordern hohen Aufwand
-	hoher administrativer Aufwand

Tabelle 8.1.: Kurzkritik für Zugriffsschutz durch Metamodell-Unterstützung

Nutzung von Views für den Zugriffsschutz

Der Zugriffsschutz kann auch personen- oder rollenbezogen durch die Restriktion auf vorbereitete Views erfolgen. Verschiedenen Benutzergruppen werden also explizit Views zugeordnet, die sie für die Erledigung ihrer jeweiligen Aufgabe benötigen. Andere Bereiche der Prozesshierarchie sind nicht zugreifbar. Tabelle 8.2 listet Vor- und Nachteile dieser Methode auf.

Kurzkritik	
+	hohe Sicherheit
+	Zugriff genau an Bedarf anpassbar, erhöht auch die Übersichtlichkeit
-	Aufwand beim Zuteilen der Zugriffsrechte, aber bereits erstellte Views können wiederverwendet werden

Tabelle 8.2.: Kurzkritik für Zugriffsschutz durch Views

Die Visualisierungskomponente könnte mit einer Art Basis-View für den jeweiligen Benutzer gestartet werden, beziehungsweise der Benutzer könnte auf einige Basis-Views Zugriff haben, von denen er eine zu Beginn seiner Sitzung lädt. Der Benutzer kann sich eigene Views erstellen (und abspeichern). Diese sind allerdings immer von einer Basis-View abgeleitet, d. h. sie stellen immer nur eine Untermenge der in der Basis-View festgelegten Prozessinformationen dar. Die Ableitung von einer Basis-View ließe sogar zu, dass sich administrativ fernsteuerbar an den vom Benutzer selbst angelegten Views nachträglich Rechte hinzufügen oder entfernen (engl. *add/revoke rights*) ließen.

8.2. Partitionierbarkeit von Prozessinformationen

Bei den Darstellungskonzepten, die sehr auf das Organisationsmodell zurückgreifen, wie Swimlane und Kalender, existieren Hauptformen oder Varianten, die voraussetzen, dass sich die in einem Prozess enthaltenen Aktivitäten eindeutig bezüglich einer Ressource partitionieren lassen. Dies kann jedoch im Allgemeinen nicht garantiert werden, vor allem, weil PMS die Zuordnung von mehreren Objekten eines Typs zu einer Aktivität erlauben, um die Mächtigkeit der Prozessmodellierung nicht einzuschränken.

Sobald einzelnen Aktivitäten mehr als ein beteiligter Bearbeiter zugeordnet wird, lässt sich eine solche Aktivität in der Rückrichtung nicht mehr eindeutig einem Bearbeiter zuordnen. Im Falle der Swimlane-Darstellung beispielsweise, lässt sich eine Aktivität dann nicht mehr eindeutig einer (Bearbeiter-)Lane zuordnen.

Für die verschiedenen Ressourcen (Ausführungseinheit, Dokumente/Daten, Systeme/Anwendung) existieren Fälle mit nicht eindeutigen Zuordnungen:

- Die Umsetzung des Vieraugenprinzips erfordert mehrere Ausführungseinheiten
- Aktivitäten können mehrere zugeordnete Bearbeiter in verschiedenen Rollen aufweisen, etwa verantwortlich, bearbeitet, berät, wird informiert. Einer Beratungsaktivität könnten so beispielsweise zwei Personen in den Rollen 'berät' und 'wird informiert' zugeordnet sein.
- Nicht selten wird bei einer Aktivität aus mehreren Dokumenten gelesen und geschrieben.

- Aktivitäten an Schnittstellen von Systemen oder Anwendungen erfordern häufig Zugriff auf zwei Systeme.

Das beschriebene Problem tritt also bei allen Arten der Partitionierung auf: Ausführungseinheiten, Dokumente und Systeme. Immer können mehrere Objekte einer Aktivität zugeordnet werden. Oftmals besteht bei der Visualisierung jedoch die Notwendigkeit der eindeutigen Zuordnung von Ressourcen zu Aktivitäten.

Mögliche Ansätze für den Umgang mit dem Problem:

- **Hauptobjekte**

Vom Prozess-Metamodell wird die Bezeichnung eines Hauptobjekts für jeden Ressourcentyp vorgesehen. Jedoch stellt sich die Frage, nach welchem Kriterium ein Objekt als Hauptobjekt deklariert wird.

- **Schatten-Aktivitäten**

Eine Aktivität wird in der Darstellung in mehrere Schatten-Aktivitäten aufgeteilt und somit mehreren Ressourcen gleichzeitig zugeordnet – in der Swimlane-Darstellung also mehreren Lanes. Die Visualisierung muss zwischen normalen Aktivitäten und Schatten-Aktivitäten unterscheiden und muss deutlich machen, dass zu einer Schatten-Aktivität einer Partition noch weitere in anderen Partitionen gehören. Es bietet sich eine halbtransparente Darstellung dieser Aktivitäten an. Bei Selektion einer Aktivität werden aller zusammengehörigen Aktivitäten hervorgehoben – auch in der Übersichtsdarstellung (siehe Abschnitt 7.5).

- **Transformation**

Einige Fälle wie das Vieraugenprinzip lassen sich auch auf andere Art modellieren. In diesem Fall durch eine Ersetzung der einzelnen Aktivität mit zwei Bearbeitern in zwei parallele Aktivitäten, gegebenenfalls durch eine Sync-Kante verbunden, mit je einem Bearbeiter. Jedoch erzeugt diese Vorgehensweise schwer verständliche komplexe Modelle.

Diese Ansätze lassen sich für eine geeignete Darstellung auch alle miteinander kombinieren. Erwähnt werden kann hier noch, dass die Matrixdarstellung von dieser Problematik nicht betroffen ist, da hier die Ressourcen selbst im Mittelpunkt stehen und nicht Aktivitäten, die zugeordnet werden müssen.

8.3. Anpassung des Nutzer-Interface an Prozessmodelle

Eine Visualisierungskomponente für Prozessinformationen ist ein hochkomplexes Stück Software, was dazu führt, dass auch die Bedienbarkeit nicht ohne weiteres einfach zu halten ist. Gerade weil die Prozessinformationen selbst niemals fest definiert sind, sondern immer anwendungsspezifische Erweiterungen des Metamodells oder des Datenmodells enthalten können, muss sich die Benutzeroberfläche an diese Änderungen anpassen lassen, was zwangsweise Konfigurationsaufwand nach sich zieht und die Oberfläche komplizierter macht. Eine Reduktion der Komplexität der Oberfläche ließe sich erreichen, indem Menüs und Einstellungen der Darstellungsformen dynamisch anpassbar gemacht werden. Für alle verwendeten Prozessmodelle ließe sich einmalig eine GUI-Konfiguration erstellen. Anhand

dieser Konfigurationen würde dann beim Betrachten von Prozessinformationen die zum jeweiligen Prozessmodell passende Konfiguration für die Nutzeroberfläche geladen.

8.4. Mitbestimmung des Betriebsrates

In deutschen Unternehmen hat der Betriebsrat ein gesetzlich verankertes Mitbestimmungsrecht was die Thematik des Datenschutzes angeht. Dieser wird bei der Protokollierung von Benutzeraktionen tangiert. Dies ist für die Nutzung eines PMS jedoch unumgänglich. Sobald jedoch die Prozessdaten nicht nur elektronisch verarbeitet, sondern zur Visualisierung aufbereitet werden, sollte der Betriebsrat möglichst frühzeitig informiert und in die Planung mit eingebunden werden.

8.5. Fazit

Die Visualisierung von Geschäftsprozessen wird für Unternehmen immer wichtiger. Es existieren heute viele fortgeschrittene Werkzeuge für Prozess-Monitoring und -Analyse. Diese bieten allerdings nur recht eingeschränkte Visualisierungsmöglichkeiten für Prozessmodelle und -Instanzen.

In dieser Arbeit wurde erstmals umfassend untersucht, welche Möglichkeiten bestehen, um Prozessdaten zu visualisieren. Dabei wurden Aspekte wie die grafische Repräsentation, die Strukturierung der Daten, das Erstellen geeigneter Sichten, die Verknüpfung mit Wissen aus Organisationsmodellen und die Anpassbarkeit an Benutzerbedürfnisse berücksichtigt. Das Ergebnis ist eine umfangreiche Kollektion von Visualisierungskonzepten und Interaktionsmöglichkeiten auf Prozessdaten. Diese wurden im Hinblick auf eine künftige systemübergreifende Visualisierungskomponente zusammengestellt und aufeinander abgestimmt.

AUSBLICK

An dieser Stelle ein kurzer Ausblick, bevor näher auf die erreichten Ergebnisse eingegangen wird. Eine Zielsetzung der Arbeit war, möglichst viele zusammenhängende Informationen aus den verfügbaren Prozessdaten zu extrahieren. Es bestehen nun noch weitere Möglichkeiten, um Prozessinformationen anzureichern. Dafür ist es allerdings notwendig, die Datenbasis der Visualisierungskomponente zu erweitern und außer Modell- und Laufzeitdaten auch die Log-Daten von PMS auszuwerten. Durch abgeleitetes Prozesswissen aus vergangenen Prozessausführungen, könnten so beispielsweise Wahrscheinlichkeiten bei Kontrollflussverzweigungen angegeben werden oder statistisch auffällige Werte dem Benutzer direkt in den Darstellungen signalisiert werden. Ein Endziel könnte darin bestehen, eine wie in dieser Arbeit beschriebene Prozessvisualisierungskomponente mit einer Process-Mining-Komponente zu verbinden. Am Schluss könnten Visualisierung, Monitoring, Analyse und Optimierung einheitlich unter einer Oberfläche vereint werden. Die vorgestellte Visualisierungskomponente soll Prozessdaten aus einem heterogenen Prozessumfeld anzeigen. Hier sind Erweiterungen in Richtung unternehmensübergreifender Prozessvisualisierung (engl. *cross organizational*) denkbar, um Interaktionen und Datenflüsse zwischen Prozessen verschiedener Unternehmen darstellen zu können.

ZUSAMMENFASSUNG

Die Zielsetzung war, Konzepte für die Visualisierung von Prozessinformationen zu entwickeln, um Anwendern einer Visualisierungskomponente möglichst alle Aspekte von Prozessinformationen, über komfortable leistungsfähige Darstellungen, zugänglich zu machen. Die Lösung dieser Aufgabe erfolgte über unterschiedliche Ansätze.

Der erste Ansatz bestand darin zu untersuchen, welchen Anforderungen eine leistungsfähige Prozessvisualisierungskomponente genügen muss. Außer Anforderungen an die allgemeine Gebrauchstauglichkeit (Usability) sollen Anwenderbedürfnisse berücksichtigt werden. Aus den Erfordernissen künftiger Nutzergruppen wie etwa Anwendern, Management oder Kunden, wurden grundlegende Aufgabentypen und Detailgrad-Anforderungen abgeleitet. Dieses Anforderungsspektrum diente als Gradmesser im Hinblick auf die zu entwickelnden Visualisierungskonzepte.

Der zweite Ansatz war, grafische Aspekte existierender Darstellungsformen, wie etwa von Prozessgraphen, zu verbessern. Die Optimierung erfolgte im Hinblick auf die Stärken und Schwächen des menschlichen Wahrnehmungssystems, um möglichst viele Informationen in einer dennoch schnell erfassbaren Form zu visualisieren. Um vom Wiedererkennungswert zu profitieren, entstand, in Anlehnung an UML und BPMN, eine grafische Notation, die eine hohe Packungsdichte zulässt, um die begrenzte Anzeigefläche möglichst effektiv auszunutzen.

Weiterer Ansatz war, strukturelle Aspekte der Prozessvisualisierung zu untersuchen. Für alle in Prozessdaten enthaltenen Informationsaspekte, wie etwa Datenfluss oder temporale Aspekte, sollte eine geeignete Struktur für die Darstellung gefunden werden. Denn Prozessinformationen sind nicht auf ihre bei der Modellierung vorgegebene Struktur festgelegt. Zeitinformationen lassen sich beispielsweise in Gantt-Diagrammen gut wiedergeben.

Über diese Ansätze entstanden so optimierte Darstellungsformen, wie Prozessgraphdarstellung, Swimlane-Darstellung, Kalendardarstellung, Tabellendarstellung, Interaktionsdiagramm, Datenflussdiagramm und eine Matrixdarstellung. Diese Formen können mit verschiedenen Darstellungsarten kombiniert werden: Schema-, Instanz-, Multi-Instanz- und Multi-Schema-Darstellungen. Dies führt zu einer großen Vielfalt verschiedenster Detail- und Übersichtsdarstellungen.

Eine Hauptschwierigkeit bei Bildschirmdarstellungen stellt die begrenzte Anzeigefläche dar, im Vergleich zu häufig sehr großen und komplexen Prozessen. Daher wurden rund um diese Visualisierungskonzepte umfangreiche Interaktions- und Navigationsmöglichkeiten für eine künftige Prozessvisualisierungskomponente vorgeschlagen. Darunter beispielsweise die View-Bildung. Sie bildet die Grundlage, zur Erzeugung unterschiedlich komplexer Sichten auf Prozesse.

Der Fokus dieser Arbeit liegt auf der Beschreibung verschiedener Darstellungskonzepte. Daher wurden für die Visualisierungsbeispiele nur Standardkonstrukte eines Prozess-Metamodells verwendet. Spezialkonstrukte wie Variable Parallelität, Mengenverarbeitung (engl. *expansion region* [Omg 05b]) oder Zeitkanten sollten sich aber auch in die hier vorgestellten Darstellungskonzepte integrieren lassen.

Diese Arbeit beschreibt eine Vision für eine mächtige Prozessvisualisierungskomponente, die umfangreiche Möglichkeiten bietet, in Prozessinformationen zu recherchieren und in

hierarchischen Prozessen zu navigieren. Solche Visionen sind nie endgültig, die Zusammenstellung der Darstellungskonzepte und deren Konfigurations- und Interaktionsmöglichkeiten kann nie im mathematischen Sinne als vollständig erachtet werden. So wichtig und nützlich eine sorgfältige und ausführliche Analyse auch ist, nur ein stetiger Verbesserungsprozess von Version zu Version durch ausführliches Nutzer-Feedback wird die Vision Wirklichkeit werden lassen, jede nur erdenkliche Auswertung von verfügbaren Prozessinformationen möglich zu machen. Dies gilt umso mehr, als dass im Rahmen dieser theoretischen Arbeit keine Nutzerumfrage möglich war. Letztlich muss die Praxis zeigen, welche Bedürfnisse die Benutzer haben und wie diese Darstellungsformen noch verfeinert werden können, welche Interaktionen noch benötigt werden und welche neuen Darstellungsformen eventuell noch notwendig werden. Wissenschaftliche Untersuchungen dazu laufen bereits, so z. B. [LM04].

A

Glossar

B

Breadcrumb trail

Eine deutsche Übersetzung existierte zur Drucklegung noch nicht. In Näherung könnte es mit 'Navigation in hierarchischen Kategorien' oder Navigationsspur wiedergegeben werden. Die Wendung entstand wohl in Anlehnung an die Brotkrümelspur von Hänsel und Gretel. Der Nutzer bekommt mehrere Links dargestellt, die ihm geordnet den Weg zurück weisen. (siehe [7.5](#) und [Abbildung 7.11](#))

BPMN

Business Process Management Notation.

BPM Werkzeuge

Business Process Management Werkzeuge dienen zum Management von Geschäftsprozessen. Sie umfassen die Teilbereiche Modellierung, Analyse, Monitoring & Steuerung von Prozessen (→ siehe [PMS](#)).

D

Darstellung

Prozessinformationen können unterschiedlich dargestellt werden, es gibt textuelle/strukturelle oder graphische Ausprägungen. Darstellungen sind Visualisierungskonzepte.

Data-Warehouse

Eine einheitliche Definition existiert derzeit nicht, eine treffende Beschreibung wäre eine zentrale Daten-Sammelstelle in der Daten aus unterschiedlichsten Quellen integriert werden, um eine globale Sicht auf die Daten zu ermöglichen, zum Zwecke einer gemeinsamen Auswertung und Weiterverwendung. Umfassende Informationen bietet ein Buch vom 'Vater des Data Warehousing' William Inmon [[Inmo96](#)].

F

Form follows function

Entwurfskonzept für die Anwendungsentwicklung. Hauptziel der Entwicklung ist, dass die Anwendung bestimmte Funktionen anbietet. Ihr Design orientiert sich an der Bereitstellung von Funktionen. Dies entspricht häufig dem Denken von Entwicklern in Kategorien wie Funktionen und Features. Das gegenteilige Konzept ist → *Goal-Directed-Design*.

G

Goal-Directed-Design

Entwurfskonzept für die Anwendungsentwicklung. Hauptziel der Entwicklung ist, dass das Design der Anwendung darauf ausgerichtet ist, möglichst effektiv die Benutzer bei der Erreichung ihrer Ziele zu unterstützen. Das gegenteilige Konzept ist → *Form follows function*

GUI

Grafische Benutzerführung (engl. *Graphical User Interface*) im Gegensatz zu Kommandozeilenbasierten Schnittstellen. Grafikelemente und Mausbedienung ermöglichen eine leicht erlernbare, standardisierte Benutzung von Programmen und Betriebssystem

H

HCI

Human Computer Interaction (dt. Mensch-Maschine-Interaktion) ist ein interdisziplinärer Forschungszweig, dessen Zielsetzung es ist möglichst gute Benutzerschnittstellen zu schaffen

Hierarchische Workflows

Unternehmensweite Geschäftsprozesse werden in hierarchischen Workflows organisiert. Sie bilden eine Hierarchie von Haupt- und Teilprozessen (siehe Abbildung 7.7).

HSL-Farbraum

(auch HSB) Farben werden durch unterschiedliche Farbräume beschrieben. Bekannter ist der RGB-Farbraum. HSL steht für Hue (Farbton), Saturation (Farbsättigung) und Lightness/Brightness (Farbhelligkeit). Der Farbton wählt eine Farbe aus einem Farbkreis, wobei Rot 0° entspricht, Gelb 60°, Grün 120°, Cyan 180°, Blau 240° und Magenta 300°. Die Sättigung gibt die Reinheit der Farbe in Prozent an, 0% bedeutet Grau. 100% Helligkeit erzeugt die Farbe Weiß, 0% die Farbe Schwarz, 50% Helligkeit entspricht dem reinen Farbton.

M

MSC

Message Sequence Charts sind Sequenzdiagramme, die ursprünglich zur Modellierung von Telekommunikationssystemen entwickelt wurden [BH89]. Die ITU-TS hat dazu Ende 1991 einen ersten Standard namens Z.120/92 verabschiedet [ITU-91]. Später kamen noch Erweiterungen hinzu. MSC's sind als Sequenzdiagramme auch Teil von UML [Omg 05b], dort unterscheiden sie sich jedoch vom ITU-TS Standard und sind objektorientiert.

O

Orthogonalität

(auch Interne Konsistenz) Es werden Ausdrucksformen benötigt, die Ähnlichkeit und Verschiedenheit gleichzeitig transportieren können. Ein Symbol kann z. B. leicht modifiziert werden, dabei aber Aspekte seiner Form oder die Farbe behalten. Auf diese Weise kann der Betrachter leicht die Ähnlichkeit von zwei Informationen erkennen, aber gleichzeitig deren Unterschiedlichkeit erfassen. Ein Beispiel wäre das Symbol für eine Aktivität, welches in zwei Farben dargestellt wird, um beispielsweise damit darzustellen, dass die Aktivität zwei unterschiedlichen Bearbeitern zugeordnet ist.

P

Personas

Ein wichtiges Konzept für das Interface-Design sind nach Alan Cooper Personas [CR03, S.55]. Diese sind möglichst präzise Beschreibungen der späteren Benutzergruppen, mit-samt ihren Motivationen und Zielen (engl. *goals*). Die Personas mit ihren Zielen sind die Grundlage des → *Goal-Directed-Design*.

PMS

Prozess-Management-Systeme sind ein Oberbegriff für Workflow Management Systeme (→ siehe *WfMS*), denn sie umfassen üblicherweise eine sehr viel weitergehendere Technologie. Alternativ wird oftmals auch von Business Process Management Systemen (BPMS) gesprochen. PMS optimieren Arbeitsabläufe in Unternehmen. Hierfür werden Geschäftsprozesse in ihre einzelnen Arbeitsschritte zerlegt und für die Hinterlegung in einem PMS modelliert. PMS bieten u. a. folgende Funktionalität im Umgang mit Prozessen: ausführen, automatisieren, kontrollieren, beobachten und analysieren.

V

View

Eine View ist eine Abbildung von der Menge aller Prozessinformationen auf eine beliebige Untermenge von Prozessinformationen. → *Darstellungen* zeigen Views an.

Visualisierungskonzept

siehe → *Darstellung*

W

WfMS

Workflow Management System (Abkürzung auch WFM) → siehe *PMS*

B

Literaturverzeichnis

- [AbsI06] ABSINT ANGEWANDTE INFORMATIK GMBH: *aiSee - Graph Visualization (Website)*. Version: 2006. <http://www.absint.com/aisee/>. – Online Ressource, Abruf: 07.02.2006
- [ades01] ADESSO AG: *LeuSmart (Version 3.2.1), Benutzerhandbuch*
- [Appl87] APPLE COMPUTER INC.: *Apple Human Interface Guidelines*. Version: 1987. <http://developer.apple.com/documentation/UserExperience/Conceptual/OSXHIGuidelines/OSXHIGuidelines.pdf>. – Online Ressource, Abruf: 27.07.2005
- [BH89] BELINA, F. ; HOGREFE, D.: *The CCITT Specification and Description Language SDL*. In: Computer Networks and ISDN Systems Vol. 16 (1989), März
- [BMI 02] BMI BUNDESMINISTERIUM DES INNEREN DER BUNDESREPUBLIK DEUTSCHLAND: *Barrierefreie Informationstechnik-Verordnung BITV*. Version: Juli 2002. <http://bundesrecht.juris.de/bitv/index.html>. – Online Ressource, Abruf: 01.02.2006
- [Bons96] BONSIPE, Gui: *Interface. Design neu begreifen*. Mannheim : Bollmann Verlag, 1996
- [Bpmi03a] BPMI ORGANIZATION: *BPMN Business Process Modeling Notation (Website)*. Version: 2003. <http://www.bpmn.org>. – Online Ressource, Abruf: 27.10.2005
- [Bpmi03b] BPMI ORGANIZATION: *Business Process Management Initiative (Website)*. Version: 2003. <http://www.bpmi.org>. – Online Ressource, Abruf: 27.10.2005
- [BRB05] BOBRIK, Ralph ; REICHERT, Manfred ; BAUER, Thomas: *Requirements for the Visualization of System-Spanning Business Processes*. In: *16th International Workshop on Database and Expert Systems Applications (DEXA 2005)*. Copenhagen, Denmark : IEEE Computer Society, August 2005, S. 948–954
- [Bris05] BRISTOL TECHNOLOGY: *TransactionVision White Paper*. Version: Juni 2005. <http://www.bristol.com/transactionvision/>. – Online Ressource, Abruf: 25.01.2006
- [Cava03] CAVALLERO, Dave: *Integration technology lets business and IT collaborate to improve business results*. IBM Software Group
- [Coll03] COLLAXA INC.: *Collaxa BPEL4WS 101 Tutorial*. Version: 2003. <http://www.cems.uwe.ac.uk/~cjwallac/UFIE8V-2004/collaxa-bpel101.pdf>. – Online Ressource, Abruf: 06.02.2006

- [Coop05] COOPER, Alan: *Article on Goal-Directed Design (Website)*. Version: 2005. http://www.cooper.com/articles/art_goal_directed_design.htm. – Online Ressource, Abruf: 13.02.2006
- [COPA06] COPA-DATA GMBH: *COPA-DATA (Website)*. Version: 2006. <http://www.copadata.de/>. – Online Ressource, Abruf: 07.02.2006
- [CR03] COOPER, Alan ; REIMANN, Robert: *About Face 2.0. The Essentials of Interaction Design*. John Wiley & Sons, 2003. – ISBN 0764526413
- [Dece05] DECEMBER, John: *HTML Station (Website)*. Version: 2005. www.december.com/html/spec/colorshadesuse.html. – Online Ressource, Abruf: 08.08.2005
- [DGS95] DEITERS, Wolfgang ; GRUHN, Volker ; STRIEMER, Rüdiger: *Der FUNSOFT-Ansatz zum integrierten Geschäftsprozessmanagement*. In: *Wirtschaftsinformatik* (1995), Nr. 5, S. pp. 459–466. – ISSN 0302–9743
- [Dodd02] DODD, John: *Perceptual factors in Information Visualisation (Website)*. Version: 2002. <http://www.bunnyfoot.com/freestuff/articles/general/visualisation.html>. – Online Ressource, Abruf: 28.10.2005
- [DOL02] DENFORD, Mark ; O'NEILL, Tim ; LEANEY, John: *Architecture-Based Visualisation of Computer Based Systems*. In: *ECBS*, 2002, S. 139–146
- [DR03] DADAM, Peter ; REICHERT, Manfred: *ADEPT - Prozess-Management-Technologie der nächsten Generation*. Abteilung Datenbanken und Informationssysteme, Universität Ulm. Version: 2003. <http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/DaRe04.pdf>. – Online Ressource, Abruf: 06.09.2005
- [e-FA00] E-FACT GMBH: *ProcessReports User Manual V1.0*
- [Ecli06] ECLIPSE.ORG: *Eclipse (Website)*. Version: 2006. <http://www.eclipse.org/>. – Online Ressource, Abruf: 15.02.2006
- [Gefa04] GEFASOFT AG: *Gefasoft (Website)*. Version: 2004. <http://www.gefasoft.de/>. – Online Ressource, Abruf: 07.02.2006
- [Heid02] HEIDMANN, Frank: *Web Design - Guidelines and Standards*. Fraunhofer IAO. Version: 2002. http://ais.informatik.uni-leipzig.de/download/2002w_v_mmk/2002w_mmk_v.12.pdf. – Online Ressource, Abruf: 18.07.2005
- [Huss84] HUSSY, W.: *Denkpsychologie, Band 1. Ein Lehrbuch*. Stuttgart : Kohlhammer, 1984
- [IBM 03a] IBM CORP.: *Business activity management: Your window of opportunity for better business operations. IBM Business integration solutions, Executive brief*
- [IBM 03b] IBM CORP.: *IBM WebSphere Business Integration Monitor V 4.2.4*. Version: September 2003. ftp://ftp.software.ibm.com/software/integration/library/specsheets/wbimonitor_G325-2210-01.pdf. – Online Ressource, Abruf: 06.02.2006
- [IBM,03] IBM, BEA SYSTEMS, MICROSOFT, SAP AG, SIEBEL SYSTEMS: *Business Process Execution Language for Web Services version 1.1*. Version: Mai 2003. <http://www-106.ibm.com/developerworks/webservices/library/ws-bpel/>. – Online Ressource, Abruf: 06.02.2006
- [IDS 03] IDS SCHEER AG: *ARIS Process Performance Manager (ARIS PPM). White Paper*. Version: 2003. http://www.ids-scheer.com/international/german/products/aris_controlling_platform/49532. – Online Ressource, Abruf: 04.02.2006
- [iGra06] IGRAFX: *iGrafx (Website)*. Version: 2006. <http://www.igrafx.de/>. – Online Ressource, Abruf: 04.02.2006

- [ILOG06] ILOG INC.: *ILOG Visualization (Website)*. Version: 2006. <http://www.ilog.com/products/visualization/>. – Online Ressource, Abruf: 07.02.2006
- [in -03] IN - INTEGRIERTE INFORMATIONSSYSTEME GMBH: *sphinx open (Website)*. Version: 2003. <http://www.in-gmbh.de/de/Produkte/Visualisierung/index.html>. – Online Ressource, Abruf: 07.02.2006
- [Inmo96] INMON, W. H.: *Building the Data Warehouse*. QED Publishing Group, 1996
- [Inos03] INOSOFT GMBH: *VisiWinStudio - Visualisierungs-Systeme. White Paper*. Version: 2003. <http://www.inosoft.com>. – Online Ressource, Abruf: 07.02.2006
- [Intr01] INTRAWARE: *Prozessdokumentation und Simulation mit Bonapart. Factsheet Bonapart*
- [ISO 98] ISO ORGANIZATION: *DIN EN ISO 9241/11. Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten - Teil 11: Anforderungen an die Gebrauchstauglichkeit*. Version: 1998. <http://www.iso.org>. – Online Ressource
- [ISO 06] ISO ORGANIZATION: *International Organization for Standardization (Website)*. Version: 2006. <http://www.iso.org>. – Online Ressource, Abruf: 01.02.2006
- [ITU-91] ITU-TS INTERNATIONAL TELECOMMUNICATIONS UNION - TELESERVICES: *Recommendation Z.120/92: Message Sequence Chart*
- [Jabl01] JABLONSKI, Stefan: *Prozessorientiertes Wissensmanagement*. In: KnowTech 2001 (2001), November. <http://www.prodato.de/presentation/knowtech2001.pdf>. – Online Ressource, Abruf: 05.02.2006
- [Jaco03] JACOBSEN, Jens: *Usability-Normen*. Version: Januar 2003. http://www.contentmanager.de/magazin/artikel_582-199_usability_normen_din_iso.html. – Online Ressource, Abruf: 01.02.2006
- [JK01] JUNGINGER, S. ; KARAGIANNIS, D.: *Entwicklung von Workflow-Anwendungen*. In: WISU - Das Wirtschaftsstudium (2001), März, S. pp. 346–354
- [KDV02] KONING, Henk ; DORMANN, Claire ; VLIET, Hans van: *Practical guidelines for the readability of IT-architecture diagrams*. In: *SIGDOC 2002 - 20th Annual International Conference on Documentation*, 2002, S. pp. 90–99
- [KJS96] KARAGIANNIS, D. ; JUNGINGER, S. ; STROBL, R.: *Business Process Modeling*. (1996), S. pp. 81–106
- [KK01] KÜHN, H. ; KARAGIANNIS, D.: *Modellierung und Simulation von Geschäftsprozessen*. In: WISU - Das Wirtschaftsstudium 30 (2001), August, S. pp. 1161–1170
- [Klot04] KLOTZ, Achim: *View-Unterstützung in Prozess-Management-Systemen*, Universität, Ulm, Diplomarbeit, Oktober 2004
- [LLW⁺01] LUTTIGHUIS, Paul O. ; LANKHORST, Marc ; WETERING, Rob van d. ; BAL, René ; BERG, Harmen van d.: *Visualising business processes*. In: *Computer Languages Vol. 27* (2001), April–Oktober, Nr. 1–3, pp. 39–59. <http://www.niii.ru.nl/~erikp/Lectures.dir/Mv0/2004/Testbed2.pdf>. – Online Ressource, Abruf: 24.06.2005. – ISSN 0096–0551
- [LM04] LAI, Maggie B. ; MUEHLEN, Michael zur: *Information Requirements of Process Stakeholders: A Framework to Evaluate Process Monitoring and Controlling Applications*. In: *Proceedings of the Pre-ICIS workshop on Process Management*. J. Akoka and I. Comyn-Wattiau and M. Favier
- [LR99] LEYMANN, Frank ; ROLLER, Dieter: *Production Workflow: Concepts and Techniques*. 1. Prentice Hall PTR, 1999. – ISBN 0130217530
- [Maps06] MAPSOLUTE GMBH: *Map24.de - Routenplaner und Stadtpläne (Website)*. Version: 2006. <http://www.map24.de/>. – Online Ressource, Abruf: 30.01.2006

- [Miha05] MIHALCA, Tiberius: *Prozessdatenintegration und -transformation für die systemübergreifende Visualisierung von Arbeitsabläufen*, Universität, Ulm, Diplomarbeit, November 2005
- [Mill56] MILLER, George: *The magic number seven, plus or minus two: Some limits on our capacity for processing information*. In: *Psychological Review* (1956)
- [Mill68] MILLER, Robert B.: *Response time in man-computer conversational transactions*. In: *Proc. AFIPS Fall Joint Computer Conference*, 1968
- [MR04] MUTSCHLER, Bela ; REICHERT, Manfred: *Usability-Metriken als Nachweis der Wirtschaftlichkeit von Verbesserungen der Mensch-Maschine-Schnittstelle*. In: *14th International Workshop on Software Measurement, DASMA Metrik Kongress (IWSM'04/Metrikon2004)*. Berlin, November 2004
- [Niel01] NIELSEN, Jacob: *useit.com: Jakob Nielsen's Website*. Version: November 2001. <http://www.useit.com/about/nographics.html>. – Online Ressource, Abruf: 03.08.2005
- [NIST93a] NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY: *FIPS Federal Information Processing Standards Publication 183, Integration Definition for Function Modelling (IDEF0)*. Version: Dezember 1993. <http://www.idef.com/pdf/idef0.pdf>. – Online Ressource, Abruf: 9.1.2006
- [NIST93b] NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY: *FIPS Federal Information Processing Standards Publication 184, Integration Definition for Function Modelling (IDEF1x)*. Version: Dezember 1993. <http://www.4dcompanion.com/downloads/standards/IDEF1X.pdf>. – Online Ressource, Abruf: 9.1.2006
- [NM94] NIELSEN, Jakob ; MACK, R.L.: *Usability Inspection Methods*. John Wiley & Sons http://www.useit.com/papers/heuristic/heuristic_list.html
- [Omg 05a] OMG ORGANIZATION: *Object Management Group (Website)*. Version: 2005. <http://www.omg.org>. – Online Ressource, Abruf: 27.10.2005
- [Omg 05b] OMG ORGANIZATION: *Unified Modeling Language (UML) v2.0*. Version: 2005. <http://www.omg.org/technology/documents/formal/uml.htm>. – Online Ressource, Abruf: 27.10.2005
- [orea04] OREAS GMBH: *oreas (Website)*. Version: 2004. <http://www.oreas.com/>. – Online Ressource, Abruf: 07.02.2006
- [Park05] PARKINSON, Cyril N.: *Zitat*. Encyclopaedia Britannica Online. Version: März 05. <http://www.britannica.com/eb/article?tocId=9058517>. – Online Ressource, Abruf: 22.03.2005
- [Peis05] PEISL, Roland: *Delivering on demand business agility with business process management. White paper (G325-1999-02)*. IBM Software Group. Version: Januar 2005. <http://www.elink.ibm.com/public/applications/publications/cgi-bin/pbi.cgi?CTY=US&FNC=SRX&PBL=G325-1999-02#>. – Online Ressource, Abruf: 03.02.2006
- [PR94] PALMER, S. E. ; ROCK, I.: *Rethinking perceptual organization: The role of uniform connectedness*. In: *Psychonomic Bulletin and Review* Vol. 1 (1994), S. pp. 29–55
- [RBRB06] RINDERLE, Stefanie ; BOBRIK, Ralph ; REICHERT, Manfred ; BAUER, Thomas: *Business Process Visualization - Use Cases, Challenges, Solutions*. In: *8th International Conference on Enterprise Information Systems (ICEIS'06)*. Paphos, Cyprus, Mai 2006. – to appear
- [RD98] REICHERT, Manfred ; DADAM, Peter: *ADEPTflex - Supporting Dynamic Changes of Workflows Without Losing Control*. In: *Journal of Intelligent Information Systems* Vol. 10 (1998), März-April, Nr. 2, pp. 93-129. <http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/ReDa98.pdf>. – Online Ressource, Abruf: 27.01.2006

- [RD00] REICHERT, Manfred ; DADAM, Peter: *Geschäftsprozeßmodellierung und Workflow-Management-Konzepte, Systeme und deren Anwendung*. In: *Industrie Management* Vol. 3 (2000), pp. 23-27. <http://www.informatik.uni-ulm.de/dbis/01/dbis/downloads/ReDa00.pdf>. – Online Ressource, Abruf: 29.01.2006
- [Rein93] REINWALD, Berthold: *Workflow-Management in Verteilten Systemen*. Stuttgart : Teubner, 1993
- [RH00] RUDOLF, Hartmut ; HALLENBERGER, Brigitte: *Tutorial: Farben im Webdesign (Website)*. Version: 2000. www.metacolor.de/farbwaehler_AuffaecherungenOLD.htm. – Online Ressource, Abruf: 08.08.2005
- [Rind04] RINDERLE, Stefanie: *Schema Evolution in Process Management Systems*, Universität, Ulm, Diplomarbeit, 2004
- [RR03] REICHERT, Manfred ; RINDERLE, Stefanie: *ADEPT - Prozess-Management-Technologie der nächsten Generation*. Abteilung Datenbanken und Informationssysteme, Universität Ulm
- [RS56] RYAN, T.A. ; SCHWARTZ, C.B.: *Speed of perception as a function of mode of representation*. In: *American Journal of Psychology* Vol. 69 (1956), S. pp. 60–69
- [Sche01] SCHEER, August-Wilhelm: *ARIS - Modellierungsmethoden, Metamodell, Anwendungen*. 4. Auflage. Berlin : Springer, 2001
- [Sche02] SCHEER, August-Wilhelm: *ARIS - Vom Geschäftsprozess zum Anwendungssystem*. 4. Auflage. Berlin : Springer, 2002. – ISBN 3–540–63835–0
- [Schu05] SCHUHMACHER, Joachim: *Usability (Website)*. Version: 2005. <http://www.multimedia-beratung.de/ergonomie/theorie/grundlagen/usability.htm>. – Online Ressource, Abruf: 26.10.2005
- [Schw03] SCHWAB, J.: *Geschäftsprozessmanagement mit Visio, ViFlow und MS Project*. Hanser Fachbuch, 2003
- [SH99] SCHEER, August-Wilhelm ; HOFFMANN, Michael: *From Business Process Model to Application System - Developing an Information System with the House of Business Engineering (HOBE)*. In: *Int'l Conf. on Advanced Information Systems Engineering, Heidelberg* Vol. 1626 (1999), Juni, S. pp. 2–9. – ISSN 0302–9743
- [SP05] SHNEIDERMAN, Ben ; PLAISANT, Catherine: *Designing the User Interface: Fourth Edition Strategies for Effective Human-Computer Interaction*. Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc., 2005. – ISBN 0–321–26978–0
- [Spal05] SPALL, Andreas: *Einführung in BPEL4WS*. Orientations in Objects GmbH. Version: April 2005. <http://www.oio.de/public/xml/einfuehrung-in-bpel4ws.htm>. – Online Ressource, Abruf: 06.02.2006
- [This00] THISSEN, Frank: *Screen Design Handbuch*. Berlin : Springer Verlag, 2000. – ISBN 3–540–64804–6
- [Togn92] TOGNAZZINI, Bruce: *Tog on Interface*. Reading, MA : Addison-Wesley, 1992. – ISBN 0–201–60842–1
- [Togn04] TOGNAZZINI, Bruce: *First Principles of Interaction Design*. Version: 2004. <http://www.asktog.com/basics/firstPrinciples.html>. – Online Ressource, Abruf: 28.01.2006
- [Ulti02] ULTIMUS: *Ultimus Workflow Suite 4 - Product Guide*. Version: Juli 2002. <http://www.quixa.com/library/docs/QLLIB-ULT-101.pdf>. – Online Ressource, Abruf: 07.02.2006

- [W3C 06a] W3C WORLD WIDE WEB CONSORTIUM: *Scalable Vector Graphics (SVG)*. Version: 2006. <http://www.w3.org/Graphics/SVG/>. – Online Ressource, Abruf: 01.02.2006
- [W3C 06b] W3C WORLD WIDE WEB CONSORTIUM: *W3C Web Content Accessibility Guidelines 2.0*. Version: Januar 2006. <http://www.w3.org/WAI/GL/WCAG20/>. – Online Ressource, Abruf: 01.02.2006
- [Ware00] WARE, Colin: *Information Visualization*. San Diego, CA : Academic Press, 2000. – ISBN 1-55860-511-8
- [Wert12] WERTHEIMER, Max: *Experimentelle Studien über das Sehen von Bewegung*. In: Zeitschrift für Psychologie Vol. 61 (1912), S. pp. 161–265
- [WfMC99] WFMC WORKFLOW MANAGEMENT COALITION: *Interface 1: Process Definition Interchange Process Model*. Workflow Management Coalition. Version: Oktober 1999. http://www.wfmc.org/standards/docs/TC-1016-P_v11_IF1_Process_definition_Interchange.pdf. – Online Ressource, Abruf: 24.10.2005
- [Whit04] WHITE, Stephen A.: *Process Modeling Notations and Workflow Patterns*. In: The Workflow Handbook 2004 (2004). <http://www.omg.org/bp-corner/pmn.htm>. – Online Ressource, Abruf: 18.12.2005
- [Will94] WILLIAMS, Robin: *The Non-Designer's Design Book*. Peachpit Press, 1994
- [Will00] WILLIAMS, Thomas R.: *Guidelines for Designing and Evaluating the Display of Information on the Web*. In: Technical Communication Vol. 47 (2000), August, Nr. 3
- [yWor06a] YWORKS GMBH: *yFiles API Dokumentation (Website)*. Version: 2006. <http://www.yworks.com/products/yfiles/doc/api/>. – Online Ressource, Abruf: 07.02.2006
- [yWor06b] YWORKS GMBH: *yFiles (Website)*. Version: 2006. http://www.yworks.com/en/products_yfiles_about.htm. – Online Ressource, Abruf: 07.02.2006

Danksagung

An dieser Stelle möchte ich den zahlreichen Leuten meinen Dank aussprechen, die zum Gelingen dieser Diplomarbeit beigetragen haben. Dieser Dank gebührt besonders:

- An erster Stelle natürlich Elmar Schoch & Oliver Hagel, auf deren Vorarbeiten diese L^AT_EX-Vorlage basiert
- Meinen Reviewern und Korrektoren Elmar Schoch, Oliver Hagel, Michael Nahler, Eduard Horber & Gabriele Dempsey
- Meinen Betreuern Dr. Manfred Reichert & Dipl. Inf. Ralph Bobrik für die vielen Hinweise und anregenden Diskussionen
- Meinen Eltern, die maßgeblichen Anteil daran haben, das ich jetzt in der Lage bin, sie an dieser Stelle zu erwähnen
- Jesus Christus, ihm verdanke ich Alles, was ich bin und habe

Erklärung

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

Ulm, den 20. Februar 2006

..... (Max Moldmann)