

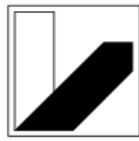
PETER C. K. WEDMANN

**AGILES GESCHÄFTSPROZESSMANAGEMENT
AUF BASIS GEBRAUCHSSPRACHLICHER
MODELLIERUNG**

DISSERTATION



**UNIVERSITÄT
BAYREUTH**



UNIVERSITÄT
BAYREUTH

Universität Bayreuth
Institut für Informatik

Agiles Geschäftsprozessmanagement auf Basis gebrauchssprachlicher Modellierung

Von der Universität Bayreuth zur Erlangung des Grades eines
Doktors der Naturwissenschaften (Dr. rer. nat.)
genehmigte Abhandlung

von

Peter Christian Karl Wiedmann
geboren in Ellwangen (Jagst)

1. Gutachter: Prof. Dr.-Ing. Stefan Jablonski
2. Gutachter: Prof. Dr. Maximilian Röglinger

Tag der Einreichung: 22. November 2016

Tag des Kolloquiums: 16. März 2017

Zusammenfassung

Die stetig steigende Digitalisierung von Kommunikation und Interaktion ermöglicht eine immer flexiblere und schnellere Erfassung und Ausführung von Aktivitäten in Geschäftsprozessen. Dabei ermöglichen technologische und organisatorische Treiber, wie beispielsweise Cloud Computing und Industrie 4.0, immer komplexere organisationsübergreifende Geschäftsprozesse. Die effektive und effiziente Einbindung aller beteiligten Menschen (z.B. IT-Experten, Endanwender) ist hierbei ein entscheidender Erfolgsfaktor. Nur wenn alle Prozessbeteiligten Kenntnis über die aktuellen Geschäftsprozesse besitzen, kann eine adäquate Ausführung dieser sichergestellt werden. Die notwendige Balance zwischen Flexibilität und Stabilität wird durch die traditionellen Methoden des Geschäftsprozessmanagements (GPM) nur unzureichend gewährleistet. Sowohl aktuelle Forschungen als auch anwendungsbezogene Studien stellen die unzureichende Integration aller Beteiligten, deren fehlendes Verständnis und die geringe Akzeptanz gegenüber GPM dar. Die Dissertation, welche im Rahmen des anwenderorientierten Forschungsprojekts „BPM@Cloud“ erstellt wird, befasst sich mit der Erarbeitung einer neuen Methode zum agilen Geschäftsprozessmanagement auf Basis gebrauchssprachlicher (alltagssprachlicher, fachsprachlicher) Modellierung von Geschäftsprozessen. Die Methode umfasst drei Bestandteile (Vorgehensweise, Modellierungssprache, Softwarewerkzeug), wodurch eine ganzheitliche Unterstützung bei der Umsetzung von GPM Projekten sichergestellt wird. Durch die Adaption und Erweiterung von agilen Konzepten der Softwareentwicklung wird die Vorgehensweise zum iterativen, inkrementellen und empirischen Management von Geschäftsprozessen beschrieben. Des Weiteren wird eine Modellierungssprache für Geschäftsprozesse entwickelt, welche zur intuitiven, gebrauchssprachlichen Erfassung von Geschäftsprozessen angewendet werden kann. Die Implementierung eines Software-Prototyps ermöglicht des Weiteren die direkte Aufnahme von Feedback während der Ausführung von Geschäftsprozessen. Die drei sich ergänzenden Bestandteile – Vorgehensweise, Sprache und Software-Prototyp – bilden eine neuartige Grundlage für eine verbesserte Erfassung, Anreicherung, Ausführung und Optimierung von Geschäftsprozessen.

Abstract

The permanent increasing digitization of communication and interaction allows a more and more flexible and fast capturing and execution of activities in business processes. Technological and organizational drivers such as Cloud Computing and Industry 4.0 allow continuously more complex cross-organizational business processes. Hereby, the effective and efficient integration of all involved people (e.g. IT professionals, end users) can be taken as a critical success factor. Only if the process specific knowledge is known by all participants the business process can be executed successfully.

Traditional methods of business process management (BPM) counter the requirement of the balance between flexibility and stability only insufficiently. Both current research and application-related studies depict the lack of integration of all participants, their lack of understanding and the general low acceptance of BPM.

The dissertation, which is a part of the user-oriented research project "BPM@Cloud", aims for the development of a new method concerning agile business process management, which is based on common language modeling of business processes. The method comprises three components (approach, modeling language, software tool). Therefore, a holistic support is ensured for the implementation of BPM projects. Through the adaption and extension of existing agile software development concepts a method for an iterative, incremental and empirical management of business processes will be described. Furthermore, a modeling language for the intuitive common language-based business process modeling will be developed. In addition, the implementation of a software prototype allows the direct capturing of feedback during the execution of business processes.

The three complementary components - approach, modelling language and software prototype - form a new basis for improved capturing, enrichment, execution and optimization of business processes.

Danksagung

Der Dank gilt allen Beteiligten, die mich bei der Erarbeitung der vorliegenden Arbeit unterstützt haben.

Besonderer Dank gilt meinem Doktorvater Herrn Prof. Dr.-Ing Stefan Jablonski für die immer offene Zusammenarbeit und die vielen hilfreichen Hinweise.

Des Weiteren danke ich Herrn Prof. Dr. Maximilian Röglinger, welcher die Zweitbegutachtung übernommen hat.

Besonders danke ich Herrn Prof. Dr. Marco Mevius, Direktor des Konstanzer Instituts für Prozesssteuerung, sowohl für die Eröffnung der Möglichkeit dieser Arbeit, als auch für die vielen prägenden Gespräche und Ratschläge.

Zudem bedanke ich mich bei Herrn Prof. em. Dr. Erich Ortner für die zahlreichen Anregungen und grundlegenden Diskussionen.

Danken möchte ich auch Florian Kurz, Fabian Meisinger und Herrn Dr. Michael Gebhart für die Zusammenarbeit und den vielseitigen Gedankenaustausch.

Weiterer Dank gilt der Geschäftsführung und allen Kollegen der AXON IVY AG für die mir erfahrene Unterstützung und die gemeinsame Arbeit.

Herzlich, für den immerwährenden Rückhalt, danke ich Sarah, meinen Geschwistern, meinen Eltern und allen Freunden.

INHALTSVERZEICHNIS

ABBILDUNGSVERZEICHNIS	IV
TABELLENVERZEICHNIS	VII
ABKÜRZUNGSVERZEICHNIS.....	IX
1. Einleitung	1
1.1. Problemstellung und Motivation	1
1.2. Zielsetzung und Beitrag	4
1.3. Aufbau der Arbeit	9
2. Management von Prozessen auf Basis der sprachbasierten Informatik.....	13
2.1. Sprachbasierte Informatik.....	13
2.1.1. Anwendungssysteme	14
2.1.2. Schema und Ausprägung als Basismodell.....	20
2.1.3. Begriffe.....	22
2.1.4. Dialogische Logik	28
2.2. Management von Prozessen.....	36
2.2.1. Geschäftsprozess	36
2.2.2. Geschäftsprozessmanagement	39
2.2.3. Geschäftsprozessmanagementsysteme	58
2.3. Existierende Ansätze aus der agilen Softwareentwicklung	62
2.3.1. Agilität	63
2.3.2. Scrum.....	67
2.3.3. Extreme Programming.....	73
2.3.4. Kanban.....	77
2.3.5. Crystal Clear	81
2.4. Agiles Geschäftsprozessmanagement.....	87
3. Gebrauchssprachliche Modellierung.....	89
3.1. Sprache.....	89

3.2.	Modell und Modellierungssprachen	97
3.2.1.	Modell.....	97
3.2.2.	Modellierungssprachen.....	99
3.3.	Begriff der gebrauchssprachlichen Modellierung.....	116
4.	BPM(N) ^{Easy} – Methode zum agilen Geschäftsprozessmanagement	128
4.1.	Methodendesign.....	129
4.2.	Entwicklung einer Vorgehensweise zum agilen GPM (Z1)	132
4.2.1.	Start und Iteration	137
4.2.2.	BPM(N) ^{Easy} Rollen	141
4.2.3.	Erfassung	145
4.2.4.	Quality Gates	158
4.2.5.	Anreicherung	165
4.2.6.	Ausführung/Analyse.....	170
4.2.7.	Optimierung.....	172
4.2.8.	Zusammenfassende Übersicht der Vorgehensweise.....	175
4.3.	Entwicklung einer gebrauchssprachlichen Modellierungssprache (Z2.1)	180
4.3.1.	Ereignisse	187
4.3.2.	Sequenzflüsse	188
4.3.3.	Aktivitäten	189
4.3.4.	Gateways	190
4.3.5.	Gebrauchssprachliche Annotation.....	192
4.4.	Werkzeug.....	194
4.4.1.	Axon.ivy BPM Suite	197
4.4.2.	BPM Touch	204
5.	Ansätze zur Unterstützung von BPM(N) ^{Easy}	207
5.1.	Entwicklung eines Prototyps zur Unterstützung des agilen GPMs (Z3)	207
5.1.1.	Technischer Aufbau des Prototypen.....	209

5.1.2.	Funktionen des Prototypen	217
5.2.	Entwicklung eines dialogische Logik-basierten BPMN ^{Easy} Pattern (Z2.2) ..	223
5.2.1.	Pattern	223
5.2.2.	Dialogische Logik-basiertes BPMN ^{Easy} Pattern	226
6.	Anwendungsbeispiel	232
6.1.	Szenario	236
6.2.	Anwendung	240
6.3.	Evaluation	255
7.	Zusammenfassung und Ausblick	257
7.1.	Zusammenfassung	257
7.2.	Kritische Betrachtung der Ergebnisse und Ausblick	261
LITERATURVERZEICHNIS		XI
EIGENE PUBLIKATIONEN		LIX
ANHANG		LXII

ABBILDUNGSVERZEICHNIS

Abbildung 1: Überblick über den Aufbau der Arbeit.....	9
Abbildung 2: Anwendungssystem in Anlehnung an Ortner [Or12] [MOW14].....	15
Abbildung 3: Sprachlogisches Kommunikationsmodell nach [Or05].....	20
Abbildung 4: Mental-realweltliche Verbindung in Anlehnung an [Or12]	21
Abbildung 5: Begriffsmodell nach Ortner [Or12].....	24
Abbildung 6: Rekonstruktionsverfahren nach Ortner [Or94][Or05, S.62]	26
Abbildung 7: Aufgliederung einer Aussage im Kontext der Dialogischen Logik	30
Abbildung 8: Darstellung des Deming-Zyklus (eigene Darstellung).....	41
Abbildung 9: Kennzahlenbasiertes Geschäftsprozessmanagement nach [Me06]	44
Abbildung 10: „Business Process Methodology“ in Anlehnung an Weske [We12].....	45
Abbildung 11: Erweiterter Geschäftsprozessmanagementzyklus nach Link [Li10]	47
Abbildung 12: Agiles Geschäftsprozessmanagement nach Silva et al. [Si09].....	50
Abbildung 13: Agiles Workflow-Management nach Weber & Wild [WW04]	52
Abbildung 14: Anforderungen an agiles Geschäftsprozessmanagement nach [Br11a] ..	53
Abbildung 15: Geschäftsprozessmanagement nach Thiemich & Puhlmann	55
Abbildung 16: Agiles Geschäftsprozessmanagement nach Çulha & Dođru.....	56
Abbildung 17: Nutzung eines GPM-Systems bei der Kundenangebotserstellung	59
Abbildung 18: Übersicht von Software zur Unterstützung des GPM nach [Sp11].....	60
Abbildung 19: Vorgehensweise in Scrum (vgl. [Gl13] [DKS13]).....	71
Abbildung 20: Beispiel einer User Story.....	72
Abbildung 21: Vorgehensmodell in XP [We00]	76
Abbildung 22: Beispiel eines Kanban Boards.....	80
Abbildung 23: Crystal Familie nach [Co02]	82
Abbildung 24: Unterteilung eines Crystal Clear Projekts (obere drei Ebenen)	84
Abbildung 25: Burn down Chart ohne Meilensteine (vgl. [Co05, S.95])	86
Abbildung 26: Schema eines Kommunikationssystems nach Shannon [Sh48]	90
Abbildung 27: Übergang in andere Sprachen nach Schienmann [Sc97].....	91
Abbildung 28: Gliederung der Gebrauchssprache nach Heinemann.....	93
Abbildung 29: Klassifizierung von Programmiersprachen (nach [Ra98]).....	94
Abbildung 30: Einfaches Modell eines Geschäftsprozesses	98
Abbildung 31: Beispiel EPK-Modell - Angebotsprozess.....	101
Abbildung 32: Beispiel eEPK-Modell – Angebotsprozess	102

Abbildung 33: Beispiel - BPMN Prozessdiagramm.....	107
Abbildung 34: Beispiel - BPMN Kollaborationsdiagramm	108
Abbildung 35: Beispiel - Choreographiediagramm	109
Abbildung 36: Beispiel - BPMN Konversationsdiagramm.....	110
Abbildung 37: Beispiel - UML Aktivitätsdiagramm.....	112
Abbildung 38: Erweiterung nach Ortner (vgl. [MOW15]).....	116
Abbildung 39: Anker der gebrauchssprachlichen Modellierung.....	121
Abbildung 40: Sprechakt - Gebrauchssprache – Modellierungssprache.....	127
Abbildung 41: Aufgaben, Probleme, BPM(N) ^{Easy} -Lösungsansätze in GPM-Projekten	128
Abbildung 42: Methodendefinition nach Ortner [Or12]	130
Abbildung 43: Darstellung der BPM(N) ^{Easy} Vorgehensweise.....	135
Abbildung 44: Beispiel einer Priorisierung in BPM(N) ^{Easy}	139
Abbildung 45: Akzeptanz von Änderungen (vgl. [Pa06 S.51ff]).....	141
Abbildung 46: BPM(N) ^{Easy} Rollenkonzept.....	144
Abbildung 47: Beispiel - Hierarchieebenen von Geschäftsprozessen.....	147
Abbildung 48: Normierung von erfassten Aussagen.....	149
Abbildung 49: Beispiel eines Schönwetterflug-Geschäftsprozessmodells	150
Abbildung 50: Beispiel Geschäftsprozessmodell mit Ausnahmen.....	151
Abbildung 51: Ausnahmen unterteilt in vier Klassen nach [Ku13]	152
Abbildung 52: Varianten der Paarmodellierung.....	157
Abbildung 53: Verfeinerung eines Qualitätsmerkmals	160
Abbildung 54: Beispiel - Verfeinerung eines Geschäftsprozessmodells	169
Abbildung 55: Beispiel eines Easy Capture Sheets.....	172
Abbildung 56: Beispiel eines Optimierungsbacklogs	174
Abbildung 57: Grafische Zusammenfassung der BPM(N) ^{Easy} Vorgehensweise.....	176
Abbildung 58: Bedarf-Fähigkeitsportfolio nach [Ro15]	178
Abbildung 59: BPMN ^{Easy} Klassendiagramm (UML).....	182
Abbildung 60: Einfaches BPMN ^{Easy} Modell (erstellt mit BPM Touch).....	186
Abbildung 61: BPMN ^{Easy} Ereignisse.....	187
Abbildung 62: Sequenzfluss in BPMN ^{Easy}	188
Abbildung 63: BPMN ^{Easy} Aktivitäten	189
Abbildung 64: BPMN ^{Easy} Gateways.....	191
Abbildung 65: Vergleich BPMN 2.0 vs. BPMN ^{Easy} -Geschäftsprozessmodell.....	191
Abbildung 66: Darstellung der Axon.ivy BPM Suite Module [AX15a].....	198

Abbildung 67: Screenshot - Axon.ivy Modeler	198
Abbildung 68: Screenshot Axon.ivy Publisher	200
Abbildung 69: Screenshot - Axon.ivy Designer.....	201
Abbildung 70: Axon.ivy Engine Architektur Übersicht nach [AX15b].....	202
Abbildung 71: Screenshot Axon.ivy Workflow Portal	203
Abbildung 72: Screenshot Axon.ivy Monitor	203
Abbildung 73: Modellierungsperspektive in BPM Touch	204
Abbildung 74: Übersicht über Detailinformationen einer Aktivität.....	205
Abbildung 75: Verwaltung der Mediafiles in BPM Touch	205
Abbildung 76: Skizze der Drei-Schichten-Architektur	211
Abbildung 77: UML Komponentendiagramm FeedbackApp.....	213
Abbildung 78: Entity-Relationship-Diagramm FeedbackApp	215
Abbildung 79: Ausschnitt FeedbackApp Prozesslogik	216
Abbildung 80: Konfiguration der FeedbackApp Datenbankbindung.....	219
Abbildung 81: Beispiel - User Dialog mit FeedbackApp Komponente.....	220
Abbildung 82: FeedbackApp User Interface - Übersicht	221
Abbildung 83: MySQL Workbench FormEditor.....	222
Abbildung 84: Loop Pattern	226
Abbildung 85: Darstellung des dialogischen Logik-basierten Patterns in BPMN ^{Easy} ..	228
Abbildung 86: Beispiel einer SWOT-Analyse	233
Abbildung 87: Darstellung des Ablaufs des vorliegenden Szenarios	236
Abbildung 88: Burn down Chart über zwei Iterationen	241
Abbildung 89: Angebotsprozess Iteration 1	244
Abbildung 90: Gebrauchssprachliche Multimediaannotationen in Iteration 1.....	245
Abbildung 91: Angereichertes Geschäftsprozessmodell.....	247
Abbildung 92: Animation des Geschäftsprozessmodell im Axon.ivy Designer	248
Abbildung 93: Aufgabenverwaltung im Axon.ivy Workflow Portal	249
Abbildung 94: User Dialog zu Aktivität "Angebot erfassen"	250
Abbildung 95: User Dialog und FeedbackApp	251
Abbildung 96: Aktualisiertes Burn down Chart.....	252
Abbildung 97: Geschäftsprozessmodell in Version 2	253
Abbildung 98: Angereichertes Geschäftsprozessmodell Version 2	254
Abbildung 99: Evaluation des Anwendungsbeispiels	255

TABELLENVERZEICHNIS

Tabelle 1: Partikelregeln der dialogischen Logik.....	31
Tabelle 2: Definition Geschäftsprozess nach Mevius	38
Tabelle 3:Werte des Agilen Manifest [Be01] (Deutsche Übersetzung)	64
Tabelle 4: Zwölf Prinzipien auf Basis des Agilen Manifest [Be01].....	65
Tabelle 5: Rollen in Scrum [SS13].....	67
Tabelle 6: Ereignisse in Scrum [SS13].....	68
Tabelle 7: Artefakte in Scrum	70
Tabelle 8: Werte in XP [BA04].....	73
Tabelle 9: Liste der XP Prinzipien [BA04]	74
Tabelle 10: Prinzipien von Kanban [An14].....	78
Tabelle 11: Allgemeine Praktiken [An14].....	78
Tabelle 12: Basisprioritäten von Crystal	82
Tabelle 13: Die sieben Eigenschaften von Crystal.....	83
Tabelle 14: Auflistung der BPMN 2.0 Basiselemente	104
Tabelle 15: Beispiele zum Mapping von XPDL zu BPMN (vgl. [Wh03]).....	114
Tabelle 16: Ausschnitt eines WS-BPEL-Geschäftsprozesses [OA07 S.21f]	115
Tabelle 17: Vier Teilakte eines Sprechakts nach Searle.....	118
Tabelle 18: Klassifikation der Sprechakte nach Searle	119
Tabelle 19: Teilaspekte von Modellqualität nach [KLS95]	124
Tabelle 20: Definition gebrauchssprachliche Modellierung	126
Tabelle 21: Fragenkatalog nach Treppler und Cockburn & Highsmith	131
Tabelle 22: Beispiele zur Erfassung und Nutzung von Mediafiles	153
Tabelle 23: Zuweisungsmöglichkeiten nach Bußler [Bu94]	154
Tabelle 24: Beispiele für Rollen und Ressourcen Organisation (vgl. [Sc04a]).....	155
Tabelle 25: Quality Gates in BPM(N) ^{Easy}	161
Tabelle 26: Funktionen und Variablen im Beispiel PAR.....	163
Tabelle 27: Art der Messung der Parameter.....	164
Tabelle 28: Interpretation der Werte	164
Tabelle 29: Erforderliche Informationen zur Anreicherung (vgl. [A114a]).....	167
Tabelle 30: Möglichkeiten der Analyse in BPM(N) ^{Easy}	171
Tabelle 31: Bewertung der Methode auf Basis des Fragenkatalogs nach [Tr12].....	176
Tabelle 32: Ausschnitt (End-Event) aus der BPMN 2.0 Semantic.xsd.....	184

Tabelle 33: BPMN ^{Easy} -XML-Ausschnitt (erstellt mit BPM Touch)	184
Tabelle 34: Verknüpfungsregeln von BPMN ^{Easy} Elementen (nach [OM11b, S.40f])..	188
Tabelle 35: Multimedia-Annotationen in BPMN ^{Easy} - Beispiele.....	193
Tabelle 36: Ausschnitt XML DataObject.....	194
Tabelle 37: Ausschnitt FeedbackApp UI Component.....	213
Tabelle 38: Ausschnitt Programmcode zur Gewichtung der Qualitätskriterien.....	216
Tabelle 39: Einbindung der FeedbackApp Komponente	218
Tabelle 40: Typen von Workflow Patterns (Workflow Patterns Initiative)	224
Tabelle 41: Beschreibung der Aktivitäten und Ereignisse des Patterns	229
Tabelle 42: Beschreibung der Persona	237
Tabelle 43: Bestandteile der Metrik CSE.....	242
Tabelle 44: Interpretation der Werte	243
Tabelle 45: Auflistung der erfassten Mediafiles.....	244
Tabelle 46: User Stories im Rahmen der Anreicherung.....	246
Tabelle 47: Übersicht der Qualitätskriterien	248
Tabelle 48: Auszug aus Easy Capture Sheet	251

ABKÜRZUNGSVERZEICHNIS

Abk.	<i>Abkürzung</i>
AM	<i>Agile Modeling</i>
BPMN	<i>Business Model and Notation</i>
BPR	<i>Business Process Reengineering</i>
bzgl.	<i>bezüglich</i>
bzw.	<i>beziehungsweise</i>
CASE	<i>Computer-aided Software Engineering</i>
CBR	<i>Case-Based-Reasoning</i>
CPO	<i>Chief Process Officer</i>
CRM	<i>Customer Relationship Management</i>
d.h.	<i>das heißt</i>
DMN	<i>Decision Modeling Language</i>
eEPK	<i>erweiterte Ereignisgesteuerte Prozesskette</i>
EPK	<i>Ereignisgesteuerte Prozesskette</i>
etc.	<i>et cetera</i>
ggfs.	<i>gegebenenfalls</i>
GPM	<i>Geschäftsprozessmanagement</i>
GPMS	<i>Geschäftsprozessmanagementsystem</i>
GPO	<i>Geschäftsprozessoptimierung</i>
HTTP	<i>Hypertext Transfer Protocol</i>
HTTPS	<i>Hypertext Transfer Protocol Secure</i>
iBPMS	<i>Intelligent Business Process Management System</i>
ISO	<i>International Organization for Standardization</i>
IT	<i>Informationstechnologie</i>
MVC	<i>Model-View-Controller</i>
Nr.	<i>Nummer</i>
OMG	<i>Object Management Group</i>
QK	<i>Qualitätskriterium</i>
S.	<i>Seite</i>
S-BPM	<i>Subjektorientiertes Business Process Management</i>
TQM	<i>Total Quality Management</i>
u.a.	<i>unter anderem</i>

UML	<i>Unified Modeling Language</i>
usw.	<i>und so weiter</i>
vgl.	<i>vergleiche</i>
WfMC	<i>Workflow Management Coalition</i>
WfMS	<i>Workflow Management System</i>
WIP	<i>Work In Progress</i>
WS-BPEL	<i>WS-Business Process Execution Language</i>
XML	<i>Extensible Markup Language</i>
XP	<i>Extreme Programming</i>
XPDL	<i>XML Process Definition Language</i>
XSD	<i>XML Schema Definition</i>
z.B.	<i>zum Beispiel</i>

1. Einleitung

Das erste Kapitel der vorliegenden Arbeit beschreibt zuerst die Problemstellung und Motivation. Danach wird die daraus abgeleitete Zielsetzung vorgestellt. Anschließend wird der Inhalt der Arbeit wissenschaftstheoretisch eingeordnet und der Aufbau beschrieben.

1.1. Problemstellung und Motivation

Die Weiterentwicklung der Vernetzung von Mensch zu Mensch, Mensch zu Maschine oder auch Maschine zu Maschine bringt stetig neue Herausforderungen für Organisationen mit sich. Aktuelle Technologien im Zeitalter der Digitalisierung, wie beispielsweise Cloud Computing¹, Big Data² und Industrie 4.0³, ermöglichen eine immer schnellere und kurzfristigere Interaktion und Synchronisation (z.B. bei der Ausführung von Aktivitäten eines Prozesses). Heutzutage stehen Informationen nahezu orts- und zeitunabhängiger allen Beteiligten (Menschen, Maschinen) zur Verfügung. Automatisierte Entscheidungsfindungen werden durch Big Data unterstützt und Prozesse und Beteiligte intelligent miteinander vernetzt.

Möglichkeiten wie diese stellen das Prozessmanagement einer Organisation vor immer neue, komplexe Aufgaben. Diese Aufgaben beziehen sich entweder auf die anzuwendende Vorgehensweise (z.B. „wie“ wird das Prozessmanagement betrieben), auf die Sprache (z.B. „welche“ Prozessmodellierungssprache wird verwendet) oder auf die Auswahl der Werkzeuge (z.B. „welche“ Softwarewerkzeuge sollen das Prozessmanagement unterstützen).

¹ Die durch das National Institute of Standards and Technology (NIST) veröffentlichte und weitverbreitete Definition [MG11] beschreibt Cloud Computing als ein Modell für schnelles Bereitstellen von Ressourcen z.B.: Speicherplatz oder Rechenleistung. Typische Merkmale für Cloud Computing sind hierbei z.B.: hohe und schnelle Skalierbarkeit.

² Der Begriff beschreibt die „wirtschaftlich sinnvolle Gewinnung und Nutzung entscheidungsrelevanter Erkenntnisse aus qualitativ vielfältigen und unterschiedlich strukturierten Informationen...“ [We15, S.13][Fi14].

³ Der Begriff Industrie 4.0 bezeichnet die intelligente Vernetzung von Maschinen und Auswertung von gesammelten Daten zur Steigerung des Wertschöpfungs- und Beschäftigungspotentials [BHV14, S.608-609]. Die Anwendung heutiger Internettechnologien innerhalb der Industrie kann als (technischer) Auslöser für die Entwicklung von Industrie 4.0 genannt werden (vgl. [DH14]).

In Bezug auf die Vorgehensweise nimmt der Bedarf nach flexibler und nachhaltiger⁴ Prozessfassung, -ausführung und -anpassung signifikant zu. Insbesondere da eine flexible und schnelle Anpassbarkeit von Prozessen zu einem immer stärker wachsendem Wettbewerbsfaktor wird [BZ11]. Darüber hinaus rückt der „Faktor Mensch“, nicht zuletzt durch aktuelle Themen wie dem „Fachkräftemangel“ [Br13a], immer stärker in den Mittelpunkt der Auswahl einer Vorgehensweise. Die unzureichende Einbindung von prozessbeteiligten Personen in das Prozessmanagement führt beispielsweise zu schnell veralteten oder (bei der Ausführung der Prozesse) nicht eingehaltenen Prozessdokumentationen [SN09]. Ein Mangel an ausreichender Kommunikationsfähigkeit (z.B. in örtlich verteilten, virtuellen Teams) des Prozessmanagements gegenüber den restlichen Beteiligten führt zum Scheitern vieler Prozessmanagementprojekten [Be13], nicht zuletzt aufgrund von subjektiven (Fehl-)Interpretationen der Beteiligten (z.B. IT-Experten) von ex ante formulierten Anforderungen (vgl. [He92]). Des Weiteren besteht das Problem [BF14], dass die Implementierung technischer Prozessapplikationen (z.B. automatisierte Prüfung einer Bestellung) häufig nicht zu den aus fachlicher Sicht modellierten Prozessmodellen passen [Ja10]. Zusätzlich besteht eine Lücke in der ganzheitlichen Steuerung von Prozessen [MJ15]. Beispielsweise werden Regelungen (z.B. definierte Verantwortlichkeiten in einem Prozess) zum Ende eines Projekts (z.B. Neugestaltung eines Prozesses) aufgehoben, anstatt - methodisch-gestützt - weiter zu erhalten. Ebenso beschreiben existierende Ansätze oftmals nur einen bestimmten Aspekt des Prozessmanagements [Aa13, S.29f].

Aus Sicht des Aspekts Sprache wird zum Beispiel nur eine neue Prozessmodellierungssprache beschrieben, es fehlen aber eine passende Vorgehensweise und ein unterstützendes Werkzeug zur betrieblichen Anwendung. Des Weiteren erfordern existierende Ansätze ausgebildete Personen, um die Regeln und Funktionen der Prozessmodellierungssprachen und deren Werkzeuge zu verstehen und anzuwenden [MM12]. Zusätzlich fehlt es an der Verständlichkeit, wobei insbesondere der Fokus auf die Einbindung aller Beteiligten (z.B. Kunde oder Mitarbeiter) liegen sollte [BRK05]. Beispielsweise ist ein Prozessmodell, welches für Maschinen lesbar und ausführbar ist,

⁴ Nach [MI13] verfolgt nachhaltiges Prozessmanagement das Ziel, die Ausführung von Prozessen sowohl in „öffentlichen Einrichtungen als auch in kommerziellen Unternehmen durch gleichberechtigte Erfassung und Überwachung aller relevanten ökologischen, ökonomischen und sozialen Kennzahlen effizienter und effektiver gestalten zu können.“

nicht zwingend ausreichend verständlich für die Interpretation durch menschliche Beteiligte [Bi14]. Studien (z.B. [Ba07][Ko10a][Be12][Pw14][Br14]) dokumentieren und spiegeln diese signifikanten Defizite wider.

In Bezug auf die Auswahl der Werkzeugunterstützung ermöglichen heutige und zukünftige (technische) Entwicklungen immer mehr neue Möglichkeiten. Insbesondere die immer stärkere Digitalisierung ermöglicht die stetig steigende Vernetzung von Menschen, Organisationen und Maschinen (vgl. [Bu15][VS15]). Zum Beispiel können Interaktionen zwischen Personen vollständig virtuell und ortsunabhängig stattfinden, sodass Durchlaufzeiten von Prozessen signifikant reduziert werden können. Zudem kann Feedback von Prozessbeteiligten, welches beispielsweise während der Ausführung des Prozesses erfasst wird, unmittelbar analysiert und zur Prozessverbesserung verwendet werden. Prozessmanagement wird somit durch den Wandel der Digitalisierung direkt beeinflusst [Be15], wie beispielsweise durch einen höheren Automatisierungsgrad von Aktivitäten in Prozessen, sodass eine Weiterentwicklung der existierenden Prozessmanagementansätze notwendig wird, um den sich ändernden Anforderungen gerecht zu werden [VS15].

Motiviert aus dieser Problemstellung und aufgrund der hohen Bedeutung (vgl. [Be15]) des Prozessmanagements für Organisationen, wird in der vorliegenden Arbeit eine Methode zum ganzheitlichen Geschäftsprozessmanagement entwickelt. Ganzheitlich beschreibt hierbei sowohl die Einführung einer Sprache zur Modellierung von Prozessen, als auch die Beschreibung einer Vorgehensweise. Zusätzlich wird ein unterstützendes Werkzeug eingeführt. Die Arbeit fokussiert insbesondere auf das Feld des agilen Geschäftsprozessmanagements (vgl. [WW04][Si09][Br11a][TP13][MSW13]). Der Begriff Agilität [Hi02a][CF04][Au05][Ve13a] bezeichnet in diesem Kontext die Fähigkeit, zum einen flexibel auf neue Anforderungen reagieren zu können, zum anderen einen hohen Grad von Stabilität zu besitzen (z.B. um möglichen Konjunkturschwankungen zu widerstehen). Zusätzlich gewährleistet die Nutzung der Gebrauchssprache [He06], dass alle Beteiligten einen durchgängigen Beitrag zum agilen Geschäftsprozessmanagement leisten und somit nachhaltig am Erfolg der Organisationen mitwirken können. Allgemeine Ziele des Geschäftsprozessmanagements, wie zum Beispiel die Verbesserung von Prozessen aus wirtschaftlicher und technischer Sicht [BMR14][SH15]), werden im agilen Geschäftsprozessmanagement auf Basis von zeitlich kurzen Iterationen und unter konsequenter Einbindung aller Prozessbeteiligten erreicht.

1.2. Zielsetzung und Beitrag

Um die dargelegte Problemstellung zu überwinden, wird eine Methode entwickelt, die Agilität ins Zentrum des Geschäftsprozessmanagements rückt. Als Grundlage dieser Methode sollen gebrauchssprachliche Konzepte dienen, sodass die Anwendungsbarriere (z.B. aufgrund von Verständnisproblemen) für alle Beteiligten eines Geschäftsmanagementprojekts signifikant verkleinert werden kann. Die Methode umfasst die Beschreibung einer agilen Vorgehensweise, Sprache und eines unterstützenden Werkzeugs im Kontext der gebrauchssprachlichen Modellierung [MOW14]. Das Akronym BPM(N)^{Easy} bezeichnet hierbei den Namen der Methode, symbolisch für ein einfaches (*easy*) Geschäftsprozessmanagement (GPM) mit der Modellierungssprache Business Process and Notation (BPMN 2.0)⁵ als Basis. BPMN ermöglicht das grafische Modellieren von Geschäftsprozessen, wobei sowohl fachliche, als auch technische Anforderungen abgebildet werden können [OM11a]. Alle Bestandteile der Methode können flexibel zum Management von Geschäftsprozessen eingesetzt werden, um vorab definierte Ziele oder Bedürfnisse [Sc04a][JN14] wie z.B. Kosteneinsparungen oder Qualitätsverbesserungen zu erreichen. Die folgende Auflistung spezifiziert die konkreten Zielsetzungen (Z1 – Z3) der vorliegenden Arbeit:

- Entwicklung einer Vorgehensweise zum agilen Geschäftsprozessmanagement (Z1, Kapitel 4.2)

In der betrieblichen Anwendung lassen sich existierende Vorgehensweisen zum Teil nur unbefriedigend einsetzen. Wird beispielsweise nur ein Teil des Geschäftsprozessmanagementzyklus unterstützt, fehlt es an Durchgängigkeit. Das bedeutet, dass häufig bestimmte Phasen in Organisationen nur unzureichend „gelebt“ werden (z.B. mangelhafte Einbindung aller Prozessbeteiligter während der Prozessplanung [DGM14]). Des Weiteren mangelt es an der einfachen Anwendbarkeit von Vorgehensweisen [Ko10a][Be12][Pw14].

Die Folgen davon sind zum Beispiel eine geringe Anwenderakzeptanz und langläufige Schulungen, um Geschäftsprozessmanagementkompetenz in Organisationen aufzubauen.

⁵ Seit Anfang 2011 liegt die Modellierungssprache in der Version 2.0 vor.

Die entwickelte Vorgehensweise berücksichtigt diese Defizite und beruht auf der Grundlage einer Anforderungsanalyse an das moderne Geschäftsprozessmanagement. Zusätzlich werden existierende Vorgehensweisen aus dem Geschäftsprozessmanagement und verwandter Disziplinen⁶ (u.a. [CH01][Hi02a][Ev03][Me06][We12][TP13]) berücksichtigt. Die Vorgehensweise fokussiert hierbei auf den Einbezug bekannter Prinzipien zum agilen Management wie (z.B.: hohe Teaminteraktion und iterative Entwicklung) und dient zur Beschreibung von Aktivitäten und Rollen innerhalb eines erfolgreich angewendeten agilen Geschäftsprozessmanagements. Anstatt starre „Maximalmodelle“ von Geschäftsprozessen zu modellieren ist das Ziel die kontinuierliche Einbindung aller Beteiligten mit der Fähigkeit agil auf Veränderungen zu reagieren [MOW15]. Die Vorgehensweise kann zum durchgängigen Management von Geschäftsprozessen eingesetzt werden und erfüllt den Anspruch einer einfachen (gebrauchssprachlichen) Verständlichkeit für alle Prozessbeteiligten.

- Entwicklung einer gebrauchssprachlichen Modellierungssprache (Z2.1, Kapitel 4.3) und eines dialogischen Geschäftsprozesspatterns (Z2.2, Kapitel 5.2)

„Die Grenzen meiner Sprache bedeuten die Grenzen meiner Welt“⁷. Dieses Zitat von Wittgenstein kann auch mit Geschäftsprozessmodellierungssprachen in Verbindung gebracht werden. Anwender und Experten aus dem Geschäftsprozessmanagement und der Informationstechnologie (IT) setzen Geschäftsprozessmodellierungssprachen ein, um effektiver und effizienter kommunizieren zu können, indem ein gemeinsames (grafisches) Modell die Grundlage der Kommunikation darstellt. In der (betrieblichen) Anwendung kommt es aber immer wieder zu Problemen (vgl. [BKO10][SS12][WGK13]). Verfügt beispielsweise ein Anwender nicht genügend Kenntnisse über die eingesetzte Modellierungssprache, können erfasste Inhalte (z.B. Aktivitätsbeschreibungen in einem Geschäftsprozess) falsch oder gar nicht verstanden werden.

⁶ Unter verwandten Disziplinen werden in der vorliegenden Arbeit Disziplinen, die direkt oder indirekt mit Geschäftsprozessmanagement in Verbindung stehen, verstanden. Zum Beispiel die Disziplin der Softwareentwicklung, welche vor allem in der Automatisierung von Prozessen eine wichtige Rolle einnimmt.

⁷ Ludwig Wittgenstein (* 1889 - † 1951) lieferte bedeutende Beiträge zur Philosophie der Logik, der Sprache und des Bewusstseins. Unter anderem zählen die „Logisch-philosophische Abhandlung“ (Tractatus logico-philosophicus) und „Philosophische Untersuchungen“ zu seinem Werk [Vo95, S.46ff].

Die in dieser Arbeit eingeführte Modellierungssprache BPMN^{Easy} zielt auf eine intuitive Anwendbarkeit. BPMN^{Easy} basiert auf dem BPMN 2.0 Standard, lässt aber im Gegensatz zu diesem nur einen eingegrenzten Elementsatz zu.

Die erlaubten Elemente [MW13] können von allen Beteiligten intuitiv gebrauchssprachlich verstanden werden. Somit wird der Grad der Einbindung aller Beteiligten durch ein einheitliches Verständnis der modellierten Geschäftsprozesse erhöht. Neben den grafischen BPMN 2.0 Elementen können zusätzlich Media-Dateien eingesetzt werden. Beispielsweise können komplexe Entscheidungswege mit Hilfe von Video- oder Audiosequenzen beschrieben werden, anstatt diese aufwendig grafisch zu modellieren. Des Weiteren wird im Rahmen dieser Arbeit ein Geschäftsprozesspattern zur Abbildung der dialogischen Logik mit BPMN^{Easy} modelliert. Die dialogische Logik [In03][Ke11] lässt im Gegensatz zur traditionellen Logik auch den Zustand des „nicht entschieden“ zu. Zusätzlich wird durch die dialogische Logik festgelegt, wie zwei Parteien (z.B. Kunde und Sachbearbeiter in einem Kundenangebotsprozess) miteinander interagieren dürfen. Pattern stellen eine Art „Schablone“ für wiederkehrende Anforderungen dar (vgl. [Ga95][RZ96][Ha05b]). Das entwickelte BPMN^{Easy} Pattern ermöglicht den Einsatz der dialogischen Logik bei der Geschäftsprozessmodellierung. Dadurch wird zusätzlich zum intuitiven Verständnis von BPMN^{Easy} eine verbesserte Modellierung von Interaktionen in Geschäftsprozessen erreicht. BPMN^{Easy} wird im BPMN 2.0 Format abgespeichert, wodurch die Kopplung zu bestehenden System ohne Probleme möglich ist.

- Implementierung eines Software-Prototyps zur mobilen Unterstützung des agilen Geschäftsprozessmanagements (Z3, Kapitel 5.1)

Entsprechend der wachsenden Anforderungen an das Geschäftsprozessmanagement, werden neue Technologien und Werkzeuge benötigt (z.B. zur Erstellung von mobilen Bedienoberflächen, Analyse von BigData, etc.), die über die etablierten Funktionen von Geschäftsprozessmanagementsystemen hinausgehen. Im Rahmen dieser Arbeit werden die zwei Werkzeuge⁸ BPM Touch und Axon.ivy BPM Suite angewendet. Mit BPM Touch [ba15] ist eine Software implementiert, welche innovative Technologien und Bedienkonzepte vereint. Es ermöglicht Anwendern Geschäftsprozesse intuitiv mobil zu

⁸ Die Nutzung der Software und die Erstellung der Geschäftsprozessapplikation erfolgt unter einer Forschungslizenz für BPM Touch und Axon.ivy.

erfassen. Des Weiteren wird mit der Axon.ivy BPM Suite [AX15a] ein umfassendes Geschäftsprozessmanagementsystem eingesetzt, welches Anwender über alle Phasen des Geschäftsprozessmanagements (z.B. fachliche Modellierung von Geschäftsprozessen, Automatisierung von Geschäftsprozessen, etc.) unterstützt.

Am Beispiel eines Kundenangebotsprozesses wird ein Software-Prototyp entwickelt, welcher zur Steigerung der Interaktion und Synchronisation aller Beteiligten (Anwender, Experten) beiträgt. Führt ein Anwender eine Aktivität in einem Geschäftsprozess aus, wird dieser während der Ausführung auf Basis von vordefinierten Fragen nach Feedback befragt (vgl. [MOW15][GMW15]). Die Fragen basieren auf zuvor definierten Qualitätskriterien. Die Ergebnisse (z.B. auf die Frage „Reichte die SOLL Durchlaufzeit der Aktivität zur Bearbeitung aus?“) können danach direkt ausgewertet und zur Verbesserung des Geschäftsprozesses genutzt werden. Die agile Vorgehensweise wird somit softwaretechnisch unterstützt. Zusätzlich wird beschrieben, wie die Softwarewerkzeuge zur Unterstützung des Geschäftsprozessmanagements genutzt werden kann.

Durch die Anwendung dieser flexiblen Methode kann mit agilem Geschäftsprozessmanagement in Kombination mit gebrauchssprachlichen Aspekten zur Schaffung einer resilienten Organisation beigetragen werden. Hamel & Välikangas [HV03] definieren vier Merkmale, welche eine Organisation benötigt, um resilient zu sein: Anpassung der Wahrnehmung, Strategie, Unternehmenspolitik und Ideologie. Ziel dabei ist es, sich innerhalb kurzer Zeit an neue, ändernde Rahmenbedingungen, z.B. neue Prozesse, anpassen zu können [Fi13].

Zur Validierung der Methode werden Projekte am Konstanzer Institut für Prozesssteuerung (kips) und mit der AXON IVY AG durchgeführt. Die Projekte dienen zur ganzheitlichen Anwendung der Methode. Zudem wird ein Anwendungsbeispiel beschrieben, welches die Durchführung eines Projekts auf Basis der agilen Geschäftsprozessmanagementmethode darstellt.

Insgesamt leistet die vorliegende Arbeit einen individuellen Beitrag in Bezug auf den Fortschritt des agilen Geschäftsprozessmanagements. Als innovative Methode, welche aus Vorgehensweise, Sprache und Werkzeug besteht, ermöglicht BPM(N)^{Easy} einen vielfältigen und intuitiven Einsatz. Die Vorgehensweise basiert auf einem konstruktiven Ansatz und somit der Idee das einzelne Schritte fortlaufend, schrittweise und zirkelfrei

durchlaufen werden. In Verbindung mit dem in dieser Arbeit entwickelten Rollenkonzepts können Prozessbeteiligte durchgängig aktiv in agile Geschäftsprozessmanagementprojekte eingebunden werden. Als weiterer Bestandteil der Methode beschreibt die Modellierungssprache BPMN^{Easy} eine intuitive grafische Modellierungssprache, welche durch multimediale Inhalte, wie z.B. Bilder und Videosequenzen, annotiert werden kann. BPMN^{Easy} adaptiert hierbei den weitverbreiteten Modellierungsstandard BPMN 2.0 und dessen Basiselemente. Zudem unterstützt der eigens entwickelte Software-Prototyp bei der digitalen Erfassung von Anwenderfeedback, sodass implementierte Geschäftsprozesse unmittelbar an die geänderten Anforderungen adaptiert werden können.

1.3. Aufbau der Arbeit

Die vorliegende Arbeit gliedert sich in sieben Kapitel (vgl. Abbildung 1). Kapitel eins bis drei dienen zur Einführung, Erarbeitung und Beschreibung der für diese Arbeit relevanten Themen. In Kapitel vier und fünf wird eine Methode zum agilen Geschäftsprozessmanagement entwickelt. Das Kapitel sechs beschreibt die Validierung der Methode, bevor in Kapitel sieben eine Zusammenfassung und ein Ausblick die Arbeit abschließen.

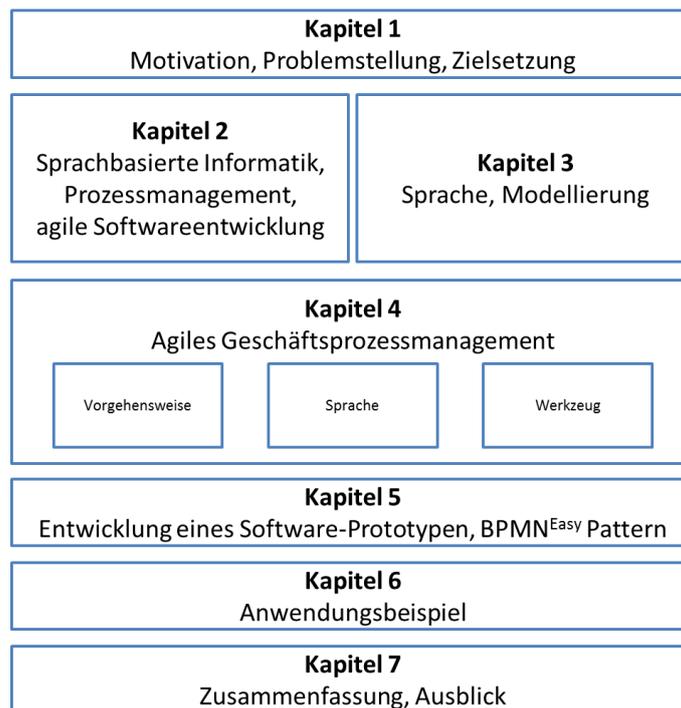


Abbildung 1: Überblick über den Aufbau der Arbeit

Das zweite Kapitel beschreibt Grundlagen zum Management von Prozessen auf Basis der sprachbasierten Informatik. Im ersten Teil dieses Kapitels wird der Begriff der sprachbasierten Informatik eingeführt [Or05]. Insbesondere wird das Modell eines Anwendungssystems nach Ortner diskutiert [Or12][MOW15] und grundlegende Themen wie das Schema-Ausprägungs-Modell [OW04], der Terminus des Begriffs [Le99][DI11] und die Dialogische Logik [In03][Ke11] beschrieben. Grundlegende Definitionen und Ansätze zu Prozessen und deren Management sind unter anderem Gegenstand des Kapitels, wobei speziell die Disziplin des Geschäftsprozessmanagements anhand existierender Arbeiten (u.a.[ATW03][WW04][Me06][Si09][We12][TP13]) diskutiert wird. Zudem wird der Begriff des Geschäftsprozessmanagementsystems [Ka95][Be09][DKK14] beschrieben und deren Bedeutung im Kontext des

Geschäftsprozessmanagements verdeutlicht. Abschließend werden existierende Definitionen des Begriffs der Agilität vorgestellt, agile Ansätze aus der Softwareentwicklung [Be00][Co05][An10][SS13][Co05] beschrieben und die Eigenschaften des agilen Geschäftsprozessmanagements dargestellt.

In Kapitel drei wird der Begriff der gebrauchssprachlichen Modellierung [MOW14][MOW15] definiert. Der erste Abschnitt des dritten Kapitels beschäftigt sich mit dem Begriff der Sprache. Insbesondere die für das Geschäftsprozessmanagement wichtigen Sprachen (Gebrauchssprache, Modellierungssprache und Programmiersprache) werden diskutiert. Zusätzlich wird die Sprechakttheorie nach Searle [Se69] beschrieben und dargestellt, dass gelungene Sprechakte die Basis einer erfolgreichen gebrauchssprachlichen Modellierung sind. Des Weiteren wird der Begriff des Modells diskutiert und allgemeine Anforderungen an die Modellierung von Geschäftsprozessen analysiert. Hierbei wird die Problematik im Umgang mit Begriffen erörtert. Beispielsweise kann es bei der Erfassung von Geschäftsprozessmodellen zu Begriffsdefekten, wie beispielsweise der Homonymie kommen (z.B. „Kiefer“, zum einen ein Begriff für einen Baum, zum anderen ein Begriff für ein Teil des (menschlichen) Körpers). Im zweiten Teil des Kapitels werden die grundlegenden „Anker“ (Interaktion, Synchronisation, Qualität) der gebrauchssprachlichen Modellierung beschrieben. Beispielsweise wird der Begriff der Interaktion erörtert und die Bedeutung dieser in der gebrauchssprachlichen Modellierung dargestellt. Abschließend wird die gebrauchssprachliche Modellierung als eigener Begriff definiert.

Das Kapitel vier umfasst die Beschreibung der agilen Methode BPM(N)^{Easy}. Die Bestandteile der Methode orientieren sich nach der Methodendefinition von Ortner [Or12]. Nach einer Beschreibung der Zielsetzung der Methodenentwicklung werden die Bestandteile einer Methode (Sprache, Vorgehensweise und Werkzeug) beschrieben. BPM(N)^{Easy} adaptiert und erweitert existierende Ansätze aus der (agilen) Softwareentwicklung und dem Geschäftsprozessmanagement. Durch die Adaption und Erweiterung der agilen Ansätze wird eine Vorgehensweise zum iterativen, inkrementellen und empirischen Geschäftsprozessmanagement implementiert. Des Weiteren wird eine Sprache zur multimedialen Modellierung von Geschäftsprozessen entwickelt, mit dem Ziel einer intuitiven und gebrauchssprachlichen Erfassung. Die eingeführte Sprache (BPMN^{Easy}) basiert hierbei auf der Modellierungssprache BPMN 2.0 [OM11a]. Die

Entwicklung des Software-Prototyps ermöglicht zudem die direkte Aufnahme von Feedback während der Ausführung von Geschäftsprozessen.

Die drei sich ergänzenden Bestandteile ermöglichen eine neuartige Art einer verbesserten Erfassung, Anreicherung, Ausführung, Analyse und Optimierung von Geschäftsprozessen.

In Kapitel fünf wird die Implementierung eines Software-Prototyps beschrieben. Hierzu werden die Werkzeuge BPM Touch [ba15] und Axon.ivy BPM Suite [AX15a] angewendet. BPM Touch wird zur mobilen Erfassung von Geschäftsprozessen eingesetzt. Axon.ivy BPM bietet eine Plattform, welche zur Automatisierung von Geschäftsprozessen dient. Mithilfe des Software-Prototyps kann Feedback unmittelbar bei Anwendern während der Ausführung von Aktivitäten erfasst werden. Vordefinierte Qualitätskriterien in Form von Fragen (z.B. „Ist die SOLL-Durchlaufzeit für die Ausführung der Aktivität ausreichend?“) dienen hierbei als Grundlage. Im zweiten Teil des Kapitels fünf wird auf Basis von BPMN^{Easy} ein Pattern entwickelt, welches die Anwendung der dialogischen Logik-Regeln innerhalb der Ausführung von Geschäftsprozessen ermöglicht. Das Pattern ermöglicht eine detaillierte und „faire“ Modellierung von Interaktionen in einem Geschäftsprozess.

In Kapitel sechs wird auf Basis verschiedener Projekte, in welchen im Rahmen dieser Arbeit mitgewirkt wurde (u.a. zur Validierung der BPM(N)^{Easy}-Methode), ein Anwendungsbeispiel beschrieben. Die Projekte fanden während der Tätigkeiten bei der AXON IVY AG und dem Konstanzer Institut für Prozesssteuerung statt. Beispielsweise wurde im Projekt „Smart City Konstanz“ die Modellierungssprache BPMN^{Easy} zur Modellierung von Administrationsprozessen in der öffentlichen Verwaltung angewendet. Eine Szenariobeschreibung leitet in das BPM(N)^{Easy}-Anwendungsbeispiel ein. Die Gliederung des Abschnitts richtet sich hierbei nach den Schritten der BPM(N)^{Easy}-Methode, sodass die Durchführung „entlang“ des agilen Zyklus (d.h. Erfassung, Anreicherung, Ausführung/Analyse, Optimierung) beschrieben werden kann.

Das Kapitel sieben bildet den Abschluss der Arbeit. Nach der Zusammenfassung aller vorherigen Kapitel werden die Ergebnisse kritisch diskutiert. Anschließend werden mögliche weiterführende Forschungsfragen und darauf aufbauende zukünftige Arbeiten beschrieben.

Die vorliegende Arbeit wurde unter Berücksichtigung der Methode des Erlanger Konstruktivismus erarbeitet und kann zur konstruktiven Informatik [Or05] zugeordnet werden. Einer der Grundsteine dieser Methode, das Prinzip „Aus der Praxis für die Praxis“ [Lo87] ist Grundlage dieser Arbeit. Das bedeutet, motiviert aus der Praxis, ist die Zielsetzung dieser Arbeit, für eben jene mehrwertstiftende Ergebnisse zu erzielen. Unter Mehrwert kann hier beispielsweise die Steigerung der Flexibilität, Senkung der Kosten, Optimierung von Abläufen oder Erhöhung der Effizienz einer Organisation gesehen werden. Die Arbeit ist Teil der Forschungsaktivitäten „BPM@Cloud“, welche am Konstanzer Institut für Prozesssteuerung durchgeführt werden (vgl. [MSW12][MSW13][MOW14][MOW15]).

2. Management von Prozessen auf Basis der sprachbasierten Informatik

Das zweite Kapitel erläutert die Grundlagen der sprachbasierten Informatik. Anschließend wird auf das Management von Prozessen eingegangen.

2.1. Sprachbasierte Informatik

Als Ursprung der sprachbasierten Informatik ist vor allem die Arbeit von Kamlah⁹ und Lorenzen¹⁰ zu nennen. Kamlah und Lorenzen gelten als Väter des Erlanger Konstruktivismus¹¹, welcher auch den Rahmen der sprachbasierten Informatik bildet. Kamlah und Lorenzen beschreiben einen Ansatz mit dem systematisch eine fehlerfreie Wissensrekonstruktion und Wissensbegründung möglich ist [EI09]. Alle Regeln und Elemente werden dabei voraussetzungsfrei, zirkelfrei und nachvollziehbar eingeführt [BM04]. Beispielsweise darf keine Aussage formuliert und bewiesen werden, indem die Aussage selbst als Voraussetzung angenommen wird¹². Von welcher Bedeutung das Verständnis der sprachbasierten Informatik ist, zeigt unter anderem die Forderung von Wedekind, Ortner und Inhetveen in ihrem Beitrag zum Thema „Informatik als Grundbildung“ (vgl. [WOI04, S.51f]). Die Autoren beschreiben die Notwendigkeit, die „Sprachlogik in der Informatik“ als fundamentales Unterrichtselement (z.B. an allgemeinbildenden Schulen) einzubinden. Insbesondere die immer enger werdende „Verknüpfung“ zwischen Mensch und Maschine (z.B. ständiges „Präsent-sein“ und Interagieren mit Hilfe von Applikationen auf mobilen Endgeräten) fordert eine Grundbildung aller Anwender (z.B. zur fehlerfreien Formulierung von Anforderungen an ein Anwendungssystem).

⁹ Wilhelm Kamlah (* 1905 – † 1976) war Gelehrter der Musikwissenschaft, Geschichte, Theologie und Philosophie, vgl. [Mö97, S.105].

¹⁰ Paul Lorenzen (* 1915 – † 1994) war Gelehrter der Mathematik und Philosophie, vgl. [GM10].

¹¹ Der Erlanger Konstruktivismus (auch als Methodischer Konstruktivismus bezeichnet [Pö15, S.30ff.]) wurde in der sogenannten Erlanger Schule von Kamlah und Lorenzen entwickelt [He06, S.37]. Das Hauptziel dieser Wissenschaftstheorie ist es, Handlungsweisen aus der Praxis zu verstehen und dadurch schrittweise methodisch begründet und zirkelfrei eine Wissenschaft zu entwickeln (vgl. „Logische Propädeutik“ [KL96]).

¹² Ein solches Vorgehen (eine Aussage, durch die Aussage selbst zu begründen) wird auch als „Zirkelschluss“ bezeichnet, vgl. [Kr12a, S.64f].

Die Aufgabe der sprachbasierten Informatik ist es, die erfassbare Welt durch (menschliche) Sprache für Rechner¹³ zugänglich zu machen [He06]. Damit diese Rechner zur Unterstützung im täglichen Leben beitragen können, ist eine genaue Rekonstruierbarkeit jeglichen Sprachtyps vorausgesetzt. Beispielsweise müssen umgangssprachlich formulierte Aussagen eines Anwenders für einen Softwareentwickler verständlich gemacht und in einer Rechner-verständlichen Sprache programmiert werden. Einzelne Bestandteile eines Anwendungssystems¹⁴, wie zum Beispiel Software, dienen „hier nicht der möglichst originalgetreuen Abbildung von Weltausschnitten, sondern der Bewältigung (fach-)praktischer Probleme“ [Jo01, S.2]. Ein wechselseitiges Verständnis von Anwendern und (IT/BPM)-Experten ist beispielsweise die Grundlage einer erfolgreichen Implementierung von Anwendungssystemen.

In den folgenden Abschnitten werden (für diese Arbeit) relevante „Bausteine“ der sprachbasierten Informatik näher beschrieben.

2.1.1. Anwendungssysteme

Der Begriff des Anwendungssystems ist in der Literatur nicht einheitlich definiert und abgegrenzt¹⁵. Beispielsweise umfasst nach Stahlknecht & Hasenkamp ein Anwendungssystem (im engeren Sinne) die Anwendungssoftware und die Anwendungsdaten. Des Weiteren schließt der Begriff (im weiteren Sinne), die für die Anwendung notwendigen Basissysteme (Hardware, Systemsoftware und Infrastruktur) mit ein [SH11]. Krcmar [Kr05] beschreibt ein Anwendungssystem als ein System, welches Informationen automatisiert verarbeitet ohne das nicht-automatisierte Aufgabenträger (z.B. Menschen die manuell Arbeit verrichten) eingeschlossen sind. Ferstl & Sinz [FS06] nennen die Erweiterung von Rechner- und Kommunikationssysteme um System- und Anwendungssoftware (z.B. Software zur Ausführung einer automatischen Bonitätsprüfung eines Kunden) ein Anwendungssystem. Ortner [Or05][Or12][MOW14] nutzt den Begriff des Anwendungssystems nicht nur zur Beschreibung eines IT-Systems in Form von Soft- und

¹³ Ein Rechner (engl. Computer) ist eine Funktionseinheit zur „Verarbeitung und Aufbewahrung von Daten. Verarbeitung umfasst die Durchführung mathematischer, umformender, übertragender und speichernder Operationen.“ [DI88].

¹⁴ Vgl. Abschnitt 2.1.1

¹⁵ Vgl. Krcmar & Schwarzer: „(...)Der Begriff Informationssystem ist in der Literatur nicht eindeutig definiert und wird teilweise synonym mit dem Begriff Anwendungssystem verwendet.(...)“ [SK10, S.9] und [AM09][LLS10].

Hardware, sondern beschreibt damit einen ganzheitlichen Ansatz (vgl. [Fi13, S.15ff]). Ganzheitlich bedeutet, dass sich im Zusammenhang mit der Anwendungssystementwicklung nicht nur mit den technischen Systemen, sondern auch mit deren Kontext befasst wird. Hierbei dient der konstruktive Ansatz „aus der Praxis, für die Praxis“ (vgl. [Lo87]) als Basis [El09]. Ein Anwendungssystem besteht nach Ortner aus mehreren Ebenen (Träger, Hardware, Daten, Programme, Wissen, Menschen und Handlungen) und ist von einem Ökosystem umgeben. Die Bestandteile des Anwendungssystems sind zu einem großen Anteil virtuell (sprachlich) (z.B. eine mit einer Programmiersprache implementierte Software) [Or12]. Die Abbildung 2 stellt diese Ebenen dar.



Abbildung 2: Anwendungssystem in Anlehnung an Ortner [Or12] [MOW14].

In Abbildung 2 sind sowohl reale, als auch mentale/digitale Ebenen visualisiert. Das umgebende Ökosystem ist hierbei ständig in Bewegung und Veränderung (biologisch, physikalisch, psychologisch,...) und besitzt einen hohen sprachlichen Anteil [Or12] (z.B. sprachliche Handlungen in Form von Diskussionen zwischen Anwender). Im Folgenden werden die einzelnen Ebenen (von oben nach unten, vgl. Abbildung 2) beschrieben:

- Handlungen

Lorenzen & Schwemmer nennen die „Befolgung einer Aufforderung“ (z.B. „Telefoniere bitte mit dem Kunden!“) eine Handlung [LS75]. Handlungen beschreiben die reale Nutzung des entwickelten Anwendungssystems durch den Menschen [MOW14]. Dabei werden strategische Zielsetzungen durch Prozesse¹⁶ in operative Handlungen umgesetzt [Ob14]. Ein Verkaufsprozess eines Produkts kann beispielsweise eine Diskussion über den Preis des Produkts enthalten. Durch eine geschickt geführte (Sprach-)Handlung (z.B. Präsentation der Vorteile des zu erwerbenden Produkts) kann dabei der Verkäufer zur Erreichung der strategischen Zielsetzungen (z.B. Gewinnerhöhung) beitragen. Während der Begriff einer Handlung das Tun von etwas bzw. ein Verhalten (z.B. das Zusammenbauen einer Maschine) beschreibt (vgl. [Ar05, S. 30ff]), definiert die Sprachhandlung¹⁷ das Äußern durch Sprache (z.B. eine Bitte, welche an einen Kommunikationspartner gerichtet ist). Es wird somit immer etwas mit Sprache „getan“, auch wenn der Kommunikationspartner nicht anwesend ist (z.B. Tagebuch) [La09]. Unabhängig vom Handlungstyp wird mit einer (Sprach-)Handlung immer eine Absicht verfolgt [Pe97].

- Menschen

Eine weitere Ebene (vgl. Abbildung 2) beschreibt die Einbindung des Menschen in die Anwendungssystemarchitektur. Beispielsweise müssen die Geschäftsprozesse, die mit dem Anwendungssystem verbunden sind, strukturiert erfasst sein, um ein aufeinander abgestimmtes Handeln zu ermöglichen [Ob14]. Nur so ist es möglich, dass vordefinierte Zeitpunkte der Informationsübergabe (z.B. Versendung eines befristeten Angebots an einen Kunden) eingehalten werden, sodass keine Probleme (z.B. Informationsverlust, Zeitverzögerungen, etc.) auftreten. Der „Menschzentrierung“ der sprachbasierten Informatik folgend (vgl. Abschnitt 2.1) soll der Mensch bei der Bewältigung des täglichen Lebens unterstützt werden. Anwender und deren Perspektiven werden deshalb in den Entwicklungsprozess aufgenommen [Ma01], wodurch eine höhere Menschzentrierung (z.B. in Form von „angenehmer“ nutzbarer Software) erreicht wird.

Neben dem Anwender können weitere Rollen in dem Ökosystem des Anwendungssystems definiert und mit Aufgaben und Verantwortlichkeiten belegt

¹⁶ Der Prozessbegriff wird in Abschnitt 2.2.1 ausführlich diskutiert.

¹⁷ Vgl. des Weiteren Abschnitt 3.3: „Exkurs Sprechakttheorie“

werden. Beispielsweise sind IT-Experten für die Implementierung der Anwenderorientierten Software zuständig¹⁸.

- Wissen

Allgemein wird Wissen als die Verbindung von Daten und Informationen beschrieben. Das Wissen ist dabei immer an Menschen gebunden. Nach Prange [Pr02, S.25] ist „Wissen Gegenstand jeder theoretischen Reflexion und jedes praktischen Handelns“. Werden zum Beispiel zur Ausführung fachlicher Aktivitäten Regeln genutzt (z.B. gesetzliche Vorschriften), ist es notwendig, dass das Wissen über diese Regeln während der Anwendungssystementwicklung berücksichtigt wird [Fi13]. Paulzen definiert Wissen als „die Summe aller Kenntnisse, die ein Individuum über seine Umwelt auf der Basis vergangener Wahrnehmungen und Erfahrungen sowie daraus abgeleiteter Schlussfolgerungen angesammelt hat. Wissen umfasst weiterhin die Fähigkeiten eines Kollektivs, die aus der gemeinsamen Ausübung der individuellen Fähigkeiten resultieren.“ [Pa06, S.31]. Des Weiteren kann der Wissensbegriff in individuelles bzw. kollektives, abstraktes bzw. konkretes und implizites bzw. explizites Wissen unterteilt werden [Pa06, S.29]. Individuelles Wissen (z.B. über die Behandlung eines Sonderfalls bei der Ausführung eines Geschäftsprozesses) kann durch Nutzung von gemeinsamen Strukturen (z.B. Einbindung anderer Beteiligter) mit kollektivem Wissen „erweitert“ werden. Zusätzlich wird zwischen dem impliziten Wissen (Wissen einer Person) und dem explizitem Wissen (das Wissen nach „außen“ zugänglich) unterschieden. Zum Beispiel dient ein BPMN Geschäftsprozessmodell zur (expliziten) Dokumentation des impliziten Wissens über mögliche Sonderfälle. Das konkrete Wissen, wie beispielsweise das Wissen eines IT-Experten zu Implementierung einer Software, wird von anderen Beteiligten als abstraktes Wissen aufgenommen.

Anwender der Software kennen normalerweise keine Details über die verwendeten Algorithmen, können aber dennoch (mit Hilfe des abstrakten Wissens) die Software beschreiben und bedienen.

- Programme und Daten

¹⁸ Im Abschnitt 4.2 werden die für diese Arbeit relevanten Rollen beschrieben.

Programme beschreiben Produkte, die digital zur Verfügung stehen (z.B.: Datenbanken, Betriebssysteme, ...). Bei der Implementierung und/oder Eingliederung dieser Elemente in ein Anwendungssystem muss besonders auf die Kompatibilität der Elemente zueinander geachtet werden [Fi13]. Trotz vorhandener Standards führt eine hohe Heterogenität der Programme und Daten oft zu Kommunikations- oder Schnittstellenproblemen (z.B. unterschiedliche Datenformate, unbekannte Dateitypen,...). Programme beschreiben dabei eine, mit einer Programmiersprache¹⁹ definierte Abfolge von Befehlen und sind nach einer Übersetzung auf einem Rechner als „Software“ ausführbar. Hierbei wird zwischen Basis –und Anwendungssoftware²⁰ unterschieden. Basissoftware bezeichnet Software, die zum Beispiel zum Betrieb eines Rechners oder zur Erstellung von Anwendungssoftware genutzt wird [Or94]. Anwendungssoftware kann wiederum in Standardsoftware (z.B. Standard-Textverarbeitungssoftware) und Individualsoftware (z.B. Kundenspezifische Software zur Verwaltung von Kundendaten) untergliedert werden [Me12b]. Daten²¹ bezeichnen „zum Zwecke der Verarbeitung zusammengefasste Zeichen“ [AM09, S.9], die dadurch Einfluss auf alle überliegenden Ebenen nehmen. Beispielsweise können fälschlich gespeicherte Produktdaten, sowohl das Wissen eines Produktverkäufers, als auch die ausgeführten Handlungen, negativ beeinflussen (z.B. durch den Verkauf eines Produkts, welches nicht den Anforderungen des Kunden entspricht).

¹⁹ Eine ausführliche Beschreibung von Programmiersprachen ist unter Abschnitt 3.1 zu finden.

²⁰ Synonym zu Basis –und Anwendungssoftware wird in der Literatur auch Systemsoftware (vgl. [Me12b]) bzw. Applikationssoftware (vgl. [Or94]) verwendet.

²¹ Für eine ausführliche Diskussion und Abgrenzung des Datenbegriffs sei auf [Bi04, S.7][AM09, S.8ff] [Me12a, S.25ff] verwiesen.

- Hardware und Träger

Hardware und Träger²² bezeichnen die unterste Ebene der Anwendungssystemarchitektur (vgl. Abbildung 2). Hierbei bezeichnen Träger, Sachen, wie beispielsweise Gegenstände des alltäglichen Gebrauchs, Gebäude oder sogar Menschen selbst [El09]. Die Träger sind physisch mit der Hardware verbunden [Fi13] und beeinflussen wiederum die Auswahl und den Einsatz der Hardware. Auswahlkriterien für den Verbund zwischen Träger und Hardware können beispielsweise die Größe, Leistung oder das Gewicht sein. Befindet sich das Anwendungssystem beispielsweise in einem mobilen Ökosystem, ist die Auswahl eines Kraftfahrzeugs als Träger und ein Navigationsgerät als Hardware möglich. Der Begriff der Hardware beschreibt alle materiellen Komponenten (z.B. Speicherchips, Prozessoren, ...), die zu einem Rechner (z.B. Desktop-Computer, Laptop, Smartphone, etc.) gehören [AM09][LLS10][GS13].

Insgesamt ist eine unabhängige Betrachtung der einzelnen Ebenen in der betrieblichen Anwendung kaum möglich. Über die oberste Ebene, die Handlungs- bzw. Prozessebene, werden vielmehr ebenenübergreifend Abläufe miteinander verbunden. Um auf mögliche (ggfs. parallel auf verschiedenen Ebenen auftretenden) Änderungen resilient reagieren zu können, wird im Rahmen dieser Arbeit eine agile Geschäftsprozessmanagementmethode beschrieben.

²² Synonym zu Technologieträger vgl. [Or12][MOW14].

2.1.2. Schema und Ausprägung als Basismodell

Zur Implementierung und Pflege von Anwendungssystemen ist es nötig die Lebenswelt rekonstruieren zu können [WOI04]. Beispielsweise kann dadurch die Fachsprache aus den Anwendungsbereichen der Anwendungssysteme konsistent verwaltet und implementierte Lösungen (automatisch) auf Korrektheit geprüft werden [Or95].

Das Begriffspaar Schema-Ausprägung spielt auch hier – wie in der gesamten Informatik²³ – eine hervorgehobene Rolle. Nach Lorenz [Lo90] stellt „ein Schema einen universellen (allgemeinen), eine Ausprägung einen singulären (besonderen) Aspekt eines Gegenstands (Objekt) einer Rede dar“. Nach Ortner „verkörpert“ das Schema Wissen, wobei mit Ausprägungen Informationen zum Ausdruck gebracht werden [Or04]. Zum Beispiel kann ein Geschäftsprozess in einer Software abgebildet sein und als Schema aufgefasst werden. Die Ausführung und Erzeugung von Instanzen gilt dann als Ausprägung [WOI04].

Sollen unterschiedliche Teilnehmer in Kommunikation treten, ist es notwendig, dass die Teilnehmer über standardisierte Schemata verfügen, um die Ausprägung des anderen verstehen zu können. Anderenfalls ist keine (bidirektionale) Kommunikation möglich. Die Abbildung 3 zeigt das zugehörige sprachlogische Kommunikationsmodell.

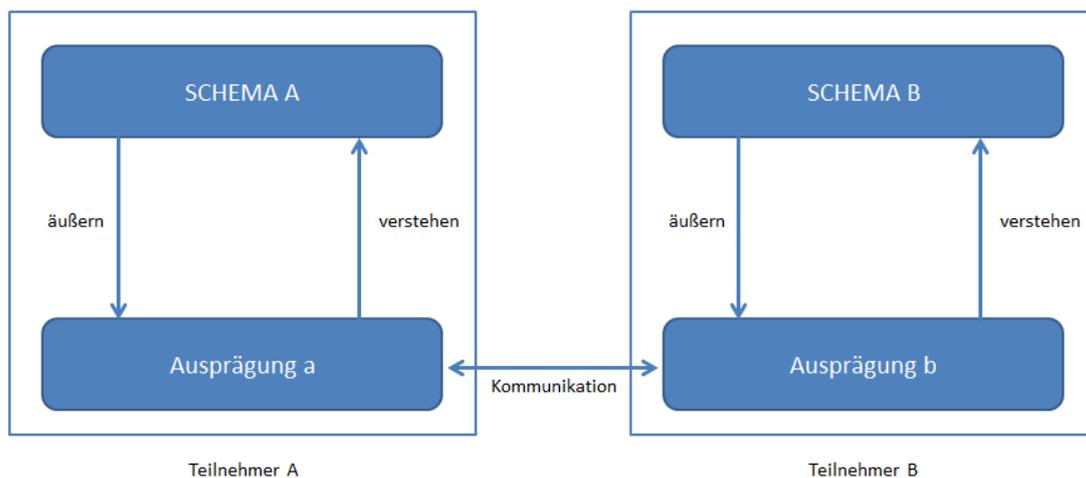


Abbildung 3: Sprachlogisches Kommunikationsmodell nach [Or05]

Abbildung 3 beinhaltet neben Schemata und Ausprägungen die Äußerungsbeziehung des „Äußerns“ und „Verstehens“ (vgl. vertikale Pfeile in Abbildung).

²³ Zum Beispiel gelten Klassen und Relationen als Schemata, Objekte und Tupel als Ausprägungen. Vgl. Beitrag zu „Informatik als Grundbildung“ [WOI04].

Diese dienen zur Verbindung und Rekonstruktion von Schema und Ausprägung. Nur wenn ein Schema vorhanden ist, kann eine entsprechende Information (Ausprägung) geäußert werden. Komplementär dazu kann eine Information ausschließlich verstanden werden, wenn das passende Schema verfügbar ist [Or04]. Während der Modellierung eines Anwendungssystems können zudem „generische Sprachhandlungstypen“ und „spezifische Sprachhandlungstypen“ eingeführt werden [Or05], um basierend auf der eingesetzten Gebrauchssprache Anforderungen zu verstehen und zu implementieren. Wird in Hinblick auf die Kommunikation der „physische Bereich“ mitberücksichtigt (Gegenstände, Träger, etc.) erhöht sich die Komplexität der Interaktion²⁴ zwischen den Kommunikationsteilnehmern. Die Abbildung 4 visualisiert den Zusammenhang, zwischen der Ebene der Sprachhandlungen und der Ebene der physischen Handlungen. Als Beteiligte eines solchen „Systems“ bestimmt Ortner [Or05] die Sprachhandelnden (z.B. Programmierer), die Handlungen selbst (z.B. Erstellung eines Angebots) und die Handlungsergebnisse (z.B. Ergebnis einer Anfrage).

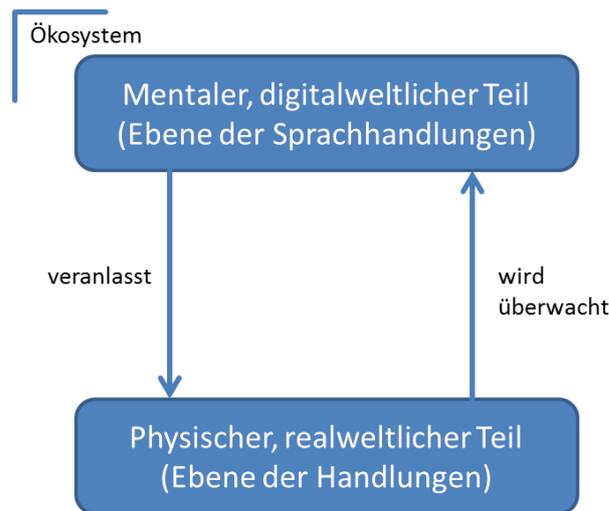


Abbildung 4: Mental-realweltliche Verbindung in Anlehnung an [Or12]

Eingebunden in ein vorgegebenes Ökosystem (Abbildung 4) veranlassen Sprachhandlungen auf mentaler, digitalweltlicher Ebene, Handlungen, die im realweltlichen Teil ausgeführt werden. Die realweltliche Ausführung wird danach durch die medialweltliche Ebene überwacht (vgl. [Or12]). Beispielsweise wird die Anweisung „Verpacke Paket zum Versand“ veranlasst, physisch durchgeführt und mental (z.B. durch Beobachtung) überwacht (vgl. Pfeilrichtungen in Abbildung 4).

²⁴ Der Begriff der Interaktion wird in Abschnitt 3.3 diskutiert.

Zur Bestimmung der Veranlassungen und der Ausführung einer Handlung in der Realwelt kommt nach Ortner [Or12] die dialogische Logik zum Einsatz. Die dialogische Logik [In03][Ke11] lässt neben den traditionellen Werten „wahr“ und „falsch“ auch den Zustand „nicht entschieden“ zu, sodass beispielsweise auf der Ebene der Sprachhandlungen der Prozess einer kollaborativen Lösung eines Problems (z.B. durch Diskussion) möglich ist²⁵.

In Bezug auf das agile Geschäftsprozessmanagement, in welchem die intensive Kommunikation zwischen allen Beteiligten ein essentieller Bestandteil der Vorgehensweise ist, ist die Sprachkompetenz ein entscheidender Faktor. Schemata helfen den Beteiligten dabei Aussagen (Ausprägungen) zu verstehen [Hr06]. Dieses gegenseitige Verständnis unterstützt die korrekte Ausführung von veranlassten Handlungen (z.B. Modellierung eines Geschäftsprozesses). Des Weiteren lassen sich anhand von Aussagen wiederum Schemata ableiten [HR06]. Besitzen die Kommunikationsteilnehmer einen unterschiedlichen Grad an Sprachkompetenz, kann dies zur Erhöhung der Fehleranfälligkeit führen. Zum Beispiel können durch Missverständnisse falsche Aktivitäten oder Beschreibungen zu Geschäftsprozessmodellen hinzugefügt werden [MOW14]. Wohlrapp [Wo12, S.31] beschreibt zusätzlich Zustände „in denen unklare, mangelhafte, fehlende Orientierung das Handeln blockiert“. Enthalten zum Beispiel Geschäftsprozessmodelle nicht alle Informationen, welche zur Ausführung nötig sind (z.B. die Verantwortlichkeit für die Ausführung einer Aktivität ist nicht definiert), können „Orientierungslücken“ auftreten.

2.1.3. Begriffe

Begriffe befinden sich auf einer abstrakten Ebene und beziehen sich auf einen Gegenstand in der Welt [Le99]. Der Einteilung von Ortner folgend kann ein Gegenstand in vier Unterebenen aufgegliedert werden. Erstens können Gegenstände in Verbindungen und Komponenten aufgeteilt werden. Zweitens gliedern sich Komponenten wiederum in Beschaffenheit und Träger auf. Drittens unterteilen sich Träger in Geschehnisse und Dinge.

²⁵ Im Kapitel 2.1.4 wird die dialogische Logik ausführlich beschrieben.

Schließlich werden viertens Geschehnisse in Vorgänge, Handlungen und Verhalten unterteilt [Or97, S.84-85]. Diese Gegenstandseinteilung dient als Grundlage für den Satzbau mit geklärten Begriffen [El09].

Beispiele:

DING – VERBINDUNG – BESCHAFFENHEIT: Aktivität hat Bearbeitungsdauer.

DING – VERBINDUNG – DING: Geschäftsprozessmodell dokumentiert Bestellabwicklung.

Begriffe sind also nötig um überhaupt eine Kommunikation (vgl. Abbildung 3) zu ermöglichen und Gegenstände und deren Bedeutung und Zusammenhänge zu verstehen. Die Bedeutung eines Begriffs wird hierbei durch dessen Intension und Extension ausgedrückt. Die Intension beschreibt die „Gesamtheit der Merkmale eines Begriffs“ [DI11]. Zwei Begriffe haben die identische Intension, wenn diese die gleichen semantischen Merkmale besitzen (z.B. Geige und Violine) [Le99]. Die Extension setzt sich aus der „Gesamtheit der Unterbegriffe eines Begriffs auf derselben Hierarchiestufe“ (Klasse) und dem Begriffsumfang zusammen [DI11]. Beispielsweise gehören zu dem Begriffsumfang des Begriffs „Software“, Textverarbeitungssoftware, Kalkulationssoftware und Präsentationssoftware. Der Begriffsumfang ist hierbei abhängig von der Hierarchiestufe. Dagegen gehören zur Klasse alle Gegenstände, die die Merkmale des Begriffs „Software“ aufweisen. Hierzu zeigt die Abbildung 5 das auf Frege²⁶ zurückgehende Begriffsmodell [Or12].

²⁶ Gottlob Frege (* 1848 - † 1925) zählt zu einem der großen Vordenker (neben Russell und Wittgenstein) der analytischen Philosophie. Als sein wichtigstes Werk werden seine Aufsätze „Funktion und Begriff“, „Über Sinn und Bedeutung“ und „Über Begriff und Gegenstand“ gezählt [Kr01][FT02, S.2-60].

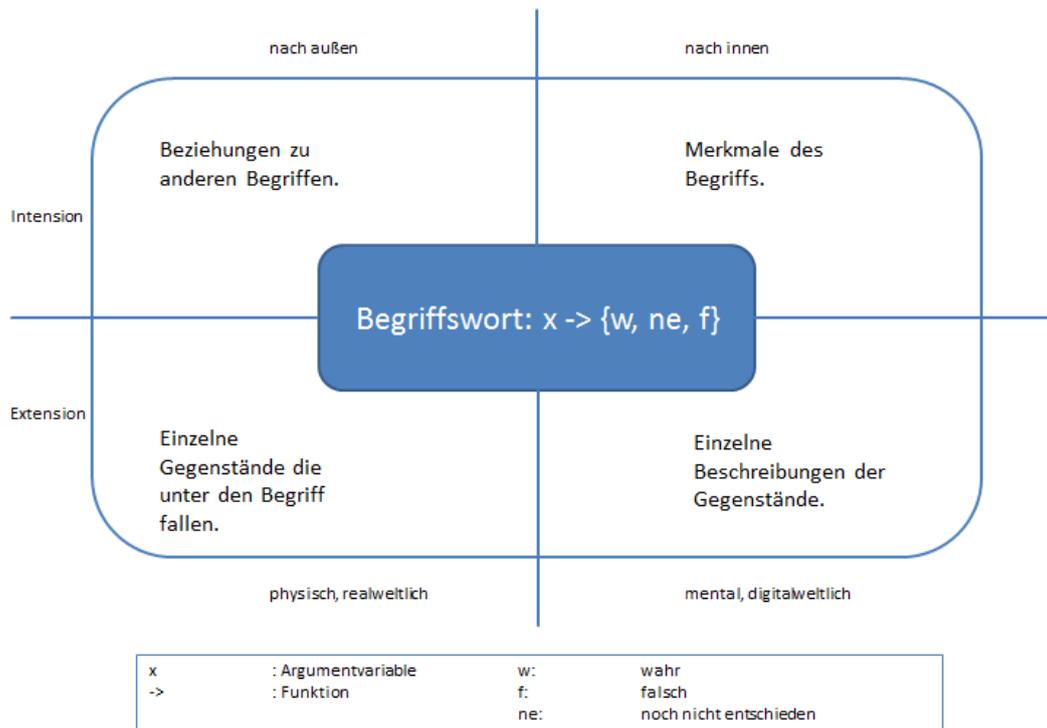


Abbildung 5: Begriffsmodell nach Ortner [Or12]

Die vier Quadranten in Abbildung 5 unterteilen Intension und Extension noch einmal in einen „nach außen/physischen“ und „nach innen/mentalen“ Teil.

Die Betrachtung der Intension zeigt, dass mögliche Verbindungen zu anderen Begriffen durch die „Intension nach außen“ (physisch, realweltlich) beschrieben werden. Die Merkmale eines Begriffs beziehen sich nach innen und sind mental bzw. digital („Intension nach innen“). Die Extension eines Begriffs beinhaltet mögliche Gegenstände („nach außen“) und die dazugehörigen einzelnen Beschreibungen („nach innen“). In Bezug auf das in Abschnitt 2.1.2 beschriebene Basismodell von Schema und Ausprägung, kann die Extension als Ausprägung und die Intension als Schema bezeichnet werden. Sowohl Intension, als auch Schema stellen die deskriptive Bedeutung dar, welche „nicht an die Dinge in der Welt gebunden sind“, während Extension und Ausprägung „alle Dinge in der außersprachlichen Welt“ umfassen (vgl. [Me07a]).

Des Weiteren (vgl. Abbildung 5 Mitte) können Begriffe und deren Begriffsworte als Funktion beschrieben werden. Diese Funktionen können „wahr“, „falsch“ oder noch

„nicht entschieden“²⁷ als Ergebnis haben und beschreiben die Zuordnung eines Gegenstands zu einem Begriff [MOW14].

Besteht keine Eindeutigkeit eines Begriffs muss die Bedeutung geklärt werden. Folgende Begriffsdefekte können im Allgemeinen unterschieden werden [Or97, S.32][Or05][He06]:

- Äquipollenz
Dieselben Objekte (Extension) werden durch die Betrachtung aus verschiedenen Blickwinkeln (Intension) unterschiedlich bezeichnet. Beispielsweise beschreibt der „Lagerbestand“ eine mengenmäßige und das „Warenkonto“ eine wertmäßige Rechnung über einen Artikelbestand.
- Homonymie
Homonyme bezeichnen gleiche Begriffsworte (gleich Laut- und Schriftform), welche eine unterschiedliche Intension und Extension haben. Zum Beispiel bezeichnet das Begriffswort „Kiefer“, zum einen Nadelholzgewächs, zum anderen einen Teil des (menschlichen) Körpers.
- Synonymie
Synonyme haben dieselbe Bedeutung (Extension und Intension). Die Begriffswörter können daher gegeneinander ausgetauscht werden, wie zum Beispiel „Sonnabend“ und „Samstag“.
- Vagheit
Ist keine klare intensionale Definition eines Begriffs vorhanden, können Unklarheiten bzgl. den unter den Begriff fallende Objekte (Extension) auftreten. Beispielsweise ist nicht eindeutig geklärt, ob mit „Gehalt“ das „Brutto-“, oder „Nettogehalt“ beschrieben ist.
- Falsche Bezeichner
Falsche Bezeichner besitzen eine andere Bedeutung als diese, die durch die Bezeichnung suggeriert wird. Zum Beispiel „Glühlampe“ anstatt „Glühbirne“ [El09, S.123].

Die Klärung und somit auch die Rekonstruktion und Geltungssicherung der Intensionen sind in Bezug auf die Anwendungsentwicklung, wie zum Beispiel der Entwicklung von

²⁷ Der Zustand des „nicht entschieden“ wird im Abschnitt 2.1.4 (Dialogische Logik) ausführlich diskutiert.

Software zur Unterstützung des Prozessmanagements von besonderer Wichtigkeit, um z.B.: Fachbegriffe für Dinge oder Geschehnisse eindeutig nutzen zu können [Or97]. Die Defekte müssen hierzu systematisch aufgedeckt (Synonym, Äquipollenz, falsche Bezeichner) oder direkt behoben werden (Homonym, Vagheiten) [Fi13]. Eine Einteilung möglicher Rekonstruktionsverfahren ist in Abbildung 6 dargestellt.

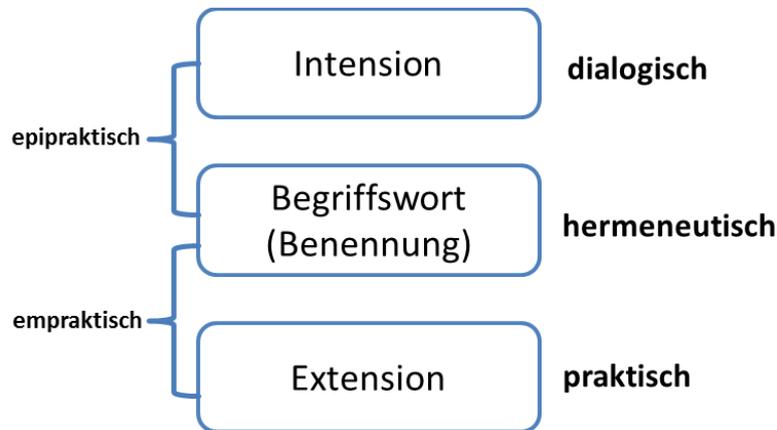


Abbildung 6: Rekonstruktionsverfahren nach Ortner [Or94][Or05, S.62]

Dialogische Verfahren bezeichnen den „gesprochenen“ Austausch von Informationen, beispielsweise während Diskussionen, Besprechungen oder Interviews.

Als hermeneutisch²⁸ werden die Verfahren beschrieben, welche zur Textinterpretation dienen. Als Beispiele können das „Studieren“ von Fachliteratur oder die Analyse von Fragebögen genannt werden. Auf der Ebene der Extension werden praktische (konstruktivistische, vgl. [HOS04, S.97-98]) Verfahren angewendet. Hierbei finden physisches Handeln und sprachliche Ausformulierung „gleichzeitig“ statt. Beispielsweise dienen Experimente oder Beobachtungen zur Rekonstruktion eines Begriffs aus „Sicht“ der Extension. Die in Abbildung 6 dargestellten Rekonstruktionsverfahren (dialogisch, hermeneutisch, praktisch) sind dabei nicht überschneidungsfrei. Zum Beispiel kann das hermeneutische Verfahren auch zur Rekonstruktion der Intension oder Extension beitragen. Neben der Unterscheidung zwischen dialogisch, hermeneutisch und praktisch

²⁸Die Hermeneutik befasst sich mit dem Verstehen. Ursprünglich diente die Hermeneutik zur Interpretation von "klassischen" Texten (z.B. theologische Texte) [Ca86]. Aufgabe ist es, „(...)Begrifflichkeiten und Korrektheitsintuitionen bzgl. des Verstehens und Interpretierens zu klären und entsprechende Begrifflichkeiten und Kriterien einzuführen, sodann aber auch Verfahren zu entwickeln, die zu Interpretation führen, die diesen Kriterien genügen(...)“ [Re14, S.3].

werden die Verfahren entweder als epipraktisch (nach einer erfolgten Praxis) oder empraktisch (die Praxis begleitend) bezeichnet (vgl. Abbildung 6) [Or05].

Die Verfahren der Rekonstruktion sind nicht auf ein sequentielles Vorgehen beschränkt, vielmehr wird der Prozess iterativ (mit vielen Rücksprüngen) durchgeführt und so „das Präzise aus dem Vagen“ gewonnen [Le99]. Für einen weiteren Überblick zu Begriffen, Gegenstandseinteilung und Rekonstruktion sei auf [Or97][Le99][Or05] und [Fi13] verwiesen.

Das iterative Vorgehen findet sich als Parallele im agilen Geschäftsprozessmanagement wieder. Hierbei liegt zum einen der Schwerpunkt auf einer schnellen Erfassung von neuen Geschäftsprozessen. Zum anderen wird auf eine flexible Kommunikation ohne aufwendige Dokumentationsverfahren fokussiert. Hierdurch wird eine stetige Sicherstellung der Korrektheit von Begriffen, durch vordefinierte Qualitätskontrollen (vgl. Quality Gates in Abschnitt 4), welche iterativ durchlaufen werden, zwingend notwendig, um z.B. Missverständnisse zwischen den Prozessbeteiligten zu vermeiden. Des Weiteren sind im Geschäftsprozessmanagement Texte häufig Grundlage für die Verständniskerngewinnung (z.B. textuelle Beschreibung von Geschäftsprozessen), sodass ein hermeneutisches Verfahren (Verstehensprozess) von besonderer Wichtigkeit ist [We07]. Wird das Geschäftsprozessmanagement zusätzlich durch Softwarewerkzeuge unterstützt (z.B. durch ein Workflow-Management-System, vgl. Abschnitt 2.2.3), ist die Korrektheit von Begriffen für die Mensch-Maschine-Kommunikation zwingend notwendig, um beispielsweise maschinelle Fehler (z.B. Programmabbruch) zu vermeiden.

2.1.4. Dialogische Logik

Einer der Hauptaspekte des Erlanger Konstruktivismus (vgl. Abschnitt 2.1) ist nach Petersen [Pe97, S.4ff] die Erkenntnisgewinnung mit Hilfe einer dialogischen Methode. Unter diesem Aspekt setzt die dialogische Logik²⁹ die antike griechische Tradition der systematischen Untersuchung von Dialogen fort [Ke11]. Im Rahmen dieser Untersuchung werden Aussagen (z.B. München liegt in Bayern) überprüft, mit dem Ziel eine gemeinsame Entscheidung bzgl. der vorherigen Aussage zu erzielen (z.B. dass München in Bayern liegt, ist wahr). Der Begriff „dialogische Logik“ setzt sich hierbei aus „Dialog“ und „Logik“ zusammen, wobei es sich um keine neue Logik handelt, sondern um einen Versuch der dialogischen Begründung von „klassischer“ und „intuitionistischer“ Logik [LL78]. Der Dialogbegriff wird in der Literatur unterschiedlich definiert. Weigand unterscheidet zwischen zwei Ebenen des Begriffs. Zum einen beschreibt der Begriff des Dialogs „traditionell“ einen formalen situativen Sprecherwechsel (z.B. zwei Menschen diskutieren über einen Geschäftsprozess). Zum anderen beschreibt Weigand eine inhaltliche Dialogebene, bei welcher der Kommunikationspartner nicht präsent ist (z.B. eine schriftliche Kommunikation) [We03, S.5f u. 34]. Des Weiteren gehören nach Ostasz [Os88, S.107] drei Bestandteile zu einem Dialog: (1) „(...)eine anfängliche Mitteilung und einen gegenseitigen Austausch(...)“, (2) „(...)ein solches Gespräch, das ein weiteres Aufschließen oder Ausarbeiten der Sache bringen kann (...)“, (3) „(...) ein umfassender Sinn, der zwischen der Sache und den Teilnehmern des Dialogs vermittelt und dabei selbst, vor allem verstandesmäßig, vermittelt werden kann(...)“. Ein Dialog entsteht folglich aus dem „Ansporn“, dass das Zusammenwirken von Denken und Sprache³⁰ zu einem Gesuchten führt (z.B. Preisverhandlung zwischen einem Anbieter und einem Kunden).

²⁹ Mittelstraß formuliert des Weiteren zur dialogische Logik: „Wenn die Vernunft, von der Philosophen so gerne sprechen, eine dialogische Natur besitzt, dann ist die Dialogische Logik ihr unmittelbarer theoretischer (...) Ausdruck. In diesem Sinne wäre die Dialogische Logik nicht nur ein theoretischer Einfall unter anderen Logikeinfällen, sondern ein Konzept, das der Logik ihre praktische und darüber hinaus ihre anthropologische Bedeutung zurückgewinnt.“ [Mi15, S.10].

³⁰ In Abschnitt 3.1 wird der Begriff der Sprache ausführlich diskutiert.

Die Logik, der zweite Bestandteil des Begriffspaar der dialogischen Logik, wird in der Literatur beispielsweise von Lipps [Li13, S.1] als „die Lehre von den Formen und Gesetzen des Denkens“ beschrieben. Nach Görz befasst sich die Logik „(...) also mit den Regeln der rationalen, zweckgerichteten Argumentation³¹ und somit insbesondere mit den Regeln des korrekten Schliessens.“ [Gö05, S.1]. Inhetveen [In03, S.11ff] stellt die Logik als Hilfsmittel einer Begründungspraxis dar. Demnach müssen Behauptungen bestimmte Kriterien als Geltungsgründe besitzen, um somit möglichen Zweifeln zu widerstehen. Zusammenfassend sei auf Beckermann verwiesen, welcher Logik als die Fähigkeit des verbesserten Argumentierens beschreibt. Des Weiteren ist es durch die Logik möglich, das Argumentieren lehrbar zu machen und den „reflektierten Umgang mit den Prinzipien und Kriterien des Argumentierens zu erhöhen.“[Be11b, S.2]. Die dialogische Logik ist hierbei als „Spiel“ aufzufassen, bei dem es darum geht, die Argumente des Dialogpartners (Mitspielers) zu widerlegen [LL78] oder die Behauptung zu bestätigen.

Im Gegensatz zu anderen Logikansätzen³² ist die dialogische Logik nicht rein monologisch und nicht-monoton. Der beschränkte Bereich der operativen Logik wird durch das dialogische Prinzip überwunden (vgl. [Lo09]). Bereits gewonnene Schlussfolgerungen können durch neue Erkenntnisse aufgehoben werden. Beispielsweise geht der dialogische Ansatz über eine reine Formalisierung von Erkenntnisprozessen hinaus und lässt „die algebraische Strukturierung von interaktiven Entscheidungsprozessen“ [Pe97, S.13] zu. Zudem ist die dialogische Logik eine materiale Logik, d.h. eine „(...) Logik, die sich nicht nur [...] mit formalen Schlüssen aus monologisch gesetzten Vordersätzen befasst, sondern auch und vor allem mit der inhaltlichen Begründung von dialogisch bestimmten Behauptungen (...)“ [Gr82, S.210]. Der Ansatz dient zur klaren, rationalen Ordnung von unregelmäßigen (Streit-)Gesprächen [St86, S.469]. Ein Dialog wird dabei von zwei Parteien durchgeführt: dem Proponent und dem Opponent [In03]. Der Proponent verfasst die (zusammengesetzte) Behauptung (z.B. „Die Namen sind Kuno Lorenz und Paul Lorenzen.“) und trägt diese vor.

³¹ Zur Definition des Begriffs der Argumentation bzw. des Arguments sei z.B. auf Beckermann [Be11b, S.4] verwiesen: „Ein Argument ist eine Folge von Aussagesätzen, mit der der Anspruch verbunden ist, dass ein Teil dieser Sätze (die Prämissen) einen Satz der Folge (die Konklusion) in dem Sinne stützen, dass es rational ist, die Konklusion für wahr zu halten, falls die Prämissen wahr sind.“

³² Für eine weitere Übersicht anderer Logiken (Modale Logik, Temporale Logik, etc.) sei auf Schenke [Sc13b] verwiesen.

Wie in Abbildung 7 dargestellt, werden die Aussagen (welche Behauptungen enthalten), solange untergliedert, bis eindeutig überprüfbare Elementaraussagen verfügbar sind. Eine Elementaraussage ist die kleinste sprachliche Einheit, der entweder der Wahrheitswert „wahr“ oder „falsch“ zugewiesen werden kann.

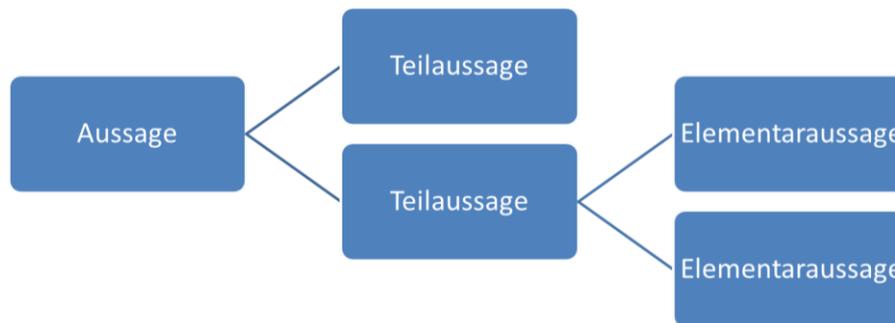


Abbildung 7: Aufgliederung einer Aussage im Kontext der Dialogischen Logik

Der Opponent kann danach Zweifel oder Einwände vortragen (z.B. Zweifel über die Korrektheit der Namen). Jeder Dialog endet nach einer endlichen Anzahl von Zügen.

Das bedeutet, dass zu einem bestimmten Zeitpunkt von keinem der Dialogpartner ein Argument mehr genannt werden kann. Eine Behauptung eines Proponenten ist daher genau dann gültig, wenn eine formale Gewinnstrategie für diese Behauptung vorhanden ist [RR98]. Eine formale Gewinnstrategie ist vorhanden, sofern eine Behauptung unabhängig vom Opponenten immer gewonnen werden kann [In03]. Neben Proponent und Opponent werden mit der dialogischen Logik auch bestimmte Regeln beschrieben. Diese Regeln sind in Rahmen- und Partikelregeln gegliedert [Rü01]. Die Rahmenregeln definieren die „allgemeinen Bestimmungen“ eines Dialogspiels. Die Partikelregeln definieren, welche Züge des Angriffs und der Verteidigung einer Behauptung erlaubt sind.

- Rahmenregeln (vgl. [Rü01][In03][Ke11][Lo15a])
 1. Start: Der Proponent beginnt den Dialog mit einer initialen Behauptung. Jeder darauffolgende Zug ist entweder ein Angriff oder eine Verteidigung.
 2. Verzögerung: Das Dialogspiel erlaubt keine Verzögerungstaktiken. Das bedeutet, dass nur Züge erlaubt sind, die die Dialogsituation verändern.
 3. Formal: Der Proponent darf nur Elementaraussagen beanspruchen, welche bereits durch den Opponent beansprucht wurden. Behauptungen, welche

keine anderen Behauptungen enthalten, werden Elementaraussagen genannt (vgl. [In03, S.13f]). Beispielsweise ist die zusammengesetzte Behauptung “Es ist kalt, weil Schnee liegt” keine Elementaraussage (vgl. Abbildung 7).

4. Gewinn: Der Proponent oder Opponent gewinnt das Dialogspiel, sobald der Gegenspieler am Zug ist, aber keinen weiteren Zug mehr durchführen kann (weder Angriff, noch Verteidigung).
5. Intuitionistisch/klassisch: Die Dialogführung ist in zwei unterschiedlichen Wegen möglich. Mit dem „intuitionistischen Weg“ ist definiert, dass die Verteidigungspflicht in der umgekehrten Reihenfolge ihrer Entstehung einzulösen ist [In03, S.40]. Alternativ kann der „klassische Weg“ zur Dialogführung genutzt werden, wodurch Angriffe in beliebiger Reihenfolge verteidigt werden können.

- Partikelregeln

Die Partikelregeln definieren, welche Angriffe und Verteidigungen in einem Dialogspiel erlaubt sind. Die Tabelle 1 stellt die formale Beschreibung der Regeln dar.

Tabelle 1: Partikelregeln der dialogischen Logik

	$\neg, \wedge, \vee, \rightarrow, \forall, \exists$	Angriff	Verteidigung
1	$\neg A$	A	(keine Verteidigung)
2	$A \rightarrow B$	A	B
3 ¹	$A \wedge B$?L(inks) ?R(rechts)	A B
3 ²	$A \vee B$?	A B
4 ¹	$\forall x A$? _n	A[n/x]
4 ²	$\exists x A$?	A[n/x]

Die Partikelregeln aus Tabelle 1 haben keine Verbindung zu dem Kontext einer Argumentation, sondern dienen als abstraktes Regelwerk [Ke11].

Zur Formulierung dieser Regeln stehen unterschiedliche Notationen zur Verfügung³³. Beispielsweise werden durch das logische Zeichen \vee Behauptungen miteinander in Form

³³ Einen Überblick über häufig angewendete Notationen ist in [In03, S.223] zu finden.

eines „oder“ verbunden (z.B. „wähle Weg A oder Weg B). Des Weiteren wird ein Gegenangriff mit einem Fragezeichen ? gekennzeichnet. Folgend werden die Zeilen der Tabelle 1 kurz beschrieben:

1. Das Negationszeichen \neg negiert eine formulierte Behauptung. Weil durch eine Negation selbst eine Behauptung gemacht wird, sind nur (Gegen-)Angriffe möglich und keine Verteidigung.
2. Die Subjunktion \rightarrow definiert, dass falls die Behauptung A wahr ist, auch die Behauptung B zutrifft. Die Behauptung $A \rightarrow B$ ist nur dann falsch, wenn A wahr ist und B falsch. Subjunktionen werden meist durch „wenn..., dann...“ formuliert.
3. Zwei weitere Junktoren (neben \neg , \rightarrow) sind die Adjunktion und Konjunktion. Ein Adjunktor \vee verbindet zum Beispiel zwei Elementaraussagen als „oder“. Ein Konjunktore \wedge hingegen verbindet als „und“. Die Kombination des Fragezeichens ? mit L oder R zeigt an, welchen Teil – links oder rechts – der Behauptung angegriffen werden soll.
4. Quantoren können als „Platzhalter“ angesehen werden. Hat der Proponent das Recht den „Platzhalter“ zu ersetzen wird das Zeichen \exists zur Beschreibung genutzt und Existenzquantor genannt. Darf der Opponent den „Platzhalter“ befüllen, wird das Zeichen \forall angewendet und der Ausdruck Allquantor genannt. Zum Einsatz kommen Quantoren immer dann, wenn es sinnvoll ist den „Platzhalter“ (also die Variable, z.B. $\exists xA$) durch Eigennamen zu ersetzen. Beispiel: Bei „ $\exists xA$ ist krank“ behauptet ein Proponent, dass irgendjemand krank wäre. Es kann nun gefordert werden (durch das Einsetzen eines Eigennamens) diesen „irgendjemand“ zu nennen (z.B. „Peter ist krank.“) (vgl. [In03, S.45ff]).

Neben den beschriebenen Dialogpartnern (Proponent, Opponent) und den Rahmen- und Partikelregeln bietet die dialogische Logik einen besonderen Umgang mit dem Wahrheitswert³⁴. Das klassische „tertium non datur“, also „die strenge, logische Formulierung, jede Aussage ist entweder wahr (richtig) oder falsch (unrichtig)“ [Os88, S.112], wird in der dialogischen Logik „erweitert“. Dialoge können im Rahmen eines Dialogspiels gewonnen werden, ohne dass der Proponent die Wahrheitswerte aller Elementaraussagen kennen muss [Gö05]. Dazu unterscheidet die dialogische Logik zwischen den Wahrheitswerten „wahr“, „falsch“ und dem Zustand „nicht entschieden“³⁵. Lorenzen (vgl. [Lo80]) beschreibt den Zustand als notwendig, da ansonsten daraus geschlossen werden könnte, dass jede Frage (Behauptung) geklärt werden könnte. Zu beachten ist, dass die mangelnde Kenntnis, also das noch nicht Entschiedene, kein dritter Wahrheitswert ist [Gö05]. Stekeler-Weithofer [St86, S.407f] beschreibt beispielsweise, dass es im Allgemeinen nicht möglich ist zu überprüfen, ob bestimmte Eigenschaften bei einer beliebigen Zahl in der unendlichen Zahlenreihe gegeben sind (es könnte immer noch eine Zahl geben, die die Eigenschaften erfüllt). Inhetveen [In03, S.63] nutzt das Beispiel von „außerirdischem Leben“. Derzeit kann die Behauptung es gäbe solches, weder als wahr noch als falsch befunden werden – die Behauptung ist also „unentschieden“.

Der konstruktive Ansatz der dialogischen Logik erhöht die Möglichkeiten der Kreativität bei Entscheidungsprozessen (vgl. [FGO10]), sodass diese eine Vielzahl von Einsatzmöglichkeiten in verschiedenen Gebieten (z.B. Philosophie, Mathematik,...) aufweist. Fach [Fa97] beschreibt den Einsatz von Dialogspielen im Rahmen der Mensch-Maschine-Interaktion. Die Dialogpartner sind in diesem Fall zum einen der Anwender, zum anderen eine Maschine. Je nach Anwenderaktion (Aktion als Proponent) z.B. Eingabe einer neuen Überschrift in einem Textverarbeitungsprogramm, reagiert die Software (Opponent) unterschiedlich z.B. mit einer automatischen Aktualisierung des Inhaltsverzeichnisses.

³⁴ Es soll an dieser Stelle nicht diskutiert werden, wann etwas als die „Wahrheit“ oder „wahr“ benannt werden darf. Nach [In03, S.59] ist diese Diskussion „alt und oft beantwortet“. In dieser Arbeit wird die Definition von Inhetveen angewendet. Demnach werden Behauptungen als „wahr“ bezeichnet, die es erlauben, dass ein Proponent gegen jeden Opponent gewinnt (vgl. [In03, S.61]).

³⁵ „Nicht entschieden“ wird auch als „non liquet“ bezeichnet (lateinisch für „es ist noch ungeklärt“) (vgl. [Lo80]). Der Begriff wird insbesondere in der Rechtswissenschaft verwendet. Beispielsweise kann bei Rechtsfragen ein „non liquet“ auftreten, welches „(...) die Überzeugung vom Erreichen des Abwägungsmaßes fehlt, also das Gegenteil der Überzeugung(...)“ beschreibt (vgl. [Ri06, S.114ff]).

Da in der dialogischen Logik immer alle Züge (Handlungen aller Spiele) berücksichtigt werden, eignet sich diese für die Modellierung solcher Aufgabestellungen [Fa97, S.4ff].

Insbesondere wird die Transparenz bei kritischen Entscheidungsfindungsprozessen (z.B. Diskussion von Prozessbeteiligten) erhöht. Im Rahmen dieser Arbeit wird ein Pattern (Entwurfsmuster) auf Basis der dialogischen Logik entwickelt (vgl. [Ku15]). Durch die Abbildung der dialogischen Regeln kann die „faire“ und transparente Interaktion zwischen zwei Dialogpartnern sichergestellt und somit eine erfolgreiche Ausführung von Aktivitäten in einem Geschäftsprozess unterstützt werden (z.B. die Diskussion zwischen Anbieter und Kunden über den Preis eines zu bestellenden Produktes). Vor allem die Ausführung von manuellen Aktivitäten, welche in einer frühen Iteration innerhalb des agilen Geschäftsprozessmanagements noch nicht vollständig ausmodelliert werden konnten, profitiert von der Anwendung des dialogischen Logik-Patterns. Für eine weitere ausführliche Diskussion der dialogischen Logik sei auf [In03][St86, S.425ff][LL78][Rü11][Ke11][MB15] verwiesen.

Zusammenfassend bietet das mit der sprachbasierten Informatik einhergehende Entwicklungsparadigma hinsichtlich dem agilen Geschäftsprozessmanagement eine Möglichkeit, die Sprachfähigkeit aller Beteiligten (z.B. Anwender und IT-Experten bei der Entwicklung eines Anwendungssystems) zu verbessern (vgl. [EI09, S.18ff]) und die somit vorher definierten Ziele (z.B. ein intuitiv nutzbares Anwendungssystem) effektiv und effizient zu erreichen. Das agile Geschäftsprozessmanagement, welches insbesondere die Handlungs- bzw. Prozessebene aus Sicht der Anwendungssystementwicklung betrifft (vgl. Abschnitt 2.1.1) unterstützt bei der flexiblen Entwicklung von ganzheitlichen Anwendungssystemen. Zudem finden sich die einzelnen Ansätze der sprachbasierten Informatik im agilen Geschäftsprozessmanagement wieder bzw. können als Grundlage angesehen werden. Trotz der zunehmenden Digitalisierung von Geschäftsprozessen und automatisierten Entscheidungsfindung, sind menschliche Dialoge weiterhin von hoher Bedeutung. Insbesondere entscheidungsintensive Geschäftsprozesse erfordern sowohl bei der Erfassung als auch Ausführung transparente und nachvollziehbare Dialoge zwischen den Prozessbeteiligten. Auf Basis der dialogischen Logik können Dialoge zwischen Proponent und Opponent gesteuert werden (vgl. Abschnitt 2.1.4). Dialogische Logik-basierte Dialoge führen hierbei zu einem, für beide Parteien (Proponent, Opponent), fairen und transparenten Ergebnis. Zudem ist das Modell von Schema und Ausprägung (vgl. Abschnitt 2.1.2) Basis für das gegenseitige

Verständnis. Schemata helfen den Beteiligten Aussagen (Ausprägungen) zu verstehen [Hr06]. Dieses gegenseitige Verständnis unterstützt die korrekte Ausführung von veranlassten Handlungen (z.B. Modellierung eines Geschäftsprozesses). Zusätzlich ist die iterative Vorgehensweise, welche zur Klärung von Begriffen (vgl. Abschnitt 2.1.3) angewendet werden kann, eine weitere Grundlage des agilen Geschäftsprozessmanagement. Der agile Ansatz fokussiert auf eine flexible und intensive Kommunikation ohne aufwendige Dokumentationsverfahren. Hierdurch wird eine stetige iterative Sicherstellung der Korrektheit von Begriffen notwendig, damit alle Prozessbeteiligten das gleiche Verständnis erreichen können.

2.2. Management von Prozessen

Der Zielsetzung dieser Arbeit folgend fokussiert die nachfolgende Erläuterung insbesondere auf die Begriffe Geschäftsprozess und Geschäftsprozessmanagement³⁶.

2.2.1. Geschäftsprozess

Die Literatur bietet eine Vielzahl von Definitionen bzgl. der Begriffe des Prozesses und des Geschäftsprozesses³⁷.

Aus Sicht der Natur- und Sozialwissenschaften bezeichnen Prozesse gerichtete Abläufe von Geschehendem [BM04]. Jablonski beschreibt Prozesse als Ansammlung von Aktivitäten, bei denen die Überführung von Input und Output, unabhängig vom Anwendungsbereich, im Vordergrund steht [Ja08]. Weitere Definitionen wurden beispielsweise von Davenport [Da93], Champy & Hammer [HC93] und Scheer [Sc02] formuliert. Hierbei werden in der Literatur verschiedene Unterscheidungsmöglichkeiten genannt. Schönig [Sc15] unterscheidet beispielsweise zwischen strikten Prozessen und agilen, personenbezogenen Prozessen. Die Gemeinsamkeit der einzelnen Definitionen liegt insbesondere in der Begründung der Beziehung von Aktivitäten, Input und Output [Ag04]. Zu der allgemeinen Sicht auf Prozesse differenzieren sich Geschäftsprozesse durch das explizite Einbeziehen von Organisationen und betriebswirtschaftlichen Aspekten [Ag04]. Österle [Ös95] nennt Geschäftsprozess eine besondere Form einer Ablauforganisation von Aufgaben, die über mehrere Organisationseinheiten verteilt sein können. Berkau [Be98, S.27] unterteilt Prozesse zum einen in technische Prozesse, zum anderen in Geschäftsprozesse. Geschäftsprozesse beschreiben nach Berkau zum Beispiel das „Einarbeiten eines neuen Mitarbeiters“. Die dazu nötigen Aktivitäten sind schwerer beschreib- und abgrenzbar als Aktivitäten, welche in technischen Prozessen vorkommen (beispielsweise bei der Montage eines Motors).

³⁶ Weite Verbreitung in der Literatur finden die Begriffe des Business Process und des Business Process Management. Geschäftsprozess(-management) bezeichnet die deutsche Übersetzung [BMR14]. Diese Übersetzung wird aus Gründen der besseren Lesbarkeit auch in der vorliegenden Arbeit genutzt. Des Weiteren werden Begriffe wie betrieblicher Ablauf, Workflow, betrieblicher Prozess oder Vorgang als Synonym für den Begriff Geschäftsprozess verwendet.

³⁷ In der vorliegenden Arbeit wird der Begriff des Geschäftsprozesses verwendet. Der Begriff Geschäftsprozess vereint eine Untermenge von Prozessen (z.B. Produktentwicklungsprozesse, Kundenbeziehungsprozesse) aus der betriebswirtschaftlichen Perspektive unter sich.

Geschäftsprozesse wirken demnach informationstransformierend (z.B. Bearbeitung eines Angebots), anstatt materiell (z.B. chemischer Prozess) [Wö10]. Des Weiteren können grundlegende Perspektiven auf Prozesse definiert werden [CKO92]. Die funktionale Perspektive umfasst die funktionalen Bausteine eines Prozesses. Die verhaltensorientierte Perspektive beschreibt das zeitliche Verhalten eines Prozesses, also zum Beispiel wann welche Aktivitäten ausgeführt werden. Diese Perspektive wird in der Regel auch als Kontrollfluss benannt [Sc15]. Eine weitere Perspektive ist die organisatorische Perspektive. Hierbei werden Personen einzelnen Aktivitäten im Prozess zugeordnet. Neben den Perspektiven, welche zur Strukturierung und Einbindung von Personen benötigt werden, dienen die operationale und datenorientierte Perspektive zur Organisation des Prozesses aus systemtechnischer Sicht. Die datenorientierte Perspektive beschreibt, welche Daten und Information innerhalb einzelner Stellen im Prozess notwendig sind bzw. welche Daten und Information wie erstellt oder verändert werden. Die operationale Perspektive definiert welche (Software-)Werkzeuge für die Ausführung eines einzelnen Prozessschritts relevant sind. Abhängig von der Domäne und dem Anwendungsgebiet können weitere unterschiedliche Perspektiven beschrieben werden [Sc15].

Dieser Arbeit liegt die Definition von Mevius [Me06, S.11] zugrunde, welche basierend auf Oberweis [Ob96] einen betrieblichen Ablauf als Geschäftsprozess definiert.

Tabelle 2: Definition Geschäftsprozess nach Mevius

Definition: Geschäftsprozess
<p><i>Ein Geschäftsprozess besteht aus einer Menge von manuellen, teilautomatisierten oder automatisierten Aktivitäten, die in einem Unternehmen ausgeführt werden. Durch die Ausführung dieser Aktivitäten wird ein definiertes Ziel unter Berücksichtigung bestimmter Regeln angestrebt. Die Aktivitäten werden durch Ressourcen ausgeführt. Es wird zwischen personellen und nicht-personellen (maschinellen) Ressourcen differenziert. Eine Aufgabe wird erfüllt bzw. nicht erfüllt, wenn eine oder mehrere Aktivitäten ausgeführt werden. Führen mindestens zwei Ressourcen eine Aktivität aus, so liegt ein kollaborativer Geschäftsprozess vor. Ein verteilter Geschäftsprozess ist dadurch gekennzeichnet, dass er nicht lokal, sondern an zwei geographisch unterschiedlichen Orten ausgeführt wird. Wird ein Geschäftsprozess ausschließlich innerhalb eines einzelnen Unternehmens ausgeführt, so beschreibt dies einen innerbetrieblichen Geschäftsprozess. Sind mindestens zwei Unternehmen an der Ausführung eines Geschäftsprozesses beteiligt, so bezeichnet dies einen überbetrieblichen Geschäftsprozess.</i></p>

Als „Startevent“ eines Geschäftsprozesses gilt der Input, welcher durch die Weiterverarbeitung innerhalb des Geschäftsprozesses – unter Einbezug von bestehenden Informationen - zu einem Output transformiert wird. Die Zuführung des Inputs erfolgt von „außen“. Zum Beispiel kann eine Artikelbestellung der Input eines Geschäftsprozesses sein. Nach Bearbeitung der Artikelbestellung ist beispielsweise eine Versandbestätigung der Output. Gleichzeitig werden neue Informationen über den Geschäftsprozess generiert [Me06]. Eine zentrale Bedeutung für Geschäftsprozesse besitzen Ressourcen. Diese „Aufgabenträger“ können entweder personell oder nicht-personell (maschinell) sein. Beispielsweise werden teilautomatisierte Aktivitäten, sowohl von personellen, als auch von nicht-personellen Ressourcen ausgeführt (z.B. Eingabe und Weiterverarbeitung von Formulardaten, welche in ein Webformular eingetragen wurden). Sind ausschließlich maschinelle Ressourcen an der Ausführung von Aktivitäten in einem Geschäftsprozess beteiligt, ist der Geschäftsprozess automatisiert (vgl. [Ba10b, S.10]).

Neben Aktivitäten und Ressourcen hat auch die „Ver- und Zuteilung“ (kollaborativ, inner- und überbetrieblich) der Aktivitäten Einfluss auf die Geschäftsprozesse. Zudem können Geschäftsprozesse zum Beispiel global und (oder) lokal ausgeführt werden und beeinflussen somit ggfs. auch andere Unternehmensbereiche als die unmittelbar beteiligten Ressourcen [Po00]³⁸.

2.2.2. Geschäftsprozessmanagement

Die Verknüpfung von wirtschaftlichen und technologischen Perspektiven, mit dem Ziel Geschäftsprozesse einer Organisation zu verbessern, ist die Aufgabe des Geschäftsprozessmanagements (vgl. [BMR14][vLS14][SH15]). Immer komplexer werdende Interaktionen von (internen) Beteiligten, Geschäftsprozessen und der Umwelt, fördern einen stetig steigenden Stellenwert von Geschäftsprozessmanagement innerhalb von Organisationen [WW06]. Dabei soll die Effektivität und Effizienz einer Organisation nachhaltig gesteigert werden (z.B. Wettbewerbsfähigkeit) [BMR14]. Trotz der allgemein anerkannten Aufgabenstellung ist eine standardisierte Definition des Begriffs Geschäftsprozessmanagement in der Literatur nicht vorhanden, dennoch dient der Termini „(...) als Sammelbegriff für alle Tätigkeiten im Unternehmen, die eine kontinuierliche Beobachtung und Verbesserung der Prozesse unterstützen“ [Ba11, S.4]. Grund dafür sind unterschiedliche Perspektiven auf das Geschäftsprozessmanagement selbst. Beispielsweise kann der Schwerpunkt auf der Technologie (z.B. Automatisierung von Geschäftsprozessen) oder einer grundsätzlichen Unterstützung bei der Änderung von Geschäftsprozessen liegen [AFG13]. Unabhängig davon „wie“ das Geschäftsprozessmanagement in einer Organisation angewendet wird, muss sichergestellt sein, dass die Anwendung in den Organisationskontext (z.B. Größe eines Unternehmens,...) und zum Organisationsziel (z.B. strategische Mission,...) passt [Vo14]³⁹. Van der Aalst⁴⁰ [ATW03] definiert als Aufgabe des Geschäftsprozessmanagements, die Unterstützung von Geschäftsprozessen mit Hilfe von

³⁸ Die Erfassung von Geschäftsprozessen wird ausführlich in Kapitel 4.2 beschrieben.

³⁹ Vom Brocke et al. definieren insgesamt zehn Prinzipien für ein erfolgreiches Geschäftsprozessmanagement: Context Awareness, Continuity, Enablement, Hoism, Institutionalization, Involment, Joint Understanding, Purpose, Simplicity, Technology Appropriation, vgl. [Vo14].

⁴⁰ Auszug aus [ATW03]: “Supporting business processes using methods, techniques, and software to design, enact, control, and analyze operational processes involving humans, organizations, applications, documents and other sources of information.”

Methoden, Techniken und Software. Diese Geschäftsprozesse sind operational und umfassen zum Beispiel Menschen, Organisationen oder auch Dokumente und andere Informationsquellen. Antunes definiert⁴¹ [AM11] Geschäftsprozessmanagement als eine Sammlung von Technologien, welche vor allem auf Computer-gestützte Aktivitäten oder Computer-gestütztes Controlling von Aufgaben abzielt. Hammer [Ha10b] beschreibt Geschäftsprozessmanagement als ein umfassendes System für die Verwaltung und Umwandlung organisatorischer Abläufe. Des Weiteren beschreibt Hammer, dass das heutige Geschäftsprozessmanagement auf zwei Säulen „steht“: dem Business Process Reengineering (BPR) und dem Deming Zyklus (bzw. kontinuierlichen Verbesserungsprozess) [Ha10b] (vgl. auch [NPW03][Ba11]). Das Business Process Reengineering definiert eine radikale Neugestaltung [Ko09] der Kernprozesse einer Organisation. Geschäftsprozesse werden dabei grundsätzlich überdacht und wenn nötig radikal neu gestaltet⁴². Unnötige Geschäftsprozesse werden im Rahmen des Reengineering komplett „ausrangiert“ [Cr99] und neue Strukturen geschaffen [NPW03]. Der Fokus liegt auf [Tø14] einer möglichen Innovation und der Frage, warum Geschäftsprozesse auf eine bestimmte Art ausgeführt werden. Bei aller Radikalität weist Hammer auf eine Schritt für Schritt Einführung der Neuerung hin, um ein mögliches Risiko des Scheiterns der Neugestaltung zu minimieren [Ha05]. Als zweite Säule des Geschäftsprozessmanagements dient der Deming Zyklus [De53]. Dieser Zyklus beschreibt vier Phasen (Plan, Do, Act, Check)⁴³, welche zu einer stetigen Verbesserung von Geschäftsprozessen genutzt werden kann. Hammer sieht trotz der Einfachheit dieser Schritte einen revolutionären Ansatz für das Management einer Organisation, da beispielsweise anstatt einem „Trial and Error“-Prinzip, also dem bloßen „Ausprobieren“, ein reflektierendes Vorgehen beschrieben wird [Ha10b]. Die Abbildung 8 stellt den Zyklus dar.

⁴¹ Auszug aus [AM11]: “Business Process Management (BPM) integrates a collection of technologies capable to translate business process models into computer supported activities, relinquishing routine management and control tasks from the organizational agents.”

⁴² Vgl. des Weiteren die Definition von Hammer & Champy [HC93, S. 32] zum Begriff Business Process Reengineering: “(...) the fundamental rethinking and radical redesign of the business processes to achieve dramatic improvements in critical, contemporary measures of performance, such as cost, quality, service and speed.”

⁴³Mögliche deutsche Übersetzung: Planen, umsetzen, überprüfen, handeln.

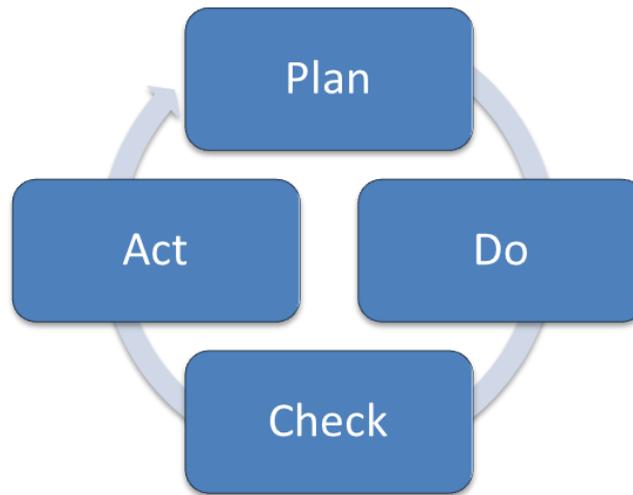


Abbildung 8: Darstellung des Deming-Zyklus (eigene Darstellung)

Die Planungsphase (Plan) dient zur Planung oder Vorbereitung einer Verbesserungs- oder Änderungsaktivität. Die geplanten Aktivitäten werden ausgeführt und getestet, beispielsweise in einem Testbetrieb (Do). Die Erfahrungen aus der „Do“-Phase werden genutzt, um die Implementierung z.B. eines Workflows zu optimieren und die reale Anwendung vorzubereiten (Check). Die Implementierung wird angewendet und weitere Maßnahmen werden bestimmt (Act). Dies führt wiederum zur „Plan“-Phase (vgl. [Mü14]). Durchlaufen Geschäftsprozesse einen derartigen Lebenszyklus (Plan-Do-Check-Act) wird dieser Zyklus in der Literatur auch „Process-Lifecycle-Modell“⁴⁴ genannt [Ba11]. Initiiert wird dieser „operative“ Zyklus von einer überliegenden Ebene: Z.B. dem strategischen Management einer Organisation (vgl. [We12][Du13][MHM14a]).

Obwohl der Begriff des Geschäftsprozessmanagements durch die zwei Säulen (BPR und Deming-Zyklus) „schärfer“ abgrenzbar ist, umfasst der Begriff des Geschäftsprozessmanagement allgemein eine Vielzahl von verwandten, „über- oder untergeordneten“ Ansätzen. Hierbei hängt es von der Perspektive des Betrachters ab, wie die Unterscheidung zwischen den einzelnen Ansätzen erfolgt⁴⁵. Als Beispiele werden unter anderem das Lean Management, Total Quality Management oder Six Sigma in der Literatur genannt [BRU00][SF04][Du13][MH15].

⁴⁴ Im folgenden Abschnitt werden exemplarisch verschiedene Ansätze solcher Modelle beschrieben ([Me06][We12][Li10][Si09][BW04][Br11a][TP13]).

⁴⁵ Eine mögliche Auflistung von Beziehungen unterschiedlicher Ansätze untereinander findet sich in [Sc04a].

Der Begriff Lean Management steht für einen Ansatz eines „permanenten, konsequenten und integrierten Bündels von Prinzipien, Methoden und Maßnahmen“, welches sowohl zur Planung von strategischen Aspekten, als auch zu operativen Aspekten entlang des gesamten Wertschöpfungsprozesses eingesetzt werden kann [WJR91] [PW94]. Ausgewiesenes Ziel hierbei ist es eine Verschwendung aller Art (z.B. Material, Geld, Zeit,...) zu vermeiden. Das Lean Management wurde in der betrieblichen Anwendung von Toyota entwickelt um trotz verschiedener negativer Ereignisse (z.B. Inflation, Krieg,...) wettbewerbsfähig zu bleiben [Li04]. Neben dem Ziel Verschwendung zu vermeiden, wird durch das Lean Management versucht, eine flexible und günstige Produktion zu erreichen [KK14]. Lean Management nutzt hierzu unterschiedliche Managementprinzipien (vgl. [BM95][Li04]), wie beispielsweise dem Management der Prozesse zur Standardisierung und Automatisierung. Der Total Quality Management (TQM) Ansatz spiegelt sich in der Definition nach DIN wider „auf Qualität ausgerichtete Unternehmensphilosophie basierend auf der Beteiligung aller Organisationsmitglieder und mit dem Ziel eines langfristigen Erfolgs durch Kundenzufriedenheit und durch Nutzen für alle Organisationsmitglieder und die Gesellschaft“ [DI09]. Dem obersten Ziel der Kundenzufriedenheit folgend werden auch alle Prozesse und Prozessbeteiligte auf dieses Ziel ausgerichtet [Ro10]. Zur Erreichung des Ziels und zur Überwachung der dazu ausgeführten Aktivitäten nutzt das TQM, sowohl „technische“ Werkzeuge (z.B. statistische Auswertungen, Datenanalyse,...), als auch „nicht-technische“ Aktivitäten (z.B. Einbindung aller Mitarbeiter, Steigerung von Teamwork,...) [AIH11]. In der Literatur werden diese Komponenten auch als „harte“ und „weiche“ Komponenten definiert [PS07][AIH11]. Six Sigma bezeichnet aus Sicht des Managements hauptsächlich eine Steigerung der Unternehmensrentabilität [Ko11b] und damit einhergehend auch eine gezielte Verbesserung der Geschäftsprozesse. Six Sigma umfasst das Design, die Implementierung und die Überwachung von Geschäftsprozessen mit dem Ziel der Kostenreduzierung und Effizienzsteigerung (z.B. Optimierung der Durchlaufzeit) [RBH04]. Die Aktivitäten innerhalb Six Sigmas werden über einen „DMAIC“ bzw. „DMADV“- Zyklus⁴⁶ ausgeführt [Co15b]. Die Verwendung von mathematischen Methoden⁴⁷ auf Basis von gemessenen Daten dienen hierbei als

⁴⁶ Engl. Abkürzung für „define – measure – analyze – improve – control“ bzw. „define – measure – analyze – design – verify“.

⁴⁷ Speziell Methoden aus der Statistik werden hier angewendet, vgl. Kapitel 3.6.3 in [Ko11b]. Six Sigma ist ein Begriff, der sich aus der Statistik ableiten bzw. herleiten lässt. Dabei definiert die Zahl 6 den Zielwert (die „gewünschte“ Genauigkeit oder Toleranz der Standardabweichung σ). [JG08].

Grundlage für Verbesserung und höchste Qualität. Beispielsweise muss eine Organisation mit einer 3-Six Sigma Genauigkeit bei einer Million Bestellungen im Jahr mit 66738 Fehlern rechnen, während eine Organisation mit einer 6-Six Sigma Genauigkeit ausschließlich 3,4 Fehler erwartet [Co15b]. Die Konzepte (Lean Management, TQM, Six Sigma, ...) dienen zur Optimierung (z.B. durch Effizienzsteigerung bei der Ausführung von Aktivitäten). Allerdings werden Fragen bzgl. „(...) einer strategischen Prozessorientierung, der Entwicklung einer Prozesskultur sowie der Ermöglichung neuer Geschäftsmodelle durch Prozessmanagement und innovative Informationssysteme (...)“ nicht beantwortet [SH14, S.55]. Dagegen existiert zudem eine Vielzahl von Ansätzen, welche den Begriff des Geschäftsprozessmanagements „ganzheitlich“ auffassen und erweitern. Beispiele dafür sind das Subjektorientierte-BPM (S-BPM) (vgl. [Fl10][Fl12][FS14]), Social BPM (vgl. [BFV12][RK14][LSW14]) und agiles BPM (vgl. [WW04][Si09][Br11a][TP13][MOW14]). Im Folgenden werden exemplarisch Ansätze zum Geschäftsprozessmanagement nach [Me06][We12][Li10][Si09][WW04][Br11a][TP13] und [ÇD14] näher vorgestellt.

- Traditionelle Ansätze des Geschäftsprozessmanagements

Mevius [Me06] beschreibt einen Ansatz, welcher speziell für kennzahlenbasiertes Geschäftsprozessmanagement⁴⁸ unter variierenden Anforderungen eingesetzt werden kann. Zudem wird ein weiterer Fokus auf die stetige Nutzung und Weiterverwendung von neu aufgebautem Wissen über Geschäftsprozesse und Kennzahlen gelegt. Die folgende Abbildung illustriert die Struktur der Vorgehensweise nach Mevius.

⁴⁸ Vgl. Mevius [Me06, S.30]: „Kennzahlenbasiertes Geschäftsprozessmanagement ist die zielgerichtete Modellierung, Analyse und Kontrolle sämtlicher Geschäftsprozesse, Kennzahlen und Kennzahlensysteme im Unternehmen oder in überbetrieblichen Geschäftsprozessen. Die einzelnen Phasen – Modellierung, Analyse und Kontrolle – des kennzahlenbasierten Geschäftsprozessmanagements stehen in einer logischen und zeitlich interdependenten Beziehung zueinander. (...)“.

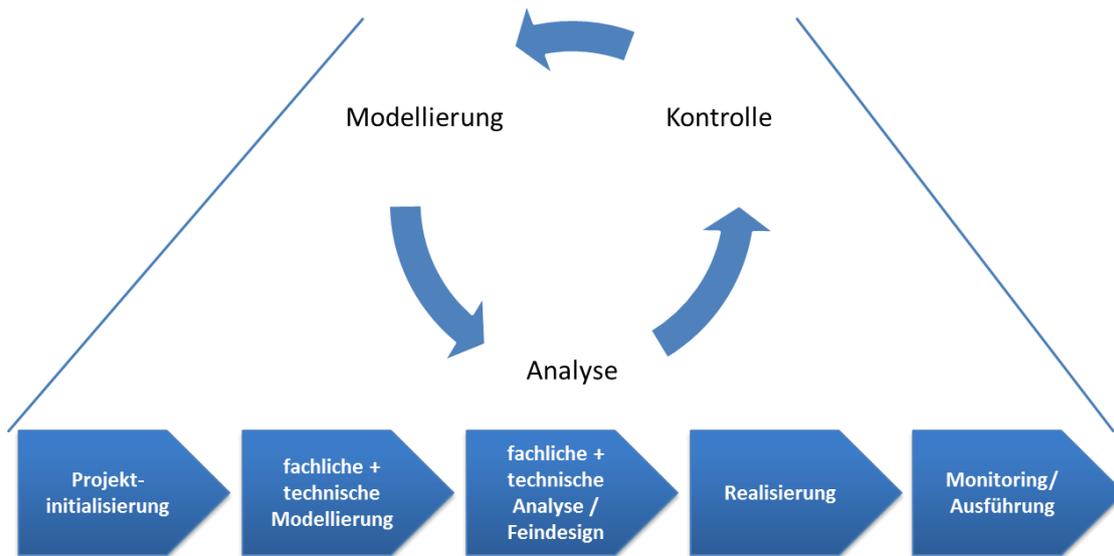


Abbildung 9: Kennzahlenbasiertes Geschäftsprozessmanagement nach [Me06]

Ein Projekt wird mit der Projektinitialisierung gestartet. Die Eingrenzung des Projekts und die Beschreibung der Motivation stehen hierbei im Vordergrund. Als Ergebnis werden Ziele auf Unternehmensebene, Geschäftsprozessebene und Aktivitätsebene definiert und zusammen mit einer Prozesslandkarte⁴⁹ festgehalten. Anschließend an die Projektinitialisierung wird mit der fachlichen und technischen Modellierung begonnen. Die Geschäftsprozesse, Kennzahlen und technischen Anforderungen werden hierdurch im Detail bestimmt. Ein „Schönwetterflug“⁵⁰ soll dazu erfasst werden. Dieser spiegelt die idealisierten Geschäftsprozesse wider. Mevius empfiehlt ferner einen Top-Down Ansatz, um die verschiedenen Geschäftsprozesse den Anforderungen gerecht in verschiedene Abstraktionsebenen zu modellieren. In der Phase des Feindesigns werden weitere fachliche/technische Details geprüft, um festzustellen wo und wie eine IT-System-Unterstützung möglich ist. Im Abschnitt der Realisierung werden die modellierten Geschäftsprozesse und Kennzahlen operativ eingeführt. Der Schwerpunkt liegt dabei auf der Einbindung aller beteiligten Mitarbeiter. Zudem werden innerhalb der Realisierungsphase die technische Anbindung von beispielsweise Datenbanken oder anderen Drittsystem in ein produktives Stadium vorgenommen. Nach Abschluss der Realisierungsphase dient ein kontinuierliches Monitoring zur Überwachung der Geschäftsprozesse. Abweichungen zum Beispiel zu vordefinierten Limits von Kennzahlen werden hierbei erfasst und dienen als zentrale Grundlage einer

⁴⁹ Diese enthält und visualisiert beispielsweise alle Leistungsprozesse, Unterstützungsprozesse und Führungsprozesse auf einer hohen Abstraktionsebene vgl. ISO 9000.

⁵⁰ Bezeichnet den Ablauf im „Idealfall“, also ohne Ausnahmen oder Abbrüche.

kontinuierlichen Verbesserung. Über alle Phasenabschnitte spannt sich ein kontinuierlicher Kreislauf (vgl. Abbildung 9), welcher je nach Projektverlauf mehrmals zur Modellierung, Analyse und Kontrolle durchlaufen werden kann.

Weske [We12] beschreibt den Geschäftsprozessmanagementzyklus in vier Phasen: Evaluierung, Design/Analyse, Konfiguration und Verfügung. Zusätzlich zu den vier Phasen führt Weske zwei überliegende Ebenen ein: „Strategie & Organisation“ und „Design Prozesslandkarte“. Hierdurch kann das Geschäftsprozessmanagement von der Organisationsstrategie bis hin zum implementierten Geschäftsprozess betrieben werden. Die Abbildung 10 skizziert die generelle Geschäftsprozessmanagement-Methode⁵¹ nach Weske.

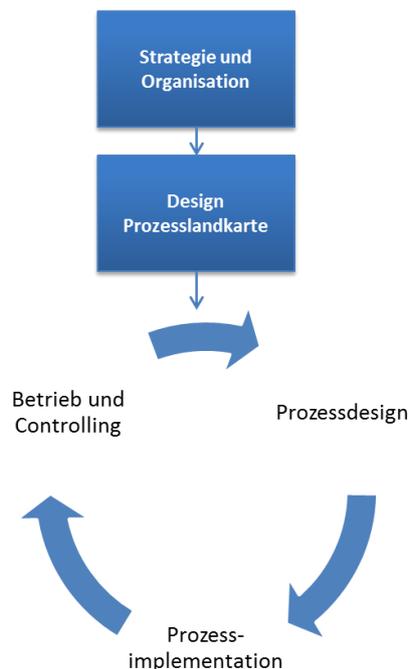


Abbildung 10: „Business Process Methodology“ in Anlehnung an Weske [We12]

Die oberste Ebene, die Strategieebene, definiert die langfristigen Ziele der Unternehmung. Beispielsweise die frühzeitige Erkennung von sich ändernden Rahmenbedingungen, die Einfluss auf bestehende Geschäftsprozesse besitzen können. Somit ist dieser Bereich auch für die Ausarbeitung von langfristigen Wettbewerbsvorteilen verantwortlich. Weske beschreibt zusätzlich die Rolle eines „Chief Process Officer“ (CPO). Der CPO verantwortet alle (GPM relevanten) Aktivitäten und

⁵¹ Engl. „Business Process Methodology“.

bindet das Management einer Organisation ein (z.B. durch Gründung eines Steuerungskomitees). Die zweite Ebene (Design Prozesslandkarte vgl. Abbildung 10) beschreibt das Design einer Prozesslandkarte. Die erstellte Prozesslandkarte, welche eine Übersicht über die für die Organisation relevanten Geschäftsprozesse darstellt (vgl. [MM13][MHM14b]) dient zur Strukturierung, wobei insbesondere die Zusammenhänge der Geschäftsprozesse identifiziert werden müssen (vgl. Input und Output von Geschäftsprozessen⁵²).

Anschließend an die zwei Ebenen (Strategie & Organisation und Design) wird der „operative“ GPM-Zyklus durchlaufen. In der Design & Modellierungsphase (Prozessdesign vgl. Abbildung 10) werden zuerst die Zusammenhänge von operativen Geschäftsprozessen und deren übergeordneten organisatorischen Elementen erfasst. Die Modellierung der Geschäftsprozesse kann beispielsweise graphisch mit der Modellierungssprache BPMN durchgeführt werden. Ergebnisse der Modellierung werden zur Dokumentation und Abstimmung aller Beteiligten genutzt. Ein weiterer Bestandteil dieser Phase ist die Verbesserung und Anreicherung von Geschäftsprozessen.

Insbesondere die Definition von Kennzahlen wird in dieser Phase durchgeführt. Nach Weske umfasst eine Kennzahl 1) das Ziel („warum wird diese Kennzahl benötigt?“), 2) Name und Datentyp der Kennzahl, 3) Algorithmus zur Berechnung der Kennzahl, 4) Zielwert der Kennzahl und 5) Ober- und Untergrenze der Kennzahl (z.B. sodass bei Überschreiten der Grenzen eine Eskalation getriggert wird). Die nächste Phase der Prozessimplementierung (Prozessimplementierung vgl. Abbildung 10) dient primär der Entscheidung, wie die zuvor erstellten Geschäftsprozesse, sowohl organisatorisch, als auch technisch realisiert werden sollen. Sollen Geschäftsprozesse automatisiert werden, muss ein entsprechendes System⁵³ zur Automatisierung zur Verfügung stehen. Falls noch kein System in Betrieb ist, ist es nötig, dass die Organisation basierend auf definierten Auswahlkriterien (z.B. Leistungsfähigkeit, Kosten,...) ein System anschafft. Zudem wird in dieser Phase ein mögliches Kontrollsystem aufgebaut.

Definierte Kennzahlen können somit mit weiteren Daten verbunden oder bei Bedarf einzeln an definierten Messpunkten erfasst werden.

⁵²Abschnitt 2.2.1.

⁵³ Vgl. Abschnitt 2.2.3 - Geschäftsprozessmanagementsysteme

Tests z.B. durch Simulation der Geschäftsprozesse validieren die Implementierung, bevor diese dann produktiv in Betrieb genommen wird. Abschließend dient die Phase des Betriebs & Controllings zur Steuerung der „täglichen Arbeit“⁵⁴. Insbesondere wird die Einhaltung der Geschäftsprozesse kontrolliert. Dadurch kann beispielsweise festgestellt werden, ob Anwender nicht-freigegebene Aktivitäten durchlaufen⁵⁵. Zusätzlich wird auf Basis der zuvor definierten Kennzahlen ein kontinuierliches Controlling durchgeführt. Kann ein Problem identifiziert werden, startet die Designphase und der Zyklus wird erneut durchlaufen.

Link [Li10] beschreibt einen Geschäftsprozessmanagementzyklus aus fünf Phasen. Der Zyklus teilt sich dabei in zwei Teile auf: der Entwicklungszeit und der Laufzeit. Die Abbildung 11 stellt den erweiterten Zyklus dar.

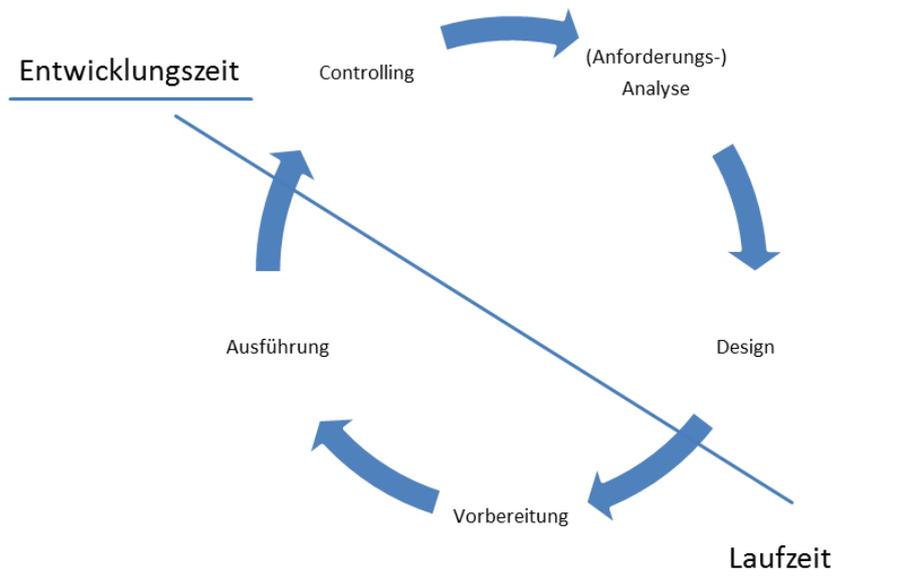


Abbildung 11: Erweiterter Geschäftsprozessmanagementzyklus nach Link [Li10]

⁵⁴ Übersetzt aus dem Englischen „daily business of the organization“.

⁵⁵ Als Beispiel kann hier das Phänomen der Schatten-IT genannt werden, bei der (zusätzliche) Aktivitäten in einem Geschäftsprozess ohne Freigabe ausgeführt werden (vgl. [Br11b][RZ12][ZRF14]). Nach Rentrop & Zimmermann [RZ12] beschreibt Schatten-IT „geschäftsprozessunterstützende IT-Systeme, IT-Serviceprozesse und IT-Mitarbeiter, die von Fachabteilungen und IT-Anwendern eigenständig eingesetzt werden. Die Schatten-IT-Instanzen sind dabei weder technisch noch strategisch in das IT-Servicemanagement der Organisation eingebunden und somit weder im Asset und Configuration Management noch im Serviceportfolio berücksichtigt.“

Nach Link werden in der Anforderungsanalyse, unter Berücksichtigung von Einflussfaktoren (z.B. strategische Ziele, Normen, ...), Vorbereitungen für die Modellierung der Geschäftsprozesse getroffen. Beispielsweise werden Begriffsdefekte⁵⁶ (z.B. Missverständnisse, Synonyme,...) identifiziert und geklärt. Während der nachfolgenden Designphase wird zwischen dispositiven und operativen Aspekten unterschieden. Die operativen Aspekte (z.B. Funktionsaspekt, Steuerungsaspekt, Informationsaspekt, Organisationsaspekt, Arbeitsmittelaspekt,...) können durch eine Modellierungssprache (z.B. BPMN) abgebildet werden. Zum Beispiel beschreiben die Aktivitäten des Funktionsaspekts „was“ während eines Geschäftsprozesses auszuführen ist. Die dispositiven Aspekte (z.B. Normenaspekt, Qualitätsaspekt, Kollaborationsaspekt, Fehlerbehandlungsaspekt,...) sind den operativen Aspekten übergeordnet und können diese beeinflussen bzw. beschränken. Beispielsweise beschreibt der Normenaspekt Regelungen (z.B. Gesetze), welche für die modellierten Geschäftsprozesse gültig sind. Durch die Trennung von operativer und dispositiver Ebene, können Modelle zu jedem Zeitpunkt gegeneinander validiert werden. Die nachfolgende Vorbereitungsphase ist sowohl in der Entwicklungszeit, als auch in der Laufzeit wiederzufinden. Die Zuteilung der Vorbereitungsphase ergibt sich aus der Notwendigkeit, dass beispielsweise Arbeitsmittel bereits während der Entwicklungszeit beschafft werden müssen. In der Laufzeit dient die Vorbereitungsphase dann zur konkreten Zuteilung der Arbeitsmittel (z.B. wird die Rolle „Sachbearbeiter“ konkret einem Mitarbeiter „Herr Müller“ zugeordnet und das benötigte Arbeitsmittel an diesen Mitarbeiter ausgeteilt). Aufgrund der Möglichkeit, dass Instanzen mit einem sehr langem Ausführungszeitraum auftreten können, endet die Vorbereitungsphase nicht vor, sondern gemeinsam mit der nächsten Phase. In der Ausführungsphase werden die Geschäftsprozesse gemäß der zuvor erfassten Modelle (operative Ebene) durchlaufen und ausgeführt. Die Phase des Controllings schließt den Kreislauf. Verwertbare Informationen, welche über die Ausführungsphase gespeichert werden konnten, können analysiert und zur Optimierung der Geschäftsprozesse genutzt werden.

Die Ansätze von Mevius [Me06], Weske [We12] und Link [Li10] sind Beispiele für ein traditionelles Geschäftsprozessmanagement. Traditionelle Ansätze zeichnen sich insbesondere durch das sequentielle Durchlaufen der einzelnen Phasen aus.

⁵⁶ Vgl. Abschnitt 2.1.3

Muss beispielweise ein Geschäftsprozess automatisiert werden, wird im Allgemeinen mit einer detaillierten Analyse (z.B. zur Festlegung der Ziele, Qualitätskriterien,...) begonnen. Danach wird in einer zweiten Phase der Geschäftsprozess modelliert. Anschließend wird die Implementierung durchgeführt (z.B. Anbindung von Drittsystemen, Erstellung von Benutzeroberflächen,...). Abschließend kann dann die Geschäftsprozessapplikation veröffentlicht werden (d.h. für die Anwender zur Anwendung zur Verfügung gestellt). Darauffolgend wird während der Anwendung der Geschäftsprozessapplikation die Ausführung überwacht. Müssen Anpassungen an der Geschäftsprozessapplikation getätigt werden (z.B. aufgrund Fehler in der Implementierung), beginnt erneut der sequentielle Durchlauf bei der Analysephase. Die Sequenzierung führt zu einem hohen zeitlichen Abstand zwischen der Veröffentlichung unterschiedlicher Versionen. Insbesondere in agilen Umgebungen führt diese Verzögerung zum Verlust der Aktualität. Geschäftsprozessmodelle und -applikationen entsprechen hierdurch nicht mehr den realen Anforderungen (vgl. Abschnitt 1.1). Des Weiteren werden Anwender in traditionellen Ansätzen nicht ausreichend in die Phasen des Geschäftsprozessmanagement eingebunden. Unzureichende Schulung der Anwender während der Modellierung (z.B. zum Erlernen der verwendeten Modellierungssprache) oder Fehlimplementierungen (z.B. innerhalb einer Geschäftsprozessapplikation) führen zu signifikanten Akzeptanzproblemen (vgl. [Ba07][Ko10a][Be12][Pw14][Br14]). Mögliche positive Auswirkungen eines sequentiell durchgeführten GPM-Projekts (z.B. langfristige Kostenreduzierung, Qualitätssteigerung, ...) werden hierbei oftmals zu spät innerhalb der Organisation bzw. von den Projektbeteiligten wahrgenommen. Die im Folgenden beschriebenen agilen Geschäftsprozessmanagementansätze adaptieren und erweitern die traditionellen Vorgehensweisen. Den dargestellten Problemen (z.B. unzureichende Aktualität, geringe Anwenderakzeptanz,...) wird hierbei durch die Flexibilisierung des traditionellen Geschäftsprozessmanagementzyklus und intensivere Integration aller Beteiligten entgegenwirkt.

- Ansätze des agilen Geschäftsprozessmanagements

Silva et al. [Si09] beschreiben einen Ansatz zum agilen Geschäftsprozessmanagement, welcher durch Wiki⁵⁷-basierte Technologien unterstützt wird.

Hierbei wird die Trennung zwischen Modellierung und Ausführung durch die Nutzung einer gemeinsamen Softwareumgebung aufgehoben. Nach Silva et al. sollen hierdurch insbesondere die Unterschiede zwischen einem vordefinierten Geschäftsprozessmodell und den real ausgeführten Instanzen reduziert werden. Um dies zu erreichen, können beispielsweise ausführende Anwender unmittelbar generische Aktivitäten, welche für deren Instanz benötigt werden, zu einem Geschäftsprozessmodell hinzufügen. Die Abbildung 12 stellt das Rollenkonzept nach Silva et al. dar.

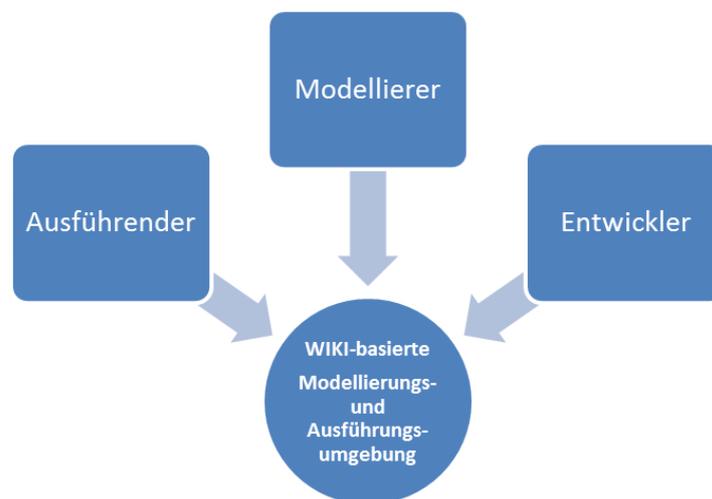


Abbildung 12: Agiles Geschäftsprozessmanagement nach Silva et al. [Si09]

Wie in Abbildung 12 dargestellt, existieren nach Silva et al. drei Rollen: Ausführender, Modellierer und Entwickler (eine Person kann hierbei auch unterschiedliche Rollen wahrnehmen). Über die gemeinsame Softwareumgebung werden Geschäftsprozesse modelliert, automatisiert und ausgeführt. Sobald innerhalb einer Instanz eine Ausnahme auftritt (z.B. ein zusätzlicher Genehmigungsschritt durch das Management) kann der Ausführende eine generische Aktivität zur Spezifikation der Ausnahme zu dem vordefinierten Geschäftsprozessmodell hinzufügen. Die Modifikation des Geschäftsprozessmodells gilt anfänglich nur für die ausgeführte Instanz.

⁵⁷ Der Begriff Wiki beschreibt ein System zum kollaborativen Nutzen und Pflegen von Inhalten. Die Interaktion (z.B. Ändern von Einträgen) mit dem System ist hierbei durch die einfache Bedienbarkeit gewährleistet (vgl. <http://de.wikipedia.org/wiki/Wiki> und [LC01][Ma13]).

Nach einer Analyse der Modifikationen passen Modellierer das eigentliche Geschäftsprozessmodell an.

Entwickler unterstützen dabei die Modellierung, indem ggfs. eine weitere Anreicherung von Aktivitäten hinsichtlich der Automatisierung unternommen wird. Als zentraler Bestandteil der Vorgehensweise von Silva et al. gilt der Folksonomy⁵⁸-Effekt. Die Beteiligten verknüpfen Schlagwörter mit den auf der GPM-Systemumgebung gespeicherten Geschäftsprozessmodellen. Auf dieser Grundlage können beispielsweise wiederum Dokumentationen zu Geschäftsprozessen von Anwendern mitgepflegt werden. Somit ist die Zusammenarbeit aller Beteiligten über den gesamten GPM-Zyklus durch den Einsatz der Wiki-basierten Technologie gewährleistet. Die für ein agiles Vorgehen typischen Eigenschaften (vgl. Abschnitt 2.3.1) werden durch den Ansatz von Silva et al. forciert. Beispielsweise wird anstatt der Erstellung von ausführlichen Geschäftsprozessmodell dokumentationen, insbesondere auf die Modellierung und Automatisierung von Aktivitäten fokussiert, welche zwingend für die reale Ausführung notwendig sind.

Weber & Wild [WW04] stellen einen agilen Ansatz mit Fokus auf die Geschäftsprozessautomatisierung vor. Ziel ist es, agil auf neue Anforderungen an Workflows reagieren zu können und „just-in-time“ Änderungen an vordefinierten Workflows zu ermöglichen. Der Ansatz basiert hierbei auf der Kombination von „traditionellen“ Regeln und dem Case-Based-Reasoning⁵⁹ (CBR). Die Abbildung 13 visualisiert die einzelnen Schritte nach Weber & Wild.

⁵⁸ Der Begriff ist auf Vander Wal (vgl. <http://vanderwal.net/folksonomy.html>) zurückzuführen und beschreibt das Erzielen von besseren (Such-)ergebnissen durch die Zusammenarbeit von Anwendern bei der Erstellung von Indexen (Sammlung von Schlagwörtern). Des Weiteren sei auf [Yi08, S.322] verwiesen.

⁵⁹ Als Case-Based Reasoning wird ein Vorgehen (bzw. Systeme) beschrieben, durch welche a priori erfasste Erfahrungen adaptiert und zur Lösung von neuen Fällen (Cases) genutzt werden können [Le94].

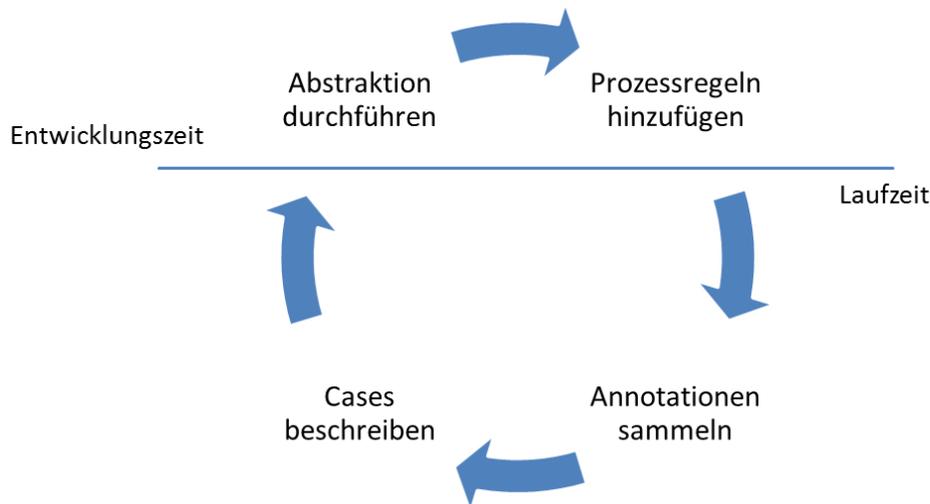


Abbildung 13: Agiles Workflow-Management nach Weber & Wild [WW04]

Der Zyklus wird in zwei Teile aufgeteilt: Entwicklungszeit und Laufzeit. Zur Entwicklungszeit werden die Aktivitäten implementiert, welche bereits ausreichend (aufgrund des existierenden Wissens während der Entwicklungszeit) innerhalb eines Geschäftsprozesses definiert werden können. Hingegen werden Aktivitäten, welche zu zeitintensiv zu analysieren sind (um eine ausreichende Basis für die Implementierung zu schaffen), nicht erfasst. Sollen zum Beispiel neue Geschäftsprozessaktivitäten vorab simuliert werden, ist für die Festlegung von realistischen Simulationsparametern (Durchlaufzeit, Kosten,...) eine (zeit-)intensive Analyse notwendig. Sobald der zu implementierende Geschäftsprozess über ausreichend Regeln verfügt, kann dieser automatisiert ausgeführt werden. Werden während der Ausführung Mängel (z.B. aufgrund einer veralteten Implementierung) durch die Anwender erkannt, können Annotationen zum Workflow erfasst werden. Jede Annotation stellt hierbei ein Case dar. Diese Annotationen (Prozessinformationen) können von weiteren Anwendern direkt genutzt werden, ohne dass der implementierte Geschäftsprozess geändert werden muss. Sobald die in einem Case gesammelten Informationen eine definierte Reife erreicht haben (vgl. Abbildung 13 – „Case beschreiben“), werden die Informationen zum Update des bestehenden Geschäftsprozessmodells genutzt. Nach der Weiterentwicklung und Abstraktion der gesammelten Informationen während der Entwicklungszeit kann der „neue“ Workflow ausgeführt werden. Der Ansatz von Weber & Wild folgt dem Agilen Manifest⁶⁰.

⁶⁰ Das Agile Manifest beschreibt Werte zur agilen Softwareentwicklung. Eine detaillierte Erläuterung findet sich in Abschnitt 2.3.1.

Das direkte Annotieren ermöglicht ein schnelles Feedback von Anwendern ohne zeitintensive und komplizierte Wege. Die Agilität des Ansatzes spiegelt sich somit durch die Fähigkeit wider, auf nicht-vorhersagbare und dynamische Änderungen unmittelbar reagieren zu können.

Bruno et al. nennen Social Software⁶¹ als eine Möglichkeit das traditionelle Geschäftsprozessmanagement zu unterstützen. Die Autoren beschreiben die Problematik von festen Strukturen innerhalb des traditionellen Geschäftsprozessmanagementzyklus. Beispielsweise kann es zu Verzögerungen bei der Anpassung von Geschäftsprozessmodellen kommen. Grund dafür können lange Abstimmungszyklen sein. Durch die Nutzung einer Echtzeit-Kollaborationsplattform wird insbesondere die flexible Interaktion der GPM-Projektbeteiligten gefördert. Die Social Software wird hierbei zur Verbesserung der Anwenderintegration und Reaktionsfreudigkeit⁶² eingesetzt. Die Abbildung 14 skizziert den Ansatz.

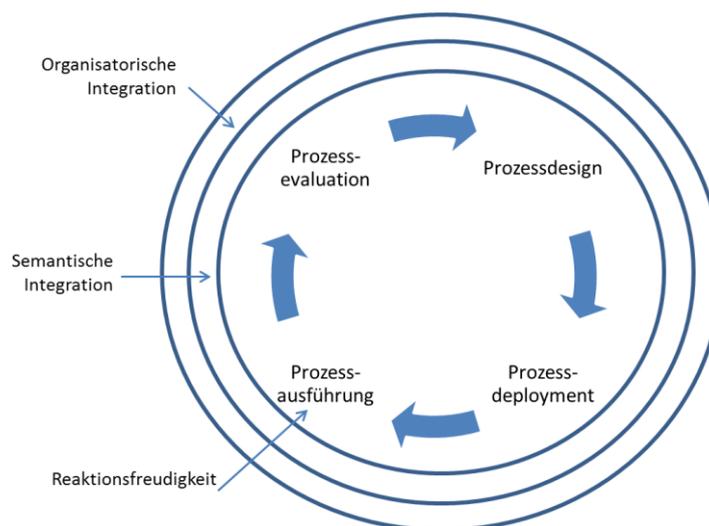


Abbildung 14: Anforderungen an agiles Geschäftsprozessmanagement nach [Br11a]

Wie in Abbildung 14 dargestellt, wirken nach Bruno et al. drei Barrieren auf den Geschäftsprozessmanagementzyklus: (1) Organisatorische Barriere (2) semantische Barriere und (3) die Barriere der „Reaktionsfreudigkeit“. Durch die organisatorische Integration werden alle Beteiligten (Stakeholder) integriert.

⁶¹ Als Social Software wird Software bezeichnet, bei der „Beziehungen zwischen Individuen bzw. Personengruppen“ im Vordergrund stehen [HW05]. Beispiele hierfür sind soziale Netze wie facebook.de oder xing.de. Vgl. des Weiteren [Ma10, S.7].

⁶² Das englische Wort responsiveness wurde mit dem deutschen Wort Reaktionsfreudigkeit übersetzt.

Dies wird durch die Möglichkeit realisiert, dass zum Beispiel auch externe Beteiligte (z.B. Kunden, externe Berater,...) Anforderungen an Geschäftsprozesse (spontan und kontextabhängig) mitteilen können. Des Weiteren ermöglicht eine semantische Integration die Verbesserung des allgemeinen Verständnisses der Geschäftsprozesse. Beispielsweise können Homonyme oder Synonyme zu Missverständnissen bei der Ausführung von Geschäftsprozessen führen. Die durch Social Software bereitgestellten Funktionen, wie z.B. die Möglichkeit der ortsunabhängigen Kollaboration von Wissen wirken dieser Barriere entgegen. Die dritte Barriere entsteht durch gehemmte Reaktionsfreudigkeit aufgrund von festen Strukturen in traditionellen Ansätzen. Hierdurch kann auf neue Anforderungen nur schwer in Echtzeit reagiert werden. Nach Bruno et al. können beispielsweise „kooperative Aktivitäten“ zur höheren Flexibilität beitragen. Hierbei werden Geschäftsprozessmodelle mit Hilfe, den von der Community, bereitgestellten Aktivitäten angereichert. Durch den kontinuierlichen Austausch von Informationen und Nutzung der gesamten Community als Wissensbasis kann ein agiles Geschäftsprozessmanagement betrieben werden.

Durch die Anwendung der Vorgehensweise nach Thiemich & Puhmann [TP13] wird ein GPM-Projekt in verschiedene Phasen und in sogenannte Sprints unterteilt. Einige der von Thiemich & Puhmann verwendeten Begriffsbezeichnungen und Aspekte orientieren sich hierbei insbesondere an der agilen Vorgehensweise Scrum (vgl. Abschnitt 2.3.2). Zum Beispiel beschreibt ein Sprint einen vordefinierten Zeitabschnitt, in welchem zuvor (durch z.B. die beteiligten Entwickler) ausgewählte Anforderungen implementiert werden. Für jede Phase und jeden Sprint beschreiben Thiemich & Puhmann Aktivitäten und Methoden, durch welche die Erreichung von vordefinierten Zielen unterstützt wird. Beispielsweise muss zu Beginn eines GPM-Projekts die Projektidee diskutiert und dokumentiert werden. Die Abbildung 15 skizziert den Ansatz nach Thiemich & Puhmann.

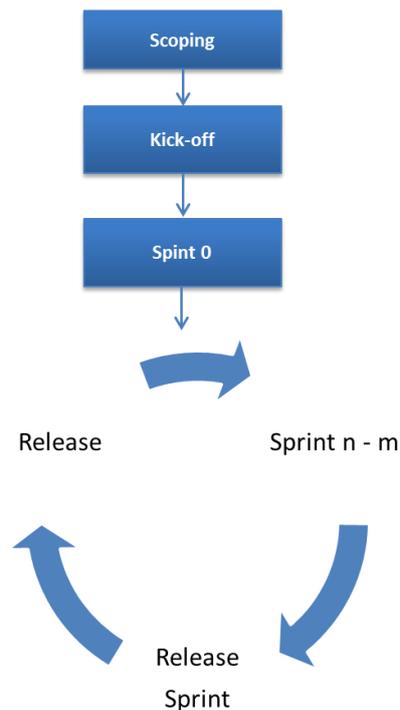


Abbildung 15: Geschäftsprozessmanagement nach Thiemich & Puhlmann

In der „Scoping“-Phase werden die Rahmenbedingungen des GPM-Projekts ermittelt. An dieser Stelle werden unter anderem das Budget und die Projektidee (z.B. Verbesserungsvorschlag bzgl. eines Geschäftsprozesses) festgelegt bzw. beschrieben. Im „Kick-off“ werden ein erster Architekturforsch sowie ein grober Releaseplan erstellt. Zusätzlich wird innerhalb des „Kick-off“ das Projektteam definiert. Der „Sprint 0“ dient zur initialen Detailplanung der zuerst zu implementierenden Anforderungen. Des Weiteren wird die sogenannte „Definition of Ready“ und „Definition of Done“ festgelegt. Die Definitionen dienen zur Beschreibung der gegenseitigen Erwartungen. Die „Definition of Ready“ enthält hierbei Regeln, wie die Anforderungen beschrieben werden müssen (z.B. für jede Anforderungen existiert eine Aufwandsschätzung). Die „Definition of Done“ erläutert die Kriterien, welche beschreiben, „wann“ eine Anforderung als vollständig implementiert gilt. Nach dem „Sprint 0“ wird der eigentliche Zyklus gestartet. Je nach Größe des Projekts wird der Zyklus mehrmals durchlaufen. Innerhalb des Zyklus wird unter Sprint, Release Sprint und Release unterschieden. Ein Sprint dient zur Implementierung von Anforderungen, wobei das Ergebnis ein anwendbares (z.B. durch Testanwender) Prozessinkrement ist. Der Release Sprint enthält zusätzliche, für Anwender aufbereitete, (Schulungs-)Dokumentationen. Ein Release kann somit im realen Betrieb verwendet werden.

Durch die iterative und inkrementelle Abarbeitung der im „Prozess-Backlog“ gespeicherten Anforderungen wird eine kontinuierliche Integration aller Beteiligten sichergestellt.

Die Vorgehensweise nach Çulha & Dođru [ÇD14] gliedert sich in sechs Phasen. Als weiterer zentraler Bestandteil wird die sogenannte „Iteration Base“ von Çulha & Dođru erläutert. Eine „Iteration Base“ beschreibt einen Zeitraum (nach [ÇD14, S.137] mit der Dauer von fünf Wochen), in welchem die Phasen-spezifischen Aufgaben bearbeitet werden. Eine „Iteration Base“ wird in drei Teile untergliedert: Verhandlung (Dauer eine Woche), Entwicklung (Dauer drei Wochen), Konsolidierung (Dauer eine Woche)⁶³. Innerhalb der Verhandlung werden die Anforderungen an das zu implementierende Produkt zwischen allen Beteiligten (z.B. Kunde und Entwickler) abgestimmt. Die Entwicklung wird durch ein kleines⁶⁴ Team durchgeführt, wobei während der Entwicklung regelmäßige Treffen zur kontinuierlichen Schulung und Diskussion zwischen allen Beteiligten dienen. Innerhalb der Konsolidierungszeit werden die entwickelten Funktionen abschließend überprüft. Die Abbildung 16 skizziert den Durchlauf eines GPM-Projekts bis hin zum implementierten Produkt.

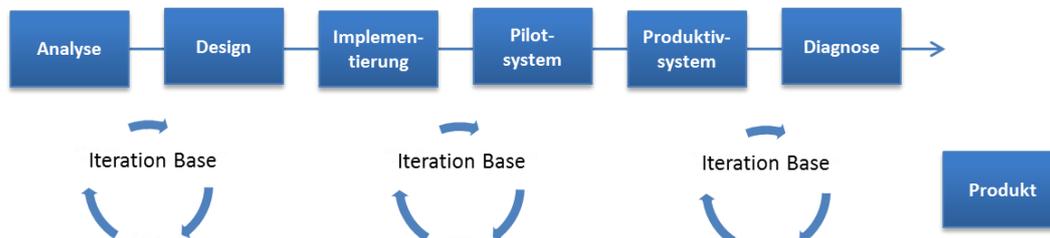


Abbildung 16: Agiles Geschäftsprozessmanagement nach Çulha & Dođru

In der Analyse- und Designphase werden die Anforderungen analysiert und die Geschäftsprozessmodelle modelliert. Çulha & Dođru empfehlen hierbei die Nutzung der BPMN Modellierungssprache und den Einsatz von Templates⁶⁵. Die weitere Implementierung erfolgt in zwei Stufen: dem Pilot- und Produktivsystem. Das Pilotsystem wird für frühzeitige Tests genutzt (z.B. Anwender testen vereinzelt Funktionen einer Geschäftsprozessapplikation). Nach der Veröffentlichung auf dem

⁶³ Vgl. engl. Übersetzung: negotiation (Verhandlung), development (Entwicklung), consolidation (Konsolidierung).

⁶⁴ Çulha & Dođru definieren keine exakte Teamgröße in [ÇD14].

⁶⁵ Vgl. [ÇD14, S.136]: „(...) any kind of template usage is encouraged because templates can accelerate the development process.“

Produktivsystem dient die Überwachung der ausgeführten Geschäftsprozesse als Diagnosephase (vgl. Abbildung 16). Durch die Diagnose identifizierte Optimierungspotentiale werden wiederum analysiert, sodass der Zyklus erneut startet. Zusätzlich zu den Phasen sollen nach Çulha & Dođru über das gesamte GPM-Projekt die Themen 1) Training 2) Dokumentation 3) Projektmanagement 4) Sponsorship berücksichtigt werden. Somit soll sowohl die intensive und nachhaltige Integration und das Verständnis⁶⁶ aller Beteiligten gewährleistet werden, als auch die Einhaltung der Projektrahmenbedingungen (z.B. Budget, Zeit,...).

Zusammenfassend kann festgestellt werden, dass sowohl bei den traditionellen als auch agilen Ansätzen die Unterstützung von Prozessbeteiligten hinsichtlich Vorgehensweise, Sprache und Werkzeug häufig nicht ganzheitlich ist. Ganzheitlich bedeutet, dass die Methode über alle Schritte des Geschäftsprozessmanagements erfolgreich angewendet werden kann. „Methodenlücken“ können unzureichende Aktualität von Geschäftsprozessmodellen, geringe Anwenderakzeptanz und fehlende Einbindung von allen Prozessbeteiligten zur Folge haben (vgl. Abschnitt 1.1). Insbesondere die Sequenzierung der einzelnen Schritte (Modellierung, Anreicherung, ect.) ohne regelmäßige Abstimmungen führt zu einem hohen zeitlichen Abstand zwischen der Veröffentlichung unterschiedlicher Geschäftsprozessmodellversionen (vgl. [MOW14]). Dokumentationen und automatisierte Prozessapplikationen entsprechen hierdurch nicht mehr den realen Anforderungen (vgl. Abschnitt 1.1). Des Weiteren ergänzen die existierenden agilen die traditionellen Geschäftsprozessmanagementansätze nur unzureichend. Beispielsweise wird hierbei häufig ebenso keine Unterscheidung zwischen der Modellierung und Automatisierung von Geschäftsprozessen vollzogen, obwohl bei der Modellierung und Automatisierung unter anderem signifikant verschiedene Anforderungen bestehen (vgl. Abschnitt 4.2.5) Des Weiteren fehlt für die Unterstützung von agilen Vorgehensweisen häufig das „passende“ Softwarewerkzeug, welches in einer flexiblen Umgebung erfolgreich zur Unterstützung eingesetzt werden kann (vgl. Abschnitt 2.2.3).

⁶⁶ Vgl. [ÇD14, S.137]: „The more the customer is trained the better the business process is understood. Training is emphasized in this methodology because of the following important reasons: * Training improves the gathering of quality requirements. * Training informs people about the process and decrease the development time. * Training reminds of the decisions take before, and people do not need to re-solve problems.

Die in dieser Arbeit erarbeitete Methode adaptiert und erweitert die existierenden Ansätze. Die Methode wird in Abschnitt 4 detailliert beschrieben.

2.2.3. Geschäftsprozessmanagementsysteme

Geschäftsprozessmanagementsysteme (GPMS)⁶⁷ unterstützen bei der Definition, Verwaltung, Anpassung, Ausführung und Auswertung von Geschäftsprozessen (und auch Organisationsstrukturen) (vgl. [Ka95][DKK14]). Je nach Autor und Funktionsumfang existieren unterschiedliche Bezeichnungen für derartige Systeme. Beispielsweise unterscheidet Draheim [Dr10] zwischen Business Modeling Tools, Business Process Management Suites und Workflow-Management-Systemen⁶⁸. Business Modeling Tools unterstützen den Anwender bei der Modellierung von Geschäftsprozessen. Workflow-Management-Systeme ermöglichen die Automatisierung von Geschäftsprozessen. Im Vergleich zu Workflow-Management-Systemen⁶⁹ (WfMS), welche speziell die Bearbeitung von Aufgaben unterstützen [Le99], erweitert der Begriff der Geschäftsprozessmanagementsysteme die Unterstützung von allen Phasen des Geschäftsprozessmanagements [ATW03]. Nach Becker et al. [Be09] muss ein Geschäftsprozessmanagementsystem, Leistungsmerkmale in 1) Modellierung und Simulation, 2) Monitoring, 3) Ausführung (d.h. Automatisierung von Geschäftsprozessen), 4) Kollaboration (d.h. Unterstützung der Zusammenarbeit von Beteiligten eines Geschäftsprozesses) und 5) Produktivitätssteigerung (d.h. zum Beispiel die zentrale Rechteverwaltung zur Steuerung von Zugriffen auf Ressourcen bzw. Systeme zur Implementierung eines durchgängigen Sicherheitskonzepts) aufweisen. Beispielsweise wird die (menschliche) Analyse und Erfassung von Geschäftsprozessen softwaretechnisch unterstützt. Chang [Ch06, S.50] beschreibt Geschäftsprozessmanagementsysteme als neue "Klasse" von Software, welche es Organisationen ermöglicht Prozess-zentrierte Applikationen zu entwickeln⁷⁰. Unter

⁶⁷ In der Literatur wird häufig die englische Abkürzung BPMS (Business Process Management System) genutzt. Zusätzlich wird der Begriff Business Process Management Suite synonym verwendet.

⁶⁸ Engl. „(...) business process modeling tools, business process management suites and workflow management systems (...)“ [Dr10, S. 246].

⁶⁹ Durch die Workflow Management Coalition, ein weltweiter Verbund zur Standardisierung im Bereich des Workflow- und Geschäftsprozessmanagements, wurde 1995 eine Referenzarchitektur für Workflowmanagement- Systeme veröffentlicht. Zur weiteren Ausführung sei auf [Ho95][LKP15] verwiesen.

⁷⁰ Engl. „BPMS is a new class of software that allows organizations to devise process-centric information technology solutions.“ [Ch06, S.50]

anderem kann dadurch ein Geschäftsprozess zur Erstellung eines Kundenangebots mit Hilfe eines GPMS ganzheitlich unterstützt werden. Die Abbildung 17 zeigt exemplarisch die Nutzung eines GPM-Systems im Rahmen der Kundenangebotserstellung.

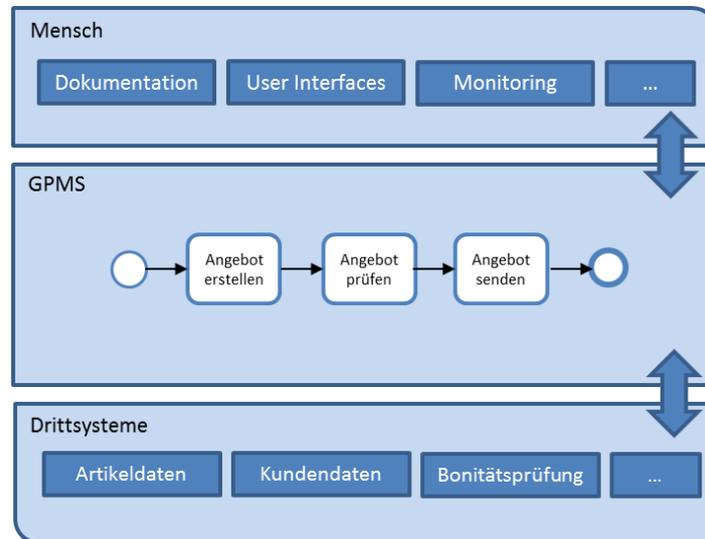


Abbildung 17: Nutzung eines GPM-Systems bei der Kundenangebotserstellung

Das System dient als zentrale „Integrationsschicht“ zwischen Menschen und bestehenden Drittsystemen (z.B. einem ERP-System in welchem Artikeldaten gespeichert sind). Zum einen kann der Geschäftsprozess aus Zwecken der Dokumentation mit Hilfe des GPMS modelliert werden. Der modellierte Geschäftsprozess in Abbildung 17 umfasst drei Aktivitäten, welche je nach GPMS mit einer beliebigen Modellierungssprache (z.B. BPMN) erfasst werden können. An dem Geschäftsprozess beteiligte Menschen nutzen die Dokumentation nachfolgend beispielsweise zur Einarbeitung in die ihnen zugeteilten Aktivitäten. Zum anderen werden User Interfaces (z.B. Eingabemasken zur Sammlung der Bestelldaten) implementiert, welche dem Anwender während der Ausführung aktivitätsabhängig angezeigt werden. Zusätzlich haben Anwender Zugriff auf ein GPMS-gestütztes Monitoring der Aktivitäten (z.B. Dauer der Angebotserstellung), wodurch eine Analyse und Optimierung des Geschäftsprozess unterstützt wird. Zudem können Aktivitäten (z.B. die Bonitätsprüfung eines Kunden) (teil-)automatisiert werden. Geschäftsprozessmanagementsysteme unterstützen hierzu zum Beispiel die Anbindung von Drittsystemen oder –Services. Wie in Abbildung 17 dargestellt, können für den Geschäftsprozess relevante Daten (z.B. Artikel– oder Kundendaten) mit Hilfe des GPMS bei der (automatisierten) Ausführung der Aktivitäten genutzt werden.

Des Weiteren ist es möglich Services aufzurufen. Beispielsweise kann die Abfrage der Bonität eines Kunden mit Hilfe eines Web Services⁷¹ durchgeführt werden.

Die Abbildung 18 fasst die Differenzierungen der Systeme⁷² zusammen und stellt einen Überblick über deren verfügbare Funktionalitäten dar.

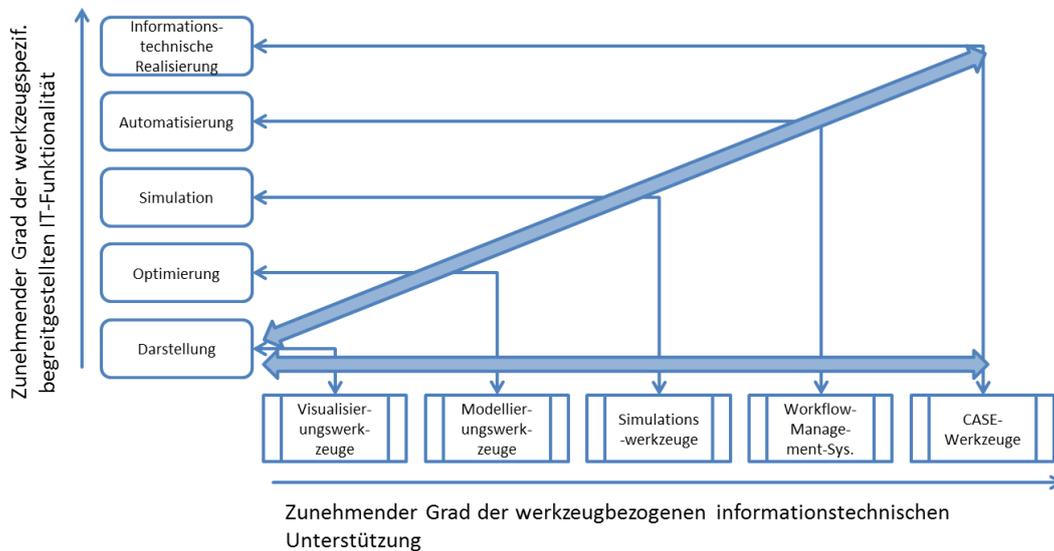


Abbildung 18: Übersicht von Software zur Unterstützung des GPM nach [Sp11].

Der Grad der informationstechnischen Unterstützung (horizontale Achse in Abbildung 18) reicht von einer reinen Visualisierung der Inhalte bis hin zur kompletten Computerunterstützung (CASE⁷³). Gleichzeitig nimmt die werkzeugspezifische Funktionalität (vertikale Achse in Abbildung 18) mit dem Grad der informationstechnischen Unterstützung zu. Dies gilt in der Regel auch dann, wenn ein Standard existiert. Zum Beispiel kann zur Automatisierung von Geschäftsprozessen ein BPMN Modell definiert sein, welches von verschiedenen Workflow-Management-Systemen unterschiedlich unterstützt wird.

⁷¹ Web Services ermöglichen die Komposition verschiedener Dienste unterschiedlicher Anbieter, beispielsweise zur Ausführung komplexer Geschäftsprozesse [BSD14].

Vgl. W3C Definition: „A Web service is a software system designed to support interoperable machine-to-machine interaction over a network.(...)“, <http://www.w3.org/TR/ws-arch/>.

⁷²Die hohe Anzahl der auf dem Markt erhältlichen Systeme erschwert einen detaillierten Funktionsüberblick. Eine Fraunhofer Studie ermittelte z.B. allein in Deutschland 55 GPMS Anbieter, vgl. [Fr13, S. 13].

⁷³ CASE ist ein Oberbegriff und wird als Akronym für „Computer-aided software engineering“ (rechnergestützte Softwareentwicklung) genutzt. Beispielsweise können mithilfe eines GPM-Systems Geschäftsprozesse zuerst modelliert und dann automatisiert werden.

Neben der Vielfalt an Funktionalität müssen Geschäftsprozessmanagementsysteme über eine flexibel veränderbare Architektur verfügen, um der Dynamik von Anforderungen (vgl. [WK14]) entgegenwirken zu können. „Intelligente“ GPMS dienen z.B. zur Echtzeitüberwachung oder Analyse von Massendaten (BigData) und der automatischen Entscheidungsfindung (z.B. zur automatischen Entscheidung über die Vertrauenswürdigkeit einer Person). Gartner⁷⁴ definiert intelligente Geschäftsprozessmanagementsysteme [Gal2a] (engl. Abk. iBPMS) als Systeme, welche neben Echtzeitüberwachung und Massendatenauswertung, auch Funktionalität hinsichtlich Social Media und Interaktionsmanagement bereitstellen müssen. Beispielsweise kann dadurch eine „werkzeuggestützte Beratung“⁷⁵ implementiert werden, welche bei der Ausführung von Aktivitäten unterstützt [Ju12]. Des Weiteren ist die signifikant zunehmende Mobilität und die damit verbundene Notwendigkeit auch Geschäftsprozesse „mobil“ zu erfassen, auszuführen und zu überwachen, eine Herausforderung. Nach einer Studie von Symatec [Sy12] sind mobile Anwendungen bereits unternehmenskritisch im Einsatz, wobei 72 % der befragten Teilnehmer auf eigene mobile Applikationen setzen. Zudem verdeutlicht eine Studie der KPMG [KP14], dass für die befragten Unternehmen die Festlegung einer Strategie bzgl. mobiler Endgeräte die größte Herausforderung darstellt. Insbesondere die virtuelle Mobilität, also die Möglichkeit durch IT ortsunabhängig „präsent“ zu sein [ZKJ02], gewinnt an Bedeutung, z.B. um Geschäftsprozessaktivitäten schneller und flexibler zu auszuführen.

In welchem Maß die (informationstechnische) Unterstützung angemessen ist, hängt von unterschiedlichen Parametern (z.B. Häufigkeit der Ausführung, Strukturierungsgrad des Geschäftsprozesses, ...) ab. Eine nicht-praktikable Unterstützung führt häufig dazu, dass diese (z.B. Checklistenformulare, Workflow-Systeme,...) nicht von den Beteiligten akzeptiert werden [JS12]. Zur Bestimmung des Unterstützungsgrads werden in der Literatur diverse Ansätze diskutiert. Beispielsweise beschreiben Jablonski und Seitz [JS13][Se16] unter anderem ein Reifegradmodell, in welchem (Geschäfts-)Prozesse auf Basis von drei Merkmalen (Güte, Unterstützung, Abbildung) analysiert werden.

Für eine weitere Beschreibung eines Geschäftsprozessmanagementsystems wird auf Abschnitt 4.4 verwiesen.

⁷⁴ Gartner, Inc., <http://www.gartner.com/technology/about.jsp>.

⁷⁵ Engl. „Process Guidance“.

2.3. Existierende Ansätze aus der agilen Softwareentwicklung

Ansätze der agilen Softwareentwicklung finden heutzutage in der betrieblichen Anwendung einen sehr hohen Zuspruch [Hu14]. In einer Umfrage von VersionOne [Di12][Ve13b, S.3] wurden über 4000 Teilnehmer befragt, wobei über 84% bestätigten, dass in ihrer Organisation agile Softwareentwicklung praktiziert wird. Neben diesem hohen Interesse betrieblicher Anwender, bestätigt eine steigende Anzahl von wissenschaftlichen Publikationen den Bedarf und das Interesse an agilen Ansätzen (vgl. [DD08][Di12]). Auch in anderen Bereichen als der Softwareentwicklung (z.B. dem Consulting oder Geschäftsprozessmanagement) kommen agile Ansätze immer häufiger zum Einsatz [CRA14, S.13-14]. Der Grund dafür ist die enge Verzahnung von IT und Business. Beispielsweise sind Business und IT gemeinsam⁷⁶ für die Erreichung von strategischen Zielen (z.B. Steigerung der Produktivität) verantwortlich [Ba10a]. Hierbei ist es insbesondere bei GPM Projekten, aufgrund einer sehr hohen Anzahl von (technischen, organisatorischen, projektspezifischen) Parametern, häufig nicht möglich eine verlässliche Einschätzung ex ante (z.B. Kostenschätzung) durchzuführen [MR13]. Des Weiteren wird der Einsatz von Software häufig direkt von der gewählten Modellierungssprache des Geschäftsprozessmanagements beeinflusst. Zum Beispiel werden Werkzeuge und Modellierungssprachen weitreichend (z.B. BPMN, BPEL) zur Automatisierung von Geschäftsprozessen eingesetzt [Ga13]. Je nach Funktionalität des Werkzeugs und verfügbaren Modellierungselementen können bestehende IT-Systeme wiederverwendet werden. Zusätzlich wird in der Softwareentwicklung, durch den Einsatz von agilen Ansätzen, sichergestellt, dass die Anforderungen und Ergebnisse übereinstimmen. Diese Herausforderung stellt sich auch im Geschäftsprozessmanagement hinsichtlich „gedachter“ Geschäftsprozesse und modellierter und automatisierter Geschäftsprozesse [TP13]. Aufgrund dieser Parallelen hinsichtlich Herausforderungen und Zielen, kann die Anwendung und Adaption von agilen Ansätzen der Softwareentwicklung im Bereich des Geschäftsprozessmanagements zu signifikanten Verbesserungen (z.B. Erhöhung der Qualität und Flexibilität) [MW13] führen.

⁷⁶In der Literatur wird weiterführend häufig der Begriff des Business-IT-Alignments zur Beschreibung des Verhältnisses zwischen Business-Anforderungen und IT-Systemen verwendet, vgl. [Si07][Ba10a][AGT13][Si07].

Nachdem der Begriff der Agilität in Abschnitt 2.3.1 diskutiert wurde, wird in den Abschnitten 2.3.2 bis 2.3.5 eine Auswahl weitverbreiteter agiler Ansätze zur Softwareentwicklung beschrieben: Scrum [SB02][Sc04b][SS13], Extreme Programming (XP) [Be00][BA04], Kanban [An10][An14] und Crystal Clear [Co02][Co03][Co05]. In Verbund mit den Erfahrungen und Anforderungen aus dem Geschäftsprozessmanagement wird auf Basis dieser Ansätze im weiteren Verlauf dieser Arbeit eine Vorgehensweise für ein agiles Geschäftsprozessmanagement entwickelt (vgl. Abschnitt 4).

2.3.1. Agilität

Erfolgreiches Geschäftsprozessmanagement zeichnet sich damit aus, dass auf unvorhergesehene Änderungen angemessen reagiert werden kann (z.B. Änderung des real ausgeführten Geschäftsprozesses aufgrund von veränderten Rahmenbedingungen). Des Weiteren sind die Akzeptanz von Änderungen (z.B. Einführung eines neuen Geschäftsprozesses in eine Organisation) und die Förderung von Wandel (z.B. Nutzung von Anwenderfeedback zur Geschäftsprozessoptimierung) grundlegende Ziele des Geschäftsprozessmanagements [MW13]. Zur Unterstützung derartiger Ziele finden agile Methoden in unterschiedlichen Bereichen (z.B. Softwareentwicklung [HRS09]) ihre Anwendung [Br13b].

Eine einheitliche Definition des Begriffs der Agilität existiert in der Literatur⁷⁷ nicht. Highsmith [Hi02a, S.xxiii] beschreibt Agilität⁷⁸ als die Fähigkeit auf Änderungen reagieren oder selbst einen Wandel einleiten zu können. Hierbei ist nach Highsmith die Agilität einer Organisation relativ und muss mit Mitbewerbern verglichen werden. Eine „höhere“ Agilität besitzt die Organisation, die im Vergleich zu anderen (in einer turbulenten Geschäftsumgebung) am besten reagieren kann. Als weiteren Aspekt von Agilität nennt Highsmith die Fähigkeit, ein Gleichgewicht zwischen Flexibilität und

⁷⁷ Leute unterteilt daher in „fünf Strömungen“: 1) Agilität als Verzicht, 2) Agilität als Fokussierung auf den Menschen als Teil eines Systems, 3) Agilität aus der industriellen Produktion, 4) Agilität als Referenzierung des Agilen Manifests und 5) Agilität als Weiterentwicklung der Softwareentwicklung, vgl. weitere Ausführung in [Le14, S.19ff].

⁷⁸ Vgl. engl. Übersetzung: „Agility is the ability to both create and respond to change in order to profit in a turbulent business environment“.

Stabilität zu schaffen. In [CF04, S.39f] beziehen Conboy & Fitzgerald Agilität⁷⁹ auf die ständige Bereitschaft re- oder pro-aktiv auf Änderungen reagieren zu können. Als Grundlage von Agilität identifizieren Conboy & Fitzgerald die Kombination von qualitativ hochwertigen, einfachen und wirtschaftlichen Komponenten und der Pflege von Beziehungen mit der Umgebung.

Im Gegensatz zu Highsmith und Conboy & Fitzgerald wird in [Au05, S.20f] Agilität⁸⁰ nicht aus der Organisationssicht sondern aus Kundensicht definiert. Augustine definiert Agilität als die Fähigkeit nicht vorhersagbaren und dynamischen Wandel zu erkennen und zu adaptieren mit dem Ziel, „Customer value“ trotz des Wandels zu generieren. Des Weiteren nennt Augustine, neben Flexibilität versus Stabilität (vgl. Highsmith) weitere konträre Beziehungen: Ordnung versus Chaos, Planung versus Ausführung, Optimierung versus Erforschung und Kontrolle versus Geschwindigkeit.

Verginadis et al. [Ve13a, S.120] definieren Agilität⁸¹ als Fähigkeit bestehende Diskrepanzen so schnell wie möglich zu identifizieren und eine passende Adaption in Zusammenhang mit dem (Neuen, Veränderteren) zu erstellen. Eine weitere Beschreibung von Agilität liefert das Agile Manifest [Be01]. Das Agile Manifest wurde im Jahre 2001 formuliert und besteht aus vier zentralen Werten, sowie zwölf Prinzipien. Es spiegelt die Philosophie der agilen Entwicklung wider [BK13a] und beschreibt somit den Begriff der Agilität grundlegend. Die Tabelle 3 listet die vier Werte des Agilen Manifests.

Tabelle 3:Werte des Agilen Manifest [Be01] (Deutsche Übersetzung)

1	Individuen und Interaktionen mehr als Prozesse und Werkzeuge
2	Funktionierende Software mehr als umfassende Dokumentation
3	Zusammenarbeit mit dem Kunden mehr als Vertragsverhandlung
4	Reagieren auf Veränderung mehr als das Befolgen eines Plans

⁷⁹ Vgl. engl. Übersetzung: „(...) the continual readiness of an entity to rapidly or inherently, proactively or reactively, embrace change, through high quality, simplistic, economical components and relationships with its environment.”

⁸⁰ Vgl. engl. Übersetzung: „Agility is the ability to deliver customer value while dealing with inherent project unpredictability and dynamism by recognizing and adapting to change. It is the ability to balance stability with flexibility, order with chaos, planning with execution, optimization with exploration and control with speed to deliver customer value reliably in the face of uncertainty and change.”

⁸¹ Vgl. engl. Übersetzung: „Agility is the ability of a subject to lead as quickly as possible, on the one hand, to the detection of its mismatch to a given context, on the other hand, to the setting up of the required adaptation.”

An erster Stelle der Werte stehen Individuen und deren Interaktionen. Anstatt Prozesse und Werkzeuge in den Mittelpunkt zu stellen, nimmt der Mensch die zentrale Rolle ein (vgl. [CH01]). Der zweite Wert beschreibt die Beziehung zwischen funktionierender Software und einer umfassenden Dokumentation. Obwohl eine funktionierende Software im Vordergrund steht, darf nicht dem Irrtum erlegen werden, keine Dokumentation zu erstellen (vgl. [BK13a]). Eine Verfeinerung der Beschreibung des Individuen-Begriffs wird mit dem dritten Wert erbracht. Die Zusammenarbeit mit dem Kunden wird höher bewertet als Vertragsverhandlungen. Eine vertragliche Vereinbarung als rechtliche Grundlage wird hierdurch nicht ersetzt [Pr12]. Dennoch soll die Beziehung zu dem Individuum „Kunden“ gemeinschaftlich sein [Tr12]. Als vierter Wert wird das Reagieren auf Veränderung höher bewertet als das Befolgen eines Plans. Insbesondere bei Unsicherheiten (z.B. ob ein geplantes Implementierungskonzept noch den Anforderungen entspricht) wird das Befolgen eines Plan durch flexibles Reagieren ersetzt. Eine vollständige Planung von Beginn an wird somit nicht unmittelbar notwendig. Basierend auf den vier Werten des Agilen Manifests (vgl. Tabelle 3) werden zwölf Prinzipien definiert [Be01]. Diese Prinzipien verfeinern die in den vier Leitsätzen definierten Werte und dienen zur konkreten Beschreibung der täglichen Arbeit in einem agilen Projekt. In Tabelle 4 werden die Prinzipien dargestellt.

Tabelle 4: Zwölf Prinzipien auf Basis des Agilen Manifest [Be01]

1	Unsere höchste Priorität ist es, den Kunden durch frühe und kontinuierliche Auslieferung wertvoller Software zufrieden zu stellen.
2	Heiße Anforderungsänderungen selbst spät in der Entwicklung willkommen. Agile Prozesse nutzen Veränderungen zum Wettbewerbsvorteil des Kunden.
3	Liefere funktionierende Software regelmäßig innerhalb weniger Wochen oder Monate und bevorzuge dabei die kürzere Zeitspanne.
4	Fachexperten und Entwickler müssen während des Projektes täglich zusammenarbeiten.
5	Errichte Projekte rund um motivierte Individuen. Gib ihnen das Umfeld und die Unterstützung, die sie benötigen und vertraue darauf, dass sie die Aufgabe erledigen.

6	Die effizienteste und effektivste Methode Informationen an und innerhalb eines Entwicklungsteams zu übermitteln, ist im Gespräch von Angesicht zu Angesicht.
7	Funktionierende Software ist das wichtigste Fortschrittsmaß.
8	Agile Prozesse fördern nachhaltige Entwicklung. Die Auftraggeber, Entwickler und Benutzer sollten ein gleichmäßiges Tempo auf unbegrenzte Zeit halten können.
9	Ständiges Augenmerk auf technische Exzellenz und gutes Design fördert Agilität.
10	Einfachheit, also die Kunst, die Menge nicht getaner Arbeit zu maximieren, ist essenziell.
11	Die besten Architekturen, Anforderungen und Entwürfe entstehen durch selbstorganisierte Teams.
12	In regelmäßigen Abständen reflektiert das Team, wie es effektiver werden kann und passt sein Verhalten entsprechend an.

Die ersten drei Prinzipien beschreiben die Fokussierung auf den Kunden. Nach [Be01] wird ein zufriedener Kunde durch die Auslieferung von „wertvoller“ Software erreicht und die Fähigkeit, schnell auf sich ändernde Anforderungen zu reagieren. Der Kunde soll demnach so früh wie möglich eine Software zur Anwendung erhalten, welche zur Generierung von Mehrwerten (z.B. Kosteneinsparung) genutzt werden kann. Des Weiteren beschreiben die Prinzipien, zum einen die Zusammenarbeit des Projektteams, zum anderen Eigenschaften in Bezug auf die zu erstellende Software. Beispielsweise soll das Projektteam selbstorganisiert interagieren, wobei insbesondere der Austausch von Fachexperten und Entwicklern angestrebt wird (vgl. u.a. Tabelle 4 – Prinzipien 4 und 11). Zusätzlich steht die zu erstellende Software im Mittelpunkt des agilen Projekts. Das zu erstellende Produkt – die Software – ist das maßgebliche „Fortschrittsmaß“. Der Einsatz von modernsten Implementierungstechniken (z.B. zur Erstellung einer Softwarearchitektur oder der Implementierung von Funktionen) bildet hierbei die notwendige Basis.

2.3.2. Scrum

Scrum basiert auf der Annahme, dass der Prozess der Softwareentwicklung nicht vollständig vorhersehbar ist [Sc95]. Scrum beschreibt eine agile Methode zum Management eines (Software-)Projekts (vgl. [SB02][Sc04b][SS13]), wobei die Vorgehensweise und der damit zusammenhängende Prozess im Vordergrund stehen [Ha10a]. Die empirische Prozesssteuerung erfolgt auf den drei Werten „Transparency“, „Inspection“ und „Adaptation“. Die geforderte Transparenz soll beispielsweise durch ein gemeinsames Verständnis von Arbeitsergebnissen erreicht werden. Des Weiteren fokussiert Scrum auf eine ständige Überprüfung („Inspection“) des Projektfortschritts. Werden Missstände entdeckt, zum Beispiel, dass Arbeitsergebnisse in Form von implementierter Software nicht den ex ante definierten Anforderungen entsprechen, muss die Vorgehensweise angepasst werden. Die Durchführung von Scrum erfordert sowohl die Definition eines Scrum Teams, als auch das Abhalten von bestimmten Ereignissen und das Verwalten von erwarteten Artefakten.

Die Tabelle 5 stellt die einzelnen Rollen eines Scrum Teams dar. Die erste Rolle, die des Product Owners, ist für die Verwaltung der Anforderungen verantwortlich. Zusätzlich ist der Product Owner für die Wertmaximierung des zu entwickelnden Produkts zuständig. Zudem beschreiben Schwaber & Sutherland [SS13], dass diese Rolle ausschließlich von einer Person besetzt werden darf.

Tabelle 5: Rollen in Scrum [SS13]

1	Product Owner
2	Development Team
3	Scrum Master

Das Entwicklungsteam stellt die zweite Rolle eines Scrum Teams dar. Das Entwicklungsteam verantwortet gemeinsam die Entwicklung (inkl. Test) und stellt nach einer Iteration das neue Produktinkrement zur Verfügung. Das Entwicklungsteam arbeitet hierbei selbstorganisiert.

Die Größe des Entwicklungsteam wird von Scrum nicht festgelegt⁸². Der Scrum Master (dritte Rolle) überwacht die Durchführung des Projekts. Aktivitäten, die nicht der Scrum Methode entsprechen, werden durch den Scrum Master berichtigt. Beispielsweise vermittelt der Scrum Master Techniken zur Interaktion (z.B. zum „richtigen“ Verständnis von agilen Werten). Außerdem werden sowohl Product Owner, Entwicklungsteam, als auch externe Beteiligte durch den Scrum Master unterstützt (z.B. Koordination der Scrum Ereignisse).

Zur Sicherstellung einer kontinuierlichen Kommunikation sind in Scrum Ereignisse definiert. Die einzelnen Ereignisse sind in Tabelle 6 dargestellt. Das zentrale Ereignis in Scrum, der Sprint, wird genutzt, um ein (potentiell nutzbares) Inkrement der zu entwickelnden Software zu erstellen. Der Sprint wird durch eine sogenannte Time Box begrenzt und ist maximal vier Wochen lang [SS13]. Neben der zeitlichen Begrenzung sind in Scrum keine Regeln (z.B. vorgegebene Programmiertechnik etc.) definiert, wie das Sprintziel erreicht werden muss. Einen Abbruch eines Sprints kann nur durch die Rolle des Product Owners vorgenommen werden.

Tabelle 6: Ereignisse in Scrum [SS13]

1	Sprint
2	Sprint Planning
3	Daily Scrum
4	Sprint Review
5	Sprint Retrospective

Innerhalb eines Sprints treten weitere Ereignisse (einmalig oder wiederkehrend) auf.

Im Sprint Planning sind zwei Fragestellungen von besonderer Bedeutung:

⁸² Schwaber & Sutherland [SS13] nennen drei bis neun Entwickler als Zielgröße eines Entwicklungsteams, vgl. : „Optimal Development Team size is small enough to remain nimble and large enough to complete significant work within a Sprint. Fewer than three Development Team members decrease interaction and results in smaller productivity gains. Smaller development Teams may encounter skill constraints during the Sprint, causing the Development Team to be unable to deliver a potentially releasable Increment. Having more than nine members requires too much coordination. Large Development Teams generate too much complexity for an empirical process to manage. (...)” [SS13, S.6].

- Was wird in dem neuen Sprint entwickelt?
- Wie kann das zu liefernde Inkrement erreicht werden?

Die „Was“-Frage wird durch die Analyse des Product Backlogs beantwortet. Der Product Owner beschreibt hierbei das zu erreichende Ziel des Sprints (und somit auch die im Product Backlog enthaltenen Anforderungen). Des Weiteren diskutiert das Entwicklungsteam die (zeitliche) Realisierbarkeit (z.B. mit Hilfe von Aufwandsschätzungen). Zur Beantwortung der zweiten Frage wird die technische Planung und Implementierung verfeinert (z.B. durch die Erstellung eines Systementwurfs).

Das Daily Scrum ist eine täglich durchgeführte 15-minütige Abstimmung, in der die Aktivitäten des Scrum Teams synchronisiert werden. Hierbei beantworten alle Mitglieder des Entwicklungsteams (mindestens) folgende Fragenstellungen:

- Was wurde am Vortrag erreicht?
- Was soll heute erledigt werden?
- Werden Hindernisse erwartet, die die Entwicklung behindern können?

Durch das Daily Scrum wird die Kommunikation aller Mitglieder des Entwicklungsteams gefördert. Zusätzlich trägt es zur Steigerung der Transparenz bei (z.B. durch die tägliche Überprüfung des Projektfortschritts). Das Sprint Review und die Sprint Retrospektive beenden einen Sprint. Innerhalb des Sprint Reviews wird das fertiggestellte Inkrement überprüft und das Product Backlog angepasst. Der Product Owner entscheidet hierbei, welche Anforderungen als „Done“ gekennzeichnet werden können. Zu diesem Ereignis können neben dem Scrum Team auch externe Teilnehmer eingeladen werden (z.B. weitere Stakeholder etc.). Die Sprint Retrospektive dient zur Analyse des vergangenen Sprints und soll zur weiterer Verbesserung der Aktivitäten beitragen. Beispielsweise können neue Lösungen zur Steigerung der Produktqualität diskutiert werden.

Des Weiteren definiert Scrum sogenannte Artefakte (vgl. Tabelle 7).

Tabelle 7: Artefakte in Scrum

1	Product Backlog
2	Sprint Backlog
3	Increment

Das Product Backlog ist eine Liste von Anforderungen an das zu entwickelnde Produkt. Anforderungen sind beispielsweise neue Funktionen, Fehlerbehebungen oder Verbesserungsvorschläge. Zusätzlich zu einer Beschreibung enthält jede Anforderung Indikatoren zur Reihenfolge, Wert und Aufwandsschätzung. Während der Ausführung von Scrum wird diese Liste kontinuierlich angepasst. Zum Beispiel können Anforderungen verändert oder neu priorisiert werden.

Das Sprint Backlog enthält die für den aktuellen Sprint aus dem Product Backlog übernommenen Anforderungen. Zusätzlich zu den bereits existierenden Informationen, werden an die Anforderungen das Sprint-Ziel und das Fertigstellungsdatum angefügt. Werden im Verlauf des Sprints weitere Arbeiten nötig, werden diese zusätzlich in den Sprint Backlog aufgenommen. Das Artefakt des Inkrements ist das Ergebnis aus allen Sprints. Das neue Inkrement, welches den Status „Done“ besitzen muss, wird im Sprint Review überprüft.

Die Abbildung 19 skizziert den generellen Ablauf der Scrum-Vorgehensweise auf Basis der Scrum Ereignisse und Scrum Artefakte.

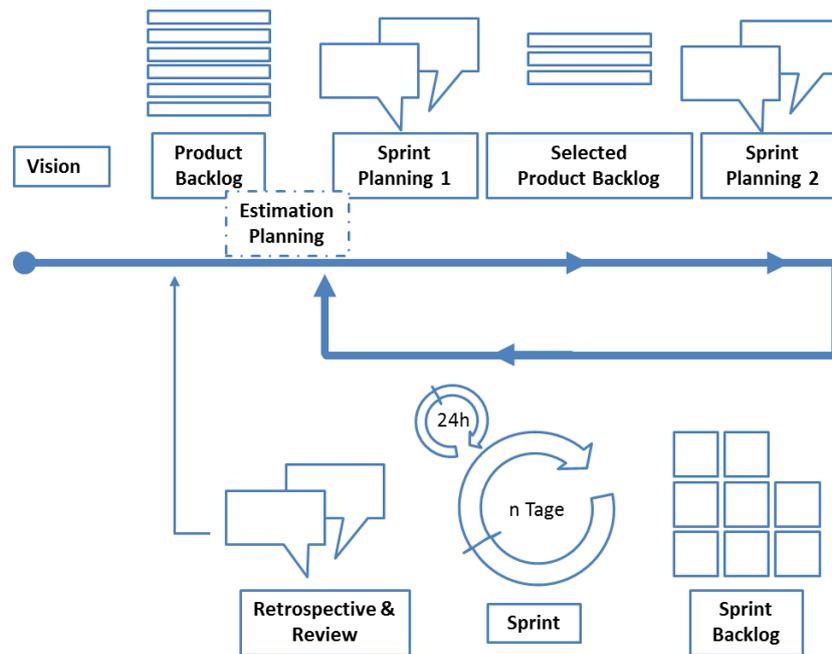


Abbildung 19: Vorgehensweise in Scrum (vgl. [GI13] [DKS13])

Nachdem eine Vision in Bezug auf das zu entwickelnde Produkt definiert wurde, werden Anforderungen in das Product Backlog abgelegt. Neben der reinen Pflege des Backlogs kann das Scrum Team ein „Estimation Meeting“⁸³ durchführen, um bereits vor der konkreten Implementierung eine (grobe) Aufwandsschätzung zu erhalten. Darauf folgen das erste Sprint Planning Meeting, in welchem die zu entwickelnden Anforderungen ausgewählt werden (vgl. „Selected Product Backlog, Abbildung 19). Nach der Durchführung des zweiten Sprint Planning Meetings, wird das Sprint Backlog abgearbeitet. Nach Abschluss der Retrospektive und des Reviews startet der Scrum Vorgehensprozess erneut.

Zur Unterstützung der Scrum Vorgehensweise werden zusätzlich verschiedene Techniken angewendet. Beispielsweise werden sogenannte User Stories [Co04][DKS13] zur Formulierung von Anforderungen verwendet. Die zur Formulierung der User Stories verwendete Sprache ist die Gebrauchssprache⁸⁴ (u.a. Alltagssprache).

Des Weiteren besteht eine User Story im Allgemeinen aus maximal zwei Sätzen und beinhaltet beteiligte Rollen, die Beschreibung der Anforderung und den Nutzen. Die Abbildung 20 visualisiert ein Beispiel einer User Story.

⁸³ Nicht offizieller Bestandteil des Scrum Methode nach Schwaber & Sutherland [SS13]. Dennoch wird das Estimation Meeting in der Praxis durchgeführt, vgl. [DKS13].

⁸⁴ Vgl. Abschnitt 3.1.

Als **Sachbearbeiter** möchte ich nach dem Start meines Prozessportals **alle anstehenden Aufgaben angezeigt** bekommen, um einen **besseren Überblick** zu erhalten.

Abbildung 20: Beispiel einer User Story

Neben der Formulierung der User Stories werden häufig zusätzlich Akzeptanzkriterien formuliert. Sollen beispielsweise beim Start des Prozessportals (vgl. Abbildung 20) nur die zehn aktuellsten Aufgaben dem Sachbearbeiter angezeigt werden, kann dieses Akzeptanzkriterium ex ante definiert werden. Zudem (vgl. [DKS13]) müssen die Mitglieder des Scrum Teams über die User Stories diskutieren, um ein weiteres – besseres – Verständnis der knapp formulierten User Stories zu erhalten.

Als weiteres Beispiel einer häufig angewendeten Technik kann der sogenannte Planungspoker (Planning Poker⁸⁵) genannt werden [Co06]. Innerhalb des Planungspokers werden Aufwandsschätzungen zu den einzelnen Anforderungen abgegeben. Die Aufwandschätzung folgt dabei einem strukturierten Prozess: Als erstes stellt der Product Owner die User Story den Beteiligten vor. Fragen zu dieser User Story (z.B. aufgrund von Verständnisproblemen) werden durch den Product Owner beantwortet. Anschließend bewertet jedes Teammitglied verdeckt (ohne anderen Teammitgliedern die Aufwandsschätzung mitzuteilen) die Schwierigkeit der User Story bzgl. der ihrer Umsetzung. Die Bewertung erfolgt mit Hilfe einer zuvor festgelegten Skala, beispielsweise einer Skala von 1 bis 10, wobei eine 1 „sehr einfach“ oder eine 10 „sehr schwer“ symbolisiert. Nachdem alle Teilnehmer eine Aufwandsschätzung abgegeben haben, werden die Ergebnisse gleichzeitig aufgedeckt. Mitglieder mit der niedrigsten und höchsten Schätzung werden aufgefordert ihre Aufwandsschätzung zu begründen. Dieser Planungspokerprozess wird solange wiederholt, bis ein Konsens aller Beteiligten ermittelt ist.

⁸⁵ Der Begriff des „Planning Poker“ ist eine geschützte Warenbezeichnung der Mountain Goat Software, <http://www.mountaingoatsoftware.com>. Planungspoker basiert auf der sogenannten Delphi Methode, einem Prognoseverfahren. Die von einem Moderator gestellten Fragen werden von den beteiligten Experten anonym beantwortet und danach ausgewertet. Dieser Vorgang wiederholt sich, bis ein gemeinsamer Konsens zur Prognose gefunden wird. Für weitere Ausführungen sei auf [DH63][LT75] [Th10, S.36f] verwiesen.

2.3.3. Extreme Programming

Extreme Programming wurde von Kent Beck entwickelt und gilt als sehr „leichtgewichtige“ Methode zur agilen Softwareentwicklung [Be00][BA04]. XP beschreibt einen Ansatz, welcher auf „(...) Wesentliches fokussiert und auf organisatorischen Ballast soweit wie möglich verzichtet.“ [Ru01, S.2]. Beispielsweise verzichtet XP auf die Erstellung von Dokumentationen. Beck unterteilt XP in drei Kategorien: Werte, Prinzipien und Praktiken.⁸⁶ Die Werte bilden die Basis von XP und werden durch die Prinzipien zu detaillierteren Handlungsempfehlungen verfeinert. Die Praktiken beschreiben Techniken, welche direkt während des Softwareentwicklungsprozesses angewendet werden.

Die fünf Werte von XP sind in der Tabelle 8 dargestellt.

Tabelle 8: Werte in XP [BA04]

1	Communication
2	Simplicity
3	Feedback
4	Courage
5	Respect

Der Begriff der Kommunikation wird als erster Wert von Beck genannt. Ein kontinuierlicher Informationsaustausch aller Projektbeteiligten dient zur Vermeidung von Problemen. Beispielsweise werden Fehlinterpretationen von Anforderungen durch eine intensive Diskussion vermieden. Zudem beschreibt der zweite Wert das Bedürfnis nach Einfachheit. Mit Hilfe von XP soll Software entwickelt werden, welche auf einfachen, schlanken Konzepten basieren. Insbesondere Anforderungen, die in der Zukunft möglicherweise auftreten könnten müssen nach Beck ignoriert werden. Feedback ist der dritte Wert von XP. Durch die Durchführung von kurzen Iterationen muss kontinuierlich

⁸⁶ Die in dieser Arbeit dargestellte Einteilung beruht auf der zweiten Ausgabe von XP, vgl. [BA04]. In dieser zweiten Ausgabe erweiterte bzw. veränderte Beck die Methode und deren Werte, Prinzipien und Praktiken auf Basis von neuen Erfahrungen. In der ersten Veröffentlichung wurden beispielsweise nur vier Prinzipien genannt (communication, simplicity, feedback, courage), vgl. [Be00].

Feedback des Kunden eingeholt und analysiert werden. Des Weiteren fordert XP die Durchführung von Tests über den ganzen Softwareentwicklungsprozess (z.B. Unit-Test durch Entwickler, Akzeptanztest durch Kunden,...) zur Erfassung von Feedback. Die abstrakten Begriffe Mut und Respekt stellen die vierten und fünften Werte dar. Nur durch die mutige (und ehrliche) Umsetzung der XP-Prinzipien -und Praktiken kann ein erfolgreiches Projekt entstehen. Respekt gegenüber allen Projektbeteiligten ist hierbei essentiell. Zusätzlich zu den Werten werden vierzehn Prinzipien definiert. Die Prinzipien verfeinern die abstrakten Werte und sollen zur leichten Ein –und Durchführung von XP dienen, indem alle angewendeten Praktiken auf die Einhaltung dieser Prinzipien geprüft werden [Ha10a]. Die Tabelle 9 listet die XP Prinzipien auf.

Tabelle 9: Liste der XP Prinzipien [BA04]

1	Humanity
2	Economics
3	Mutual Benefit
4	Self-Similarity
5	Improvement
6	Diversity
7	Reflection
8	Flow
9	Opportunity
10	Redundancy
11	Failure
12	Quality
13	Baby steps
14	Accepted Responsibility

An erster Stelle der Prinzipien steht der Begriff der Menschlichkeit. Der Softwareentwicklungsprozess wird von Menschen durchlebt. Alle XP-Projektbeteiligten sollen menschenwürdige Arbeitsbedingungen vorfinden (z.B. Integration im Team, Respekt gegenüber anderen,...). Zudem wird von allen Projektbeteiligten das Bewusstsein zum wirtschaftlichen Handeln gefordert. Das dritte Prinzip beschreibt die Anforderung entstandene Vorteile (z.B. Einführung von neuen Techniken) zu teilen. Durch das Prinzip der Selbstgleichheit (viertes Prinzip) werden die Projektbeteiligten motiviert bereits angewendete Problemlösungen auch für neue Probleme einzusetzen. Redundanz (zehntes Prinzip), beispielsweise durch parallel entwickelte Problemlösungen, wird in XP dennoch akzeptiert.

Eine ständige Verbesserung (fünftes Prinzip) durch Reflexion (siebtes Prinzip) stellen weitere Prinzipien von XP dar. Zum Beispiel werden in einer Planungsphase Fehler von vergangenen Iterationen berücksichtigt und somit das Vorgehen verbessert. Jedes Problem ist aus der Perspektive von XP auch eine Gelegenheit der Verbesserung (Prinzip neun). Innerhalb von XP wird versucht, ein vielfältiges (sechstes Prinzip) Projektteam in einen einheitlichen Fluss (achtes Prinzip) zu bringen. Diese Vielfältigkeit fördert kreative Lösungsansätze, welche in kleinen Schritten (dreizehntes Prinzip) iterativ implementiert werden können. Das implementierte Produkt muss hierbei einem hohen Qualitätsanspruch genügen (zwölftes Prinzip), wobei das Projektteam die Verantwortung annimmt und akzeptiert (vierzehntes Prinzip).

Die an die Prinzipien anschließenden Praktiken sind in primäre und begleitende Praktiken unterteilt⁸⁷. Beispielsweise dürfen in XP-Projekten keine Überstunden entstehen. Nach Beck sind Überstunden zu vermeiden, um Konzentrationsfähigkeit, Qualität und die Freude an der Arbeit zu gewährleisten. Eine weitere Praktik ist die Paarprogrammierung. Hierbei wird die Implementierung von Software durch zwei Entwickler durchgeführt. Während der erste Entwickler programmiert, überprüft der zweite Entwickler die Implementierung fortlaufend. Mit Hilfe dieses Vier-Augen-Prinzips können Fehler schneller entdeckt oder direkt vermieden werden. Als weiteres Beispiel einer Praktik kann das Verfolgen eines „einfachen Designs“ genannt werden. Die in XP entwickelte Software soll auf einer möglichst einfachen Lösung basieren.

⁸⁷ Für eine weitere ausführliche Beschreibung aller Praktiken sei auf [Be00][BA04][Ha10a] verwiesen.

Anforderungen die zum Zeitpunkt des Design noch nicht bekannt sind werden ignoriert: „Was noch nicht bekannt ist, kann auch nicht designt werden.“ [Ha10a, S.27].

Die Werte, Prinzipien und Praktiken werden während den Softwareentwicklungstypischen Phasen von Planung, Design, Implementierung und Testen angewendet. In welcher Reihenfolge die Praktiken zum Einsatz kommen ist in XP nicht spezifiziert und kann agil festgelegt werden [Ha10a]. Die Abbildung 21 visualisiert ein XP-Vorgehensmodell nach [We00].

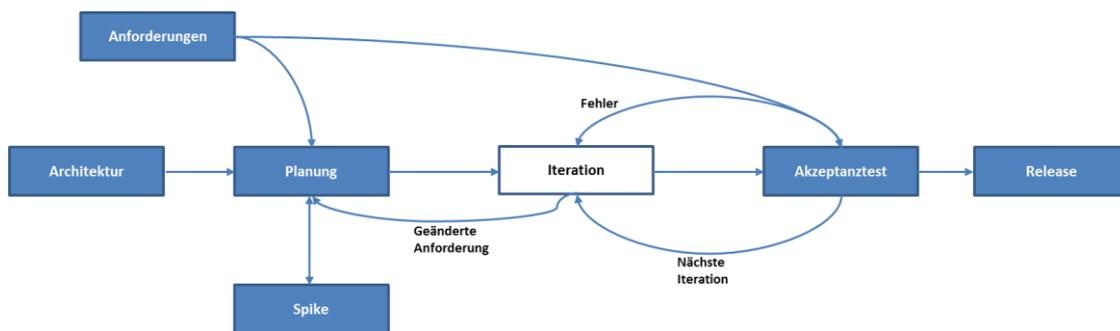


Abbildung 21: Vorgehensmodell in XP [We00]

In einem Architektur-Meeting und möglichen prototypischen Akzeptanztests werden die Vision, das Budget und die Zeitplanung (vgl. [BA04]) der zu entwickelnden Software definiert. Die nachfolgende Phase der Planung dient neben der Detaillierung der Vision zur Analyse und Strukturierung der Anforderungen. Zusätzlich können aus der Planungsphase heraus sogenannte Spikes⁸⁸ gestartet werden. Mit Hilfe eines Spikes können beispielsweise einzelne Anforderungen auf prinzipielle Machbarkeit geprüft werden. Als weiteres Element der Vorgehensweise wird eine Iteration gestartet. Jede Iteration⁸⁹ besteht aus einer Planungs-, entwicklungs- und testphase, wobei die zu verwendeten Tests bereits vor der Implementierung erstellt werden.

Treten Unsicherheiten in Bezug auf die zu implementierende Anforderung auf wird der Kunde befragt. Kommt es hierbei zu einer geänderten Anforderung ist ein Rücksprung in die Planungsphase (vgl. Abbildung 21) möglich. Als weitere Phase schließt sich die Prüfung der Akzeptanz an. Während dem Akzeptanztest wird die aktuelle Version der zu entwickelnden Software getestet. Auftretende Fehler oder Änderungen der

⁸⁸ Vgl. dazu Beck [Be99, S.289f]: „(...) I call this a „spike“ because we are driving a spike through the entire design. We are not searching for completeness. Instead, we want to illustrate the kinds of responsibilities accepted by all the major objects in the system.“

⁸⁹ Vgl. <http://www.extremeprogramming.org/map/iteration.html>

Anforderungen haben einen Rücksprung in die vorgelagerte Phase zur Folge. Abschließend wird die fertiggestellte Software ausgeliefert (Release).

Ein Rollenkonzept wird in XP nicht explizit vorgegeben. Im Allgemeinen werden die Rollen des Kunden, Entwicklers und Managers festgelegt (vgl. [JAH00][Pr12]). Der Kunde ist Auftraggeber des Projekts und verantwortet die Formulierung der Anforderungen. Insbesondere durch die von XP geforderte ständige Anwesenheit des Kunden im Projektteam kann dieser fortlaufend Einfluss auf die Entwicklung (z.B. durch Beantwortung von fachlichen Fragen) nehmen. Die Rolle des Entwicklers übernimmt sowohl das Design der Software, als auch die Entwicklung von (automatisierten) Tests und der eigentlichen Software. Auf Basis der ständigen Kommunikation und Interaktion (z.B. Paarprogrammierung) aller Projektbeteiligten können Fragen und Probleme direkt gemeinschaftlich gelöst werden. Die Koordination des Projekts (z.B. Zeit –oder Ressourcenplanung) wird von der Rolle des Managers übernommen. Des Weiteren verantwortet der Manager den Kommunikations- und Interaktionsprozess aller Projektbeteiligten und unterstützt bei möglichen Komplikationen (z.B. Auflösung von Konflikten innerhalb des Projektteams).

2.3.4. Kanban

Die Kanban Methode wurde ursprünglich von Taiichi Ohno für den Automobilhersteller Toyota entwickelt. Die Methode zielt insbesondere auf die Optimierung des Produktionssteuerungsprozess und Vermeidung von Ballast ab. Zum Beispiel werden lokale Bestände minimiert, indem vorgelagerte Bestandteile eines Endprodukts nur produziert werden, sofern diese benötigt werden (vgl. [SU77][Ön88][Ep11]).

Anderson [An10][An14] adaptiert die „traditionelle“ Kanban Methode für den Bereich der Softwareentwicklung. Kanban ist kein vorstrukturiertes Vorgehensmodell, sondern beschreibt eine lose Zusammenstellung von Elementen (z.B. ein Set von Prinzipien, welche beschreiben wie Kanban angewendet werden muss) [Ep11, S.19].

Folglich ersetzt Kanban keine bereits existierende Methode, sondern setzt darauf auf. Wird noch keine Methode angewendet kann Kanban dennoch, beispielsweise zur Visualisierung von Prozessschritten, eingesetzt werden.

Zur Beschreibung von Kanban definiert Anderson vier grundlegende Prinzipien und sechs generelle Praktiken [An14]. Die Tabelle 10 stellt die Prinzipien von Kanban dar.

Tabelle 10: Prinzipien von Kanban [An14]

1	Start with what you do now
2	Agree to pursue incremental, evolutionary change
3	Respect the current process, roles, responsibilities & titles
4	Encourage acts of leadership at all levels in your organization

Das erste Prinzip verdeutlicht, dass Kanban kein Softwareentwicklungsprozess vorschreibt oder eine definierte Vorgehensweise fordert. Kanban wird folglich während der Ausführung der bestehenden Aktivitäten gestartet. Mögliche Widerstände (z.B. fehlendes Engagement der Mitarbeiter) aufgrund der Einführung einer neuen Methode werden somit minimiert. Das zweite Prinzip beschreibt den Bedarf der inkrementellen und evolutionären Veränderung. Nach Anderson ist ohne die Akzeptanz gegenüber einer kontinuierlichen Verbesserung keine passende Umgebung für die Anwendung der Kanban Methode vorhanden⁹⁰. Das dritte Prinzip beschreibt die Haltung gegenüber dem IST-Zustand. Sowohl bestehende Prozesse, Rollen, Verantwortlichkeiten als auch Job-Titel sollen respektiert werden. Dieser Respekt trägt zur Etablierung von Kanban in einer Organisation bei. Als viertes Prinzip nennt Anderson die Förderung von „Leadership“ auf allen Ebenen einer Organisation. Organisationsmitglieder sollen hierdurch ermutigt werden Veränderungen zu beschreiben und umzusetzen.

Neben den Kanban-Prinzipien definiert Anderson sechs weitere Praktiken. Die Tabelle 11 stellt diese Praktiken dar.

Tabelle 11: Allgemeine Praktiken [An14]

1	Visualize (the work, workflow and business risks)
2	Limit WIP

⁹⁰ Vgl. [An14]: „Without agreement that a slow, gentle, evolutionary, incremental approach is the right way forward then there won't be the right environment or management support for a Kanban initiative.”

3	Manage Flow
4	Make Policies Explicit
5	Implement Feedback Loops
6	Improve Collaboratively, Evolve Experimentally (using models/scientific method)

Als erste Praktik gilt, dass alles „Unsichtbare“ sichtbar gemacht werden soll. Beispielsweise soll der Prozess der Bearbeitung von Anforderungen visualisiert werden. Mögliche Engpässe (z.B. zu wenige Entwickler stehen zur Verfügung) können somit besser entdeckt werden (vgl. Abbildung 22). Die zweite Praktik beschreibt die Limitierung des Work In Progress (WIP). Durch dieses Kanban-System wird die Begrenzung gleichzeitig begonnener Arbeiten festgelegt. Zum Beispiel kann das WIP-Limit bei der Entwicklung auf „3“ festgelegt werden. Dies bedeutet, dass maximal 3 Anforderungen gleichzeitig entwickelt werden. Falls die Bearbeitung der Anforderungen nicht zufriedenstellend ist (z.B. zu lange Durchlaufzeit) wird das WIP-Limit reduziert, sodass die Anforderungen schneller durch das Kanban-System und den damit verbunden Prozess laufen können (vgl. Abbildung 22)⁹¹.

Als dritte Praktik beschreibt Anderson die Notwendigkeit vordefinierte Parameter zu messen und diese mit den SOLL- Kennzahlen zu vergleichen.

Der Prozess, welchen die Anforderungen durchlaufen (z.B. Implementierung, Test, Release) wird somit überwacht. Mögliche Schwachstellen (z.B. zu lange Durchlaufzeiten) können schneller identifiziert und beseitigt werden.

Der Fokus der Praktiken vier bis sechs liegt auf der Interaktion aller Projektbeteiligten. Informationen müssen explizit in der Organisation allen Beteiligten bekannt gemacht werden. Zusätzlich sollen zyklische Abstimmungen (z.B. tägliche Meetings) eine enge

⁹¹Zusätzlich wird auf Leopold & Kaltenecker verwiesen: „Kanban ist ein Pull-System - das bedeutet, dass immer „ein Stück Arbeit“ aus der gesamten Aufgabenmenge herausgenommen, bearbeitet und dann dem nachfolgenden Prozessschritt als fertig signalisiert wird. Fälschlicherweise wird dies manchmal mit der Wasserfallmethode in der Softwareentwicklung verglichen. Bei dieser traditionellen Vorgehensweise müssen immer alle Aufgaben fertiggestellt sein, die für einen Prozessschritt vorab definiert wurden. Dann werden fertige Aufgaben gesammelt an den nächsten Schritt weitergereicht. Genau das ist in Kanban aber nicht der Fall.“ [LK14, 3.2].

Synchronisation (z.B. Verbreitung von Neuigkeiten an alle Beteiligten) ermöglichen. Des Weiteren soll eine kontinuierliche Verbesserung nach Anderson gemeinsam von allen Beteiligten (und ggfs. unter Anwendung von wissenschaftlichen Methoden) erreicht werden.

Neben den Prinzipien und Praktiken können zur Unterstützung der Ausführung von Kanban unterschiedliche Techniken⁹² eingesetzt werden. Beispielsweise dienen „Daily Standup Meetings“ (vgl. [An10, S.82f]) zur täglichen Abstimmung. Diese Abstimmungen finden in der Regel am Morgen statt und dienen zum Austausch von Informationen. Die Projektbeteiligten beantworten sich gegenseitig hierbei insbesondere die Fragen „Was wurde gestern erreicht?“, „Was wird heute getan?“ und „Gibt es Probleme/ wird Hilfe benötigt?“. Eine weitere Technik kann als Diskussionsgrundlage angewendet werden. Das Kanban Board wird zur Strukturierung aller Anforderungen oder deren Status genutzt. Ein Beispiel ist in Abbildung 22 dargestellt.

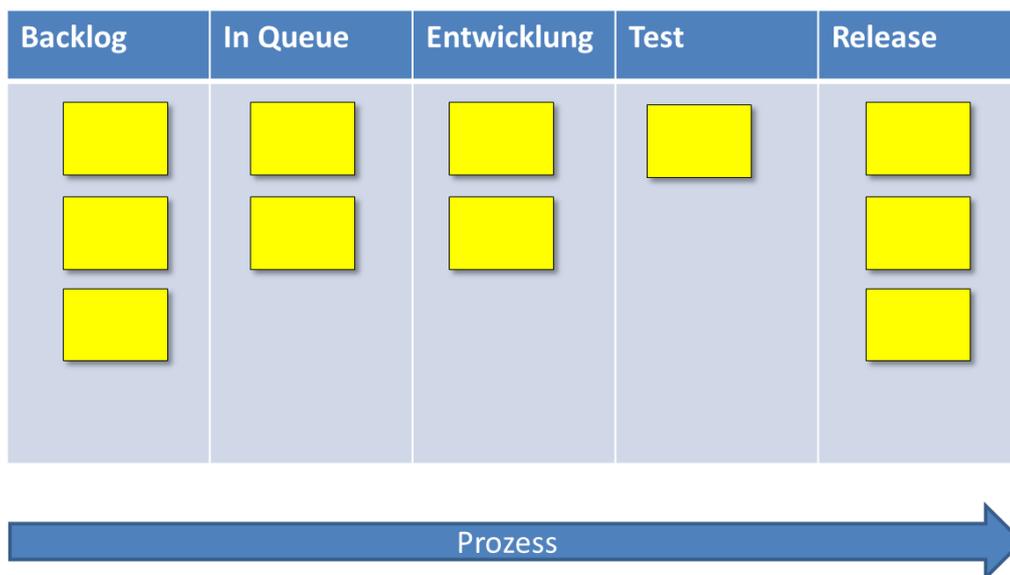


Abbildung 22: Beispiel eines Kanban Boards

Das in der Abbildung 22 dargestellte Kanban Board beinhaltet fünf Spalten. Jede Spalte enthält eine bestimmte Anzahl von Karten (vgl. gelbe Elemente, Abbildung 22), wobei jede Karte Informationen über die zu implementierende Anforderung enthält. Welche Struktur das Board besitzt wird in Kanban nicht festgelegt, nur dass die Spalten „In Queue“ und „Release“ enthalten sein sollten [LK14]. Alle Anforderungen der zu

⁹² Hinweis: Die unterschiedliche Verwendung von Begriffen (z.B. Prinzipien, Praktiken, Techniken,...) richtet sich nach den beschriebenen Methoden. Zur Beschreibung der zusätzlich in Abschnitt 2.3 diskutierten Methoden, werden möglicherweise die Begriffe mit einer anderen Bedeutung verwendet.

implementierenden Software sind im „Backlog“ abgelegt. Die Spalte „In Queue“ enthält die Anforderungen, welche als nächstes – in „Entwicklung“ – bearbeitet werden sollen. Nach der Fertigstellung der Entwicklung wird die Anforderung in „Test“ verschoben und getestet. Abschließend werden die implementierten Anforderungen in „Release“ abgelegt und können an den Kunden ausgeliefert werden.

Eine Anforderung wird auf einer sogenannten (Signal-) Karte⁹³ beschrieben. Diese Karte enthält detaillierte Informationen bzgl. der zu implementierenden Anforderung wie beispielsweise (vgl. [Ep11, S. 115]):

- eine eindeutige Bezeichnung der Anforderung,
- eine Kurzbeschreibung der Anforderung,
- ein Eintritts – und (erwartetes) Austrittsdatum und
- eine Information über die beteiligten Personen/Rollen.

Nach Anderson definiert die Kanban Methode keine Rollen (vgl. [KS10]). Dennoch können Rollen (z.B. Scrum Product Owner, vgl. Abschnitt 2.3.2) von anderen, existierenden Methoden übernommen werden, um beispielsweise Verantwortlichkeiten (z.B. Kontrolle von Lieferfristen) zu verteilen.

2.3.5. Crystal Clear

Die von Alistair Cockburn entwickelte Methode Crystal Clear (vgl. [Co02][Co03][Co05]) beschreibt eine Sammlung von Erfahrungen, welche im Kontext der agilen Entwicklung angewendet werden können. Im Gegensatz zu anderen Vorgehensweisen (z.B. XP, Scrum,...) beschreibt Crystal nicht den Ansatz „(...) „so muss man es machen“, sondern (...) „in dieser Situation haben Teams gute Erfahrung gemacht, jenes zu machen mit folgenden Konsequenzen“ (...)“ [Co09].

Crystal Clear ist Bestandteil einer Methodenfamilie („Crystal Family“) wie sie schematisch in Abbildung 23 dargestellt ist.

⁹³ Hierdurch erhält die Methode auch ihren Namen. Kanban stammt aus dem Japanischen und besteht aus „kan“ (dt. Signal) und „ban“ (dt. Karte) vgl. [Ep11, S.23] [LK14].

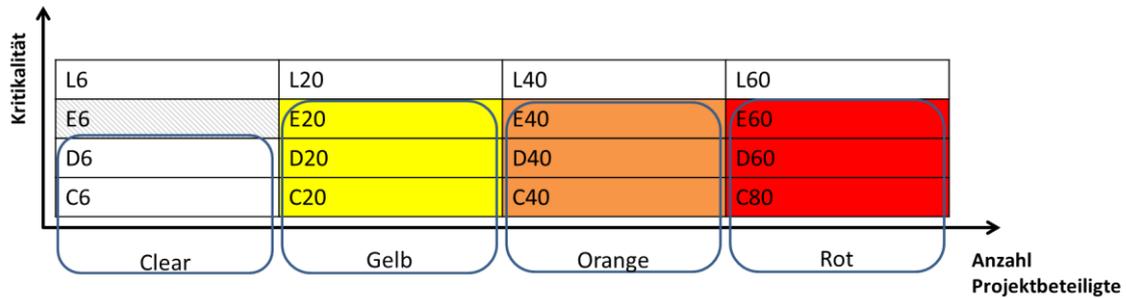


Abbildung 23: Crystal Familie nach [Co02]

Abhängig von den zwei Parametern (vgl. x-y-Achsen in Abbildung 23) „Kritikalität“ und „Anzahl der Projektbeteiligten“ wird die, für das Projekt passende Crystal-Variante ausgewählt und angewendet. Cockburn nutzt zur besseren Visualisierung der Anzahl von Projektbeteiligten verschiedene Farbtöne. Ein dunklerer Farbton symbolisiert eine höhere Anzahl von Projektbeteiligten. Der Parameter der Kritikalität wird von Cockburn in vier Ausprägungen unterteilt⁹⁴: „C“ (Verlust von Komfort), „D“ (Verlust von frei verfügbaren Geldern), „E“ (Verlust von kritischen Geldern) und „L“ (Verlust von Leben).⁹⁵ Muss beispielsweise ein Projekt mit 40 Beteiligten und Kritikalität D durchgeführt werden, muss die Variante Crystal Orange gewählt werden. Die in diesem Abschnitt ausführlich beschriebene Variante Crystal Clear eignet sich für „C6“ und „D6“ (vgl. Abbildung 23), wobei „E6“ abhängig von den Projektbeteiligten⁹⁶ zusätzlich für den Einsatz von Crystal Clear gewählt werden kann.

Unabhängig von der Einteilung der unterschiedlichen Varianten der Crystal Familie werden drei vorgeschriebene Basisprioritäten von Cockburn vorgegeben.

Tabelle 12: Basisprioritäten von Crystal

1	Safety in the project outcome
2	Efficiency in the development
3	Habitability of the conventions

⁹⁴ Vgl. engl. Übersetzung: C (loss of comfort), D (loss of discretionary moneys), E (loss of essential moneys), L (loss of life).

⁹⁵ Vgl. Cockburn [Co05, S.234]: „The name Crystal derives from these two dimensions [Kritikalität und Anzahl der Prozessbeteiligte], in an analogy with geological crystals, which also are characterized in two dimensions: color and hardness (...).“

⁹⁶ Vgl. Cockburn [Co02, S.166]: „(...) Crystal Clear does not explicitly address “essential moneys” projects, but that the team may be able to stretch Crystal Clear to such a situation.“

Sicherheit bzgl. des Projektergebnisses ist eine Priorität von Crystal im Allgemeinen. Projektprioritäten und Rahmenbedingungen werden daher besonders überwacht. Des Weiteren wird die Effizienz der Entwicklung als ausschlaggebende Messgröße festgelegt, um ökonomische Ziele nicht zu verfehlen. Zusätzlich muss die Anwendbarkeit der Methode gewährleistet sein⁹⁷. Die, durch die angewendete Methode, aufkommenden Pflichten und Vorschriften dürfen die Projektbeteiligten nicht belasten.

Um diese drei Basisprioritäten und die dadurch entstehenden Herausforderungen bewältigen zu können, sind in Crystal (Clear) sieben Eigenschaften definiert.

Tabelle 13: Die sieben Eigenschaften von Crystal

1	Frequent Delivery
2	Reflective Improvement
3	Osmotic Communication
4	Personal Safety
5	Focus
6	Easy Access to Expert Users
7	Technical Environment with Automated Tests, Configuration Management, and Frequent Integration

In Crystal Clear müssen nach Cockburn die Eigenschaften 1- 3 verpflichtend umgesetzt werden. Somit ist eine regelmäßige Auslieferung einer lauffähigen Software an den Kunden vorgeschrieben. Beispielsweise können dadurch Abweichungen von Kundenanforderungen unmittelbar erkannt werden. Des Weiteren zählen regelmäßige Abstimmungen zwischen den Projektbeteiligten zu den Crystal Eigenschaften. Innerhalb einer solchen Abstimmung wird reflektiert, welche Verbesserungsmöglichkeiten in z.B. die nächste Iteration übernommen werden können. Als dritte verpflichtende Eigenschaft

⁹⁷ Das engl. Wort "habitability" wurde frei mit "Anwendbarkeit" übersetzt. Vgl. Cockburn [Co05, S.233]: „The habitability priority means that the rules need to be such that the people on the team can live with them.“

von Crystal Clear beschreibt Cockburn eine osmotische Kommunikation. Eine osmotische Kommunikation ist möglich, sobald kleine Teams in einem Raum arbeiten, sodass durch die räumliche Nähe der Projektbeteiligten bereits Informationen ausgetauscht werden (vgl. [Ha10a]). Die weiteren Eigenschaften – Persönliche Sicherheit, Fokus, Einfache Kommunikation mit Anwendern, Technische Umgebung – müssen in Crystal Clear nicht verpflichtend eingehalten werden.

Crystal Clear Projekte werden in unterschiedliche Ebenen gegliedert. Cockburn unterscheidet hierbei zwischen dem Projekt, Delivery-Abschnitten und Iterations-Abschnitten (vgl. Abbildung 24). Im Unterschied zu einer Delivery beschreibt eine Iteration „nur“ das Ende eines internen Entwicklungsabschnitt, während das Ende einer Delivery die Abnahme von Usern (Anwender der implementierten Software) vorschreibt⁹⁸.

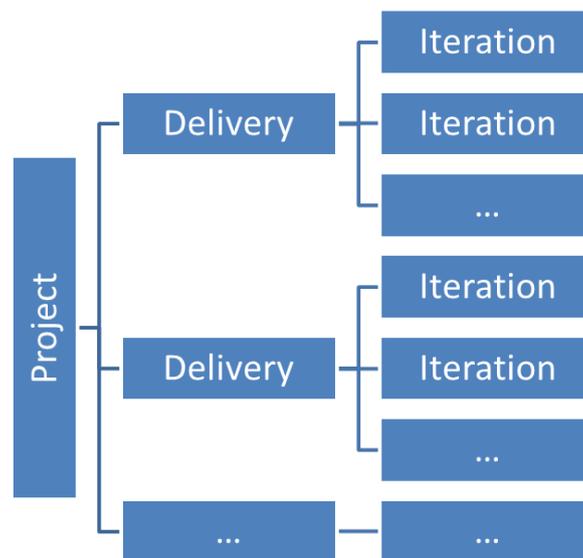


Abbildung 24: Unterteilung eines Crystal Clear Projekts (obere drei Ebenen)

Eine Iteration selbst kann noch einmal in „Day“, „Integration“ und „Episode“⁹⁹ unterteilt werden. Diese drei Ebenen dienen zur detaillierten Strukturierung. Hierbei wird sowohl die Tagesplanung („Day“), als auch die Planung einer Integration von Neuentwicklungen

⁹⁸ Vgl. des Weiteren Cockburn [Co05, S.133]: „In Crystal Clear, you are allowed to have just one iteration per delivery cycle, but if you do so, you must have some intermediate viewings by real users.”

⁹⁹ Cockburn verweist selbst auf Ward Cunningham bzgl. der Definition einer „Episode“: <http://c2.com/ppr/episodes.html>

berücksichtigt. Eine Episode ist die Basiseinheit eines Entwicklers und beschreibt z.B. eine neu entwickelte Funktion (inkl. Softwaredesign und Test).

In einem Crystal Clear Projekt werden insgesamt acht Rollen¹⁰⁰ definiert: Auftraggeber, erfahrener Anwender, Chefdesigner, Designer-Programmierer, Fachexperte, Koordinator, Tester und Autor (vgl. [Co05, S.140ff]). Die Rollen des Auftraggebers, Anwenders, Chefdesigners und Designer-Programmierer müssen durch unterschiedliche Personen besetzt werden. Der Auftraggeber verantwortet das Projekt und dessen Finanzierung. Diese Rolle entspricht daher auch der Bezeichnung des „Kunden“ [Ha10a]. Der erfahrene Anwender besitzt Wissen über die spätere Anwendung der zu implementierenden Software und dient als Auskunftsperson für die in der Entwicklung tätigen Rollen. Zusätzlich kann die Rolle des Fachexperten ggfs. durch den erfahrenen Anwender übernommen werden. Der Fachexperte besitzt Wissen über IST- und SOLL-Geschäftsabläufe. Die Rolle des Chefdesigners verantwortet die Führung des Projekts und das Design und die Implementierung der Software.

Neben dem Chefdesigner dient die Rolle des Designer-Programmierer¹⁰¹ zur Implementierung der einzelnen Kundenanforderungen eingesetzt. Die Rollen des Koordinators und Autors besitzen Aufgaben zur Koordinierung und Dokumentation. Beispielsweise übernimmt der Koordinator die Projektprotokollierung und Projektplanung, während der Autor die implementierte Software dokumentiert (z.B. in Form von Benutzerhandbüchern). Die Rolle des Testers verantwortet die Kontrolle der implementierten Anforderungen.

Zusätzlich zu den für Crystal Clear vorgeschriebenen Prioritäten, Eigenschaften, Strukturen und Rollen beschreibt Cockburn eine Vielzahl von Praktiken (vgl. [Co05, S.45ff]). Diese Praktiken sind für die Crystal Clear Methode nicht verpflichtend, unterstützen dennoch deren Anwendung. Cockburn unterteilt diese Praktiken noch einmal in Strategien und Techniken. Zum Beispiel kann die „Exploratory 360°“-Strategie angewendet werden, um zu Beginn eines Projekts Aspekte zu untersuchen, welche Auswirkungen auf die tatsächliche Ausführung haben. Hierbei werden zum Beispiel der

¹⁰⁰ Vgl. engl. Übersetzung: Sponsor, Expert User, Lead Designer, Designer-Programmer, Business Expert, Coordinator, Tester und Writer.

¹⁰¹ Die Verbindung der zwei Begriffe „Designer“ und „Programmierer“ zu dem Begriff Designer-Programmierer beschreibt Cockburn wie folgt [Co05, S.142]: „I merge the words „designer“ and „programmer“ in the designer-programmer role to highlight that each person both designs and programs (...).“

Geschäftswert („Business Value“) diskutiert, grundlegende Anforderungen festgelegt und das Projektteam fixiert. Ein weiteres Beispiel ist das „Methodology Shaping“. Diese Technik dient zur Optimierung der Methode hinsichtlich der aktuellen und zukünftigen Anwendung. Nicht-bewährte Praktiken werden ersetzt, andere erweitert, sodass diese in die Organisationsstruktur integriert werden können. Die „Burn down Charts“ dienen zur Visualisierung des Projektfortschritts und sind nach Cockburn ein einfaches aber effektives Mittel. Die Abbildung 25 skizziert ein Burn down Chart Beispiel. Die y-Achse des Charts visualisiert den Grad der Fertigstellung (z.B. in Prozent), die x-Achse stellt den zeitlichen Verlauf an (z.B. Zeit in Tagen). Die in Abbildung 25 dargestellten Linien (schwarz, rot) skizzieren sowohl den geplanten, als auch den aktuellen Projektlauf. Verzögerungen (z.B. durch Erfassung und Implementierung von veränderten Anforderungen) werden direkt deutlich.

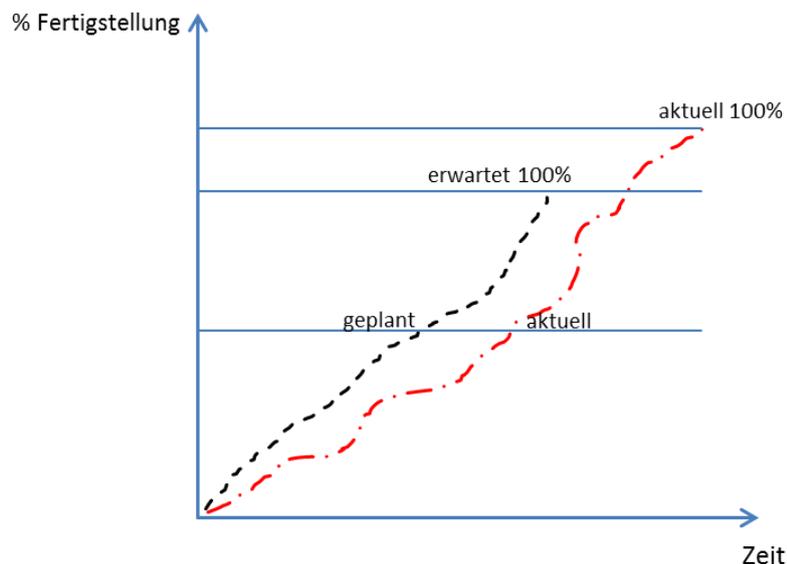


Abbildung 25: Burn down Chart ohne Meilensteine (vgl. [Co05, S.95])

Um eine höhere Zuverlässigkeit (z.B. Einhaltung des Zeitplans) beschreibt Cockburn die Notwendigkeit weitere Meilensteine (neben dem Start- und Endpunkt) zu definieren. Beispielsweise kann ein „Delivery“ (vgl. Abbildung 24) aus einzelnen Funktionen bestehen, die bereits nach der Hälfte der Zeit an den Kunden ausgeliefert werden können.

Zusammenfassend wird festgestellt, dass aufgrund der Parallelen hinsichtlich Herausforderungen und Zielen (z.B. Kostenreduzierung, Verbesserung der Zusammenarbeit der Prozessbeteiligten,...), die Anwendung von agilen Ansätzen der Softwareentwicklung im Bereich des agilen Geschäftsprozessmanagements (vgl.

Abschnitt 2.4) durchgeführt werden kann. Die in dieser Arbeit beschriebene Methode adaptiert und erweitert Ansätze aus Scrum, Extreme Programming, Kanban und Crystal Clear (vgl. Abschnitt 4).

2.4. Agiles Geschäftsprozessmanagement

Die Balance zwischen Flexibilität und Stabilität ist nicht nur eine Grundlage der Definition des Agilitätsbegriffs (vgl. Abschnitt 2.3.1), sondern auch ein essentieller Faktor in Bezug auf das agile Geschäftsprozessmanagement. Zum einen unterstützen agile Methoden dabei, dass erfasste Geschäftsprozessmodelle sämtliche Alternativpfade enthalten, welche in der betrieblichen Anwendung nötig sind, um auch bei möglichen Ausnahmen (vgl. Abschnitt 4.2.3) die Stabilität innerhalb der Ausführung zu erhalten. Zum anderen müssen Geschäftsprozessmodelle, um deren Aktualität zu gewährleisten, kurzfristig und flexibel an Änderungen angepasst werden können. Hierbei stellt der Begriff der Agilität für agile Methoden keine formale Definition dar, sondern vermittelt eine bestimmte Grundhaltung [TN14]. Methoden, wie beispielsweise BPM(N)^{Easy} (vgl. Abschnitt 4), können nach Rumpe [Ru12] als agil bezeichnet werden, wenn diese insbesondere auf die Aspekte Effizienz, Reaktivität, Flexibilität, Einfachheit und Menscheneinbindung fokussiert. Das effiziente Ausführen des Gesamtprozesses und dessen einzelnen Aktivitäten ist zum Beispiel durch die Einsparung von Kosten durch das Weglassen von zu implementierenden Zusatzfunktionen erreichbar. Zudem müssen agile Methoden eine hohe Reaktivität und Flexibilität ermöglichen. Beispielsweise müssen Anforderungen kurzfristig in die Planung aufgenommen bzw. bereits bestehende Implementierungen adaptiert werden können. Die Flexibilität bezieht sich hierbei auch auf die Methode selbst, d.h. diese muss unter bestimmten Voraussetzungen (z.B. kleine Anzahl von Projektbeteiligten) flexibel anpassbar sein. Zusätzlich beschreibt Rumpe [Ru12, S.26] die Notwendigkeit der Einfachheit, sodass eine pragmatische Umsetzung der Methode möglich ist. Insbesondere der "Faktor Mensch" soll in agilen Methoden berücksichtigt werden: „den Fähigkeiten, Kenntnissen und Bedürfnissen der Projektbeteiligten wird im Projekt Rechnung getragen.“ [Ru12, S.26]. In [MJA15] wird agiles Vorgehen mit Hilfe der Einteilung in Ebenen beschrieben. Die grundlegende Ebene bilden „agile Charakteristiken“, wie beispielsweise „Flexibilität“. Die Ebene der „agilen Werte“ baut darauf auf und beschreibt zum Beispiel, dass die Reaktion auf eine Änderung höher gewertet wird, als die Einhaltung eines davor festgelegten Plans. Daran

anschließend definiert die Ebene der „agilen Prinzipien“ unter anderem, dass Änderungsanforderungen unabhängig vom Zeitpunkt (d.h. auch kurz vor der Fertigstellung) berücksichtigt werden. Die Ebene der „agilen Praktiken“ beschreibt konkret definierte Vorgehensweisen zur Durchführung von agilen (GPM-)Projekten. Zusammenfassend können für Methoden des agilen Geschäftsprozessmanagements insbesondere folgende Merkmale identifiziert werden.

- 1) Agiles Geschäftsprozessmanagement definiert einen iterativen und inkrementeller Ansatz um Anforderungen und Optimierungspotentiale während der Erfassung, Anreicherung und Ausführung von Geschäftsprozessen flexibel zu berücksichtigen.
- 2) Alle Prozessbeteiligten werden direkt in die unterschiedlichen Schritte des GPMs integriert. Beispielsweise werden Geschäftsprozesse gemeinsam von Anwendern und Modellierungsexperten erfasst und weiterentwickelt.
- 3) Agiles Geschäftsprozessmanagement ermöglicht es auf unvorhergesehene Änderungen angemessen zu reagieren. Werden beispielsweise Änderungen in real ausgeführten Geschäftsprozessen, aufgrund von veränderten Rahmenbedingungen, identifiziert, werden diese direkt in der nächsten Iteration berücksichtigt werden. Eine Iteration (vgl. Abschnitt 2.3.2) umfasst hierbei „nur“ einen kurzen Zeitraum (z.B. zwei bis vier Wochen).
- 4) Innerhalb des agilen Geschäftsprozessmanagement werden Qualität und Akzeptanz der Geschäftsprozesse stetig überprüft. Hierbei werden insbesondere die real ausführenden Anwender der Geschäftsprozesse eingebunden, um deren Feedback direkt an Modellierungs- und IT-Experten weitergeben zu können.

3. Gebrauchssprachliche Modellierung

Die in diesem Kapitel eingeführte gebrauchssprachliche Modellierung dient als Grundlage der signifikanten Verbesserung von Kommunikation, Interaktion sowie des gegenseitigen Verständnisses aller Beteiligten im Kontext des agilen Geschäftsprozessmanagements (vgl. [MOW14]). Zur Charakterisierung des Begriffs der gebrauchssprachlichen Modellierung folgend wird zuerst die Bedeutung von Sprache diskutiert. Des Weiteren werden die für diese Arbeit relevanten Sprachtypen (Gebrauchssprachen, Modellierungssprachen und Programmiersprachen) beschrieben. Dabei wird insbesondere der Zusammenhang von Sprache, Geschäftsprozessmodellen und implementierten Anwendungssystemen dargestellt. Anschließend wird der Begriff des Modells analysiert und die Eigenschaften von Modellen beschrieben. Zusätzlich werden für die Geschäftsprozessmodellierung relevante Modellierungssprachen (vgl. [Sc99][OM11b][OM11a][Wf12][OA07]) ausgewählt und präsentiert. Daraufhin wird die Sprechakttheorie nach Searle [Se69] diskutiert. Ein Sprechakt stellt die kleinste Einheit des Sprechens dar und bildet somit die Grundlage einer erfolgreichen gebrauchssprachlichen Modellierung. Abschließend wird der Begriff der gebrauchssprachlichen Modellierung abgegrenzt und definiert.

3.1. Sprache

Für den Begriff der Sprache bietet die Literatur zahlreiche Definitionen. Formal kann Sprache als „eine Menge von Zeichenreihen aus Σ^* , wobei Σ ein bestimmtes Alphabet darstellt (...)“ definiert werden [HMU11, S.56]. Ferdinand de Saussure [Sa01a, S.17f] beschreibt Sprache als „(...) ein System von Zeichen, in dem einzig die Verbindung von Sinn und Lautzeichen wesentlich ist (...)“. Für ihn ist die Sprache ein abgesondertes Objekt, welches sich vom Sprechen unterscheidet und separat erforscht werden kann. Mit Sprache kann folglich etwas einem anderen Menschen mitgeteilt werden (z.B. Gefühle, Gedanken), ohne notwendigerweise sprechen zu müssen (vgl. [Tr08, S.10ff]). Lorenzen & Kamlah [KL96] beschreiben Sprache als Mittel, die Welt rekonstruierend abzubilden oder konstruierend neu zu erschaffen [He06]. Die Sprache dient folglich zur Erschließung der Welt, beispielsweise durch die Identifizierung von Bekanntem oder durch das Einordnen von Unbekanntem [Go96]. In einer weiterführenden Definition nach Ortner wird als Sprache „(...)diejenige Menge von Terminalsymbol-Folgen (Sprachartefakten

oder Ausdrücken) bezeichnet, die von einem Startsymbol ausgehend, durch wiederholte Anwendung von syntaktischen Regeln (Grammatik), d.h. von Substitutionen, hergeleitet bzw. konstruiert werden können.(...)“ [Or05, S.27]. Ortner zählt zur Sprache generell folgende Bausteine: Gegenstandseinteilung, Grammatik, Wortschatz und Modi der Äußerung von Sprachartefakten. Die Gegenstandseinteilung¹⁰² und Grammatik¹⁰³ können als syntaktische Dimension einer Sprache aufgefasst werden. Unabhängig der Definition wird Sprache zur Kommunikation verwendet. Die Kommunikation kann hierbei durch verschiedene existierende Kommunikationstheorien –und modelle beschrieben werden (vgl. [Ke04, S.18ff][Do14, S.13ff]. Die Abbildung 26 stellt exemplarisch eines der grundlegenden Modelle dar, das Kommunikationsmodell nach Shannon [Sh48]. Die Kommunikation ist hierbei als unidirektionaler Prozess definiert, d.h. die Übertragung einer Nachricht erfolgt von Quelle zum Ziel ohne eine Rückkopplung.

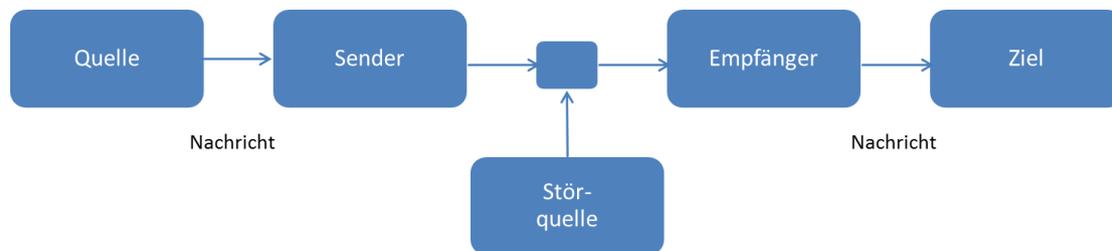


Abbildung 26: Schema eines Kommunikationssystems nach Shannon [Sh48]

Das Modell nach Shannon umfasst die folgenden Elemente zur Beschreibung einer Nachrichtenübertragung: die Quelle, welche die Nachricht erzeugt (z.B. durch eine Person ausgesprochener Satz in Gebrauchssprache). Den Sender, welcher die Nachricht entsprechend des Übertragungskanals kodiert. Muss die Nachricht der Quelle beispielsweise über Telefon übertragen werden, wird das Gesprochene zuerst in ein elektrisches Signal umgewandelt und (nach weiterer Verarbeitung) über einen Kanal (z.B. Telefonleitung) gesendet. Der Empfänger empfängt das Signal und dekodiert dieses, sodass die Nachricht dem Ziel (z.B. Zuhörer am Telefon) übermittelt werden kann. Das Modell von Shannon beschreibt des Weiteren mögliche Störquellen, welche die durch die Quelle erzeugte Nachricht beeinflussen kann.

Die übertragene Sprache kann nicht nur in einer Form auftreten. Je nach Hintergrund (z.B. kulturelle Tradition oder gemeinsame Lebenspraxis) wird eine Sprache geprägt

¹⁰² Vgl. Abschnitt 2.1.3.

¹⁰³ Zur weiteren Diskussion des Grammatikbegriffs sei auf [NP75, S.30ff][Sa01a, S.160ff][He12b, S.25ff] verwiesen.

[Go96]. Folglich gibt es nicht nur eine Sprache sondern beliebig viele Sprachen. Schienmann [Sc97] beschreibt zum Beispiel mehrere Sprachen, welche im Rahmen eines Fachentwurfs (z.B. zur Implementierung eines Anwendungssystems) genutzt werden. Dieser Ansatz vermeidet die Ausgrenzung von Beteiligten (z.B. Anwendern ohne technische Kenntnisse werden in die Anwendungssystementwicklung eingebunden).

Hierbei ist sicherzustellen, dass der Übergang von einem in den anderen Sprachtyp ohne Probleme (z.B. Missverständnisse zwischen den Kommunikationspartnern) möglich ist. Abbildung 27 zeigt die Wechselbeziehung der eingesetzten Sprachen.

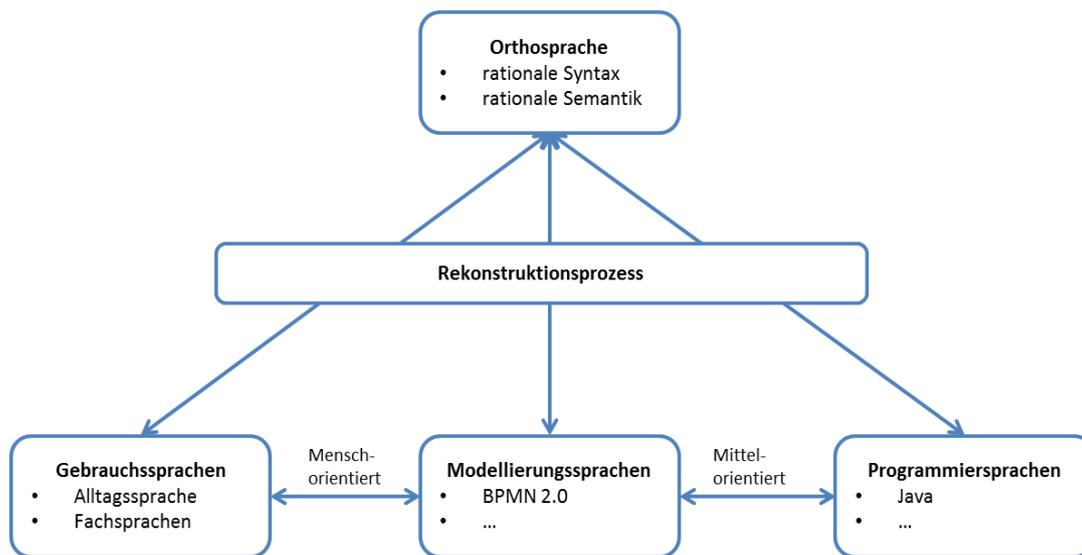


Abbildung 27: Übergang in andere Sprachen nach Schienmann [Sc97]

Die „Orthosprache“¹⁰⁴ dient als „Sprach-Repository“ auf oberster Ebene. Die „Orthosprache“ wurde von Lorenzen eingeführt (vgl. [Lo74]) und zielt auf die Etablierung einer allgemein anerkannten Wissenschaftssprache ab [RC13][Ma14]. Diese Wissenschaftssprache soll zur Verbesserung der Verständigung (z.B. im wissenschaftlichen oder politischen Bereich) dienen [Ga12c]. Die „Orthosprache“ ist als „(...) die um umgangssprachliche Defekte (z.B. Vagheiten) bereinigte Fachsprache eines Anwendungsbereichs aufzufassen, d.h. als eine konstruktiv aufgebaute Logiksprache, in der jedes Wort zirkelfrei und ausdrücklich in seiner Verwendung definiert ist (...).“ [Sc97, S.75][JKM74, S.43ff]. Die „Orthosprache“ enthält eine rationale Syntax und Semantik. Die Syntax umfasst ein Regelwerk, welches bestimmt, ob eine Zeichenfolge (z.B. eine Folge von Wörtern) der korrekten Ausdrucksform der Sprache unterliegen. Im Gegensatz

¹⁰⁴ Schienmann [Sc97] nutzt anstatt „Orthosprache“ den Begriff der „Normsprache“, um einem möglichen Verdacht zu entgegnen es gäbe nur „eine richtige Sprache“.

dazu beschreibt die Semantik die Bedeutung der Zeichenfolge (vgl. [Gr05][He12b]). Für die Sprachdefinition schließt an die Ebene der „Orthosprache“ ein Rekonstruktionsprozess an. Der Rekonstruktionsprozess beschreibt den Vorgang in dem alle Begriffe der Sprachen ermittelt, präzisiert und eindeutig festgelegt werden [Sa01b]. Begriffe, die nicht eindeutig zugeordnet und definiert werden können, erfordern eine Begriffsklärung¹⁰⁵ [El09]. Auf der untersten Ebene finden sich die Gebrauchssprachen, Modellierungssprachen und Programmiersprachen wieder. Diese dienen beispielsweise zur Beschreibung von Anforderungen an ein Anwendungssystem und den damit verbundenen Geschäftsprozessen. Diese Anforderungen sollen durch eine Sprache erfasst werden, welche die Überprüfung auf Vollständigkeit, Konsistenz und Eindeutigkeit zulässt [Al14b]. Zusätzlich sind Gebrauchssprachen, Modellierungssprachen und Programmiersprachen miteinander und mit den überliegenden Ebenen verbunden (vgl. gerichtete Kanten in Abbildung 27). Die Verbindungen zeigen die Möglichkeit der Transformation auf. Beispielsweise können gebrauchssprachliche Anforderungen in ein grafisches Modell überführt werden. Die Überführung bewirkt dabei eine Änderung des Formalisierungsgrads. Es kann zwischen präformal¹⁰⁶, semiformal und formal unterschieden werden [Wi05]. Vom Brocke [Vo03a, S.67] nennt zum Beispiel kreative Skizzen präformal, Ereignis-gesteuerte Prozesskettenmodelle¹⁰⁷ semiformal und Petri-Netze¹⁰⁸ formal. Im Folgenden werden die einzelnen Sprachtypen der untersten Ebene aus Abbildung 27 im Detail beschrieben:

- Gebrauchssprachen

Die Sprache, die im täglichen Leben verwendet wird, wird als Gebrauchssprache oder auch natürliche Sprache bezeichnet (vgl. [JKM74][HOS04][He06]). Mit Hilfe der Gebrauchssprache kommunizieren beispielsweise Anwender ihre Wünsche und Anforderungen an ein zu entwickelndes Anwendungssystem. Gebrauchssprachen verfügen dabei nicht über eine (durchweg) explizit verfasste Grammatik und bilden eher ein historisch gewachsenes und eingespieltes Gebilde [Re14].

¹⁰⁵ Vgl. Kapitel 2.1.3.

¹⁰⁶ „präformal“ wird synonym zu „informal“ oder „unformal“ verwendet.

¹⁰⁷ Vgl. Kapitel 3.2.2.

¹⁰⁸ Petri-Netze wurden nach Carl Adam Petri benannt. Mit Petri-Netzen können formale Modelle erstellt und z.B. zur Simulation oder Ausführung genutzt werden. Für weitere Ausführungen sei auf [Pe62] [Me06, S.84ff][Re10] verwiesen.

Die Gebrauchssprache kann noch einmal in weitere Bereiche unterteilt werden. Die Abbildung 28 stellt diese Einteilung dar.

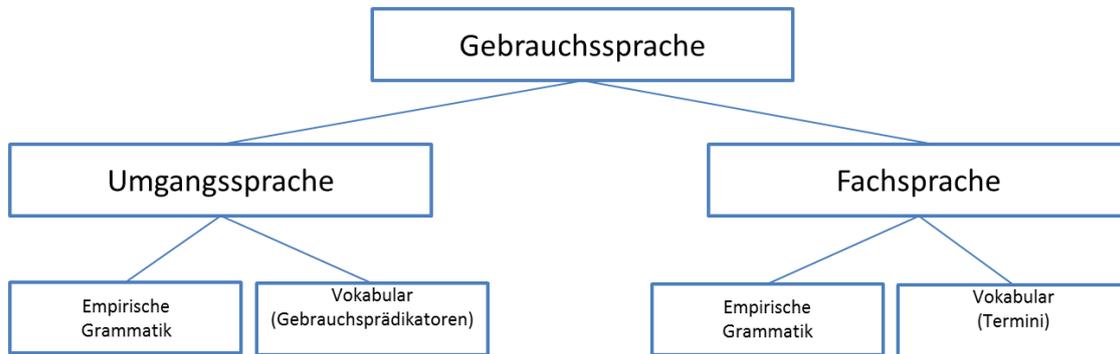


Abbildung 28: Gliederung der Gebrauchssprache nach Heinemann

Zur alltäglichen Verständigung wird die Umgangssprache genutzt (Abbildung 28, linke Seite). Hierbei ist das gegenseitige Verstehen im Gegensatz zur Fachsprache unproblematisch [LM11]. Die Fachsprache wird explizit eingeführt (z.B. durch IT-Experten zur Diskussion über die Implementierung einer neuen Software), sodass grundsätzliche Schwierigkeiten des Verständnisses entstehen können (z.B. beim Einsatz der IT-Experten Fachsprache in der Kommunikation mit einem „IT-Laien“). Beide Sprachen, also Umgangssprache und Fachsprache, bestehen aus ihrer empirischen, (im Laufe der Zeit) gewachsenen Grammatik und ihres Vokabulars. Das Vokabular der Umgangssprache besteht aus sogenannten Gebrauchsprädikatoren. Die Termini bilden das Vokabular der Fachsprache. Zum Beispiel nutzt ein Nicht-Biologe den Begriff „Eiweiß“, der Experte (Biologe) hingegen den Begriff „Protein“ [He06, S.39].

- Programmiersprachen

Zur Implementierung von Programmen werden Programmiersprachen eingesetzt [TW14]. Hierbei gibt es zahlreiche Programmiersprachen (vgl. [Sc09a]), welche sich beispielsweise nach Funktionsumfang oder Einsatzgebiet unterscheiden. Welche Programmiersprache eingesetzt wird, hängt von dem zu implementierenden Programm ab. Beispielsweise eignet sich die algorithmische (imperative) Programmiersprache C (vgl. [Ri93]) für Ressourcen-effiziente Operationen, während sich die Skriptsprache Python (vgl. [Py14]) hinsichtlich Textmanipulationen eignet [TW14]. Weitere Kriterien für die Nutzung von Programmiersprachen können unter anderem die Einfachheit (z.B. wie schnell ist die Programmiersprache erlernbar) oder der Standardisierungsgrad (z.B. wie weit ist die Programmiersprache verbreitet) sein. Weitere Kriterien dienen dazu, Programmiersprachen beispielsweise nach dem Programmier-, Implementierungs- oder Typen-Paradigma einzuteilen [Ba13]. Eine mögliche Klassifizierung von Programmiersprachen unternimmt Rajaraman [Ra98]. Die Abbildung 29 zeigt einen Überblick.

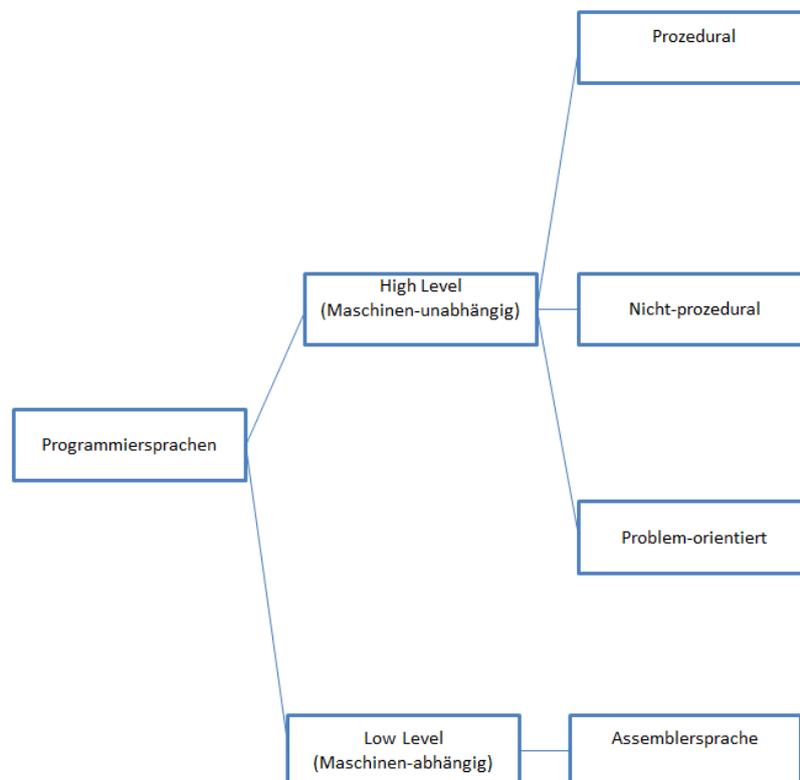


Abbildung 29: Klassifizierung von Programmiersprachen (nach [Ra98])

Wie in Abbildung 29 dargestellt, kann zwischen vier Kategorien von Programmiersprachen unterschieden werden.

Auf Ebene der „High-Level“-Sprachen, welche Maschinen-unabhängig genutzt werden können, teilt Rajaraman [Ra98, S.47ff] in prozedural, nicht-prozedural¹⁰⁹ und Problemorientiert auf. Programme, welche mit einer prozeduralen Programmiersprache (z.B. Java¹¹⁰) implementiert werden, bestehen aus einer Folge von Befehlen, die durch die Maschine nacheinander ausgeführt werden. Die prozeduralen Programmiersprachen „beantworten“ folglich die Frage, „wie“ ein Problem gelöst werden soll. Im Gegensatz dazu dienen die nicht-prozedurale Programmiersprachen zur Beschreibung, „was“ ein ausführbares Programm zurückgeben muss. Beispielsweise dient die nicht-prozedurale Programmiersprache SQL¹¹¹ zur Formulierung der erwarteten Ergebnisse bei einer Datenbankabfrage [AM09]. Problemorientierte Programmiersprachen dienen zur Lösung von Problemen in bestimmten Anwendungsbereichen. Die Befehle problemorientierter Programmiersprachen sind an den Anwendungsbereich angepasst und ermöglichen es, Experten aus bestimmten Fachbereichen, Probleme einfach zu formulieren und zu lösen. Beispielsweise wird MATLAB¹¹² zur Lösung von mathematischen Problemen eingesetzt. Assemblersprachen dienen zur direkten „Kommunikation“ mit der Maschine [Sc09a] und werden als „Low-Level“-Sprachen bezeichnet (vgl. Abbildung 29). Mit Hilfe eines Assemblers wird die Assemblersprache in Maschinensprache (bestehend aus einer Folge von 0 und 1 Bitwerten) übersetzt. Eine Assemblersprache ist daher nur für bestimmte Rechner einsetzbar, bietet aber (im Vergleich zur Maschinensprache) besser verständliche Befehle [AM09]. Zum Beispiel wird der Befehl „SUB“ zur Subtraktion zweier Werte genutzt.

- Modellierungssprachen

Modellierungssprachen dienen als lösungsorientiertes Mittel, die Wünsche und Anforderungen des Anwenders gegenüber den Experten (z.B. BPM -oder IT-Experten) durch Modelle verständlich zu machen [MOW14].

¹⁰⁹ Synonym zu den Begriffen „prozedural“ und „nicht-prozedural“ können die Begriffe „imperativ“ bzw. „deklarativ“ genutzt werden, vgl. [Ra98][Sc09a].

¹¹⁰ <https://www.java.com/de/>.

¹¹¹ SQL ist die Abkürzung für „Standard Query Language“ und bezeichnet eine Programmiersprache, welche zum Zugriff auf relationale Datenbank dient (vgl. [Wi09][Sc13a]).

¹¹² <http://de.mathworks.com>.

Beispielsweise werden Modellierungssprachen dazu genutzt, Beziehungen zwischen Aktivitäten, Informationen und Menschen bzw. Maschinen darzustellen [Lü11]. Der Formalisierungsgrad unterschiedlicher Modellierungssprachen variiert hierbei. Beispielsweise können Elemente einer graphischen Modellierungssprache (z.B. Knoten, Pfeile, ...) durch zusätzliche natürlichsprachliche Beschreibungen erweitert werden, sodass eine semiformale Modellierungssprache vorliegt. Die Ergänzung von formalen Ausdrücken [RLR11] unterstützt den formalen Charakter einer Modellierungssprache und kann zum Beispiel zur Definition von automatisiert prüfbar Regeln genutzt werden. Wie viele unterschiedliche Modellierungssprachen existieren, ist aufgrund der hohen Anzahl nicht eindeutig bestimmbar. Für die Modellierung von Geschäftsprozessen können beispielsweise Ereignisgesteuerte Prozessketten (EPK), Business Process Modeling and Notation (BPMN), Unified Modeling Language (UML) Aktivitätsdiagramme, die XML Process Definition Language (XPDL) oder die WS-Business Process Execution Language (BPEL) genannt werden (vgl. [Sc99][OM11b][OM11a][Wf12] [OA07]).

Aufgrund der Bedeutung von Modellierungssprachen für diese Arbeit wird der Abschnitt 3.2 zur ausführlichen Beschreibung von Modellen und Modellierungssprachen genutzt.

3.2. Modell und Modellierungssprachen

Dieser Abschnitt beschreibt im ersten Teil den Begriff des Modells und dessen Eigenschaften. Im zweiten Teil werden ausgewählte Modellierungssprachen beschrieben und bewertet.

3.2.1. Modell

Der Begriff des Modells wird in zahlreichen Zusammenhängen und Disziplinen (z.B. der Mathematik, Informatik, etc.) definiert und undefiniert [Ja02][Or05] angewendet. Beispielsweise können Modelle zur komprimierten Abbildung von Wissen genutzt werden. Modelle dienen dann zur Weitergabe und Aneignung des zu vermittelnden Wissens oder zur Kommunikationsunterstützung (z.B. in Zusammenhang mit der Ausführung von Geschäftsprozessen) [Pr10][Kr12b]. Nach Scheer [Sc90, S.157] bezeichnet ein „(...) Modell die Abbildung eines realen Systems (...)“. Stachowiak [St73, S.131ff] definiert drei Hauptmerkmale des Modellbegriffs: das Abbildungsmerkmal (Modelle beziehen sich auf etwas (Originale), das Verkürzungsmerkmal (Modelle bilden nicht alle Attribute des Originals ab) und das pragmatische Merkmal (Modelle sind an Zeit, Zweck und „Empfänger“ gebunden). Rosemann [Ro96, S.17] definiert als Modell „(...) abstrahierende, immaterielle Abbilder eines Ausschnitts der realen Strukturen bzw. des realen Verhaltens für Zwecke eines Subjekts (...)“. Zusammenfassend beschreibt Prilla [Pr10, S.79f] folgende Eigenschaften:

- Modelle dienen zur Abbildung von Sachverhalten aus der Realität (z.B. ein Schienenverkehrsnetzplan zur Darstellung des realen Schienennetzes des öffentlichen Verkehrs in München).
- Modelle enthalten nicht alle Details des Gegenstandsbereichs.
- Modelle dienen zur Reduktion der Komplexität. Das bedeutet Informationen werden systematisch und verständlich dargestellt.
- Der Abstraktionsgrad eines Modells wird von dessen Zweck beeinflusst.
- Modelle können sowohl den IST-Zustand als auch einen SOLL-Zustand abbilden.
- Modelle besitzen eine Subjektivität, welche durch den Einfluss der beteiligten Modellierer entsteht.

- Modelle erfüllen eine konstruierende Wirkung, durch die das Verhalten und die Anordnung von Sachverhalten beeinflusst werden kann.

Wie Modelle erfasst werden, hängt von den daran gestellten Anforderungen ab. Beispielsweise können Modelle textuell oder grafisch (z.B. als WS-BPEL– bzw. BPMN-Modell)¹¹³ erfasst werden. Modelle, welche Geschäftsprozesse abbilden, werden meistens mit Hilfe von grafischen Modellierungssprachen erfasst [Wö10].

Die Abbildung 30 stellt ein einfaches Modell eines Geschäftsprozessmodells dar.

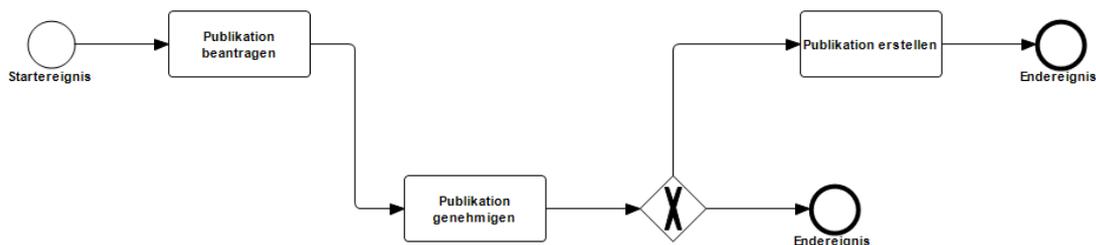


Abbildung 30: Einfaches Modell eines Geschäftsprozesses

Das Modell in Abbildung 30 besteht aus fünf unterschiedlichen grafischen Elementen¹¹⁴: Das Startelement (Kreis mit dünnem Rand) symbolisiert den Start des Geschäftsprozesses. Die rechteckigen Elemente beschreiben die verschiedenen Aktivitäten innerhalb des Geschäftsprozesses (z.B. „Publikation beantragen“ oder „Publikation genehmigen“). Die X-Symbol-beinhaltende Raute stellt eine „Entweder-Oder“-Entscheidung dar. Der Geschäftsprozess kann entweder durch die vorherige Nicht-Genehmigung der Publikation direkt beendet werden oder (nach einer erfolgreichen Genehmigung) die Publikation wird erstellt bevor der Geschäftsprozess endet. In beiden Fällen symbolisiert das Endelement (Kreis mit dickem Rand) das Ende.

Zusätzlich zu grafischen (zwei-dimensionalen) Elementen ist es möglich, multimediale Elemente (z.B. Audio- oder Videosequenzen) als nicht formalisiertes Beschreibungsmittel zu verwenden [Sc97][KMW14]. Des Weiteren wird bei dem Begriff des Modells (wie auch bei Programmiersprachen¹¹⁵) zwischen imperativen und deklarativen Modellen unterschieden [HZ14]. Bei der deklarativen Modellierung wird

¹¹³ Vgl. Abschnitte 3.2.2.

¹¹⁴ Im Beispiel werden Elemente der BPMN Modellierungssprache verwendet, vgl. Abschnitt 3.2.2.

¹¹⁵ Vgl. Abschnitt 3.1.

insbesondere auf die Identifizierung von „Spielregeln“ fokussiert [Ig12][Ha14]. Die Beschreibung eines deklarativen Modells umfasst den Startpunkt, das Ende und die Regeln, welche bei der Ausführung von Aktivitäten beachtet werden müssen. Deklarative Modelle beschreiben somit das „WAS“, also zum Beispiel dass eine Publikation genehmigt oder abgelehnt werden muss. Imperative Modelle definieren dagegen das „WIE“, also z.B. welche Sequenz von Aktivitäten durchgeführt werden muss, um eine Publikation zu genehmigen oder abzulehnen (vgl. Abbildung 30). Für eine weitere Diskussion des Modellbegriffs sei auf [St73][Ro99][Th05] verwiesen.

3.2.2. Modellierungssprachen

Modelle dienen nicht nur zur reinen Abbildung der Realität, sondern auch zur Erfassung und Kommunikation. Beispielsweise werden Geschäftsprozessmodelle zur Schulung und Einarbeitung von neuen Mitarbeitern verwendet. Damit die Kommunikation erfolgreich ist, muss ex ante definiert werden, wie ein Modell aufgebaut werden soll (Syntax) und was deren Bedeutung ist (Semantik)¹¹⁶. Die Menge dieser Definitionen (z.B. das grafische Symbol eines Rechtecks symbolisiert eine Geschäftsprozessaktivität) wird als Modellierungssprache bezeichnet [Wö10].

Es existiert eine Vielzahl unterschiedlicher Modellierungssprachen, welche abhängig von vordefinierten Zielen (z.B. Dokumentation und Simulation von Geschäftsprozessen) angewendet werden können. Modellierungssprachen können hierbei im Allgemeinen in zwei Typen eingeteilt werden: textuelle und grafische Modellierungssprachen (vgl. [GU99][HR00][Mo09]). Textuelle Modellierungssprachen bestehen aus beispielsweise gebrauchssprachlichen Sätzen oder (semi-)formalen Beschreibungen. Grafische Modellierungssprachen bestehen aus grafischen Elementen (z.B. Rechteck, Punkt,...). Die Ziele und die Umgebung, in der die Modellierungssprache eingesetzt werden soll, bestimmen hierbei den verwendeten Typ [SH14]. Unterschiedliche Perspektiven (z.B. zur Darstellung von Organisationseinheiten oder deren Verwendung) können die Auswahl der Modellierungssprache beeinflussen (vgl. [HZ14]).

¹¹⁶ Vgl. u.a. [Gr10, S.3f]: „Semantik leistet einen wichtigen Beitrag für unmissverständliche Kommunikation zwischen den Beteiligten und kann einen hohen Grad an Automatisierung über interoperable Werkzeuge ermöglichen.“

Im Kontext des Geschäftsprozessmanagements werden insbesondere grafische Modellierungssprachen verwendet (vgl. [FL03][Wö10]). Das Sprichwort „ein Bild sagt mehr als tausend Worte“ beschreibt nach Hitchman den Grund hierfür [Hi95]. Insbesondere tragen grafische Modelle zum besseren Verständnis und Kommunikation der Modelle bei (vgl. [WK06][Aa14]).

Im folgenden Abschnitt werden die in der Anwendung weitverbreiteten grafischen Modellierungssprachen beschrieben und diskutiert. Als erstes wird die Ereignisgesteuerte Prozesskette (EPK) und deren Erweiterung, die sogenannte erweiterte Ereignisgesteuerte Prozesskette (eEPK) beschrieben. EPK ist insbesondere im deutschsprachigen Raum weit verbreitet. Als zweites wird die Business Process Model and Notation (BPMN) diskutiert. BPMN ist seit dem Jahr 2013 ein offizieller Standard der International Organization for Standardization (ISO) und wird weltweit zur Modellierung von Geschäftsprozessen eingesetzt. Die Unified Modeling Language (UML) gilt als die vorherrschende grafische Modellierungssprache zur Modellierung von Software-Systemen. Das UML Aktivitätsdiagramm kann hierbei zur Modellierung von (Geschäftsprozess-)Aktivitäten in oder mit Softwaresystemen verwendet werden. Abschließend werden die WS-Business Process Execution Language (WS-BPEL) und die XML Process Definition Language (XPDL) beschrieben, welche insbesondere hinsichtlich der Geschäftsprozessautomatisierung eine wichtige Rolle in der Modellierung einnehmen.

- Ereignisgesteuerte Prozessketten

Die Ereignisgesteuerten Prozessketten wurden bereits 1992 entwickelt [KNS92]. Ziel der Entwicklung war es, Geschäftsprozesse einfach und realitätsnah modellieren zu können [Sc02]. EPK-Modelle sind gerichtete, zusammenhängende Graphen mit einer Multiplizität von Eins [KKS04]. Den Mittelpunkt der EPK bilden Funktionen, die in deren zeitlich-logischer Abfolge modelliert werden. Zusätzlich, um eine Kontrollflusssteuerung berücksichtigen zu können, umfasst die EPK sogenannte Verknüpfungsoperatoren. Insgesamt gehören Funktionen, Ereignisse, Verknüpfungsoperatoren zum Elementset [KKS04]. Die Abbildung 31 stellt ein einfaches EPK-Modell, welches den Geschäftsprozess einer Kundenanfrage beschreibt, dar.

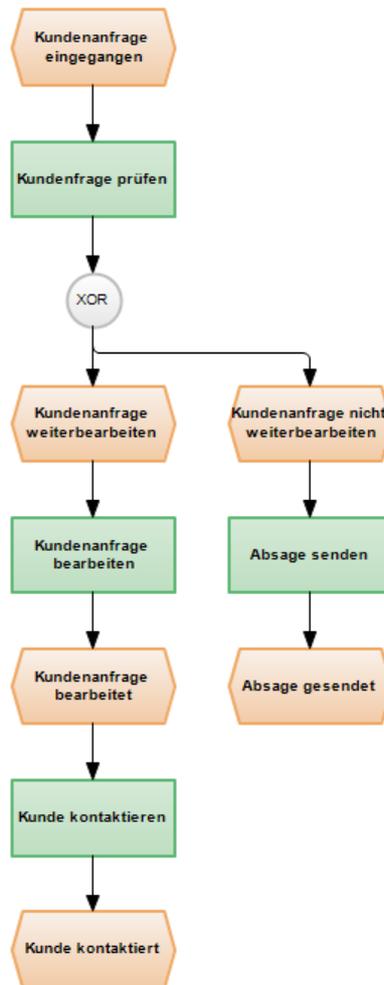


Abbildung 31: Beispiel EPK-Modell - Angebotsprozess

Ereignisse sind passive Elemente, die den Eintritt von neuen Systemzuständen darstellen bzw. betriebswirtschaftliche Bedingungen symbolisieren (z.B. „Kundenanfrage eingegangen“). Mit dem Element „Funktion“ werden Objekte von einem Startzustand in einen Endzustand transformiert (z.B. „Kundenanfrage prüfen“).

Die Verknüpfungsoperatoren werden in drei Typen unterteilt: AND, OR und XOR. Das konjunktive Element AND symbolisiert die Splittung des Geschäftsprozesses in (mindestens) zwei parallele Pfade. OR und XOR symbolisieren ein „Entweder-Oder“, wobei das OR für eine inklusive und das XOR für eine exklusive Modellierung von Pfaden verwendet wird¹¹⁷. Beispielsweise teilt sich der Sequenzfluss nach der Funktion

¹¹⁷ Vgl. [KKS04, S.4]: „Die Verwendung dieser Konstrukte ist hierbei reglementiert, indem ein OR bzw. XOR nach einem Ereignis nicht zugelassen wird, da ein Ereignis keine formal eindeutigen Handlungsanweisungen aussprechen kann.“

„Kundenanfrage prüfen“ durch ein XOR, entweder in eine „Weiterbearbeitung“ oder „nicht Weiterbearbeitung“ der Kundenanfrage (vgl. Abbildung 31).

Eine Weiterentwicklung der EPK ist die erweiterte ereignisgesteuerte Prozesskette¹¹⁸. Hierbei wurden weitere grafische Elemente zur Modellierungssprache hinzugefügt (z.B. Organisationseinheiten oder Anwendungssysteme). Die Abbildung 32 illustriert den Kundenanfrageprozess in Form einer eEPK.

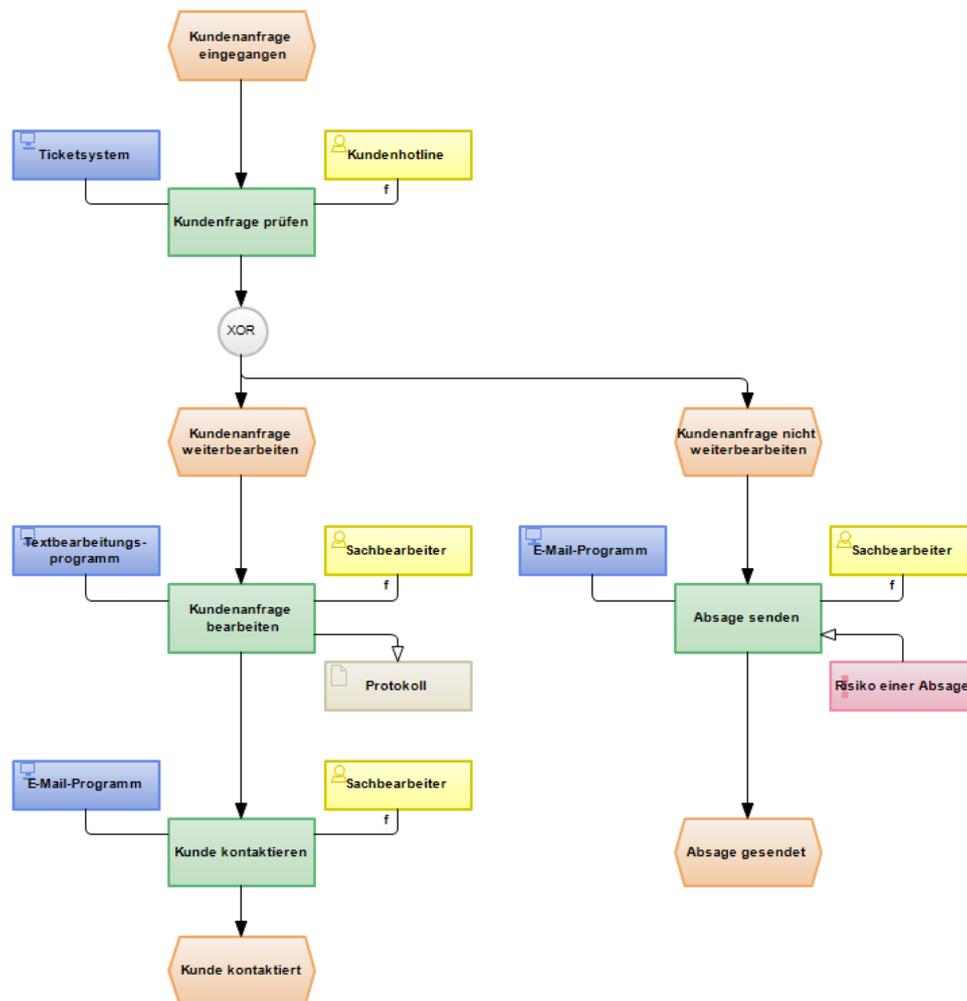


Abbildung 32: Beispiel eEPK-Modell – Angebotsprozess

Die Abbildung 32 beinhaltet eine Auswahl der zusätzlichen Elemente. Zum Beispiel können an Funktionen weitere Informationen hinzugefügt werden. Die Funktion „Kundenanfrage bearbeiten“ wird von der Organisationseinheit „Sachbearbeiter“ ausgeführt. Zur Bearbeitung der Kundenanfrage kann hierbei ein Anwendungssystem

¹¹⁸ Die EPK wurde nach Aufnahme in kommerzielle Modellierungswerkzeuge um weitere - pragmatische - Elemente erweitert, vgl. [KKS04].

genutzt werden („Textverarbeitungsprogramm“) und als expliziter Output wird ein „Protokoll“ ausgegeben. Um mögliche Fehler und deren Risiko visualisieren zu können, beinhaltet die eEPK ein Element zur Beschreibung des Prozessrisikos. Nach [KKS04, S.7] lassen sich hierdurch „(...) potenzielle Hindernisse grafisch hervorheben und durch präventive Maßnahmen auf ein Minimum reduzieren. Dies führt schließlich zu einer Erhöhung der Durchführungssicherheit und damit der Prozessqualität“.

Neben der Möglichkeit der detaillierten Modellierung von Geschäftsprozessen kann als Nachteil der (e)EPK der hohe Einführungsaufwand genannt werden. Durch die Vielzahl der Elemente und Regeln ist die Verständlichkeit für Nicht-Experten nicht ohne Probleme möglich. Ein weiterer Nachteil ist, dass die Modellierungssprache auf die reine Dokumentation von Geschäftsprozessen fokussiert und somit eine direkte Automatisierung aufgrund der fehlenden Formalisierung nicht möglich ist¹¹⁹. Zusätzlich kann die Korrektheit eines (e)EPK-Modells nicht automatisiert durchgängig überprüft werden. Grund hierfür ist wiederum die fehlende Formalisierung der Modellierungssprache [Ba10b].

- Business Process Model and Notation

Die grafische Modellierungssprache Business Process Model and Notation wurde erstmals 2004 als Business Process Modeling Notation durch die Object Management Group (OMG)¹²⁰ als Entwurf veröffentlicht und im Jahre 2006 als finale Version 1.0 veröffentlicht [OM04][OM06]. Das Ziel von BPMN ist die Unterstützung des Geschäftsprozessmanagements unter Berücksichtigung der Anforderungen von sowohl fachlichen, als auch technischen Anwendern. Zum Beispiel sollen die Modellierungselemente der BPMN für alle Anwender verständlich und einsetzbar sein. Seit der Version 2.0 [OM11a] beschreibt BPMN insgesamt vier Modelltypen zur Darstellung von Kollaboration, Geschäftsprozesse, Konversationen und Choreographien. Zudem erlaubt der BPMN 2.0-Standard die formale Speicherung der Geschäftsprozessmodelle, sodass diese als Grundlage einer Automatisierung genutzt werden können. Im Folgenden werden die BPMN Basiselemente erläutert.

¹¹⁹ Der Nachteil der Nicht-Formalisierung kann ausgeglichen werden, indem EPK-Modelle in andere, formalisierbare Modellierungssprachen transformiert werden. Beispielsweise ist eine Transformation in BPMN möglich. Für eine ausführliche Beschreibung der Transformation von EPK nach BPMN sei auf [DTP09][FRH10][BG14] verwiesen.

¹²⁰ <http://omg.org>

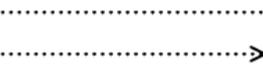
Zusätzlich wird das für diese Arbeit relevante BPMN Prozessdiagramm beschrieben. Abschließend werden die weiteren BPMN Modelltypen (Kollaborationsdiagramm, Choreographiediagramm, Konversationsdiagramm) skizziert und deren Verwendungszweck diskutiert.

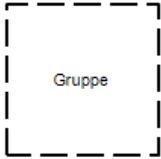
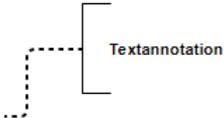
BPMN Basiselemente

Mit BPMN 2.0 sind über 100 Elemente beschrieben [Ko11a]. Mit Hilfe dieser Vielzahl von Elementen können, sowohl einfach Geschäftsprozessmodelle, als auch komplexe Choreographie – und Konversationsdiagramme modelliert werden. In Bezug auf die in dieser Arbeit vorgestellte Modellierungssprache BPMN^{Easy} sind ausschließlich die Basiselemente (vgl. [OM11a, S.25ff]) von Relevanz. Diese Elemente werden in Tabelle 14 beschrieben.

Tabelle 14: Auflistung der BPMN 2.0 Basiselemente

Element	Beschreibung	Symbol
Ereignis (Event)	Mit einem Ereignis wird beschrieben, dass etwas „eingetreten ist“. Ereignisse werden als Kreis dargestellt, wobei die Stärke des Kreisrands den Typ des Ereignisses widerspiegelt: dünner Rand = Startereignis, doppelter Rand = Zwischenereignis, dicker Rand = Endereignis.	
Aktivität	Durch eine Aktivität wird dargestellt, dass Arbeit verrichtet werden muss. Hierbei können Aktivitäten entweder atomar oder nicht-atomar sein. Atomar bedeutet, dass die Aktivität nicht weiter unterteilt werden kann (Task). Nicht-atomare Aktivitäten können als Subprozesse weiter verfeinert werden.	
Gateway	Gateways splitten Pfad(e) eines Geschäftsprozessmodells auf oder führen diese zusammen. Durch verschiedene	

	<p>Marker innerhalb des Symbols wird visualisiert, welches Gateway dargestellt werden soll. Ein XOR wird mit Hilfe eines X, ein AND durch ein + modelliert.</p>	
Sequenzfluss	<p>Ein Sequenzfluss ist eine gerichtete Kante, welche die Abfolge der Aktivitäten in einem Geschäftsprozess festlegt.</p>	
Nachrichtenfluss	<p>Nachrichtenflüsse stellen den Austausch von Nachrichten zwischen zwei oder mehreren Teilnehmern eines Geschäftsprozesses dar.</p>	
Assoziation	<p>Mit einer Assoziation können Artefakte (z.B. Datenobjekte, Kommentare,...) an andere BPMN Elemente annotiert werden.</p>	
Pool	<p>Ein Pool symbolisiert einen Teilnehmer eines Geschäftsprozesses. Besteht ein Geschäftsprozess aus mehreren Teilnehmern dienen Pools als Abgrenzung der Teilnehmer gegeneinander innerhalb einer Kollaboration. Ein Pool kann als Lane verwendet werden (vgl. Zeile unterhalb).</p>	
Lane	<p>Eine Lane (auch Swimline genannt) wird zur Strukturierung der BPMN Elemente genutzt. Beispielsweise können Lanes alle Aktivitäten einer Rolle enthalten. Eine Lane muss nicht zwingend in einem Pool liegen.</p>	
Data Object	<p>Das Data Object beschreibt, welche Daten für eine Ausführung einer Aktivität benötigt und welche Daten ggfs. als Ergebnis der Ausführung erstellt werden.</p>	

<p>Nachricht (Message)</p>	<p>Eine Nachricht, dargestellt durch einen Briefumschlag, symbolisiert einen Inhalt einer Konversation zwischen zwei Teilnehmern einer Kommunikation. Ist das Symbol „leer“ handelt es sich um eine ausgehende Nachricht, ist das Symbol „ausgefüllt“ um eine eingehende Nachricht.</p>	
<p>Gruppe (Group)</p>	<p>Eine Gruppe besitzt keine Auswirkung auf z.B. einen Sequenzfluss. Eine Gruppe dient ausschließlich zur grafischen Strukturierung eines BPMN Modells, beispielsweise zur Zusammenfassung von „gleichartigen“ Aktivitäten.</p>	
<p>Textannotat ion</p>	<p>Das Hinzufügen von Kommentaren (z.B. zusätzliche Informationen zu einer Aktivität) können über eine Textannotation an andere BPMN Elemente angeheftet werden. Die Textannotation dient ausschließlich zur Dokumentation.</p>	

BPMN Prozessdiagramm

In BPMN wird zwischen privaten und öffentlichen Geschäftsprozessen unterschieden. Als private Geschäftsprozesse, werden interne, organisationsspezifische Geschäftsprozesse bezeichnet. Private Geschäftsprozesse werden des Weiteren in ausführbar und nicht ausführbar unterteilt. Hierbei ist der Verwendungszweck entscheidend (d.h. wird das Modell mit dem Ziel der Automatisierung erfasst?) und die Einhaltung der Ausführungssemantik. Die Modellierung von Web Services-Aufrufen wird in BPMN „Orchestrierung“ genannt.

Hierbei verwendet die BPMN-Spezifikation die Begriffe „Prozess“ und „Orchestrierung“ synonym¹²¹. Ein Geschäftsprozessmodell besteht in BPMN insbesondere aus Aktivitäten, Ereignissen, Gateways und Sequenzflüssen.

Die Abbildung 33 stellt ein einfaches Beispiel einer Bezahlprüfung mit anschließendem Warenversand dar. Der private Geschäftsprozess findet innerhalb einer Organisation statt, wodurch alle verwendeten Lanes innerhalb eines Pools modelliert sind¹²².

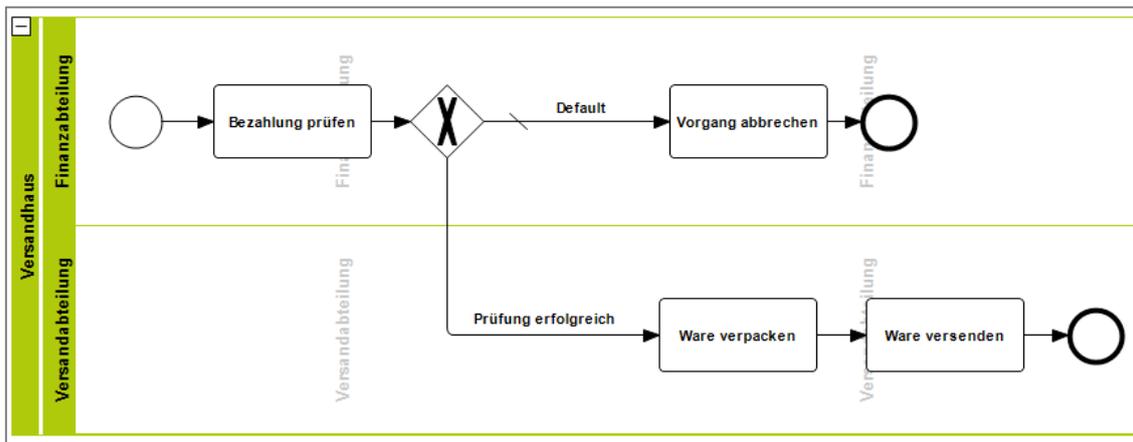


Abbildung 33: Beispiel - BPMN Prozessdiagramm

Der Geschäftsprozess in Abbildung 33 startet mit dem Eingang einer Bezahlung. Die Finanzabteilung überprüft die Richtigkeit der Zahlung. Mit Hilfe des anschließenden XOR Gateways führt der Sequenzfluss entweder zum Vorgangsabbruch oder zur Weiterleitung an die Versandabteilung. Hierbei müssen die ausgehenden Kanten des Gateways markiert sein, wobei eine der beiden als Default-Sequenzfluss gekennzeichnet werden kann (d.h. falls keine andere Bedingung wahr ist, wird dieser Pfad ausgewählt). Die Versandabteilung verpackt und versendet die Ware. Danach ist der Geschäftsprozess beendet. Müssen organisationsübergreifende Geschäftsprozesse modelliert werden, werden Nachrichtenflüsse modelliert, welche im Gegensatz zu einem BPMN Prozessdiagramm in einem Kollaborationsdiagramm verwendet werden dürfen.

¹²¹ Vgl. [OM11a, S.147]: „(...) These Processes have been generally called workflow or BPM Processes (...). Another synonym typically used in the Web services area is the Orchestration of services.“

¹²² Für ein BPMN Prozessdiagramm sind Pools und Lanes keine notwendige Bedingung. Dennoch werden diese z.B. aus Gründen der besseren Strukturierung und die damit einhergehende verbesserte Lesbarkeit genutzt.

BPMN Kollaborationsdiagramm

Das BPMN Kollaborationsdiagramm erlaubt die Modellierung von Kollaborationen mehrerer Teilnehmer eines Geschäftsprozesses. Die unterschiedlichen Teilnehmer werden mit Hilfe von Pools abgegrenzt und die Verbindung in Form eines Nachrichtenaustauschs wird über Nachrichtenflüsse visualisiert. Der in Abbildung 33 dargestellte Geschäftsprozess wird in Abbildung 34 durch einen weiteren Pool ergänzt. Nachdem der Geschäftsprozess durch die Finanzabteilung gestartet wurde, wird die Bezahlung geprüft. Diese Prüfung findet in Kollaboration mit dem zweiten Teilnehmer, der Bank, statt. Die Bank erhält hierbei eine Nachricht, dass eine Kontoprüfung durchgeführt werden muss. Wurde diese ausgeführt sendet die Bank eine Nachricht an die Finanzabteilung zurück, sodass der Geschäftsprozess innerhalb des Versandhauses weitergeführt werden kann.

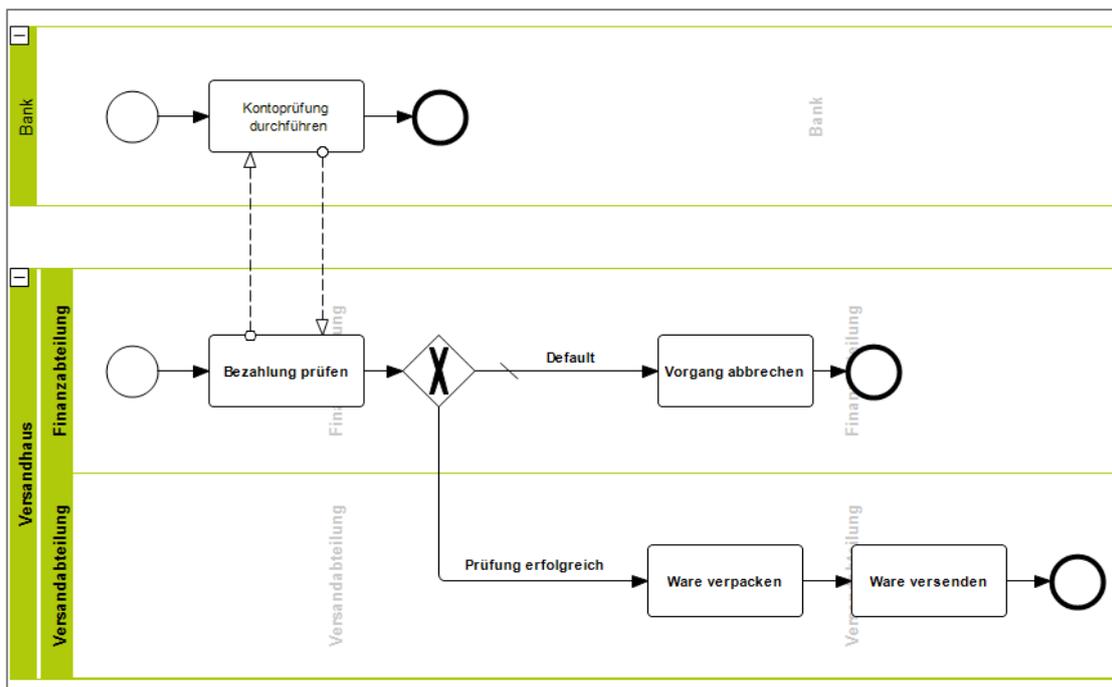


Abbildung 34: Beispiel - BPMN Kollaborationsdiagramm

Neben dem in Abbildung 34 dargestellten Modell bietet BPMN weitere verschiedene Möglichkeiten der Visualisierung von Kollaborationsdiagrammen. Beispielsweise können sogenannte Black Boxen genutzt werden, um einen Pool „zugeklappt“ darzustellen.

Insbesondere wenn die internen Geschäftsprozessaktivitäten von anderen Teilnehmern nicht bekannt sind, kann eine solche Darstellung gewählt werden. Die Nachrichtenflüsse werden dann an die Kante des Pools angeheftet (vgl. [OM11a, S.110ff]).

BPMN Choreographiediagramm

Das BPMN Choreographiediagramm ermöglicht das Modellieren von Kollaborationen unabhängig von der Geschäftsperspektive. Hierbei steht die Interaktion der Teilnehmer und weniger der Fluss der Geschäftsprozesse im Vordergrund. Zum Beispiel kann eine vertragliche Vereinbarung visualisiert werden¹²³. Eine Choreographie beschreibt den Nachrichtenaustausch zwischen zwei oder mehreren Kollaborationsteilnehmern. Des Weiteren können ggfs. die Abhängigkeiten der Kollaborationsteilnehmer zueinander dargestellt werden (vgl. [Ka12, S.38]). Hierbei werden Choreographien (im Gegensatz zu Kollaborationsdiagrammen) ohne Pools modelliert. Die Abbildung 35 stellt das bereits beschriebene Beispiel einer Kontenprüfung als Choreographiemodell dar.

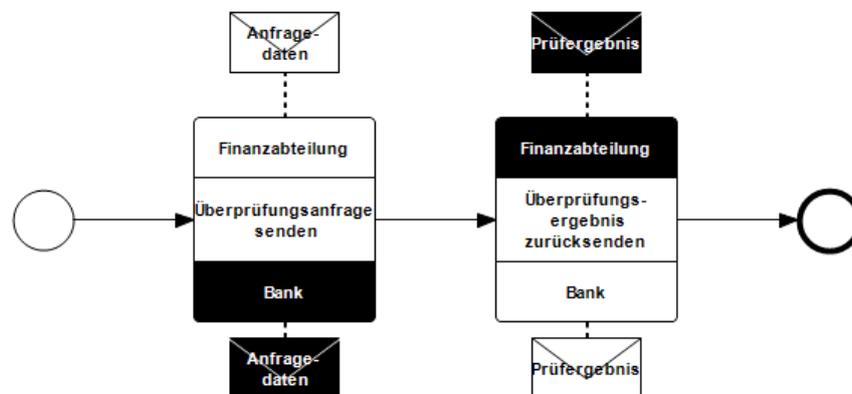


Abbildung 35: Beispiel - Choreographiediagramm

Zentraler Bestandteil des Modells sind die Choreographieaktivitäten. An eine „normale“ BPMN-Aktivität werden die einzelnen Kollaborationsteilnehmer angeheftet (vgl. Abbildung 35). Zusätzlich werden nicht-ausgefüllte Elemente (Briefumschläge) zur Markierung der Sendernachricht und schwarz ausgefüllte Elemente zur Markierung der Empfängernachricht annotiert. Beispielsweise sendet die Finanzabteilung Anfragedaten (z.B. Kontonummer) an die Bank. Der Sequenzfluss zwischen „Überprüfungsanfrage

¹²³ Vgl. [OM11a, S.315]: „A Choreography is a definition of expected behavior, basically a procedural business contract, between interacting participants.“

senden“ und „Überprüfungsergebnis zurücksenden“ stellt die Abhängigkeit der Choreographieaktivitäten dar. Komplexere Choreographien können mit Hilfe von Teil-Choreographien hierarchisiert werden.

BPMN Konversationsdiagramm

Zur Vereinfachung der Darstellung von Kollaborationen kann das BPMN Konversationsdiagramm verwendet werden. Die grob granular modellierten Konversationen können bei Bedarf mit Hilfe von Choreographie- oder Kollaborationsdiagrammen ergänzt werden. Mit dem „Conversation Link“ und dem „Conversation Node“ enthält das Konversationsdiagramm zwei grafische Elemente, die nur in diesem Modelltyp von BPMN verwendet werden dürfen. Mit Hilfe eines „Conversation Link“ (doppelte durchgezogene Kante) wird eine einfache, nicht in weitere Konversationen aufteilbare Konversation dargestellt. Der „Conversation Node“ (Sechseck-Symbol) dient als Verbindungselement verschiedener Nachrichtenflüsse einer Konversation. Die Abbildung 36 stellt zwei Möglichkeiten der Visualisierung dar.

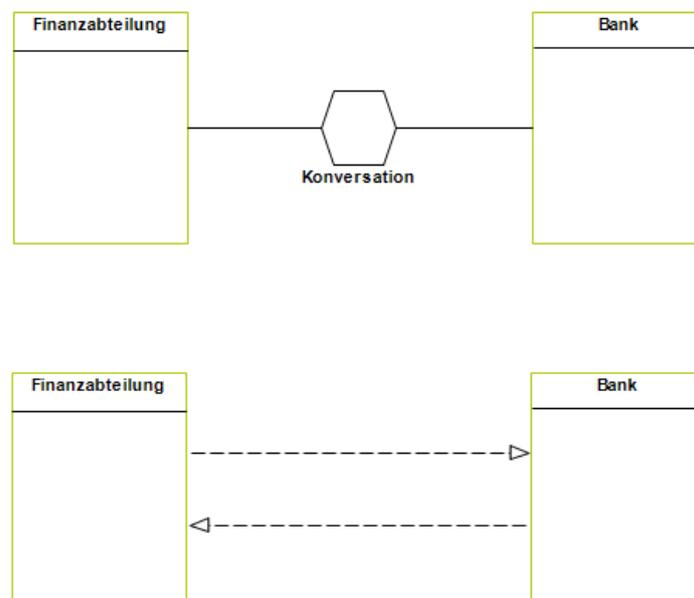


Abbildung 36: Beispiel - BPMN Konversationsdiagramm

Konversationen können sowohl mit Hilfe des „Conversation Nodes“ modelliert werden, als auch mit BPMN Nachrichtenflüssen. Beide Varianten der Abbildung 36 stellen dar, dass die Finanzabteilung mit der Bank eine Konversation führt.

Für weitere detaillierte Beschreibung der BPMN Modellierungssprache sei auf [FRH10][OM11a][A113] verwiesen. BPMN in der Version verfügt über eine hohe Anzahl unterschiedlicher Modellierungselemente, wodurch das allgemeine Verständnis der Modellierungssprache erschwert wird (vgl. [AK15]). Des Weiteren lässt die Spezifikation Interpretationsspielraum, sodass insbesondere bei der Umsetzung von GPM-Werkzeugen es zu unterschiedlichen Implementierungen kommen kann [Na11] (z.B. durch Probleme bei der Verständlichkeit des BPMN Metamodells, vgl. [CDR12]).

- Unified Modeling Language

Die Unified Modeling Language¹²⁴ [OM11a] ist eine grafische Modellierungssprache, die für die Spezifikation und Dokumentation von Anwendungssystemen entwickelt wurde. UML umfasst eine Vielzahl von unterschiedlichen Modelltypen (insgesamt 14 verschiedene) (vgl. [RQ12]). Das Anwendungsfalldiagramm¹²⁵ dient beispielsweise zur Spezifikation des Verhaltens eines Anwendungssystems nach außen und stellt dar wie User mit dem System interagieren können. Ein weiteres Beispiel ist das Zustandsdiagramm. Hiermit werden die zur Laufzeit erlaubten Zustände eines Objekts (oder Systems) modelliert. Zusätzlich kann definiert werden, welche Ereignisse einen Übergang in einen anderen Zustand triggern. Das Aktivitätsdiagramm visualisiert detailliert Abläufe und verfeinert beispielsweise ein Anwendungsdiagramm hinsichtlich dynamischer Aspekte eines Anwendungssystems. Das Aktivitätsdiagramm kann neben dem ursprünglichen Einsatzzweck auch für die Geschäftsprozessmodellierung eingesetzt werden [Me12c]. Zudem können automatisierbare Geschäftsprozesse modelliert werden [FRH10][OM11a, S.324ff]. Dies wird durch die Einführung von semantisch präzisen Ablauf- und Objektsflussregeln ab der UML Version 2 möglich¹²⁶.

Die Abbildung 37 zeigt ein Aktivitätsdiagramm am Beispiel eines Kundenanfrageprozesses (in Anlehnung an [OM11a, S.330]). Das rechteckige Element „Kundenanfrage“ liegt auf dem abgrenzenden Rahmen¹²⁷ und identifiziert die Visualisierung „Kundenanfrage eingegangen“ als Aktivität.

¹²⁴ Aktuell ist die UML Spezifikation in der Version 2.4.1 standardisiert und veröffentlicht [OM11a]. Die Version 2.5. ist zum Zeitpunkt dieser Arbeit noch nicht offiziell veröffentlicht (Beta-Status).

¹²⁵ In der Literatur wird häufig die englische Übersetzung „Use Case Diagram“ verwendet.

¹²⁶ Das Ausführen und Durchlaufen von Aktionen wird mit Hilfe von Token (Marken) überwacht. Für weitere Ausführungen zur Anwendung von Token vgl. [RQ12, S.275ff][HM08, S.70ff].

¹²⁷ Vgl. [OM11a, S.330]: „The definition (...) use the border notation to indicate that it is an activity.“

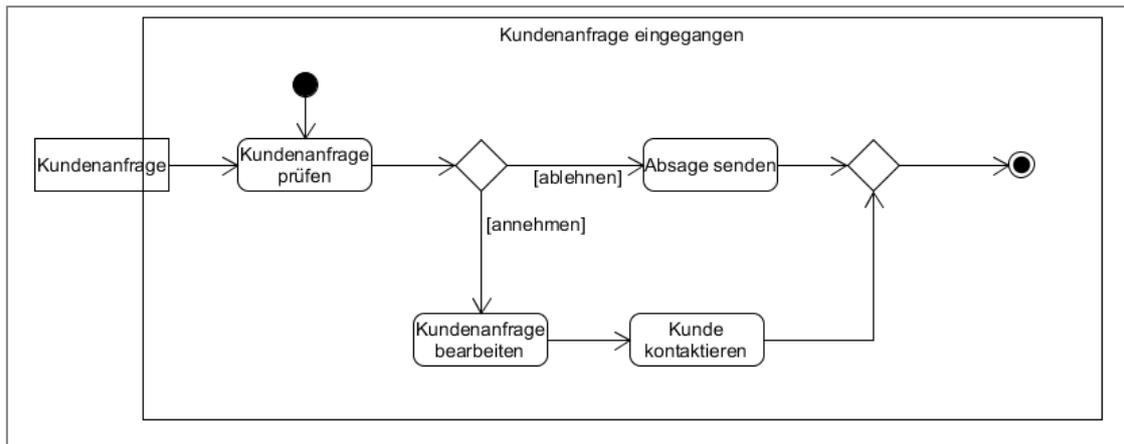


Abbildung 37: Beispiel - UML Aktivitätsdiagramm

Im Allgemeinen können die Elemente des Aktivitätsdiagramms in Aktionen, Kontrollstrukturen, Objektknoten und Kanten aufgeteilt werden.

Aktionen bilden das kleinste ausführbare Element des Aktivitätsdiagramms, wobei Aktionen selbst von anderen Aktionen verfeinert werden können. Der Input einer Aktion wird während der Ausführung in einen Output transformiert. Sowohl Input als auch Output können hierbei leer sein. Neben dem Standardelement der Aktion bietet UML weitere spezialisierte Aktionen. Beispielsweise sendet die „BroadcastSignal-Aktion“ ein Signal an alle möglichen Zielobjekte. In Abbildung 37 wird zuerst eine Aktion zur Prüfung der Kundenanfrage ausgeführt, bevor je nach Entscheidung eine Absage versendet wird und die Aktionen „Kundenanfrage bearbeiten“ und „Kunde kontaktieren“ ausgeführt werden. Entscheidungen werden über sogenannte Kontrollknoten modelliert. Sowohl Objekt- als auch Kontrollflüsse können durch Kontrollknoten beeinflusst werden. In Abbildung 37 zum Beispiel ist die Bedingung, dass die Kundenanfrage bearbeitet wird eine erfolgreiche (positive) Prüfung. Neben solchen Verzweigungen und Zusammenführungen können Kontrollknoten auch zur Parallelisierung und Synchronisation genutzt werden. Die Objektknoten stellen die mit der Aktivität verbundenen Objekte dar. Das Element kann sowohl für die Visualisierung von Input- oder Output-Objekten genutzt werden. In Abbildung 37 wird beispielsweise der Objektknoten „Kundenanfrage“ zur Übergabe von Daten verwendet. Alle Elemente des Aktivitätsdiagramms werden mit gerichteten Kanten verbunden, welche die Sequenz der Ausführung anzeigen. Falls die Komplexität eines Aktivitätsdiagramms zu hoch wird (z.B. durch eine hohe Anzahl der beinhaltenden Aktionen) können Teile des Modells ausgelagert werden (hierarchische Dekomposition).

Den Vorteilen von des UML Standards (z.B. Geschäftsprozesse können sowohl fachlich modelliert, als auch direkt ausgeführt werden) stehen unterschiedliche Nachteile gegenüber. Die Dokumentation des UML Standards ist sehr umfangreich¹²⁸. Eine erfolgreiche Nutzung der Modellierungssprache setzt somit einen hohen Einarbeitungsaufwand – insbesondere für Nicht-Experten – voraus. Fettke (vgl. [Fe05]) kritisiert insbesondere auftretende Mehrdeutigkeiten in UML Konstrukten und eine kognitive Unzugänglichkeit verschiedener UML Bestandteile. Zusätzlich zu der umfangreichen Standarddokumentation kann UML individuell erweitert werden. Durch “Semantic Variation Points” können neue Datentypen oder Ereignisse definiert werden. Diese individuelle Erweiterung¹²⁹ erhöht die Komplexität (z.B. die Pflege des angepassten Standards) der Modellierungssprache zusätzlich.

Aufgrund der ursprünglichen Zielsetzung von UML, der Spezifikation und Dokumentation von Anwendungssystemen, eignet sich die Modellierungssprache und deren Visualisierung im Vergleich zu anderen weniger zur Geschäftsprozessmodellierung.

- XML Process Definition Language (XPDL)

XPDL ist eine auf XML¹³⁰ basierte Modellierungssprache. XPDL dient zum einen zur Definition von ausführbaren Geschäftsprozessmodellen, zum anderen wird die Sprache als Austauschformat zwischen unterschiedlichen GPM-Werkzeugen eingesetzt [Wf15]. Die Nutzung als Austauschformat wird hierbei von der Workflow Management Coalition (WfMC) auch als Schwerpunkt von XPDL dargestellt¹³¹. Die WfMC entwickelt und pflegt die offizielle Spezifikation von XDPL (aktuell in Version 2.2) [Wf12]. Die Spezifikation selbst schreibt eine relativ geringe Anzahl von vordefinierten Modellierungselementen vor¹³². Stattdessen ermöglicht XPDL eine flexible Erweiterung, beispielsweise zur Beschreibung von produktspezifischen Elementen [Rü08]. Des

¹²⁸ Die aktuelle Dokumentation des UML Standards der Version 2.4.1 umfasst 748 Seiten [OM11a].

¹²⁹ Rumpe [Ru11, S.6] vergleicht UML mit der Deutschen Sprache: „Damit erhält UML Charakteristika eine Umgangssprache wie zum Beispiel Deutsch, die es ebenfalls erlaubt, das Vokabular in Form von Fachsprachen oder Dialekten anzupassen.“

¹³⁰ XML ist die Abkürzung für Extensible Markup Language und beschreibt eine (textbasierte) Sprache zur Formulierung von hierarchisch strukturierter Daten, vgl. <http://www.w3.org/XML/>.

¹³¹ Vgl. [Wf12, S.8]: „A variety of different tools may be used to analyse, model, describe and document a business process. The process definition interface defines a common interchange format, which supports the transfer of process definitions between separate products.“

¹³² Als Hauptelemente von XPDL können die Package-, Application-, WorkflowProcess-, Activity-, Transition-, Participant-, DataField- und DataType – Elemente genannt werden, vgl. [Aa03a, S.3f].

Weiteren ist es möglich ein weitreichendes Mapping z.B. zwischen XPDL und BPMN durchzuführen, sodass XPDL wiederum als Austauschformat verwendet werden kann. Die Tabelle 15 stellt eine Auswahl von Mapping-Möglichkeiten als Beispiel dar.

Tabelle 15: Beispiele zum Mapping von XPDL zu BPMN (vgl. [Wh03])

Bezeichnung	BPMN Darstellung	XPDL Darstellung
Aktivität		<code><Activity> <Implementation> <Tool/> <Performer/> </Implementation> </Activity></code>
Subprozess (collapsed)		<code><Activity> <Implementation> <SubFlow/> </Implementation> </Activity></code>
Sequenz		<code><Transition/></code>

Neben der Möglichkeit, dass XPDL als gemeinsames Austauschformat genutzt werden kann, führt die offene Spezifikation zu unkontrollierten Erweiterungen (z.B. durch unterschiedliche Erweiterung der Funktionalität). Ein problemloser Austausch der Geschäftsprozessmodelle ist somit in der Praxis nicht immer gewährleistet [Rü08]. Seit der Serialisierbarkeit von BPMN (ab Version 2.0) ist es jedoch nicht zwingend notwendig Geschäftsprozessmodelle in XPDL abzuspeichern.

- WS-BPEL

WS-BPEL wurde von der OASIS spezifiziert und liegt aktuell in der Version 2.0 vor [OA07]. WS-BPEL ist eine textbasierte Modellierungssprache (Speicherung mit Hilfe von XML). WS-BPEL dient zur Strukturierung bestehender Web Services. Im Gegensatz zu anderen Modellierungssprachen (z.B. BPMN) unterstützt WS-BPEL die direkte Interaktion mit menschlichen Akteuren nicht. WS-BPEL wird zur Orchestrierung von bestehenden Web Services verwendet (vgl. [LLN11]), wobei WS-BPEL-Geschäftsprozesse entweder in abstrakter (z.B. zur Dokumentation eines Geschäftsprozesses) oder ausführbaren Form¹³³ modelliert werden können. Beispielsweise können dadurch im Hintergrund (Backend) laufende Geschäftsprozesse

¹³³ Vgl. [OA07, S.7]: „A WS-BPEL Abstract Process is a partially specified process that is not intended to be executed and that must be explicitly declared as ‘abstract’. Whereas Executable Processes are fully specified and thus can be executed, an Abstract Process may hide some of the required concrete operational details expressed by an Executable artifact.”.

(z.B. automatische Bezahlabwicklung) dargestellt oder mit Hilfe einer WS-BPEL-Engine ausgeführt werden.

In WS-BPEL sind unterschiedliche Elemente definiert, welche zur Modellierung eines Geschäftsprozesses dienen (vgl. [OA07 S.21ff]). Die Tabelle 16 stellt ein Ausschnitt eines WS-BPEL-Geschäftsprozesses dar.

Tabelle 16: Ausschnitt eines WS-BPEL-Geschäftsprozesses [OA07 S.21f]

```
<invoke partnerLink="scheduling"
        portType="lns:schedulingPT"
        operation="sendShippingSchedule"
        inputVariable="shippingSchedule">
  <documentation>
    Complete Production Scheduling
  </documentation>
  <targets>
    <target linkName="ship-to-scheduling" />
  </targets>
</invoke>
```

Die in Tabelle 16 dargestellte `<invoke>`-Aktivität kann als Partneroperation (sendShippingSchedule) aufgerufen werden. Damit dieser Aufruf möglich ist, muss über das Attribut `partnerLink` die Partnerbeziehung spezifiziert werden. Ein Invoke kann hierbei entweder als „One-Way“- oder „Request-Response“-Aufruf ausgeführt werden. Durch die ausschließliche Beschreibung des Attribut `inputVariable` (vgl. Tabelle 16) wird ein „One-Way“-Aufruf gekennzeichnet. Durch `<target>` wird die Nachfolger-Beziehung beschrieben. `<target>` beschreibt, dass die dargestellte Aktivität eine Nachfolger-Aktivität ist und folglich von einer Vorgänger-Aktivität aufgerufen wird (Vorgänger nicht in Tabelle 16 dargestellt). `<documentation>` dient zur Annotation von Text, welcher das menschliche Lesen eines WS-BPEL-Geschäftsprozesses vereinfachen soll.

Zusätzlich zu einfachen Annotationen erweitert BPEL4People WS-BPEL, um die Möglichkeit menschliche Interaktion mit in einen WS-BPEL-Geschäftsprozess zu modellieren (vgl. [Ag07][SSP14]).

3.3. Begriff der gebrauchssprachlichen Modellierung

In diesem Abschnitt wird der Begriff der gebrauchssprachlichen Modellierung definiert. Als Grundlage der Begriffsdefinition wird im ersten Abschnitt dieses Kapitels eine Erweiterung der Definition des Anwendungssystembegriffs von Ortner (vgl. Abschnitt 2.1.1 und [Or12]) beschrieben. Zusätzlich werden die Grundlagen der Sprechakttheorie nach Searle [Se69] erläutert. Sprechakte (z.B. das Treffen einer Vereinbarung) stellen die kleinste Einheit des Sprechens dar und bilden die Basis einer erfolgreichen gebrauchssprachlichen Interaktion.

Die Abbildung 38 skizziert die Erweiterung¹³⁴ der Anwendungssystemarchitektur. Die oberste Ebene stellen Handlungen dar. Als Handlung wird die Nutzung eines Anwendungssystems durch den Menschen bezeichnet (vgl. [MOW14]). Zum Beispiel kann ein Anwendungssystem zur Steuerung eines Publikationsprozesses eingesetzt werden, wobei die einzelnen Aktivitäten des Publikationsprozesses Handlungen darstellen können (z.B. das Schreiben einer Publikation). Als Erweiterung der sieben Ebenen (vgl. Abbildung 38) kann die oberste Ebene präzisiert werden. Nach Mevius et al. [MOW15] enthält die Ebene der Handlungen sogenannte Interaktionsprozesse.



Abbildung 38: Erweiterung nach Ortner (vgl. [MOW15])

Interaktionsprozesse werden durch die unterliegenden Ebenen überwacht und unterstützt.

¹³⁴ In [MOW15] wird der Begriff des Social Computings verwendet. Durch Social Computing werden Anwender unabhängig ihrer IT-Erfahrungen befähigt, Anwendungssysteme zu nutzen [PW07].

Die Unterstützung der Interaktion findet beispielsweise durch (Software-)Programme statt (z.B. virtuelle Interaktion zweier Menschen über das Internet). Zudem werden Interaktionen überwacht, um auf Basis der erfassten Messdaten einzelne Parameter (z.B. Interaktionsdauer) zu optimieren. Hierbei können Interaktionsprozesse in verschiedenen Kontexten stattfinden.

Als Beispiele nennen Mevius et al. [MOW15] Interaktionsprozesse in der Verwaltung, innerhalb eines kulturellen Rahmens und im Kontext des Privat- und Arbeitslebens (vgl. Abbildung 38). Auch aus Sicht des Geschäftsprozessmanagements müssen Interaktionsprozesse identifiziert und erfasst werden, wobei die in Geschäftsprozessen inbegriffenen Interaktionen entweder manuell (Mensch-Mensch-Interaktion) oder teilautomatisiert (Mensch-Maschine-Interaktion) ablaufen. Die Ausführung dieser Geschäftsprozesse erfolgt entweder innerhalb einer Ebene und einem Kontext (z.B. manueller Geschäftsprozess auf Menschebene im Kontext des Arbeitslebens, vgl. Abbildung 38) oder „grenzübergreifend“ über verschiedene Ebenen verteilt. Damit eine gesamtheitliche Integration (d.h. sowohl bei der Erfassung, als auch Ausführung der Geschäftsprozesse) aller Beteiligten von Anfang an möglich ist, muss sichergestellt werden, dass die Integration auf Basis einer Sprache stattfindet, welche von alle Beteiligten beherrscht wird.¹³⁵ Die Gebrauchssprache (vgl. Abschnitt 3.1) definiert den „kleinsten gemeinsamen Nenner“ aller Beteiligten zur Kommunikation. Die Gebrauchssprache, welche Alltags- bzw. Umgangssprache umfasst, ermöglicht zum Beispiel die Äußerung eines Satzes, indem die Bedeutung der verwendeten Wörter nicht bekannt sein muss¹³⁶. Ein Wort der Umgangssprache kann nach Pöschek [Pö05, S.8] „(...) eine Bedeutung haben, es gibt aber viele Wörter in ihr, die keine haben.“. Die Herausforderung, die durch die scheinbar von geringen Anforderungen¹³⁷ geprägten Gebrauchssprache wird hierbei durch die Aussage von Wittgenstein [Wi53, §43] „die Bedeutung eines Wortes ist sein Gebrauch in der Sprache.“ deutlich. Unabhängig von der Erfassungsmethode (z.B. Extraktion der Information durch Interviews, Fragebögen,...) werden Geschäftsprozessmodelle und die darin enthaltenen Informationen zu beliebigen Zeitpunkten zwischen den einzelnen Beteiligten diskutiert bzw. ausgetauscht.

¹³⁵ Die existierende Literatur umfasst hierbei eine Vielzahl verschiedener Ansätze, vgl. z.B. [Ev03] [Co04][CSI11].

¹³⁶ Vgl. Wittgenstein [Wi03, § 4.001-4002]: „Die Gesamtheit der Sätze ist die Sprache. Der Mensch besitzt die Fähigkeit Sprachen zu bauen, womit sich jeder Sinn ausdrücken lässt, ohne eine Ahnung davon zu haben, wie und was jedes Wort bedeutet. (...)“.

¹³⁷ Vgl. Wittgenstein [Wi03, § 4.002]: „Die Umgangssprache ist ein Teil des menschlichen Organismus“.

Beispielsweise können während der gemeinsamen Modellierung von Aktivitäten mehrere Beteiligte über deren Sequenz innerhalb des Geschäftsprozesses diskutieren. Der Gebrauch innerhalb der Sprache muss folglich zwischen den Beteiligten synchronisiert sein, um ein gemeinsames Verständnis der Bedeutung zu erreichen. Die Gebrauchssprache kann hierbei in eine untergeordnete Einheit¹³⁸ des „miteinander Sprechens“ unterteilt werden. Die sogenannten Sprechakte¹³⁹ bilden die Basis der Strukturierung von Kommunikation¹⁴⁰ (vgl. [Be07, S.277]). Gelingen durchgeführte Sprechakte ermöglichen die erfolgreiche Verwendung, Stabilisierung und Erweiterung der Gebrauchssprache.

Exkurs Sprechakttheorie

Die Sprechakttheorie gründet auf den Ausführungen von Austin [Au62] und Searle [Se69]. Der Sprechakttheorie zufolge ist das Sprechen selbst eine Handlung. Eine Unterscheidung zwischen ‚tun‘ und ‚sprechen‘ wird in der Sprechakttheorie nicht berücksichtigt¹⁴¹ [Hi10]. Zur Sprechakttheorie existieren in der Literatur unterschiedliche Ansätze, im Folgenden werden die Ausführungen von Searle [Se69] [Se83] dargestellt. Nach Searle umfasst ein Sprechakt vier Teilakte, vgl. Tabelle 17.

Tabelle 17: Vier Teilakte eines Sprechakts nach Searle

Sprechakt	Äußerungsakt	Beschreibung einer Handlung, in welcher der Sprecher eine Äußerung (z.B. Wörter, Sätze) artikuliert.
	Propositionaler Akt	Beschreibung der Handlung, in welcher der Sprecher eine Referenz und Prädikation auf etwas definiert. Beispielsweise kann eine Äußerung eine Referenz auf einen Gegenstand enthalten. Des Weiteren können

¹³⁸ Nach Searle bestehe „der Grund für die Konzentration auf die Untersuchung von Sprechakten einfach darin, dass zu jeder sprachlichen Kommunikation sprachliche Akte gehören. Die Grundeinheit der sprachlichen Kommunikation ist nicht, wie allgemein angenommen wurde, das Symbol, das Wort oder der Satz, sondern die Produktion oder Hervorbringung des Symbols oder Wortes oder Satzes im Vollzug des Sprechaktes.“ [Se83, S.30].

¹³⁹ Die Kommunikation muss nicht zwingend mündlich geschehen. Beispielsweise kann der Informationsaustausch durch eine schriftliche Nachricht (z.B. Brief) erreicht werden, vgl. Henne [He75].

¹⁴⁰ Der Begriff der Kommunikation ist in der Literatur unterschiedlich definiert. In dieser Arbeit wird Kommunikation als ein Bestandteil einer Interaktion aufgefasst. Vgl. des Weiteren zur Diskussion der Begriffe [Wa87, S.28ff].

¹⁴¹ Vgl. [Hi10, S.4]: „Sowohl Reden als auch konkretes praktisches Tun sind Handlungen; sie stellen jedoch jeweils verschiedene Handlungstypen [praktische Handlungen, sprachliche Handlungen] dar (...)“.

		durch die Prädikation dem „Etwas“ (z.B. dem Gegenstand) Eigenschaften zugeteilt werden.
	Illokutionärer Akt	Beschreibung der Handlung, mit der der Sprecher eine bestimmte Form der Äußerung erzielt (z.B. Behauptung, Frage, Befehl).
	Perlokutionärer Akt	Beschreibung der Handlung, mit der eine bestimmte Form der Wirkung beim Empfänger der Äußerung erzielt wird (z.B. Überzeugung, Einschüchterung,...).

Searle definiert zudem auf Basis des illokutionären Teilakts fünf Arten von Sprechakten. Obwohl keine eindeutige Abgrenzung der unterschiedlichen Arten möglich ist [Hi10], ermöglicht die Klassifikation eine verbesserte Übersicht über die (gemeinsamen) Eigenschaften der unterschiedlichen Sprechakte¹⁴². Zur Unterscheidung der Arten dienen beispielsweise der Zweck und die Ausrichtung der Äußerung. Die Tabelle 18 listet die Klassifikation nach Searle auf.

Tabelle 18: Klassifikation der Sprechakte nach Searle

repräsentativ	Der Sprecher drückt die Wahr –oder Falschheit der Proposition aus (z.B. Feststellung, Behauptung).
direktiv	Der Sprecher fordert den Empfänger (z.B. Zuhörer) zu der Ausführung einer Handlung auf (z.B. Bitte, Befehl).
kommissiv	Der Sprecher verpflichtet sich selbst zu einer Handlung (z.B. Versprechung, Drohung).
expressiv	Der Sprecher äußert seinen psychischen Zustand (z.B. Entschuldigung, Danksagung).

¹⁴² Vgl. des Weiteren [Hi10, S.43]: „Auf diese Weise kann man eine gewisse Übersicht über die Vielfalt der Phänomene schaffen und kann durch die Einführung entsprechender Bezeichnungen für Gruppen von Sprechhandlungen ein fachsprachliches Vokabular etablieren, das beim Sprechen über Sprechhandlungen sehr hilfreich ist.“

deklarativ	Der Sprecher schafft durch die Äußerung neue Sachverhalte institutioneller Art (z.B. Taufe, Erlass).
------------	--

Ob Sprechakte gelungen oder fehlgeschlagen sind, wird durch sogenannte Gelingensbedingungen beschrieben [Sa03]. Nach Sander [Sa03, S.14] kann eine Äußerung „(...) zum einen als gelungen bezeichnet werden, wenn der jeweilige Sprecher sein Redeziel mit Hilfe des Sprechaktes tatsächlich realisiert hat – in diesem Sinne ist etwa ein Befehl gelungen, wenn der Adressat die Handlung, deren Ausführung gefordert wurde, tatsächlich ausführt. Es bietet sich zu Zwecken der terminologischen Differenzierung an, in diesem Fall vom Erfolg eines Sprechaktes zu reden. Andererseits kann man sich mit dem Ausdruck ‘gelingen’ auf diejenigen Bedingungen beziehen, die erfüllt sein müssen, damit man die jeweilige sprachliche Handlung als korrekt bezeichnen kann – so gehört es in diesem Sinne zu den Gelingensbedingungen für Befehle, dass der Autor der Sprechhandlung ein Vorgesetzter (sensu lato) des Adressaten ist. (...)“.

Searle [Se83, S.84ff] definiert Regeln (Regeln des propositionalen Gehalts, Einleitungsregeln, Regeln der Aufrichtigkeit, wesentliche Regeln) auf deren Basis das Gelingen eines Sprechaktes festgestellt werden kann. Für weitere Ausführungen zur Sprechakttheorie sei auf [Au62][Se69][Sa03][Hi10][Vi11] verwiesen.

Zur Modellierung von Geschäftsprozessen muss sichergestellt werden, dass die auf Basis der Sprechakte erfahrenen Informationen, für alle Beteiligten verständlich erfasst werden. Neben der notwendigen Bedingung der Interaktion¹⁴³ zur Entstehung von Sprechakten muss des Weiteren die Synchronisation aller Beteiligten und die Qualität der Modellierungsergebnisse gewährleistet werden. Insbesondere wenn mehrere Beteiligte an einem GPM-Projekt mitwirken, unterstützt eine regelmäßige Synchronisation den Abgleich von Informationen und Wissensständen. Beispielsweise können direktiv geäußerte Sprechakte (z.B. Festlegung eines bestimmten Modellierungsstandards) durch eine Synchronisation an alle Beteiligten kommuniziert werden. Zusätzlich muss gewährleistet werden, dass die Qualität von Geschäftsprozessmodellen, den vordefinierten Anforderungen der Beteiligten entsprechen (z.B. ein Geschäftsprozessmodell muss alle ausführungsrelevanten Aktivitäten enthalten). Die Einhaltung der vordefinierten Qualitätskriterien fördert hierbei eine erfolgreiche

¹⁴³ Sprechakte sind Bestandteil einer Interaktion und treten nicht unabhängig davon auf [Mi09].

Anwendung und Akzeptanz (z.B. durch Anwender). Basierend auf den vorangehenden Ausführungen, können die in Abbildung 39 dargestellten Anker (Interaktion, Synchronisation, Qualität) zur gebrauchssprachlichen Modellierung identifiziert werden.

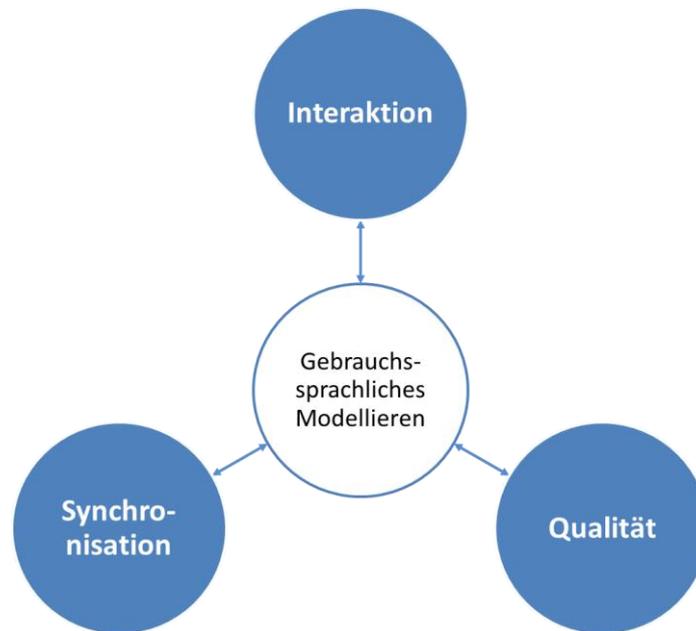


Abbildung 39: Anker der gebrauchssprachlichen Modellierung

Der erste Anker umfasst den Begriff der Interaktion¹⁴⁴. Interaktion wird in der Literatur nicht einheitlich definiert. In Fuchs-Heinritz [Fu95, S.308] wird soziale Interaktion als „(...) die durch Kommunikation (Sprache, Symbole, Gesten usw.) vermittelten wechselseitigen Beziehungen zwischen Personen und Gruppen und die daraus resultierenden wechselseitige Beeinflussung ihrer Einstellungen, Erwartungen und Handlungen. (...)“ definiert. Im Gegensatz zum Begriff der Kommunikation ist die Bedeutung des Interaktionsbegriffs weiter gefasst, wobei insbesondere die Wechselwirkung zwischen zwei oder mehreren Menschen im Fokus steht [Ke04]. Interaktion kann des Weiteren als produktiver Prozess aufgefasst werden.

Beispielsweise beschreibt Blechschmitt [Bl05, S.28] „Menschen, die miteinander interagieren, können eine Idee besprechen und diese interaktiv verfeinern.“. Insbesondere

¹⁴⁴ Im Kontext dieser Arbeit wird der Begriff der Interaktion aus zwei unterschiedlichen Perspektiven betrachtet. Zum einen wird auf Basis der Gebrauchssprache die Interaktion aller Beteiligten während der Erfassung, Anreicherung, Ausführung, Analyse und Optimierung von Geschäftsprozessen diskutiert (vgl. Abschnitt 4.2). Zum anderen werden die Möglichkeiten der Erfassung von Interaktion in Geschäftsprozessmodellen beschrieben (vgl. Abschnitte 4.3 und 5.2.2). Für die Modellierung von Interaktion sei des Weiteren auf Harrison-Broninski verwiesen [Ha05a][Ha15].

während der gemeinsamen Beschreibung von (Geschäfts-)prozessen wird das für die Ausführung notwendige Wissen generiert [Mü01]. Dabei ist im Kontext der Generierung von neuem Wissen (z.B. Schulung eines Geschäftsprozesses) bestimmend „(...) wie es gelingt wechselseitigen Beziehungen aufzubauen und die Einstellungen, Erwartungen und Handlungen als Grundlage der Wissensprozesse gegenseitig zu beeinflussen.“ [Re03, S.76]. Aus der Sicht einer Mensch-Maschine-Beziehung „(...) entsteht die Interaktion erst durch deren [Hardware- und Softwarekomponenten] Zusammenspiel mit dem Menschen. Zwar sind durch die Systemauslegung Vorgaben gemacht, welcher Art diese sind, kann aber erst in der Interaktion selbst beobachtet und erfahren werden.“ [OI07, S.389]. Zum Beispiel kann nur unmittelbar während der Interaktion zwischen Mensch und Maschine festgestellt werden, ob die IT-Unterstützung zur Ausführung einer Geschäftsprozessaktivität hilfreich ist. Eine Interaktion führt hierbei immer zur Verteilung von Wissen. Wissen, welches anfangs nur von einzelnen Menschen generiert wurde, kann durch ständige Interaktion auf eine organisationale Ebene (d.h. für alle Beteiligte) transferiert werden [Be14].

Neben der Interaktion von Beteiligten (z.B. spontaner Austausch von Ideen zwischen zwei Personen) muss sichergestellt werden, dass der relevante Informations- bzw. Wissensstand in Bezug auf die Geschäftsprozesse und deren Modelle an alle Beteiligten vermittelt wird. Wurde beispielsweise, die in einem GPM-Projekt verwendete Gebrauchssprache, um neue Fachbegriffe angereichert (z.B. während eines Modellierungsworkshops), müssen nicht anwesende Personen über die Weiterentwicklung informiert werden. Dieser Schritt der Informationsübermittlung kann als Synchronisation bezeichnet werden und definiert den zweiten Anker der gebrauchssprachlichen Modellierung. Nach Brauer et al. beschreibt der Begriff der Synchronisation die „(...) Einschränkung von Nebenläufigkeit (...)“ [Br87, S.119], wobei Handlungen in kausaler und zeitlicher Hinsicht abgestimmt werden. Im technischen Kontext von Systemprozessen (z.B. zwei Systeme bearbeiten gleichzeitig ein Set von Daten) erläutert Freisleben die Notwendigkeit von Synchronisation an zwei Beispielen [Fr87, S.9]: „Wenn beispielsweise ein Prozess eine bestimmte Aktion durchgeführt haben muss, bevor ein zweiter weiterlaufen kann (z.B. das Setzen einer Variable oder das Senden einer Nachricht), dann müssen die Abläufe der beiden Prozesse aufgrund einer bestimmten Bedingung beeinflusst werden. (...) Eine Synchronisation wird zusätzlich notwendig, wenn sich Prozesse gemeinsame Betriebsmittel wie

Prozessor, Speicherplätze oder Ein-Ausgabe-Geräte teilen. Denn wenn mehrere Prozesse mit unvorhersagbaren Geschwindigkeiten gleichzeitig auf gemeinsamen Daten operieren, dann ist auch das Resultat unvorhersagbar, da keiner der Prozesse die Möglichkeit hat, herauszufinden, wie andere die Daten verändert haben.“

Analog zu Brauer und Freisleben müssen die Ergebnisse des Geschäftsprozessmanagements (z.B. Geschäftsprozessmodelle, Geschäftsprozess-applikationen) zur ganzheitlichen Einbindung aller Beteiligten regelmäßig synchronisiert werden (vgl. [ŠT13]). Wird die Synchronisation auf Basis der Gebrauchssprache durchgeführt, ist sichergestellt, dass unabhängig des Vorwissens der Beteiligten relevante Informationen kommuniziert werden können. Die Modellierung von regelkonformen Geschäftsprozessen (vgl. [Sc14b]) wird durch die unmittelbare Überprüfung durch alle Beteiligte unterstützt. Hierdurch wird gewährleistet, dass beispielsweise neue Richtlinien unmittelbar bei der Modellierung berücksichtigt werden und die ausgeführten Geschäftsprozesse den vordefinierten Qualitätskriterien genügen. Zusätzlich wird sichergestellt, dass ein „ständiger Kontakt“ zwischen Anwender und Modellierer besteht. Nach Kolb et al. hat insbesondere die soziale Beziehung zwischen den Beteiligten Einfluss auf die Genauigkeit von Geschäftsprozessmodellen (vgl. [Ko14]).

Der dritte Anker hebt die Notwendigkeit der Einhaltung von ex ante definierten Qualitätsanforderungen hervor. Insbesondere bei der „Übersetzung“ von Sprache zu Modell können relevante Informationen verloren gehen. Nach Pitschke ist diese Übersetzungslücke verstärkt ausgeprägt, wenn die Modellierung und Ausführung durch verschiedene Individuen erfolgt [Pi12]. Der Begriff der Qualität wird in der Literatur unterschiedlich definiert (vgl. [Ga84]). Crosby definiert Qualität als die „Erfüllung der vereinbarten Anforderungen“ [Cr80]. Nach [DI05] beschreibt Qualität einen „Grad, in dem ein Satz inhärenter Merkmale Anforderungen erfüllt“ wird. Des Weiteren definiert Garvin [Ga88, S.49ff] acht Sichtweisen zur Analyse von Qualität eines Produkts oder Services: Performance, Eigenschaften, Zuverlässigkeit, Konformität, Widerstandsfähigkeit, Gebrauchsfähigkeit, Ästhetik, Wahrnehmung¹⁴⁵. Diese unterschiedlichen Sichtweisen können über Produkte und Services auch für Geschäftsprozesse adaptiert werden. Insbesondere wenn der Einsatz der

¹⁴⁵ Vgl. engl. Übersetzung: Performance (performance), Eigenschaften (features), Zuverlässigkeit (reliability), Konformität (conformance), Widerstandsfähigkeit (durability), Gebrauchsfähigkeit (serviceability), Ästhetik (aesthetics), Wahrnehmung (perceived quality) [Ga88, S.49ff].

Gebrauchssprache vernachlässigt wird, können Geschäftsprozessmodelle nicht mehr von allen Beteiligten verstanden werden, wodurch beispielsweise die Zuverlässigkeit bei der Ausführung von Geschäftsprozessen beeinträchtigt wird. Becker et al. [BRU00] beschreiben daher die Notwendigkeit, dass über die syntaktische Korrektheit der Geschäftsprozessmodelle der Aspekt der Qualität berücksichtigt werden muss. Krogstie et al. [KLS95] definieren ein Rahmenwerk, welches den Begriff der Qualität in neun Teilaspekte gliedert, welche einzeln auf (konzeptuelle) Geschäftsprozessmodelle angewendet werden können. Neben den technisch messbaren Kriterien (z.B. physische Verfügbarkeit der Geschäftsprozessmodelle) berücksichtigen Krogstie et al. „weiche“ Qualitätskriterien (z.B. Angemessenheit der Sprache gegenüber den Beteiligten). Die Tabelle 19 listet die Teilaspekte und deren Beschreibungen auf.

Tabelle 19: Teilaspekte von Modellqualität nach [KLS95]

Teilaspekt der Qualität	Beschreibung
Wissen	Die Qualität des Wissens bezieht sich auf das Wissen der Beteiligten in Bezug auf die Domäne. Nach Krogstie et al. ist der „perfekte“ (unrealistische) Zustand erreicht, wenn die Beteiligten zu einem bestimmten Zeitpunkt alles über die Domäne wissen und keine falschen Annahmen treffen.
Sprache	Die Qualität in Bezug auf Sprache wird insbesondere durch die Angemessenheit der Sprache gegenüber der Domäne, den Beteiligten und der technischen Möglichkeit einer Automatisierung definiert.
Semantik	Die semantische Qualität umfasst nach Krogstie et al. zwei Parameter: Validität und Vollständigkeit. Mit Validität ist definiert, dass alle in einem Modell verwendeten Elemente korrekt und für das Problem (z.B. Modellierung eines domänenspezifischen Geschäftsprozesses) von Bedeutung sind. Der Aspekt der Vollständigkeit beschreibt, dass die für die Erstellung des Modells verwendete Sprache alle,

	für die Domäne relevanten, Elemente zu einer korrekten Modellierung eines Problems enthält ¹⁴⁶ .
Wahrgenommene Semantik	Die wahrgenommene semantische Qualität beschreibt das Verhältnis zwischen der Interpretation der Beteiligten und deren Wissen in Bezug auf die Domäne. Zur Messung der Qualität definieren Krogstie et al. zwei Parameter: wahrgenommene Validität und wahrgenommene Vollständigkeit.
Pragmatik	Die pragmatische Qualität beschreibt das Verhältnis zwischen dem erfassten Modell und dessen Interpretation durch die Beteiligten. Hierbei wird als Qualitätskriterium ausschließlich der Grad der Möglichkeit des Verständnisses nach Krogstie et al. berücksichtigt. Beispielsweise sind sehr große Modelle (d.h. z.B. Modelle mit vielen Elementen) für Beteiligte schwieriger zu verstehen als kleine, übersichtliche Modelle (vgl. des Weiteren [Me07b]).
Syntax	Die syntaktische Korrektheit wird durch diesen Teilaspekt beschrieben. Als syntaktisch Korrekt wird ein Modell bezeichnet, wenn alle verwendeten Modellierungselemente entsprechend der vorgegebenen Syntax sind.
physisch	Die physische Qualität resultiert auf dem Maß der persistenten Speicherung des Modells und dem Grad der Verfügbarkeit des Modells für alle Beteiligten.
sozial	Die relative bzw. absolute „Übereinkunft“ von Wissen und Interpretation wird von Krogstie et al. als soziale Qualität beschrieben. Als relativ werden unterschiedliche Vorstellungen

¹⁴⁶ Die semantische Qualität muss hierbei an die „reale Machbarkeit“ angepasst werden, vgl. : „Hence, for our semantic goals to be realistic, they have to be somewhat relaxed, by introducing the concept of Feasibility. Attempts at reaching a state of total validity and completeness will lead to unlimited spending of time and money for the modeling activity. The time to terminate a modeling activity is thus not when the model is perfect, but when it has reached a state where further modeling is less beneficial than applying the model in its current state.“ [KLS95, S.224].

	(z.B. der Bedeutung von Modellierungselementen) von Beteiligten bezeichnet, welche sich nicht gegenseitig widersprechen. Die absolute Übereinkunft spiegelt den Grad der gleichen Vorstellungen aller Beteiligten wider.
--	--

Die Qualitätskriterien nach Krogstie et al. können bei der Implementierung von konkret messbaren Qualitätsindikatoren (vgl. Abschnitt 4.2.4) einen Rahmen bilden. Hierbei muss sichergestellt werden, dass sowohl Aspekte in Bezug auf weiche als auch harte Aspekte der Qualität berücksichtigt werden. Um das Verhältnis zwischen den drei Ankern Interaktion, Synchronisation und Qualität zusammenfassend zu beschreiben, stellt die Tabelle 20 die Definition des Begriffs der gebrauchssprachlichen Modellierung dar.

Tabelle 20: Definition gebrauchssprachliche Modellierung

Definition: Gebrauchssprachliche Modellierung
<i>Werden die, während einer Interaktion gelungenen Sprechakte, für alle Prozessbeteiligten unabhängig deren Vorwissens, unter Berücksichtigung ex ante definierter Qualitätskriterien, dynamisch in Modelle erfasst, wird dies als gebrauchssprachliche Modellierung bezeichnet.</i>

Die Beziehung von Sprechakt, Gebrauchssprache und Modellierungssprache ist in Abbildung 40 schematisch dargestellt. Sprechakte werden mit Hilfe der Gebrauchssprache geäußert. Die während einer Interaktion getätigten Sprechakte werden gegen ex ante definierte Qualitätskriterien geprüft, um mögliche auftretende Fehler (z.B. dass ein direkter Sprechakt zu einer falschen Modellierungshandlung führt) zu vermeiden.

Fehlende Informationen (z.B. Unklarheiten in Bezug auf die Verwendung von bestimmten Begriffen - vgl. Abschnitt 2.1.3) werden nach Klärung dynamisch ergänzt. Anschließend können gelungene Sprechakte mit Hilfe einer Modellierungssprache in einem Modell erfasst werden.

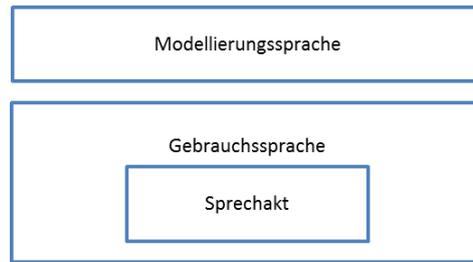


Abbildung 40: Sprechakt - Gebrauchssprache – Modellierungssprache

Die in Abbildung 40 dargestellten Ebenen stehen in einer bidirektionalen Beziehung, d.h. es muss gewährleistet sein, dass die Extraktion von Informationen aus den erfassten Modellen gebrauchssprachlich möglich ist. Muss beispielsweise zur Schulung von Anwendern ein Geschäftsprozessmodell erklärt werden, muss hierbei die gemeinsame Gebrauchssprache ausreichend sein.

In Bezug auf das agile Geschäftsprozessmanagement ist zu berücksichtigen, dass während der Modellierung von Geschäftsprozessen kein Ungleichgewicht entsteht. Nach Braten [Br73] besteht das Problem des „Modellmonopols“, welches zu einem Ungleichgewicht zwischen den einzelnen Prozessbeteiligten führen kann. Modellieren beispielsweise fachliche Prozessbeteiligte initial ein für sie aussagekräftiges Modell, können Asymmetrien entstehen, sofern die technischen Prozessbeteiligten, welche die Anreicherung von Geschäftsprozessen durchführen, die fachlichen Modelle nicht korrekt interpretieren können. Hierdurch können erhebliche Missverständnisse und Folgefehler auftreten. Durch die Nutzung der Gebrauchssprache (vgl. [He06]) als Kommunikationsmittel über das gesamte Geschäftsprozessmanagement wird dem Entstehen eines Ungleichgewichts entgegengewirkt. Auf Basis der gebrauchssprachlichen Modellierung werden zunächst die Geschäftsprozessmodelle initial erfasst und nachfolgend gemeinsam iterativ und inkrementell verfeinert. Zudem ermöglicht die gebrauchssprachliche Modellierung die Einbindung aller Prozessbeteiligten unabhängig ihrer Vorqualifikation (z.B. Besitz von Kenntnissen bzgl. einer bestimmten Modellierungssprache). Diese Eigenschaft ist für agile Vorgehensweisen (vgl. Abschnitte 2.2.2 und 2.3), welche auf die Interaktion aller Beteiligten fokussieren, essentiell. Im nachfolgenden Abschnitt (Abschnitt 4) wird zur signifikanten Verbesserung der Kommunikation, Interaktion sowie des gegenseitigen Verständnisses der Prozessbeteiligten, die Methode $BPM(N)^{Easy}$ beschrieben. Die Methode ermöglicht eine durchgängige Unterstützung des agilen Geschäftsprozessmanagements.

4. BPM(N)^{Easy} – Methode zum agilen Geschäftsprozessmanagement

In diesem Kapitel wird basierend auf den vorherigen Ausführungen die Entwicklung der Methode BPM(N)^{Easy} beschrieben. Abbildung 41 stellt hierbei mögliche Lösungsansätze von BPM(N)^{Easy}, für die in Abschnitt 1 identifizierten Herausforderungen und Probleme, zusammengefasst dar.

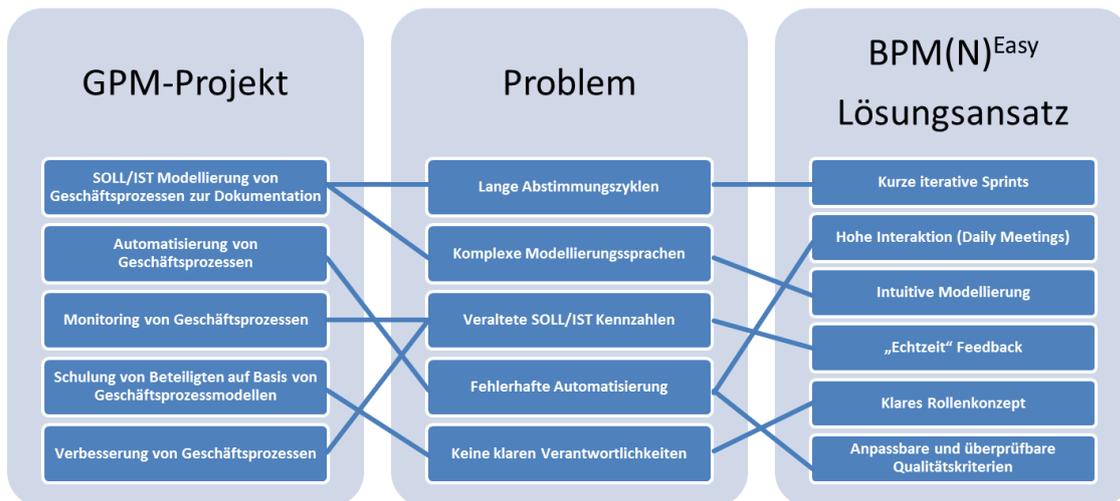


Abbildung 41: Aufgaben, Probleme, BPM(N)^{Easy}-Lösungsansätze in GPM-Projekten

In Abschnitt 4.1 wird die für diese Arbeit gültige Bedeutung des Methodenbegriffs definiert und das zugrundeliegende Methodendesign vorgestellt. Anschließend wird in Abschnitt 4.2 die BPM(N)^{Easy} Vorgehensweise dargestellt. Neben der Beschreibung „wie“ und für „was“ BPM(N)^{Easy} angewendet werden kann, wird beschrieben „wer“ die Aktivitäten innerhalb von BPM(N)^{Easy} ausführt. Beispielsweise ist die Rolle des „IT-Experten“ („wer“) für die Automatisierung der Geschäftsprozesse („was“) verantwortlich. Die zuvor erfassten BPMN^{Easy}-Geschäftsprozessmodelle werden hierzu (z.B. durch Anbindung von Drittsystemen) angereichert („wie“). Zusätzlich wird dargestellt wie in BPM(N)^{Easy} die Aspekte der gebrauchssprachlichen Modellierung Anwendung finden.

In Abschnitt 4.3 wird die grafische Modellierungssprache BPMN^{Easy} beschrieben. Zunächst wird das grafische Elementset, welches aus Event-, Aktivitäten- und Gateway-Elementen besteht, dargestellt. Zusätzlich zu dieser Beschreibung werden die Möglichkeiten der Multimediaannotation (z.B. Verknüpfung einer Aktivität mit einer Audiosequenz) diskutiert. Abschließend wird veranschaulicht wie BPMN^{Easy}-Modelle BPMN 2.0 konform gesichert werden können. Durch die Nutzung des BPMN 2.0

Standards zur formalen Speicherung der Modelle kann BPMN^{Easy} einfach zwischen unterschiedlichen Werkzeugen ausgetauscht werden.

In Abschnitt 4.4 werden zwei Werkzeuge vorgestellt. Zum einen wird mit BPM Touch [ba15] eine Software dargestellt, welche als App auf Tablets ausgeführt werden kann. BPM Touch erlaubt das mobile Erfassen von BPMN^{Easy}-Modellen. Insbesondere bei einer agilen (ortsunabhängigen) Erfassung von Geschäftsprozessen stellt die Mobilität einen signifikanten Vorteil dar. Zum anderen wird mit der Axon.ivy BPM Suite [AX15a] ein Geschäftsprozessmanagementsystem beschrieben, welches sowohl als Modellierungs- und Automatisierungswerkzeug, als auch zum Monitoring von Geschäftsprozessen eingesetzt werden kann. Mit Axon.ivy können somit BPMN^{Easy}-Modelle angereichert werden. Des Weiteren kann Axon.ivy als Entwicklungsplattform zur Implementierung von eigenen Geschäftsprozessapplikationen genutzt werden, welche selbst wiederum zur Verwaltung und Monitoring von Geschäftsprozessen angewendet werden können.

4.1. Methodendesign

BPM(N)^{Easy} beschreibt eine agile Methode zum Geschäftsprozessmanagement. In diesem Abschnitt wird zunächst der Begriff der Methode diskutiert. Zusätzlich werden die dieser Arbeit zugrundeliegenden Aspekte des Methodendesigns dargestellt. Danach wird die Methode BPM(N)^{Easy} mit ihren Bestandteilen – Vorgehensweise, Sprache, Werkzeug – beschrieben.

Es existieren eine Vielzahl von agilen Methoden in der Literatur (vgl. [DD08][Di12][Hu14]), wobei das Verständnis des Methodenbegriffs nicht einheitlich ist. Beispielsweise definiert Goebels [GS97, S.18] eine Methode als „ein nicht auf eine einzige Situation zugeschnittene zielorientiertes Vorgehen, das mehrere Schritte umfasst und ein Ergebnis liefert.“ Die Definition weist auf den Aspekt der Wiederverwendbarkeit hin. Methoden bezeichnen somit nicht nur einen einmaligen Ablauf. Für Greiffenberg [Gr03, S.10] ist eine Methode im Allgemeinen „(...) eine planmäßige Art und Weise des Handelns mit überprüfbaren Ergebnissen.“ Greiffenberg beschreibt zusätzlich die Aspekte der Anleitung, Zielorientierung und Systematik als zentrale Elemente einer Methode. Lindemann [Li05, S.48] fasst eine Methode wie folgt zusammen: „(...) die Beschreibung eines regelbasierten und planmäßigen Vorgehens (...) nach dessen Vorgabe bestimmte Tätigkeiten auszuführen sind. Methoden sind präskriptiv, das heißt

als eine Vorschrift zu verstehen. Sie sind zielorientiert, als auf die Lösung eines Problems oder einer Aufgabenstellung fokussiert.“. Insgesamt verbindet die unterschiedlichen Definitionen der Fokus auf Planung und Zielerreichung im Kontext einer vorgegebenen Anforderung. Ortner [Or12, S.130] erweitert die „Vorgehensweise-fokussierte“ Sichtweise um zwei weitere Aspekte: Sprache und Werkzeug, vgl. Abbildung 42.

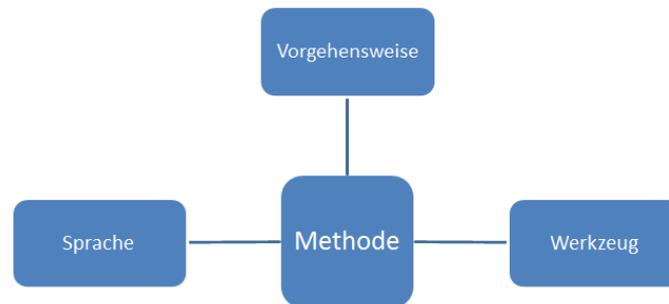


Abbildung 42: Methodendefinition nach Ortner [Or12]

Durch die Vorgehensweise werden Handlungs- oder Operationszusammenhänge beschrieben. Mit dem expliziten Einbezug einer Sprache wird sichergestellt, dass Erläuterungen innerhalb der Vorgehensweise kommuniziert werden können (vgl. [Or05, S.230f]). Des Weiteren sollen zur Unterstützung der Anwendung von Vorgehensweisen und Sprachen Werkzeuge (z.B. Softwarewerkzeuge zur Modellierung von Geschäftsprozessen) eingesetzt werden. Die Definition von Ortner ist die Grundlage der in dieser Arbeit entwickelten Methode. Bevor die drei Bestandteile in den Abschnitten 4.2, 4.3 und 4.4 beschrieben werden, muss neben dem Verständnis des Methodenbegriffs, ein Rahmen zum Design einer Methode definiert werden. Insbesondere die Anforderungen an eine neue Methode müssen beschrieben werden. Treppner [Tr12, S.16ff] definiert auf Basis der Ansätze von Cockburn [Co03] und Highsmith [Hi02a] einen Fragenkatalog im Kontext des Methodendesigns. Dieser Fragenkatalog kann als Grundlage der Entwicklung von agilen Methoden verwendet werden. Zusätzlich ist ein Vergleich von unterschiedlichen Methoden auf Basis des Fragebogens möglich. Der Fragenkatalog besteht aus neun Fragen und ist in Tabelle 21 dargestellt.

Tabelle 21: Fragenkatalog nach Treppler und Cockburn & Highsmith

1	Existiert für die betrachtete Methode eine Domänenabgrenzung?
2	Sind die Methodenelemente und Praktiken klar definiert?
3	Werden die geforderten Designprinzipien erfüllt?
4	Existieren ein phasenbezogenes Rahmenwerk, Projekt Templates und Entwicklungsszenarien?
5	Enthält die Methode Werte und Prinzipien? Sind Projektziele und Projektcharakteristika definiert?
6	Existieren Prozesse zur Methodenanpassung?
7	Existieren Richtlinien für die Übertragung auf große Projekte?
8	Berücksichtigt die Methode relevante Soft Factors?
9	Ist die Methode ausreichend flexibel?

Durch die Frage (1) wird überprüft, inwiefern eine Methode abgegrenzt werden kann. Methoden sollen hierbei keine unnötigen Bestandteile besitzen und auf eine bestimmte Aufgabe fokussieren. Beispielsweise können Methoden aus der Softwareentwicklung, aufgrund der vielen Parallelen zu GPM, ggfs. für die Entwicklung von Geschäftsprozessen angewendet werden, für andere Disziplinen sind diese dagegen nicht direkt adaptierbar. Die Frage (2) kann in zwei Teile gesplittet werden: Methodenelemente und Praktiken. Mit Methodenelemente werden die einzelnen Bestandteile der Methode beschrieben. Des Weiteren wird die Definition von Praktiken in Bezug auf Kollaboration, Projektmanagement und Entwicklung gefordert. Zum Beispiel müssen Methoden nach Cockburn [Co03] dreizehn Bestandteile (Prozesse, Meilensteine, Aktivitäten, Teams, Produkte, Praktiken, Rollen, Standards, Werkzeuge, Qualität, Fertigkeiten, Persönlichkeit, Teamwerte) berücksichtigen. Unter der Erfüllung von Designprinzipien (Frage (3) und Frage (5)) wird die Integration von bestehenden Prinzipien, wie beispielsweise die des Agilen Manifests [Be01], verstanden. Zudem müssen typische Projektziele und Projektcharakteristika (z.B. typische Teamgröße, Unsicherheiten,...) durch die Methode erläutert werden. Daran anschließend wird eine weitere Anforderung

an eine Methode durch Frage (4) formuliert. Die Beschreibung einer Methode soll demnach in einer abstrakten Form erfolgen, ohne den Rahmen der Methode unnötig zu begrenzen. Zudem sollen Templates und Beispiele durch die Methodenbeschreibung mitformuliert werden, um die Auswahl (d.h. die Entscheidung, ob die Methode zur aktuellen Aufgabenstellung passt) zu unterstützen. Die Fragen (6), (7) und (9) dienen zur Überprüfung der Anpassbarkeit und Übertragbarkeit der Methode. Die Methode muss eine flexible Anpassung (z.B. Anpassung der Iterationsdauer), sowohl vor, als auch während der Anwendung zulassen. Dies kann zum Beispiel durch domänenspezifische Templates erreicht werden. Die Frage (8) zielt auf die Fähigkeit des Umgangs mit „Soft Factors“ ab. Als „Soft Factor“ kann beispielsweise die Fähigkeit bezeichnet werden, eine Konfliktlösung (z.B. Kommunikationsprobleme, Streit,...) innerhalb des Projektteams zu unterstützen. Im Gegensatz zu „Hard Factors“ (z.B. Anzahl von Fehlern in einer Software, Projektkosten,...) widmen sich „Soft Factors“ den nicht „einfach“ zu messenden Kriterien (vgl. Abschnitt 4.2.4).

Im Folgenden werden die Kriterien des Fragenkatalogs im Rahmen der zu entwickelten Vorgehensweise berücksichtigt. Die weiteren Bestandteile einer Methode (Sprache, Werkzeug) dienen hierbei zur Unterstützung der Vorgehensweise.

4.2. Entwicklung einer Vorgehensweise zum agilen GPM (Z1)

„Die Vorgehensweise ist im Methodischen Konstruktivismus¹⁴⁷ bekanntlich durch die Attribute >>praktisch<<, >>schrittweise<<, >>zirkelfrei<<, >>explizit<<, >>terminiert<< und >>fortlaufend<< charakterisiert“ [Or12, S.130]. Die in diesem Abschnitt dargestellte Vorgehensweise BPM(N)^{Easy} folgt dieser Charakterisierung und adaptiert bzw. erweitert ausgewählte Aspekte existierender Vorgehensweisen des Geschäftsprozessmanagements und der agilen Softwareentwicklung. Die agile Vorgehensweise BPM(N)^{Easy} ist iterativ und evolutionär¹⁴⁸. Den Problemstellungen aus der Praxis (Attribut >>praktisch<<) wird in BPM(N)^{Easy} durch eine hohe

¹⁴⁷ Vgl. Abschnitt 2.1.

¹⁴⁸ Vgl. im Gegensatz zu den iterativen, evolutionäre Vorgehensweisen, die traditionellen „plan-driven“ Vorgehensweisen [Wi07b]. Derartige Vorgehensweisen erfordern zu Beginn eine detaillierte Anforderungsanalyse. Auf Basis der erfassten Anforderungen wird anschließend eine Planung (z.B. Zeitplan, Kostenplan,...) zur Entwicklung erstellt. Beispiele für solche Vorgehensweisen sind zum einen das „Wasserfallmodell“, zum anderen das „V-Modell“ (vgl. [Vo03b][FS06][Wi07b]). Des Weiteren sei auf die im Abschnitt 2.2.2 beschriebenen traditionellen Ansätzen im Kontext des Geschäftsprozessmanagements verwiesen.

Anwenderorientierung und flexible Anwendbarkeit begegnet. Ist zum Beispiel die Automatisierung von Geschäftsprozessen das Ziel eines GPM-Projekts werden IT-Experten aktiv in die Vorgehensweise mit eingebunden. Werden im Gegensatz dazu „nur“ Geschäftsprozesse zur Dokumentation modelliert, kann der Schritt der Anreicherung (zur Automatisierung) übersprungen werden. Die Attribute >>schrittweise<< und >>zirkelfrei<< werden durch die Definition von logisch aufeinander folgenden Schritten in BPM(N)^{Easy} implementiert. Beispielsweise muss vor der Anreicherung von Geschäftsprozessmodellen eine erste Erfassung von Informationen vorausgegangen sein. Durch diesen schrittweisen Aufbau entstehen begründete Ergebnisse, d.h. Geschäftsprozessmodelle, welchen den Erwartungen aller Beteiligten bzw. der vorgegebenen Wirklichkeit entsprechen (soweit Rekonstruktion möglich).

Hierbei darf bei „(...) jedem Schritt nur auf solche Hilfsmittel zurückgegriffen werden (...), die entweder bei diesem Aufbau bereits erarbeitet wurden oder die uns aus der vorwissenschaftlichen Praxis bereits zur Verfügung stehen.“ [Zi91, S.81]. Des Weiteren sind die BPM(N)^{Easy}-Schritte zirkelfrei durchzuführen, d.h. ohne dass Begriffe verwendet werden, welche bisher nicht oder nur implizit definiert wurden. Die Attribute >>explizit<< und >>terminiert<< beschreiben, dass die Anforderungen und die damit verbundenen Aufgabenstellungen eindeutig definiert und (zeitlich) abgegrenzt sein müssen. Beispielsweise dienen sogenannte User Stories in BPM(N)^{Easy} zur kurzen und expliziten Beschreibung einer Anforderung, welche (durch das ex ante definierte Iterationsende terminiert) umgesetzt werden muss. Zusätzlich werden die Schritte (z.B. Erfassung eines Geschäftsprozessmodells) in BPM(N)^{Easy} >>fortlaufend<< in kurzen Abständen zyklisch durchlaufen. In BPM(N)^{Easy} wird hierbei ein Zyklus als Iteration bezeichnet. In kurzen regelmäßigen Abständen kann somit dynamisch auf Veränderungen (z.B. die Verantwortlichkeiten für Aktivitäten in einem Geschäftsprozess ändern sich) agil reagiert werden.

Des Weiteren werden in BPM(N)^{Easy} die Anker der gebrauchssprachlichen Modellierung¹⁴⁹ berücksichtigt: Interaktion, Synchronisation und Qualität. Die gebrauchssprachliche Modellierung bildet die Grundlage für eine signifikant verbesserte Kommunikation und Erhöhung des Verständnisses aller Beteiligten in einem GPM-

¹⁴⁹ Vgl. insbesondere Abschnitt 3.3.

Projekt. Die in der gebrauchssprachlichen Modellierung vorgegebenen Anker dienen als „Kernprinzipien“¹⁵⁰ von BPM(N)^{Easy} und beschreiben die unterschiedlichen Aspekte eines GPM-Projekts. Das Kernprinzip der Interaktion wird in den iterativen und inkrementellen Zyklen von BPM(N)^{Easy} umgesetzt und „gelebt“. Beispielsweise werden Geschäftsprozessmodelle in Teams erfasst. Menschseitige Fehlinterpretationen (z.B. Begriffsdefekte wie Synonyme oder Homonyme) werden durch die stattfindende Interaktion schneller erkannt und behoben. Das Kernprinzip der Synchronisation beschreibt die Notwendigkeit von „Fixpunkten“ zwischen den einzelnen Schritten.

Im Gegensatz zur reinen Interaktion ist es hierbei verpflichtend, dass alle Beteiligten über neue Informationen benachrichtigt werden. Beispielsweise können unterschiedliche Aufgaben während der Modellierung interaktiv ausgeführt werden (z.B. Modellierung von Teilprozessen je nach Fachwissen). Die Ergebnisse der Aufgaben werden danach synchronisiert. Eine einfache Überführung in andere Modellierungs- oder Fachsprachen ist aufgrund des gemeinsamen (synchronen) Geschäftsprozessverständnisses aller Beteiligten gewährleistet. Nur wenn Geschäftsprozessmodelle die wahrheitsgemäße Rekonstruktion der Wirklichkeit beinhalten, können diese zum erfolgreichen (agilen) Geschäftsprozessmanagement beitragen. Das dritte Kernprinzip spiegelt daher den Aspekt der Qualität wider. In BPM(N)^{Easy} werden vordefinierte Qualitätsmodelle während den vereinbarten Synchronisationspunkten angewendet. Die Qualitätsmodelle werden hierbei je nach Anforderung während einer BPM(N)^{Easy} Iteration angepasst.

Bevor eine weitere Beschreibung von BPM(N)^{Easy} vorgenommen wird, werden die einzelnen Bestandteile und Rollen der Vorgehensweise beschrieben. In Abbildung 43 ist die Vorgehensweise skizziert.

¹⁵⁰ Scott Ambler beschreibt die Definition von (Kern-)Prinzipien als ein notwendiges Mittel der Abgrenzung einer Vorgehensweise, vgl. Abschnitt aus “Agile Modeling” (AM) [Am02, S.37]: „(...)By failing to adopt one of the core principles or practices of AM, you reduce the method’s effectiveness. Yes, you can benefit by only adopting a portion of AM, but you likely won’t obtain dramatic improvements in your effectiveness because of the drop in synergy. In short, feel free to adopt whatever aspects of AM that you see fit, just please don’t claim that you’re doing AM when you’ve only partially adopted it (...)”.

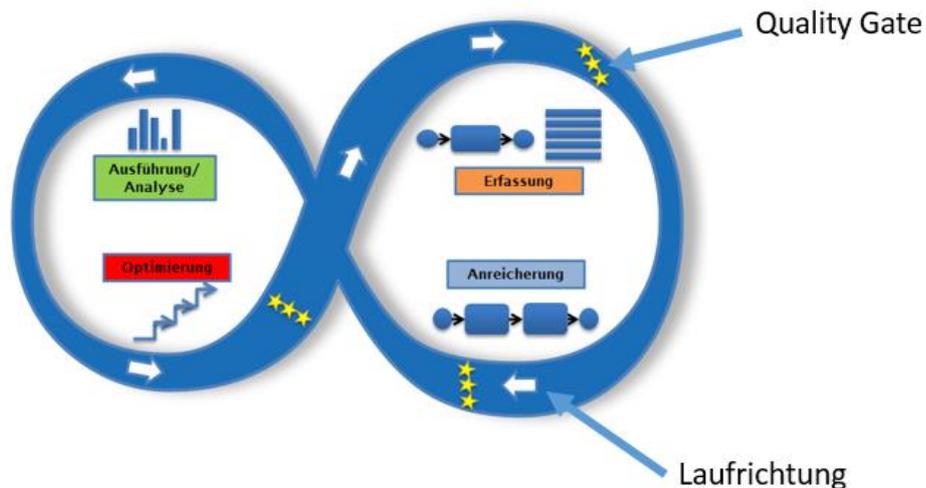


Abbildung 43: Darstellung der BPM(N)^{Easy} Vorgehensweise

BPM(N)^{Easy} besteht aus zwei Zyklen, wobei der Zyklus der Erfassung und Anreicherung (vgl. rechte Seite Abbildung 43) mit dem Zyklus der Ausführung/Analyse und Optimierung verbunden ist (vgl. linke Seite Abbildung 43). Muss zum Beispiel ein Geschäftsprozessmodell erfasst werden, welches die Angebotserstellung in einem Unternehmen darstellt, werden während des Schritts der Erfassung alle verfügbaren Informationen erfasst. Als Ergebnis dieser Erfassung wird ein erstes Geschäftsprozessmodell erhalten. Das Geschäftsprozessmodell kann mit weiteren Informationen verknüpft sein, zum Beispiel mit einer Videosequenz, welche die Bedienung des Customer-Relationship-Management-Systems (CRM-System) bei der Angebotserstellung zeigt. Im Schritt der Anreicherung werden die erfassten Informationen nach Bedarf in weitere Modellierungs- oder Programmiersprachen überführt. Beispielsweise kann daraus eine ausführliche Dokumentation des Geschäftsprozesses in Form eines BPMN 2.0-Modells erstellt werden. Während der Ausführung werden – im Beispiel der Angebotserstellung – Angebote an z.B. Kunden nach Vorgabe des modellierten Geschäftsprozesses erstellt. Mögliche Analyseergebnisse (z.B. durch Überwachung der Durchlaufzeiten) und das Feedback der Ausführenden wird in BPM(N)^{Easy} innerhalb des zweiten Zyklus erfasst. Innerhalb der Optimierung werden diese Informationen gemeinsam mit den Beteiligten bewertet und bei Bedarf für die anschließende nächste Erfassungs –und Anreicherungsiteration priorisiert. Das fortlaufende Durchführen von BPM(N)^{Easy} ermöglicht eine durchgehende Dokumentation von Anforderungen, die Sicherung eines gemeinsamen Verständnisses und die Erhaltung von Verbesserungsideen hinsichtlich der Geschäftsprozesse. Die

„Durchführungsrichtung“ von BPM(N)^{Easy} (vgl. weiß-markierte gerichtete Pfeile in Abbildung 43) ist im Fall der Erfassung und Anreicherung im Uhrzeigersinn, während der Zyklus der Ausführung/Analyse und Optimierung gegen den Uhrzeigersinn läuft. Während des Durchlaufens der Zyklen müssen sogenannte Quality Gates (vgl. [Sa08][GMW14a]) durchlaufen werden (vgl. „Sternsymbole“ Abbildung 43). Diese dienen zur Sicherstellung der Qualität (z.B. „entsprechen die Geschäftsprozessmodelle den vordefinierten Anforderungen?“) und zur Synchronisation aller Beteiligten. Die Anzahl der Durchläufe wird nicht durch BPM(N)^{Easy} bestimmt, sondern ist von den Anforderungen an das BPM Projekt abhängig (z.B. Komplexität der zu erfassenden Geschäftsprozesse).

BPM(N)^{Easy} erweitert und adaptiert hierbei mehrere Aspekte existierender Ansätze (vgl. Abschnitte 2.2.2 und 2.3). Zum einen verfolgt BPM(N)^{Easy} das Ziel einer ganzheitlichen Unterstützung aller Prozessbeteiligten, d.h. BPM(N)^{Easy} beschreibt eine Vorgehensweise, welche von der Erfassung bis hin zur Ausführung und Optimierung von Geschäftsprozessen reicht. Bestehende Methoden decken meist nur einen ausgewählten Teil des Geschäftsprozessmanagementzyklus ab (z.B. die fachliche Modellierung von Geschäftsprozessen). Zum anderen fokussiert BPM(N)^{Easy} auf die konsequente Einbindung des Menschen. Die Interaktion und Synchronisation aller Prozessbeteiligten steht somit im Vordergrund. Bestehende Methoden gewichten andere Faktoren (z.B. Geschäftsprozessmodelle) höher, wodurch das gegenseitige Verständnis aller, insbesondere in Bezug auf ereignisintensive Geschäftsprozesse, vernachlässigt wird. Zudem enthält BPM(N)^{Easy} eines auf die Anforderungen des agilen Geschäftsprozessmanagements adaptiertes Rollenkonzept (vgl. Abschnitt 4.2.2). Dieses Rollenkonzept beschreibt unter anderem einen BPM(N)^{Easy}-Master, welcher explizit als Bindeglied aller Prozessbeteiligten fungiert und die agile Vorgehensweise steuert. Bestehende agile Methoden hingegen übernehmen meist „nur“ die traditionelle Rollen des Geschäftsprozessmanagements oder der Softwareentwicklung. Des Weiteren stellt BPM(N)^{Easy} die intuitive Anwendbarkeit in den Fokus, wobei durch die gebrauchssprachliche Modellierung (vgl. Abschnitt 3) das Verständnis aller Prozessbeteiligten in Bezug auf die erfassten Geschäftsprozessmodelle von signifikanter Bedeutung ist. Existierende Methoden beschreiben hierbei meist entweder keinen konkreten Modellierungsansatz oder verweisen auf komplexe Modellierungssprachen, welche wiederum zu Problemen (z.B. Missverständnisse bei der Modellierung) führen (vgl. Abschnitt 1.1).

4.2.1. Start und Iteration

Dieser Abschnitt beschreibt wie die BPM(N)^{Easy} Vorgehensweise unter verschiedenen Rahmenbedingungen gestartet werden kann. Des Weiteren wird auf Basis existierender Ansätze der Aspekt der Iterationsdauer diskutiert.

- Start (und Ende)

BPM(N)^{Easy} beschreibt eine agile Vorgehensweise zur Unterstützung des operativen GPM. Mit operativem GPM wird die konkrete Gestaltung und Ausführung von Geschäftsprozessen beschrieben. Grundvoraussetzung ist folglich die ex ante definierte Organisationsstrategie in Bezug auf das Management von Geschäftsprozessen (vgl. [Sc04a][Al05][KF11b][We12]). Zudem erfordert die Anwendung einer agilen Vorgehensweise ein gemeinsames Verständnis aller Beteiligten bzgl. agiler Werte.

Beispielsweise ist ein hohes gegenseitiges Vertrauen zwischen den Projektbeteiligten eine Grundvoraussetzung. Nur so wird eine intensive Interaktion in einer Organisation zugelassen [Za12][Br11a][Le14] (vgl. Abschnitt 2.3.1). Sind die organisatorischen Rahmenbedingungen definiert, ist der Zeitpunkt „wann“ BPM(N)^{Easy} gestartet wird sehr flexibel wählbar und zu jedem einzelnen Schritt (Erfassung, Anreicherung, Ausführung/Analyse, Optimierung) möglich. Hierbei gilt der aus der Kanban Vorgehensweise entstandene Grundsatz „Start with what you do now“ [An14] (vgl. Abschnitt 2.3.4). Werden Geschäftsprozesse bereits ausgeführt, können Rückmeldungen (z.B. User Feedback, Datenanalysen...) zunächst erfasst werden. Die Rückmeldungen, welche durch die Diskussion aller Beteiligten zur Optimierung der Geschäftsprozesse ausgewählt wurden, bilden die Grundlage des nächsten Erfassungsschritts. Der Start der Erfassung bzw. Anpassung der Geschäftsprozessmodelle ist mit der Detailplanung der Anforderungsumsetzung verbunden (vgl. Abschnitt 4.2.3). Sind bereits Geschäftsprozessmodelle oder Anforderungen definiert, ist beispielsweise der direkte Start zur Anreicherung der BPM(N)^{Easy} zu (ausführbaren) BPMN 2.0 Geschäftsprozessmodellen möglich. Im Allgemeinen wird zu Beginn eines BPM(N)^{Easy}-Projekts ein sogenannter „Exploratory 360°“ („Rundblick“) durchgeführt (vgl. [Co05][GPR06]). Der „Exploratory 360°“ dient zur Erkundung von Rahmenbedingungen des Projekts. Beispielsweise wird die mögliche Teamzusammensetzung diskutiert, technische Machbarkeiten geprüft und erste Kosten-Nutzen-Rechnungen durchgeführt.

Die Form der grafischen Darstellung von BPM(N)^{Easy} (vgl. Abbildung 43) wurde analog zu dem aus der Mathematik bekannten Unendlichzeichen (∞) gewählt. Ein Ende des BPM(N)^{Easy} Zyklus ist somit nicht definiert, vielmehr wird auf die kontinuierliche Verbesserung von Iteration zu Iteration verwiesen. Eine Iteration beginnt und endet am zuvor gewählten Startzeitpunkt.

- Iteration

Bevor eine Iteration durchgeführt werden kann, muss zusätzlich deren Dauer und Inhalt bestimmt werden. Beispielsweise kann durch die Projektbeteiligten definiert werden, dass eine Iteration die Dauer von vier Wochen besitzt (Dauer). Ebenso wird definiert, dass bis zum Ende dieser vier Wochen eine erste Version eines automatisierten Geschäftsprozesses entstanden sein soll (Inhalt).

Agile Vorgehensweisen zeichnen sich durch relativ kurze Iterationslängen aus [Wi07b]. Beispielsweise definiert Beck [Be00][BA04] (vgl. Abschnitt 2.3.3) sogenannte „Weekly“ und „Quarterly“-Zyklen. Das „Weekly“ bezeichnet eine wöchentliche Planung von neu umzusetzenden Anforderungen. Des Weiteren werden die bestehenden Anforderungen verfeinert und der Implementierungsaufwand geschätzt. Das „Quarterly“ wird alle drei Monate durchgeführt und dient zur ausführlichen Reflexion (z.B. des Fortschritts, Projektstatus,...) und längerfristigen Planung. Cockburn [Co05] definiert die Länge einer Iteration zwischen einer Woche und zwei Monaten. Neben der Planung einer Iteration und täglichen kurzen Abstimmung (Daily), beschreibt Cockburn das Ende einer Iteration als „Completion Ritual“. Innerhalb dieser Abnahme werden reflektierende Fragen gestellt (z.B. „Was waren die Stärken und Schwächen während der Iteration?“), mit dem Ziel Rückschlüsse zur Optimierung der nachfolgenden Iteration zu erhalten [Co05] (vgl. Abschnitt 2.3.5). Scrum [SS13] (vgl. Abschnitt 2.3.2) definiert die Iterationsdauer auf maximal vier Wochen und nennt diesen Abschnitt „Sprint“. Eine Änderung der Sprintlänge während des Projekts soll hierbei vermieden werden. Zusätzlich dienen Dailys zur täglichen Abstimmung, in denen zum Beispiel ex ante formulierte Anforderungen und deren subjektive Interpretation bei der Modellierung bzw. Implementierung durch die Beteiligten diskutiert werden. Insbesondere Änderungen die das Sprint-Ziel gefährden, dürfen während der Iteration nicht vorgenommen werden. Dyer et al. [Za12] differenzieren die Iterationslänge auf Basis der Erfahrung von agilen GPM-Projektteams. Projektteams, welche noch keine Erfahrung mit agilen

Vorgehensweisen besitzen, wird eine Iterationsdauer von ein bis zwei Wochen empfohlen. Erfahrene Projektteams hingegen können eine Iterationsdauer von zwei bis drei Wochen zur Umsetzung der Anforderungen bestimmen. Die maximale Iterationsdauer nach Dyer et al. ist vier Wochen. Nach Mevius et al. [MSW12] [MSW13][MW13] ist eine Iteration in BPM(N)^{Easy} zwei Wochen lang. Die Einhaltung der kurzen Iterationszyklen trägt zur signifikanten Steigerung der Interaktion bei. Des Weiteren finden neben den Synchronisationen am Start und Ende einer Iteration zusätzliche Synchronisationstreffen statt. Innerhalb dieser Synchronisationstreffen werden die vordefinierten Quality Gates durchlaufen (vgl. Abschnitt 4.2.4). Hierdurch wird sichergestellt, dass die Modellierung bzw. Implementierung mit den aktuellen Anforderungen übereinstimmt.

Das Ergebnis einer Iteration wird als Inkrement bezeichnet. Ein BPM(N)^{Easy}-Inkrement bildet immer die ex ante definierte Anforderung „End-to-End“ ab. End-to-End (vgl. [AB09, S.26][Pw11, S.29]) bedeutet, dass der umzusetzende Geschäftsprozess ganzheitlich von Startereignis bis Endereignis durchgehend modelliert bzw. implementiert wird. Hierdurch wird von Beginn der ersten Iteration an, eine größtmögliche Transparenz (z.B. bzgl. Interaktion, Ressourcen, IT-Systeme,...) gewährleistet. Welcher Inhalt eine Iteration enthält und wie einzelne Bestandteile umgesetzt werden, ist abhängig von einer Vielzahl von Parametern. Neben der festgelegten Dauer nehmen beispielsweise Teamgröße, Komplexität und die Priorisierung Einfluss auf die Planung. Die Abbildung 44 stellt ein Geschäftsprozessmodell dar, welches innerhalb von drei Iterationen zur Automatisierung angereichert werden soll.

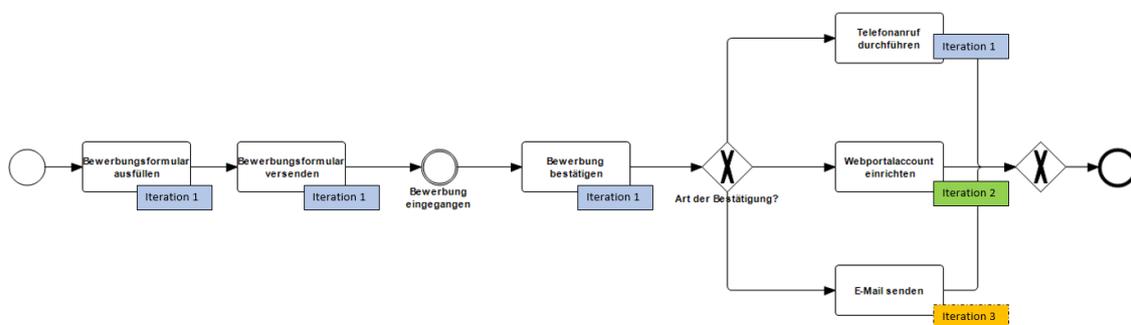


Abbildung 44: Beispiel einer Priorisierung in BPM(N)^{Easy}

Auf welche Aktivitäten in den einzelnen Iterationen fokussiert wird, ist durch die farbigen Rechtecke markiert. Zum Beispiel werden während der ersten Iteration alle Aktivitäten bis zum ersten XOR-Gateway implementiert (blaue Rechtecke „Iteration 1“). Zusätzlich dient die erste Iteration zur Sicherstellung, dass die ausführende Rolle „Telefonanrufe durchführen“ kann. Während der zweiten und dritten Iteration werden zum einen Änderungen (z.B. Fehlerkorrekturen) durchgeführt, zum anderen die Aktivitäten „Webportalaccount einrichten“ und „E-Mail senden“ implementiert. Mit Hilfe dieser iterativen und inkrementellen Vorgehensweise (vgl. [SS13]) können die Anwender sofort nach der ersten Iteration den Geschäftsprozess „leben“ (anwenden). Zudem kann durch die einfach verständliche Dokumentation auf Basis der BPMN^{Easy}-Modellierungssprache eine ganzheitliche Anwenderintegration sichergestellt werden. Bei Änderungen (z.B. neue Anforderungen), die während einer Iteration auftreten, muss individuell entschieden werden, ob diese aufgenommen werden können oder in einer der folgenden Iterationen zur Geltung kommen (vgl. [Za12]).

Je nach Anwendungsfall können in BPM(N)^{Easy} einzelne Schritte übersprungen werden. Wurde beispielsweise ein Geschäftsprozess für eine längerfristige Ausführung automatisiert, kann der Entwicklungszyklus „pausiert“ werden. Ziel ist hierbei, dass auch während einer „Entwicklungspause“ regelmäßig Interaktion und Synchronisation stattfinden. Die variierte Anwendung des BPM(N)^{Easy} Zyklus stellt sicher, dass während der Ausführung zu jeder Zeit Rückmeldung zur Geschäftsprozessoptimierung gegeben werden kann (vgl. Abschnitt 4.2.6). Die ex ante festgesetzten Synchronisationstreffen markieren dabei Start und Ende einer Iteration. Wird in einer Organisation bereits ein Ideen- und Innovationsmanagement betrieben, sollte dieses während der Anwendung von BPM(N)^{Easy} berücksichtigt werden. Durch das Ideen- und Innovationsmanagement wird unter anderem der Prozess der Ideengenerierung, -bewertung, -auswahl und -realisierung unterstützt (vgl. z.B. [Lä02][Be06][BE15]). Insbesondere Ideen, die indirekt positiven Einfluss auf die ausgeführten Geschäftsprozesse nehmen können (z.B. Idee zur Optimierung der Durchlaufzeit), werden somit besser erkannt. Liegt eine ausreichende Anzahl von neuen Anforderungen vor, kann die nächste Iteration wiederum ganzheitlich durchlaufen werden.

4.2.2. BPM(N)^{Easy} Rollen

In der Literatur existiert keine einheitliche Auffassung bzgl. der notwendigen Rollen und Rollenbezeichnungen im Geschäftsprozessmanagement (vgl. [LG09]). Je nach Perspektive (z.B. Fokus auf fachliche Modellierung oder technische Automatisierung) können Rollen unterschiedlich benannt und mit Verantwortungsbereichen bzw. Beteiligten verbunden sein. Beispielsweise kann die Rolle des Modellierers sowohl für die fachliche, als auch technische Modellierung verantwortlich sein oder nur für einen der Bereiche.

Welche Rollen in einer Organisation besetzt werden, muss abhängig von z.B. Organisationsgröße und Organisationsstruktur individuell definiert werden. Die Auswahl der Rollen muss hierbei eine interaktive Zusammenarbeit ermöglichen, welche die Akzeptanz von Änderungen fördert. Insbesondere in agilen Umgebungen muss sowohl die Bereitschaft, als auch Fähigkeit schnell und flexibel auf Änderungen reagieren zu können, gegeben sein. Zusätzlich ist eine interdisziplinäre Rollenverteilung ein entscheidender Erfolgsfaktor des GPMs (vgl. [RBW08]), wobei das „Kennen“, „Können“, „Wollen“ und „Sollen“ der Beteiligten berücksichtigt werden muss (vgl. Abbildung 45).

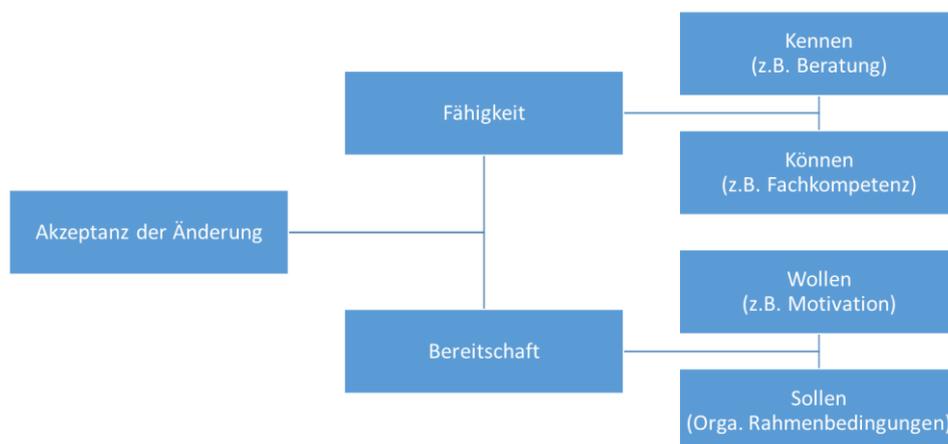


Abbildung 45: Akzeptanz von Änderungen (vgl. [Pa06 S.51ff])

Damit eine Änderung erfolgreich durchgeführt werden kann, müssen nach [RRH97] [Pa06 S.51ff] vier Bedingungen (Kennen – Können – Wollen – Sollen) erfüllt werden. Eine Änderung kann zum Beispiel die Teilautomatisierung von Aktivitäten innerhalb eines Geschäftsprozesses (z.B. von Antrag in Papierform zu Antrag als Web-Formular)

darstellen. Um Geschäftsprozessmanagement erfolgreich durchzuführen und somit eine kontinuierliche Änderung (Optimierung) zu ermöglichen, müssen dessen Rollen alle Akzeptanzbedingungen erfüllen. Organisatorische Rahmenbedingungen müssen das Geschäftsprozessmanagement unterstützen, indem beispielsweise neue Rollen geschaffen und besetzt werden (z.B. Chief Process Officer). Zudem müssen durch Anreize (z.B. durch eine Änderung wird die Arbeitszeit zu Gunsten der Mitarbeiter reduziert) alle Beteiligten zur Veränderung motiviert werden. Diese Motivation muss über den gesamten Änderungsprozess hinweg gewährleistet sein. Nach Pink [Pi11] entsteht die eigene Motivation¹⁵¹ insbesondere durch Autonomie, Fähigkeit und Sinnhaftigkeit¹⁵². Das agile Vorgehen und die damit kontinuierliche Interaktion, Synchronisation und Prüfung der Qualität (von Geschäftsprozessen) unterstützen hierbei diese Aspekte. Zum Beispiel werden Geschäftsprozesse „nur“ soweit erfasst und modelliert, wie es für die Anwender sinnvoll ist und deren Autonomie nicht einschränkt. Ebenso werden Fähigkeiten (z.B. Beherrschung einer Modellierungssprache) von Beginn an durch die gebrauchssprachliche Modellierung aufgebaut und kontinuierlich weiterentwickelt. Der Aspekt der „Fähigkeit“ (vgl. Abbildung 45) beschreibt die Anforderung, Fachkompetenz abzurufen bzw. aufzubauen. Des Weiteren ist es notwendig, Änderungen innerhalb der Organisation direkt an alle Beteiligten zu kommunizieren um deren Ziele zu manifestieren.

Neben den Rollenkonzepten der agilen Softwareentwicklung (vgl. Abschnitt 2.3) beschreibt Hanser [Ha08] vier Rollen: Kunde, Kommunikationsmanager, Integration Engineer und Team. Der Kunde gilt als Auftraggeber des Projekts und ist eng verbunden mit dem restlichen Projektteam. Die Projektüberwachung und Förderung der Kommunikation wird über den Kommunikationsmanager durchgeführt. Des Weiteren ist dieser für die Anpassung der angewendeten Vorgehensweise zuständig (z.B. um auf Qualitätsdefizite zu reagieren werden automatisierte Tests eingeführt). Der Integration Engineer verantwortet die technische Integration der einzelnen Inkremente.

Das Team verantwortet die Implementierung der Anforderungen. Im Kontext des Geschäftsprozessmanagements beschreibt Müller [Mü11] die Rollen des Chief Process Officers, Process Owners und Process Stakeholders. Der Chief Process Officer verantwortet hierbei die gesamte Geschäftsprozesslandschaft einer Organisation.

¹⁵¹ Für eine weitere Diskussion des Motivationsbegriff sei auf [SL09][Pi11][CR14] verwiesen.

¹⁵² Vgl. engl. Übersetzung: Autonomie - autonomy, Fähigkeit - mastery, Sinnhaftigkeit - purpose.

Unterhalb des CPOs sind die sogenannten Process Owner hierarchisch einzelnen Geschäftsprozessen zugeteilt und verantworten deren Management. Process Stakeholder sind Personen aus den beteiligten Fach- oder IT-Abteilungen. Insbesondere zur Berücksichtigung der realen Anforderungen und Verbesserungsvorschläge ist die Rolle des Process Stakeholders von signifikanter Bedeutung.

Gadatsch [Ga12b] gliedert die Rollen des GPMs in drei Ebenen auf: Strategieentwicklung, Prozessmanagement und Workflowmanagement. Zusätzlich unterscheidet Gadatsch zwischen der Prozessausführung und Prozessveränderung. Die ganzheitliche Verantwortung auf Ebene der Strategieentwicklung übernimmt der CPO (ggfs. mit Unterstützung von sogenannten Strategieberatern). Die Rollen Process Owner, Projektleiter, Prozessberater und Prozessmodellierer sind der zweiten Ebene zugeteilt. Insbesondere die operativen Aufgaben des GPMs (z.B. Modellierung von Geschäftsprozessen) werden durch diese Rollenbeteiligten übernommen. Auf der untersten Ebene, dem Workflowmanagement, unterstützen die Rollen des Prozessmitarbeiters, Workflowmodellierers und Software-Entwicklers die Ausführung und Implementierung von (automatisierten) Geschäftsprozessen. Zusätzlich dient die Rolle des Prozessauditors zur Überwachung von ausgeführten Geschäftsprozessen und Änderungsprojekten.

Kurz & Fleischmann beschreiben im Kontext des Subjektorientierten GPM-Ansatzes (vgl. [F110][F112][FS14]) unter anderem die Rollen des Process Contributors, Process Innovators, Modeling Specialist, Method Expert, Domain Expert, Process Owner, Mediator und Champion [KF11a, S.68ff]. Geschäftsprozessausführende Anwender werden durch die Process Contributor-Rolle beschrieben. Der Process Innovators besitzt (im Gegensatz zum Process Contributor) die Möglichkeit, Geschäftsprozessmodelle direkt zu modifizieren und informale zu formalen Informationen zu transferieren. Die Expertenrollen (Modeling Specialist, Method Expert und Domain Expert) verantworten die effektive und effiziente Umsetzung der Anforderungen und des Geschäftsprozessmanagements. Der Process Owner verantwortet die Geschäftsprozesse und entscheidet, welche Anforderungen bzw. Änderungen modelliert bzw. implementiert werden. Neben dem Mediator ist die Rolle des Champion für die Unterstützung des GPMs zuständig. Zum Beispiel ist die Lösung von Konflikten (z.B. Klärung von Interessenskonflikten) die Aufgabe des Mediators. Des Weiteren verbindet der Champion beispielsweise das Management mit den hierarchisch-unterliegenden Ebenen, um

gemeinsame Lösungen (z.B. zur Optimierung eines Geschäftsprozesses) erzielen zu können.

Basierend auf den existierenden Ansätzen des GPMs und der agilen Softwareentwicklung wird das in Abbildung 46 dargestellte BPM(N)^{Easy}-Rollenkonzept entwickelt.

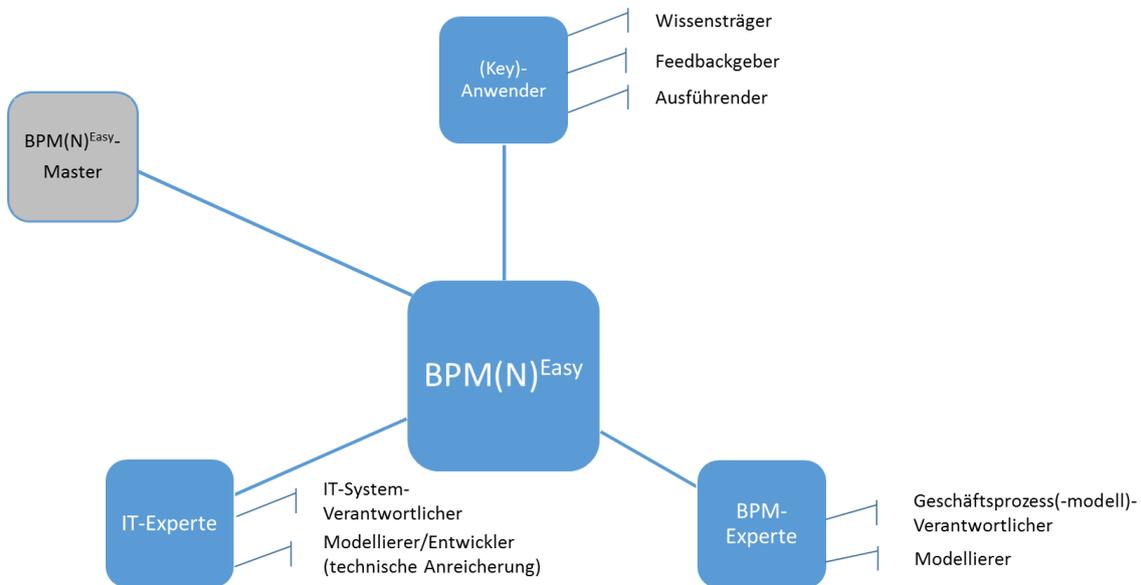


Abbildung 46: BPM(N)^{Easy} Rollenkonzept

Ein BPM(N)^{Easy}-Projektteam besteht aus vier Rollen: BPM-Experte, Anwender, IT-Experte und BPM(N)^{Easy}-Master. Jede Rolle muss in einem BPM(N)^{Easy}-Projekt vergeben sein, wobei eine beteiligte Person auch mehrfach Rollen übernehmen kann. Das Kernprojektteam sollte nicht mehr als sechs Beteiligte besitzen (vgl. Abschnitt 2.3.5). Die Verteilung der Rollen und die Größe des Projektteams gewährleisten eine strukturierte Abarbeitung der Anforderungen.

Der BPM-Experte verantwortet die fachliche Modellierung und Verwaltung von Geschäftsprozessen. Zusätzlich vermittelt der BPM-Experte Wissen (z.B. über die Anwendung von Modellierungssprachen) an andere Teammitglieder und besitzt methodisches Wissen zum Geschäftsprozessmanagement. Zur Sicherstellung der Geschäftsprozess(-modell)-qualität übernimmt der BPM-Experte zusätzlich die Aufgabe des „Product Owners“¹⁵³. Der (Key)-Anwender ist die Hauptrolle in BPM(N)^{Easy}. Als real Ausführender ist der Anwender, sowohl Wissensträger, als auch Feedbackgeber. Sind an

¹⁵³ Im Kontext GPM kann als Produkt der Geschäftsprozess bzw. das Geschäftsprozessmodell verstanden werden.

einem Geschäftsprozess zu viele Anwender beteiligt, können sogenannte Key-Anwender bestimmt werden. Unter einem Key-Anwender wird ein zentraler Ansprechpartner verstanden, wobei diese stellvertretend für ihre Bereiche bzw. Fachabteilungen stehen [Fe04, S.77f]. In kontinuierlicher Interaktion mit den anderen Rollen werden Anforderungen erfasst, implementiert und vor der Ausführung getestet.

Die Einbindung der IT wird über die Rolle des IT-Experten ermöglicht. Der IT-Experte ist sowohl für die Integration von Drittsystemen, als auch die Bereitstellung der Software zur Unterstützung des Geschäftsprozessmanagements verantwortlich. Des Weiteren übernimmt der IT-Experte die Modellierung und Anreicherung von technischen Geschäftsprozessen (Automatisierung). Der BPM(N)^{Easy}-Master bildet das Bindeglied zwischen allen Beteiligten. Die Rolle nimmt die Stellung eines „Coach“ ein (vgl. [BA04, S.143f]). Insbesondere fördert der BPM(N)^{Easy}-Master die für den Erfolg des Projekts notwendige (vgl. [RBW08]) Kommunikation zwischen IT und Fachabteilung. Zudem koordiniert der BPM(N)^{Easy}-Master die Teamtreffen (z.B. zur Synchronisation, Planung,...) und überwacht die Einhaltung der Rahmenbedingungen (z.B. Budget, Iterationsdauer,...).

4.2.3. Erfassung

Während der Erfassung werden Informationen zu einem ausgewählten Geschäftsprozess aufgenommen. Welche und wie die Informationen erfasst werden, ist von der ausgewählten Modellierungssprache und den vordefinierten Zielen des gesamten GPM-Projekts abhängig (vgl. [Me06]). Im Allgemeinen wird zur Erfassung von Geschäftsprozessen ein Top-Down -oder Bottom-Up-Ansatz¹⁵⁴ gewählt. Top-Down bedeutet in diesem Kontext, dass z.B. ausgehend von der Unternehmensstrategie Geschäftsprozesse identifiziert und modelliert werden. Im Gegensatz dazu beschreibt der Bottom-Up-Ansatz das Modellieren ausgehend von einzelnen Aktivitäten bis hin zu „kompletten“ Geschäftsprozessen und deren überliegenden Ebenen (vgl. [Sc04a, S.77f][FG08, S.130ff][KBL13, S.248ff][Ko10b, S.73ff]). Die Abgrenzung einzelner Geschäftsprozesse gegeneinander erfolgt unter Berücksichtigung verschiedener Aspekte.

¹⁵⁴ Des Weiteren ist eine Mischung von Top-Down und Bottom-Up möglich. Hierbei werden bestimmte Strukturen Top-Down vorgegeben und anschließend Bottom-Up detailliert, vgl. [KS97, S.173f][Sc98, S.187][Se15, S.31].

In [BM03] wird zwischen einer „allgemeinen und individuellen Identifikation“¹⁵⁵ von Geschäftsprozessen unterschieden. Bei der allgemeinen Identifikation liegt die Annahme zugrunde, dass allgemeingültige Geschäftsprozesse existieren, die in allen Unternehmen gleich sind (z.B. Strategieplanungsprozess, Liquiditätssicherungsprozess,...). Bei der individuellen Identifikation wird von unterschiedlichen Geschäftsprozessen ausgegangen, welche singulär erfasst werden müssen (z.B. aufgrund von unterschiedlichen Kundenbedürfnissen). Unabhängig davon kann nach [Bu13] beispielsweise eine Fachaufgabe (z.B. Erstellung eines Angebots) als „Grenze“ von Geschäftsprozessen dienen. Zudem können vorhandene Schnittstellen (z.B. Schnittstelle zu angrenzenden Drittsystemen) einen einzelnen Geschäftsprozess identifizieren. Zusätzlich muss der Wechsel von Verantwortlichkeiten berücksichtigt werden. Wechselt beispielsweise „(...) eine Fachaufgabe innerhalb eines Prozesses in die Zuständigkeit einer anderen Organisationseinheit (z.B. andere Fachabteilung) liegt zumindest eine logische Schnittstelle vor. In solchen logischen Schnittstellen kann eine sinnvolle Prozessgrenze liegen, die Abhängigkeit der folgenden Prozesse sollte aber in jedem Fall in die Betrachtung einbezogen werden.“ [Bu13, S. 11]. In [Kr97] wird der Begriff der Leistungstransaktionen eingeführt und zur Identifikation von Geschäftsprozessen und deren Schnittstellen genutzt¹⁵⁶. Die Schnittstellen-Perspektive ermöglicht eine stufenweise Detaillierung der Geschäftsprozesse und somit „(...) einen sukzessiven Einstieg in Modelle mit zunehmender Detaillierungstiefe, sodass komplexe Modelle leichter verstanden werden können.“ [Kr97, S. 98]. Die Abbildung 47 skizziert eine dreistufige Hierarchie. Die oberste Ebene (A.1 – C.1, vgl. Abbildung 47) beschreibt „grob“ den Ablauf des Geschäftsprozesses. „A.1“ wird mit Hilfe einer zweiten Ebene detaillierter dargestellt, wobei diese Ebene selbst wiederum auf Ebene drei (A3.1. – A.3.4, Abbildung 47) näher beschrieben wird. Der Informationsgehalt nimmt somit von Ebene zu Ebene zu [BK13b].

¹⁵⁵ In dieser Arbeit wird das Wort „Identifikation“ im Kontext der Erfassung von Geschäftsprozessen synonym zu dem Wort „Abgrenzung“ verwendet.

¹⁵⁶ Vgl. [Kr97, S.96ff].

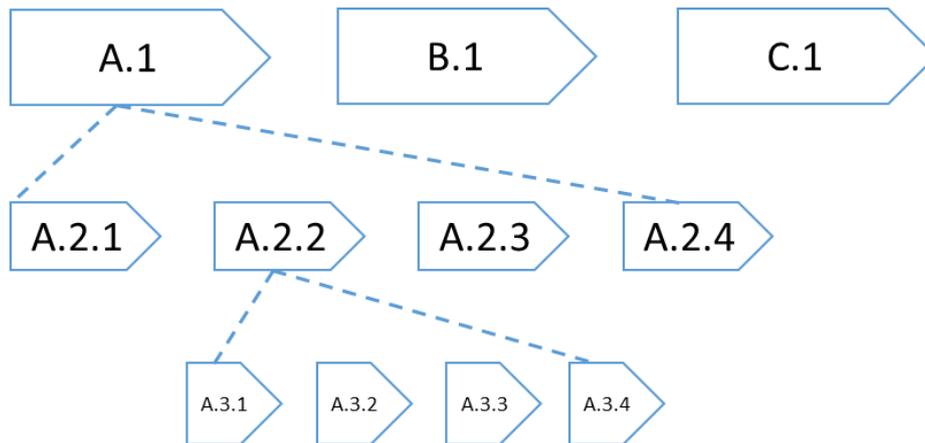


Abbildung 47: Beispiel - Hierarchieebenen von Geschäftsprozessen

Wird eine BPM(N)^{Easy}-Iteration durch die Erfassung eines Geschäftsprozesses gestartet, muss zuvor entschieden werden, welche Ebene(n) erfasst wird bzw. werden (vgl. Abbildung 47).

Als weiteres Hilfsmittel zur Bestimmung der zu modellierenden Ebene kann eine vertikale oder horizontale Aufteilung unternommen werden¹⁵⁷. Beispielsweise werden bei einer vertikalen Aufteilung die „direkt unterliegenden“ (Teil-)Geschäftsprozesse berücksichtigt (z.B. A.1 – A.2.1 – A.3.1, vgl. Abbildung 47). Bei einer horizontalen Aufteilung werden die (Teil-)Geschäftsprozesse „auf einer Ebene“ durchgängig berücksichtigt (z.B. A.1 – B.1 – C.1, Abbildung 47). Aufgrund der Anforderung von BPM(N)^{Easy}, am Ende jeder Iteration ein End-to-End-Geschäftsprozess als Ergebnis zu erhalten, wird auf die horizontale Aufteilung in dieser Arbeit fokussiert. Zum Beispiel genügt im Allgemeinen einem Kunden eine grobe Beschreibung wie ein bestelltes Paket geliefert wird (obere Ebene, vgl. Abbildung 47). Dagegen müssen aus Sicht des Paketdienstleisters die Aktivitäten innerhalb des Versand-Geschäftsprozesses detailliert dokumentiert sein (z.B. um eine effiziente Paketverteilung zu organisieren). Neben der reinen Erfassung muss der Verwendungszweck der Ergebnisse (z.B. Nutzung als technische Prozessdokumentation) bekannt sein. Ohne diese Strukturierung kann es zu Problemen, insbesondere bei der Festlegung von Detaillierungsgraden, kommen [KS97] [Ga12b]. Obwohl BPM(N)^{Easy} schrittweise und zirkelfrei durchlaufen wird – und somit unabhängig Vorwissen und Erfahrungen der Beteiligten genutzt werden kann - sollte die ausgewählte Modellierungssprache auf anerkannten Standards aufsetzen. Hierdurch wird

¹⁵⁷ Ein vergleichbares Vorgehen ist das „vertikales bzw. horizontales Prototyping“, welches in der Softwareentwicklung angewendet wird (vgl. [Bu92, S.39ff]).

„(...) eine einheitliche, überbetriebliche Modellierung, Implementierung und Kontrolle (...)“ der relevanten Informationen gefördert¹⁵⁸ [Me06, S.133]. Die Erfassung in BPM(N)^{Easy} wird in Anlehnung an Mevius [Me06, S.249ff] durchgeführt:

1. Ergebnisse und Auslöser identifizieren

Nachdem definiert wurde, welche Ebene und welcher (Teil eines) Geschäftsprozess (vgl. Abbildung 47) erfasst werden muss, müssen die Ergebnisse und Auslöser des Geschäftsprozesses präzisiert werden. Durch die Definition der Ergebnisse eines Geschäftsprozesses wird bestimmt, welchen Output¹⁵⁹ der zu erfassende Geschäftsprozess hat. Zum Beispiel kann dieser Output in Form von hergestellten Produkten oder durchgeführten Leistungen vorliegen. „Negative“ Ergebnisse, wie beispielsweise Absagen und Ablehnungen, können außerdem ein Ergebnis darstellen. Das retrograde Vorgehen – d.h. die rückwärtsgerichtete Erfassung eines Geschäftsprozesses (vom Ergebnis startend) – ermöglicht eine direkte Zielformulierung, welche den gesamten Geschäftsprozess beeinflusst. Beispielsweise können Aktivitäten, die nicht wertschöpfend zum Ergebnis beitragen, schnell erkannt und entfernt werden (vgl. [BM03][Me06]). Im zweiten Schritt wird der Auslöser eines Geschäftsprozesses definiert. Auslöser können die Ergebnisse anderer Geschäftsprozesse sein. Beispielsweise wird der Warenversand erst nach der Produktion der Ware getriggert. Externe Trigger, wie beispielsweise eine Kundenanfrage, können zudem als Auslöser definiert werden.

Neben der Definition von Ergebnis und Auslöser müssen die an dem GPM-Projekt beteiligten Personen eine Begriffsklärung¹⁶⁰ durchführen, um unter anderem Missverständnisse bei der Verwendung von Begriffen vorzubeugen. Die während einer Interaktion der Beteiligten geäußerten Sprechakte bilden hierbei die Grundlage. Die in Abbildung 48 dargestellte Aussagentransformation wird in BPM(N)^{Easy} iterativ durchgeführt. Nur wenn erfasste Aussagen auf gelungenen Sprechakten basieren, kann die weitere Normierung (vgl. Abbildung 48) erfolgreich durchgeführt werden.

¹⁵⁸ Nach Prilla & Nolte [PN12] haben insbesondere „Nicht-BPM-Experten“ Probleme (grafische) Modellierungssprachen anzuwenden. Durch den Einsatz von etablierten und bekannten Standards – und somit der Nutzung von einem generellen Vorwissen - besteht die Möglichkeit diese zu reduzieren.

¹⁵⁹ Vgl. Definition des Begriffs Geschäftsprozess in Abschnitt 2.2.1.

¹⁶⁰ Vgl. Abschnitt 2.1.3.

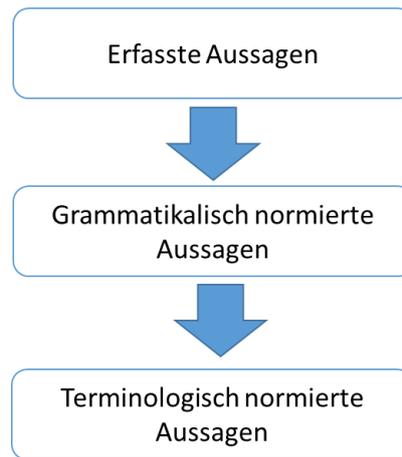


Abbildung 48: Normierung von erfassten Aussagen¹⁶¹

Anstatt eine vollständige Begriffsnormierung am Anfang jeder Erfassung von Geschäftsprozessmodellen durchzuführen, wird der gemeinsame Konsens aller Beteiligten schrittweise gebildet und fortlaufend verbessert. Aussagen, welche erfasst wurden (z.B. „Die Aktivität X wird in dem Geschäftsprozess Y ausgeführt, indem das Formular Z ausgefüllt wird.“), werden danach grammatikalisch normiert. Hierdurch wird vermieden, dass aufgrund von grammatikalischen Umformungen unbeabsichtigte Aussageveränderungen vorgenommen werden. Die abschließende terminologische Normierung dient zum Aufbau einer gemeinsam verwendeten Fachsprache. Das dabei entstehende „Fachwörterbuch“ kann als Grundlage zur Prüfung der korrekten Verwendung von Begriffen eingesetzt werden.

2. „Schönwetterflug“

Bei der Identifikation von Geschäftsprozessen handelt es sich um kreative und innovative Handlungen. Mit Hilfe dieser Handlungen (z.B. Diskussion der Beteiligten über das Ergebnis eines Geschäftsprozesses) werden aus Aktivitäten strukturierte Geschäftsprozesse „geformt“ [BM03, S.116].

¹⁶¹ Die grammatikalische Normierung wird in [Le99] in acht Schritte unterteilt (Substitution von Pronomen durch die entsprechenden Substantive – Zerlegung komplexer Aussagen – Ersetzung mehrdeutiger Formulierungen – Umformung passiv formulierter Aussagen – Substantivumformung - Verbumformung – Satzlogik prüfen – Entfernung von redundanten Aussagen), die terminologische Normierung in fünf weitere (Selektion der Fachbegriffskandidaten – Normierung der Flexion – Anwendung einer Stoppwortliste – Rekonstruktion der Fachbegriffskandidaten – Ergänzung der Stoppwortliste). Für eine ausführliche Beschreibung der Aussagensammlung sei auf [Sc97][Or97][Le99] verwiesen.

Diese Geschäftsprozesse werden bei der ersten Erfassung (erste Iteration von BPM(N)^{Easy}) als sogenannter Schönwetterflug¹⁶² abgebildet. Die Konzentration auf den „normalen“ Fluss des Geschäftsprozesses unterstützt die strukturierte Koordination der Handlungen. Die Erfassung des Schönwetterflugs ermöglicht eine einfache und schnelle Modellierung ohne zusätzliche (unnötige) Komplexität wie beispielsweise Ausnahmen, die sehr selten eintreten (z.B. Antragsformular wird durch einen Experten geprüft). Abbildung 49 stellt einen Antragsprozess im Schönwetterflug dar.

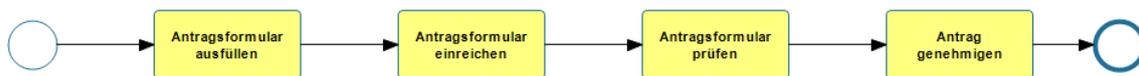


Abbildung 49: Beispiel eines Schönwetterflug-Geschäftsprozessmodells

Die zu erfassenden Aktivitäten und deren Sequenz (z.B. der Antrag wird zuerst geprüft und danach genehmigt, vgl. Abbildung 49) werden in BPM(N)^{Easy} durch alle für den Geschäftsprozess relevanten Rollen (z.B. IT-Experte, BPM-Experte und Anwender) bestimmt¹⁶³. Hierbei können unterschiedliche Erfassungstechniken angewendet werden, um zu einem Geschäftsprozessmodell zu gelangen. Welche Erfassungstechnik angewendet wird, hängt von mehreren Parametern ab. Insbesondere das Vorhandensein von bestehenden Dokumentationen und das Vorwissen der Beteiligten beeinflusst hierbei die Auswahl und Anwendung der Erfassungstechnik(en). Die existierende Literatur¹⁶⁴ listet eine Vielzahl von Erfassungstechniken auf, wobei Lehmann [Le99] in vier Klassen aufgeteilt: Selbstaufschreibung, Beobachtung, Schriftgutuntersuchung und Befragung. Während einer (automatischen) Selbstbeschreibung werden Informationen arbeitsbegleitend erfasst. Diese Erfassung kann entweder durch Menschen oder durch eine Maschine erfolgen. Die Erfassung durch Beobachtung beschreibt, dass wahrgenommene Sachverhalte während einer Ausführung protokolliert werden. Die Beobachtung kann hierbei über eine vordefinierte Zeit oder als Stichprobe durchgeführt werden. Bei der Schriftgutuntersuchung werden bestehende Schriften (z.B. Richtlinien, Dokumentationen,...) analysiert. Die Befragung erfolgt zum Beispiel mit Hilfe eines Fragebogens oder Interviews. Hierbei werden Personen, die an dem Geschäftsprozess

¹⁶² Der Begriff des Schönwetterflugs [Me06] ist synonym mit dem Begriff des „Happy Path“ zu verwenden, vgl. [WM08, S.179] [Ku13, S.130f].

¹⁶³ Vgl. Abschnitt „Rollenkonzept von BPM(N)^{Easy}“.

¹⁶⁴ Vgl. Erfassungstechniken u.a.: Übersicht [Sc97, S.116] [FS99, S.64] [GL08] [Hä10]. Des Weiteren sei im Kontext agiler Vorgehensweisen auf Inayat et al. [In14a] verwiesen („A systematic literature review on agile requirements engineering practices and challenges“).

beteiligt sind, zu Sachverhalten im Kontext des Geschäftsprozesses befragt (z.B. „Wie lange dauert die Ausführung der Aktivität XYZ?“). Da jede der Erfassungstechniken auch Nachteile besitzt, werden die einzelnen Ansätze in der betrieblichen Anwendung oftmals in Kombination angewendet. Beispielsweise können Akzeptanzprobleme durch eine personelle Trennung (Interview-führender und Befragter) entstehen [Le00]. Die mit BPM(N)^{Easy} geforderte Interaktion aller Beteiligten trägt zum Abbau solcher Probleme bei, wobei die effiziente Sammlung aller relevanten Anforderungen im Vordergrund steht. Die ausreichend detaillierte Beschreibung von Anforderungen ist für die Aufwandsschätzung und die ggfs. nachfolgende Anreicherung notwendig. Die durch die angewendeten Erfassungstechniken explizierten Information werden in BPM(N)^{Easy} mit Hilfe eines grafischen Geschäftsprozessmodells und sogenannten Mediafiles (vgl. Abschnitt 4.2.3, Punkt 4) dokumentiert.

3. Sonderfälle ergänzen

Nach der Modellierung des Schönwetterflugs wird das Geschäftsprozessmodell um Ausnahmen ergänzt. In Abbildung 50 (vgl. rot-gefärbte Elemente) wurde der Geschäftsprozess der Antragstellung beispielhaft erweitert. Im Gegensatz zum Schönwetterflug wird nun die Information, dass ein Antrag auch fehlerhaft sein kann und ggfs. zudem von einem Experten begutachtet werden muss, hinzugefügt.

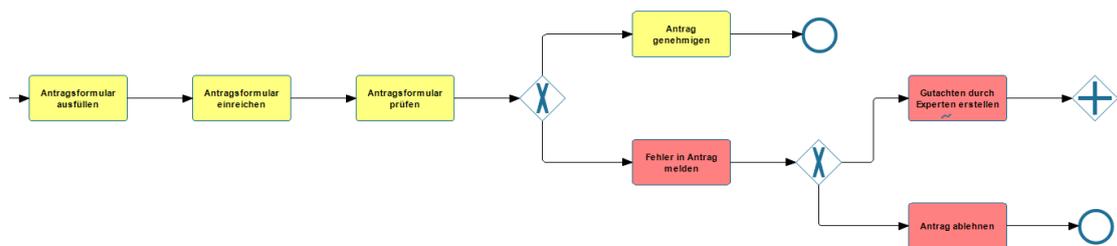


Abbildung 50: Beispiel Geschäftsprozessmodell mit Ausnahmen

Nach Kurz (vgl. [Ku13]) können Ausnahmen hierbei in vier Klassen eingeteilt werden: „Keine Ausnahmen“¹⁶⁵, „Routineausnahmen“, „geringfügige Ausnahmen“ und „schwere Ausnahmen“. Routineausnahmen sind Ausnahmen die bereits bei der Erfassung bekannt sind und somit bei der Modellierung berücksichtigt werden können, beispielsweise kann die Ausnahme bestehen, dass der Antrag ein fehlerhaftes Datum enthält. Falls die

¹⁶⁵ Zur Vollständigkeit ist „keine Ausnahmen“ mit aufgeführt, vgl. [Ku13, S.127] engl. Übersetzung: „Normal process execution“ – „Routine Exceptions“ – „Minor Exceptions“ – „Major Exceptions“.

Routineausnahme eintritt, kann diese auf Basis des Geschäftsprozessmodells behandelt werden. Im Gegensatz dazu sind „geringfügige Ausnahmen“ vorhersagbar, aber nicht im Vorhinein vollständig planbar. Die Behandlung einer solchen Ausnahme folgt zum Zeitpunkt des Auftretens einer individuellen Vorgehensweise (ggfs. auf Basis von grob granularen Geschäftsprozessmodellen). Beispielsweise können die Fehler in einem Antrag sehr unterschiedlich sein, sodass die Korrekturmaßnahmen nicht ex ante im Detail planbar sind. Je nach verwendeter Modellierungssprache können derart durchgeführte Aktivitäten unterschiedlich dargestellt werden. Zum Beispiel kann in BPMN^{Easy} das „Generic Task“-Element (vgl. Abschnitt 4.3.3) zur Darstellung einer nicht-planbaren Aktivität genutzt werden. Das Element dient hierbei „nur“ als Platzhalter, trägt aber zur Sensibilisierung aller Prozessbeteiligten bei, dass mögliche geringfügige Ausnahmen eintreten können. Hierbei enthält das „Generic Task“-Element eine Vorgabe „was“ bei Eintritt einer solchen Ausnahme getan werden muss. Eine weitere Klasse von Ausnahmen sind die „schweren Ausnahmen“. Diese Ausnahmen sind weder vorhersagbar noch planbar. Beispielsweise könnte eine unerwartete Gesetzesänderung den Antragsprozess während der Ausführung beeinflussen. Schwere Ausnahmen müssen aufgrund der Nicht-Planbarkeit zum Zeitpunkt des Auftretens behandelt werden, eine vordefinierte Behandlung ist nicht möglich (vgl. grau-markierter Quadrant in Abbildung 51). Die Abbildung 51 skizziert die vier Klassen von Ausnahmen nach [Ku13].

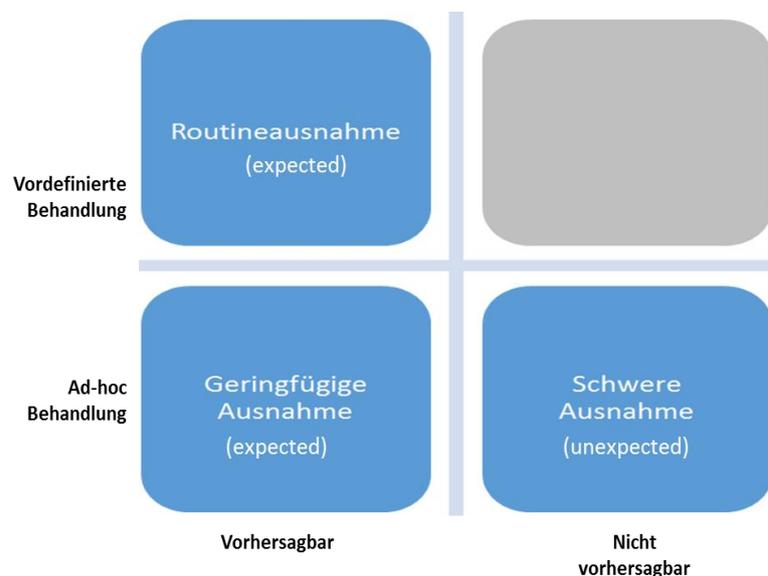


Abbildung 51: Ausnahmen unterteilt in vier Klassen nach [Ku13]

Des Weiteren wird auf Nehmer und Reuter [NR08] verwiesen, welche die Behandlung von Ausnahmen auf Basis eines „Exception Frameworks“ beschreiben. Unabhängig von

der Klassifizierung der Ausnahme sollte bei der Erfassung von Geschäftsprozessmodellen nicht auf jede „scheinbar“ mögliche Ausnahme eingegangen werden, sondern ausschließlich auf die, die den Geschäftsprozess wahrscheinlich und ausschlaggebend beeinflussen können [Fi06]¹⁶⁶. In BPM(N)^{Easy} werden im ersten Schritt zur gebrauchssprachlichen Erfassung (komplexe) Ausnahmen mit Hilfe von Multimediaannotationen beschrieben.

4. Mediafiles hinzufügen

Die Modellierungssprache BPMN^{Easy} ermöglicht das Hinzufügen von Multimedia¹⁶⁷. In BPM(N)^{Easy} werden die hinzugefügten multimedialen Informationen als „Mediafiles“ bezeichnet. Innerhalb des jeweiligen Mediafiles werden zur Übermittlung der Inhalte Zeichensysteme und Kodierungen verwendet. Kritzenberger et al. [KH04, S.61ff] unterscheiden hierbei zwischen Sprache (z.B. Vortrag hören), Schrift (z.B. textuelle Beschreibung eines Geschäftsprozesses), Grafik (z.B. Foto einer Maschine) und Musik. Insbesondere unterstützt die Nutzung von Mediafiles die Erfassung von wissensintensiven Aktivitäten (vgl. [Da15]). Anforderungen, die nicht (oder nur durch sehr komplexe Konstrukte) grafisch modelliert werden können, werden mit Hilfe von Mediafiles (z.B. Video) gebrauchssprachlich erfasst. Des Weiteren unterstützt Multimedia „(...) den fachübergreifenden Wissenstransfer durch anschauliche Präsentation von Wissen.“ [Ma99, S.233].

Die Tabelle 22 listet eine Auswahl von BPMN^{Easy}-Elementen auf (vgl. Abschnitt 4.3), die durch die Erfassung von Mediafiles zusätzlich angereichert werden können.

Tabelle 22: Beispiele zur Erfassung und Nutzung von Mediafiles

BPMN ^{Easy} -Element	Beispiel
Start, -End,- Zwischenereignis	Audio –und Video können genutzt werden, um gebrauchssprachlich Start,-End,-und Zwischenereignisse zu beschreiben.

¹⁶⁶ Vgl. engl. Übersetzung: „(...) In the short term, stick to the core process and the obvious exceptions rather than attempting to cater for every possible scenario (...)“ [Fi06, S.177].

¹⁶⁷ Der Multimediabegriff ist in der existierenden Literatur nicht einheitlich definiert. Im Kontext dieser Arbeit wird unter Multimedia die Integration mehrerer Text-,Bild- und/oder Tonmedien-Techniken mit Hilfe digitaler Technik verstanden (vgl. [LW97]). Zur detaillierten Diskussion des Begriffs Multimedia sei auf [Ha03][Be11a] verwiesen.

Manuelle Task	Insbesondere Bilder und Videos können zur Visualisierung von manuell auszuführenden Tasks dienen.
User Task	Mit Hilfe von z.B. Bildern und Hilfetexten kann die Ausführung von User Tasks unterstützt werden. Soll beispielsweise ein Web-Formular zur Beantragung einer Sache (z.B. Investitions, -oder Urlaubsantrag,...) ausgefüllt werden, kann ein Screenshot (Bild) eines beispielhaft ausgefüllten Antrags als Vorlage dienen.
XOR-und AND-Gateway	Insbesondere bei Gateways, die als Verzweigung von Entscheidungen dienen, können Mediafiles zur Beschreibung des Sachverhalts genutzt werden (anstatt z.B. die Kanten der Sequenzflüsse mit komplexen Ausdrücken zu annotieren, vgl. BPMN 2.0).

5. Erforderliche Rollen, Ressourcen und Personen ergänzen

Damit die erfassten Geschäftsprozesse real ausführbar sind, müssen an die modellierten Elemente zusätzliche Informationen zugewiesen werden. Die Konkretisierung, welche Rollen (z.B. Sachbearbeiter, Manager,...), Ressourcen (z.B. ERP-System, CRM-System,...) und Personen die Ausführung des Geschäftsprozesses übernehmen ist hierbei von besonderer Bedeutung, um Fehler (z.B. falsche Zuweisung einer Aufgabe) in der Ausführungszeit zu vermeiden. Die Tabelle 23 stellt die einzelnen Möglichkeiten einer Zuweisung (vgl. [Bu94]) dar.

Tabelle 23: Zuweisungsmöglichkeiten nach Bußler [Bu94]

Konzept	Beschreibung und Beispiel
Agent	Die Zuweisung erfolgt an einen Agenten z.B. an die Person Max Mustermann.
Rolle	Eine Aktivität kann einer Rolle „Manager“ zugewiesen werden, um zu definieren „wer“ die Aktivität ausführen soll.

	In diesem Fall wird die Aufgabe von einer beliebigen Person, welche die Rolle „Manager“ besitzt ausgeführt.
Klassifizierte Rolle	Die Zuweisung der Rolle basiert auf Basis des konkreten Agenten. Durch diese Klassifizierung kann beispielsweise sichergestellt werden, dass nur der für einen Mitarbeiter zuständige Manager für eine bestimmte Aufgabe zugewiesen wird.
Bedingte Zuweisung	Bedingungsabhängige Zuweisung einer Aufgabe - zum Beispiel darf ein Manager bis einschließlich 1000 Euro eine Investition genehmigen, der Manager des Managers hingegen Investitionen größer 1000 Euro.

Neben der Aufgabenzuweisung müssen bei Bedarf Ereignisse mit zusätzlichen Informationen angereichert werden. Muss zum Beispiel ein Geschäftsprozess automatisiert werden, ist es notwendig die Zugriffskontrolle (z.B. „wer“ ein Startereignis auslösen darf) konform zu den fachlichen und organisatorischen Anforderungen (z.B. Sicherheitsrichtlinien) zu implementieren (vgl. [KWE08]). Insbesondere die Organisationform beeinflusst hierbei die Auswahl der Rollen- und Ressourcen (z.B. Verfügbarkeit und Auslastung der Anwender). Anwender (Personen) in einer funktionalen Aufbauorganisation sind häufig höher spezialisiert, dafür nur für einen Bereich zuständig. In einer prozessorientierten Aufbauorganisation sind die Anwender über den gesamten Geschäftsprozess eingebunden [Wi07a]. Die Tabelle 24 listet mögliche Organisationformen auf:

Tabelle 24: Beispiele für Rollen und Ressourcen Organisation (vgl. [Sc04a])

Organisationsform	Beschreibung
Reine Funktionsorganisation	In einer funktionalen Organisation verantworten einzelne Rollen/Anwender spezialisierte Aufgabenbereiche (z.B. Vertrieb, Entwicklung,...). Eine (Abteilungs-)übergreifende Verantwortung ist nicht implementiert.

Matrix- Prozessorganisation	Geschäftsprozesse werden funktionsübergreifend in der Organisation „gelebt“. Gleichzeitig bleiben bestimmte Funktionen und Spezialisierungen erhalten.
Reine Prozessorganisation	Anwender sind ganzheitlich in die zu ausführenden Geschäftsprozesse integriert. Die Verantwortung bzgl. Funktionen wird aufgehoben und aus der Geschäftsprozessperspektive organisiert.

Die in Tabelle 24 dargestellten Organisationsformen beeinflussen die Erfassung von Geschäftsprozessen, sodass bei der Zusammenstellung eines BPM(N)^{Easy}-Projektteams auf die Verantwortungsbereiche und Kompetenzen aller Beteiligten geachtet werden muss. Zudem sind weitere, insbesondere für die Ausführungszeit relevanten, Fragen innerhalb der Erfassung zu diskutieren bzw. bereits zu beantworten (vgl. [Mu04]). Zum einen muss zwischen einer Rollen und personenbezogene Worklist unterschieden werden. Eine Worklist ist eine Liste mit Aufgaben, die durch eine Person ausgeführt werden sollen, wobei Aufgaben entweder für eine/mehrere Rolle(n) relevant sind (d.h. jede Person mit dieser Rolle kann diese Aufgabe ausführen) oder nur für eine/mehrere bestimmte Person(en). Zum anderen kann eine weitere Unterscheidung zwischen Rollen und personenbezogene Eigenschaften notwendig sein. Beispielweise können Personen eine gleiche Rolle besitzen (z.B. Rolle Sachbearbeiter), aber abhängig ihres Arbeitsstandorts (z.B. Unterscheidung nach Region, Stadt,...) können unterschiedliche Aufgaben relevant sein. Ebenso müssen Regeln zur Zuweisung von Aufgaben an Rollen und Personen definiert werden, die sich zum Beispiel zur Regelung in Bezug auf vorhergehende (auf die nachfolgende Aufgabe einflussnehmende) Aufgaben anwenden lassen.

6. Qualitätsprüfung

Die Qualität wird kontinuierlich während der Erfassung geprüft. Die kontinuierliche Kontrolle erfolgt im Rahmen einer Paarmodellierung (vgl. [BA04]). In Abbildung 52 werden die zwei möglichen Formen der Paarmodellierung dargestellt (in Anlehnung an [PN12]).

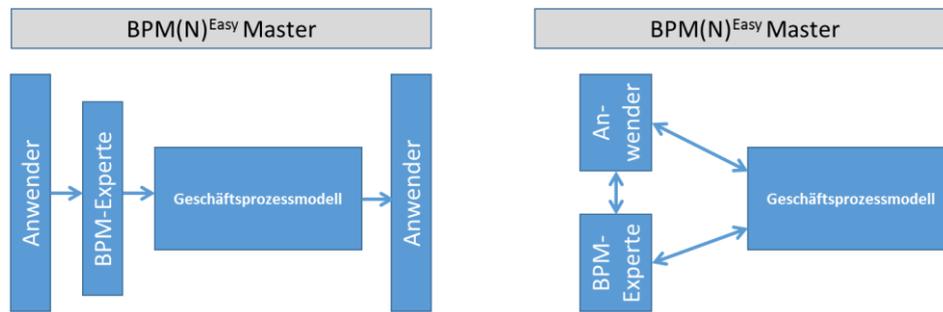


Abbildung 52: Varianten der Paarmodellierung

Die erste Variante der Paarmodellierung beschreibt den Ansatz, dass ausschließlich der BPM-Experte die Modellierung der Geschäftsprozesse durchführt (vgl. linke Seite, Abbildung 52). Anwender kommunizieren hierbei ihre Anforderungen und Wünsche, danach erfasst der BPM-Experte diese in einem Geschäftsprozessmodell. Die veranlasste Handlung wird wiederum durch den Anwender überwacht (vgl. Abschnitt 2.1.2). Beispielsweise können Fehler in der Modellierung somit direkt erkannt und behoben werden. Alternativ können Geschäftsprozesse „vollständig gemeinsam“ erstellt werden, indem sowohl Anwender als auch BPM-Experten ein Geschäftsprozessmodell erfassen (vgl. rechte Seite, Abbildung 52). Welche Variante der Paarmodellierung angewendet wird, hängt von individuellen Rahmenbedingungen ab (z.B. Vorwissen der Anwender, Teamgröße,...). Die abschließende Abnahme des Erfassungsergebnisses wird durch sogenannte Quality Gates sichergestellt.

4.2.4. Quality Gates

Agile Vorgehensweisen tragen (z.B. durch kurze Iterationen und hohe Interaktion der Beteiligten) sowohl zur Sicherung, als auch Steigerung der Qualität von Geschäftsprozessen(-modellen) bei. Dennoch existieren Herausforderungen die ein zusätzliches Qualitätsmanagement erfordern (vgl. [MAA13]). Beispielsweise ist es möglich, dass aufgrund von Zeitmangel die Interaktion der Beteiligten eines GPM-Projekts nur sehr reduziert stattfinden kann. Ein weiteres Problem tritt auf, wenn die Beteiligten in (geographisch) unterschiedlichen Standorten einer Organisation beschäftigt sind. Um derartigen Herausforderungen zu entgegnen werden in BPM(N)^{Easy} sogenannte Quality Gates definiert. Salger et al. [Sa08]¹⁶⁸ beschreiben den Begriff des Quality Gates als einen vordefinierten Zeitpunkt, an welchem erstellte Artefakte (und deren vorangegangenen Erstellungsprozesse) bewertet werden. Nach Zürn [Zü10, S.19] strukturieren Quality Gates Vorgehensweisen, „(...) indem die einzelnen Phasen durch Quality Gates getrennt werden. An diesen Toren überprüfen definierte Pförtner die Qualität der Ergebnisse oder auch die verwendeten Methoden hinsichtlich definierter Kriterien anhand von Checklisten.“. Ex ante definierte Anforderungen, die nicht eingehalten werden, sind somit sofort für alle Beteiligten transparent (vgl. [Mo10, S.66]). Damit eine Bewertung durchgeführt werden kann, müssen vordefinierte Qualitätskriterien existieren, mit welchen der IST-Zustand zum jeweiligen Messzeitpunkt verglichen werden kann. Aufgrund der unterschiedlichen Definitionen des Qualitätsbegriffs, können die Qualitätskriterien an unterschiedlichen Zielen ausgerichtet sein. Beispielsweise beschreiben Krogstie et al. [KLS95] neun Teilaspekte der Qualität, sodass z.B. Qualitätskriterien in Bezug auf die Korrektheit der Geschäftsprozessmodellsyntax und -semantik definiert werden können (vgl. Abschnitt 3.3). Ried [Ri12] unterteilt zwischen Qualitätskriterien die auf der Modell-, Instanz- und Akteurebene Anwendung finden. Des Weiteren unterscheidet Ried zwischen funktionalen und nicht-funktionalen Eigenschaften. Die Instanzebene wird sowohl durch die Modellebene als auch Akteurebene beeinflusst. Als Qualitätskriterien auf Modellebene wird zum Beispiel die eindeutige Strukturierung des Geschäftsprozesses in atomare Arbeitsschritte beschrieben. Auf Akteurebene nennt Ried die „operativen

¹⁶⁸ Vgl. engl. Übersetzung [Sa08, S.208]: „Quality gates are comprehensive evaluations executed at specific points in time assessing the maturity and sustainability of produced artefacts (milestones) and the processes followed to produce them.“

Qualitätseigenschaften“ [Ri12, S.1320], wie beispielsweise die Reaktionszeit eines Beteiligten. Die in den BPM(N)^{Easy}-Schritten verankerten Quality Gates ermöglichen die kontinuierliche Überprüfung der Geschäftsprozesse, sodass deren Modelle je nach Bedarf angepasst werden können. Hierbei kann innerhalb des Qualitätsbegriffs zwischen sogenannten harten und weichen Qualitätskriterien unterschieden werden (vgl. [PS07][AIH11]). Harte Qualitätskriterien sind direkt (z.B. Anzahl der syntaktischen Fehler im Geschäftsprozessmodell, Durchlaufzeit, ...), weiche Qualitätskriterien sind dagegen nicht direkt messbar (z.B. Steigerung der Kundenzufriedenheit, Verbesserung der Mitarbeiterereinbindung,...). In der existierenden Literatur werden insbesondere harte Qualitätskriterien diskutiert (vgl. [YJS05][Va07][HP10][Sá13]), wobei die darin identifizierten Qualitätskriterien auch in BPM(N)^{Easy} angewendet werden können. Zum Beispiel kann ein Quality Gate zur Überprüfung der Leistung des eingesetzten GPM-Systems dienen (z.B. Messung der Systemauslastung, Reaktionszeit des Systems,...). Diese Arbeit leistet einen Beitrag in Bezug auf die Messung und Optimierung von weichen Qualitätskriterien unter Berücksichtigung agiler Aspekte. So können Geschäftsprozesse, anstatt diese aufwendig (d.h. mit hohem Zeit-, Kosten- und Personalaufwand) von Anfang an sehr detailliert in Modellen zu erfassen, iterativ und inkrementell je nach Bedarf der Beteiligten modelliert¹⁶⁹ werden. Durch die Anwendung von weichen Qualitätskriterien wird hierbei gewährleistet, dass die Beteiligten im Kontext der Ausführung von Geschäftsprozessen motiviert bleiben¹⁷⁰ und deren Bedarfe gedeckt sind [Kr12b, S.283ff]. In der Literatur existieren unterschiedliche Ansätze hinsichtlich weicher Qualitätskriterien. Unter anderem beschreiben Mendling et al. [MRC07] „wann“ Geschäftsprozessmodelle für Beteiligte verständlich sind und „wann“ nicht. Beispielsweise beeinflusst die Größe eines Geschäftsprozessmodells (d.h. wie viele Elemente in einem Modell für den Anwender sichtbar sind) wie verständlich das Modell für einen Anwender ist. Gebhart et al. [GMW15] beschreiben den Ansatz, dass während der Ausführung von Geschäftsprozessen weiche Qualitätskriterien unmittelbar erfasst und auszuwerten sind. Die direkte Aufnahme von Feedback erfolgt durch ein mobiles Endgerät (z.B. Smartphone), sodass die Ausführenden des Geschäftsprozesses ortsunabhängig agieren können. Somit kann zum Beispiel festgestellt werden, wann und

¹⁶⁹ Vgl. Abschnitt 2.3.

¹⁷⁰ Vgl. des Weiteren Krogstie [Kr12b, S.283]: „It is important that the persons working in the enterprise are pleased with their situation. If the individuals' goals and motivation are in strong contrast to the overlying business goals, a conflict situation will often arise. Based on this, one should also consider the employees' needs and wishes when designing or rethinking business processes.“

wo die Ausführung (z.B. eines Außendienstmitarbeiters) am „angenehmsten“ war. Einen weiteren Ansatz beschreiben Lohrmann & Reichert [LR13, [Lo15b], wobei Qualitätskriterien aus fachlicher Sicht auf modellierte Geschäftsprozesse angewendet werden. Lohrmann & Reichert verbinden die Qualität eines Geschäftsprozesses insbesondere mit zwei Merkmalen. Zum einen wird analysiert, ob ein Geschäftsprozess zur Erreichung eines Geschäftsziels beiträgt, zum anderen wird überprüft, wie hierbei der Bedarf und Verbrauch von wirtschaftlichen Ressourcen (z.B. Material, Maschinenkapazitäten,...) ist.

Um messbare Qualitätskriterien zu erhalten, müssen abstrakt definierte Qualitätskriterien verfeinert werden. Ein Ansatz ist (vgl. [MRW77][GA11]) die Qualität bzw. die Qualitätskriterien als „unterteilbare“ Qualitätsmerkmale zu formulieren. Die Abbildung 53 skizziert die Verfeinerung eines Qualitätsmerkmals.

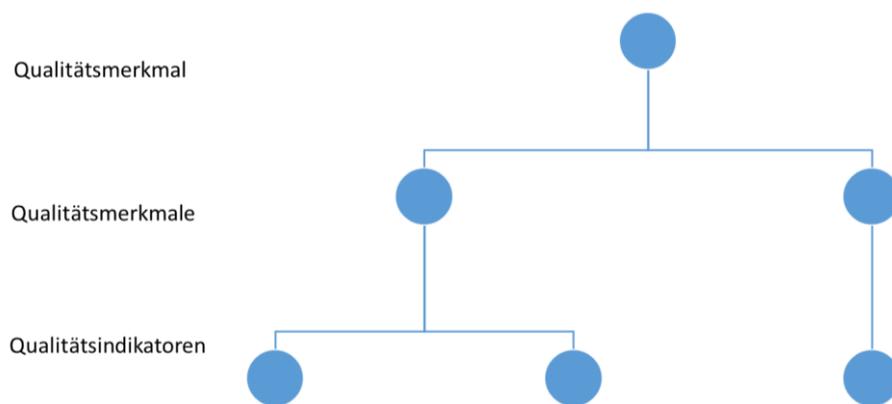


Abbildung 53: Verfeinerung eines Qualitätsmerkmals

Ausgehend von einem Qualitätskriterium wird ein Qualitätsmerkmal definiert. Das Qualitätsmerkmal kann hierbei schrittweise zur Verfeinerung in weitere Qualitätsmerkmale unterteilt werden. In Abbildung 53 sind zur Visualisierung drei Ebenen dargestellt. Auf unterster Ebene befinden sich die sogenannten Qualitätsindikatoren als konkrete Ausprägung eines Qualitätsmerkmals. Qualitätsindikatoren haben die Eigenschaft, dass sie auf Basis erfasster Informationen konkret bestimmt werden können. Die Messung der Qualitätsindikatoren kann entweder (teil-)automatisiert oder manuell stattfinden (vgl. [Ge11, S.38]).

Die automatisierte Analyse von Qualitätsindikatoren ist möglich, wenn formal beschriebene Informationen vorhanden sind. Beispielsweise kann automatisiert überprüft werden, wie viele Aktivitäten in einem Geschäftsprozessmodell vorliegen. Des Weiteren kann die semantische und syntaktische Korrektheit eines BPMN^{Easy}-Modells festgestellt werden. Insbesondere bei weichen Qualitätskriterien ist eine automatisierte Analyse nicht immer möglich. Hierdurch wird es notwendig, dass mit Hilfe von Mensch-Mensch-Interaktion zwischen BPM(N)^{Easy}-Master und den restlichen Beteiligten (Anwender, BPM/IT-Experte) die erforderlichen Informationen zur vollständigen Analyse des Qualitätsindikators manuell erfasst werden. Auf Basis der Qualitätsindikatoren definiert der BPM(N)^{Easy} gebrauchssprachliche Fragen, welche für alle Beteiligten verständlich sind und beantwortet werden können. Die Fragen werden – wie auch die für die automatisierte Analyse benötigten Algorithmen – ex ante, d.h. vor dem Erreichen eines Quality Gates definiert. Die Qualitätskriterien und die damit verbundenen Qualitätsmerkmale und -Indikatoren werden in den in Tabelle 25 dargestellten Quality Gates überprüft.

Tabelle 25: Quality Gates in BPM(N)^{Easy}

Quality Gate	Zeitpunkt der Anwendung	Beschreibung
1	Nach Erfassung des Geschäftsprozessmodells	Im Schritt der Erfassung werden in BPM(N) ^{Easy} Geschäftsprozessmodelle auf Basis der Interaktion aller Beteiligten modelliert. Harte (z.B. Syntaxfehler in den Modellen) und weiche (z.B. Verständlichkeit der Modelle) Qualitätskriterien werden im ersten Quality Gate überprüft.

2	Nach Anreicherung des Geschäftsprozessmodells	Nachdem ein Geschäftsprozessmodell für die (Teil)Automatisierung angereichert wurde, müssen die Beteiligten die daraus resultierende Geschäftsprozessapplikation testen, um mögliche Implementierungsfehler (z.B. Fehler bei der Implementierung von XOR-Bedingungen) zu identifizieren.
3	Nach der Ausführung des Geschäftsprozesses	Das während der Ausführung erfasste Anwenderfeedback (-> Schwerpunkt weiche Qualitätskriterien) und die automatisiert dokumentierten Messungen (-> Schwerpunkt harte Qualitätskriterien) werden zum Zeitpunkt des dritten Quality Gates ausgewertet.

Zur Verdeutlichung der in Quality Gates überprüfbaren Qualitätskriterien wird im Folgenden (in Anlehnung an [LR13][GMW14b][GMW14a]) die Messung eines Qualitätskriteriums zur Überprüfung von Rollen- und Personenzuweisungen in Geschäftsprozessen beschrieben. Durch dieses Qualitätskriterium soll sichergestellt werden, dass zu jeder ausführenden Rolle mindestens zwei ausführende Personen zugeordnet sind. Die Wahrscheinlichkeit von möglichen Ausfällen (z.B. durch Krankheit einer Person) kann somit reduziert und der unnötige Verbrauch (vgl. [LR13]) von Ressourcen (z.B. durch Entstehung von Engpässen bei der Ausführung) vermieden werden. Die Formel (*) stellt die formale Definition des Qualitätsindikators dar:

$$PAR(bp) = \frac{|F(R(bp),r,HSP(r))|}{|R(bp)|} \quad (*)$$

Die Tabelle 26 dient zur Untergliederung und Beschreibung der einzelnen Bestandteile der Formel (*). Hierbei sind die Elementbezeichnungen und Abkürzungen frei gewählt. Zusätzlich ist zu beachten, dass das Qualitätskriterium im vorliegenden Beispiel (1 Rolle mit $n > 1$ Personen) nur unter bestimmten Bedingungen „sinnvoll“ anwendbar ist. Zum Beispiel dann, wenn sichergestellt sein soll, dass eine Aufgabe zusätzlich durch mindestens einen Stellvertreter ausgeführt werden kann. Zu beachten ist hierbei, dass die Qualitätskriterien zu einzelnen Zeitpunkten, den wiederkehrenden Quality Gates (vgl. Abbildung 43), angewendet werden, sodass weiterhin ein „Restrisiko“ besteht, dass zum Beispiel zu einem späteren Zeitpunkt die Auswertung des Qualitätskriteriums zu einem anderen Ergebnis führt.

Tabelle 26: Funktionen und Variablen im Beispiel PAR

Element	Beschreibung
PAR(bp)	Person Availability of Roles: Grad, wie viele Rollen mehr als eine Person in Geschäftsprozess bp besitzen.
R(bp)	Role of Business Process: Rollen, welche im Geschäftsprozess bp Anwendung finden.
F(e, v, c)	Filter: Filter, welche die Elemente e nach der Kondition c filtert, wobei v als Iterator verwendet wird.
HSP(r)	Role Has Several Persons: Wahr, wenn die Rolle mehr als eine Person besitzt.

Innerhalb der Quality Gates muss sichergestellt werden, dass alle Funktionen und Variablen anwendbar sind. Die Tabelle 27 listet die einzelnen Bestandteile der Formel (*) auf und beschreibt, wie diese im Einzelnen abgefragt werden können (automatisiert oder manuell). Beispielsweise kann automatisiert überprüft werden, ob alle Aktivitäten eines BPMN^{Easy}-Modells eine Rolle besitzen. Ob alle Rollen mindestens zwei ausführende Personen beinhalten, muss durch eine direkte gebrauchssprachliche Befragung der Beteiligten geklärt werden, z.B. „Sind jeder der Rollen R1 bis R5 mindestens zwei Personen zugewiesen?“. Eine weitere Verfeinerung der Rollen-Ressourcen- und Personenzuweisung, wie in Abschnitt 4.2.3 (vgl. zudem [Bu94]) beschrieben, kann durch zusätzliche Qualitätskriterien und daraus resultierenden Fragen unterstützt werden.

Tabelle 27: Art der Messung der Parameter

Element	Beschreibung
bp	Der Geschäftsprozess ist in einem BPMN ^{Easy} -XML abgespeichert und kann technisch verarbeitet werden.
PAR(bp)	Diese Funktion stellt eine Komposition anderer Funktionen dar.
R(bp)	Die Rollen sind an einem BPMN ^{Easy} -Geschäftsprozessmodell annotiert und können technisch ausgewertet werden.
F(e, v, c)	Diese Funktion stellt eine Komposition aus anderen Funktionen dar. Eine Iteration mit Hilfe des Iterators v kann technisch automatisiert werden.
HSP(r)	Die Information, ob eine Rolle mehr als eine Person besitzt, wird nicht direkt in einem BPMN ^{Easy} -Modell abgespeichert. Zur Qualitätsüberprüfung wird deshalb eine geschlossene Frage mit Hilfe der Gebrauchssprache formuliert, beispielsweise: „Kann die Aktivität x durch mehrere Personen ausgeführt werden?“ Als Eingabeparameter der Funktion gelten entweder die Wahrheitswerte „wahr“ oder „falsch“.

Anstatt die Ergebnisse der Formel (*) gebrauchssprachlich auszudrücken wird in BPM(N)^{Easy} ein einfach zu interpretierender numerischer Wert genutzt. Die Anwendung der Formel (*) liefert als Ergebnis einen Wert zwischen 0 und 1 zurück. Durch die Formalisierung¹⁷¹ wird das Risiko von Missverständnissen aufgrund ungeklärter Begriffe reduziert. Die Tabelle 28 enthält die Beschreibung der Interpretation im Beispiel der Rollen-Personen-Zuweisung.

Tabelle 28: Interpretation der Werte

Wert	Interpretation
0	Keine Rolle besitzt mindestens zwei Personen.
zwischen 0 und 1	Es existieren Rollen mit weniger als zwei Personen.

¹⁷¹ Vgl. engl. Übersetzung [GMW14b, S.283]: „Furthermore, the informal description requires interpretation effort that can result in misunderstandings and wrong measures.“

1	Alle Rollen des Geschäftsprozesses besitzen mindestens zwei Personen.
---	---

Der numerische Wert 0 dient zur Visualisierung, dass das Qualitätskriterium nicht erfüllt wird, d.h. keine Rolle eine Personenzuordnung besitzt. Der Geschäftsprozess könnte in dieser Form nicht real ausgeführt werden. Ein Wert zwischen 0 und 1 indiziert, dass nicht alle Rollen der Anforderung des Qualitätskriteriums entsprechen. Umso höher der Wert (z.B. 0.8), umso mehr Rollen besitzen mindestens zwei Personen. Der Wert 1 spiegelt wider, dass alle Rollen die Anforderung erfüllen und mindestens zwei Personen der Rolle zugeordnet sind. Der BPM(N)^{Easy}-Master entscheidet gemeinsam mit allen Beteiligten in jedem Quality Gate pro Iteration, ob die existierenden Qualitätskriterien weiter gelten oder verändert werden müssen.

Je nach Anforderung können bei der Durchführung eines BPM(N)^{Easy}- Projekts andere (oder weitere) Qualitätskriterien definiert werden¹⁷², sodass sowohl weiche als auch harte Faktoren analysiert werden können.

4.2.5. Anreicherung

Müssen zuvor erfasste Geschäftsprozessmodelle weiter verfeinert werden, wird diese Verfeinerung während der „Anreicherung“ durchgeführt. Insbesondere fachliche Geschäftsprozessmodelle sind häufig (aus technischer Sicht) grob granular modelliert (vgl. [FG08]). Der Begriff der Anreicherung beschreibt, dass daher zusätzliche Informationen (z.B. technische Details, Regeln,...) zu den existierenden Geschäftsprozessmodellen hinzugefügt werden. Hierbei muss zunächst entschieden werden, welche Modellierungssprach(en) die Anreicherung unterstützen. Beispielsweise eignen sich Geschäftsprozessen, welche viele Ausführungsvarianten und lose Strukturen beinhalten, für deklarative Modellierungssprachen. Die Modellierungssprache DPIL (Declarative Process Intermediate Language) erlaubt beispielsweise das Modellieren und Ausführen von unstrukturierten Geschäftsprozessen aus multiplen Perspektiven (vgl. [SZ15]). Zudem sind hybride Ansätze denkbar, d.h. dass die Anreicherung eines Geschäftsprozesses sowohl deklarativ (für eher unstrukturierte Teile) als auch imperativ

¹⁷² Vgl. Abschnitt 6 (Anwendungsbeispiel).

(für eher strukturierte Teile) durchgeführt wird. Die in dieser Arbeit beschriebene Methode nutzt für die Anreicherung die imperative Modellierungssprache BPMN^{Easy}.

BPMN^{Easy}-Modelle und deren Mediafiles sind hierbei die Grundlage für z.B. komplexere BPMN 2.0-Modelle. Wie weit die Anreicherung reicht, ist individuell zu definieren. Sind die BPMN^{Easy}-Modelle aus dem Erfassungsschritt bereits ausreichend für die Erfüllung der Anforderungen, kann die Anreicherung auch übersprungen werden. Der Schwerpunkt von BPM(N)^{Easy} liegt auf der Anreicherung hinsichtlich einer Geschäftsprozessautomatisierung. Ziel der Anreicherung ist die Implementierung einer Geschäftsprozessapplikation. Nach Stiehl [St14] unterstützen prozessorientierte Applikationen¹⁷³ die Wiederverwendung von Daten und Funktionen in einer Organisation. Organisatorische– und IT-System-Grenzen (z.B. unterschiedliche Abteilungen, ...) werden hierbei durch eine End-to-End-Sicht auf die Geschäftsprozesse überwunden. Rumpe [Ru01] beschreibt zusätzlich die Möglichkeit, die Implementierung von Applikationen noch effizienter und in kürzeren Zyklen zu gestalten, indem aus Modellen (z.B. BPMN 2.0 Geschäftsprozessmodell) direkt Programmcode generiert wird.

Zum Start des Anreicherungsschritts muss zum einen das gemeinsame Verständnis aller Beteiligten (IT-Experte, BPM-Experte, Anwender) bzgl. Geschäftsprozessautomatisierung sichergestellt werden¹⁷⁴.

Zum anderen wird ein sogenanntes Story Pulling durchgeführt. Orientiert an dem Kanban-Pullsystem (vgl. [An10][Ep11]) entscheiden die Beteiligten, welche Anforderungen (aus Modellen und User Stories) in der aktuellen Iteration implementiert werden. Die Anzahl der ausgewählten User Stories ist dadurch limitiert, dass am Ende der Iteration der implementierte Geschäftsprozess End-to-End durchlaufbar sein muss. Anforderungen, die ggfs. nicht vollständig innerhalb der Iteration umgesetzt werden können, werden somit nicht ausgewählt.

Zudem dient das Story Pulling zur Feststellung ob weitere Informationen, welche ggfs. zur Automatisierung des Geschäftsprozesses notwendig sind, fehlen. Fehlende

¹⁷³ Vgl. engl. Übersetzung [St14, S.19]: „Process-driven applications are business-oriented applications that support differentiating end-to-end business processes spanning functional, system, and organizational boundaries by reusing data and functionality from platforms and applications.“.

¹⁷⁴ Die Erfahrung des Autors zeigt, dass insbesondere Anwender sehr unterschiedliche Vorstellungen bzgl. dem Begriff der Geschäftsprozessautomatisierung besitzen.

Informationen werden entweder direkt durch eine Diskussion ergänzt. Anderenfalls müssen diese in einer weiteren Iteration während der Erfassung erarbeitet, oder eigenständig durch die IT-Experten hinzugefügt werden (z.B. Verbindungsparameter zu einem Drittsystem). Die Tabelle 29 listet die notwendigen Informationen zur Anreicherung eines Geschäftsprozessmodells auf.

Wurde die Erfassung mit Hilfe von BPMN^{Easy} durchgeführt, können die Mediafiles zusätzlich zur Implementierungsunterstützung angewendet werden. BPM-Experten und IT-Experten wandeln die gebrauchssprachlichen Informationen hierbei beispielsweise in BPMN 2.0-Aktivitäten, -Ereignisse oder -Gateways um.

Tabelle 29: Erforderliche Informationen zur Anreicherung (vgl. [A114a])

Rollen und Anwender	Zur Automatisierung muss der BPM-Engine ein formal strukturiertes Rollen- und Anwendermodell vorliegen. Unsicherheiten (z.B. welche Rolle die Aktivität XYZ ausführt) müssen beseitigt werden.
Dialoge	Grafische Benutzeroberflächen dienen zur Teilautomatisierung eines Geschäftsprozessmodells. BPMN ^{Easy} Mediafiles (z.B. Foto von einem Antragspapier) können als Vorlage der Benutzeroberfläche dienen.
Regeln	Regelbeschreibungen (z.B. Audiofile zur Beschreibung einer komplexen Geschäftsregel) werden formalisiert und stellen die Grundlage zur Ablaufsteuerung des automatisierten Geschäftsprozessmodells dar.
Externe Systeme	Für die Anbindung von externen Systemen (z.B. ERP –oder CRM-Systeme) müssen Verbindungsparameter (z.B. Port, Zugriffsrechte,...) bekannt sein.
Automatisierte Funktionen	Zur vollständigen Automatisierung von z.B. Aktivitäten müssen Algorithmen definiert und durch die BPM-Engine ausführbar werden.

Daten, Informationen und Daten-Mappings	Das fachliche Datenmodell (z.B. Auflistung der, für die Antragsstellung relevanten Daten, wie Name, Adresse...) muss um plattformspezifische Datentypen angereichert werden. Zudem werden durch den IT-Experten technisch relevante Daten (z.B. temporäre Variablen,...) ergänzt.
Technische Exceptions	Neben den fachlichen Regeln zur Ablaufsteuerung und der reinen Anbindung von externen Systemen, müssen technische Ausnahmeregel (Exceptions) definiert werden. Des Weiteren muss sichergestellt sein, dass technische Ausnahmen wiederum zu aus Anwendersicht „verständlichen“ Meldungen führen. Soll beispielsweise ein ERP System aufgerufen werden, muss ex ante definiert werden, wie im Fehlerfall (z.B. System nicht erreichbar) reagiert wird.

Neben der Implementierung der User Stories, muss die Verbindung zwischen fachlicher und technischer Sicht auf das Geschäftsprozessmodell berücksichtigt werden. Insbesondere für die nachfolgende Qualitätskontrolle und die Abstimmung mit den Anwendern muss gewährleistet sein, dass Anreicherungen nachvollzogen werden können. Je nach Implementierungsstrategie werden die technischen Anreicherungen 1) in einem gemeinsamen Modell 2) getrennt in zwei unterschiedliche Layer oder 3) getrennt in zwei unterschiedliche Layer mit einem zusätzlichen „Service-Contract“¹⁷⁵ durchgeführt (vgl. [St14]). Die Abbildung 54 stellt ein Beispiel zur Anreicherung eines fachlichen Geschäftsprozessmodells auf Basis zweier unterschiedlicher Layer dar. Beispielsweise wurde die Aktivität „ERP System aufrufen“ durch den Anwender vorgegeben und durch den BPM-Experte modelliert. Zum realen Aufruf des ERP Systems ist es notwendig, die Aktivität durch technische Subaktivitäten zu verfeinern.

¹⁷⁵ Neben dem in Abbildung 54 dargestellten Beispiel kann die Anreicherung über einen „Service Contract Implementation Layer“ durchgeführt werden. Diese Zwischenschicht ermöglicht eine Kapselung der unterschiedlichen Geschäftsprozessmodelle, vgl. [St14, S.73f].

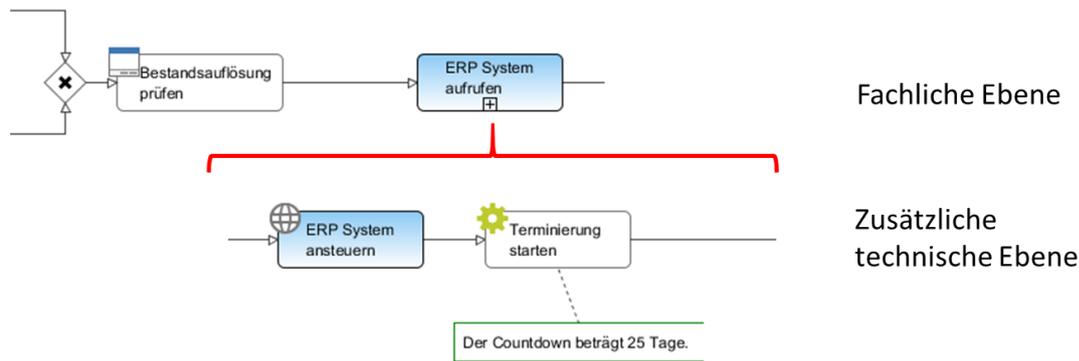


Abbildung 54: Beispiel - Verfeinerung eines Geschäftsprozessmodells

Eine besondere Herausforderung besteht, wenn die durch die Anwender vorgegebenen Aktivitäten durch die Automatisierung reduziert oder verschoben werden. Werden beispielsweise fachlich modellierte Aktivitäten durch eine technische Aktivität (z.B. implementierter Algorithmus) ersetzt, muss diese Änderung für alle Beteiligten rückverfolgbar sein. Diese Rückverfolgbarkeit kann beispielsweise durch Anwendertests oder der Abfrage von fachlichen Qualitätsfragen erfolgen. Wird eine Rückverfolgbarkeit technisch vollständig unterstützt, d.h. die Beziehungen zwischen den fachlichen und technischen Elementen eines Geschäftsprozessmodell sind zu jeder Zeit bekannt, so wird dieser Zustand als sogenannter „BPM Roundtrip“ bezeichnet (vgl. [Al11][Me13]). Je nach Komplexität des zu implementierenden Geschäftsprozesses ist der Pflegeaufwand des fachlich-technischen „Mappings“ mit hohem (zeitlichen) Aufwand verbunden. Beispielsweise können Interaktionen des Anwenders (z.B. Klick auf einen Button im Frontend) bereits zu Veränderungen im Sequenzfluss des Geschäftsprozessmodells im Hintergrund (Backend) führen (vgl. [Ka12, S.186ff]). Zur Sicherstellung der Synchronisation zwischen allen Beteiligten werden in BPM(N)^{Easy} Daily Meetings durchgeführt (vgl. [Co05][SS13]). Während der kurzen täglichen Abstimmung (dem Daily Meeting) werden technische und fachliche Fragestellungen im Team diskutiert. Zusätzlich werden bei Unsicherheiten bzgl. des Verständnisses die Rolle des BPM-Experten oder Anwender mit einbezogen.

4.2.6. Ausführung/Analyse

In BPM(N)^{Easy} wird neben der Entwicklungszeit der Geschäftsprozessmodelle (Erfassung und Anreicherung) die Laufzeit berücksichtigt. Während der Laufzeit werden sowohl die Geschäftsprozesse real ausgeführt, also auch analysiert. Die Ausführung der Geschäftsprozessaktivitäten erfolgt hierbei entweder manuell, teilautomatisiert oder automatisiert (vgl. Abschnitt 2.2.1). Manuelle Aktivitäten können durch die Zuhilfenahme der erfassten Geschäftsprozessmodelle und Mediafiles (z.B. Dokumentation oder Schulungsunterlagen) ausgeführt werden. Des Weiteren dienen implementierte Geschäftsprozessapplikationen zur Unterstützung von teilautomatisierten Aktivitäten.

Innerhalb der BPM(N)^{Easy}-Iteration wird diese Ausführung unmittelbar analysiert. Die Analyse basiert sowohl auf (in Echtzeit) gemessenen Kennzahlen, als auch auf der Erfassung von Anwenderfeedback. Beispielsweise können Monitoring-Komponenten¹⁷⁶ von Geschäftsprozessmanagementsystemen zur Visualisierung des aktuellen (oder zukünftigen) Zustands von Geschäftsprozessen oder deren Aktivitäten (z.B. Bearbeitungsfortschritt, Durchlaufzeit,...) eingesetzt werden [Sc12]. Bei einem Unter- oder Überschreiten von ex ante definierten Grenzwerten kann hierbei direkt reagiert werden [SS05]. Ist beispielsweise eine Ressource mit der Ausführung einer bestimmten Aktivität überlastet, kann ggfs. die Zuordnung zukünftiger Instanzen derselben Aktivität auf eine andere Ressource erfolgen. Die Analyse von Geschäftsprozessen wird in der Literatur in Form einer Vielzahl von Ansätzen und implementierten Werkzeugen detailliert beschrieben (vgl. zu weiterem Überblick [Me06, S.231ff][Th13, S.37ff] [Wi15]). Hierbei ist „die Prozessleistung nach Fehlern oder Durchlaufzeiten im Verhältnis der Bearbeitungszeit zu bewerten, (...) verhältnismäßig einfach. Aber wie viel Kosteneinsparung durch einen nicht gemachten Fehler oder ein schnelleres Arbeiten erreicht wird, ist sehr schwer zu ermitteln“ [Th13, S.12]. Des Weiteren erfordert die Analyse von Monitoring-Ergebnissen immer „(...) einen gewissen Grad an manuellen Entscheidungen und Interpretationen der zugrunde liegenden Daten und der abgeleiteten

¹⁷⁶ Monitoring kann im erweiterten Sinne auch zu Analyse gezählt werden [He02]. Vgl. des Weiteren [Ga15, S.13]: „(...) Prozess-Monitoring: Laufende Analyse der Prozess-Kennzahlen im Hinblick auf die Erreichung der Prozessziele (...)“.

Maßnahmen (...)“ [Ba14a, S.552]. Die Integration der Anwender ist deshalb von signifikanter Bedeutung (vgl. [Th13 S.92]).

BPM(N)^{Easy} adaptiert und erweitert die existierenden Ansätze um die vertiefte Einbindung der ausführenden Anwender. In Tabelle 30 sind die Hilfsmittel von BPM(N)^{Easy} aufgelistet.

Tabelle 30: Möglichkeiten der Analyse in BPM(N)^{Easy}

FeedbackApp	Das Erfassen von Anwenderfeedback während der Ausführung von Aktivitäten erfolgt Software-gestützt. Hierzu werden vordefinierte Qualitätskriterien kontextabhängig während der Ausführung in Form von Fragen dem Anwender angezeigt. Die durch traditionelle Ansätze erfassten Daten (z.B. Messung von Kennzahlen) werden somit durch in Echtzeit gesammeltes User Feedback angereichert. Ein Software-Prototyp und die Anwendung dessen wird in den Abschnitten 5 und 6 beschrieben.
Easy Capture Sheet ¹⁷⁷	Das sogenannte Easy Capture Sheet (vgl. [MOW15]) ermöglicht die Darstellung eines Geschäftsprozess in einfacher Tabellenform. Es bietet somit eine Alternativdarstellung zur reinen grafischen Darstellung des Geschäftsprozessmodells. Insbesondere Anwender ohne Kenntnisse bzgl. grafischer Modellierungssprache können somit intensiver integriert werden. Zudem kann das Easy Capture Sheet zur direkten Erfassung von Feedback genutzt werden.

Die Abbildung 55 stellt ein Beispiel eines Easy Capture Sheets dar. Die obere Hälfte der Abbildung visualisiert einen Ausschnitt eines BPMN^{Easy}-Modells. Der Ausschnitt beinhaltet drei Aktivitäten: „Formular erfassen“ (teilautomatisierte Aktivität), „Abstimmung durchführen“ (manuelle Aktivität) und „Kunde informieren“ (automatisierte Aktivität). Diese Sequenz muss für die Erstellung eines Easy Capture Sheets transformiert werden. Existierende Softwarewerkzeuge, wie beispielsweise das BPM Touch (vgl. Abschnitt 4.4.2), ermöglichen bereits eine derartige Transformation.

¹⁷⁷ Zusätzlich sei auf Baumann et al. [Ba14b] verwiesen, welche die Transformation von „Human-driven“ (Geschäfts-)Prozessmodellen (Prozesse, welche sich insbesondere durch menschliche Aktivitäten auszeichnen), in eine papierbasierte Schritt-für-Schritt-Checkliste beschreibt.

Die untere Hälfte der Abbildung 55 zeigt die drei Spalten des Easy Capture Sheets. Elemente (auch parallel-ausgeführte Aktivitäten) werden hierbei sequenziell aufgelistet.

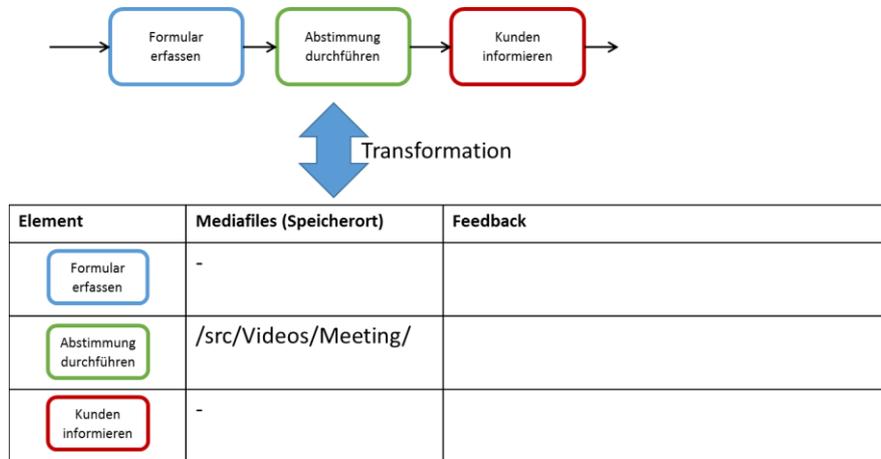


Abbildung 55: Beispiel eines Easy Capture Sheets

Des Weiteren wird ggfs. der Speicherort der annotierten Mediafiles aufgelistet. Mit Hilfe der „Feedback“-Spalte können die ausführenden Anwender während der Ausführung einfach Kommentare abgegeben. Wird der Einsatz solcher Easy Capture Sheets durch das BPM(N)^{Easy}-Projektteam beschlossen, verantwortet der BPM(N)^{Easy}-Master die Verfügbarkeit und Pflege. Das erfasste Feedback dient im nächsten Schritt als Grundlage der Optimierung der Geschäftsprozesse.

4.2.7. Optimierung

Der Schritt der Optimierung in BPM(N)^{Easy} dient zum einen zur kontinuierlichen Verbesserung der Geschäftsprozesse und deren Modelle. Zum anderen beinhaltet die Optimierung eine Retrospektive der vorangegangenen Iteration zur Anpassung der Vorgehensweise selbst.

- Optimierung der Geschäftsprozesse und Geschäftsprozessmodelle

Die Geschäftsprozessoptimierung (GPO) ist ein zentraler Bestandteil des Geschäftsprozessmanagements (vgl. Abschnitt 2.2.2). Nach Gadatsch [Ga12b, S.19] wird die Optimierung aller Geschäftsprozesse insbesondere zur nachhaltigen Ausrichtung an den Kundenanforderungen durchgeführt. Im Kontext des agilen Geschäftsprozessmanagement wird diese Sicht um die Berücksichtigung der Anwenderanforderungen erweitert. Neben dem Ziel der Erfüllung von

Kundenanforderungen (z.B. schnellere Reaktionszeit) liegt der Fokus auf der Einbindung aller Ausführenden. Die Ziele der Optimierung können hierbei, sowohl die Geschäftsprozesse an sich, als auch deren Modelle betreffen. Zum Beispiel kann durch das Eliminieren von überflüssigen Aktivitäten, Parallelisieren oder Automatisieren von Aktivitäten eine Optimierung erreicht werden [BW09] (z.B. Einsparung von Kosten, Reduzierung der Durchlaufzeit,...). Zudem kann die Optimierung auf Basis der Beseitigung der Analyse der Geschäftsprozessmodelle durchgeführt werden (z.B. zur Beseitigung von Deadlocks¹⁷⁸) [VTM08]. In BPM(N)^{Easy} wird die Zieldefinition mit Hilfe von Qualitätskriterien festgelegt (vgl. Abschnitt 4.2.4). Beispielsweise kann definiert werden, dass die Durchlaufzeit des erfassten und ausgeführten Geschäftsprozesses nicht länger als fünf Tage betragen darf oder dass alle an einem Geschäftsprozess Beteiligten vor der Ausführung der Aktivitäten geschult werden müssen (vgl. [GMW14a]). Diese SOLL-Werte werden während des Optimierungsschritts mit den IST-Werten verglichen. IST-Werte können dabei gemessene Kennzahlen oder erfasstes Anwenderfeedback darstellen. Werden Kennzahlen eingesetzt, muss nach Weske [We12] sichergestellt sein, dass neben dem Ziel (warum wird diese Kennzahl definiert?), der Name, Datentyp, Berechnungsalgorithmus, Zielwert und Ober- und Untergrenze ex ante definiert wurden (z.B. während der Erstellung der Qualitätskriterien im Erfassungsschritt). Neben Kennzahlen wird das, während der Ausführung gesammelte, Feedback analysiert. Die Analyse der Optimierungsvorschläge und Kennzahlen führt zu einer Priorisierung der Anforderungen und zur Vorbereitung der nächsten Iteration. Welcher Ansatz zur tatsächlichen Gewichtung und Einordnung gewählt wird, ist abhängig von den Projektrahmenbedingungen zu entscheiden. Zum Beispiel kann eine Optimierung mit Hilfe einer Bewertungsmatrix und zusätzlichen, zur Gewichtung herangezogenen Kriterien (z.B. Kosten, Qualität, Zeit), durchgeführt werden [Ba99, S.119ff]. Für weitere Beispiele und Erläuterungen in Bezug auf Geschäftsprozessoptimierung sei auf [VTM08][BW09][NRM11][GHS13] verwiesen.

¹⁷⁸ Der Begriff des Deadlocks beschreibt den Zustand, in dem unterschiedliche Geschäftsprozesse gegenseitig aufeinander warten. Vgl. [Dh06, S.484]: „A deadlock involving a set of processes D is a situation in which each process P_i in D satisfies two conditions 1. Process P_i is blocked on some event e_j . 2. Event e_j can be caused only by actions of other process(es) in D .“

Als Ergebnis der Optimierung werden in BPM(N)^{Easy} priorisierte User Stories aus dem Optimierungsbacklog dem nachfolgenden Erfassungsschritt übergeben. Abbildung 56 stellt ein Beispiel eines Optimierungsbacklogs dar.

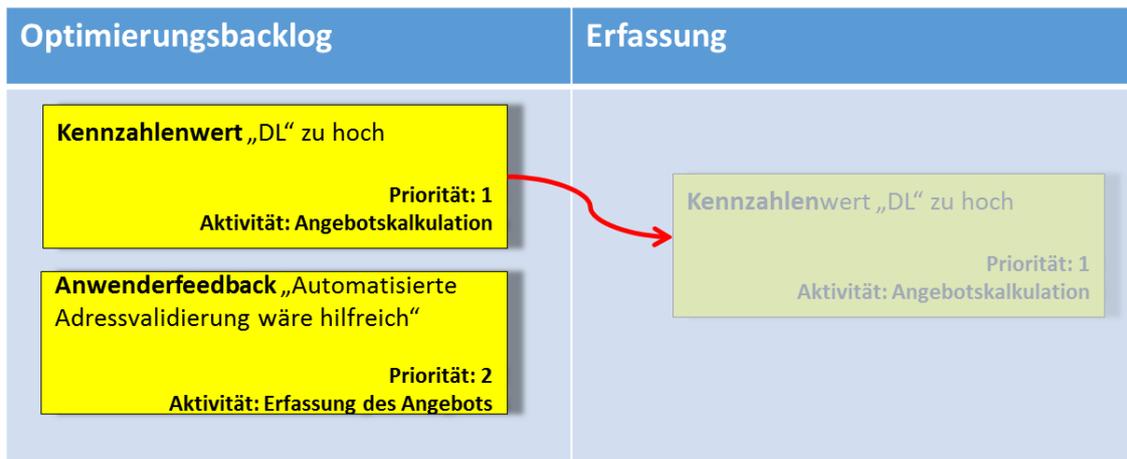


Abbildung 56: Beispiel eines Optimierungsbacklogs

Die Abbildung 56 bildet zwei User Stories ab. Die erste User Story des Beispiels resultiert aus der Analyse (z.B. Echtzeit-Monitoring während der Ausführung) eines Kennzahlenwerts „DL“ (Durchlaufzeit). Die Optimierung des Werts „DL“ wird mit der Priorität 1 eingestuft, wobei 1 die höchste Priorität darstellt. Die zweite User Story enthält ein Feedback eines Anwenders. Basierend auf den Projektanforderungen wird dieser Optimierungsvorschlag mit Priorität 2 eingestuft und im Gegensatz zu der ersten User Story nicht in der nächsten Iteration berücksichtigt.

- Optimierung der Vorgehensweise (Retrospektive und Shaping)

Innerhalb des Optimierungsschritts werden die Erfahrungen und Ereignisse der vorhergegangenen Iteration diskutiert. Hierbei leitet der BPM(N)^{Easy}-Master das Synchronisationstreffen und die Diskussion und fordert alle Teammitglieder zum Feedback auf. Der in [SS13][Co05] beschriebene Ansatz einer Retrospektive dient in BPM(N)^{Easy} als Grundlage. Folgende Fragen sind zu formulieren:

1. Wie verlief in der vergangenen Iteration die Interaktion zwischen den beteiligten Menschen? Hierbei werden mögliche Konflikte unter allen Beteiligten analysiert (z.B. Probleme beim gegenseitigen Verständnis von IT-Experte und Anwender).
2. Welche Schritte sind zufriedenstellend durchlaufen worden? Welche Verbesserungspotentiale gibt es? Insbesondere die Gewichtung der einzelnen

Schritte ist zu diskutieren. Wurde beispielsweise für die Anreicherung eines Geschäftsprozessmodells zu wenig Zeit eingeplant, kann möglicherweise die Iterationsdauer verlängert werden.

3. Wie können die identifizierten Verbesserungspotentiale in Bezug auf die Vorgehensweise umgesetzt werden?
4. Wurden die Projektrahmenbedingungen eingehalten? Zur verbesserten Visualisierung von Zeit und Budget kann beispielsweise ein Burn down Chart verwendet werden, vgl. Abschnitt 2.3.5.

Zusätzlich können die Ergebnisse der Retrospektive gesammelt werden und als „Best Practices“ für weitere GPM-Projekte dienen.

4.2.8. Zusammenfassende Übersicht der Vorgehensweise

Mit BPM(N)^{Easy} ist eine Vorgehensweise zum agilen Geschäftsprozessmanagement beschrieben, die zur iterativen, inkrementellen und empirischen Erfassung, Anreicherung, Ausführung, Analyse und Optimierung angewendet werden kann. Hierbei stehen die Beteiligten (z.B. Anwender eines Geschäftsprozesses) und deren Motivation, Bereitschaft und Fähigkeiten (vgl. Abschnitt 4.2.2) im Fokus. Im Rahmen von regelmäßigen Interaktionen und Synchronisationstreffen werden die unterschiedlichen Erfahrungen und Expertisen der Beteiligten geteilt. Zum Beispiel wird in kurzen Iterationen mit Hilfe der Paarmodellierung sichergestellt, dass ein erfasster Geschäftsprozess den Anforderungen der Anwender entspricht. Durch die Selbstorganisation und Kompetenzzuteilung wird intrinsische Motivation aller Beteiligten gefördert.

Des Weiteren können Verständnisfragen, beispielsweise zur agilen Vorgehensweise selbst, direkt durch den BPM(N)^{Easy}-Master beantwortet werden. Kontinuierliche Schulungen, die gebrauchssprachliche Interaktion und die enge Vernetzung aller Projektteammitglieder tragen zu einer zusätzlichen Stabilisierung innerhalb eines GPM-Projekts bei.

Die Abbildung 57 visualisiert die in den Abschnitten 4.2.1 - 4.2.7 beschriebenen Schritte und Praktiken¹⁷⁹ (vgl. obere Hälfte der Abbildung). Das Fundament dieser Schritte und Praktiken bilden die Anker Interaktion, Synchronisation und Qualität, welche wiederum auf dem Methodischen Konstruktivismus aufsetzen (vgl. untere Hälfte der Abbildung).

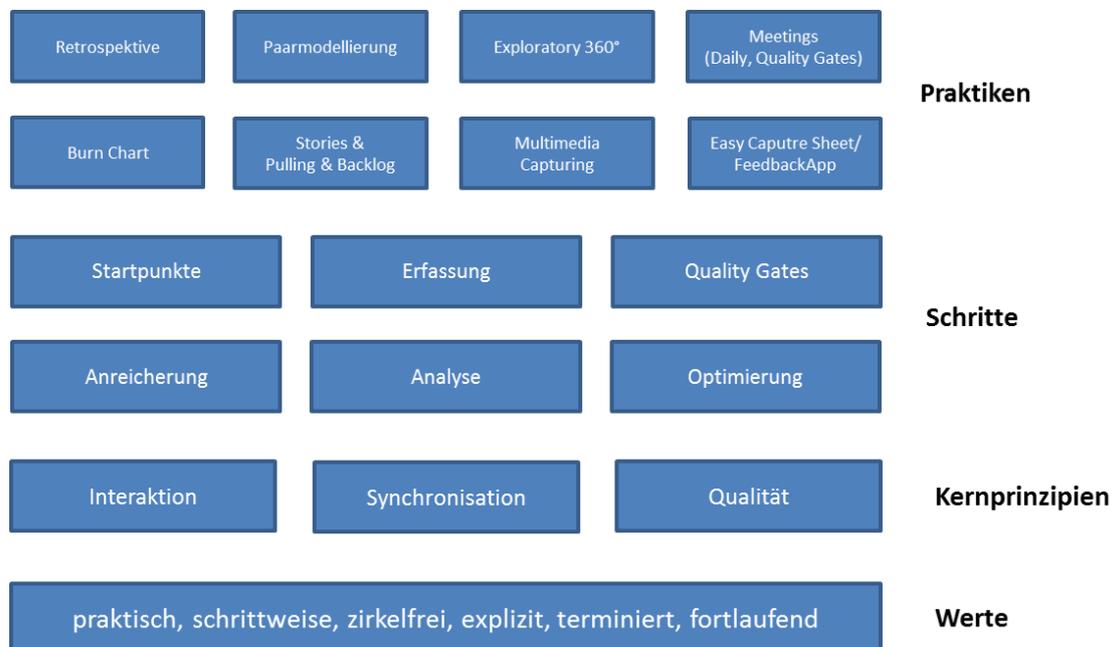


Abbildung 57: Grafische Zusammenfassung der BPM(N)^{Easy} Vorgehensweise

Die Abbildung 57 dient ausschließlich als zusammenfassender Überblick der Vorgehensweise. Die Anordnung der einzelnen Rechtecke steht in keiner Verbindung zur Reihenfolge der Anwendung. Beispielsweise kann die Praktik der Paarmodellierung, sowohl in der Erfassung, als auch Anreicherung angewendet werden. Die in Abschnitt 4.1 beschriebenen Aspekte des Methodendesigns werden durch BPM(N)^{Easy}-Vorgehensweise weitestgehend erfüllt. Die Tabelle 31 dient zur Beantwortung des Fragenkatalogs (vgl. [Tr12]).

Tabelle 31: Bewertung der Methode auf Basis des Fragenkatalogs nach [Tr12]

1	Existiert für die betrachtete Methode eine Domänenabgrenzung?	BPM(N) ^{Easy} ist im Kontext Geschäftsprozessmanagement einsetzbar und fokussiert auf die Unterstützung der Erfassung,
---	---	---

¹⁷⁹ Im Kontext dieser Arbeit wird unter Praktik eine in der Praxis unmittelbar anwendbare Sache eingeordnet. Vgl. des Weiteren von Rosing et al., welche [MJA15, S.555]: „(...) finally are the agile practices that form the basis for work within an agile setting.“

		Anreicherung, Ausführung/Analyse und Optimierung von Geschäftsprozessen.
2	Sind die Methodenelemente und Praktiken klar definiert?	BPM(N) ^{Easy} basiert auf den Anker Interaktion, Synchronisation und Qualität und beschreibt explizit Schritte (Erfassung, Anreicherung, Ausführung/Analyse, Optimierung) und Praktiken (z.B. Paarmodellierung).
3	Werden die geforderten Designprinzipien erfüllt?	BPM(N) ^{Easy} stellt eine agile Methode dar und beinhaltet die Prinzipien des Agilen Manifests.
4	Existieren ein phasenbezogenes Rahmenwerk, Projekt Templates und Entwicklungsszenarien?	BPM(N) ^{Easy} kann als Rahmenwerk für ein agiles Vorgehen in Projekten genutzt werden. Bzgl. Entwicklungsszenarien vgl. Abschnitt 6.1.
5	Enthält die Methode Werte und Prinzipien? Sind Projektziele und Projektcharakteristika definiert?	Der konsequente Einsatz der Gebrauchssprache und die gesamtheitliche Integration der Beteiligten stehen im Fokus von BPM(N) ^{Easy} . Projektziele sind nicht explizit definiert.
6	Existieren Prozesse zur Methoden Anpassung?	Die Anpassung der Methode (z.B. aufgrund neuer Anforderungen) wird in BPM(N) ^{Easy} durch regelmäßige Retrospektiven ermöglicht.
7	Existieren Richtlinien für die Übertragung auf große Projekte?	Es bestehen keine expliziten Richtlinien. Dennoch kann BPM(N) ^{Easy} beispielsweise für Unterteilprojekte in ein großes Projekt eingegliedert werden.
8	Berücksichtigt die Methode relevante Soft Factors?	Die BPM(N) ^{Easy} Quality Gates stellen eine kontinuierliche Überprüfung von weichen und harten Qualitätskriterien sicher. Des Weiteren

		agiert der BPM(N) ^{Easy} -Master als Mediator bei Konflikten innerhalb des Projektteams.
9	Ist die Methode ausreichend flexibel?	BPM(N) ^{Easy} besteht aus mehreren Schritten die individuell angepasst bzw. bei Bedarf übersprungen werden können. Des Weiteren kann die Iterationsdauer flexibel angepasst werden.

Die beschriebene Methode kann zur kontinuierlichen Verbesserung von Geschäftsprozessen und des Geschäftsprozessmanagements angewendet werden. BPM(N)^{Easy} ermöglicht hierbei die schrittweise Annäherung an den von Rosemann beschriebenen „Perfect Match“ [Ro15], d.h. den zu einem bestimmten Zeitpunkt bestmöglichen Einsatz von Geschäftsprozessmanagement. Die in Abbildung 58 abgebildeten Quadranten zeigen zum einen den Bedarf an GPM (z.B. Erfassung von Geschäftsprozessmodellen), zum anderen indiziert diese die Fähigkeit dieses „richtig“ einzusetzen. Sowohl die x-Achse, als auch y-Achse bilden ein Kontinuum ab (vgl. [Ro15, S.269]).

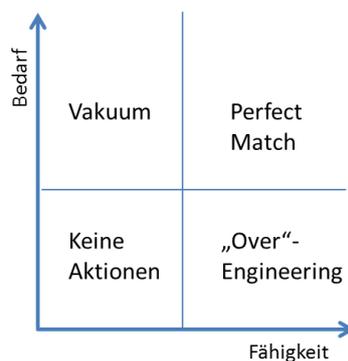


Abbildung 58: Bedarf-Fähigkeitsportfolio nach [Ro15]

Rosemann beschreibt den Einsatz des Portfolios im Zusammenhang mit, von einem GPM-Team¹⁸⁰ angebotenen, Services. Im Kontext von BPM(N)^{Easy} können durchgeführte

¹⁸⁰ In Rosemann [Ro15, S.267] wird das zentrale GPM-Team als „Center of Excellence (aka BPM Support Office)“ bezeichnet.

GPM-Projekte in die Quadranten eingeordnet werden, sodass die Ansicht beispielsweise als „strategische Retrospektive“ dient. Der „Perfekte Match“ ist erreicht, wenn Anforderungen der Anwender durch die Fähigkeiten des GPM (und dessen Beteiligten) vollständig „passend“ erfüllt werden können. Als „Over“-Engineering wird bezeichnet, wenn Fähigkeiten aufgebaut werden zu welchen kein Bedarf besteht. Beispielsweise ist es nicht zwingend notwendig BPMN^{Easy}-Geschäftsprozessmodelle in BPMN 2.0 zu transformieren, wenn kein Bedarf dazu besteht. Ein Vakuum besteht, wenn der Bedarf nicht durch bestehende Fähigkeiten gedeckt werden kann. Durch die flexiblen Startpunkte von BPM(N)^{Easy} kann dieser Zustand behoben bzw. vermieden werden. Keine Aktionen (vgl. Abbildung 58) werden durchgeführt, wenn weder Bedarf noch die Fähigkeit besteht.

Unabhängig davon in welcher Form die agile Vorgehensweise von BPM(N)^{Easy} angewendet wird, muss beachtet werden, dass die Vorgehensweise mehrfache Abstimmungen (z.B. das Daily zur täglichen Abstimmung) aller Prozessbeteiligten vorschreibt (vgl. Abschnitt 4.2). Hierdurch kann es zu einem vermeidlichen Mehraufwand in Vergleich zu anderen (traditionellen) Vorgehensweisen kommen. Mehraufwand beschreibt im Allgemeinen, Aufwand, welcher nicht unmittelbar Nutzen erzeugt. Die regelmäßigen Abstimmungen sind notwendig, um Änderungen von Anforderungen zu diskutieren bzw. Qualitätskontrollen durchzuführen (vgl. Quality Gates, Abschnitt 4.2.4). Des Weiteren werden die Inhalte der nächsten Iterationen geplant bzw. die Aufgabenverteilung innerhalb der Projektbeteiligten agil festgelegt. Um diesen Mehraufwand zu begrenzen oder diesen selbst in direkten Nutzen umzuwandeln ist eine zielgerichtete und zeitlich eingeschränkte Durchführung aller Abstimmungen notwendig. Insbesondere muss, zum Beispiel durch den BPM(N)^{Easy}-Master, auf die Zeitdauer der einzelnen Abstimmungen geachtet werden. Die tägliche Abstimmung sollte beispielsweise nicht länger als 15 Minuten benötigen (vgl. Abschnitt 2.3.2). Die hohe Abstimmungsfrequenz begünstigt insbesondere die Interaktion und Synchronisation der einzelnen Prozessbeteiligten. Zudem sind durch die Anwendung der agilen Vorgehensweise alle Aktivitäten des Geschäftsprozessmanagements für jeden Prozessbeteiligten ständig transparent. Eine kontinuierliche Analyse (z.B. durch Anwenderfeedback) unterstützt hierbei die Aufdeckung von möglichen Optimierungspotentialen. Der neugenerierte Bedarf kann dann wiederum in einer neuen BPM(N)^{Easy}- Iteration gestillt werden.

4.3. Entwicklung einer gebrauchssprachlichen Modellierungssprache (Z2.1)

In diesem Abschnitt wird die Modellierungssprache BPMN^{Easy} beschrieben (vgl. [MW13][MOW14][KMW14]). „Die Grenzen meiner Sprache bedeuten die Grenzen meiner Welt“. Dieses Zitat von Wittgenstein kann auch mit Geschäftsprozessmodellierungssprachen und deren Anwendung verbunden werden (vgl. Abschnitt 1.2). Nur wenn alle Prozessbeteiligten das gleiche Verständnis in Bezug auf die einzusetzende Modellierungssprache besitzen, kann eine erfolgreiche Kommunikation und spätere Ausführung der Geschäftsprozesse erzielt werden. Beim Einsatz von existierenden Modellierungssprachen (vgl. Abschnitt 3.2.2) kommt es häufig zu Problemen (vgl. [BKO10][MRI07][MNA07][GHA11][Ot12][Mo13][SS12][WGK13]). Ist zum Beispiel die eingesetzte Modellierungssprache für einen Anwender, aufgrund eines komplexen Elementsets, nicht verständlich, können erfasste Inhalte (z.B. Aktivitätsbeschreibungen in einem Geschäftsprozess) falsch oder gar nicht verstanden werden.

Die in dieser Arbeit eingeführte Modellierungssprache BPMN^{Easy} zielt auf eine intuitive Anwendbarkeit. Hierbei steht BPMN^{Easy} nicht im Widerspruch zu Wittgenstein. Durch die Möglichkeit der gebrauchssprachlichen Anreicherung während der gebrauchssprachlichen Modellierung, wie zum Beispiel das Anhängen von gebrauchssprachlichen Sprachnotizen per Audioaufnahme) werden „die Grenzen“ der einzelnen Prozessbeteiligten iterativ im Rahmen der agilen Vorgehensweise ausgedehnt.

BPMN^{Easy} basiert auf zwei Kernelementen: 1) auf einem BPMN 2.0-konformen grafischen Elementset, 2) auf Mediafiles, zur Annotation und Speicherung von Informationen in multimedialer Form. Der BPMN 2.0 Standard (vgl. Abschnitt 3.2.2) wurde als Grundlage von BPMN^{Easy} gewählt, da dieser heutzutage weit verbreitet und anerkannt ist [A113], wodurch wiederum eine breite Verwendung von BPMN^{Easy} begünstigt wird. Ein Nachteil des BPMN 2.0 Standards ist die hohe Anzahl unterschiedlicher BPMN 2.0-Modellierungselemente und deren Interpretationsmöglichkeiten. Die intuitive Modellierung mit BPMN 2.0 ist nur unzureichend möglich. Insbesondere Anwender ohne Erfahrung im Kontext des Geschäftsprozessmanagements benötigen einen intuitiven Zugang zu einer Modellierungssprache (vgl. [Mo09][Sc12, S.57f][He12a, S.149f][Go13]).

BPMN^{Easy} verdichtet das komplexe Elementset und verwendet ausschließlich grundlegende Elemente von BPMN 2.0 („Basis BPMN Modeling Elements“, vgl. [OM11b, S.26f]). Das reduzierte BPMN^{Easy} Elementset dient einzig einem möglichst einfachen Verständnis der BPMN^{Easy}-Modelle. Zur weiteren Vereinfachung können mit BPMN^{Easy} ausschließlich Geschäftsprozesse vom BPMN 2.0-Modelltyp „private“ modelliert werden (keine Darstellung von Kollaborationen und Choreographien, vgl. [OM11b, S.39ff]). Das zweite Kernelement, die Mediafiles, ermöglichen das Annotieren von Video- und Audiosequenzen, Bildern oder anderen Dokumenten an die grafischen BPMN^{Easy}-Modellierungselemente. Durch die Annotation werden die Geschäftsprozessmodelle zu multimedialen Geschäftsprozessmodellen erweitert¹⁸¹. Insbesondere die Rekonstruktion von Informationen aus BPMN^{Easy}-Modellen in Gebrauchssprache wird durch die Kombination von reduziertem Elementset und multimedialen Annotationen unmittelbar unterstützt.

Mit BPMN^{Easy} wird das Ziel verfolgt, allen Beteiligten (z.B. Anwender, Modellierer, IT-Experten) – insbesondere für die Ersterfassung/Erstbetrachtung von Geschäftsprozessmodellen – eine intuitiv und gebrauchssprachlich einsetzbare Modellierungssprache zur Verfügung zu stellen. Der Begriff der Intuition ist in der Literatur vielfältig beschrieben. Nach Fischbein [Fi87, S. 14] ist Intuition unter anderem durch Selbstevidenz charakterisiert. Selbstevidenz definiert einen Zustand, in dem z.B. eine Handlung als richtig empfunden wird, ohne dass diese begründet werden muss. Weigend beschreibt Intuition als eine Handlung die „(...) schnell und aus einem Gefühl heraus erfolgt.“ [We07, S.14]. Im Kontext der Geschäftsprozessmodellierung bedeutet das, dass Anwender direkt „die richtigen“ Elemente zur Modellierung einsetzen bzw. die Elemente in einem Geschäftsprozessmodell unmittelbar „richtig“ verstehen. Krenzin [Kr08, S.44] definiert Intuition als „(...) das Erfahrungswissen eines Individuums und die Fähigkeit zur Erfassung, Erkenntnis und Wahrnehmung von Sinnzusammenhängen, Ereignissen und Situationen (...).“. Zusätzlich definiert Pommeranz [Po11, S.94] Intuition als „(...) ein inneres Wissen, das sich als Schnittmenge von Wissen und Erfahrung ergibt.“. Rationale und emotionale Aspekte werden durch eine „Synthese zwischen Denken und Fühlen“ betrachtet¹⁸². BPMN^{Easy} verfolgt hierbei die Annahme,

¹⁸¹ Vgl. des Weiteren ähnliche Ansätze aus den Bereichen der Softwareentwicklung und des Geschäftsprozessmanagements, beispielsweise [Ch89] (Soft System Methodology), [Ch89][MRR10] (Usage of Labels and Icons) und [BPB13] (Wireframes in der agilen Softwareentwicklung).

¹⁸² Für weitere Ausführungen zum Begriff der Intuition sei auf [Fi87][Po11][Si11] verwiesen.

dass das Erfahrungswissen, welches jedem Individuum durch die Gebrauchssprache zur Verfügung steht, zur Geschäftsprozessmodellierung ausreichend ist. Eine weitere notwendige Eigenschaft ist die formale Beschreibbarkeit einer Modellierungssprache (vgl. [ATW03][HV01]). Nur durch eine formal eindeutige Beschreibung von Sachverhalten können Ineffizienzen (z.B. Syntaxfehlern) bei der Modellierung vermieden werden [Hi02b]. Des Weiteren können durch die Möglichkeit der formalen Speicherung von Geschäftsprozessmodelle, diese bei Bedarf weiter angereichert und als Basis der Geschäftsprozessautomatisierung verwendet werden. Mediafiles sind hierbei als eigenständige Datenobjekte verknüpft, sodass die formale Beschreibung (z.B. durch ein nicht-formalisierbare Audiosequenz) nicht negativ beeinflusst wird.

Die Abbildung 59 skizziert¹⁸³ das BPMN^{Easy} Klassendiagramm, welches als Grundlage der nachfolgenden Formalisierung dient. Die Elemente, deren Bezeichnungen und Beziehungen werden auf Basis der BPMN 2.0-Spezifikation (vgl. [OM11b, S.146f, S.151f, S.157f, S.205f, S.288f] vereinfacht dargestellt.

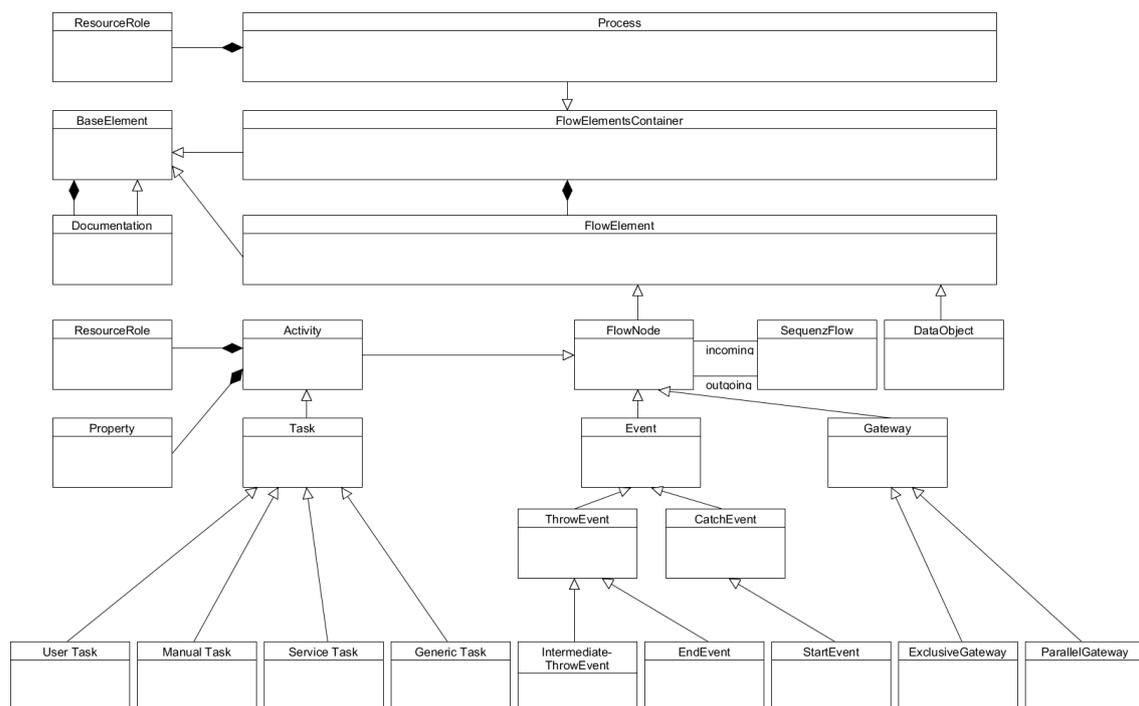


Abbildung 59: BPMN^{Easy} Klassendiagramm (UML)

¹⁸³ Für eine ausführliche Dokumentation aller BPMN 2.0 Klassen sei auf die BPMN 2.0-Spezifikation verwiesen [OM11b, S.146f, S.151f, S.157f, S.205f, S.288f]. Abbildung 59 stellt die Hauptklassen vor BPMN^{Easy} dar.

Die in Abbildung 59 dargestellten Klassen sind durch zwei Arten von gerichteten Pfeilen verbunden: Dem Pfeil als Symbol einer Komposition (Pfeil mit schwarzer Raute als Spitze) und dem Pfeil als Symbol einer Vererbung (Pfeil mit weißem Dreieck als Spitze). Die Komposition symbolisiert eine Zusammensetzung eines Ganzen aus einer Menge von abhängigen Einzelteilen (vgl. [OM11a]). Beispielsweise besitzt ein BPMN^{Easy} Geschäftsprozess („Process“-Klasse) ein Element „ResourceRole“. Über die „ResourceRole“ wird definiert, welche Ressource den Geschäftsprozess ausführt oder für diesen verantwortlich ist. Ressourcen können beispielsweise Individuen, Gruppen oder Organisationen sein.

Durch die Vererbung erhalten unterliegende Klassen die Eigenschaften der Oberklasse, wodurch eine hierarchische Gliederung möglich wird. Die „FlowElementsContainer“-Klasse vererbt beispielsweise deren Eigenschaften an die „Process“-Klasse. Sowohl die „FlowElementsContainer“-Klasse, als auch die „FlowElement“-Klasse sind abstrakte Klassen und dienen als Container für alle verwendeten Elemente in einem Geschäftsprozessmodell. Durch die abstrakte Klasse „BaseElement“ wird die Verbindung zur Klasse „Documentation“ erstellt. Die Klasse „Documentation“ ermöglicht das Speichern von zusätzlichen Attributen an einem Element (z.B. Kosten, Durchlaufzeit,...). Die weiteren Elemente werden in den nachfolgenden Abschnitten (vgl. Abschnitte 4.3.1-4.3.5) beschrieben.

Die Persistierung der Geschäftsprozessmodelle wird unter der Berücksichtigung einer XML Schema Definition (XSD) durchgeführt. Die XSD beschreibt die Regeln, welche bei der Generierung der zur Persistierung benötigten XML eingehalten werden müssen (vgl. [St14]). Diese valide Speicherung stellt sicher, dass die BPMNEasy-Modelle softwareübergreifend ausgetauscht werden können, sofern die Drittsoftware ebenso dasselbe XSD verwendet (vgl. Schema – Ausprägung, Kap. 2). BPMN^{Easy} ist eine Teilmenge von BPMN 2.0. Zur Definition von BPMN^{Easy} werden daher die XSD-Dokumente¹⁸⁴ verwendet: BPMN20.xsd, BPMNDI.xsd, DC.xsd und DI.xsd und Semantic.xsd. Das Semantic.xsd-Dokument enthält hierbei die Definition aller zur

¹⁸⁴Vgl. BPMN 2.0 XSDs:

<http://www.omg.org/spec/BPMN/20100501/BPMN20.xsd>,
<http://www.omg.org/spec/BPMN/20100501/BPMNDI.xsd>,
<http://www.omg.org/spec/BPMN/20100501/DC.xsd>,
<http://www.omg.org/spec/BPMN/20100501/DI.xsd>,
<http://www.omg.org/spec/BPMN/20100501/Semantic.xsd>

Modellierung verfügbaren Elemente in BPMN 2.0. Die Tabelle 32 zeigt einen Ausschnitt aus der Semantic.xsd zur Beschreibung eines BPMN 2.0-End-Events.

Tabelle 32: Ausschnitt (End-Event) aus der BPMN 2.0 Semantic.xsd

```
<xsd:element name="endEvent" type="tEndEvent"
substitutionGroup="flowElement"/>
<xsd:complexType name="tEndEvent">
<xsd:complexContent>
<xsd:extension base="tThrowEvent"/>
</xsd:complexContent>
</xsd:complexType>
```

Über die Definition wird zum Beispiel definiert, zu welchen Oberklassen ein Endereignis (engl. Event) gehört. Ein Endereignis gehört zu der Gruppe „FlowElement“ und erbt die Eigenschaften der Klasse „ThrowEvent“. Um eine Unterscheidung von BPMN^{Easy}- und BPMN 2.0 Geschäftsprozessmodellen zu ermöglichen, wird die Semantic.xsd, um die in BPMN^{Easy} erlaubten Elemente, verdichtet.

Softwarewerkzeuge, wie beispielsweise BPM Touch, erzeugen auf Basis dieser XSD-Dokumente konkrete XML-Dateien von modellierten BPMN^{Easy} Geschäftsprozessen. In Tabelle 33 ist ein Ausschnitt einer XML-Datei dargestellt.

Tabelle 33: BPMN^{Easy} -XML-Ausschnitt (erstellt mit BPM Touch)

```
<process id="_5a4adb25-d6d1-41b2-99e2-9ce9d6e5e222"
name="Antragsprozess" isExecutable="false" processType="Private">
<documentation>
<bpmtouch author="Peter Wiedmann"/>
</documentation>
<startEvent id="_a237e7c8-513c-4856-82c5-a0d6958e5410"
name="startEvent"/>
<sequenceFlow id="_9c6a8d7e-52b8-4b19-b2fe-4d856922a96b"
sourceRef="_a237e7c8-513c-4856-82c5-a0d6958e5410"
targetRef="_ac46fba8-ab61-46b7-ac41-48132c336cb3"/>
<userTask id="_ac46fba8-ab61-46b7-ac41-48132c336cb3" name="Antrag
formulieren"/>
<sequenceFlow id="_e87c1f48-9a44-47a0-997e-2ea702eed7d"
sourceRef="_ac46fba8-ab61-46b7-ac41-48132c336cb3"
targetRef="_b6bac40e-7663-4079-8486-0b9da288269c"/>
<serviceTask id="_b6bac40e-7663-4079-8486-0b9da288269c"
name="Antrag speichern"/>
<sequenceFlow id="_fcd319c8-7fe2-4c23-9d2d-c81dffe7e4e5"
sourceRef="_b6bac40e-7663-4079-8486-0b9da288269c"
targetRef="_fef30d0d-0926-4791-b8ea-f78f83482031"/>
<endEvent id="_fef30d0d-0926-4791-b8ea-f78f83482031"
name="endEvent"/>
</process>
<bpmndi:BPMNDiagram>
<bpmndi:BPMNPlane bpmnElement="Minimal">
<bpmndi:BPMNShape bpmnElement="_a237e7c8-513c-4856-82c5-
a0d6958e5410">
```

```

    <dc:Bounds height="50" width="50" x="35" y="325"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_9c6a8d7e-52b8-4b19-b2fe-
4d856922a96b">
    <di:waypoint y="350" x="85"/>
    <di:waypoint y="350" x="120"/>
    <di:waypoint y="350" x="120"/>
    <di:waypoint y="350" x="155"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_ac46fba8-ab61-46b7-ac41-
48132c336cb3">
    <dc:Bounds height="100" width="150" x="155" y="300"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_e87c1f48-9a44-47a0-997e-
2ea702eed7d">
    <di:waypoint y="350" x="305"/>
    <di:waypoint y="350" x="330"/>
    <di:waypoint y="350" x="330"/>
    <di:waypoint y="350" x="355"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_ac46fba8-ab61-46b7-ac41-
48132c336cb3">
    <dc:Bounds height="100" width="150" x="155" y="300"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNShape bpmnElement="_b6bac40e-7663-4079-8486-
0b9da288269c">
    <dc:Bounds height="100" width="150" x="355" y="300"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_fcd319c8-7fe2-4c23-9d2d-
c81dffe7e4e5">
    <di:waypoint y="350" x="505"/>
    <di:waypoint y="350" x="546"/>
    <di:waypoint y="350" x="546"/>
    <di:waypoint y="350" x="587"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_b6bac40e-7663-4079-8486-
0b9da288269c">
    <dc:Bounds height="100" width="150" x="355" y="300"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNShape bpmnElement="_fef30d0d-0926-4791-b8ea-
f78f83482031">
    <dc:Bounds height="50" width="50" x="587" y="325"/>
  </bpmndi:BPMNShape>
</bpmndi:BPMNPlane>
</bpmndi:BPMNDiagram>
</definitions>

```

Das XML besteht aus zwei Teilen. Der obere Teil („<process> </process>“) enthält die einzelnen Elemente und deren Eigenschaften. Das Beispiel enthält hierbei zwei Events (Start, Ende), zwei Tasks (User Task, Service Task) und drei Sequenzen, die zur Verbindung der anderen Elemente dienen. Der untere Teil („<bpmndi:BPMNDiagram> </bpmndi:BPMNDiagram>“) enthält Daten, welche zur grafischen Darstellung der Elemente notwendig sind (z.B. x-y-Koordinaten). Die Abbildung 60 stellt das BPMN^{Easy}-Geschäftsprozessmodell dar.

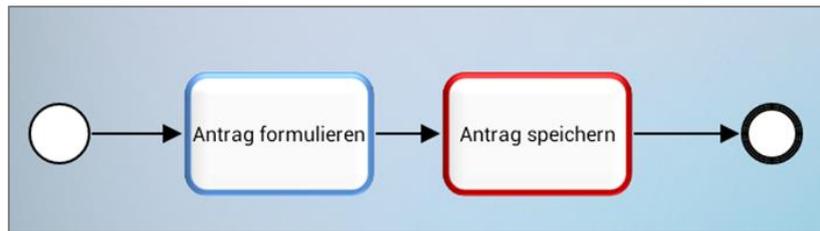


Abbildung 60: Einfaches BPMN^{Easy} Modell (erstellt mit BPM Touch)

Die farbliche Kennzeichnung der Umrandung von Tasks wird nicht in der XML-Datei hinterlegt. Die visuelle Markierung ist eine optional einzuhaltende Konvention in BPMN^{Easy} (vgl. Abschnitt 4.3.3).

4.3.1. Ereignisse

Zur Modellierung einer vordefinierten Zustandsänderung werden sogenannte Ereignisse¹⁸⁵ verwendet. BPMN^{Easy} unterscheidet zwischen drei Ereignistypen: Startereignis, Zwischenereignis und Endereignis. Die Abbildung 61 stellt die unterschiedliche Visualisierung der Ereignisse dar.



Abbildung 61: BPMN^{Easy} Ereignisse

Der Start eines Geschäftsprozesses wird durch das Startereignis (StartEvent) repräsentiert, welches durch einen „dünn umrandenden“ Kreis dargestellt ist. In BPMN^{Easy} besitzt jeder Geschäftsprozess genau ein Startereignis. Dieses Startereignis hat keine eingehenden Sequenzen. Zum Beispiel wird der Geschäftsprozess der Angebotserstellung mit dem Erhalt einer Kundenanfrage gestartet.

BPMN^{Easy} verwendet als grafisches Symbol eines Zwischenereignis (IntermediateThrowEvent) einen zweifach „dünn umrandenden“ Kreis. Die Zwischenereignisse können ohne nähere Spezifikation verwendet werden und weisen auf eine Statusänderung hin. Zwischenereignisse können ausschließlich zwischen Start- und Endereignissen modelliert werden. Des Weiteren besitzen Zwischenereignisse genau eine eingehende und genaue eine ausgehende Sequenz.

Das Endereignis (EndEvent) repräsentiert das Ende von Geschäftsprozessen. Wie die Startereignisse muss ein BPMN^{Easy}-Geschäftsprozessmodell mindestens ein Endereignis besitzen, wobei auch mehrere Endereignisse möglich sind. Zum Beispiel kann ein Angebotsprozess direkt nach einer negativen Bonitätsprüfung abgebrochen und beendet werden. Alternativ könnte der Angebotsprozess vollständig (positiv) durchlaufen werden. Zusätzlich besitzt ein Endereignis genau eine eingehende Sequenz. Das Endereignis ist durch einen „dick umrandenden“ Kreis markiert.

¹⁸⁵ Vgl. [OM11b, S.26]: „An Event [Ereignis] is something that „happens“ during the course of a Process (...).“.

4.3.2. Sequenzflüsse

Ein Sequenzfluss¹⁸⁶ wird zur Verbindung der BPMN^{Easy}-Elemente verwendet. Zusätzlich wird durch den Sequenzfluss die Reihenfolge der auszuführenden Aktivitäten bestimmt. Die Abbildung 62 stellt einen Sequenzfluss dar.



Abbildung 62: Sequenzfluss in BPMN^{Easy}

Die Verknüpfung der einzelnen Elemente unterliegt vordefinierten Regeln. Ein Starterereignis kann zum Beispiel nicht durch einen Sequenzfluss mit einem anderen Starterereignis verbunden werden. Die Tabelle 34 beschreibt die grundlegenden Verknüpfungsregeln. Das „+“-Zeichen zeigt, dass das Element der Zeile mit dem in der Spalte verknüpft werden kann.

Tabelle 34: Verknüpfungsregeln von BPMN^{Easy} Elementen (nach [OM11b, S.40f])

Von\Zu	○	▭	⊗ ⊕	⊙	◯
○		+	+	+	+
▭		+	+	+	+
⊗ ⊕		+	+	+	+
⊙		+	+	+	+
◯					

Zu beachten ist, dass BPMN^{Easy} ausschließlich Sequenzflüsse erlaubt. Das Element „DataObject“, welches in BPMN 2.0 über sogenannte Assoziationen¹⁸⁷ (also nicht durch Sequenzflüsse) an andere Elemente angeheftet ist, wird in BPMN^{Easy} nicht zur Modellierung zugelassen.

¹⁸⁶ Vgl. [OM11b, S.27]: „A Sequence Flow is used to show the order that Activities will be performed in a Process (...).”

¹⁸⁷ Vgl. Abschnitt 3.2.2 und [OM11b, S.65f]: „An Association is used to associate information and Artifacts with Flow Objects. (...).”

Stattdessen dient das „DataObject“ zur (technischen) Speicherung der Mediafiles, sofern diese zur gebrauchssprachlichen Anreicherung der Geschäftsprozessmodelle genutzt werden (vgl. Abschnitt 4.3.5). Eine visuelle Darstellung des „DataObjects“ ist keine notwendige Anforderung von BPMN^{Easy}.

4.3.3. Aktivitäten

Basierend auf der BPMN 2.0 Spezifikation ist die Aktivität ein weiteres Element in einem Geschäftsprozess. Eine Aktivität wird entweder durch eine atomare Aufgabe (Task) oder Subprozess repräsentiert [OM11b, S.248]. BPMN^{Easy} erlaubt ausschließlich die Verwendung von atomaren Aufgaben, also Aktivitäten, die¹⁸⁸ nicht weiter verfeinert werden. BPMN^{Easy} unterscheidet vier Tasktypen: manuelle, benutzergesteuerte, automatisierte und generische Tasks. Die Taskelemente werden zusätzlich zu ihrer Kennzeichnung mit Hilfe eines Namensattribut (z.B. „User Task“) durch ein farbigen umrahmtes Rechteck dargestellt, vgl. Abbildung 63.



Abbildung 63: BPMN^{Easy} Aktivitäten

Die farbliche Markierung unterstützt den intuitiven Umgang aller Beteiligten mit den Geschäftsprozessmodellen. Speziell kognitive Funktionen (z.B. die Wahrnehmung von etwas) des Menschen werden durch Farben unterstützt [GHA11]. Nach Moody [Mo09] sollen Farben nicht für eine grundlegende Unterscheidung zwischen Elementen verwendet werden (z.B. aufgrund von Farbenblindheit). Dennoch kann zum Beispiel durch die farbliche Markierung auf „den ersten Blick“ identifiziert werden, ob ein modellierter Geschäftsprozess vollständig automatisiert ist (d.h. es befinden sich ausschließlich „rote“ Tasks im Geschäftsprozessmodell).

User Tasks werden zum Modellieren von Aktivitäten genutzt, in welchen der ausführende Anwender direkt mit einem IT-System interagiert. Beispielsweise ist die Task „Angebot erstellen“ eine User Task, sofern diese mit Hilfe z.B. einer Textverarbeitungssoftware ausgeführt wird. Manuelle Tasks beschreiben Aktivitäten, die ohne IT-Unterstützung

¹⁸⁸ Vgl. [OM11b, S.30]: „A Task is an atomic Activity that is included within a Process. A Task is used when the work in the Process is not broken down to a finer level of Process detail.”

ausgeführt werden. Zusätzlich zu den manuellen und teilautomatisierten Tasks kann das Service Task-Element zur Modellierung von automatisierten Tasks angewendet werden. Automatisierte Tasks sind Tasks, welche ohne menschliche Aktivität durchgeführt werden, z.B. die automatisierte Bonitätsprüfung eines Kunden innerhalb eines Angebotsprozesses. Ist zum Zeitpunkt der Modellierung noch nicht identifiziert, welcher Tasktyp genutzt werden soll, wird die generische Task ausgewählt.

Zu jeder Task muss eine Bezeichnung vergeben werden, welche innerhalb des Task-Elements dargestellt wird (vgl. Abbildung 63). Zusätzlich können weitere Eigenschaften (Properties) an eine Task angeheftet werden, wie beispielsweise Informationen zur Durchlaufzeit, Liegezeit, Kosten und Rollen. Im Gegensatz zu BPMN 2.0 wird in BPMN^{Easy} auf das Element Lane verzichtet. Die sogenannten Lanes¹⁸⁹ dienen zur Strukturierung der Elemente in einem Geschäftsprozessmodell. BPMN 2.0 definiert die Bedeutung¹⁹⁰ der Lanes nicht. Zum Beispiel können diese nach internen Organisationseinheiten oder IT-Systemen benannt werden. Geschäftsprozesse mit mehreren Verzweigungen und einer Vielzahl von unterschiedlichen ausführenden Rollen können hierbei unzureichend visualisiert werden. Insbesondere die dynamische Zuweisung von Rollen während der Ausführung kann zu Problemen in der grafischen Darstellung führen (vgl. zusätzlich Abschnitt 4.3.4, Beispiel - Abbildung 65). In BPMN^{Easy} wird die Rolleninformation mit Hilfe der Klasse ResourceRole abgespeichert und nicht direkt als grafisches Element dargestellt.

4.3.4. Gateways

Gateways dienen zur Steuerung des Sequenzflusses. BPMN^{Easy} definiert ausschließlich zwei Typen: Das exklusive Gateway und das parallele Gateway. Die Gateways sind ausschließlich datenbasiert, d.h. der Sequenzfluss wird auf Basis der gewählten Bedingungen bestimmt. Die mit BPMN 2.0 beschriebenen „Ereignisbasierten Gateways“ (der Sequenzfluss wird auf Basis eintreffender Ereignisse bestimmt) werden in BPMN^{Easy} nicht unterstützt.

Die Abbildung 64 stellt die zwei möglichen Gateways in BPMN^{Easy} dar.

¹⁸⁹ In der Literatur im Allgemeinen als Swimlane bezeichnet.

¹⁹⁰ Vgl. [OM11b, S.305]: „The meaning of the Lanes is up to the modeler. BPMN does not specify the usage of Lanes. (...)”.

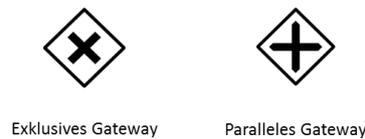


Abbildung 64: BPMN^{Easy} Gateways

Das exklusive Gateway (ExclusiveGateway, „XOR“) wird zur Modellierung einer „Entweder-oder“-Entscheidung verwendet. Ein eingehender Sequenzfluss wird somit zu mehreren (mindestens zwei) alternativen Sequenzflüssen gesplittet, wobei nur genau eine der Alternativen zur Ausführung gebraucht werden darf. Im Gegensatz zu BPMN 2.0 (vgl. [OM11b, S.290f]) müssen ausgehende Sequenzflüsse nicht zwingend bezeichnet werden.

Werden Sequenzflüsse durch ein exklusives Gateway getrennt, müssen diese wiederum durch ein exklusives Gateway zusammengeführt werden. Die Abbildung 65 zeigt ein Beispiel zur Anwendung eines exklusiven Gateway in BPMN^{Easy}.

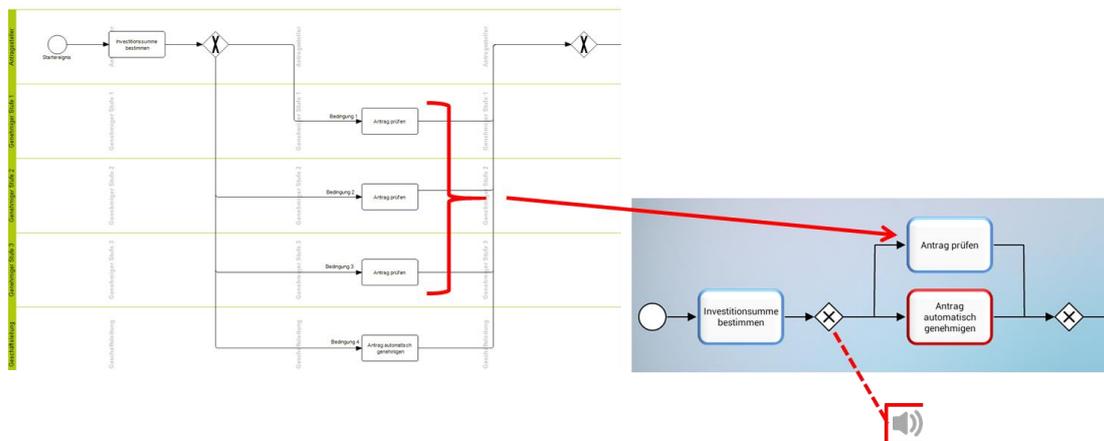


Abbildung 65: Vergleich BPMN 2.0 vs. BPMN^{Easy}-Geschäftsprozessmodell

Das Geschäftsprozessmodell in Abbildung 65 beschreibt einen Ausschnitt aus einem Investitionsantragsprozess. Durch die in einer ersten Aktivität (User Task) bestimmte Investitionssumme wird je nach Höhe der Summe (z.B. bis 1000 Euro => Genehmiger 1, bis 2000 Euro => Genehmiger 2,...) der weitere Sequenzfluss bestimmt. Zur Darstellung des Geschäftsprozesses kann die Aktivität „Investitionsantrag prüfen“ mehrfach modelliert und auf die unterschiedlichen Rollen (d.h. mit einer Swimlanes-Darstellung) verteilt werden. In BPMN^{Easy} werden die Rollen als Attribute an die Aktivität gebunden und gespeichert. Eine grafische Visualisierung der Rollen ist nicht zwingend nötig. Das

exklusive Gateway besitzt zusätzlich eine Audiosequenz als Annotation, mit der die Entscheidungsregeln gebrauchssprachlich wiedergegeben werden können.

Das parallele Gateway (ParallelGateway, „AND“) entspricht dem logischen „UND“. Sequenzflüsse werden hierdurch parallelisiert. Hierbei hat das parallele Gateway mindestens zwei ausgehende Sequenzflüsse. Parallelisierte Sequenzflüsse müssen wiederum durch ein zusammenführendes paralleles Gateway synchronisiert werden. Da parallele Sequenzflüsse nicht zwingend parallel ablaufen [FRH10] müssen, „wartet“ das zusammenführende parallele Gateway bis alle eingehenden Sequenzflüsse dieses erreicht haben (vgl. [OM11b, S.293]).

4.3.5. Gebrauchssprachliche Annotation

Eine traditionelle Anforderung an die Geschäftsprozessmodellierung ist die Möglichkeit physische oder virtuelle Objekte, welche während der Ausführung eines Geschäftsprozesses erstellt, manipuliert oder gebraucht werden, darzustellen [OM11b, S.202]. In BPMN^{Easy} wird die zweidimensionale grafische Darstellung von BPMN 2.0 durch die Möglichkeit der Annotation von Mediafiles erweitert. Mediafiles sind Datenobjekte (Data Objects), welche Informationen in Form von z.B. Bildern, Video- oder Audiosequenzen speichern. Softwarewerkzeuge, wie beispielsweise BPM Touch, ermöglichen die direkte Darstellung der Mediafiles, sodass eine Interaktion mit dem Geschäftsprozessmodell selbst ermöglicht wird (z.B. durch Abspielen eines Videos). Die Geschäftsprozessmodelle werden durch Mediafiles gebrauchssprachlich angereichert. Anwender formulieren ihre Anforderungen gebrauchssprachlich, wobei die Erfassung der Informationen unmittelbar durch z.B. eine Videoaufnahme gespeichert wird. Die Informationen¹⁹¹ können somit zu einem späteren Zeitpunkt unter anderem zu Schulungszwecken¹⁹² verwendet werden. Insbesondere die Digitalisierung von Geschäftsprozessen (z.B. digitale Dokumentation, digitale Zusammenarbeit,...) kann durch die Multimediaannotation unterstützt werden. Zusätzlich können Datenobjekte, wie zum Beispiel Rechnungsdokumente oder Antragsformulare als digitale Dokumente oder Bilder angeheftet werden. Diese Datenobjekte dienen dann entweder als direkte

¹⁹¹ Vgl. Diskussion zum Begriff Multimedia in Abschnitt 4.2.3.

¹⁹² An dieser Stelle soll als Beispiel auf den Bereich des eLearnings verwiesen werden. Die Annotation von Multimedia wird hierbei z.B. zur digitalen Aufbereitung von Lerninhalten erfolgreich eingesetzt, vgl. [Ma04][Sc06][Di11][Uf11].

Vorlage für die Ausführung einer Aktivität oder als Grundlage einer Anreicherung hinsichtlich der Geschäftsprozessautomatisierung. IT-Experten können hierbei auf Grundlage z.B. handgezeichneter und abfotografierter Bilder Benutzeroberflächen für einen User Task (teilautomatisierte Aktivität) implementieren. Die Tabelle 35 listet die unterschiedlichen Typen von Mediafiles auf.

Tabelle 35: Multimedia-Annotationen in BPMN^{Easy} - Beispiele

Typ	Möglicher Verwendungszweck
Audiosequenz	Komplexe Entscheidungssachverhalte können gebrauchssprachlich und somit für alle Beteiligten nachvollziehbar dokumentiert werden.
Videsequenz	Manuelle Tasks, also Tasks in denen beispielsweise in Handarbeit ein Produkt gefertigt wird, können durch Videosequenzen angereichert werden.
Bild	<p>Bereits bearbeitete Antragsformulare können als Bild an eine Aktivität annotiert werden und als Muster dienen. Ebenso können, während der Implementierung von Geschäftsprozessapplikationen, zum Beispiel Handzeichnungen von Anwendern, die zukünftige Benutzeroberflächen¹⁹³ darstellen, annotiert werden.</p> <p>Diese dienen dann als Vorlage der softwaretechnischen Umsetzung.</p>
Sonstiges (z.B. Dokumente)	Beispielsweise können Richtlinien, Dokumentvorlagen etc. direkt an eine Aktivität annotiert werden.

¹⁹³ Die Gestaltung von Benutzeroberflächen bedarf einer engen Zusammenarbeit zwischen Anwender und IT-Experten, vgl. [Ka12, S.189]: „Die fachlichen Aspekte der Frontend-Modellierung beziehen sich auf die grundlegende Bedienung der zu realisierenden Benutzeroberfläche. Diese muss in enger Zusammenarbeit mit dem Fachbereich erstellt werden. Darunter fällt beispielsweise die Modellierung der möglichen Interaktionen, die zwischen den Benutzern und der Anwendung auftreten können. Auch die Festlegung der Struktur von zu realisierenden Seiten sowie die Festlegung der verschiedenen Benutzerrollen inklusive deren Berechtigungen fallen in diesen Bereich. Ein weiterer Aspekt ist die Navigation, die bereits bei der fachlichen Modellierung definiert werden soll. Ebenso müssen neben der Navigation weitere wichtige Aspekte von Benutzeroberflächen wie etwa Personalisierung und Mehrsprachigkeit berücksichtigt werden.“

Die Mediafiles werden mit Hilfe der BPMN Klasse „DataObject“ abgespeichert. Die Tabelle 36 stellt einen XML-Ausschnitt eines solchen Datenobjekts dar.

Tabelle 36: Ausschnitt XML DataObject

```
<dataObject id="_2421dfd2-a79f-4c17-907f-d5802e8fb362"
name="//share/bilder/antragsformular.png">
  <dataState id="_478eed1d-98e9-4904-84d7-2c3646c0223b"
name="Vorlage" />
</dataObject>
```

Neben einer Objekt-ID wird in BPMN^{Easy} der Speicherort des Mediafiles abgespeichert. Insbesondere bei verteilten Ressourcen muss die Zugänglichkeit zu diesen Mediafiles gewährleistet sein. Des Weiteren kann das Attribut „dateState“ als Indikator für den Verwendungszweck genutzt werden. Wird beispielsweise ein Mediafile in Form eines Dokuments annotiert, kann dieses den Status „Vorlage“ besitzen. Die Bezeichnung des „dateState“ ist nicht durch BPMN^{Easy} definiert.

4.4. Werkzeug

Der dritte Bestandteil der Methode, das Werkzeug, (vgl. [Or12]) besitzt eine unterstützende Funktion. Beispielsweise können Werkzeuge zur automatisierten Überprüfung von Sprache (z.B. auf die Korrektheit eines Programmcodes) eingesetzt werden. Der Begriff des Werkzeugs wird hierbei in der Literatur unterschiedlich verwendet¹⁹⁴, wobei drei Aspekte zur Charakterisierung von Werkzeugen gebraucht werden können (vgl. [WR85][GM91]): 1) Funktionalität, 2) Spezifität, 3) Funktionsprinzipien. Demnach sind Werkzeuge Objekte, die „(...) nur durch ihre objektspezifische Verwendung ihren Zweck erfüllen. So stellt z.B. ein Hammer, dessen Funktionalität, d.h. dessen Rolle als Mittel im Rahmen eines Handlungszusammenhangs, einem potentiellen Werkzeugbenutzer nicht vertraut ist, für diesen kein Werkzeug dar, sondern lediglich einen Gegenstand.“ [GM91, S.354]. Des Weiteren besitzt ein Werkzeug eine Spezifität. Sowohl der Zweck als auch die Wirkung des Werkzeugs bei der Anwendung ist spezifisch. Beispielsweise kann mit einem Bohrer, welcher speziell für das Bohren in Holz entwickelt wurde, nur unzureichend in Beton gebohrt werden. Das Bohren von Holz ist dagegen problemlos möglich. Die Funktionsprinzipien beschreiben die Art der Verstärkung, Erweiterung und Externalisierung eines Menschen, mit Hilfe

¹⁹⁴ Für zusätzliche Diskussion zum Werkzeugbegriff sei auf [GM91, S.354ff] und [LG99, S.271ff].

eines Werkzeugs. Werkzeuge beeinflussen hierbei motorische, sensorische und intellektuelle Funktionsbereiche des Menschen. Zum Beispiel verstärkt der Einsatz eines Lichtmikroskops die Sehfähigkeit des Menschen (vgl. [GM91, S.355ff]).

Coy et al.[LG99][Or12][Co92] definieren Werkzeuge als „Dinge, mit denen wir das angestrebte Arbeitsergebnis hervorbringen, indem wir das Material bearbeiten. Dabei steht Werkzeug zwischen uns und dem Material.“ Der Beschreibung von Coy et al. folgend werden ex ante definierte Arbeitsergebnisse (z.B. Herstellung eines Motors) „nur“ durch den Einsatz von Werkzeugen erreicht. Des Weiteren wird vorausgesetzt, dass der Einsatz eines Werkzeugs das Existieren von Material (z.B. unbearbeitetes Eisen) besteht. Die Bearbeitung eines Materials wird auf mechanische bzw. physikalisch-chemische Art durchgeführt. Beispielsweise können Bohrer oder Fräsen zur mechanischen Bearbeitung von Eisen genutzt werden (vgl. [Ev98]).

Im Kontext der Softwareentwicklung kann der Begriff des Werkzeugs zum Begriff des Softwarewerkzeugs¹⁹⁵ erweitert werden. Dumke [Du03, S.147] ordnet „alle rechnergestützte[n] Hilfsmittel zur Unterstützung der Entwicklung und Wartung eines Software-Produkts.“ dem Begriff des Werkzeugs zu. Analog zum „klassischen“ Verständnis eines Werkzeugs (z.B. Bohrer, Hammer,...) dienen Softwarewerkzeuge als Hilfsmittel zur Erreichung eines vordefinierten Software-Produkts. Dumke nennt explizit die Bereiche der Entwicklung und Wartung von Software, wobei „Anwendersoftware“ (z.B. ein Textverarbeitungssoftware) nicht unter den Begriff des Softwarewerkzeugs fällt. Ebenso kann nach Dumke die Softwarewerkzeugunterstützung sowohl automatisiert als auch teilautomatisiert erfolgen. Rechenberg [Re85] erweitert den Begriff des Softwarewerkzeugs. Nach Rechenberg dienen Softwarewerkzeuge, neben dem Einsatz als Entwicklungs- und Programmierumgebung, zusätzlich zur Konfigurations- und Versionsverwaltung von Programmen. Unter anderem können Softwarewerkzeuge zur automatisierten Prüfung von Programmiercode oder zur Erstellung von Dokumentationen eines Software-Produkts (z.B. Funktionsumfang, Systemvoraussetzungen,...) genutzt werden.

In Bezug auf Geschäftsprozessmanagement werden Softwarewerkzeuge zur Unterstützung bei der Definition, Verwaltung, Anpassung, Ausführung und Auswertung von Geschäftsprozessen eingesetzt (vgl. Abschnitt 2.2.3). Zusätzlich zur Entwicklung

¹⁹⁵ Im Kontext der Softwareentwicklung wird auch der Begriff „Tool“ genutzt [Ku92].

von Software-Produkten (z.B. Software zur automatisierten Ausführung von Aktivitäten in einem Geschäftsprozess) dienen Softwarewerkzeuge hierbei beispielsweise zur Modellierung oder Simulation (z.B. zur Prüfung von Durchlaufzeiten eines Geschäftsprozess) von Geschäftsprozessmodellen. Heutzutage existiert eine Vielzahl solcher Softwarewerkzeuge (vgl. GPM-Studien/Marktüberblicke [Fr13][Br14] [Ad14]), wobei diese sich insbesondere in der Breite der Funktionen unterscheiden. Zum Beispiel stellt eine Vielzahl von Softwarewerkzeugen eine Entwicklungsplattform zur Automatisierung dar, andere können zur Erfassung und Dokumentation von Geschäftsprozessen genutzt werden¹⁹⁶ (vgl. Abschnitt 2.2.3).

Als konkrete Beispiele solcher Softwarewerkzeuge werden den Abschnitten 4.4.1 und 4.4.2 zwei ausgewählte Produkte beschrieben. In Abschnitt 4.4.1 wird die Axon.ivy BPM Suite vorgestellt¹⁹⁷. Die Axon.ivy BPM Suite kann als Softwarewerkzeug zur Unterstützung des gesamten Geschäftsprozessmanagementzyklus eingesetzt werden (z.B. Modellierung von Geschäftsprozessen). Des Weiteren kann die Axon.ivy BPM Suite dafür verwendet werden, um wiederum eigene Softwarewerkzeuge zu entwickeln (vgl. Abschnitt 5). In Abschnitt 4.4.2 wird mit BPM Touch ein Softwarewerkzeug beschrieben, welches zur mobilen Modellierung von Geschäftsprozessen angewendet werden kann.

Die Auswahl dieser Werkzeuge beruht auf mehreren Kriterien: Axon.ivy ist eine mehrfach ausgezeichnetes¹⁹⁸ und für die Forschung und Lehre freizugängliches Produkt, mit welcher auf Basis moderner Technologie die Modellierung und Automatisierung von Geschäftsprozess unterstützt wird. Zudem finden aktuelle Axon.ivy Implementierungsprojekte mehrheitlich erfolgreich agil statt, sodass der die Eignung und der Einsatz dieser Software im Rahmen der Dissertation nahe liegen. Das zweite ausgewählte Produkt, BPM Touch, unterstützt des Weiteren durch die bereitgestellten Funktionen, die flexible und intuitive Erfassung von Geschäftsprozessmodellen. BPM Touch ermöglicht hierbei die Modellierung mit BPMN^{Easy}, sodass neben der grafischen Modellierung von Geschäftsprozessmodellen, zusätzlich die direkte Erfassung gebrauchssprachlichen Annotationen in Form von z.B. Audio oder Video möglich ist.

¹⁹⁶ Vgl. Abschnitt 2.2.3.

¹⁹⁷ Die Anwendung der Axon.ivy BPM Suite und des BPM Touch wird in Abschnitt 5 und Abschnitt 6 beschrieben.

¹⁹⁸ Vgl. u.a. Ergebnisse der Fraunhofer Studien [Fr13] und [Ad14].

Auf Basis dieser Softwarewerkzeuge wurde der im Rahmen dieser Arbeit entwickelte Software-Prototyp implementiert, mit welchem User Feedback direkt während der Ausführung von Geschäftsprozessen erfasst werden kann (vgl. Abschnitt 5). Zusätzlich wurde zur Modellierung der Geschäftsprozesse mit der Modellierungssprache BPMN^{Easy} das BPM Touch Werkzeug eingesetzt. Insbesondere die ortsunabhängige effiziente Erfassung von Geschäftsprozessen erhöht die Chancen und Möglichkeiten der Interaktion und Synchronisation im Rahmen der BPM(N)^{Easy} Methode.

4.4.1. Axon.ivy BPM Suite

Die Axon.ivy BPM Suite ist ein Geschäftsprozessmanagementsystem, welches alle Phasen des Geschäftsprozessmanagementzyklus unterstützt. Axon.ivy basiert auf einer Client-Server-Architektur und ist in sechs Module untergliedert. Die unterschiedlichen Client-Applikationen werden abhängig vom Anwendungsszenario (z.B. Modellierung von Geschäftsprozessen) entweder über das Entwicklungsframework Eclipse¹⁹⁹, oder über einen Standard-Webbrowser aufgerufen. Erfasste und entwickelte Inhalte (z.B. Geschäftsprozessmodelle, User Dialoge...) können mehrsprachig gepflegt und angezeigt werden. Grundlegender Bestandteil der Axon.ivy Serverkomponenten ist der Apache Tomcat Webserver²⁰⁰.

Des Weiteren wird Axon.ivy in „fachliches“ (Schwerpunkt Modellierung) und „technisches“ (Schwerpunkt Automatisierung) BPM unterteilt. Die Abbildung 66 stellt die einzelnen Module und deren Einordnung in „fachliches“ und „technisches“ BPM dar

¹⁹⁹ <https://eclipse.org/>

²⁰⁰ <http://tomcat.apache.org/>

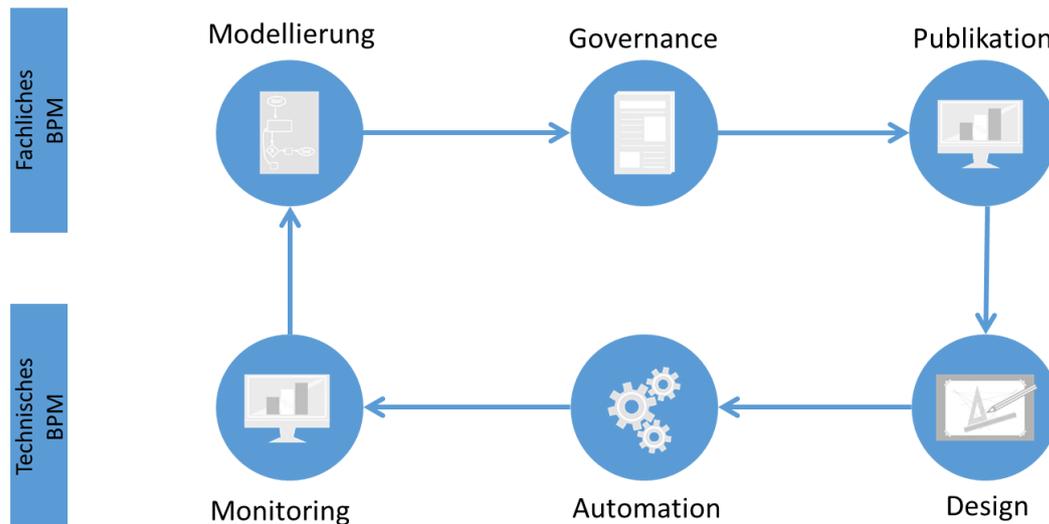


Abbildung 66: Darstellung der Axon.ivy BPM Suite Module [AX15a]

Das Modul Axon.ivy Modeler dient zur Modellierung von grafischen Modellen. Neben der Möglichkeit Geschäftsprozesse mit Hilfe von grafischen Modellierungssprachen zu erfassen (z.B. EPK, BPMN 2.0), kann der Axon.ivy Modeler beispielsweise zur Modellierung von Organigrammen und IT-Landkarten verwendet werden. Auf Basis einer Client-Server-Architektur werden die erstellten Modelle in einem zentralen Repository abgespeichert. Ein kollaboratives Arbeiten der Modellierer wird somit direkt unterstützt (z.B. Modelle werden automatisch für andere Modellierer gesperrt (Lesemodus), sobald ein Modellierer diese bereits bearbeitet). Die Abbildung 67 stellt den Axon.ivy Modeler in der Modellierungsperspektive dar.

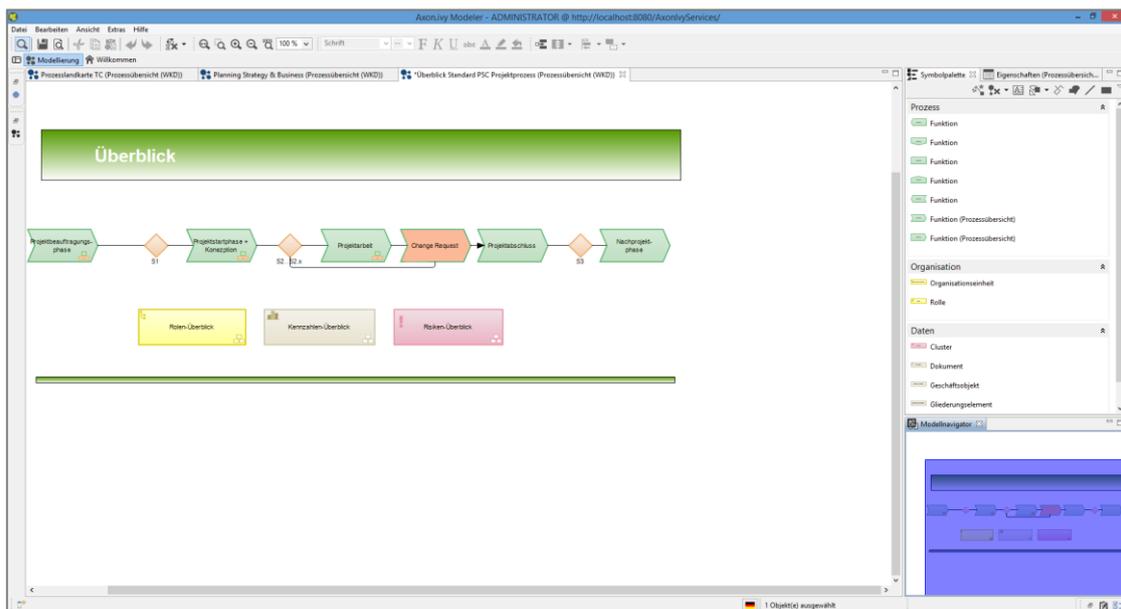


Abbildung 67: Screenshot - Axon.ivy Modeler

Neben dem Menü zur Benutzerführung (obere Leisten, vgl. Abbildung 67) und der zentralen Oberfläche zur Modellierung können zusätzliche Komponenten angezeigt werden. Beispielsweise dient die Symbolpalette als „Elementpool“ aus dem der Modellierer die für das Geschäftsprozessmodell notwendigen Elemente per Drag’n-&Drop ziehen kann. Des Weiteren dient eine kontinuierliche Konsistenzprüfung zur Kontrolle der Modellqualität. Hierbei werden ex ante definierte Regeln (z.B. Syntaxregel bzgl. der Modellierungssprache) überprüft. Werden Inkonsistenzen entdeckt, wird der Modellierer durch eine visuelle Warnung darauf hingewiesen. Ist ein Geschäftsprozessmodell aus Sicht des Modellierers vollständig erstellt, verknüpft (z.B. mit anderen relevanten Geschäftsprozessen) und geprüft, kann dieses freigegeben werden. Die Freigabe wird über das Governance Modul gesteuert. Im Standard muss für jedes Modell ein Verantwortlicher angegeben werden. Neben den anderen beliebigen Modellattributen (z.B. Angabe von Soll-Zeiten, Kosten,...) wird der Verantwortliche direkt als Attribut abgespeichert. Der durch das Governance Modul gesteuerte Genehmigungsprozess sperrt das zu freigebende Modell, sodass keine Änderungen mehr vorgenommen werden können, und benachrichtigt den Verantwortlichen über die Aufgabe einer neuen Freigabe. Zusätzlich zur strukturierten Freigabe von Geschäftsprozessen wird durch die Governance-Komponente das freigegebene Modell als PDF/A²⁰¹-Kopie revisionssicher abgespeichert. Die freigegebenen Modelle erscheinen im Axon.ivy Publisher. Das Portal kann per Standard-Webbrowser aufgerufen werden und wird von Endanwender als zentrale GPM-Informations- und Interaktionsplattform genutzt. Die Abbildung 68 stellt einen Screenshot des Portals dar.

²⁰¹ PDF/A ist ein Format zur Langzeitarchivierung nach ISO 19005-1:2005, vgl. http://www.iso.org/iso/catalogue_detail?csnumber=38920

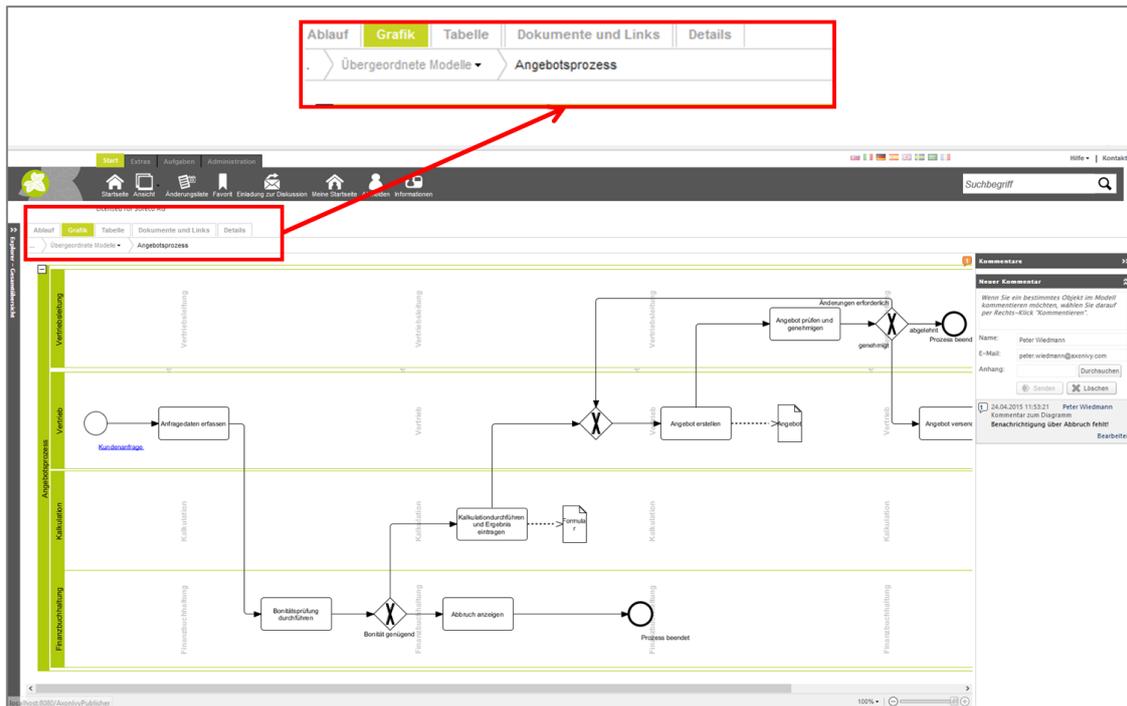


Abbildung 68: Screenshot Axon.ivy Publisher

Das Portal visualisiert alle freigegebenen Modelle. Verknüpfungen, zum Beispiel zwischen einer Prozesslandkarte und den unterliegenden Geschäftsprozessen, können per Klick geöffnet werden. Unterhalb der Benutzerführung wird der Pfad des aufgerufenen Modells visualisiert (vgl. Abbildung 68). Die Benutzerführung selbst enthält weitere Funktionen zur Bedienung des Portals, wie beispielsweise Login/Logout oder die Möglichkeit „Einladung zur Diskussion“ aufzurufen. Über „Einladung zur Diskussion“ können Anwender andere Anwender, per Direktlink zu einem Modell, zur Diskussion einladen. Des Weiteren können Kommentare zu jedem Modell erfasst werden (vgl. rechte Spalte, Abbildung 68). Sowohl die Funktion der Diskussionseinladung als auch die Feedbackfunktion dienen zur einfacheren und schnelleren Interaktion zwischen Anwendern und Modellierern bzw. Modellverantwortlichen. Beispielsweise können Fehler (z.B. es fehlt eine Aktivität im Geschäftsprozessmodell) unmittelbar mitgeteilt werden.

Die Anreicherung der Geschäftsprozessmodelle wird mit Hilfe des Axon.ivy Designers durchgeführt. Importierte oder neu erstellte BPMN 2.0 Geschäftsprozessmodelle werden hierbei um technische Informationen ergänzt, mit dem Ziel der Geschäftsprozessautomatisierung. Zum Beispiel können mit dem Axon.ivy Designer User Dialoge entwickelt werden, welche während der Ausführung zur Eingabe von

Geschäftsprozess-relevanten Daten dienen (z.B. Eingabe von Kontodaten bei einem Bestellprozess). Die Abbildung 69 zeigt die Modellierungsperspektive des Axon.ivy Designers. Vergleichbar mit dem Axon.ivy Modeler können neben dem Menü (obere Leisten, vgl. Abbildung 69) weitere Komponenten je nach Kontext angezeigt werden.

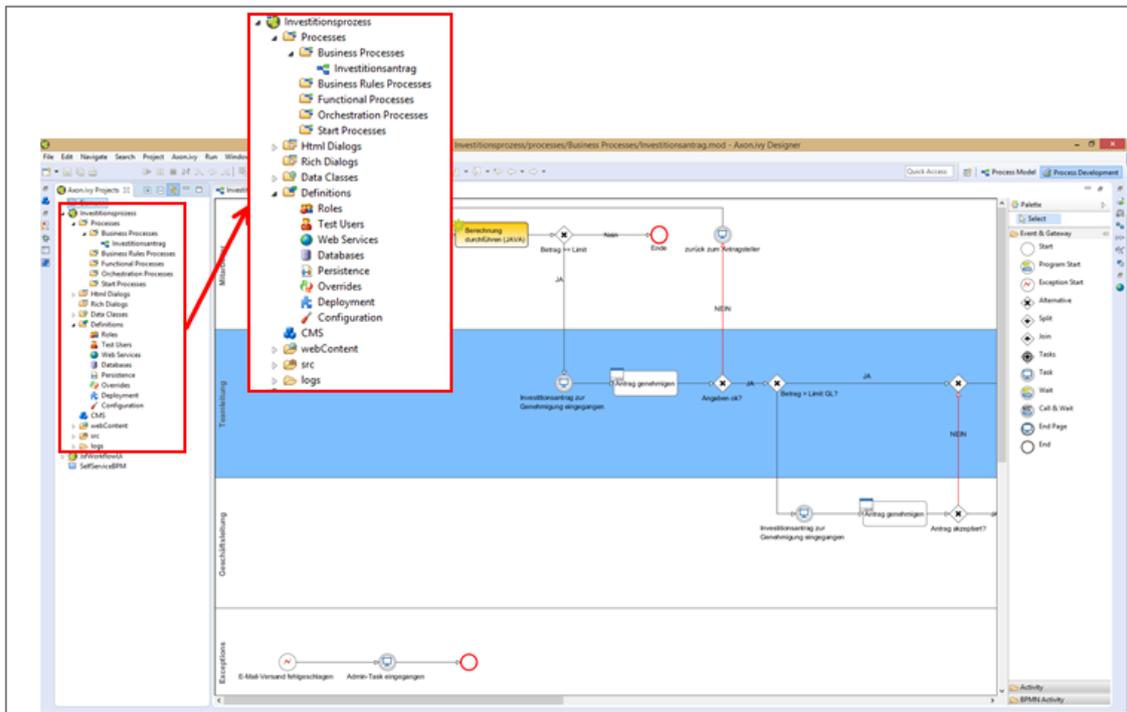


Abbildung 69: Screenshot - Axon.ivy Designer

Die zur Anreicherung relevanten Daten und Informationen werden in einem Axon.ivy Designer „Workspace“ gespeichert und verwaltet. Dieser „Workspace“ (vgl. linke Spalte, Abbildung 69) enthält beliebig viele Einzelprojekte, wobei jedes Projekt wiederum beliebig viele Geschäftsprozesse enthalten kann. Zusätzlich zur Modellierung der Geschäftsprozessmodelle können über die Projektstruktur weitere Elemente aufgerufen werden. Zum Beispiel können Daten, welche während der Ausführung einer Geschäftsprozessinstanz gespeichert werden sollen, definiert werden. Es können hierzu sowohl alle Standarddatentypen (z.B. String, Integer,...) ausgewählt werden, als auch eigene Datentypen erstellt werden. Zur Integration von Drittsystemen (z.B. ERP-, CRM-Systeme) können Konnektoren (z.B. Webservices) konfiguriert werden. Eine weitere Perspektive ermöglicht die Entwicklung von User Dialogen. User Dialoge dienen zur Interaktion mit dem Endanwender. Muss beispielsweise die Aktivität „Antrag ausfüllen“ teilautomatisiert werden, dient ein User Dialog zur Eingabe der Antragsdaten. Sowohl

Konnektoren, als auch User Dialoge werden zentral verwaltet und können projektübergreifend wiederverwendet werden.

Nach der Anreicherung können die Geschäftsprozessmodelle und deren Projekt auf die Axon.ivy Engine überführt werden. Auf Basis der Axon.ivy Engine können die (teil-)automatisierten Geschäftsprozesse produktiv ausgeführt werden. Bei Bedarf können mehrere Versionen eines Geschäftsprozesses parallel betrieben werden. Die Abbildung 70 skizziert die Architektur der Axon.ivy Engine. Durch einen HTTP/HTTPS-Aufruf können Anwender (Clients) auf die zur Verfügung stehenden Geschäftsprozessapplikationen zugreifen. Dieser Zugriff kann entweder direkt auf die Axon.ivy Engine oder indirekt über einen zusätzlichen Webserver (welcher z.B. zur Lastverteilung eingesetzt werden kann) erfolgen. Die Axon.ivy Engine wird durch drei weitere Komponenten ergänzt. Zum einen können optional Drittsysteme angebunden werden, um geschäftsprozessrelevante Daten zu lesen, zu schreiben oder zu aktualisieren. Zum anderen müssen eine Systemdatenbank (z.B. zur Speicherung der Instanzen, Aufgaben,...) und ein Speicherort für Axon.ivy-Projektdateien an die Engine angeschlossen werden.

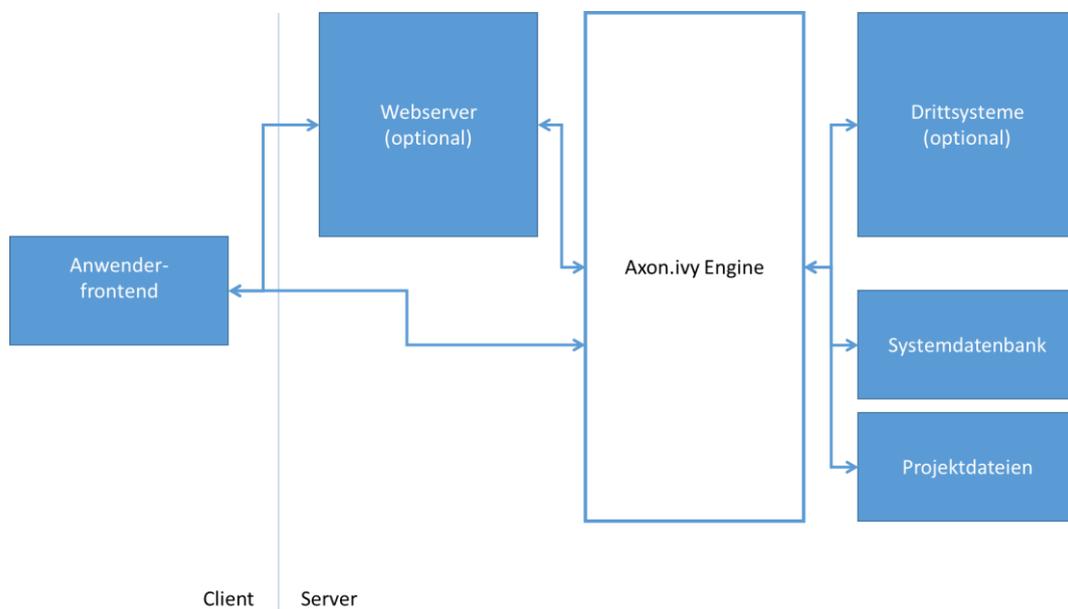


Abbildung 70: Axon.ivy Engine Architektur Übersicht nach [AX15b]

Neben einer Administrationsoberfläche zur Verwaltung der Axon.ivy Engine Projekte, dient das Axon.ivy Workflow Portal zur Einbindung der Anwender. Die Abbildung 71 stellt eine mögliche Startseite des Portals dar.

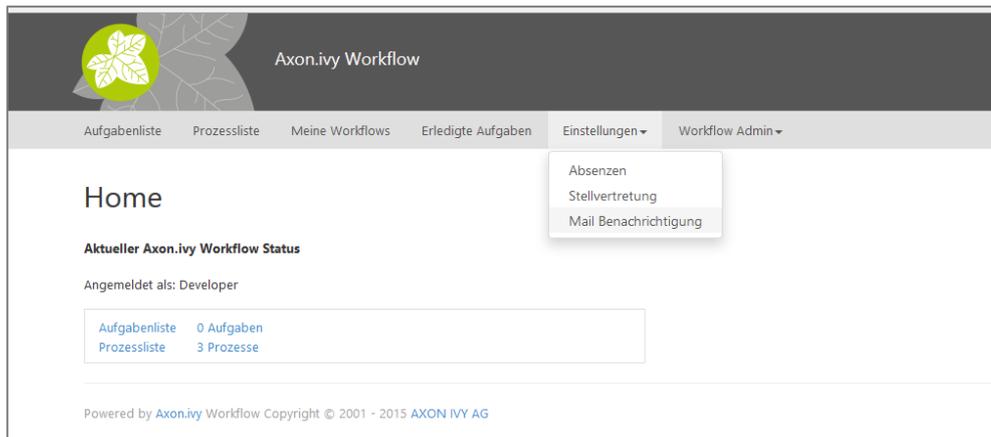


Abbildung 71: Screenshot Axon.ivy Workflow Portal

Damit Anwender die individuell erstellten (teil-)automatisierten Geschäftsprozesse aufrufen können, wird eine Schnittstelle benötigt. Das Axon.ivy Workflow Portal ermöglicht hierbei das Verwalten bzw. Bearbeiten von zugeteilten Aufgaben. Zusätzlich können Endanwender zugeteilte Aufgaben delegieren und kommentieren. Des Weiteren können mit Hilfe der Funktion „Workflow-Statistik“ Status und Durchlaufzeiten der ausgeführten Instanzen überwacht werden. Zur Messung von weiteren Kennzahlen kann der Axon.ivy Monitor angewendet werden. Mit Hilfe des Axon.ivy Monitor (vgl.

Abbildung 72) können sowohl die durch Axon.ivy als auch die von anderen Drittsystemen gespeicherten Daten analysiert und visualisiert werden. Werden ex ante definierte Grenzwerte (z.B. Überschreitung von Maximaldurchlaufzeiten) gemessen, können Verantwortliche Rollen/Personen unmittelbar alarmiert werden. Die Daten können hierarchisch gegliedert werden, sodass eine Drilldown-Navigation ermöglicht wird.



Abbildung 72: Screenshot Axon.ivy Monitor

4.4.2. BPM Touch

BPM Touch ist ein Softwarewerkzeug zur Modellierung von Geschäftsprozessen auf mobilen Endgeräten. Als App kann BPM Touch auf einem Android²⁰² Betriebssystem installiert und betrieben werden. BPM Touch unterstützt die Modellierung mit BPMN^{Easy}, sodass neben der grafischen Modellierung von Geschäftsprozessmodellen, auch die direkte Erfassung von multimedialen Inhalten möglich ist. Verschiedene Touch-Gesten können zur Bedienung der App angewendet werden. Zum Beispiel können mit einem Touch (z.B. Berührung einer Aktivität mit dem Finger) existierende Modellierungselemente verschoben oder editiert werden. Die Abbildung 73 stellt die Modellierungsperspektive von BPM Touch dar.

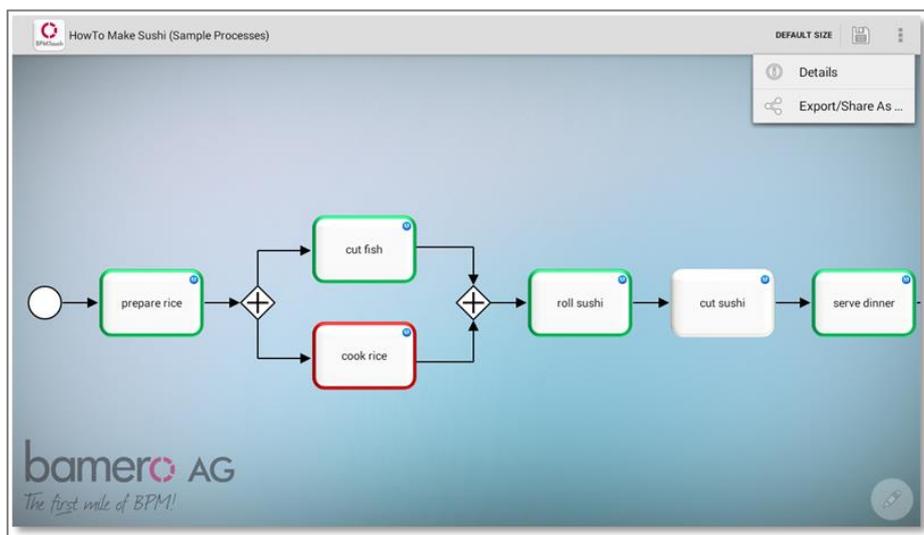


Abbildung 73: Modellierungsperspektive in BPM Touch

BPM Touch unterstützt den Modellierer durch zusätzliche Bedienhilfen. Unter anderem ist ein BPMN^{Easy} Event automatisch ein End-Event, sofern kein weiteres Element folgt. Des Weiteren wird jede BPMN^{Easy} Aktivität gekennzeichnet, welche mit Mediafiles verbunden ist. Insbesondere die Möglichkeit der intuitiven Bedienung wird hierdurch signifikant gesteigert. Die modellierten Geschäftsprozesse werden standardmäßig in einem Default-Repository gespeichert. Durch das Anlegen von eigenen Projekt-Repositories können Geschäftsprozessmodelle gruppiert und verschoben werden.

Zur Erfassung der Mediafiles verwendet BPM Touch die Standardfunktionalität des mobilen Endgeräts.

²⁰² <https://www.android.com/>

Soll beispielsweise ein Bild an eine BPMN^{Easy} Aktivität geknüpft werden, kann hierzu der Speicher des mobilen Endgeräts durchsucht werden. Optional kann die Kamera des mobilen Endgeräts gestartet werden (sofern eine solche existiert). Die Abbildung 74 zeigt den BPM Touch Screen zur Ansicht der Details einer Aktivität.

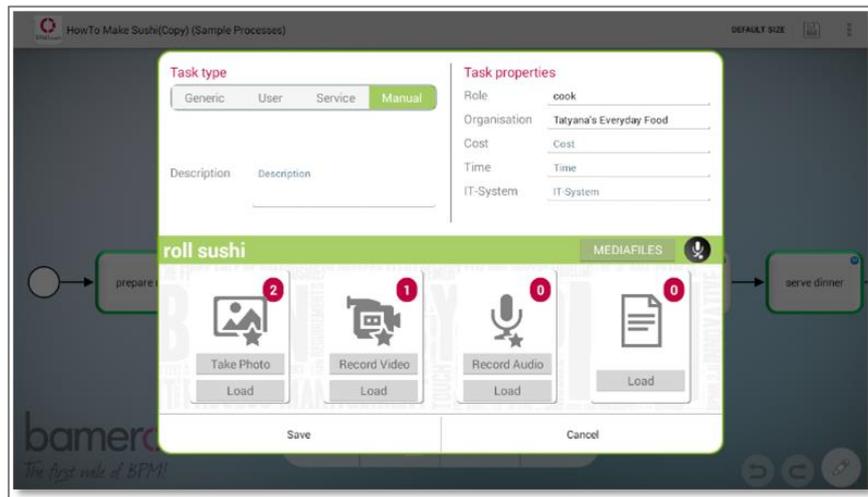


Abbildung 74: Übersicht über Detailinformationen einer Aktivität

Über die Detailansicht erfolgt die Pflege von Rollen, Organisation, Zeit, Kosten, IT-Systeme und Mediafiles (vgl. Abbildung 74). Des Weiteren wird die Anzahl der Mediafiles sortiert nach Typ (Bild, Video, Audio, Dokument) angezeigt. Zur besseren Übersicht über die Mediafiles kann in den Mediafile-Screen gewechselt werden.

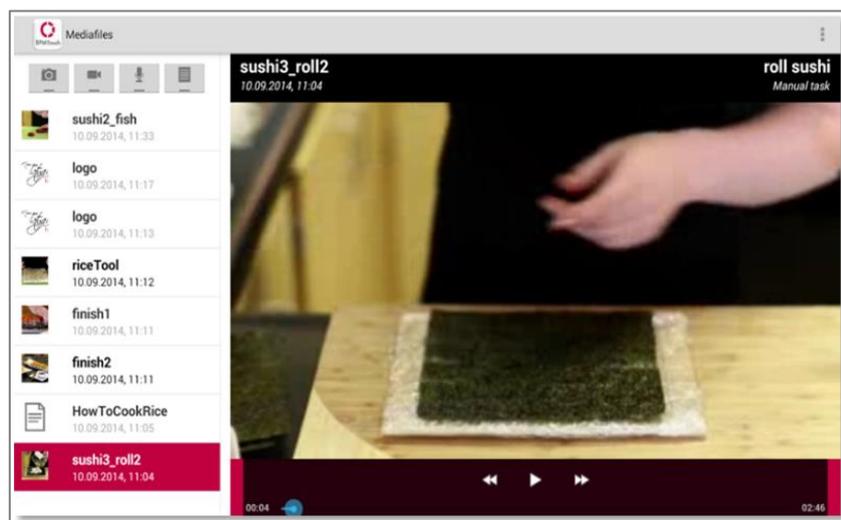


Abbildung 75: Verwaltung der Mediafiles in BPM Touch

Mit Hilfe des Mediafile-Screens haben Anwender des BPM Touchs einen direkten Zugriff auf die annotierten Multimediainhalte, wodurch eine weitere Perspektive auf die Geschäftsprozessmodelle zur Verfügung steht.

Zum Beispiel können Ausführende eines Geschäftsprozesses mit Hilfe dieser Perspektive „bildlich“ durch die zu bearbeiteten Aktivitäten geführt werden.

Zur Einbindung von anderen Projektbeteiligten und zur Distribution der Geschäftsprozessmodelle verfügt BPM Touch über unterschiedliche Schnittstellen. Beispielsweise können Geschäftsprozessmodelle (inkl. Mediafiles) direkt per E-Mail versendet werden. Des Weiteren können die Geschäftsprozessmodelle als BPMN 2.0-XML-Datei abgespeichert werden (vgl. Abschnitt 4.3), sodass diese in einem weiteren Softwarewerkzeug (z.B. Axon.ivy BPM Suite) verfeinert werden können.

5. Ansätze zur Unterstützung von BPM(N)^{Easy}

In diesem Abschnitt werden zwei Ansätze zur Unterstützung der BPM(N)^{Easy} Methode vorgestellt. Zum einen wird die Entwicklung einer prototypischen Software beschrieben. Die Software kann zur Unterstützung der BPM(N)^{Easy} Vorgehensweise angewendet werden und dient zur Erfassung von Anwenderfeedback während der Ausführung von Geschäftsprozessen. Zum anderen wird in Abschnitt 5.2 die Konzeption eines BPMN^{Easy} Pattern beschrieben. Das Pattern stellt die dialogische Logik-basierte Interaktion zur Unterstützung einer manuellen Entscheidungsfindung im Rahmen einer Geschäftsprozessaktivität dar.

5.1. Entwicklung eines Prototyps zur Unterstützung des agilen GPMs (Z3)

Mobile Applikationen (Apps) sind heutzutage allgegenwärtig (vgl. [Te13][In14b]) und können zur ständigen Erreichbarkeit, ortsunabhängiger Interaktion und Kommunikation dienen. Nach Gebhart et al. [GMW15] können „(...) insbesondere bei flexiblen und zeitkritischen Geschäftsprozessen [...] mobile Endgeräte eine signifikante Effizienzsteigerung hervorrufen. Beispielsweise können Außendienstmitarbeiter direkt über neue Aktivitäten mobil informiert werden, ohne dass es zu zeitlichen Verzögerungen und damit zu Kundenunzufriedenheit führt.“.

Häufig werden Geschäftsprozessmanagementsysteme (vgl. Abschnitt 2.2.3) insbesondere zur Modellierung und Automatisierung von Geschäftsprozessen eingesetzt. Zum Beispiel können mit BPM Touch [ba15] (vgl. Abschnitt 4.4.2) Geschäftsprozesse mobil erfasst werden. Der im Rahmen dieser Arbeit entwickelte Software-Prototyp dient zur Erfassung von Anwenderfeedback während der Ausführung von Aktivitäten eines Geschäftsprozesses. Nach Ambler [Am02, S.22f] ist die Erfassung von Feedback die einzige Möglichkeit, die Abstraktion eines (Geschäftsprozess-)Modells mit den realen Anforderungen zu vergleichen²⁰³. Die klassische Überwachung von Geschäftsprozessen (z.B. Messung von Zeiten auf Basis von Kennzahlen) wird durch die Erfassung von Feedback erweitert. Ex ante definierte Qualitätskriterien dienen zur Einordnung des Feedbacks (vgl. Abschnitte 3.3 und 4.2.4). Beispielsweise kann erfasst werden, wie

²⁰³ Vgl. engl. Übersetzung [Am02, S.22]: „The only way that you can determine whether your work is correct is to obtain feedback, and that includes feedback regarding your models.“.

zufrieden der Anwender während der Ausführung einer Geschäftsprozessaktivität ist (z.B. wie er sich fühlt, ob dieser ungestört arbeiten kann,...). Um die agile (z.B. ortsunabhängig) Erfassung zu gewährleisten, wird der Software-Prototyp insbesondere für die Anwendung auf mobilen Endgeräten optimiert.

Die Software (im Folgenden u.a. als FeedbackApp bezeichnet) wird als Prototyp entwickelt. Prototypen²⁰⁴ dienen zur Aufdeckung von Vor- und Nachteilen innerhalb früher Entwicklungsstadien. Hierbei decken Prototypen nicht alle Anwendungsfälle ab, sondern stellen die prinzipielle Lösung von Anforderungen dar (vgl. [Sc09b][eB14]). Die Ziele des in dieser Arbeit implementierten Prototypen sind:

- Ermöglichung der Speicherung von Anwenderfeedback auf Basis von ex ante definierten Qualitätskriterien und vorformulierten Fragen,
- Bereitstellung eines User Interfaces, welches insbesondere für mobile Endgeräte optimiert ist und als Komponente in bestehende Applikationen eingebunden werden kann,
- Optimierung der Feedbackerfassungszeitdauer²⁰⁵,
- Implementierung einer Benutzeroberfläche zur Auswertung des erfassten Feedbacks.

Insgesamt dient die Implementierung zur Erfassung von Feedback, welches manuell ausgewertet und als Grundlage für einen Dialog - auf Basis des dialogischen Logik -

²⁰⁴ Scheer [Sc09b, S.88] nutzt den Begriff des „Forschungsprototypen“. Vgl. : „(...) Die Prototypen konnten nicht in der Praxis eingesetzt werden, da Forschungsprototypen keine Produktanforderungen erfüllen. Forschungsprototypen sind: nicht stabil, decken nicht alle möglichen Anwendungsfälle ab, sondern zeigen lediglich prinzipielle Lösungen, sie besitzen keine komfortable Benutzerführung, sie sind nicht umfangreich dokumentiert, es besteht kein Marketing- und Vertriebskonzept, es ist keine Weiterentwicklungsstrategie definiert, es steht keine feste Entwicklungsgruppe auf Dauer zur Verfügung, es werden keine kommerziell üblichen technischen Plattformen bevorzugt, usw.(...)“.

²⁰⁵ Ambler beschreibt den zeitlichen Aufwand der Erfassung als kritischen Parameter. Vgl. [Am02, S.23]: “ (...) Why is timescale important? Because the more immediate the feedback is, the less likely your models are to deviate from what you actually need. Although all forms of feedback have their place, given the opportunity you should prefer the feedback provided through teamwork because it is immediate. Having said that, there is something to be said about proving your models with code because everything looks good on paper until you actually try it out.”

Patterns (vgl. Abschnitt 5.2) – verwendet werden kann. Des Weiteren dient der Prototyp als Grundlage für fortführende Implementierungen²⁰⁶ in Bezug auf das automatisierte Durchlaufen des dialogischen Logik-Patterns.

5.1.1. Technischer Aufbau des Prototypen

Dieser Abschnitt beschreibt die verwendeten Technologien und die Softwarearchitektur des entwickelten Software-Prototypen.

- Technologien

Zur Entwicklung des Software-Prototypen wurde das Softwarewerkzeug Axon.ivy Designer eingesetzt (vgl. Abschnitt 4.4.1). Auf Basis der Entwicklungsplattform Eclipse stellt der Axon.ivy Designer unterschiedliche Frameworks, Programmiersprachen und Plugins zur Verfügung. Das technische Prozessmodell wurde mit Hilfe von BPMN 2.0 und den Axon.ivy spezifischen BPMN 2.0 Erweiterungen modelliert. Das Modell beschreibt unter anderem, welches Ereignis die Erfassung des Feedbacks auslöst. Zudem werden Aktivitäten, wie zum Beispiel der „Database-Step“ (Axon.ivy BPMN 2.0 Erweiterung), zur direkten Anbindung der für den Software-Prototyp notwendigen Datenbanken genutzt.

Zusätzlich zu BPMN 2.0 werden die Programmiersprache Java und das Entwicklungsframework PrimeFaces eingesetzt. Zur Verarbeitung von Informationen im Hintergrund dienen in Java implementierte Methoden. Diese Methoden können aus dem Prozessmodell direkt aufgerufen werden. Beispielsweise ist die Methode „selectQualityCriterion“, welche für die Auswahl des Feedback-relevanten Qualitätskriteriums genutzt wird, in Java implementiert. Die mobilen Benutzeroberflächen der FeedbackApp basieren auf PrimeFaces. PrimeFaces²⁰⁷ ist eine Open Source Bibliothek, welche auf JavaServer Faces aufsetzt und in Axon.ivy integriert ist. Die mit PrimeFaces entwickelten Benutzeroberflächen können plattformunabhängig aufgerufen und angezeigt werden. Zur Speicherung der Qualitätskriterien und des erfassten Feedbacks wird das Datenbankmanagementsystem MySQL²⁰⁸ eingesetzt. Die

²⁰⁶ Die fortführenden Implementierungen sind nicht Teil der vorliegenden Arbeit.

²⁰⁷ <http://primefaces.org/>

²⁰⁸ <http://www.mysql.com/>

„MySQL Community Edition“ steht hierbei frei zur Verfügung und kann zur Verwaltung relationaler Datenbanken eingesetzt werden.

- Architektur

Zentrale Bestandteile der FeedbackApp-Architektur sind das Model-View-Controller (MVC)- und das Observer-Pattern (vgl. [Ga95]). Das MVC-Pattern beschreibt die Trennung von Daten (Model), Präsentation (View) und Steuerung (Controller). Das Model besitzt ausschließlich Daten (ggfs. auch Geschäftslogik), welche durch die View(s) dargestellt werden. Neben der Darstellung der Daten übernimmt die View die Interaktion zum Anwender (z.B. Registrierung eines Mausklickevents auf einen Button der View). Model und View stehen nicht direkt in Verbindung. Der Controller übernimmt die Steuerung zwischen Model und View(s). Zum Beispiel werden die durch die View registrierten Ereignisse durch den Controller verarbeitet. Müssen Daten geändert werden, wird das Model danach informiert.

Basierend auf dem MVC-Pattern besteht die FeedbackApp aus einer drei-schichtigen Client-Server-Architektur²⁰⁹. Durch diese Architektur wird insbesondere die flexible Skalierbarkeit gewährleistet. Beispielsweise kann die Anwendungsschicht bei hoher Auslastung unabhängig von den anderen Schichten repliziert werden (vgl. [HM05, S.8f] [GD13, S.300]). Die Abbildung 76 stellt die Drei-Schichten-Architektur der FeedbackApp dar. Die oberste Schicht dient zur Integration des Anwenders. Die FeedbackApp kann hierbei über Desktop- oder Mobilgerät bedient werden. Des Weiteren können die Qualitätskriterien über eine Desktop-View gepflegt und das Feedback ausgewertet werden. Die mittlere Ebene beinhaltet die Steuerungslogik und teilt sich wiederum in drei Bereiche auf. Zum einen beinhaltet diese die für die Ausführung relevanten Geschäftsprozessmodelle, zum anderen ist die Logik zur Erfassung des Anwenderfeedbacks in der mittleren Schicht implementiert. Zusätzlich wird die Verwaltung und Auswertung der Qualitätskriterien gesteuert. Die unterste Ebene dient zur Speicherung der Daten (Datenschicht bzw. Model).

²⁰⁹ In der Literatur wird häufig auch die engl. Bezeichnung „Three Tier Architecture“ verwendet. Wobei die Trennung von Präsentationsschicht, Anwendungsschicht und Datenschicht charakteristisch ist.

Die Speicherung der Geschäftsprozessmodelle wird durch Axon.ivy im Standard gesteuert. Zusätzlich werden die Qualitätskriterien und das erfasste Feedback in einer separaten Datenbank gespeichert.

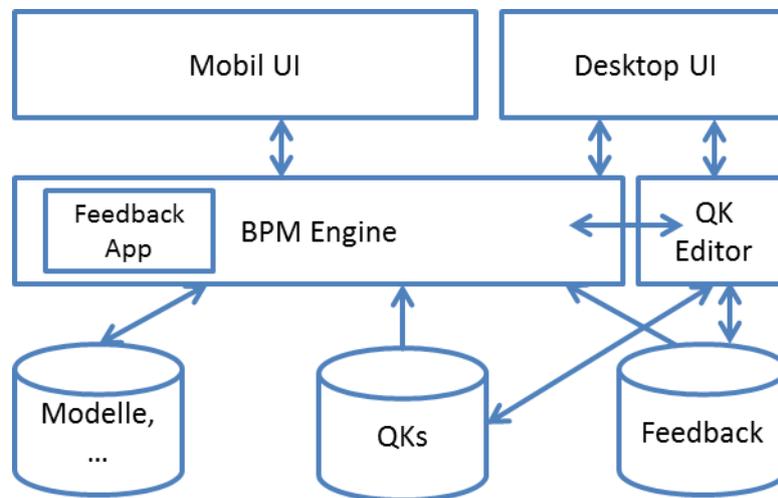


Abbildung 76: Skizze der Drei-Schichten-Architektur

Für die Verwaltung der Qualitätskriterien (QK) wird im Prototyp die MySQL Workbench²¹⁰ genutzt. Die MySQL Workbench ist ein freiverfügbares grafisches Softwarewerkzeug zur Verwaltung von MySQL Datenbanken und der darin gespeicherten Daten. Die Änderung von Qualitätskriterien kann direkten Einfluss auf die Steuerung der FeedbackApp besitzen. Obwohl die BPM Engine und der QK Editor zwei getrennte Anwendungen sind, ist in Abbildung 76 eine direkte Verbindung zwischen den zwei Anwendungen dargestellt. Zur Anzeige der Qualitätskriterien und zur Auswertung des Feedbacks (z.B. mit Hilfe von grafischen Diagrammen) kann die BPM Engine auf die QK- und Feedback-Datenschicht lesend zugreifen.

Das zweite zentrale Pattern, das Observer-Pattern, dient zur Verbindung der Axon.ivy automatisierten Geschäftsprozessapplikationen mit der FeedbackApp. Das Observer-Pattern beschreibt die Beziehung zwischen einem „beobachtbaren“ Objekt und einem oder mehreren Beobachtern. Das „beobachtbare“ Objekt (Subjekt) besitzt eine Schnittstelle über diese sich Beobachter registrieren können. Sobald Änderungen (z.B. Änderung eines Systemzustands) im Subjekt auftreten, können alle registrierten Beobachter informiert werden. Im Kontext des Prototyps nimmt die FeedbackApp die Rolle des Beobachters ein. Soll das Feedback während der Axon.ivy-unterstützten

²¹⁰ <https://www.mysql.de/products/workbench/>

Ausführung einer Aktivität erfasst werden, „informiert“ die Geschäftsprozessapplikation die FeedbackApp.

Die FeedbackApp startet und blendet den Feedbackdialog ein. Die lose Kopplung zwischen Subjekt und Beobachter ermöglicht die einfache Einbindung der FeedbackApp in bestehende Geschäftsprozessapplikationen, ohne dass diese z.B. Kenntnis über die Logik besitzen muss.

- Implementierung

Bei der Implementierung einer mobilen App kann zwischen nativen Apps, Web Apps und hybriden Apps unterschieden werden (vgl. [Pe13] [NB13]). Native Apps werden direkt auf dem Betriebssystem des Endgeräts installiert. Beispielsweise muss BPM Touch zuerst auf dem Android Betriebssystem installiert werden, bevor es ausgeführt werden kann. Vorteil von nativen Apps ist, dass diese im Allgemeinen auf Funktionen, welche das Betriebssystem über eine Schnittstelle zur Verfügung stellt, zugreifen können. Web Apps hingegen sind Webseiten, welche für mobile Endgeräte optimiert werden (z.B. hinsichtlich der Bildschirmauflösung). Anstatt diese zu installieren, können diese über einen Browser aufgerufen werden. Hybride Apps stellen eine Kombination zwischen nativen Apps und Web Apps dar. Durch den nativen Anteil der App können die Betriebssystemfunktionen genutzt werden. Gleichzeitig ist es möglich, durch den Web Anteil kontinuierlich neue Inhalte „von außen“ dem Anwender anzuzeigen. Basierend auf der Axon.ivy Architektur wird die FeedbackApp als Web App implementiert, um größtmögliche Plattformunabhängigkeit zu gewährleisten. Im Folgenden werden einzelne Teile der Implementierung im Detail beschrieben.

Komponentenbasierte Implementierung

Die FeedbackApp kann als Komponente in einen existierenden User Dialog eingebunden werden. Wird beispielsweise ein User Dialog zur Erfassung von Urlaubsanträgen eingesetzt, könnte durch die Einbindung der FeedbackApp, Feedback der Anwender bzgl. der Beantragung von Urlaub erfasst werden. Die lose Kopplung zwischen FeedbackApp und User Dialog ist in Abbildung 77 dargestellt.

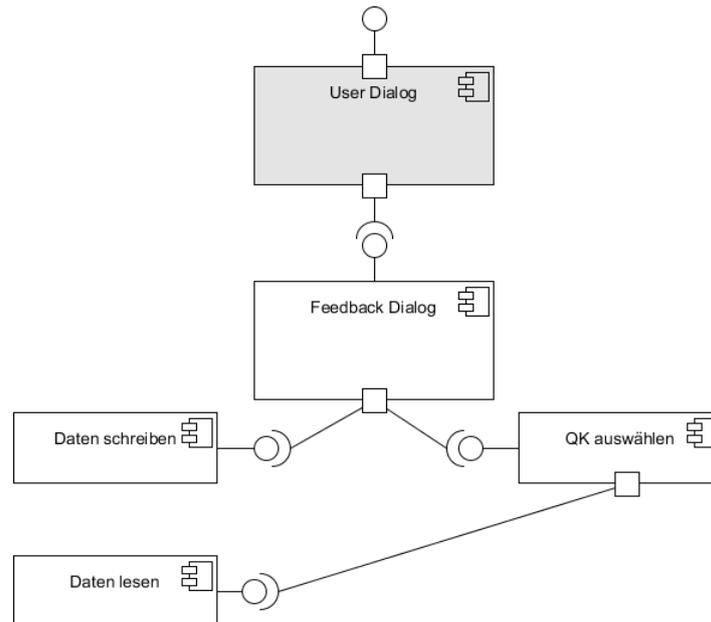


Abbildung 77: UML Komponentendiagramm FeedbackApp

Die FeedbackApp besteht aus einem eigenen User Dialog, welcher über PrimeFaces komponentenbasiert eingebunden werden kann. Des Weiteren enthält die FeedbackApp Komponenten zum Lesen und Schreiben von Daten und zur Auswahl der anzuzeigenden Qualitätskriterien. Die Tabelle 37 enthält einen Ausschnitt der FeedbackApp PrimeFaces-Komponente:

Tabelle 37: Ausschnitt FeedbackApp UI Component

```

<cc:interface componentType="IvyComponent">
  <cc:attribute name="qualitygroup"/>
</cc:interface>

<cc:implementation>
  <p:outputLabel value="{data.qualitycriteria}"></p:outputLabel>
  <p></p>
  <p:inputTextarea style="width: 100%"value="{data.feedback}">
  </p:inputTextarea>
  <p:commandButton value="Feedback senden"
  ActionListener="{logic.save}">
  </p:commandButton>
</cc:implementation>
  
```

Innerhalb des `<cc:interface componentType="IvyComponent">`-Tags ist die Schnittstelle der FeedbackApp definiert. Die FeedbackApp enthält ausschließlich einen Schnittstellenparameter zur Beschreibung der Quality Group. Der Modellierer kann somit entscheiden, aus welcher Gruppe von Qualitätskriterien ein Qualitätskriterium automatisiert ausgewählt werden soll. Innerhalb des `<cc:implementation>`-Tags wird

der Feedback Dialog definiert. Der Dialog aus Tabelle 37 besteht aus dem angezeigten Qualitätskriterium, einem Eingabefeld zur Eingabe des Feedbacks und einem Button zum Speichern des Feedbacks. Durch das Klicken des Buttons wird durch den Action Listener `actionListener="#{logic.save}"` der Controller der FeedbackApp über die Änderung des Zustands benachrichtigt. Als Folge wird das Feedback in einer Datenbank gespeichert.

Datenbankdesign

Die Axon.ivy Standarddatenbank verwaltet alle Daten, welche zur Ausführung der Geschäftsprozessapplikationen relevant sind. Zum Beispiel wird der Status der einzelnen Aktivitäten und die damit verbundenen Systeminformationen gespeichert (z.B. Startzeit der Bearbeitung, Liegezeit,...). Müssen weitere individuelle Daten gespeichert werden, muss dies individuell organisiert werden. Die Abbildung 78 stellt die Entitäten der FeedbackApp-Datenbank und deren Beziehungen dar. Als Attribute der Entität „Qualitätskriterium“ (qualitycriteria) sind Bezeichnung des Qualitätskriteriums und die konkrete Fragestellung (question) verbunden. Des Weiteren besitzt jedes Qualitätskriterium eine definierte Gewichtung und eine Zuordnung zu einer Qualitätskriterien-Gruppe.

Die Entität „Qualitätskriterien-Gruppe“ (qualitycriteria_group) enthält ausschließlich eine Bezeichnung und ein Kommentarfeld (z.B. zur Speicherung eines Beispielfeedbacks). Das, durch die Anwender erfasste, Feedback wird mit Hilfe der Entität „Feedback“ (feedback) abgespeichert. Als Zusatzinformation wird die Axon.ivy Instanz-ID, das verbundene Qualitätskriterium und die Erfassungszeit abgespeichert.

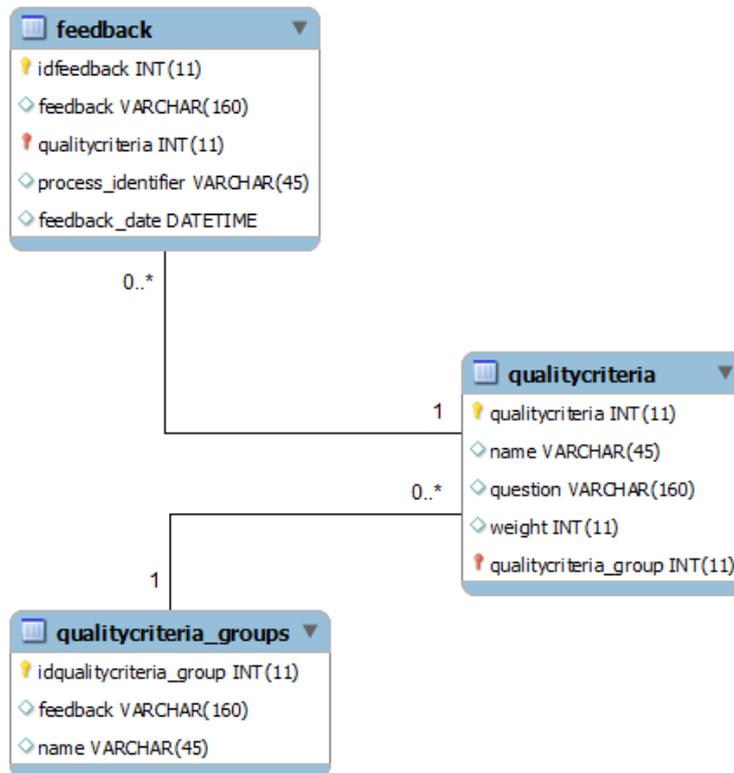


Abbildung 78: Entity-Relationship-Diagramm FeedbackApp

Wie in Abbildung 78 skizziert, sind sowohl die Entitäten „Qualitätskriterium“ und „Qualitätskriterium Gruppe“, als auch die Entitäten „Feedback“ und „Qualitätskriterium“ durch eine „0..* - 1“-Verbindung verbunden. Beispielsweise können somit Gruppen für die Einordnung von Qualitätskriterien angelegt werden, ohne dass diese direkt mit Qualitätskriterien befüllt werden müssen.

Der Aufruf einer Datenbank wird in Axon.ivy durch das grafische Modellierungselement „Database Step“ realisiert. Der Database Step, wie auch alle anderen Ereignisse und Aktivitäten des FeedbackApp-Prozesses, werden grafisch modelliert. Durch spezifische Konfigurationseditoren²¹¹ lassen sich diese Elemente nachfolgend konfigurieren. Die Abbildung 79 visualisiert einen Modellausschnitt zum Aufruf des FeedbackApp Dialogs. Das Startereignis löst den Start des Prozesses aus und übergibt die Parameter (festgelegt durch die Schnittstellendefinition) an die nachfolgenden Elemente. Im ersten Schritt werden durch den Database Step die möglichen Qualitätskriterien aus der verbundenen Datenbank gelesen. Das Ergebnis wird an eine zweite Aktivität (Script Step) übermittelt.

²¹¹ Für eine ausführliche Beschreibung der Axon.ivy Konfiguratoren und Modellierungselemente sei auf <http://developer.axonivy.com/doc/latest/DesignerGuideHtml/> verwiesen.

Die Script Step-Aktivität ruft den in Java implementierten Algorithmus zur Auswahl des Qualitätskriteriums auf Basis der Gruppe und der Gewichtung auf.

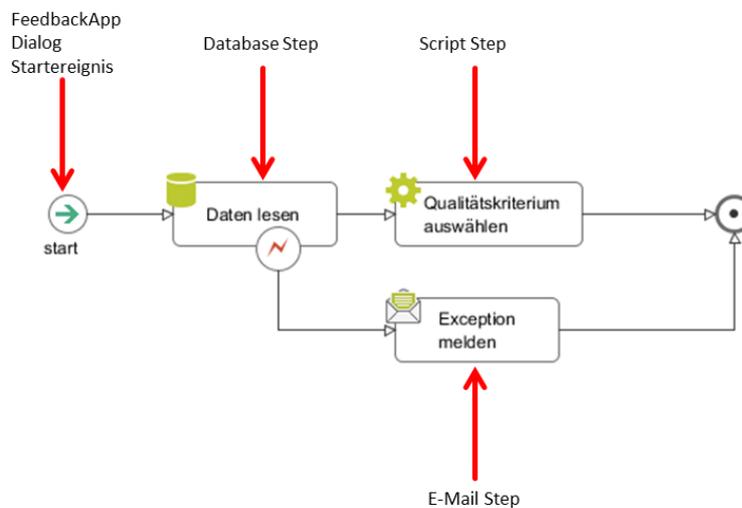


Abbildung 79: Ausschnitt FeedbackApp Prozesslogik

Das Endereignis dient als Rücksprung in die übergeordnete Komponente und zur Anzeige des Feedback Dialogs auf dem mobilen Endgerät. Im Fehlerfall (z.B. Datenbank ist nicht erreichbar) wird eine Fehlermeldung im Dialog angezeigt. Zusätzlich wird eine Ausnahme (Exception) ausgelöst. Die Ausnahme triggert den Versand einer Informations-E-Mail an z.B. den Geschäftsprozessverantwortlichen.

Programmlogik

Zur Verarbeitung der System- und Anwenderdaten besitzt die FeedbackApp unterschiedliche Algorithmen. Beispielsweise werden die ex ante definierten Qualitätskriterien nach ihrer Gewichtung automatisch ausgewählt. Die Tabelle 38 stellt einen Ausschnitt der Methode “getQualityCriterium” dar:

Tabelle 38: Ausschnitt Programmcode zur Gewichtung der Qualitätskriterien

```

//...
Random random = new Random();
ArrayList<Number> qc_weightlist = new ArrayList<Number>();
for (int number : weights) {
    for(int i=0; i< number; i++) {
        qc_weightlist.add(number);
    }
}
random.nextInt(qc_weightlist.size());
//...
  
```

Zur Auswahl des, für die Instanz gültigen, Qualitätskriteriums wird ein einfacher Algorithmus angewendet. Zuerst wird eine Liste zur Darstellung der Gewichtung erstellt, worauf danach eine Stelle der Liste ausgewählt wird, Beispiel:

- (1) in der Datenbank sind drei Qualitätskriterien in einer Gruppe, gewichtet mit den ganzen Zahlen 1, 2, 3, wobei die höhere Zahl eine höhere Gewichtung symbolisiert.
QK1-> 1, QK2->2, QK3->3
- (2) Eine temporäre Liste wird erstellt.
Liste: 122333
- (3) Es wird eine zufällige Zahl zwischen 0 und 5 ermittelt (die Liste hat 6 Zeichen).
Zufallszahl: 5
- (4) Das fünfte Zeichen der Liste ist eine 3. 3 ist die Gewichtung des Qualitätskriteriums QK3. Das QK3 wird somit ausgewählt und angezeigt.

Eine durch den Axon.ivy Designer automatisierte, (technische) Dokumentation des Prototyps ist im Anhang dieser Arbeit zu finden.

5.1.2. Funktionen des Prototypen

In diesem Abschnitt wird zum einen die Einbindung der FeedbackApp in eine bestehende Axon.ivy Geschäftsprozessapplikation beschrieben, zum anderen werden die grundlegenden Funktionen des Prototyps aus Sicht der Anwender dargestellt.

Technische Vorbereitungen

Die Tabelle 39 zeigt einen Ausschnitt eines existierenden Axon.ivy User Dialogs, in welchen die FeedbackApp eingebunden wird. Der User Dialog dient zur Erfassung von Daten zur Angebotserstellung (z.B. Angabe der Kundendaten, Lieferdatum,...). Wie auch die FeedbackApp ist der User Dialog auf Basis eines mobilfähigen Layouts erstellt. Durch den `<ic:userFeedback.feedbackComponent qualitygroup="2" width="100%"/>`-Tag wird die FeedbackApp, welche zuvor im Namensraum `userFeedback` abgelegt wurde, eingebunden. Die Komponente kann hierbei an einer beliebigen Stelle hinzugefügt werden, abhängig beispielsweise von den Benutzeroberflächen-Designanforderungen (vgl. Tabelle 39, gelbe Markierung). Über das Attribut `qualitygroup="2"` wird die Qualitätskriterien-Gruppe angegeben, die für den

aufgerufenen User Dialog relevant ist. Im Beispiel wird statisch immer die Gruppe mit der ID 2 aufgerufen, je nach Anforderung könnte dieser Parameter auch dynamisch bestimmt werden. Das Attribut `width="100%"` bestimmt die Größe des FeedbackApp Dialogs. Durch die Angabe von 100% steht dem Anwender ein Feedback-Eingabefeld zur Verfügung, welches sich über die komplette Breite des Bildschirms ausdehnt.

Tabelle 39: Einbindung der FeedbackApp Komponente

```

<h:body>
<ui:composition template="/layouts/mobile.xhtml">
<ui:define name="title">Mobile Client</ui:define>
<ui:define name="content">
<h3>
Angebotsprozess
</h3>
<div class="float-right-small-device"
style="margin-bottom: 10px; margin-top: -10px;">
<ic:userFeedback.feedbackComponent qualitygroup="2"
width="100%"/>
</div>
<h:form id="form">
<h:panelGrid columns="2">
<f:facet name="header">
<p:messages></p:messages>
</f:facet>
<p:outputLabel for="dataDatum" value="Datum"/>
<p:calendar id="dataDatum"
value="#{data.data.Datum}" pattern="dd.MM.yyyy" size="12">
<f:convertDateTime pattern="dd.MM.yyyy"/>
</p:calendar>
<p:outputLabel for="dataKunde" value="Kunde"/>
<p:inputText
id="dataKunde" value="#{data.data.Kunde}"></p:inputText>
<p:outputLabel for="dataTitel" value="Titel"/>
<p:inputText
id="dataTitel" value="#{data.data.Titel}"></p:inputText>
<p:outputLabel for="dataBeschreibung" value="Beschreibung"/>
<p:inputText
id="dataBeschreibung" value="#{data.data.Beschreibung}">
</p:inputText>
<p:outputLabel for="dataArtikel" value="Anzahl Lizenzen"/>
<p:inputText
id="dataArtikel" value="#{data.data.Artikel}"></p:inputText>
<br/>
<p:commandButton value="Angebot abschicken"
update="form" icon="ui-icon-check"
actionListener="#{logic.erkassen}"/>
</h:panelGrid>
</h:form>
</ui:define>
</ui:composition>
</h:body>

```

Neben der Einbindung der Komponente mit Hilfe von PrimeFaces muss sichergestellt werden, dass die FeedbackApp auf die zur Anwendung notwendige Datenbank zugreifen kann. Über einen Datenbankkonfigurator in Axon.ivy Designer lässt sich die Datenbankverbindung zentral pflegen. Innerhalb des Database Step (z.B. zum Lesen der Qualitätskriterien aus der Datenbank) werden die konfigurierten Verbindungsparameter dann abgerufen. Die Abbildung 80 stellt einen Ausschnitt aus dem Axon.ivy Designer zur Konfiguration der FeedbackApp Datenbank dar.

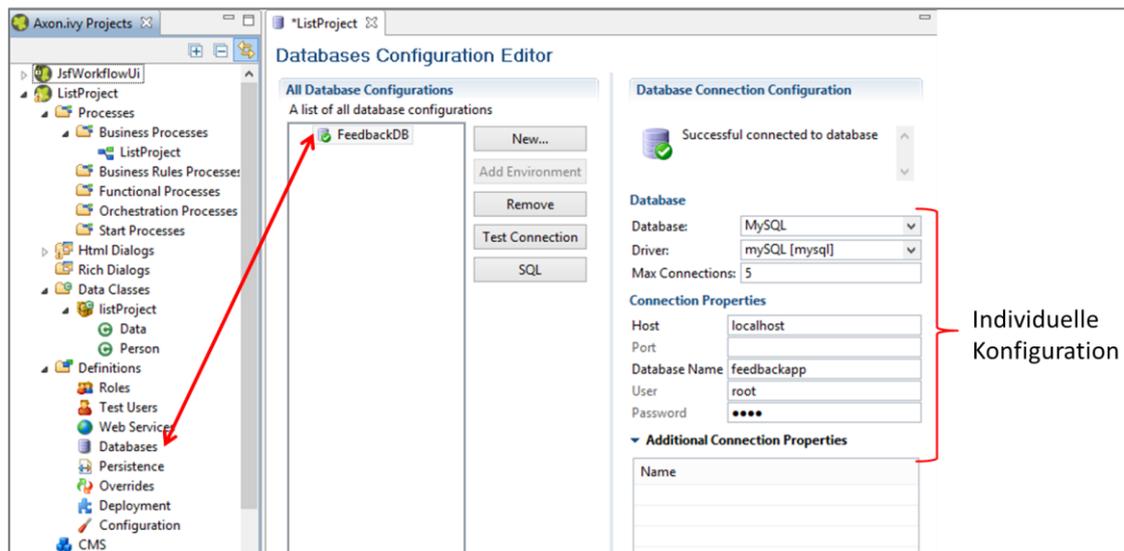


Abbildung 80: Konfiguration der FeedbackApp Datenbankanbindung

Beschreibung der Funktionen aus Sicht des Anwenders

Nach Abschluss der Komponenten- und Datenbankkonfiguration kann die Axon.ivy Engine gestartet und die Geschäftsprozessapplikation ausgeführt werden. Die Abbildung 81 visualisiert das Ergebnis. Bei Aufruf des User Dialogs „Angebot erfassen“ werden dem Anwender drei Bereiche auf dem mobilen Endgerät angezeigt: Header, FeedbackApp Komponente und User Dialog.

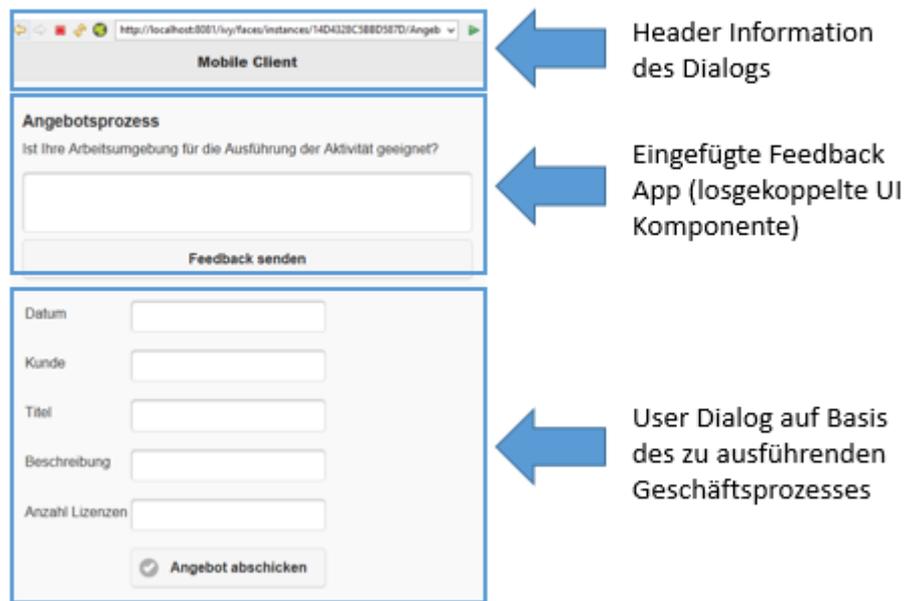


Abbildung 81: Beispiel - User Dialog mit FeedbackApp Komponente

Im Beispiel (vgl. Abbildung 81) dient der Header-Bereich ausschließlich zur Anzeige eines Dialogtitels. In komplexeren Szenarien kann der Header auch zur Anzeige einer Geschäftsprozessübersicht genutzt werden. Unterhalb des Headers wird die FeedbackApp eingeblendet. Durch das Verwenden desselben Layouts ist es für den Anwender nicht ersichtlich, dass der Dialog eine Komposition aus zwei Dialogen ist. Die Erfassung des Feedbacks ist in diesem Beispiel optional, wobei bei Bedarf das Feedback auch „erzwingen“ werden könnte (z.B. durch die Markierung des Eingabefelds als Pflichtfeld). Als dritter Bereich wird der individuell entwickelte User Dialog angezeigt.

Zur Visualisierung des Feedbacks dient eine weitere, mit dem Axon.ivy Designer implementierte PrimeFaces Benutzeroberfläche (vgl. Abbildung 82). Diese Benutzeroberfläche ist ein eigenständiger User Dialog, welcher unabhängig von der FeedbackApp eingebunden und gestartet werden kann. Die Übersicht unterstützt beispielsweise den Geschäftsprozessverantwortlichen bei der Analyse (z.B. Analyse der Häufigkeit von Feedback) des in Echtzeit erfassten Feedbacks.

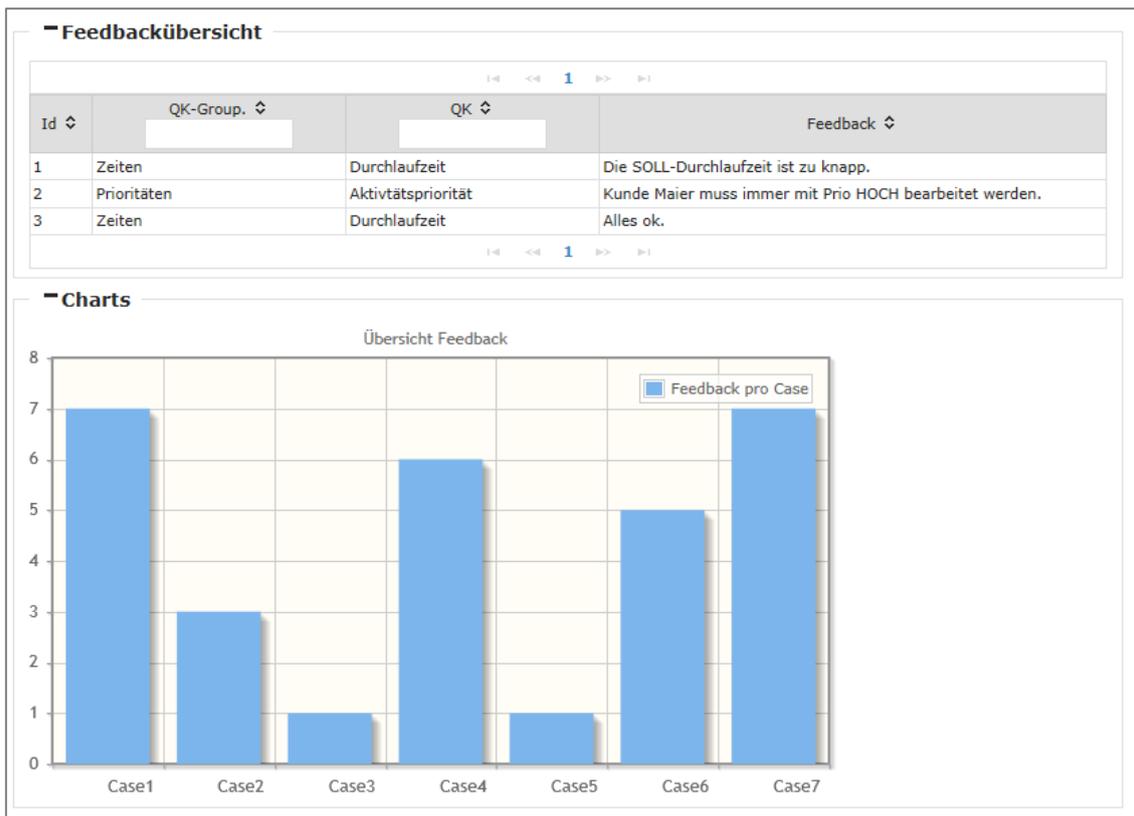


Abbildung 82: FeedbackApp User Interface - Übersicht

Die FeedbackApp-Übersicht ist in zwei Bereiche aufgeteilt. Der obere Teil listet das erfasste Feedback auf. Hierbei kann nach den einzelnen Spalten (Gruppe, OK, Feedback) sortiert und nach einzelnen Einträgen gesucht werden. Das Balkendiagramm im unteren Teil der Benutzeroberfläche stellt die Anzahl des erfassten Feedbacks pro Geschäftsprozessinstanz (Case) dar. Beispielsweise wurden während der Ausführung des Case1 sieben Anwenderfeedbacks erfasst. Eine Erweiterung von weiteren Informationen (z.B. Anzeige des Erfassungsdatums,...) ist durch eine Konfiguration der Feedbackübersicht-Spalten möglich.

Somit kann je nach Anforderung das erfasste Feedback ausgewertet werden. Die Analyse der Feedbackhäufigkeit und des gebrauchssprachlichen Feedbacks wird manuell durch die Prozessbeteiligten bzw. Prozessverantwortlichen durchgeführt. Hierbei ergänzt das erfasste Feedback die traditionell erhobenen SOLL- und IST-Daten (z.B. Liege- oder Durchlaufzeiten). Mögliche identifizierte Optimierungspotentiale, können dann direkt in den betroffenen Geschäftsprozess übernommen werden. Die Benutzeroberfläche in Abbildung 83 zeigt den MySQL Workbench Form Editor. Mit Hilfe des Form Editors können bequem Qualitätskriterien neu erfasst, editiert oder gelöscht werden.

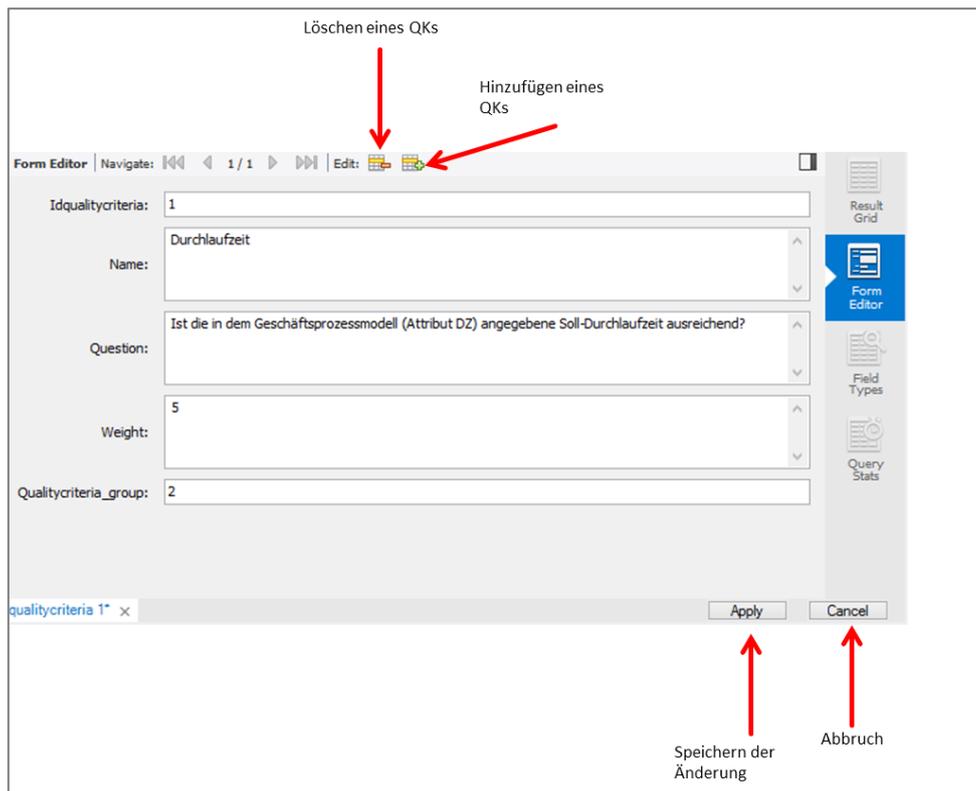


Abbildung 83: MySQL Workbench FormEditor

Das Qualitätskriterium „Durchlaufzeit“ besitzt beispielsweise die eindeutige ID „1“ und ist mit einer Anwenderfrage verbunden (vgl. Abschnitte 3.3 und 4.2.4). Die Gewichtung „5“ ist relativ (die Skala wird individuell z.B. durch den Administrator festgelegt). Des Weiteren ist das Qualitätskriterium der Gruppe „2“ zugeordnet.

5.2. Entwicklung eines dialogische Logik-basierten BPMN^{Easy} Pattern (Z2.2)

Aktivitäten in Geschäftsprozessen können zunehmend mehr automatisiert werden. Zum Beispiel ermöglicht die Decision Modeling Language (DMN) [OM15] eine formale Beschreibung von Regeln, welche während der Laufzeit automatisiert angewendet werden können. Trotz der kontinuierlich steigenden Digitalisierung und automatisierten Entscheidungsfindung sind menschliche Dialoge weiterhin von hoher Bedeutung. Insbesondere wissensintensive bzw. entscheidungsintensive Aktivitäten (vgl. [DMR15]) sind hierbei eine Herausforderung für die Erfassung und Ausführung im Kontext des Geschäftsprozessmanagements. Auf Basis der dialogischen Logik können Dialoge zwischen Proponent und Opponent gesteuert werden (vgl. Abschnitt 2.1.4). Dialogische Logik-basierte Dialoge führen hierbei zu einem, für beide Parteien (Proponent, Opponent), fairen und transparenten Ergebnis.

In diesem Abschnitt wird ein Geschäftsprozesspattern entwickelt, welches zur Modellierung von Dialogen zur (manuellen) Entscheidungsfindung genutzt werden kann. Hierfür werden zuerst der Begriff des Patterns und existierende Ansätze dargestellt. Anschließend wird die Implementierung des dialogischen Logik-basierten Patterns beschrieben.

5.2.1. Pattern

Pattern²¹² sind im Allgemeinen abstrakte Lösungsmuster zur Behandlung von bestimmten Problemen. Die Lösungsmuster sind hierbei Technologie- bzw. Notations-unabhängig. In der Literatur existiert eine Vielzahl von Definitionen des Begriffs „Pattern“. Alexander et al. [AIS77]²¹³ beschreiben ein Pattern als eine Regel, welche die Beziehung zwischen einem Problem, einem Kontext und einer Lösung darstellt. Beispielsweise können Pattern als Entwurfsmuster einer Softwarearchitektur im Kontext der Softwareentwicklung dienen (vgl. [Ha05b, S.11ff]). Gamma et al. [Ga95]²¹⁴ definieren Pattern als angepasste

²¹² In der Literatur wird zusätzlich der deutsche Begriff „Muster“ oder „Entwurfsmuster“ verwendet.

²¹³ Vgl. engl. Übersetzung: „Each pattern is a three part rule, which expresses a relation between a certain context, a problem, and a solution.“ [AIS77, S.247].

²¹⁴ Vgl. engl. Übersetzung: „(...) descriptions of communicating objects and classes that are customized to solve a general design problem in a particular context.“ [Ga95, S.3].

Objekte und Klassen, welche ein Entwurfsproblem in einem bestimmten Kontext lösen. Für Riehle & Züllighoven [RZ96]²¹⁵ ist das Wiederauftreten von Problemen im Mittelpunkt der Patterndefinition. Es wird somit die Prämisse aufgestellt, dass eine erfolgte Problemlösung auch für andere (zukünftige) Probleme nutzbar ist [Fo96]. Des Weiteren ist nach Riehle & Züllighoven ein Pattern eine Abstraktion eines konkreten Problems. Definitionsübergreifend besitzen Patterns hierbei immer (mindestens) eine Problembeschreibung, Lösung und eine Kontextbeschreibung [Ha05b]. Für einen Überblick über weitere Patternschemata sei auf [Ga95][RZ96][FE03][Ha05b] verwiesen.

Nach Fowler [Fo96, S.10] bietet die Verwendung von Pattern auch Chancen für die Modellierung von (Geschäfts-)Prozessen. Glushko & McGrath [GM05, S.318ff] nennen hierzu folgende Vorteile von Pattern: 1) Vereinfachung der Arbeit, indem durch die Verwendung von Pattern die Konsistenz und Standardisierung erhöht wird. 2) Nutzung von „Best Practices, sodass bereits bewährte Vorgehensweisen wiederverwendet werden können. 3) Unterstützung von Analysen durch z.B. den Vergleich von neuen Geschäftsprozessen und deren Struktur mit bestehenden Patterns. 4) Identifizierung von Ineffizienzen 5) Entfernung von Redundanzen 6) Vereinheitlichung von Schnittstellen 7) Verbesserung der Modularität und Austauschbarkeit.

Im Rahmen der Workflow Patterns Initiative²¹⁶ wurde eine Sammlung von Pattern, welche im Kontext des Geschäftsprozessmanagements Anwendung findet, veröffentlicht. Beispielsweise können die Patterns zur Evaluation einer Workflow-Engine oder einer Geschäftsprozessmodellierungssprache eingesetzt werden. Das Set von Pattern dient hierbei als Maßstab zur Messung des Funktionsumfangs. Die Workflow Patterns Initiative teilt die Pattern in die in Tabelle 40 dargestellten vier Kategorien auf.

Tabelle 40: Typen von Workflow Patterns (Workflow Patterns Initiative)

Typ	Beschreibung
Control-flow perspective	Die Kontrollflusspatterns enthalten Lösungsmuster zur Behandlung von Kontrollflussabhängigkeiten. Unter anderem wird beschrieben wie mit der Aufsplittung eines

²¹⁵ Vgl. engl. Übersetzung: „A pattern is the abstraction from a concrete form which keeps recurring in specific non-arbitrary contexts.“ [RZ96, S.1].

²¹⁶ <http://www.workflowpatterns.com>

	Pfades in mehrere parallel-ausführbare Pfade umgegangen werden kann, (vgl. [Aa03b][Ru06]).
Data perspective	Die Pattern der Datenperspektive beschreiben die Behandlung von Daten eines Geschäftsprozesses. Zum Beispiel wird erläutert wie Datenelemente zur Speicherung von individuellen Tasks einer Geschäftsprozessinstanz genutzt werden können, (vgl. [Ru04]).
Resource perspective	Die Zuweisung, Delegation etc. von Ressourcen zu Aktivitäten wird mit den Ressourcenpattern adressiert. Beispielsweise beschreibt das „Direct Distribution“-Pattern die Funktion einer direkten Zuweisung einer konkreten Ressource zu einer Geschäftsprozessaktivität während der Modellierung, (vgl. [Ru05]).
Exception handling perspective	Die Anforderungen an eine Workflow-Engine (oder prozessorientierten Softwareapplikationen) in Bezug auf Exception Handling wird von van der Aalst et al. unter der „Exception handling perspective“ zusammengefasst. Es werden hierbei Lösungsmuster präsentiert, die beschreiben, wie beim Eintritt einer Ausnahme (Exception) mit der betroffenen Aktivität und Instanz während der Ausführung verfahren werden muss (vgl. [RAH06]).

Mit der Modellierungssprache BPMN können, die von van der Aalst et al. definierten, Workflow Patterns modelliert werden können (vgl. White [Wh04] und Wohed et al. [Wo05]). Die Control-Flow Patterns [Aa03b][Ru06] enthalten hierbei unter anderem das sogenannte „Structured Loop“-Pattern. Das Pattern beschreibt die Ausführung einer Aktivität (Task oder Subprozess) in einer Schleife (loop). Die Abbruchbedingung (d.h. wie oft die Schleife durchlaufen wird) kann sowohl durch einen „Pre-Test“ (am Anfang der Schleife), als auch „Post-Test“ (am Ende der Schleife) überprüft werden. Die „Post-

Test“-Variante des „Structured Loop“-Patterns ist die Grundlage des dialogischen Logik-Patterns. Die Abbildung 84 stellt ein in BPMN modelliertes Beispiel einer strukturierten Schleife (structured loop) dar.

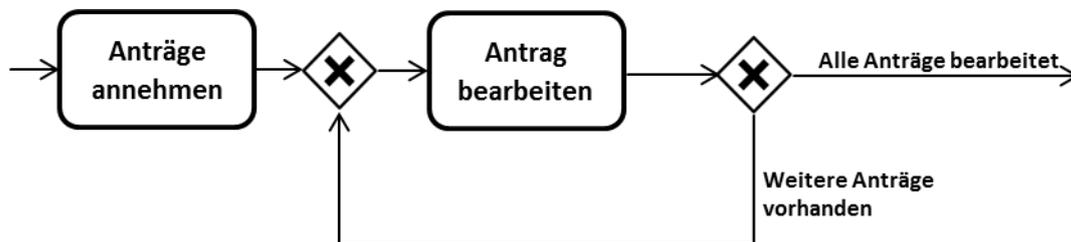


Abbildung 84: Loop Pattern

Im Beispiel (vgl. Abbildung 84) werden eine beliebige Anzahl von Anträgen angenommen („Anträge annehmen“).

Nachfolgend wird die Schleife gestartet und der erste Antrag bearbeitet („Antrag bearbeiten“). Nach der Bearbeitung wird die Bedingung (XOR-Gateway) überprüft, ob noch weitere Anträge zur Bearbeitung anstehen oder nicht. Falls ja, führt der alternative Sequenzfluss „weitere Anträge vorhanden“ zurück, sodass die Aktivität „Antrag bearbeiten“ für den nächsten Antrag ausgeführt werden kann. In BPMN kann durch einen Schleifenindikator eine Aktivität als Schleife markiert werden. Das explizite Modellieren von Schleifen zum mehrmaligen Durchlaufen einer Aktivität ist somit nicht unbedingt notwendig. Aus Gründen der Lesbarkeit ist die explizite Modellierung einer Schleife aber dennoch empfehlenswert (vgl. [FS12]).

5.2.2. Dialogische Logik-basiertes BPMN^{Easy} Pattern

Können bei der Ausführung eines Geschäftsprozess nicht direkt Entscheidungen automatisiert getroffen werden, ist es notwendig, dass die Entscheidungsfindung durch ein menschliches Eingreifen vollzogen wird. Die Entscheidungsfindung kann hierbei durch einen Dialog (z.B. zwischen zwei Anwendern) stattfinden. Zum Beispiel ist es im Kontext eines Investmentantragsprozesses verpflichtend, dass der Antragssteller Argumente äußert, warum das Investment (z.B. Anschaffung einer neuen Unternehmenssoftware) getätigt werden soll. Sind bereits andere Investments geplant, muss eine Diskussion bzgl. der zusätzlichen Anschaffung geführt werden. Der Antragssteller startet den Dialog mit einer Aussage. Im Folgenden argumentieren die Dialogteilnehmer solange, bis alle Argumente der Aussage geprüft und ein gemeinsamer

Konsens zur Entscheidungsfindung gefunden wurde. Ein weiteres Beispiel stellt eine Workflow-Engine dar, die automatisiert alle Preisrabatte bis 40% auf Basis eines ex ante definierten Regelwerks bearbeitet. Beispielsweise wird bei einem Preisnachlass von 10% automatisiert ein Vermerk in der Verkaufshistorie gespeichert, während bei einem Preisnachlass von 25% zusätzlich ein Manager informiert wird. Wird ein Preisnachlass größer 40% gemeldet, muss dieser Rabatt ausgehandelt werden. Der zuständige Vertriebsmitarbeiter formuliert hierbei seine Argumente in einer Aussage. Ob der Preisnachlass genehmigt wird entscheidet das Ergebnis des Dialogs. Das Konzept des in Abschnitt 5.2.1 eingeführten „Structured Loop“-Patterns wird zur Modellierung von Dialogregeln verwendet. Hierbei wird solange eine Schleife durchlaufen, bis alle Elementaraussagen auf Basis der dialogischen Logik geprüft wurden. Im Folgenden werden die für eine vollständige Beschreibung eines Patterns notwendigen Bestandteile auf Basis des Patternschemas von Förster & Engels [FE03] beschrieben. Nach Förster & Engels muss ein Pattern zur Beschreibung von Geschäftsprozessen folgenden Inhalt aufweisen: 1) Name des Patterns 2) Beschreibung 3) Anwendungskontext 4) Organisatorischer Kontext²¹⁷ 5) Problem 6) Lösung 7) Ergebnis 8) Gründe der Anwendung 9) Hinweis auf verwandte Pattern. Im folgenden Abschnitt wird das dialogische Logik-basierte Pattern beschrieben:

Name

Strukturierter Dialog

Beschreibung

Der strukturierte Dialog dient zur Entscheidungsfindung und transparenten Interaktion zwischen zwei Dialogpartnern.

Anwendungskontext

Das Pattern ist einsetzbar, falls Entscheidungen nicht automatisiert (auf Basis ex ante definierten Regeln) getroffen werden können, sondern ein Dialog zur Erreichung einer Entscheidung geführt werden muss.

²¹⁷ Vgl. [FE03, S.5]: „(...) A specialty concerning patterns for business processes is the section “Organizational Context”. A business process pattern usually requires the existence of certain organizational units that are responsible for performing activities. In this section all information about such organizational units is put together.“

Organisatorischer Kontext

Nicht-automatisierte bzw. nicht-automatisierbare Entscheidungen werden durch zwei Dialogpartner (Proponent, Opponent) getroffen. Eine Organisation verfügt über eine Rollen- und Hierarchiestruktur die die Ausführung dialogisch Logik-basierter Dialoge erlaubt.

Problem

Bei der Ausführung von Geschäftsprozessen kann der Zustand eintreten, dass vordefinierte Regeln nicht zur weiteren Steuerung ausreichend sind. In diesem Fall müssen menschliche Teilnehmer über den weiteren Pfad der Geschäftsprozessinstanz entscheiden. Hierbei können ungeprüfte Aussagen (z.B. nicht-wahre Behauptungen einer Sache) die Entscheidungsfindung innerhalb eines ausgeführten Geschäftsprozesses negativ beeinflussen.

Lösung

Die Partikel- und Strukturregeln (vgl. Abschnitt 2.1.4) der dialogischen Logik dienen als Grundlage des Geschäftsprozesspatterns. Zur Darstellung der Lösung wurde die Modellierungssprache BPMN^{Easy} gewählt. BPMN^{Easy} ermöglicht zum einen die grafische Abbildung bestimmter Dialogregeln, zum anderen können grafisch schwierig abbildbare Dialogregeln mit Hilfe von Mediafiles (vgl. Abschnitt 4.3) annotiert und für Anwender gebrauchssprachlich dokumentiert werden. Die Abbildung 85 stellt das modellierte Pattern dar.

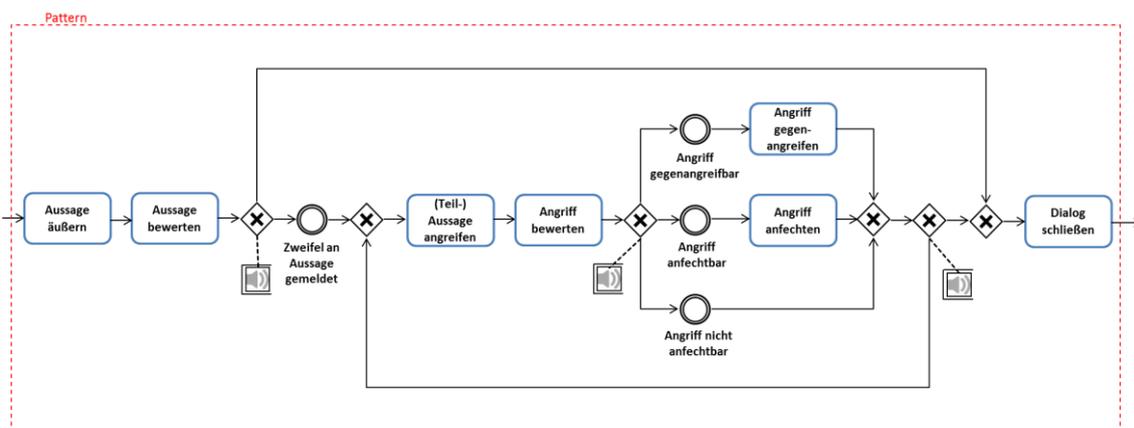


Abbildung 85: Darstellung des dialogischen Logik-basierten Patterns in BPMN^{Easy}

Die Aktivitäten, Ereignisse und Gateways des Patterns werden in Tabelle 41 beschrieben. Aufgrund der besseren Lesbarkeit wird zur Beschreibung einiger Aktivitäten ausschließlich das Wort „Angriff“ genutzt, auch wenn es sich hierbei um einen Gegenangriff oder einer Verteidigung handeln kann.

Tabelle 41: Beschreibung der Aktivitäten und Ereignisse des Patterns

Aktivitäten	Beschreibung
Aussage äußern	Damit ein strukturierter Dialogs gestartet werden kann, muss eine Aussage durch einen Proponent getätigt werden (1.Strukturregel).
Aussage bewerten	Innerhalb dieser Aktivität kann der Opponent bewerten, ob dieser einen Dialog starten möchte. Entscheidungen die ohne Dialog getroffen werden können erfolgen somit ohne Verzögerung direkt.
(Teil-)Aussage angreifen	Beim ersten Durchlauf der Schleife startet der Opponent mit einem Angriff auf die durch den Proponenten formulierte Aussage (bzw. auf einer Teilaussage dieser). Im weiteren Durchlauf der Schleife muss die Aktivität abwechselnd dem Proponenten und Opponenten zugewiesen werden.
Angriff bewerten	Um einen Konsens bilden zu können, muss bewertet werden, wie mit dem zuvor durchgeführten Angriff verfahren werden soll. Hierbei kann beispielsweise ein „Faktencheck“ durchgeführt werden.
Angriff gegenangreifen	Falls erlaubt, kann (z.B. durch eine Negation, 1.Partikelregel) ein Angriff auf eine vorangehende Behauptung des gegnerischen Dialogpartners durchgeführt werden.
Angriff anfechten	Eine Verteidigung auf einen vorhergehenden gegnerischen Angriff gemäß den Partikelregeln.

Dialog schließen	Ein Dialog kann geschlossen werden, wenn dem gerade Ausführenden kein erlaubtes Argument mehr zur Verfügung steht. Der Dialogabschluss dient zur Synchronisation der Entscheidung (z.B. finales Abschlussgespräch über den gefundenen Konsens).
Ereignisse	Beschreibung
Zweifel an Aussage gemeldet	Die Ereignisse markieren einen hervorzuhebenden Status innerhalb des Patterns. Hierbei sind die Ereignisse dem Zustand entsprechend benannt. Beispielsweise markiert das Ereignis „Zweifel an Aussage gemeldet“ den Start der Schleife und somit des strukturierten Dialogs.
Angriff gegenangreifbar	
Angriff anfechtbar	
Angriff nicht anfechtbar	
Gateways	Beschreibung
XOR Gateway (Audio 1)	Der strukturierte Dialog soll nur stattfinden, wenn ein Bedarf besteht (z.B. durch einen Zweifel).
XOR Gateway (Audio 2)	Nach Bewertung des Angriffs bietet das Gateway drei Alternativen: 1. Der Angriff ist gegenangreifbar, 2. Der Angriff ist anfechtbar (verteidigbar) oder 3. Der Angriff kann weder verteidigt noch gekontert werden.
XOR Gateway (Audio 3)	Nach Klärung aller Elementaraussagen endet der Dialog. Alternativ routet das Gateway zurück zum Beginn der Schleife, sodass ein weiterer Zug durchgeführt werden kann.

Ergebnis

Als Ergebnis erhalten die Dialogpartner eine auf Wahrheitsbedingungen geprüfte Aussage.

Verwandte Patterns

Das „Strukturierte Dialog“-Pattern ist mit dem „Structured loop“-Pattern (vgl. [Aa03b][Ru06]), aufgrund des strukturierten Durchlaufens einer Schleife bis zu einer definierten Abbruchbedingung, verwandt.

Gründe der Anwendung und kritische Bewertung

Die dialogische Logik erhöht die Möglichkeiten der Kreativität bei Entscheidungsprozessen (z.B. in dem beide Dialogpartner gleichberechtigt das Recht zum „Angriff“ und der „Verteidigung“ besitzen). Zudem können nicht-ex ante definierbare Entscheidungsregeln durch den „Strukturierten Dialog“ kompensiert werden. Nachteile des Patterns ergeben sich durch die Komplexität der zugrundeliegenden Partikel- und Strukturregeln (im vorliegenden Beispiel wurde das Pattern mit BPMN^{Easy} umgesetzt, wobei komplexe Regeln mit Hilfe von gebrauchssprachlichen Informationen per z.B. Audiosequenz annotiert werden, sodass die Dialogpartner zusätzlich bei der Ausführung des Dialogs unterstützt werden). Des Weiteren kann die Erreichung einer transparenten Dialogführung und eines fairen Ergebnisses durch die Länge des Dialogs negativ beeinflusst werden, da die dialogische Logik die Dauer eines Dialogs nicht ausreichend beschränkt. Ist beispielsweise im Moment der Dialogführung eine Elementaraussage nicht direkt entscheidbar, tritt der Zustand „non liquet“ ein, welcher die Ausführung von realen Geschäftsprozessinstanzen auf unbestimmte Zeit verzögert.

6. Anwendungsbeispiel

In diesem Kapitel wird ein durchgängiges Anwendungsbeispiel von BPM(N)^{Easy} beschrieben. Die Basis des Anwendungsbeispiels bilden Projekte, welche im Rahmen der Tätigkeiten am kips und bei der AXON IVY AG durchgeführt wurden. Unter anderem war eine Zielsetzung (vgl. [Ku15][MOW15]) des „Smart City²¹⁸ Konstanz“-Projekts den Geschäftsprozess der „Bewohnerparkausweisbeantragung“ zu erfassen, um diesen (zu einem späteren Zeitpunkt) automatisiert über ein Internetportal ausführen zu können. Die Anforderungen wurden von den Fachanwendern und Bürgern der Stadt Konstanz formuliert und mit dem Softwarewerkzeug BPM Touch digital abgespeichert. Darüber hinaus wurden Qualitätskriterien (z.B. Verfügbarkeit der IT-Systeme) erfasst und in der Modellierung berücksichtigt. Eine ganzheitliche Sicht auf die von BPMN^{Easy} geforderten Anker (Interaktion – Synchronisation – Qualität) war daher möglich. Des Weiteren wurde in Zusammenarbeit mit der AXON IVY AG die Entwicklung von Geschäftsprozessapplikationen auf Basis der agilen Vorgehensweise durchgeführt. Durchgeführte (und laufende) Projekte zeigen, dass insbesondere bei der Geschäftsprozessautomatisierung, die agile Einbindung von Beteiligten und kurze Iterationen Erfolgsfaktoren für das Gelingen der Projekte sind. Zum Beispiel wurde ein Geschäftsprozess realisiert mit welchem die Anmeldung einer neuen Stromerzeugungsanlage ermöglicht wird. Zu Beginn des Projektes waren hierbei noch keine eindeutigen Aussagen über konkreten Geschäftsprozessablauf möglich. Darüber hinaus war es nötig, alle Beteiligten (Fachanwender, IT-Experten, Drittsystem-Anbieter und BPM-Experten) aufgrund der hohen Informationsasymmetrie in kurzen Abständen zu synchronisieren. Zusätzlich wurde der entwickelte Software-Prototyp (zur agilen Erfassung von Anwender-Feedback) prototypentechnisch getestet (vgl. [GMW15]). Dazu wurden während der Ausführung von Geschäftsprozessen die Anwender automatisiert zu Qualitätskriterien befragt (z.B. „Ist die berechnete Durchlaufzeit der ausgeführten Aktivität ausreichend?“). Das Feedback wurde zur Optimierung des Geschäftsprozesses während der nächsten BPM(N)^{Easy} Iteration verwendet (z.B. um Erwartungen bzgl. der Durchlaufzeit an Realwelt anzupassen).

²¹⁸ Der Begriff “Smart City” wird in der Literatur unterschiedlich definiert, wobei der Aspekt der Integration von IT in den “Alltag” von Städten und deren Bürgern den Definitionen gemeinsam ist (vgl. [Ne14] [Sa14] [BJA15]). Anwendungen im Kontext von Smart Cities sind zum Beispiel internetbasierte Bürgerportale (vgl. [Co15a]).

Fortführend zu Abschnitt 2.3.1 wurden Teilnehmer zur Einschätzung der Stärken, Schwächen, Chancen und Risiken (vgl. SWOT-Analyse²¹⁹) von agilen Ansätzen befragt („Wie schätzen Sie Agilität im Kontext von BPM ein?“). Das Ziel dieser Befragung war es, Erkenntnisse der allgemeinen Akzeptanz agiler Ansätze in der betrieblichen Anwendung zu erhalten bzw. zu evaluieren. Die Abbildung 86 stellt exemplarisch ein Ergebnis einer durchgeführten SWOT-Analyse dar.



Abbildung 86: Beispiel einer SWOT-Analyse

Als eine besondere Stärke (vgl. Abbildung 86 „Big Picture, Greifbar, Geschwindigkeit“) agiler Vorgehensweisen wurde die Möglichkeit des schnellen Fortschritts und die dadurch verbesserte „Greifbarkeit“ genannt. Zum Beispiel wird eine anfangs ungenaue Projektvision, von agiler Iteration zu Iteration, immer mehr zu einer konkreten Lösung (z.B. in Form eines Geschäftsprozessmodells). Daraus resultiert ein gemeinsam wachsendes Verständnis aller Beteiligten (z.B. für neue Geschäftsprozessaktivitäten oder implementierte Funktionen). Des Weiteren (vgl. Abbildung 86 „Rollenverteilung“) werden die festgelegten Rollen als signifikanter Mehrwert der agilen Vorgehensweise angesehen. Durch die Rollen (z.B. BPM Master) werden Verantwortlichkeiten definiert, sodass Zuständigkeiten, wie beispielsweise die Priorisierung der Umsetzung von Anforderungen an die Geschäftsprozesse, transparent verteilt sind.

²¹⁹ Die SWOT-Analyse ist ein Ansatz zur Sammlung und Formulierung von Stärken (S=engl. Strength), Schwächen (W= engl. Weaknesses, Chancen (O=engl. Opportunities) und Risiken (T=engl. Threats) im Kontext einer Unternehmens –oder Umfeldanalyse, vgl. des Weiteren [Pa03][Sc14a].

Die fehlende „Kundenbindung“ wurde als Schwäche agiler Vorgehensweisen genannt. Unter anderem wurde festgestellt, dass das agile Vorgehen bei Kunden (z.B. Anwender aus einer Fachabteilung) mit keiner oder wenig Erfahrung im Bereich der Agilität schwierig einzuführen ist. Insbesondere die kurzen Abstimmungszyklen und die damit verbundene „Schritt-für-Schritt“- Implementierung bedingt ein Umdenken traditioneller Vorgehensweisen. Dazu anschließend wurde das Risiko der „einheitlichen Integration“ beschrieben. Durch die inkrementelle Implementierung erhöht sich erfahrungsgemäß die Komplexität der Mensch- und Systemintegration. Beispielsweise erfordert die „stückweise“ Integration von Drittsystemen (z.B. Datenbanken) einen erhöhten Aufwand der Koordination, da Schnittstellen teilweise nicht komplett getestet werden können und somit viele Einzeltests nötig sind. Als weiteres Risiko wurde die „Projektorganisation“ beschrieben (vgl. Abbildung 86 „Zuständigkeit, Zeit“). Gelingt es beispielsweise nicht, ausreichend Zeit für Aufgaben in einem BPM Projekt zu reservieren, können die Beteiligten in vielen Fällen nicht ihren Zuständigkeiten nachkommen. In diesem Fall entsteht das Risiko, dass die agile Vorgehensweise nicht konsequent umgesetzt werden kann. Konträr dazu werden als Chancen insbesondere die „enge Zusammenarbeit“ aller Beteiligten genannt. Hierbei sind nicht nur die regelmäßigen Treffen aller Beteiligten zur Interaktion während eines BPM Projekts von Bedeutung, sondern auch die kontinuierliche Synchronisation während der produktiven Nutzung der Geschäftsprozesse. „Akzeptanz erhöhen“ ist ein weiteres Thema, welches als Chance von agilen Vorgehensweisen erfasst wird. Durch die ganzheitliche Einbindung aller Interessensgruppen (z.B. während der Modellierung oder in Zwischentests) ist jeder Beteiligte ein Teil des BPM Projekts. Durch die eigene Mitarbeit folgt unmittelbar eine Erhöhung der Akzeptanz (z.B. von Aktivitäten in einem Geschäftsprozess).

In den folgenden Abschnitten wird im Rahmen eines Anwendungsbeispiels beschrieben, inwieweit derartige Stärken, Schwächen, Chancen und Risiken mit der agilen Vorgehensweise von BPM(N)^{Easy} in Verbindung stehen. Zur besseren Lesbarkeit wird die Beschreibung in zwei Abschnitte gegliedert. In Abschnitt 6.1 wird das Szenario skizziert, d.h. die für das Anwendungsbeispiel geltenden Rahmenbedingungen beschrieben. In Abschnitt 6.2 werden die einzelnen Schritte der BPM(N)^{Easy} Vorgehensweise und die Anwendung der Sprache und des Software-Prototyps ausführlich dargestellt, wobei die Anwendung von BPM(N)^{Easy} über zwei Iterationen erläutert wird.

Die Abschnitte 6.1 und 6.2 basieren auf gesammelten Erfahrungen aus Forschung und der betrieblichen Anwendung, welche insbesondere,

- aus verschiedenen Projekten der AXON IVY AG, mitwirkend als BPM-Experte, IT-Experte, Projektleiter (Projekte mit bis 500 Personentagen, Zeitraum 2012 - 2016),
- aus Leitung von GPM-(Experten-)Workshops und Vorträgen auf Anwenderforen (u.a. Stuttgarter Softwaretechnik Forum, EVU Kongress Berlin, Process Solution Day Köln, PzM Summit Wien, Zeitraum 2012 - 2016) und
- aus der Mitarbeit und Veröffentlichungen im Rahmen von Forschungsaktivitäten zu „Smart City Constance“ und „BPM@Cloud“ des kips (Zeitraum 2014 - 2016)

stammen.

6.1. Szenario

Das Szenario beschreibt den Einsatz von $\text{BPM(N)}^{\text{Easy}}$ zur Erfassung, Anreicherung, Ausführung/Analyse und Optimierung eines Angebotsprozesses. Im Allgemeinen kann $\text{BPM(N)}^{\text{Easy}}$ sehr flexibel gestartet und angewendet werden. Im vorliegenden Szenario wird als Startpunkt die erstmalige Erfassung gewählt. Wie in Abbildung 87 visualisiert, werden nach dem Start zwei Iterationen durchlaufen. Hierbei umfasst die erste Iteration den gesamten Zyklus, die zweite Iteration wird auf die Schritte der weiteren Erfassung und Anreicherung beschränkt.

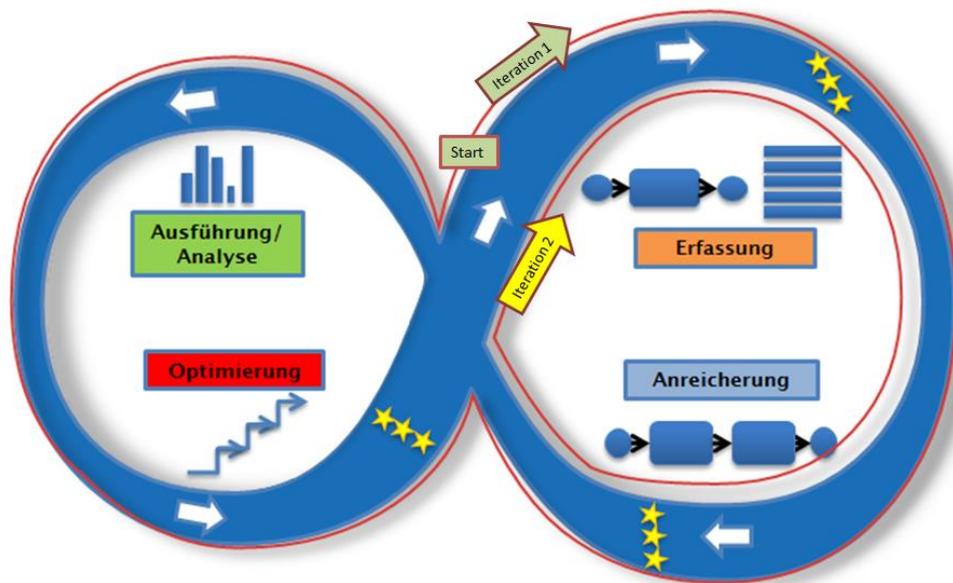


Abbildung 87: Darstellung des Ablaufs des vorliegenden Szenarios

Um das Anwendungsbeispiel ganzheitlich beschreiben zu können, werden im Folgenden die einzelnen $\text{BPM(N)}^{\text{Easy}}$ -Teammitglieder mit Hilfe von Persona-Profilen spezifiziert. Nach Cooper ist eine Persona eine fiktive Person, die als Prototyp für eine konkrete Rolle definiert wird [Co99]. Eine Persona besteht beispielsweise aus einem Namen, Bild und einer Beschreibung von Eigenschaften [MBH09, S.364]. Die Verwendung einer Persona ermöglicht die Generierung eines gemeinsamen Verständnisses [Mo12], welches insbesondere bei der ersten Anwendung von $\text{BPM(N)}^{\text{Easy}}$ bei der Zuordnung der Rollen hilfreich ist.

Tabelle 42: Beschreibung der Persona

Name	Rolle	Profil
Sarah Hagen	BPM-Expertin	Sarah Hagen (29) ist studierte Wirtschaftsinformatikerin. Neben einer BPMN 2.0 Zertifizierung, besitzt Sarah Hagen mehrjährige Erfahrung bei der Modellierung von organisationsübergreifenden Geschäftsprozessen. Des Weiteren verantwortet Sie die Pflege der allgemeinen Prozesslandkarte ihres Unternehmens.
Philipp Meyer	IT-Experte	Philipp Meyer (41) arbeitet als Softwarearchitekt und Softwareentwickler in der internen IT-Abteilung der Organisation. Er ist Technologie begeistert und spezialisiert auf Webprogrammierung. Des Weiteren verantwortet er zusammen mit seinen Kollegen die Betreuung und Wartung des ERP-Systems.
Torsten Buchner	Anwender	Torsten Bucher (36) arbeitet als Vertriebsleiter in der Organisation. Er sorgt im täglichen Betrieb stets für einen zufriedenstellenden Kundenkontakt und bearbeitet neue Angebote gemeinsam mit den Kollegen der Produktion. Des Weiteren ist er Leiter des gesamten Vertriebsteams der Organisation und verantwortet die Weiterbildung seiner Kollegen und Kolleginnen.
Julia Lang	BPM(N) ^{Easy} -Master	Julia Lang (34) hat nach Ihrem BWL-Studium mehrere Jahre in einer Unternehmensberatung gearbeitet. Heute arbeitet Sie als Inhouse-Consultant im Bereich Qualitäts- und Geschäftsprozessmanagement und unterstützt interne Projekte Abteilungsübergreifend.

Das Szenario beschreibt einen Geschäftsprozess zur Erstellung eines Angebots. Die Abteilung Vertrieb erhält hierbei eine Kundenanfrage, wodurch eine Instanz des Geschäftsprozesses gestartet wird. Nach Erfassung der relevanten Daten (z.B. Kundenadresse, Ansprechpartner,...) wird der Angebotsentwurf an das Back Office weitergereicht. Dieses prüft beispielsweise die Liquidität und Anforderungsdetails des Kunden und entscheidet, ob das Angebot weiter bearbeitet oder direkt verworfen wird. Bei einer positiven Entscheidung wird die für die Produktion der angefragten Artikel verantwortliche Abteilung informiert. Das Angebot wird durch eine Analyse (z.B. Schätzung des Aufwands) verfeinert und an den Vertrieb zurückgegeben. Der Vertrieb finalisiert das Angebot und informiert den Kunden über das Angebot. Zur Unterstützung der Ausführung wird sowohl eine gesamte Dokumentation des Geschäftsprozesses benötigt (z.B. zur schnellen Einarbeitung von neuen Mitarbeitern), als auch eine Teilautomatisierung der Aktivitäten (z.B. Softwareunterstützung bei der Erfassung der Kundendaten). Zusätzlich müssen alle relevanten IT-Systeme (z.B. ERP-System) bei der Ausführung berücksichtigt und ggfs. eingebunden werden. Aufgrund der wachsenden Kundenanforderungen muss gewährleistet werden, dass die Durchlaufzeit der Erstellung eines Angebots möglichst kurz gehalten wird. Des Weiteren soll der Einsatz einer mobilen App die flexible Einbindung von Geschäftsprozessbeteiligten zusätzlich verbessern. Durch die mobile App kann User Feedback während der Ausführung agil erfasst werden. Die folgende Auflistung fasst den Ablauf des vorliegenden Szenarios zusammen, wobei das Anwendungsbeispiel direkt an eine reale betriebliche Anwendung angelehnt ist.

1. Start – Iteration 1

Das Projekt wird auf Grundlage einer existierenden Vision gestartet. Die Vision beschreibt die Möglichkeit, dass der Angebotsprozess zum einen für alle Anwender (z.B. Vertrieb, Kunden) verständlich dokumentiert werden muss, zum anderen soll auf der Grundlage des Geschäftsprozessmodells eine Geschäftsprozessapplikation implementiert werden, welche die Beteiligten bei der Ausführung durch die Digitalisierung von Angeboten unterstützt. Um diese Rahmenbedingungen besser einschätzen zu können, wird zum Start ein „Exploratory 360°“ durchgeführt und die Parameter für ein Burn down-Chart (z.B. Budget, Iterationsdauer) festgelegt.

2. Erfassung des ersten Geschäftsprozessmodells

Während des ersten Modellierungsworkshops ist das gesamte Team anwesend. Das Team besteht aus vier Mitgliedern, wobei jeweils ein Mitglied genau eine Rolle (Anwender, IT- und-BPM-Experte, BPM(N)^{Easy}-Master) einnimmt. Innerhalb des Workshops werden in Paarmodellierung das erste End-to-End-Geschäftsprozessmodell in BPMN^{Easy} erfasst und durch Annotation von Mediafiles ergänzt. Des Weiteren werden die ersten Qualitätskriterien definiert, welche nach endgültiger Fertigstellung des ersten Modellinkrements zur Kontrolle in einem Quality Gate abgefragt werden.

3. Anreicherung und Implementierung der ersten Geschäftsprozessapplikation

Das erfasste Geschäftsprozessmodell dient als Grundlage der Anreicherung. Der IT-Experte führt im ersten Schritt ein Story Pulling durch. Zusätzlich zu der Auswahl der zu implementierenden Anforderungen wird gleichzeitig die FeedbackApp Komponente integriert, um bei der Ausführung Feedback bzgl. der Qualitätskriterien erfassen zu können. Nach Abschluss der Implementierung nehmen BPM(N)^{Easy}-Master, Anwender und IT-Experte an einem weiteren Synchronisationsmeeting teil.

4. Ausführung und Analyse

Nach erfolgreicher Abnahme des Geschäftsprozessmodells und der dazugehörigen Applikation kann die Ausführung gestartet werden. Der Software-Prototyp FeedbackApp wird hierbei zur Erfassung von Feedback eingesetzt. Jedes Feedback wird als User Story markiert und bildet eine mögliche Grundlage zur Optimierung des Geschäftsprozesses innerhalb der zweiten Iteration.

5. Optimierung

Das Szenario beschreibt zudem das innerhalb des Optimierungsschritts durchgeführte Shaping. Die Teammitglieder diskutieren hierbei die Erfahrungen der vergangenen Iteration. Hierdurch wird neben der Priorisierung des Feedbacks (zur weiteren Optimierung des Geschäftsprozesses) auch die Vorgehensweise selbst optimiert (z.B. Anpassung der Iterationsdauer).

6. Start Iteration 2

Im beschriebenen Szenario wird der Optimierungsschritt von Iteration 1 mit dem Start der Iteration 2 zusammengefasst und innerhalb eines Synchronisationsmeetings durchgeführt. Der BPM(N)^{Easy}-Master weist hierbei wiederholt auf die Projektrahmenbedingungen hin und unterstützt bei der Lösung von Problemen (z.B. Einhaltung des Budgets).

7. Erfassung der neuen Anforderungen

Innerhalb des Erfassungsschritts der zweiten Iteration wird das BPMN^{Easy}-Modell auf Basis des Feedbacks optimiert. Zusätzlich werden Mediafiles aktualisiert bzw. weitere Anforderungen an das Geschäftsprozessmodell annotiert. Des Weiteren wird im Anwendungsbeispiel dargestellt, wie neue Qualitätskriterien während des laufenden Projekts hinzugefügt werden können.

8. Update der Geschäftsprozessapplikation

Zum Abschluss des Anwendungsbeispiels wird im Szenario die während der Iteration 1 erstellte Geschäftsprozessapplikation in Bezug auf die neuen Anforderungen erweitert. Hierzu wird beschrieben, wie insbesondere die Interaktion von IT-Experte und Anwender zur Synchronisation und Qualitätssicherung der Geschäftsprozessapplikation dient.

6.2. Anwendung

In diesem Abschnitt wird, basierend auf der Beschreibung in Abschnitt 6.1, der Ablauf eines BPM(N)^{Easy}-Projekts beschrieben. Zu Beginn eines BPM(N)^{Easy}-Projekts werden mit Hilfe des „Exploratory 360°“ die Rahmenbedingungen des GPM-Projekts „erkundet“. Neben der Benennung der Beteiligten (d.h. der Bestimmung der Anwender, Experten, BPM(N)^{Easy}-Master) wird eine erste „grobe“ Anforderungsanalyse durchgeführt. Durch die Anforderungsanalyse wird insbesondere der „Business Value“²²⁰, d.h. der Beitrag zur langfristigen Sicherung und Steigerung des Unternehmenswerts, des anstehenden Projekts erkundet. Des Weiteren wird die „grobe“ technische Machbarkeit geprüft, sodass ausgeschlossen werden kann, dass ein Projekt

²²⁰ Für eine ausführliche Definitionsübersicht des Begriffs „Business Value“ sei auf [RDS09] verwiesen.

aufgrund von nicht-realisierten Anforderungen abgebrochen werden muss. Abschließend wird ein erster Projektplan erstellt.

Im Anwendungsbeispiel wird ein Projekt mit annähernd zwei Iterationen durchlaufen. Ein Burn down Chart, wie in Abbildung 88 dargestellt, wird zur Visualisierung der Schritte genutzt.

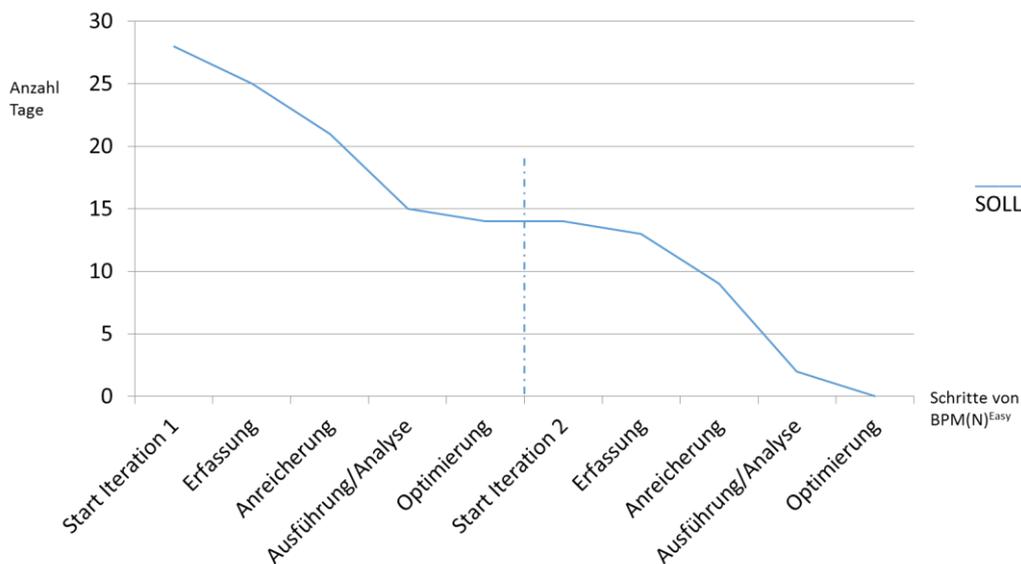


Abbildung 88: Burn down Chart über zwei Iterationen

Die y-Achse des Burn down Charts spiegelt die Anzahl der verbleibenden Tage des Projekts wider. Die 28 Projekt Tage werden hierbei in zwei Iterationen mit jeweils 14 Tagen Laufzeit unterteilt. Die x-Achse ist durch die einzelnen BPM(N)^{Easy}-Schritte unterteilt. Durch die Steigung der einzelnen Kurvenabschnitte (vgl. Abbildung 88 blaue SOLL-Linie) kann der zeitliche Aufwand ermittelt werden. Beispielsweise sind in der ersten Iteration für die Anreicherung des Geschäftsprozessmodells zu einer Geschäftsprozessapplikation sechs Tage eingeplant.

Durch den „Exploratory 360°“ können die Beteiligten direkt relevante Qualitätskriterien identifizieren und dokumentieren.

Der im Anwendungsbeispiel beschriebene Angebotsprozess beinhaltet Aktivitäten, die direkten Kundenkontakt beinhalten. Um sicherzustellen, dass eine hohe Kundenzufriedenheit erreicht wird, muss der erfasste Geschäftsprozess ausführlich dokumentiert und die Ausführung der Aktivitäten auf Basis dieser Dokumentation ausreichend geschult werden. Dieses Qualitätskriterium wird in Anlehnung an

[LR13][GMW14b] mit dem von Lohrmann & Reichert [LR13] beschriebenen Qualitätsmerkmal „kontrollierter Einsatz von Fähigkeiten“ verbunden.

Die Formel (**) beschreibt die Formalisierung des Qualitätskriteriums, wobei die Analyse des gemessenen Qualitätsindikators einen direkten Überblick bietet, wie hoch der Grad von gebrauchssprachlich annotierten und geschulten Aktivitäten ist.

$$CSE(p) = \frac{DT(bp)+TT(bp)}{2} \quad (**)$$

Die Formel (**) setzt sich wiederum aus zwei Teilformeln (***) und (****) zusammen. Zudem wird im vorliegenden Anwendungsbeispiel definiert, dass die gebrauchssprachliche Dokumentation von manuellen und teilautomatisierten Aktivitäten und deren Schulung jeweils 50% in das Ergebnis zählen (der Prozentsatz 50% ist hierbei frei gewählt).

$$DT(bp) = \frac{|F(MUT(bp),t,D(t))|}{|MUT(bp)|} \quad (***)$$

$$TT(bp) = \frac{|F(MUT(bp),t,T(t))|}{|MUT(bp)|} \quad (****)$$

Die einzelnen Bestandteile der Formeln (**), (***) und (****) sind in Tabelle 43 beschrieben.

Tabelle 43: Bestandteile der Metrik CSE

Element	Beschreibung
CSE(bp)	Controlled Skill Employment: Die Funktion liefert als Ergebnis einen Wert zwischen 0 und 1, welcher den Anteil der im Geschäftsprozess bp dokumentierten bzw. trainierten Aktivitäten widerspiegelt.
DT(bp)	Documentation of Tasks: Anteil der (manuellen und teilautomatisierten) Aktivitäten im Geschäftsprozess bp, die durch eine gebrauchssprachliche Annotation dokumentiert sind.
D(t)	Documentation: wahr, wenn Aktivität t dokumentiert ist.
TT(bp)	Training of Tasks: Anteil der Aktivitäten t, welche im Geschäftsprozess bp trainiert ausgeführt werden.

T(t)	Training: wahr, wenn Aktivität t trainiert wurde.
MUT(bp)	Manual/User Tasks: Manuelle M oder teilautomatisierte U Aktivitäten (Tasks) im Geschäftsprozess bp.
F(e, v, c)	Filter: Filter, welche die Elemente e nach der Kondition c flitert, wobei v als Iterator verwendet wird.

Die möglichen Ergebnisse der Funktion CSE (vgl. Formel (**)) in Tabelle 44 aufgelistet.

Tabelle 44: Interpretation der Werte

Wert	Interpretation
0	Keine der manuellen oder teilautomatisierten Aktivitäten wurden trainiert und unterstützend durch eine gebrauchssprachliche Annotation dokumentiert.
Zwischen 0 und 1	Eine Teilmenge der manuellen oder teilautomatisierten Aktivitäten wurde trainiert und mit einer gebrauchssprachlichen Annotation angereichert dokumentiert.
1	Alle manuellen oder teilautomatisierten Aktivitäten wurden trainiert und gebrauchssprachlich annotiert dokumentiert.

Nach der Definition der Qualitätskriterien wird die Erfassung des Geschäftsprozessmodells gestartet. Anwender und BPM-Experte modellieren gemeinsam (Paarmodellierung) eine erste Version des Geschäftsprozessmodells mit der Modellierungssprache BPMN^{Easy}. Hierbei muss insbesondere sichergestellt werden, dass der Geschäftsprozess von Start bis Ende erfasst wird. Die Abbildung 89 stellt den, mit Hilfe von BPM Touch, modellierten Geschäftsprozess dar.

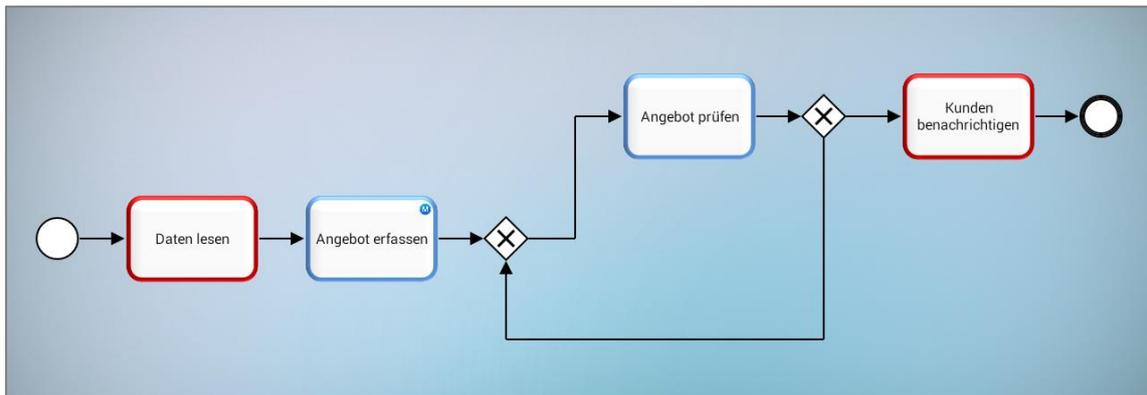


Abbildung 89: Angebotsprozess Iteration 1

Der erfasste Geschäftsprozess umfasst vier Aktivitäten, wobei neben der grob-granularen grafischen Repräsentation weitere Informationen mit Hilfe von Mediafiles annotiert wurden. Die Tabelle 45 listet die Mediafiles geordnet nach BPMN^{Easy}-Elementen auf.

Tabelle 45: Auflistung der erfassten Mediafiles

BPMN ^{Easy} -Element	Annotation
Daten lesen	Keine.
Angebot erfassen	Annotation in Form einer Grafik (vgl. Abbildung 90).
Angebot prüfen	Keine.
XOR Gateway (auf „Angebot prüfen“ folgend)	Audiofile zur gebrauchssprachlichen Beschreibung der Entscheidungsregeln: „ <i>Abhängig der detaillierten Analyse der Anforderungen des Kunden, ergänzt das Back Office das erfasste Angebot mit Informationen. Auf Basis dieser Informationen entscheide ich, ob das Angebot an den Kunden gesendet wird oder nicht.</i> “
Kunden benachrichtigen	Annotation eines Dokuments, welches ein E-Mail-Template beinhaltet. E-Mails, welche zur Benachrichtigung des Kunden versendet werden, müssen die Struktur des Template ausweisen.

Die Aktivitäten „Angebot erfassen“ und „Angebot prüfen“ sind teilautomatisierte Aktivitäten. Ziel ist es, die Anwender bei der Ausführung mit Benutzeroberflächen zu

unterstützen, welche auf deren Bedarf direkt angepasst sind. Das Mediafile, welches an die Aktivität „Angebot erfassen“ annotiert ist, bietet hierbei eine direkte Orientierungshilfe bei der Implementierung. Durch eine einfache grafische Skizze und gebrauchssprachlichen Kommentaren beschreibt der Anwender direkt seine Anforderungen. Anforderungen, welche aus Sicht des IT-Experten „andersartig“ implementiert werden müssen, werden innerhalb der Interaktionen und Synchronisation (Quality Gate 1) im Projekt direkt mit den anderen Beteiligten diskutiert. Beispielsweise wird diskutiert, welche Buttons zur Bedienung des Formulars notwendig sind. Die Abbildung 90 stellt das annotierte Multimediafile der Aktivität „Angebot erfassen“ dar.

The screenshot shows a user interface for the 'Angebot erfassen' (Offer Capture) activity. At the top, there is a horizontal process flow with four steps: 'Erfassen', 'Verfeinern', 'Prüfen', and 'Abschließen'. Below this, there are two input fields: 'Kunde' (Customer) and '8/22/2015'. A large empty box below these fields contains the text 'Hier werden Angebotdetails aufgelistet...'. At the bottom, there is a table with two columns: 'Artikel/Leistung' and 'Beschreibung'. The first row has a checked checkbox and the text 'Beschreibung...'. The second row has an unchecked checkbox and the text 'Beschreibung...'. The table is partially obscured by a grey bar.

▼ Artikel/Leistung	▼ Beschreibung
<input checked="" type="checkbox"/>	Beschreibung...
<input type="checkbox"/>	Beschreibung....

Abbildung 90: Gebrauchssprachliche Multimediaannotationen in Iteration 1

Mit dem Abschluss der Erfassung wird das erste Quality Gate erreicht. Der BPM(N)^{Easy}-Master verantwortet die Kontrolle der Qualitätskriterien. Im Rahmen des Anwendungsbeispiels wird hierdurch sichergestellt, dass bereits von Anfang die Dokumentation und Schulung der Aktivitäten ausreichend priorisiert sind. Des Weiteren dient das Quality Gate als Startpunkt der Anreicherung. Durch eine Analyse des fachlichen Geschäftsprozessmodells und den damit verbundenen Annotationen durch den IT-Experten, wird entschieden, welche Anforderungen in Iteration 1 technisch implementiert werden können. Die Tabelle 46 listet die, durch das Story Pulling, selektierten User Stories auf.

Tabelle 46: User Stories im Rahmen der Anreicherung

Story Nr.	Titel	Beschreibung
1	Modellübernahme	Das erfasste Geschäftsprozessmodell muss in das Softwarewerkzeug Axon.ivy zur technischen Anreicherung importiert werden.
2	Daten laden	Sowohl Kundendaten, als auch der Artikelstamm müssen aus bestehenden Drittsystemen geladen werden.
3	User Dialoge erstellen	Auf Basis der annotierten Grafik wird eine grafische Oberfläche technisch implementiert.
4	Gateway-Regeln implementieren	Die gebrauchssprachlich formulierten Regeln, müssen in formale Regeln umgewandelt werden. Hierzu wird das Audiofile analysiert.
5	E-Mail Versand	Zum Versand einer E-Mail an einen Kunden muss ein E-Mail-Server angebunden werden.
6	Test & FeedbackApp Integration	Zum Abschluss der technischen Implementierung wird die Ausführung des Geschäftsprozesses in Zusammenarbeit des IT-Experten und Anwenders simuliert. Des Weiteren wird die FeedbackApp Komponente in die Geschäftsprozessapplikation integriert.

Die Überführung des fachlichen Geschäftsprozessmodells in ein ausführbares Modell ist der erste Schritt während der Anreicherung. Im Rahmen des Anwendungsbeispiels wird das BPMN^{Easy}-Modell in BPMN 2.0 transferiert. Die Abbildung 91 stellt einen ersten Entwurf des Angebotsprozesses im Softwarewerkzeug Axon.ivy Designer dar.

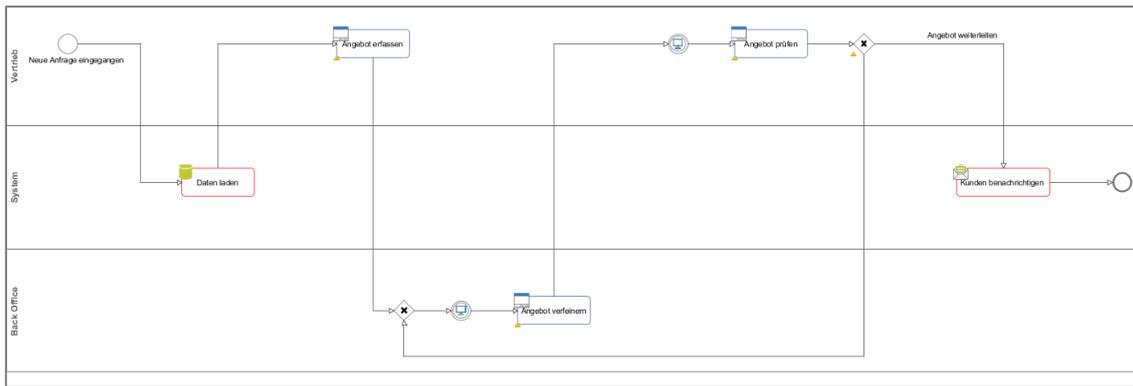


Abbildung 91: Angereichertes Geschäftsprozessmodell

Mit Hilfe des Axon.ivy Designer kann der IT-Experte die einzelnen Aktivitäten konfigurieren. Beispielsweise wird die automatisierte Aktivität „Daten lesen“ durch einen sogenannten „Database-Step“ realisiert. Der „Database-Step“ ist eine Axon.ivy spezifische Aktivität, welche die Anbindung an eine bestehende Datenbank per grafischen Editor ermöglicht. Des Weiteren wird durch den IT-Experten ein Datenmodell implementiert, auf Basis welchem die Daten einer Angebotsprozessinstanz gespeichert werden können. Hierzu werden die gebrauchssprachlichen Informationen des fachlichen Geschäftsprozessmodells analysiert. Zum Beispiel enthält das Audiofile des zweiten XOR-Gateways die Information, dass ein Angebot genehmigt oder abgelehnt werden kann. Zur formalen Definition dieser Regel wird eine Variable vom Typ Boolean angelegt.

Die Erstellung von User Dialogen und die Integration der FeedbackApp ist eine weitere Aufgabe des IT-Experten. Damit die User Dialoge unabhängig vom Endgerät aufgerufen werden können, werden diese auf Basis von HTML implementiert. Das Mediafile (vgl. Abbildung 90), welches eine Skizze einer möglichen Benutzeroberfläche darstellt, dient hierbei als Entwurfsvorlage.

Während des zweiten Quality Gates wird die Funktionalität der Geschäftsprozessapplikation überprüft. Die Beteiligten werden durch die Animationsfunktion des Axon.ivy Designers unterstützt. Wie in Abbildung 92 visualisiert, wird der Geschäftsprozess während der Testausführung animiert, sodass durch eine rote Einfärbung das jeweils aktuell ausgeführte Element markiert wird. Hierdurch können Anwender und Experten ihre Erwartungen hinsichtlich des Sequenzflusses mit der real implementierten Geschäftsprozessapplikation vergleichen.

Implementierungsfehler werden somit unmittelbar erkannt und können durch den IT-Experten beseitigt werden.

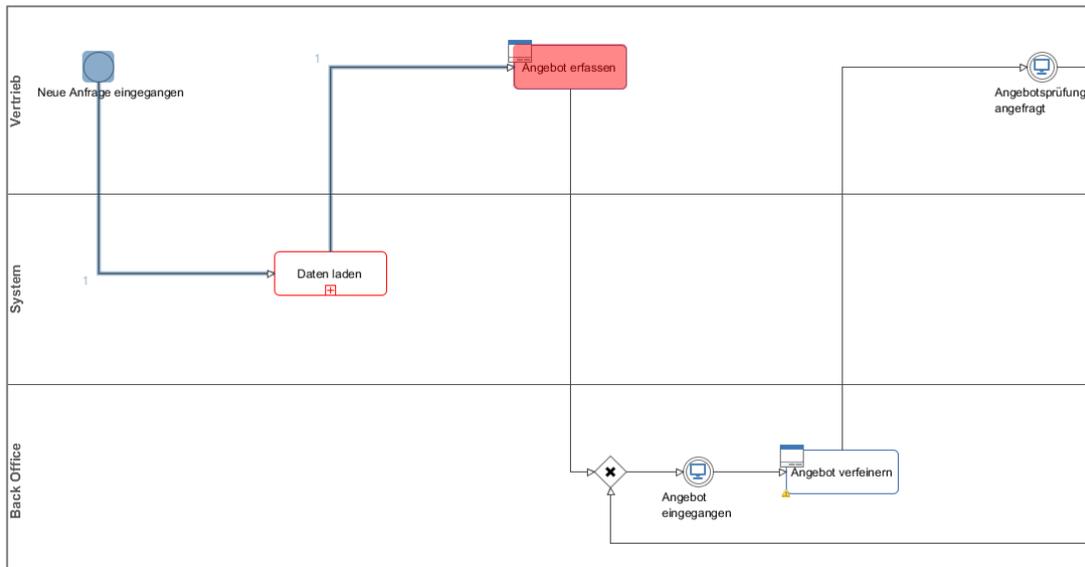


Abbildung 92: Animation des Geschäftsprozessmodell im Axon.ivy Designer

Zusätzlich zur Kontrolle der Geschäftsprozessapplikation wird ein weiteres Qualitätskriterium durch die Beteiligten definiert. Bei der Ausführung der Aktivität „Angebot prüfen“ soll Feedback erfasst werden, sodass analysiert werden kann, ob die Rückkopplung von Vertrieb und Back Office ausreichend ist. Die Tabelle 47 listet die aktualisierten Qualitätskriterien nach der Synchronisation im zweiten Quality Gate auf.

Tabelle 47: Übersicht der Qualitätskriterien

Bezeichnung	Beschreibung
CSE(bp)	Vgl. Formel (**): Messung des Anteils von trainierten und gebrauchssprachlich dokumentierten manuellen bzw. teilautomatisierten Aktivitäten.
Kommunikation	Die Kommunikationsqualität zwischen Vertrieb und Back Office ist nicht direkt messbar und wird als weiches Qualitätskriterium eingestuft. Durch die Frage „Haben Sie ausreichend Informationen für eine Entscheidung verfügbar?“ wird Feedback erfasst, welches ggfs. zur Optimierung des Geschäftsprozesses verwendet werden kann.

Die Produktivsetzung des Geschäftsprozesses wird durch den BPM-bzw.-IT-Experten durchgeführt. Zum einen werden die Geschäftsprozessmodelle für alle Beteiligte zugänglich digital verfügbar gemacht, zum anderen erhalten die Anwender Zugang zu einer Plattform, auf welcher die automatisierten Geschäftsprozesse gestartet werden können. Anwender können über eine derartige Plattform die ihnen zugewiesenen Aktivitäten öffnen und ausführen. Ebenso können, sofern die individuellen Zugriffsberechtigungen verfügbar sind, neue Geschäftsprozessinstanzen erzeugt bzw. bestehende überwacht werden.

The screenshot displays the 'Aufgabenliste' (Task List) interface. The top navigation bar includes 'Aufgabenliste', 'Prozessliste', 'Meine Workflows', 'Erledigte Aufgaben', 'Einstellungen', and 'Workflow Admin'. The main content area shows a table of tasks with columns for 'Aufgabe', 'Verantwortlich', 'Erstellt am', 'Frist', and 'Start'. Two tasks are listed: 'Angebot eingegangen' (assigned to 'Backoffice') and 'Kreditprüfung durchführen' (assigned to 'financial accounting'). Below the table, there are filters for 'Priorität' and 'Verantwortlich'. A red arrow points from the 'Aufgabenliste' to the 'Aufgaben Details' (Task Details) view, which shows a detailed view of the 'Angebot eingegangen' task. The details view includes a toolbar with actions like 'Aufnehmen', 'Zurücksetzen', 'Bemerkung erfassen', 'Delegieren', 'Fristen ändern', 'Priorität ändern', and 'Reservieren'. The task details include fields for 'TaskId', 'Name', 'Beschreibung', 'Status', 'Priorität', 'Benutzer', 'Verantwortlich', 'Verant. nach Eskalation', 'Fall', 'Erstellt am', 'Frist', 'Sperrfrist', and 'Erledigt'.

Aufgabe	Verantwortlich	Erstellt am	Frist	Start
Angebot eingegangen	Backoffice	19.07.15 10:57		
Kreditprüfung durchführen	financial accounting	19.07.15 10:53		

Powered by Axon.ivy Workflow Copyright © 2001 - 2015 AXON IVY AG

Abbildung 93: Aufgabenverwaltung im Axon.ivy Workflow Portal

Die Abbildung 93 stellt die Aufgabenliste aus Sicht eines Anwenders mit Administratorenrechten dar. Die Aufgabenliste enthält hierbei exemplarisch zwei Aufgaben aus zwei unterschiedlichen Geschäftsprozessinstanzen. Neben dem eigentlichen Annehmen und Starten der Aktivität können zusätzliche Informationen angezeigt werden. Beispielsweise werden die gebrauchssprachlich formulierten Informationen zur Aktivität dem Anwender angezeigt. Mit Hilfe der „Prozessliste“ (vgl. Abbildung 93) können neue Instanzen des Geschäftsprozesses gestartet werden.

Die Abbildung 94 stellt die erste Aktivität des Angebotsprozesses dar. Die teilautomatisierte Aktivität „Angebot erfassen“ besteht aus zwei Teilen. Zum einen aus einer „Prozesskette“ zur Übersicht in welchem Schritt sich der Anwender im Moment befindet. Die „Prozesskette“ wurde als Komponente implementiert und kann dynamisch in unterschiedliche User Dialoge eingebunden werden. Zum anderen können relevante Daten über die Eingabefelder erfasst werden. Beispielsweise kann über ein Suchfeld nach einem Kunden gesucht werden, wobei die Kundendaten aus einer externen Datenbank geladen werden.

Angebotsprozess

1 Angebot erfassen 2 Angebot verfeinern 3 Angebot prüfen 4 Angebot abschließen

Bitte die nachfolgenden Felder ausfüllen

Kunde: Maier Software GmbH

Angebotsdatum: 31.08.2015

Wj

Maier Software GmbH

WorkBench Team

Artikel/Leistung	Beschreibung
●	Konzeptionsworkshop Paket XL
●	Software-Potential-Analyse
●	Implementierung Individualsoftware

Aktivität abbrechen Aktivität abschließen

Abbildung 94: User Dialog zu Aktivität "Angebot erfassen"

Der Abschluss der Erfassung wird durch einen Klick auf „Aktivität abschließen“ markiert. Neben den individuellen Angebotsdaten speichert das zugrundliegende GPM-System Axon.ivy automatisch weitere Daten, wie beispielsweise die reale Bearbeitungszeit der Aktivität. Bei Bedarf (z.B. zur Identifizierung von Schwachstellen) können diese Daten zur Analyse des Geschäftsprozesses ausgewertet werden.

Ein weiteres Beispiel ist in Abbildung 95 dargestellt: In der Aktivität „Angebot prüfen“ wird zusätzlich die eingebundene FeedbackApp-Komponente angezeigt. Der ausführende Anwender kann somit direkt Feedback, während der Ausführung der Aktivität, erfassen.

Abbildung 95: User Dialog und FeedbackApp

Nachdem der vordefinierte Zeitraum des Ausführungs- und Analyse-Schritt vergangen ist, wird das dritte Quality Gate erreicht. Dieses Quality Gate dient zum einen zur Kontrolle der ex ante definierten Qualitätskriterien, zum anderen zur Überprüfung der BPM(N)^{Easy}-Vorgehensweise selbst durch eine Retrospektive. Das Easy Capture Sheet (vgl. Tabelle 48) spiegelt das Anwenderfeedback wider, welches während der Ausführung erfasst wurde.

Tabelle 48: Auszug aus Easy Capture Sheet

BPMN ^{Easy} Element	Mediafile	Feedback
Angebot prüfen	Keine.	„Es fehlt die Möglichkeit direkt mit dem Back Office zu kommunizieren.“
...

Zusätzlich zu dem Anwenderfeedback kann der Anteil der trainierten und gebrauchssprachlich annotierten manuellen bzw. teilautomatisierten Aktivitäten gemessen werden. Da alle bisher erfassten Aktivitäten, sowohl gemeinsam modelliert als auch getestet wurden, wird festgelegt, dass alle Aktivitäten als dokumentiert und trainiert eingestuft werden können. Das Ergebnis der Funktion CSE(Angebotsprozess) ist somit 1. Zusätzlich wird die Interaktion und Synchronisation im Team durch eine Retrospektive

bewertet. Mögliche Probleme z.B. bei der Kommunikation zwischen Teammitgliedern können hierbei, moderiert durch den BPM(N)^{Easy}-Master, gelöst werden. Unterstützend wird ein aktualisiertes Burn down Chart erstellt, welches zur Visualisierung des SOLL-IST-Zustands dient. Wie in Abbildung 96 dargestellt, konnte in Iteration 1 die Erfassung schneller durchgeführt werden, sodass insgesamt ein Zeitraum von einem Tag länger die Ausführung und Analyse des Geschäftsprozesses betrieben werden konnte. Werden über mehrere Iterationen größere Abweichungen zwischen SOLL- und IST-Zustand festgestellt, kann die Vorgehensweise flexibel angepasst werden.

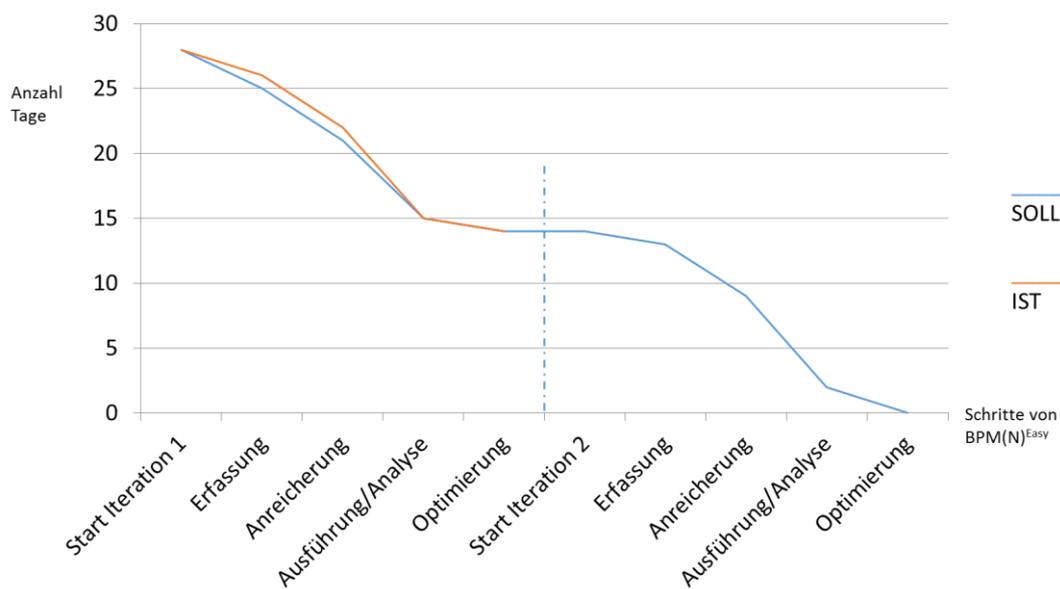


Abbildung 96: Aktualisiertes Burn down Chart

Das Ende des dritten Quality Gates markiert im Anwendungsbeispiel den unmittelbaren Beginn der zweiten Iteration. Hierbei wird im ersten Schritt das Geschäftsprozessmodell auf Basis der Optimierungsvorschläge angepasst. Innerhalb des erneuten Paarmodellierungsworkshops erfassen BPM-Experte und Anwender eine Version 2 des Geschäftsprozessmodells. Die Abbildung 97 stellt das Geschäftsprozessmodell in BPM Touch dar (vgl. zusätzlich XML-Darstellung in Anhang 1 dieser Arbeit).

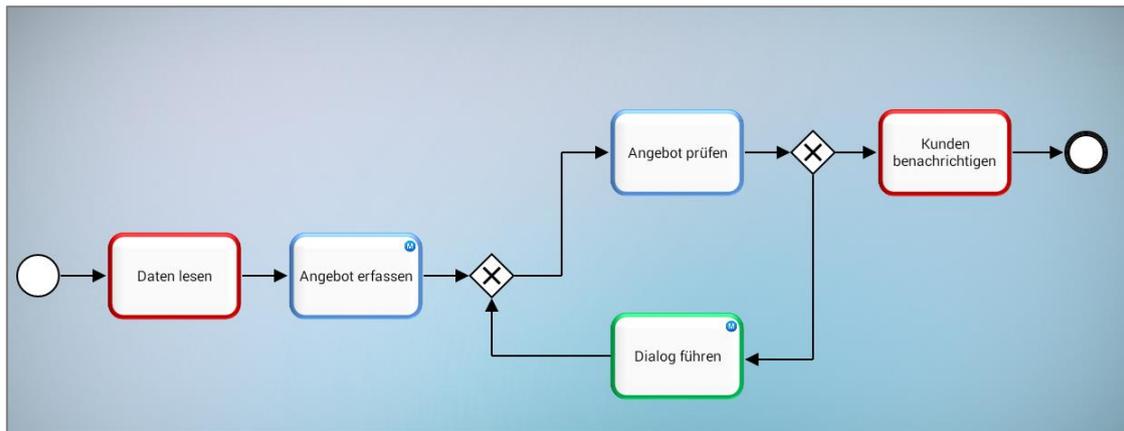


Abbildung 97: Geschäftsprozessmodell in Version 2

Auf Basis des Feedbacks wird durch die Anwender entschieden, dass die Rückkopplung von Vertrieb zu Back Office durch eine weitere manuelle Aktivität („Dialog führen“) unterstützt werden soll. Der BPM-Experte im Anwendungsbeispiel greift hierfür auf das dialogische Logik-Pattern zurück (vgl. Abschnitt 5.2). Auf Basis des Patterns können Dialoge zwischen Vertrieb und Back Office (Proponent und Opponent) strukturiert und gesteuert werden, um ein faires und transparentes Ergebnis (ob ein Angebot abgelehnt oder genehmigt wird) zu erreichen.

Nach erneutem Durchlaufen des ersten Quality Gates, kann die Geschäftsprozessapplikation aktualisiert werden. Der IT-Experte erweitert hierzu das technische Modell um ein Subprozess-Element. Zusätzlich zur fachlichen Dokumentation werden innerhalb des Subprozesses von Axon.ivy Aufgaben generiert. Die Aufgaben dienen ausschließlich als Hinweis in der Aufgabenverwaltung (vgl. Abbildung 93), dass ein Dialog zur weiteren Abstimmung bzgl. der Angebotsprüfung notwendig ist. Die Abbildung 98 stellt die Version 2 der Geschäftsprozessapplikation dar. Das dialogische Logik-Pattern kann in Form einer gebrauchssprachlichen Dokumentation direkt bei der Ausführung des Geschäftsprozesses aufgerufen werden.

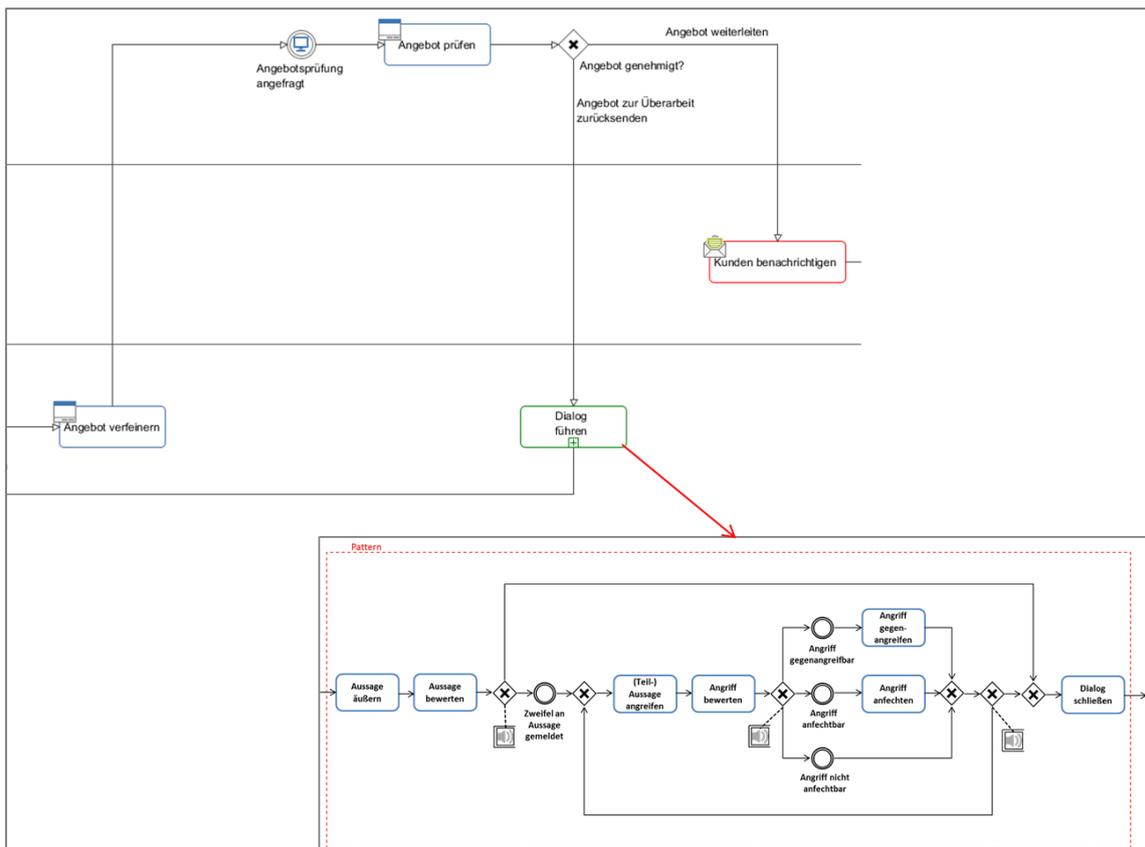


Abbildung 98: Angereichertes Geschäftsprozessmodell Version 2

Um sicherzustellen, dass im produktiven Betrieb keine Instanzen des automatisierten Geschäftsprozesses abgebrochen oder gelöscht werden, können nach dem Test (Quality Gate 2) bei Bedarf mehr als eine Version der Geschäftsprozessapplikation im Axon.ivy GPM-System betrieben werden.

Sobald von Version 1 keine Instanzen mehr ausgeführt werden, wird die Version 1 automatisch archiviert. Neue Instanzen werden dann auf Basis der Version 2 ausgeführt. Eine ausführliche Dokumentation der Geschäftsprozessapplikation findet sich im Anhang dieser Arbeit.

6.3. Evaluation

Die Evaluation der BPM(N)^{Easy}-Methode stützt sich auf drei Säulen. Zum einen wurde in Abschnitt 4 beschrieben wie die identifizierten Anforderungen (vgl. Abschnitt 1) durch BPM(N)^{Easy} gelöst werden können. Zum anderen dienen mehrere eigens wissenschaftliche Publikationen als Grundlage und iterative Evaluation von BPM(N)^{Easy} (vgl. insbesondere [MSW12], [MW13], [MOW14], [MOW15]). Zusätzlich verdeutlicht die Evaluation des Beispiels die erfolgreiche Anwendbarkeit von BPM(N)^{Easy}. In der Abbildung 99 sind die Anforderungen des Anwendungsbeispiels der allgemeinen Zielsetzung gegenübergestellt.

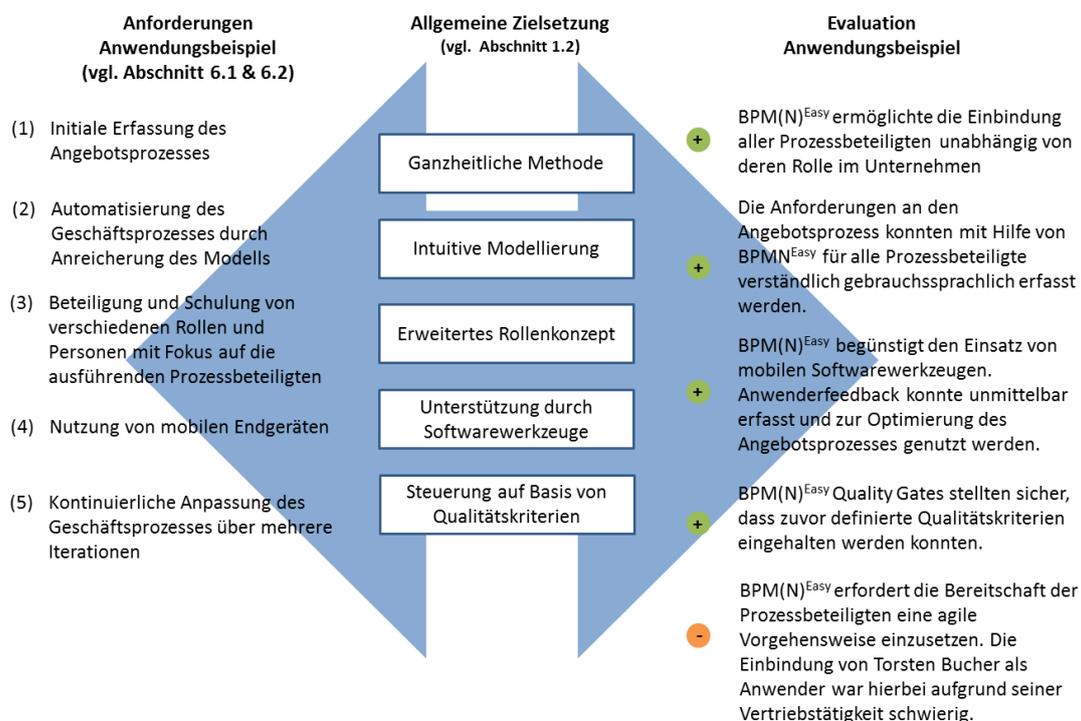


Abbildung 99: Evaluation des Anwendungsbeispiels

Es konnte festgestellt werden (vgl. rechte Seite Abbildung 99), dass BPM(N)^{Easy} die Anforderungen weitgehend erfüllt. Die drei Bestandteile der Methode - Vorgehensweise, Sprache und Softwarewerkzeug – ergänzen sich hierbei gegenseitig, sodass eine ganzheitliche Unterstützung bei der Ausführung des GPM-Projekts möglich ist. Insbesondere die in kurzen Intervallen iterative Überarbeitung der erfassten Inhalte ermöglichte die Erarbeitung eines Geschäftsprozessmodells und einer Prozessapplikation, welchen den aktuellen Anforderungen aller Prozessbeteiligten entsprechen.

Als kritischen Aspekt konnte festgestellt werden, dass BPM(N)^{Easy} eine enge und direkte Zusammenarbeit erfordert, sodass die regelmäßigen Abwesenheiten von Prozessbeteiligten den Ablauf der Vorgehensweise erschweren.

Neben dem beschriebenen Anwendungsbeispiel spiegeln der Einsatz und die daraus resultierenden Erfahrungen aus der realen betrieblichen Anwendung Gleiches wider. Als einen Erfolgsfaktor für die agile Vorgehensweise konnte die bindende Rolle des BPM(N)^{Easy}-Masters (vgl. Abschnitt 4.2.2) identifiziert werden. Die Projektbeteiligten können durch den BPM(N)^{Easy}-Master bei der Interaktion unterstützt und beraten werden. Zusätzlich konnte in Projekten beobachtet werden, dass die ständige Integration von Prozessbeteiligten die Akzeptanz gegenüber der GPM-Projekte erhöht. Hierbei zeichnete sich ein Iterationszyklus von zwei bis drei Wochen als empfehlenswert aus. Zudem sind, durch Verknüpfung gebrauchssprachlicher Informationen (z.B. Bilder, Mockups) an Geschäftsprozessmodelle, bereits nach der ersten Iteration erste Prozessapplikationen generierbar, wobei BPMN^{Easy}-Modelle ohne Probleme in andere Softwarewerkzeuge, welche BPMN 2.0 unterstützen, importiert und weiterbearbeitet werden können.

7. Zusammenfassung und Ausblick

Heutige und zukünftige Herausforderungen bringen die Ansätze des traditionellen Geschäftsprozessmanagements immer stärker an deren Leistungsgrenzen. Beispielsweise ermöglicht die zunehmende Digitalisierung immer schnellere Veränderungen von Geschäftsprozessen auf Basis von flexiblen ortsunabhängigen Interaktionsbeziehungen zwischen allen Ausführenden der Geschäftsprozessaktivitäten. Insbesondere die effektive und effiziente Einbindung aller Beteiligten (Anwender, IT- und BPM-Experten) unterliegt hierbei einer stetig wachsenden Bedeutung und fordert methodische Unterstützung über den gesamten Geschäftsprozessmanagementzyklus (Erfassung – Anreicherung – Ausführung/Analyse – Optimierung). Die in dieser Arbeit beschriebene Methode bietet einen Ansatz zum agilen Geschäftsprozessmanagement auf Basis gebrauchssprachlicher Modellierung an. Durch die Verwendung der Gebrauchssprache wird die Verständlichkeit von Geschäftsprozessmodellen von Anfang an gewährleistet. Des Weiteren dient die Agilität als Grundlage einer flexiblen, iterativen, inkrementellen und empirischen Vorgehensweise. Mit Hilfe der entwickelten Modellierungssprache und des implementierten Software-Prototyps konnte die Vorgehensweise bereits angewendet und erprobt werden.

In Abschnitt 7.1 werden die vorangegangenen Kapitel zusammengefasst. Danach werden in Abschnitt 7.2 die Ergebnisse der Arbeit kritisch diskutiert. Abschließend wird ein Ausblick auf weiterführende Forschungsfragen und Aktivitäten gegeben.

7.1. Zusammenfassung

Basierend auf der formulierten Zielsetzung wurden zunächst die für diese Arbeit relevanten Grundlagen eingeführt. Es wurde diskutiert, wie im Rahmen der sprachbasierten Informatik systematisch eine fehlerfreie Wissensrekonstruktion und Wissensbegründung möglich ist. Nachfolgend wurde beschrieben, dass sich die Anwendungssystementwicklung nicht nur mit den technischen Systemen, sondern auch mit deren Kontext befasst, sodass auch Geschäftsprozesse mitbetrachtet werden. Insbesondere wurde die zentrale Rolle von Schema, Ausprägung und Begriffe in Bezug auf die Kommunikation unterschiedlicher Teilnehmer dargestellt. Es wurde gezeigt, dass die Teilnehmer (z.B. Teilnehmer eines GPM-Projekts) über standardisierte Schemata verfügen müssen, um die Ausprägung des anderen verstehen zu können.

Ebenso wurden als Basis (z.B. einer fairen und transparenten Kommunikation während der Ausführung von Geschäftsprozessaktivitäten) die Konzepte der dialogischen Logik eingeführt. Die dialogische Logik ermöglicht einen Dialog bei dem Argumente der Dialogpartner widerlegt oder bestätigt werden.

Zusätzlich wurden existierende Ansätze zum Geschäftsprozessmanagement vorgestellt, wobei sowohl traditionelle, als auch agile Methoden berücksichtigt wurden. Neben der Vorstellung wurden die Ansätze zu der in dieser Arbeit vorgestellten Methode abgegrenzt. Zusätzlich wurde zur besseren Einordnung, der in dieser Arbeit implementierten und verwendeten Software, das in der Literatur vorherrschende Verständnis in Bezug auf Geschäftsprozessmanagementsysteme dargestellt. Es wurde gezeigt, dass der Einsatz von Geschäftsprozessmanagementsystemen über die reine Softwareunterstützung der Modellierung und Automatisierung hinausgeht. Indes wird intelligente und mobilfähige Software verstärkt im Kontext des GPMs eingesetzt. Danach wurden verschiedene Ansätze der agilen Softwareentwicklung vorgestellt, unter der Annahme, dass das Geschäftsprozessmanagement vergleichbare Ziele verfolgt wie die Softwareentwicklung. Gemeinsame Ziele des Geschäftsprozessmanagements und der Softwareentwicklung sind zum Beispiel die Reduzierung der Kosten während der Umsetzung (Implementierung bzw. Modellierung) und die kontinuierliche Einbindung aller Beteiligten über den kompletten Lebenszyklus (der Software bzw. des Geschäftsprozesses) hinweg. Hierbei wurde dargestellt, wie die einzelnen Ansätze Aspekte wie beispielsweise Teamstruktur und Regeln zur Durchführung agiler Projekte adressieren.

Den Hauptteil der vorliegenden Arbeit bilden die Beschreibung der gebrauchssprachlichen Modellierung und die Konzeption der BPM(N)^{Easy}-Methode. Die eingeführte gebrauchssprachliche Modellierung bietet die Grundlage zur signifikanten Verbesserung der Kommunikation, Interaktion sowie des gegenseitigen Verständnisses aller in einem GPM-Projekt Beteiligten. Hierzu wurden die für das Geschäftsprozessmanagement relevanten Sprachen (Gebrauchs-, Modellierungs- und Programmiersprachen) diskutiert. Es wurde beschrieben, wie diese beispielsweise zur Beschreibung von Anforderungen im Rahmen der Anwendungssystementwicklung dienen. Des Weiteren wurde der Zusammenhang zwischen den einzelnen Sprachen aufgezeigt und die Probleme der Sprachübergänge (z.B. Transformation von Gebrauchssprache zu Modellierungssprache) diskutiert.

Auf Basis der Sprechakttheorie wurde dargestellt, dass ausschließlich gelungene Sprechakte eine erfolgreiche Modellierung von Geschäftsprozessen ermöglichen. Zusätzlich wurden nach einer entsprechenden Anforderungsanalyse, als grundlegende Anker der gebrauchssprachlichen Modellierung und des agilen Geschäftsprozessmanagements, die Aspekte Interaktion, Synchronisation und Qualität definiert. Der Begriff der gebrauchssprachlichen Modellierung definiert hierbei, dass während einer Interaktion die gelungenen Sprechakte für alle Prozessbeteiligten, unabhängig deren Vorwissens, unter Berücksichtigung ex ante definierter Qualitätskriterien, dynamisch in Modelle erfasst werden. Aufbauend auf diesen Ankern wurde eine Methode zum agilen Geschäftsprozessmanagement entwickelt. Die Methode besteht aus einer Vorgehensweise, Werkzeug und Sprache. Die Vorgehensweise BPM(N)^{Easy} basiert auf ausgewählten und weiterentwickelten Aspekten existierender Vorgehensweisen des Geschäftsprozessmanagements und der agilen Softwareentwicklung. Die kontinuierliche Integration des Menschen und die Fokussierung auf Problemstellungen der Praxis standen bei der Entwicklung im Vordergrund. Die Vorgehensweise ist iterativ, inkrementell und evolutionär.

Zusätzlich wurde mit BPMN^{Easy} eine grafische Modellierungssprache beschrieben, die über die reine grafische Modellierung das Annotieren von Multimediainhalten zulässt. Es wurde das Ziel verfolgt eine Sprache zu definieren, welche es allen Beteiligten (z.B. Anwender, Modellierer, IT-Experten) von Anfang (insbesondere für die Ersterfassung/Erstbetrachtung von Geschäftsprozessmodellen) ermöglicht intuitiv und gebrauchssprachlich Geschäftsprozessmodelle zu erfassen. Neben Vorgehensweise und Sprache wurden existierende Softwarewerkzeuge vorgestellt. Es wurde dargestellt wie auf Basis der Axon.ivy BPM Suite und dem BPM Touch das Geschäftsprozessmanagement softwaretechnisch unterstützt werden kann. Nachfolgend wurde ein Software-Prototyp entwickelt. Der entwickelte Prototyp kann zur (mobilen) Erfassung von Anwenderfeedback während der Ausführung von Aktivitäten eines Geschäftsprozesses verwendet werden. Durch die Komponentenarchitektur kann der FeedbackApp-Prototyp einfach lose-gekoppelt in bestehende oder neue Geschäftsprozessapplikationen eingebunden werden. Hierbei wird das Anwenderfeedback auf Basis von ex ante definierten Qualitätskriterien erfasst und gespeichert. Somit wird eine bisherige Lücke des traditionellen Monitorings durch anwenderzentriertes Echtzeitfeedback geschlossen. Des Weiteren wurde ein

Geschäftsprozesspattern entwickelt, welches zur Modellierung von Interaktionen innerhalb von Geschäftsprozessen eingesetzt werden kann. Basis für das Pattern bildet die dialogische Logik und deren Rahmen- bzw. Partikelregeln. Tritt beispielsweise bei der Ausführung eines Geschäftsprozesses eine Ausnahme auf (z.B. vordefinierte Regeln genügen nicht den Anforderungen der aktuellen Instanz), kann das dialogische Logik-Pattern zur strukturierten Aushandlung dienen. Die beteiligten Ausführenden führen hierdurch einen transparenten Dialog über die Behandlung der aufgetretenen Ausnahme. Zur Erprobung der Methode wurde ein Anwendungsbeispiel beschrieben. Auf der Basis von in der Praxis durchgeführten Projekten, konnte ein durchgängiges Szenario dargestellt werden. Die Erfassung, Anreicherung, Ausführung, Analyse und Optimierung eines Angebotsprozesses wurde hierbei beschrieben und diskutiert. Durch die Anwendung aller Bestandteile der Methode – der Vorgehensweise, der gebrauchssprachlichen Modellierungssprache, des Geschäftsprozesspatterns und des Software-Prototypen – konnte die grundsätzliche Anwendbarkeit der Methode evaluiert werden.

7.2. Kritische Betrachtung der Ergebnisse und Ausblick

BPM(N)^{Easy} leistet einen Beitrag zum Fortschritt des agilen Geschäftsprozessmanagements. Als Methode, welche aus Vorgehensweise, Sprache und Werkzeug besteht, ermöglicht BPM(N)^{Easy} einen vielfältigen und einfachen Einsatz. Unabhängig von den existierenden Rahmenbedingungen kann BPM(N)^{Easy} gestartet und zur Erfassung, Anreicherung, Ausführung, Analyse und Optimierung von Geschäftsprozessen eingesetzt werden. Die Vorgehensweise basiert auf einem konstruktiven Ansatz, wobei die einzelnen Schritte fortlaufend, schrittweise und zirkelfrei durchlaufen werden. Zusätzlich wurde in der Arbeit ein Rollenkonzept erarbeitet. Die definierten Rollen werden bei der Anwendung von BPM(N)^{Easy} den einzelnen Beteiligten zugeordnet. Beispielsweise nimmt ein Mitarbeiter der IT-Abteilung die Rolle des IT-Experten und der GPM-Beauftragte der Organisation die Rolle des BPM-Experten ein. Mit BPM(N)^{Easy} wird somit gesamtheitlich vorgegeben „wer“, „wann“, „welche“ Aufgaben ausführen muss und verantwortet. Als weiterer Bestandteil der Methode beschreibt die Modellierungssprache BPMN^{Easy} eine grafische Modellierungssprache, welche durch multimediale Inhalte annotiert werden kann. BPMN^{Easy} adaptiert hierbei den weitverbreiteten Modellierungsstandard BPMN 2.0, wobei ausschließlich die grafischen Basiselemente zur Modellierung zugelassen werden. Dieser Ansatz trägt zu einem verbesserten Verständnis aller Beteiligten bei, indem die künstliche Modellierungssprache die gebrauchssprachliche Modellierung unterstützt. Das bedeutet, dass eine Rekonstruktion der Gebrauchssprache auf Basis der BPMN^{Easy}-Geschäftsprozessmodelle möglich ist. Die gebrauchssprachlich annotierten Mediafiles werden mit Hilfe eines Datenobjekts gespeichert und können als Dokumentation oder Grundlage der Anreicherung von Geschäftsprozessmodellen zu Geschäftsprozessapplikationen verwendet werden. Die Verwendung eines etablierten Modellierungsstandards gewährleistet die kompatible Nutzung mit existierenden Softwarewerkzeugen. Zusätzlich, zu den in der Arbeit beschriebenen Softwarewerkzeugen, wurde ein Software-Prototyp implementiert. Dieser Software-Prototyp (FeedbackApp) dient zur Erfassung von Anwenderfeedback während der Ausführung von Geschäftsprozessaktivitäten. Den ausführenden Anwendern wird hierbei ein grafischer Benutzerdialog zur Abfrage eines Qualitätskriteriums angezeigt.

Um eine möglichst hohe Wiederverwendbarkeit des Software-Prototyps zu gewährleisten, wurde die FeedbackApp als lose-gekoppelte Softwarekomponente entwickelt.

Sowohl Neuimplementierungen, als auch bestehende Geschäftsprozessapplikationen auf Basis der Axon.ivy BPM Suite können somit flexibel um die FeedbackApp-Komponente erweitert werden.

Das Ziel der Arbeit war es, den Herausforderungen des digitalen Zeitalters im Kontext des Geschäftsprozessmanagements auf Basis von Agilität und gebrauchssprachlicher Modellierung zu entgegnen. Ein signifikanter Unterschied zu existierenden Ansätzen liegt darin, dass insbesondere die konsequente Einbindung des Menschen in den Vordergrund gestellt wird. Hierbei stehen die Interaktion und Synchronisation innerhalb des Geschäftsprozessmanagements im Fokus, wodurch insbesondere das gegenseitige Verständnis aller Beteiligten in Bezug auf wissens- und entscheidungsintensive Geschäftsprozesse (welche über die Möglichkeiten der Digitalisierung) hinausgehen, erreicht. Die dadurch erreichbare Geschäftsprozessoptimierung trägt zur nachhaltigen Wettbewerbsfähigkeit von Organisationen bei. Dennoch besitzen agile Methoden wie BPM(N)^{Easy} auch Grenzen der erfolgreichen Anwendbarkeit.

Eine Einschränkung der Anwendbarkeit ergibt sich, wenn die Fähigkeit und Bereitschaft agile Methoden anzuwenden durch die Beteiligten (z.B. Anwender, BPM-Experten) einer Organisation nicht vorhanden sind. Fehlt zudem die Top-Down-Unterstützung bei der Einführung solcher (neuen) Methoden, kann dies zu Misstrauen und Ablehnung führen. Des Weiteren erfordert der Ansatz eine enge und vertrauensvolle Zusammenarbeit aller Beteiligten. Die regelmäßigen Synchronisationstreffen und Interaktion können indes über internetbasierte Technologien durchgeführt werden (z.B. Videokonferenzen, Online-Chat). Dennoch ist eine „Face-to-Face“-Kommunikation – insbesondere zum Start und Ende der Iterationen – ein bedeutender Aspekt des agilen Vorgehens und nur begrenzt durch virtuelle Abstimmungen zu kompensieren. Organisationen, welche über örtlich verteilte Standorte und Verantwortlichkeiten verfügen, können hierdurch auf zusätzliche Herausforderungen stoßen. Eine weitere Herausforderung ergibt sich bei GPM-Projekten, in welche eine hohe Anzahl von beteiligten Personen involviert ist. Durch eine mehrfache Vergabe der Rollen (z.B. mehrere IT-Experten, mehrere BPM-Experten) müssen diese ggfs. aufwendig synchronisiert werden. Dadurch ist es notwendig eine zusätzliche

Einbindung der agilen Vorgehensweise in eine übergreifende Organisationseinheit durchzuführen.

Diese übergreifende Organisationseinheit übernimmt zum Beispiel die Gesamtplanung des GPM-Projekts und erfüllt übergreifende Funktionen (z.B. Überwachung des Budgets), um den BPM(N)^{Easy}-Master bei der Koordination zu unterstützen. Des Weiteren kann die Rekonstruktion der Gebrauchssprache aus einem erfassten Geschäftsprozessmodell durch eine hohe Anzahl von Beteiligten erschwert werden. Das gemeinsame Verständnis, beispielsweise von verwendeten Fachbegriffen, wird innerhalb des agilen Vorgehens iterativ und empirisch erhöht. Große Teams erschweren hierbei die Diskussion und gegenseitige Abstimmung. Hierdurch kann ein zusätzliches Risiko in Form von schwer auflösbaren Missverständnissen (z.B. Synonymen, Homonymen,...) bei der Rekonstruktion entstehen. Das in dieser Arbeit entwickelte Geschäftsprozesspattern kann zur dialogischen Logik-basierten Dialogführung eingesetzt werden. Hierdurch wird sichergestellt, dass transparente und faire Ergebnisse eines Dialogs erzielt werden. Hierbei beschränkt die dialogische Logik die Dauer eines Dialogs nicht ausreichend. Insbesondere, falls eine Elementaraussage im Moment der Ausführung nicht direkt entscheidbar ist und somit der Status „non liquet“ eintritt, können Dialoge reale Geschäftsprozessinstanzen verzögern.

Dem Methodischen Konstruktivismus folgend, basiert die Methode und deren Lösungsansätze auf bekannten und wahrgenommenen Herausforderungen („aus der Praxis, für die Praxis“). Eine stetige Überprüfung und Weiterentwicklung der Methode ist somit notwendig, um gleichermaßen auf zukünftige Probleme anwendbar zu sein.

Die auf diese Arbeit aufbauenden Aufgaben resultieren zum einen aus der Weiterentwicklung des Software-Prototyps, zum anderen aus der weiteren Anwendung der dialogischen Logik.

Der Software-Prototyp soll um zusätzliche Funktionen ergänzt werden. Neben der textuellen Erfassung von Feedback soll auch die multimediale Aufnahme und Speicherung (z.B. Video, Audio) möglich sein. Ein weiteres Thema stellt die Konzeption und Entwicklung von zusätzlichen Datenbanken über eine zentrale Administrationsoberfläche dar. Hierbei soll die Administration von unterschiedlichen Qualitätsmetriken und Qualitätskriterien ermöglicht werden.

Zusätzlich soll untersucht werden, wie Anwender und deren User Experience durch die Anwendung derartiger Software beeinflusst werden. In diesem Zusammenhang soll zusätzlich identifiziert werden, zu welchen Zeitpunkten die Erfassung am effektivsten und effizientesten ist. Eine Portierung auf andere GPM-Systeme, der in Axon.ivy implementierten FeedbackApp, ist eine zusätzliche Aufgabe.

Darüber hinaus sollen die Ansätze der dialogischen Logik nicht ausschließlich innerhalb von Geschäftsprozessmodellen, sondern über das in der Arbeit vorgestellte Geschäftsprozess-Pattern hinausgehend genutzt werden. Aufbauend auf die vorliegende Arbeit soll untersucht werden, in wie weit, auf Basis der dialogischen Logik, Dialoge zwischen den GPM-Projektbeteiligten angewendet werden. Zum Beispiel kann die dialogische Logik als Grundlage der Aushandlung von Optimierungsvorschlägen genutzt werden. Zusätzlich soll zur Sicherung und Erweiterung der erreichten Ergebnisse, die Methode über den Kontext des Geschäftsprozessmanagements angewendet werden. Hierzu sollen gemeinsame Herausforderungen verwandter Disziplinen identifiziert und beschrieben werden. Beispielsweise kann die Methode zur Erfassung von IT-Architekturmodellen im Rahmen des IT-Architekturmanagements eingesetzt werden. Des Weiteren ist die Anwendung im Rahmen der Erfassung und Implementierung von Administrations- und Interaktionsprozessen im Kontext von Smart Cities möglich. In Planung befindliche Projekte dienen hierbei zur Erprobung und zur Grundlage der Weiterentwicklung.

LITERATURVERZEICHNIS

- [Aa03a] Aalst van der, W. M.: Patterns and XPD L: A Critical Evaluation of the XML Process Definition Language.
<http://www.workflowpatterns.com/documentation/documents/ce-xpdl.pdf>, 27.07.2015.
- [Aa03b] Aalst van der, W. et al.: Workflow Patterns. In Distributed and Parallel Databases, 2003, 14; S. 5–51.
- [Aa13] van der Aalst, Wil M. P.: Business Process Management: A Comprehensive Survey. In ISRN Software Engineering, 2013, 2013; S. 1–37.
- [Aa14] Aalst van der, W. M. P.: Business process management as the “Killer App” for Petri nets. In Software & Systems Modeling, 2014.
- [AB09] ABPMP: GUIDE TO THE BUSINESS PROCESS MANAGEMENT COMMON BODY OF KNOWLEDGE (BPM CBOK). Version 2.0 - Second Release, 2009.
- [Ad14] Adam, S. et al.: Business Process Management. Marktanalyse 2014. ; BPM Suites im Test. Fraunhofer Verlag, s.l., 2014.
- [Ag04] Aguilar-Savén, R. S.: Business process modelling: Review and framework. In International Journal of Production Economics, 2004, 90; S. 129–149.
- [Ag07] Agrawal, A.; Amend, M.; Das, M.; Ford, M.; Keller, C.; Kloppmann, M.; König, D.; Leymann, F.; Müller, R.; Pfau, G.; Plösser, K.; Rangaswamy, R.; Rickayzen, A.; Rowley, M.; Schmidt, P.; Trickovic, I.; Yiu, A.; Zeller, M.: WS-BPEL Extension for People (BPEL4People), Version 1.0.
http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-bpel4people/BPEL4People_v1.pdf, 27.07.2015.
- [AGT13] Aversano, L.; Grasso, C.; Tortorella, M.: A Literature Review of Business/IT Alignment Strategies. In (Cordeiro, J.; Maciaszek, L. A.; Filipe, J. Hrsg.): Enterprise Information Systems. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013; S. 471–488.

-
- [AIH11] Anvari, A.; Ismail, Y.; Hojjati, Seyed Mohammad Hossein: A study on total quality management and lean manufacturing: through lean thinking approach. In *World applied sciences journal*, 2011, 12; S. 1585–1596.
- [AIS77] Alexander, C.; Ishikawa, S.; Silverstein, M.: *A pattern language. Towns, buildings, construction*. Oxford University Press, New York, 1977.
- [AK15] Aagesen, G.; Krogstie, J.: *BPMN 2.0 for Modeling Business Processes*. In (Vom Brocke, J.; Rosemann, M. Hrsg.): *Handbook on Business Process Management 1*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015; S. 219–250.
- [AI05] Allweyer, T.: *Geschäftsprozessmanagement. Strategie, Entwurf, Implementierung, Controlling*. W3L-Verl., Herdecke, Bochum, 2005.
- [AI11] Allweyer, T.: *BPM-Round-Trip: Wunsch oder Wirklichkeit?* In (Komus, A. Hrsg.): *BPM Best Practice*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011; S. 219–234.
- [AI13] Allweyer, T.: *BPMN 2.0 - Business Process Model and Notation. Einführung in den Standard für die Geschäftsprozessmodellierung*. Books on Demand, Norderstedt, 2013.
- [AI14a] Allweyer, T.: *BPMS. Einführung in Business Process Management-Systeme*. Books on Demand, Norderstedt, 2014.
- [AI14b] Alpár, P.: *Anwendungsorientierte Wirtschaftsinformatik. Strategische Planung, Entwicklung und Nutzung von Informationssystemen*. Springer Vieweg, Wiesbaden, 2014.
- [Am02] Ambler, S. W.: *Agile modeling. Effective practices for eXtreme programming and the unified process*. J. Wiley, New York, 2002.
- [AM09] Abts, D.; Mülder, W.: *Grundkurs Wirtschaftsinformatik. Eine kompakte und praxisorientierte Einführung*. Vieweg + Teubner, Wiesbaden, 2009.
- [AM11] Antunes, P.; Mourão, H.: *Resilient Business Process Management: Framework and services*. In *Expert Systems with Applications*, 2011, 38; S. 1241–1254.

-
- [An10] Anderson, D. J.: Kanban. Successful evolutionary change for your technology business. Blue Hole Press, Sequim, Washington, 2010.
- [An14] Anderson, D.: THE PRINCIPLES & GENERAL PRACTICES OF THE KANBAN METHOD. <http://www.djaa.com/principles-general-practices-kanban-method>, 27.07.2015.
- [Ar05] Arielli, E.: Unkooperative Kommunikation. Eine handlungstheoretische Untersuchung. Lit, Münster, 2005.
- [ATW03] Aalst, Wil van der; Ter Hofstede, A.; Weske, M.: Business process management: A Survey. International conference, BPM 2003, Eindhoven, the Netherlands, June 26-27, 2003 proceedings. Springer, Berlin, New York, 2003.
- [Au05] Augustine, S.: Managing agile projects. Prentice Hall Professional Technical Reference, Upper Saddle River, NJ, 2005.
- [Au62] Austin, J. L.: How to do things with words. Clarendon Press, Oxford, 1962.
- [AX15a] AXON IVY AG: Axon.ivy BPM Suite. <http://www.axonivy.com/de-de/>, 27.07.2015.
- [AX15b] AXON IVY AG: Axon.ivy Engine. <http://developer.axonivy.com/doc/latest/EngineGuideHtml/introduction.html#introduction.overview>, 27.07.2015.
- [BA04] Beck, K.; Andres, C.: Extreme programming explained. Embrace change. Addison-Wesley, Boston, MA, 2004.
- [Ba07] Bandara, W. et al.: Major Issues in Business Process Management: An Expert Perspective: ECIS 2007 Proceedings. Paper 89., 2007.
- [Ba10a] Basili, V. R. et al.: Linking Software Development and Business Strategy Through Measurement. In Computer, 2010, 43; S. 57–65.
- [Ba10b] Bartsch, C.: Modellierung und Simulation von IT-Dienstleistungsprozessen. KIT Scientific Publishing, 2010.

-
- [Ba11] Balzert, S. et al.: Heft 193 Oktober 2011 Vorgehensmodelle im Geschäftsprozessmanagement Operationalisierbarkeit von Methoden zur Prozesserhebung. In Veröffentlichungen des Instituts für Wirtschaftsinformatik, DFKI im Deutschen Forschungszentrum für Künstliche Intelligenz, 2011.
- [Ba13] Bansal, A.: Introduction to Programming Languages. Taylor & Francis, 2013.
- [Ba14a] Bauernhansl, T.: Industrie 4.0 in Produktion, Automatisierung und Logistik. Anwendung, Technologien und Migration. Springer Vieweg, Wiesbaden, 2014.
- [Ba14b] Baumann, M. et al.: Enhancing Feasibility of Human-driven Processes by Transforming Process Models to Process Checklists, in 15th IFIP WG8.1 Working Conference on Business Process Modeling, Development and Support (BPMDS), 2014.
- [ba15] bamero: The first mile in BPM. <http://bamero.de/de/>, 27.07.2015.
- [Ba99] Bauske, J.: Ein objektorientiertes Verfahren zur Optimierung von Geschäftsprozessen unter Verwendung eines genetischen Algorithmus. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999.
- [Be00] Beck, K.: Extreme programming eXplained. Embrace change. Addison-Wesley, Reading, MA, 2000.
- [Be01] Beck, K.: Manifesto for Agile Software Development. <http://agilemanifesto.org/>, 27.07.2015.
- [Be06] Beyer, G.: Innovationskultur und Ideenmanagement. Strategien und praktische Ansätze für mehr Wachstum ; [jetzt mit eBook, Fachinformation per Mausklick]. Symposium, Düsseldorf, 2006.
- [Be07] Bertsche, B. et al.: Entwicklung und Erprobung innovativer Produkte - Rapid Prototyping. Grundlagen, Rahmenbedingungen und Realisierung. Springer, Berlin, 2007.
- [Be09] Becker, J. et al.: Geschäftsprozessmanagement. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

-
- [Be11a] Beck, H.: Medienökonomie. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [Be11b] Beckermann, A.: Einführung in die Logik. De Gruyter, Berlin, New York, 2011.
- [Be12] BearingPoint GmbH: Business Process Management-Studie 2012. Stärkung der Prozessorientierung im Unternehmen durch nachhaltige Optimierung der Prozess- und IT-Landschaft, 2012.
- [Be13] Bergener, K. et al.: On the importance of agile communication skills in BPM education: Design principles for international seminars. In Knowledge Management & E-Learning: An International Journal (KM&EL), 2013, 4; S. 415–434.
- [Be14] Bertram, J. et al.: Lernen und Wissensaustausch in Organisationen: Individuelle und kollektive Ansätze. In (Cress, U.; Hesse, F. W.; Sassenberg, K. Hrsg.): Wissenskollektion. Springer Fachmedien Wiesbaden, Wiesbaden, 2014; S. 1–53.
- [Be15] BearingPoint GmbH: Business Process Management-Studie 2015 -Messbare Verbesserung der Leistungsfähigkeit durch Prozessmanagement. <http://www.bearingpoint.com/de-de/7-11740/business-process-management-studie-2015>, 27.07.2015.
- [BE15] Blumenschein, A.; Ehlers, I. U.: Ideen managen. Eine verlässliche Navigation im Kreativprozess. Gabler, Wiesbaden, 2015.
- [Be98] Berkau, C.: Instrumente der Datenverarbeitung für das effiziente Prozesscontrolling. In Kostenrechnungspraxis, 1998; S. 27–32.
- [Be99] Beck, K.: Kent Beck's guide to better Smalltalk. Cambridge University Press, Cambridge, U.K., New York, 1999.
- [BF14] Böger, E.; Fleischmann, A.: Closing the Gap between Business Process Models and their Implementation:Towards Certified BPMs, 2014.
- [BFV12] Brambilla, M.; Fraternali, P.; Vaca Ruiz, C. K.: Combining social web and BPM for improving enterprise performances. In (Mille, A. et al. Hrsg.): the 21st international conference companion, 2012; S. 223.

-
- [BHV14] Bauernhansl, T.; Hompel, M. ten; Vogel-Heuser, B. Hrsg.: Industrie 4.0 in Produktion, Automatisierung und Logistik. Springer Fachmedien Wiesbaden, Wiesbaden, 2014.
- [Bi04] Biethahn, J.: Ganzheitliches Informationsmanagement. Oldenbourg, München, Wien, 2004.
- [Bi14] Bittmann, S. et al.: Additional Information in Business Processes: A Pattern-Based Integration of Natural Language Artefacts. In (Fill, H.-G. et al. Hrsg.): GI-Edition Proceedings Band 225 - Modellierung 2014 -. 19.-21. März 2014 in Wien. Köllen, Bonn, 2014.
- [BJA15] Boughzala, I.; Janssen, M.; Assar, S.: E-Government 2.0: Back to Reality, a 2.0 Application to Vet. In (Boughzala, I.; Janssen, M.; Assar, S. Hrsg.): Case Studies in e-Government 2.0. Springer International Publishing, Cham, 2015; S. 1–14.
- [BK13a] Broy, M.; Kuhrmann, M.: Projektorganisation und Management im Software Engineering. Imprint: Springer Vieweg, Berlin, Heidelberg, 2013.
- [BK13b] Bayer, F.; Kühn, H.: Prozessmanagement für Experten. Impulse für aktuelle und wiederkehrende Themen. Imprint: Springer Gabler, Berlin, Heidelberg, 2013.
- [BKO10] Birkmeier, D. Q.; Kloeckner, S.; Overhage, S.: An empirical comparison of the usability of BPMN and UML activity diagrams for business users, 2010.
- [BI05] Blechschmitt, E.: Adaptive Mensch-Maschine Interaktion für mobile Agenten, 2005.
- [BM03] Becker, J.; Meise, V.: Strategie und Ordnungsrahmen. In (Becker, J.; Kugeler, M.; Rosemann, M. Hrsg.): Prozessmanagement. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003; S. 107–157.
- [BM04] Blasche, S.; Mittelstrass, J.: Enzyklopadie Philosophie und Wissenschaftstheorie. J. B. Metzler, Stuttgart, Weimar, 2004.
- [BM95] Bösenberg, D.; Metzen, H.: Lean Management. Vorsprung durch schlanke Konzepte. Verl. Moderne Industrie, Landsberg/Lech, 1995.

-
- [BMR14] Brocke vom, J.; Mathiassen, L.; Rosemann, M.: Business Process Management. In WIRTSCHAFTSINFORMATIK, 2014.
- [BPB13] Berenbrink, V.; Purucker, J.; Bahlinger, T.: Die Bedeutung von Wireframes in der agilen Softwareentwicklung. In HMD Praxis der Wirtschaftsinformatik, 2013, 50; S. 27–34.
- [Br73] Braten, S.: Model monopoly and communication: Systems theoretical notes on democratization. In Acta Sociologica, 16-2, 1973.
- [Br87] Brauer, W. et al.: Rechensysteme. Springer Berlin Heidelberg, Berlin, Heidelberg, 1987.
- [Br11a] Bruno, G. et al.: Key challenges for enabling agile BPM with social software. In Journal of Software Maintenance and Evolution: Research and Practice, 2011, 23; S. 297–326.
- [Br11b] Brenner, W. et al.: Bewusster Einsatz von Schatten-IT: Sicherheit & Innovationsförderung. In Studie SIT/01 Universität St. Gallen, 2011.
- [Br13a] Bruns, G.: Führung stärken, Veränderungen meistern, Mitarbeiter binden, 2013.
- [Br13b] Brauk, S.: Zurückeroberung der Zukunft — Chancen agiler IT. In HMD Praxis der Wirtschaftsinformatik, 2013, 50; S. 6–16.
- [Br14] Brucker-Kley, E. et al.: Business Process Management 2014. Status quo und Perspektiven eines ganzheitlichen Geschäftsprozessmanagements. vdf Hochschulverlag AG, 2014.
- [BRK05] Bartmann, D.; Rajola, F.; Kallinikos, J. Hrsg.: Information systems in a rapidly changing economy. Proceedings. Institute for Management of Information Systems, Regensburg, op. 2005.
- [BRU00] Becker, J.; Rosemann, M.; Uthmann, C.: Guidelines of Business Process Modeling. In (Aalst, W.; Desel, J.; Oberweis, A. Hrsg.): Business Process Management. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000; S. 30–49.
- [BSD14] Bouguettaya, A.; Sheng, Q.Z.; Daniel, F. Hrsg.: Advanced Web Services. Springer New York, New York, NY, 2014.

-
- [Bu13] Bundesamt für Sicherheit in der Informationstechnik: Band AH, Kapitel 3: Hilfsmittel zum HV-Kompendium. AH 3.1: Leitfaden zur Identifikation und Analyse kritischer Geschäftsprozesse.
https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Hochverfuegbarkheit/BandAH/AH3_1_Leitfaden_Phase_S.pdf?__blob=publicationFile, 27.07.2015.
- [Bu15] Bundesministerium für Wirtschaft und Energie: Industrie 4.0 und Digitale Wirtschaft. Impulse für Wachstum, Beschäftigung und Innovation.
<http://www.bmwi.de/DE/Mediathek/publikationen,did=704620.html>, 27.07.2015.
- [Bu92] Budde, R. et al.: Prototyping. An Approach to Evolutionary System Development. Springer Berlin Heidelberg, Berlin, Heidelberg, 1992.
- [Bu94] Bußler, Christoph: Policy resolution in workflow management systems. Digital Tech. J. 6, 4, 1994, S. 26–49.
- [BW09] Best, E.; Weth, M.: Geschäftsprozesse optimieren. Der Praxisleitfaden für erfolgreiche Reorganisation. Gabler, Wiesbaden, 2009.
- [BZ11] Bekele, T. M.; Zhu, W.: Towards collaborative business process management development current and future approaches: 2011 IEEE 3rd International Conference on Communication Software and Networks (ICCSN), 2011; S. 458–462.
- [Ca86] Capurro, R.: Hermeneutik der Fachinformation. K. Alber, Freiburg, 1986.
- [ÇD14] Çulha, D.; Dođru, A.: Towards an Agile Methodology for Business Process Development. In (van der Aalst, W. et al. Hrsg.): S-BPM ONE - Scientific Research. Springer International Publishing, Cham, 2014; S. 133–142.
- [CDR12] Cornax, M. C.; Dupuy-Chessa, S.; Rieu, D.: Choreographies in BPMN 2.0: New Challenges and Open Questions. In (Schönberger, A.; Kopp, O.; Lohmann, N. Hrsg.): CEUR Workshop, 2012; S. 50–57.
- [CF04] Conboy, K.; Fitzgerald, B.: Toward a conceptual framework of agile methods. In (Mehandjiev, N. et al. Hrsg.): ACM Workshop 2004, 2004; S. 37.

-
- [CH01] Cockburn, A.; Highsmith, J.: Agile software development, the people factor. In *Computer*, 2001, 34; S. 131–133.
- [Ch06] Chang, J. F.: *Business process management systems. Strategy and implementation*. Auerbach Publications, Boca Raton, FL, 2006.
- [Ch89] Checkland, P.: *Soft System Methodology: Human System Management* Vol.6 No.4, 1989; S. 273–289.
- [CKO92] Curtis, B.; Kellner, M.; Over, J.: Process modeling. *Commun. ACM* 35, 9 (September 1992), 75-90. DOI=<http://dx.doi.org/10.1145/130994.130998>.
- [Co02] Cockburn, A.: *Agile software development*. Addison-Wesley, Boston, 2002.
- [Co03] Cockburn, A.: *People and Methodologies in Software Development*. Doctoral Dissertation, Oslo, 2003.
- [Co04] Cohn, M.: *User stories applied. For agile software development*. Addison-Wesley, Boston, 2004.
- [Co05] Cockburn, A.: *Crystal clear. A human-powered methodology for small teams*. Addison-Wesley, Boston, 2005.
- [Co06] Cohn, M.: *Agile estimating and planning*. Prentice Hall Professional Technical Reference, Upper Saddle River, NJ, 2006.
- [Co09] Coldewey, J.: *Crystal Methodenfamilie in einer Minute*. <http://blog.coldewey.com/agile/2009/02/12/crystal-methodenfamilie-in-einer-minute/>, 27.07.2015.
- [Co15a] Code for America Labs: *Citizens have the power to help their cities*. www.codeforamerica.org, 27.07.2015.
- [Co15b] Conger, S.: *Six Sigma and Business Process Management*. In (Vom Brocke, J.; Rosemann, M. Hrsg.): *Handbook on Business Process Management 1*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015; S. 127–146.
- [Co92] Coy, W. et al.: *Sichtweisen der Informatik*. Vieweg+Teubner Verlag, Wiesbaden, 1992.
- [Co99] Cooper, A.: *The inmates are running the asylum*. Sams, Indianapolis, IN, 1999.

-
- [CR14] Comelli, G.; Rosenstiel, L. v.: Führung durch Motivation. Mitarbeiter für Unternehmensziele gewinnen. Vahlen, München, 2014.
- [Cr80] Crosby, P. B.: Quality is free. The art of making quality certain. New American Library, New York, Scarborough (Ontario), 1980.
- [CRA14] Conforto, E. C.; Rebutisch, E.; Amaral, D. C.: Project Management Agility Global Survey, 2014.
- [CSI11] Cheng, R.; Sadiq, S.; Indulska, M.: Framework for Business Process and Rule Integration: A Case of BPMN and SBVR. In (van der Aalst, W. et al. Hrsg.): Business Information Systems. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011; S. 13–24.
- [Da15] Davenport, T. H.: Process Management for Knowledge Work. In (Vom Brocke, J.; Rosemann, M. Hrsg.): Handbook on Business Process Management 1. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015; S. 17–35.
- [Da93] Davenport, T. H.: Process innovation. Reengineering work through information technology. Harvard Business School Press, Boston, Mass., 1993.
- [DD08] Dybå, T.; Dingsøy, T.: Empirical studies of agile software development: A systematic review. In Information and Software Technology, 2008, 50; S. 833–859.
- [De53] Deming, W. E.: Statistical techniques in industry. In Advanced Management 18(11): 8-12, 1953.
- [DGM14] Dombrowski, U.; Grundei, J.; Melcher, P. R.] Dombrowski, U.; Grundei, J.; Melcher, P. R.: Prozessorganisation in deutschen Unternehmen, 2014.
- [Dh06] Dhamdhere, D. M.: Operating systems. A concept-based approach. McGraw-Hill, Boston, 2006.
- [DH14] Drath, R.; Horch, A.: Industrie 4.0: Hit or Hype? [Industry Forum]. In IEEE Industrial Electronics Magazine, 2014, 8; S. 56–58.

-
- [DH63] Dalkey, N.; Helmer, O.: An Experimental Application of the DELPHI Method to the Use of Experts. In *Management Science*, 1963, 9; S. 458–467.
- [DI05] DIN: Qualitätsmanagementsysteme - Grundlagen und Begriffe (ISO 9000:2005); Dreisprachige Fassung EN ISO 9000:2005, 2005.
- [DI09] DIN: Informationstechnik - Lernen, Ausbilden und Weiterbilden - Qualitätsmanagement, -sicherung und -metriken - Teil 1. DIN Deutsches Institut für Normung e.V., 2009.
- [Di11] Dittler, U. Hrsg.: E-Learning. Einsatzkonzepte und Erfolgsfaktoren des Lernens mit interaktiven Medien. Oldenbourg, München, 2011.
- [DI11] DIN: Begriffe der Terminologielehre. DIN Deutsches Institut für Normung e.V., 2011.
- [Di12] Dingsøy, T. et al.: A decade of agile methodologies: Towards explaining agile software development. In *Journal of Systems and Software*, 2012, 85; S. 1213–1221.
- [DI88] DIN: Informationsverarbeitung - Begriffe - Aufbau digitaler Rechensysteme. DIN Deutsches Institut für Normung e.V., 1988.
- [DKK14] Drawehn, J.; Kochanowski, M.; Kötter, F.: Das Themenfeld Geschäftsprozessmanagement. In (Weisbecker, A.; Drawehn, J. Hrsg.): *Business Process Management Tools 2014. Marktüberblick*. Fraunhofer Verlag, Stuttgart, 2014; S. 9–18.
- [DKS13] Dräther, R.; Koschek, H.; Sahling, C.: *Scrum. Kurz & gut*. O'Reilly Verlag, Köln, 2013.
- [DMR15] Di Ciccio, C.; Marrella, A.; Russo, A.: Knowledge-Intensive Processes: Characteristics, Requirements and Analysis of Contemporary Approaches. In *Journal on Data Semantics*, 2015, 4; S. 29–57.

-
- [Do14] Donecker, A.: Selektions- und Rezeptionsprozesse im Kommunikationsraum Museum . Eine Erkundungsstudie am Fallbeispiel der Ausstellung „Foto + Film“ im Deutschen Museum München unter Verwendung von kommunikations- wissenschaftlichen Ansätzen der Mediennutzungs- und Rezeptionsforschung. http://www.diss.fu-berlin.de/diss/servlets/MCRFileNodeServlet/FUDISS_derivate_000000014910/01_Donecker_Alexandra.Diss.pdf, 27.07.2015.
- [Dr10] Draheim, D.: Business Process Technology: A Unified View on Business Processes, Workflows and Enterprise Applications. Springer, 2010.
- [Du03] Dumke, R.: Software Engineering. Eine Einführung für Informatiker und Ingenieure: Systeme, Erfahrungen, Methoden, Tools. Vieweg+Teubner Verlag, Wiesbaden, 2003.
- [Du13] Dumas, M. et al.: Fundamentals of Business Process Management. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [eB14] eBusinss-Lotse Aachen: Prototyping zur Verbesserung der Benutzerfreundlichkeit mobiler Software. Potenziale benutzerfreundlicher Software-Entwicklung. <http://www.mittelstand-digital.de/MD/Redaktion/DE/PDF/prototyping-mobile-software,property=pdf,bereich=md,sprache=de,rwb=true.pdf>, 27.07.2015.
- [El09] Eller, B.: Usability Engineering in Der Anwendungsentwicklung: Systematische Integration Zur Unterstützung Einer Nutzerorientierten Entwicklungsarbeit. Gabler, 2009.
- [Ep11] Epping, T.: Kanban für die Softwareentwicklung. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [Ev03] Evans, E.: Domain-driven design. Tackling complexity in the heart. Addison-Wesley, Boston, MA, 2003.
- [Ev98] Eversheim, W.: Werkzeugbau mit Zukunft. Strategie und Technologie. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.
- [Fa97] Fach, P. W.: Dialog-Spiele in der Mensch-Computer-Interaktion. Waxmann, 1997.

-
- [FE03] Förster, A.; Engels, G.: Quality Ensuring Development of Software Processes Helsinki, Finland, September 1-2, 2003, Proceedings: EWSPT 2003 Helsinki, 2003; S. 62–73.
- [Fe04] Feyhl, A.: Management und Controlling von Softwareprojekten. Software wirtschaftlich auswählen, entwickeln, einsetzen und nutzen. Gabler Verlag, Wiesbaden, 2004.
- [Fe05] Fettke, P.: Unified Modeling Language. In (Khosrow-Pour, M. Hrsg.): Encyclopedia of Information Science and Technology, Volume I–V. Hershey, PA, USA, 2005; S. S. 2921-2928.
- [FG08] Freund, J.; Götzer, K.: Vom Geschäftsprozess zum Workflow. Ein Leitfaden für die Praxis. Hanser, Carl, München, 2008.
- [FGO10] Fischer, M.; Grollius, T.; Ortner, E.: An Application of Constructive Logic in Computer Science. In (Ortner, E. Hrsg.): Konstruktive Informatik. Modellierung und Entwicklung von Anwendungssystemen. Shaker, 2010.
- [Fi06] Fischer, L.: Workflow handbook 2006. Future Strategies, [S.l.], 2006.
- [Fi13] Fischer, M.: Logikbasierte Prozessmodellierung. Ein ereignisorientierter Ansatz zur kontinuierlichen Modellierung und Qualitätssicherung von Geschäftsprozessen. Kovač, Hamburg, 2013.
- [Fi14] Fischer, S.: Big Data: Herausforderungen und Potenziale für deutsche Softwareunternehmen. In Informatik-Spektrum, 2014, 37; S. 112–119.
- [Fi87] Fischbein, E.: Intuition in science and mathematics. An educational approach. D. Reidel; Sold and distributed in the U.S.A. and Canada by Kluwer Academic, Dordrecht, Boston, Norwell, MA, U.S.A., 1987.
- [FL03] Frank, U.; Laak van, B. L.: Anforderungen an Sprachen zur Modellierung von Geschäftsprozessen. http://www.wi-inf.uni-duisburg-essen.de/FGFrank/documents/Arbeitsberichte_Koblenz/Nr34.pdf, 27.07.2015.
- [Fl10] Fleischmann, A.: What Is S-BPM? In (Buchwald, H. et al. Hrsg.): S-BPM ONE – Setting the Stage for Subject-Oriented Business Process Management. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010; S. 85–106.

-
- [Fl12] Fleischmann, A.: Subject-oriented business process management. Springer, Berlin, New York, 2012.
- [Fo96] Fowler, M.: Analysis Patterns: Reusable Object Models. Addison-Wesley Professional, 1996.
- [Fr13] Fraunhofer IESE: Studie – BPM Suites 2013, 2013.
- [Fr87] Freisleben, B.: Mechanismen zur Synchronisation paralleler Prozesse. Springer Berlin Heidelberg, Berlin, Heidelberg, 1987.
- [FRH10] Freund, J.; Rücker, B.; Henninger, T.: Praxishandbuch BPMN. Hanser, München, Wien, 2010.
- [FS06] Ferstl, O. K.; Sinz, E. J.: Grundlagen der Wirtschaftsinformatik. Oldenbourg, München, Wien, 2006.
- [FS12] Fischer, L.; Shapiro, R.: BPMN 2.0 handbook. Methods, concepts, case studies and standards in business process modeling notation. Future Strategies Inc., Lighthouse Point, Fla., 2012.
- [FS14] Fleischmann, A.; Schmidt, W.: Cognitive Support for S-BPM User Interfaces Intertwining Modeling and Execution. In (Stary, C. Hrsg.): Proceedings of the 2014 European Conference on Cognitive Ergonomics, 2014; S. 1–4.
- [FS99] Frieling, E.; Sonntag, K.: Lehrbuch Arbeitspsychologie. Huber, Bern, 1999.
- [FT02] Frege, G.; Textor, M.: Funktion, Begriff, Bedeutung. Vandenhoeck & Ruprecht, Göttingen, 2002.
- [Fu95] Fuchs-Heinritz, W.; Lautmann, R.; Rammstedt, O.; Wienold, H.: Lexikon zur Soziologie. 3.Auflage, Westdt. Verl., Opladen, 1995.
- [GA11] Gebhart, M.; Abeck, S.: Metrics for Evaluating Service Designs Based on SoaML. In International Journal on Advances in Software, 2011.
- [Ga12a] Gartner: Magic Quadrant for Intelligent Business Process Management Suites, 2012.
- [Ga12b] Gadatsch, A.: Grundkurs Geschäftsprozess-Management. Methoden und Werkzeuge für die IT-Praxis: Eine Einführung für Studenten und Praktiker. Vieweg+Teubner Verlag, Wiesbaden, 2012.

-
- [Ga12c] Gabriel, G.: Die Idee der Orthosprache. In (Mittelstraß, J. Hrsg.): Zur Philosophie Paul Lorenzens. Mentis, Münster, 2012; S. 11–25.
- [Ga13] García-Borgoñon, L. et al.: Software Process Management: A Model-Based Approach. In (Linger, H. et al. Hrsg.): Building Sustainable Information Systems. Springer US, Boston, MA, 2013; S. 167–178.
- [Ga15] Gadatsch, A.: Geschäftsprozesse analysieren und optimieren. Praxistools zur Analyse, Optimierung und Controlling von Arbeitsabläufen. Springer Vieweg, Wiesbaden, 2015.
- [Ga84] Garvin, D. A.: What Does "Product Quality" Really Mean? In Sloan Management Review, 1984; S. 25–43.
- [Ga88] Garvin, D. A.: Managing quality. The strategic and competitive edge. Free Press; Collier Macmillan, New York, London, 1988.
- [Ga95] Gamma, E.: Design patterns. Elements of reusable object-oriented software. Addison-Wesley, Reading, Mass., 1995.
- [GD13] Goll, J.; Dausmann, M.: Architektur- und Entwurfsmuster der Softwaretechnik. Mit lauffähigen Beispielen in Java. Imprint: Springer Vieweg, Wiesbaden, 2013.
- [Ge11] Gebhart, M.: Qualitätsorientierter Entwurf von Anwendungsdiensten. KIT Scientific Publishing, 2011.
- [GHA11] Genon, N.; Heymans, P.; Amyot, D.: Analysing the Cognitive Effectiveness of the BPMN 2.0 Visual Notation. In (Hutchison, D. et al. Hrsg.): Software Language Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011; S. 377–396.
- [GHS13] Ghosh, S.; Heching, A. R.; Squillante, M. S.: A Two-phase Approach for Stochastic Optimization of Complex Business Processes: Proceedings of the 2013 Winter Simulation Conference: Simulation: Making Decisions in a Complex World. IEEE Press, Piscataway, NJ, USA, 2013; S. 1856–1868.
- [GL08] Gläser, J.; Laudel, G.: Experteninterviews und qualitative Inhaltsanalyse. Als Instrumente rekonstruierender Untersuchungen. VS Verlag für Sozialwissenschaften, Wiesbaden, 2008.

-
- [Gl13] Gloger, B.: Scrum. Produkte zuverlässig und schnell entwickeln. Hanser, München, 2013.
- [GM05] Glushko, R. J.; McGrath, T.: Document engineering. Analyzing and designing documents for business informatics & Web services. MIT Press, Cambridge, Mass., 2005.
- [GM10] Gethmann, C. F.; Mittelstraß, J.: Paul Lorenzen zu Ehren. UVK, Univ.-Verl. Konstanz, Konstanz, 2010.
- [GM91] Grünupp, A.; Muthig, K.-P.: Intellektuelle Werkzeuge und werzeugvermittelte Erfahrung: Bemerkungen zum Gegenstandsbereich der Software-Ergonomie. In (Ackermann, U. Hrsg.): Software-Ergonomie '91. Benutzerorientierte Software-Entwicklung. Vieweg+Teubner Verlag, Wiesbaden, 1991; S. 352–362.
- [GMW14a] Gebhart, M.; Mevius, M.; Wiedmann, P.: Application of Business Process Quality Models in Agile Business Process Management: eKNOW 2014, 2014; S. 152–158.
- [GMW14b] Gebhart, M.; Mevius, M.; Wiedmann, P.: Business Process Evaluation in Agile Business Process Management Using Quality Models. In International Journal on Advances in Life Sciences, 2014; S. 279–290.
- [GMW15] Gebhart, M.; Mevius, M.; Wiedmann, P.: Wie erleben Anwender ihre Geschäftsprozesse? User Feedback mittels Mobile App: Tagungsband Informatik 2015, 2015.
- [Gö05] Görz, G.: Dialogische Logik und mathematischer Unterrichtsdiskurs. In (Löffladt, G. Hrsg.): Mathematik - Logik - Philosophie. Ideen und ihre historischen Wechselwirkungen. Deutsch, Frankfurt am Main, 2005; S. 185–211.
- [Go13] Golnam, A. et al.: A Framework for Modeling Value in Service-Oriented Business Models – Conceptualizations and Graphical Representation. In (van der Aalst, W. et al. Hrsg.): Business Modeling and Software Design. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013; S. 125–147.
- [Go96] Golecki, R.: Können Computer denken? Unterrichtsmodul 2: Schlüsselproblem Sprachverstehen, 1996.

-
- [GPR06] Gruhn, V.; Pieper, D.; Röttgers, C.: MDA. Effektives Softwareengineering mit UML2 und Eclipse. Springer, Berlin, 2006.
- [Gr03] Greiffenberg, S.: Methoden als Theorien der Wirtschaftsinformatik. In (Uhr, W.; Esswein, W.; Schoop, E. Hrsg.): Wirtschaftsinformatik 2003 Band II. Physica-Verlag HD, Heidelberg, 2003; S. 947–967.
- [Gr05] Grimm, R.: Digitale Kommunikation. Oldenbourg, München, 2005.
- [Gr10] Grönniger, H.: Systemmodell-basierte Definition objektbasierter Modellierungssprachen mit semantischen Variationspunkten. Shaker, Aachen, 2010.
- [Gr82] Gröschner, R.: Dialogik und Jurisprudenz. Die Philosophie des Dialogs als Philosophie der Rechtspraxis. Mohr, Tübingen, 1982.
- [GS13] Gumm, H.-P.; Sommer, M.: Einführung in die Informatik. Oldenbourg, München, 2013.
- [GS97] Goebels, G.; Schnorrenberg, U.: Risikomanagement in Projekten. Methoden und ihre praktische Anwendung. Vieweg+Teubner Verlag, Wiesbaden, 1997.
- [Gu99] Gurr, C. A.: Effective Diagrammatic Communication: Syntactic, Semantic and Pragmatic Issues. In Journal of Visual Languages & Computing, 1999, 10; S. 317–342.
- [Ha03] Handke, J.: Multimedia im Internet. Konzeption und Implementierung. Oldenbourg, München, 2003.
- [Ha05a] Harrison-Broninski, K.: Human Interactions: The Heart and Soul of Business Process Management. Meghan-Kiffer Press, 2005.
- [Ha05b] Hagen, M.: Definition einer Sprache zur Beschreibung von Prozessmustern zur Unterstützung agiler Softwareentwicklungsprozesse.
<http://lips.informatik.uni-leipzig.de/files/2005-11.pdf>, 27.07.2015.

-
- [Ha08] Hanser, E.: Rollenveränderung in der Agilen Software-Entwicklung: Das "Projektmanagement-Labor" der BA Lörrach. In (Herzwurm, G.; Mikusz, M. Hrsg.): Industrialisierung des Software-Managements. Fachtagung, Stuttgart, 12-14.11.2008. Gesellschaft für Informatik, Bonn, 2008; S. 105–124.
- [Ha10a] Hanser, E.: Agile Prozesse: Von XP über Scrum bis MAP. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
- [Ha10b] Hammer, M.: What is Business Process Management? In (Vom Brocke, J.; Rosemann, M. Hrsg.): Handbook on business process management. Springer, Berlin, London, 2010.
- [Hä10] Häder, M.: Empirische Sozialforschung. Eine Einführung. VS, Verl. für Sozialwiss., Wiesbaden, 2010.
- [Ha14] Haisjackl, C. et al.: Understanding Declare models: strategies, pitfalls, empirical results. In Software & Systems Modeling, 2014.
- [Ha15] Harrison-Broninski, K.: Dealing with Human-Driven Processes. In (Vom Brocke, J.; Rosemann, M. Hrsg.): Handbook on Business Process Management 2. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015; S. 573–599.
- [HC93] Hammer, M.; Champy, J.: Reengineering the corporation. A manifesto for business revolution. HarperBusiness, New York, NY, ©1993.
- [He02] Heimig, I.: Grammatikbasierte Beschreibung von Geschäftsprozessen. Methodik für das strukturierte Verarbeiten von Modellen. Gabler Verlag, Wiesbaden, 2002.
- [He06] Heinemann, E.: Sprachlogische Aspekte rekonstruierten Denkens, Redens und Handelns. Aufbau einer Wissenschaftstheorie der Wirtschaftsinformatik. Dt. Univ.-Verl., Wiesbaden, 2006.
- [He12a] Herrmann, T.: Kreatives Prozessdesign. Konzepte und Methoden zur Integration von Prozessorganisation, Technik und Arbeitsgestaltung. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [He12b] Hedtstück, U.: Einführung in die theoretische Informatik. Formale Sprachen und Automatentheorie. Oldenbourg, München, 2012.

-
- [He75] Henne, H.: Sprachpragmatik. Nachschrift einer Vorlesung. M. Niemeyer, Tübingen, 1975.
- [He92] Heckmann, F.: Interpretationsregeln zur Auswertung qualitativer Interviews und sozialwissenschaftlich relevanter "Texte" Anwendungen der Hermeneutik für die empirische Sozialforschung. In (Hoffmeyer-Zlotnik, Jürgen H. P Hrsg.): Analyse verbaler Daten. Über den Umgang mit qualitativen Daten. Westdt. Verl., Opladen, 1992; S. 142–167.
- [Hi02a] Highsmith, J.: Agile software development ecosystems. Addison-Wesley, Boston, 2002.
- [Hi02b] Hitchman, S.: The Details of Conceptual Modelling Notations are Important - A Comparison of Relationship Normative Language.
<http://aisel.aisnet.org/cgi/viewcontent.cgi?article=2745&context=cais>, 27.07.2015.
- [Hi10] Hindelang, G.: Einführung in die Sprechakttheorie. Sprechakte, Äusserungsformen, Sprechaktsequenzen. De Gruyter, Berlin, New York, 2010.
- [Hi95] Hitchman, S.: Practitioner perceptions on the use of some semantic concepts in the entity–relationship model. In European Journal of Information Systems, 1995, 4; S. 31–40.
- [HM05] Heinzl, S.; Mathes, M.: Middleware in Java. Leitfaden zum Entwurf verteilter Anwendungen -- Implementierung von verteilten Systemen über JMS -- Verteilte Objekte über RMI und CORBA. Vieweg+Teubner Verlag, Wiesbaden, 2005.
- [HM08] Heinrich, G.; Mairon, K.: Objektorientierte Systemanalyse. Oldenbourg, München, 2008.
- [HMU11] Hopcroft, J. E.; Motwani, R.; Ullman, J. D.: Einführung in Automatentheorie, formale Sprachen und Berechenbarkeit. Pearson Studium, München [u.a.], 2011.
- [Ho95] Hollingsworth, D.: Workflow Management Coalition - The Workflow Reference Model, 1995.

-
- [HOS04] Heinemann, E.; Ortner, E.; Sternhuber, J.: Sprachbasierte Wissenrekonstruktion am Beispiel des Einkommensteuergesetzes: MobIS, 2004; S. 91–111.
- [HP10] Heinrich, R.; Paech, B.: Defining the Quality of Business Processes: Modellierung, 2010; S. 133–148.
- [HR00] Harel, D.; Rumpe, B.: Modeling Languages: Syntax, Semantics and All That Stuff, Part I: The Basic Stuff. Weizmann Science Press of Israel, Jerusalem, Israel, Israel, 2000.
- [HRS09] Hruschka, P.; Rupp, C.; Starke, G.: Warum Agilität? In (Hruschka, P.; Rupp, C.; Starke, G. Hrsg.): Agility kompakt. Spektrum Akademischer Verlag, Heidelberg, 2009; S. 1–11.
- [Hu14] Hummel, M.: State-of-the-Art: A Systematic Literature Review on Agile Information Systems Development: 2014 47th Hawaii International Conference on System Sciences (HICSS), 2014; S. 4712–4721.
- [HV01] Hofacker, I.; Vetschera, R.: Algorithmical approaches to business process design: Computers & Operations Research 28, 2001; S. 1253–1275.
- [HV03] Hamel, G.; Välikangas, L.: The quest of resilience. In Harvard Business Manager, 2003; S. 52–63.
- [HW05] Hippner, H.; Wilde, T.: Social Software. In WIRTSCHAFTSINFORMATIK, 2005, 47; S. 441–444.
- [HZ14] Haisjackl, C.; Zugal, S.: Investigating Differences between Graphical and Textual Declarative Process Models. In (van der Aalst, W. et al. Hrsg.): Advanced Information Systems Engineering Workshops. Springer International Publishing, Cham, 2014; S. 194–206.
- [Ig12] Iglar, M.: ESProNa - Eine Constraintsprache zur multimodalen Prozessmodellierung und navigationsgestützten Ausführung. <https://epub.uni-bayreuth.de/id/eprint/227>, 27.07.2015.
- [In03] Inhetveen, R.: Logik. Eine dialog-orientierte Einführung. Edition am Gutenbergplatz, Leipzig, 2003.

-
- [In14a] Inayat, I. et al.: A systematic literature review on agile requirements engineering practices and challenges. In *Computers in Human Behavior*, 2014.
- [In14b] Initiative D21: Digital-Index 2014.
- [Ja02] Janich, P.: Modell und Modelliertes. Zwecke und Methoden. In (Gethmann, C. F.; Lingner, S. Hrsg.): *Integrative Modellierung zum Globalen Wandel*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002; S. 15–31.
- [Ja08] Jablonski, S.: Prozessdesign und- modellierung für ein holistisches Prozessmanagement. In (Heinemann, E. Hrsg.): *Anwendungsinformatik. Die Zukunft des Enterprise-Engineering ; Festschrift für Erich Ortner zum 60. Geburtstag*. Nomos, Baden-Baden, 2008.
- [Ja10] Jablonski, S.: Do We Really Know How to Support Processes? Considerations and Reconstruction. In (Hutchison, D. et al. Hrsg.): *Graph Transformations and Model-Driven Engineering*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010; S. 393–410.
- [JAH00] Jeffries, R.; Anderson, A.; Hendrickson, C.: *Extreme programming installed*. Addison-Wesley, Boston, 2000.
- [JG08] Jochem, R.; Gundlach, C.: *Six Sigma - Fehler vermeiden, Prozesse verbessern, Kosten senken*, 2008.
- [JKM74] Janich, P.; Kambartel, F.; Mittelstrass, J.: *Wissenschaftstheorie als Wissenschaftskritik*. Aspekte Verlag, Frankfurt am Main, 1974.
- [JN14] Jeston, J.; Nelis, J.: *Business process management. Practical guidelines to successful implementations*, 2014.
- [Jo01] Jochum, F.: *Konstruktive Methoden der Softwareentwicklung*. Interner Forschungsbericht SG 2/2001 der Fachgruppe Systemgestaltung Fachbereich Informatik, FH Köln. <http://www.gm.fh-koeln.de/~fgsg/uwep/Gelesenes/KoMe.pdf>, 27.07.2015.
- [JS12] Jablonski, S.; Seitz, M.: Evolutionäres Prozess-Engineering — der angemessene Grad an Prozessunterstützung, *HMD Praxis der Wirtschaftsinformatik*, DOI 10.1007/BF03340742, 2012.

-
- [JS13] Jablonski, S.; Seitz, M.: Evolutionary Process Engineering: User Guide and Case Study for Adequate Process Support, Third International Conference on Business Intelligence and Technology (BUSTECH 2013), Valencia, 2013.
- [Ju12] Julian Krumeich et al.: Business process learning on the job: A design science oriented approach and its empirical evaluation. In Knowledge Management & E-Learning: An International Journal, 2012, 4; S. 395–414.
- [Ka12] Karle, T.: Kollaborative Softwareentwicklung auf Basis serviceorientierter Architekturen. KIT Scientific Publishing, Karlsruhe, 2012.
- [Ka95] Karagiannis, D.: BPMS. In ACM SIGOIS Bulletin, 1995, 16; S. 10–13.
- [KBL13] Krallmann, H.; Bobrik, A.; Levina, O.: Systemanalyse im Unternehmen. Prozessorientierte Methoden der Wirtschaftsinformatik. Oldenbourg-Verl., München, 2013.
- [Ke04] Kessler, H.: Didaktische Strategien beim Wissenstransfer im Spannungsfeld von bildungsdidaktischen und kommunikationswissenschaftlichen Ansprüchen. http://edoc.ub.uni-muenchen.de/3246/1/Kessler_Hansjoachim.pdf, 27.07.2015.
- [Ke11] Keiff, L.: Dialogical Logic. In (Edward N. Zalta Hrsg.): The Stanford Encyclopedia of Philosophy, 2011.
- [KF11a] Kurz, M.; Fleischmann, A.: BPM 2.0: Business Process Management Meets Empowerment. In (Fleischmann, A. et al. Hrsg.): Subject-Oriented Business Process Management. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011; S. 54–83.
- [KF11b] Knuppertz, T.; Feddern, U.: Prozessorientierte Unternehmensführung. Prozessmanagement ganzheitlich einführen und verankern. Schäffer-Poeschel, Stuttgart, 2011.
- [KH04] Kritzenberger, H.; Herczeg, M.: Multimediale und interaktive Lernräume. De Gruyter, München, 2004.
- [KK14] Komus, A.; Kamlowski, W.: Gemeinsamkeiten und Unterschiede von Lean Management und agilen Methoden, 2014.

-
- [KKS04] Klein, R.; Kupsch, F.; Scheer, A.-W.: Modellierung inter-organisationaler Prozesse mit Ereignisgesteuerten Prozessketten.
http://www.econbiz.de/archiv/sb/usb/iwi/modellierung_prozesse_prozessketten.pdf, 27.07.2015.
- [KL96] Kamlah, W.; Lorenzen, P.: Logische Propädeutik. Vorschule des vernünftigen Redens. Metzler, Stuttgart, Weimar, 1996.
- [KLS95] Krogstie, J.; Lindland, O. I.; Sindre, G.: Defining Quality Aspects for Conceptual Models: Proceedings of the IFIP International Working Conference on Information System Concepts: Towards a Consolidation of Views. Chapman & Hall, Ltd, London, UK, UK, 1995; S. 216–231.
- [KMW14] Kurz, F.; Mevius, M.; Wiedmann, P.: Nutzerorientierte Multimedia-Geschäftsprozessmodelle als Basis der Serviceorchestrierung. In (Schmietendorf, A.; Simon, F. Hrsg.): BSOA/BCloud 2014. 9. Workshop Bewertungsaspekte service- und cloudbasierter Architekturen, 04. November 2014, Frankfurt/Main. Shaker, Herzogenrath, 2014; S. 49–62.
- [KNS92] Keller, G.; Nüttgens, M.; Scheer, A.-W.: Semantische Prozeßmodellierung auf der Grundlage „Ereignisgesteuerter Prozeßketten (EPK)“, 1992.
- [Ko09] Ko, Ryan K. L.: A computer scientist's introductory guide to business process management (BPM). In Crossroads, 2009, 15; S. 11–18.
- [Ko10a] Komus, A.: Status und Trends im BPM, 2010.
- [Ko10b] Kobler, M.: Qualität von Prozessmodellen. Kennzahlen zur analytischen Qualitätssicherung bei der Prozessmodellierung. Logos, Berlin, 2010.
- [Ko11a] Kocian, C.: Geschäftsprozessmodellierung mit BPMN 2.0. Business Process Model and Notation im Methodenvergleich. https://www.hs-neu-ulm.de/fileadmin/user_upload/Forschung/HNU_Working_Paper/HNU_WP_16_Kocian_Geschaeftsprozessmodellierung.pdf, 27.07.2015.
- [Ko11b] Koch, S.: Einführung in das Management von Geschäftsprozessen. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

-
- [Ko14] Jens Kolb et al.: How Social Distance of Process Designers Affects the Process of Process Modeling: Insights From a Controlled Experiment: 29th Symposium On Applied Computing (SAC 2014), Enterprise Engineering Track. ACM Press, 2014; S. 1364–1370.
- [KP14] KPMG: Digitale Transformation in der Schweiz.
<http://www.kpmg.com/CH/de/Library/Articles-Publications/Documents/Advisory/pub-20141013-digital-transformation-in-der-schweiz-de.pdf>, 27.07.2015.
- [Kr01] Kreiser, L.: Gottlob Frege. Leben, Werk, Zeit. F. Meiner, Hamburg, 2001.
- [Kr05] Krcmar, H.: Informationsmanagement. Springer Berlin Heidelberg, 2005.
- [Kr08] Krenzin, M.: >>Dass wir etwas wissen, ohne zu wissen, warum...<< Intuition im Coaching. In (Bentner, A. Hrsg.): Erfolgsfaktor Intuition. Systemisches Coaching von Führungskräften. Vandenhoeck & Ruprecht, Göttingen, 2008.
- [Kr12a] Krimphove, D.: Logik. Einführung in das Denken. Haufe, Freiburg, 2012a.
- [Kr12b] Krogstie, J.: Model-Based Development and Evolution of Information Systems. Springer London, London, 2012b.
- [Kr97] Krumbiegel, J.: Integrale Gestaltung von Geschäftsprozessen und Anwendungssystemen in Dienstleistungsbetrieben. Deutscher Universitätsverlag, Wiesbaden, 1997.
- [KS10] Kniberg, H.; Skarin, M.: Kanban and Scrum. Making the most of both. C4Media, Inc., [S.l.], 2010.
- [KS97] Klockhaus, E.; Scheruhn, H.-J.: Modellbasierte Einführung betrieblicher Anwendungssysteme. Deutscher Universitätsverlag, Wiesbaden, 1997.
- [Ku13] Kurz, M. et al.: Planning for the Unexpected: Exception Handling and BPM. In (Fischer, H.; Schneeberger, J. Hrsg.): S-BPM ONE - Running Processes. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [Ku15] Kurz, F. et al.: Design of Interactional End-to-End Web Applications for Smart Cities: Proceedings of WWW 2015, Florence, 2015; S. 551–556.

-
- [Ku92] Kurz, E.: Einordnung von Expertensystemwerkzeugen im Software-Entwicklungsprozeß. In (Bullinger, H.-J. Hrsg.): Expertensysteme in Produktion und Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg, 1992; S. 149–174.
- [KWE08] Klarl, H.; Wolff, C.; Emig, C.: Abbildung von Zugriffskontrollaussagen in Geschäftsprozessmodellen. http://cm.tm.kit.edu/CM-Web/05.Publikationen/2008/%5BKW+08%5D_Abbildung_von_Zugriffskontrollaussagen_in_Geschaeftsprozessmodellen.pdf, 27.07.2015.
- [Lä02] Läge, K.: Ideenmanagement. Grundlagen, optimale Steuerung und Controlling. Deutscher Universitätsverlag, Wiesbaden, 2002.
- [La09] Lange, K.: Und am Morgen Freude. Die Texte unserer Gedanken und Empfindungen. Peter Lang GmbH, Europaischer Verlag der Wissenschaften; Peter Lang [Distributor], Frankfurt am Main, Bern 9, 2009.
- [LC01] Leuf, B.; Cunningham, W.: The Wiki way. Quick collaboration on the Web. Addison-Wesley, Boston, 2001.
- [Le00] Leinenbach, S.: Interaktive Geschäftsprozessmodellierung. Dokumentation von Prozesswissen in einer Virtual Reality-gestützten Unternehmungsvisualisierung. Deutscher Universitätsverlag, Wiesbaden, 2000.
- [Le14] Leute, J.: Eine neue Definition agilen Projektmanagements. Analyse konzeptioneller Merkmale agilen Projektmanagements. Eul, Lohmar, Köln, 2014.
- [Le94] Leake, D. B.: Case-based reasoning. In The Knowledge Engineering Review, 1994, 9; S. 61.
- [Le99] Lehmann, F. R.: Fachlicher Entwurf von Workflow -Management-Anwendungen. Teubner, Stuttgart, 1999.
- [LG09] Lederer Antonucci, Y.; Goeke, R. J.: Analysis of Business Process Management Skills and Characteristics. <http://www2.widener.edu/~yantonic/surveyresults/ResultsMay09.pdf>, 27.07.2015.

-
- [LG99] Lauber, R.; Göhner, P.: Prozessautomatisierung 2. Modellierungskonzepte und Automatisierungsverfahren, Softwarewerkzeuge für den Automatisierungsingenieur, Vorgehensweise in den Projektphasen bei der Realisierung von Echtzeitsystemen. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999.
- [Li04] Liker, J. K.: The Toyota way. 14 management principles from the world's greatest manufacturer. McGraw-Hill, New York, 2004.
- [Li05] Lindemann, U.: Methodische Entwicklung technischer Produkte. Methoden flexibel und situationsgerecht anwenden. Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 2005.
- [Li10] Link, M.: Zweistufiger Modellierungsansatz zum nachhaltigen Prozessmanagement. In (Ortner, E. Hrsg.): Konstruktive Informatik. Modellierung und Entwicklung von Anwendungssystemen. Shaker, 2010.
- [Li13] Lipps, T.: Grundzüge der Logik. Salzwasser Verlag GmbH, 2013.
- [LK14] Leopold, K.; Kaltenecker, S.: Kanban in der IT. Eine Kultur der kontinuierlichen Verbesserung schaffen. Hanser, Carl, München, 2014.
- [LKP15] Leymann, F.; Karastoyanova, D.; Papazoglou, M. P.: Business Process Management Standards. In (Vom Brocke, J.; Rosemann, M. Hrsg.): Handbook on Business Process Management 1. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015; S. 595–624.
- [LL78] Lorenzen, P.; Lorenz, K.: Dialogische Logik. Wissenschaftliche Buchgesellschaft [Abt. Verl.], Darmstadt, 1978.
- [LLN11] Lessen van, T.; Lübke, D.; Nitzsche, J.: Geschäftsprozesse automatisieren mit BPEL. Dpunkt.verlag, Heidelberg, 2011.
- [LLS10] Laudon, K. C.; Laudon, J. P.; Schoder, D.: Wirtschaftsinformatik. Eine Einführung. Pearson Studium, München, Boston, Mass. [u.a.], 2010.
- [LM11] Lorenz, K.; Mittelstrass, J.: Philosophische Variationen. Gesammelte Aufsätze unter Einschluss gemeinsam mit Jürgen Mittelstrass geschriebener Arbeiten zu Platon und Leibniz. De Gruyter, Berlin, 2011.

-
- [Lo09] Lorenz, K.: Dialogischer Konstruktivismus. Walter De Gruyter, Berlin, 2009.
- [Lo15a] Lorenz, K.: Zur Herkunft der Dialogbedingung im Dialogischen Aufbau der Logik. In (Mittelstrass, J.; Bülow, C. v. Hrsg.): Dialogische Logik. mentis Verlag GmbH, 2015a; S. 55–74.
- [Lo15b] Lohrmann, M.: Business Process Quality Management, 2015b.
- [Lo74] Lorenzen, P.: Konstruktive Wissenschaftstheorie. Suhrkamp, Frankfurt am Main, 1974.
- [Lo80] Lorenzen, P.: Rationale Grammatik. In Gethmann, C. F.: Theorie des wissenschaftlichen Argumentierens. Suhrkamp, Frankfurt am Main, 1980; S.73-94.
- [Lo87] Lorenzen, P.: Lehrbuch zur konstruktiven Wissenschaftstheorie, Mannheim [etc.], 1987.
- [Lo90] Lorenz, K.: Einführung in die philosophische Anthropologie. Wissenschaftliche Buchgesellschaft, Darmstadt, 1990.
- [LR13] Lohrmann, M.; Reichert, M.: Understanding Business Process Quality. In (Glykas, M. Hrsg.): Business Process Management. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013; S. 41–73.
- [LS75] Lorenzen, P.; Schwemmer, O.: Konstruktive Logik, Ethik und Wissenschaftstheorie. Bibliographisches Institut, Mannheim, 1975.
- [LSW14] Lohmann, N.; Song, M.; Wohed, P. Hrsg.: Business Process Management Workshops. Springer International Publishing, Cham, 2014.
- [LT75] Linstone, H. A.; Turoff, M.: The Delphi method. Techniques and applications. Addison-Wesley Pub. Co., Advanced Book Program, Reading, Mass., 1975.
- [Lü11] Lübbe, A.: Tangible Business Process Modeling, 2011.
- [LW97] Ludes, P.; Werner, A.: Multimedia-Kommunikation. VS Verlag für Sozialwissenschaften, Wiesbaden, 1997.
- [Ma01] Maguire, M.: Methods to support human-centred design. In International journal of human-computer studies, 2001, 55; S. 587–634.

-
- [Ma04] Mayer, H. O.: Handlungsorientiertes Lernen und eLearning. Grundlagen und Praxisbeispiele. Oldenbourg Verlag, München, 2004.
- [Ma10] Manouchehri Far, S.: Social Software in Unternehmen. Nutzenpotentiale und Adoption in der innerbetrieblichen Zusammenarbeit. Eul, Lohmar [u.a.], 2010.
- [Ma13] Mayer, F. L.: Erfolgsfaktoren von Social Media: Wie \("funktionieren"\) Wikis? Eine vergleichende Analyse kollaborativer Kommunikationssysteme im Internet, in Organisationen und in Gruppen. Lit, Berlin [u.a.], 2013.
- [Ma14] Martinsen, R.: Auf den Spuren des Konstruktivismus – Varianten konstruktivistischen Forschens und Implikationen für die Politikwissenschaft. In (Martinsen, R. Hrsg.): Spurensuche: Konstruktivistische Theorien der Politik. Springer Fachmedien Wiesbaden, Wiesbaden, 2014; S. 3–41.
- [Ma99] Mainzer, K.: Computernetze und virtuelle Realität. Leben in der Wissensgesellschaft. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999.
- [MAA13] Mohammad, A. H.; Alwada'n, T.; Ababneh, J. ".: Agile Software Methodologies: Strength and Weakness. In International Journal of Engineering Science and Technology (IJEST), 2013; S. 455–459.
- [MB15] Mittelstrass, J.; Bülow, C.v. Hrsg.: Dialogische Logik. mentis Verlag GmbH, 2015.
- [MBH09] Malaka, R.; Butz, A.; Hussmann, H.: Medieninformatik. Eine Einführung. Pearson Studium, München, Boston, Mass. [u.a.], 2009.
- [Me06] Mevius, M.: Kennzahlenbasiertes Management von Geschäftsprozessen mit Petri-Netzen. Verl. Dr. Hut, München, 2006.
- [Me07a] Meibauer, J.: Einführung in die germanistische Linguistik. Metzler, Stuttgart, 2007.
- [Me07b] Mendling, J.: Detection and prediction of errors in EPC business process models, 2007.
- [Me12a] Meinke, J.: Wissensmanagement im Bereich der universitären Forschung. Ergebnisse einer Delphi - Studie im Hochschulbereich, 2012.

-
- [Me12b] Mertens, P. et al.: Grundzüge der Wirtschaftsinformatik. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [Me12c] Meerkamm, S.: Ein Rahmenwerk für das Prozessdesign zur Identifikation, Klassifikation und Umsetzung von Anforderungen. - Dargestellt an der Konzeption des Prozesskonfigurators. https://epub.uni-bayreuth.de/253/1/Dissertation_Meerkamm.pdf, 27.07.2015.
- [Me13] Menge, F.: BPMN Roundtrip works: Interchange Demo with 8 Tools | BPM-Guide.de. <http://www.bpm-guide.de/2013/08/27/bpmn-roundtrip-works-interchange-demo-with-8-tools/>, 27.07.2015.
- [MG11] Mell, P.; Grance, T.: NIST SP 800-145, The NIST Definition of Cloud Computing, 2011.
- [MH15] Marchildon, P.; Hadaya, P.: The Key Contributions of the Operation Management and Information Systems Disciplines to Business Process Management: eKNOW 2015, 2015.
- [MHM14a] Malinova, M.; Hribar, B.; Mendling, J.: A Framework for Assessing BPM Success: 22nd European Conference on Information Systems, Tel Aviv, 2014.
- [MHM14b] Malinova, M.; Hendrik, L.; Mendling, J.: A Meta-Model for Process Map Design. In (Loucopoulos, P.; Pastor, O.; Petrounias, I. Hrsg.): The Practice of Enterprise Modeling. 7th IFIP WG 8.1 Working Conference, PoEM 2014, Manchester, UK, November 12-13, 2014. Proceedings. Springer, Berlin, Heidelberg, 2014.
- [Mi09] Mironovschi, L.: Komplimente und Komplimenterwiderungen im Russischen und im Deutschen. Ein interkultureller Vergleich. P. Lang, Frankfurt am Main, New York, 2009.
- [MI13] Mevius, M.; Iskhakova, G.: Nachhaltige Geschäftsprozesse durch Social BPM, 2013.
- [Mi15] Mittelstraß, J.: Dialogische Logik - Eine Einführung. In (Mittelstrass, J.; Bülow, C. v. Hrsg.): Dialogische Logik. mentis Verlag GmbH, 2015; S. 9–11.

-
- [MJ15] Markus, M. L.; Jacobson, D. D.: The Governance of Business Processes. In (Vom Brocke, J.; Rosemann, M. Hrsg.): Handbook on Business Process Management 2. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015; S. 311–332.
- [MJA15] Mark von Rosing; Joshua von Scheel; Asif Qumer Gill: Applying Agile Principles to 5BPM6. In (Scheel, Mark von RosingAugust-Wilhelm ScheerHenrik von Hrsg.): The Complete Business Process Handbook. Morgan Kaufmann, Boston, 2015; S. 553–577.
- [MM12] Malinova, M.; Mendling, J.: A Qualitative Research Perspective on BPM Adoption and the Pitfalls of Business Process Modeling Tallinn, Estonia, September 3, 2012. Revised Papers: Proceedings of Business Process Management Workshops - {BPM} 2012 International Workshops, 2012; S. 77–88.
- [MM13] Malinova, M.; Mendling, J.: The Effect Of Process Map Design Quality On Process Management Success: ECIS 2013; S. 160.
- [MNA07] Mendling, J.; Neumann, G.; Aalst van der, W.: Understanding the Occurrence of Errors in Process Models Based on Metrics. In (Hutchison, D. et al. Hrsg.): On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007; S. 113–130.
- [Mo09] Moody, D. L.: The “Physics” of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering: IEEE Transactions on Software Engineering, 2009; S. 756–779.
- [Mo10] Moness, V.: Gestaltung von Qualitätskennzahlensystemen für Geschäftsprozesse. - Eine theoretische und empirische Untersuchung -, 2010.
- [Mo12] Moser, C.: User Experience Design. Mit Erlebniszentrierter Softwareentwicklung zu Produkten, die Begeistern. Springer, Berlin, 2012.

-
- [Mo13] Motzko, C.: Praxis des Bauprozessmanagements. Termine, Kosten und Qualität zuverlässig steuern. Ernst, Berlin, 2013.
- [Mö97] Mösgen, P.: Ars vitae - ars moriendi zur Anthropologie Wilhelm Kamlahs, 1997.
- [MOW14] Mevius, M.; Ortner, E.; Wiedmann, P.: Gebrauchssprachliche Modellierung als Grundlage für agiles Geschäftsprozessmanagement. In (Fill, H.-G. et al. Hrsg.): GI-Edition Proceedings Band 225 - Modellierung 2014 -. 19.-21. März 2014 in Wien. Köllen, Bonn, 2014; S. 169–184.
- [MOW15] Mevius, M.; Ortner, E.; Wiedmann, P.: Enhanced Stakeholder Socialization using Common Language in Agile BPM - Living business processes models instead of rigid documentations: eKNOW 2015, 2015.
- [MR13] Mutschler, B.; Reichert, M.: Understanding the Costs of Business Process Management Technology. In (Glykas, M. Hrsg.): Business Process Management. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013; S. 157–194.
- [MRC07] Mendling, J.; Reijers, H.; Cardoso, J.: What Makes Process Models Understandable? In (Alonso, G.; Dadam, P.; Rosemann, M. Hrsg.): Business Process Management. Springer Berlin Heidelberg, 2007; S. 48–63.
- [MRI07] Muehlen zur, M.; Recker, J.; Indulska, M.: Sometimes Less is More: Are Process Modeling Languages Overly Complex?: 2007 11th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOC Workshops), 2007; S. 197–204.
- [MRR10] Mendling, J.; Recker, J.; Reijers, H. A.: On the Usage of Labels and Icons in Business Process Modeling. In International Journal of Information System Modeling and Design, 2010, 1; S. 40–58.
- [MRW77] McCall, J. A.; Richards, P. K.; Walters, G. F.: Factors in Software Quality. Volume I. Concepts and Definitions of Software Quality.
<http://www.dtic.mil/cgi-bin/GetTRDoc?Location=U2&doc=GetTRDoc.pdf&AD=ADA049014>, 27.07.2015.

-
- [MSW12] Mevius, M.; Stephan, R.; Wiedmann, P.: BPM(N)Easy – Agiles cloud- und servicebasiertes Geschäftsprozessmanagement. In (Schmietendorf, A.; Patzer, K. Hrsg.): BSOA 2012. Shaker, Aachen, 2012; S. 15–28.
- [MSW13] Mevius, M.; Stephan, R.; Wiedmann, P.: Innovative Approach for Agile BPM: eKNOW 2013, 2013; S. 160–165.
- [Mu04] Muehlen zur, M.: Organizational Management in Workflow Applications – Issues and Perspectives, *Information Technology and Management* 5, 271–291, 2004.
- [Mü01] Müller, H.J. et al. Hrsg.: Geschäftsprozessorientiertes Wissensmanagement - Von der Strategie zum Content. (Vorwort), 2001.
- [Mü11] Müller, T.: Zukunftsthema Geschäftsprozessmanagement.
http://www.pwc.de/de_de/de/prozessoptimierung/assets/pwc-gpm-studie.pdf, 27.07.2015.
- [Mü14] Müller, E.: KVP und Innovation. In (Müller, E. Hrsg.): Qualitätsmanagement für Unternehmer und Führungskräfte. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014; S. 79–90.
- [MW13] Mevius, M.; Wiedmann, P.: BPM(N)Easy 1.2 – Gebrauchssprachliche Gestaltung IT-basierter Prozesse. In (Schmietendorf, A.; Hanin, M. Hrsg.): BSOA/BCloud 2013. 8. Workshop Bewertungsaspekte service- und cloudbasierter Architekturen, 12. November 2013, Basel/Schweiz. Shaker, Herzogenrath, 2013; S. 31–46.
- [Na11] Natschläger, C.: Towards a BPMN 2.0 Ontology. In (van der Aalst, W. et al. Hrsg.): Business Process Model and Notation. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011; S. 1–15.
- [NB13] Nielsen, J.; Budiu, R.: Mobile Usability. Für iPhone, iPad, Android und Kindle. mitp, Heidelberg, 2013.
- [Ne14] Neirotti, P. et al.: Current trends in Smart City initiatives: Some stylised facts. In *Cities*, 2014, 38; S. 25–36.
- [NP75] Noreen, A.; Pollack, H.-W.: Die wissenschaftliche Betrachtung der Sprache. Durchgesehene Übersetzung ausgewählter Teile des schwedischen Werkes "Vart Sprak". Olms, Hildesheim [u.a.], 1975.

-
- [NPW03] Neumann, S.; Probst, C.; Wernsmann, C.: Kontinuierliches Prozessmanagement. In (Becker, J.; Kugeler, M.; Rosemann, M. Hrsg.): Prozessmanagement. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003; S. 309–335.
- [NR08] Nehmer, N.; Reuter, A.: An Exception Handling Framework, 18th ECOOP Doctoral Symposium and PhD Student Workshop, 2008.
- [NRM11] Niedermann, F.; Radeschütz, S.; Mitschang, B.: Business Process Optimization Using Formalized Optimization Patterns. In (van der Aalst, W. et al. Hrsg.): Business Information Systems. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011; S. 123–135.
- [OA07] OASIS: Web Services Business Process Execution Language Version 2.0. <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>, 27.07.2015.
- [Ob14] Obermeier, S. et al.: Wesen von Geschäftsprozessen? In (Obermeier, S. et al. Hrsg.): Geschäftsprozesse realisieren. Springer Fachmedien Wiesbaden, Wiesbaden, 2014; S. 1–15.
- [Ob96] Oberweis, A.: Modellierung betrieblicher Abläufe. In (Oberweis, A. Hrsg.): Modellierung und Ausführung von Workflows mit Petri-Netzen. Vieweg+Teubner Verlag, Wiesbaden, 1996; S. 31–51.
- [Ol07] Oltersdorf, K. M.: Über HMI - Versuch einer Definition: 7. Berliner Werkstatt Mensch-Maschine-Systeme, 2007; S. 387–390.
- [OM04] OMG: Business Process Modeling Notation (BPMN). Version 1.0 - May 3, 2004. http://www.omg.org/bpmn/Documents/BPMN_V1-0_May_3_2004.pdf, 27.07.2015.
- [OM06] OMG: Business Process Modeling Notation Specification. http://www.omg.org/bpmn/Documents/OMG_Final_Adopted_BPMN_1-0_Spec_06-02-01.pdf, 27.07.2015.
- [OM11a] OMG: Unified Modeling Language Superstructure, Version 2.4.1. <http://www.omg.org/spec/UML/2.4.1/Superstructure/PDF>, 27.07.2015.
- [OM11b] OMG: Documents Associated with Business Process Model and Notation (BPMN) Version 2.0. <http://www.omg.org/spec/BPMN/2.0/>, 27.07.2015.

-
- [OM15] OMG: Decision Model and Notation (DMN). (in progress version).
<http://www.omg.org/spec/DMN/>, 27.07.2015.
- [Ön88] Ōno, T.: Toyota production system. Beyond large-scale production.
Productivity Press, Cambridge, Mass., 1988.
- [Or04] Ortner, E.: Anthropozentrik und Sprachbasierung in der (Wirtschafts-) Informatik. In (Kuhlen, R. et al. Hrsg.): Wissen in Aktion. Der Primat der Pragmatik als Motto der Konstanzer Informationswissenschaft Festschrift für Rainer Kuhlen. UVK Verlagsgesellschaft, Konstanz, 2004.
- [Or05] Ortner, E.: Sprachbasierte Informatik. Wie man mit Wörtern die Cyber-Welt bewegt. Ed. am Gutenbergplatz, Leipzig, 2005.
- [Or12] Ortner, E.: Semantisch normierte Anwendungssysteme und die >>eingeschränkte Freiheit<< der IT-Nutzer. In (Mittelstraß, J. Hrsg.): Zur Philosophie Paul Lorenzens. Mentis, Münster, 2012; S. 127–162.
- [Or94] Ortner, E.: KASPER - Ein Projekt zur natürlichsprachlichen Entwicklung von Informationssystemen. In WIRTSCHAFTSINFORMATIK, 1994; S. 570–579.
- [Or95] Ortner, E.: Elemente einer methodenneutralen Konstruktionsprache für Informationssysteme. In Informatik - Forschung und Entwicklung, 1995, 10; S. 148–160.
- [Or97] Ortner, E.: Methodenneutraler Fachentwurf. Zu den Grundlagen einer anwendungsorientierten Informatik. Teubner, Stuttgart, 1997.
- [Os88] Ostasz, L.: Der Geregelt Dialog. Der Satz Vom Ausgeschlossenen Dritten Und Ein Dialog Über Ihn. In Philosophica, 1988.
- [Ös95] Österle, H.: Business engineering. Prozess- und Systementwicklung; [Geschäftsstrategie, Prozess, Informationssystem]. Springer, Berlin [u.a.], 1995.
- [Ot12] Ottensooser, A. et al.: Making Sense of Business Process Descriptions: An Experimental Comparison of Graphical and Textual Notations: Journal of Systems and Software (85)., 2012; S. 596–606.
- [OW04] Ortner, E.; Wedekind, H.] Ortner, E.; Wedekind, H., 2004.

-
- [Pa03] Panagiotou, G.: Bringing SWOT into focus. In Business Strategy Review, 2003; S. 8–10.
- [Pa06] Paulzen, O.: Qualität im Wissensmanagement - Modellierung und Bewertung von Wissensprozessen. Denk!Institut, Wiesbaden, 2006.
- [Pe13] Pelletier, J.: Mobile App Manual. How to Start Creating Mobile Apps Using jQuery Mobile and PhoneGap Build. BookBaby, Cork, 2013.
- [Pe62] Petri, C. A.: Kommunikation mit Automaten, 1962.
- [Pe97] Petersen, P.: Monographien zur konstruktiven Erziehungswissenschaft, 1997.
- [Pf96] Pfanmutter, K. et al.: Steigerung der Performance von Informatikprozessen. Führungsgrößen, Leistungsmessung und Effizienz im IT-Bereich. Vieweg+Teubner Verlag, Wiesbaden, 1996.
- [Pi11] Pink, D. H.: Drive. The surprising truth about what motivates us, 2011.
- [Pi12] Pitschke, J.: Was ist „Language Based BPM“? Eine kurze Erklärung. <http://www.enterprise-design.eu/files/images/downloads-wissen/wasistlanguagebasedbpm.pdf>, 27.07.2015.
- [PN12] Prilla, M.; Nolte, A.: Integrating Ordinary Users into Process Management: Towards Implementing Bottom-Up, People-Centric BPM. In (van der Aalst, W. et al. Hrsg.): Enterprise, Business-Process and Information Systems Modeling. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012; S. 182–194.
- [Po00] Pohland, S.: Globale Unternehmensarchitekturen. Methode zur Verteilung von Informationssystemen. Weißensee-Verl., Berlin, 2000.
- [Pö05] Pöschek, A.: Die Bedeutung in der Sprache. Die Bedeutung der Semantik in Wittgensteins Sprachtheorien. <http://www.poeschek.at/files/publications/wittgenstein-bedeutung-semantik.pdf>, 27.07.2015.
- [Po11] Pommeranz, I.: Komplexitätsbewältigung im Multiprojektmanagement: Die Handlungsperspektive der Multiprojektleiter. <https://opus.bibliothek.uni-augsburg.de/opus4/files/1590/DissertationInnaPommeranz.pdf>, 27.07.2015.

-
- [Pö15] Pörksen, B.: Schlüsselwerke des Konstruktivismus. VS Verlag für Sozialwissenschaften, Wiesbaden, 2015.
- [Pr02] Prange, C.: Organisationales Lernen und Wissensmanagement. Fallbeispiele aus der Unternehmenspraxis. Gabler, Wiesbaden, 2002.
- [Pr10] Prilla, M.: Wissensmanagement-Unterstützung für die Entwicklung und Nutzung von Prozessmodellen als wissensvermittelnde Artefakte. Eul, Lohmar, Köln, 2010.
- [Pr12] Pröpper, N.: Agile Techniken für klassisches Projektmanagement. Qualifizierung zum PMI-ACP. mitp, Heidelberg, München, Landsberg, Frechen, Hamburg, 2012.
- [PS07] PSYCHOGIOS, A.: Understanding Total Quality Management in Context: Qualitative Research on Managers' Awareness of TQM Aspects in the Greek Service Industry, 2007.
- [PW07] Parameswaran, M.; Whinston, A. B.: Social Computing: An Overview. In Communications of the Association for Information Systems, 2007.
- [Pw11] PwC: Zukunftsthema Geschäftsprozessmanagement. http://www.pwc.de/de_de/de/prozessoptimierung/assets/pwc-gpm-studie.pdf, 27.07.2015.
- [Pw14] PwC: Prozessmanagement – notwendiger Baustein für die Verwaltungsmodernisierung, 2014.
- [PW94] Pfeiffer, W.; Weiss, E.: Lean-Management. Grundlagen der Führung und Organisation lernender Unternehmen. Erich Schmidt, Berlin, 1994.
- [Py14] Python Software Foundation: Python. <https://www.python.org/>, 27.07.2015.
- [Ra98] Rajaraman, V.: Programming languages. In Resonance, 1998, 3; S. 43–54.
- [RAH06] Russell, N.; Aalst van der, W.; Hofstede ter, A.: Workflow Exception Patterns. In (Dubois, E.; Pohl, K. Hrsg.): Advanced Information Systems Engineering. Springer Berlin Heidelberg, 2006; S. 288–302.
- [RBH04] Revere, L.; Black, K.; Huq, A.: Integrating Six Sigma and CQI for improving patient care. In The TQM Magazine, 2004, 16; S. 105–113.

-
- [RBW08] Ravesteyn, P.; Betenburg, R.; Waal de, B.: In Search of Competencies Needed in BPM Projects. In Communications of the IIMA, 2008.
- [RC13] Rahman, S.; Clerbout, N.: Constructive Type Theory and the Dialogical Approach to Meaning. In Baltic International Yearbook of Cognition, Logic and Communication, 2013, 8.
- [RDS09] Racheva, Z.; Daneva, M.; Sikkel, K.: Value Creation by Agile Projects: Methodology or Mystery? In (Bomarius, F. et al. Hrsg.): Product-Focused Software Process Improvement. 10th International Conference, PROFES 2009, Oulu, Finland, June 15-17, 2009. Proceedings. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009; S. 141–155.
- [Re03] Renzl, B.: Wissensbasierte Interaktion. Selbst-evolvierende Wissensströme in Unternehmen. Deutscher Universitätsverlag, Wiesbaden, 2003.
- [Re10] Reisig, W.: Petrinetze. Modellierungstechnik, Analysemethoden, Fallstudien. Vieweg+Teubner Verlag / Springer Fachmedien Wiesbaden, Wiesbaden, Wiesbaden, 2010.
- [Re14] Reinmuth, F.: Logische Rekonstruktion. Ein hermeneutischer Traktat, 2014.
- [Re85] Rechenberg, P.: Werkzeuge der Softwaretechnik. Eine kommentierte Literaturlauswahl/ Software Engineering Tools - A Selected and Annotated Bibliography. In Elektronische Rechenanlagen, 27.Jahrgang, Heft 2/1985 1985; S.106-110.
- [Ri06] Riehm, T.: Abwägungsentscheidungen in der praktischen Rechtsanwendung. Argumentation, Beweis, Wertung. Beck, München, 2006.
- [Ri12] Ried, D.: Komposition von Prozessinstanzen unter Berücksichtigung von Leistungs- und Qualitätsmerkmalen. In (Ursula Goltz, Marcus Magnor, Hans-Jürgen Appelrath, Herbert Matthies, Wolf-Tilo Balke, Lars Wolf Hrsg.): INFORMATIK 2012. Gesellschaft für Informatik e.V. (GI), Bonn, 2012; S. 1318–1329.
- [Ri93] Ritchie, D.: The Development of the C Language†, 1993.

-
- [RK14] Rangiha, M. E.; Karakostas, B.: Process recommendation and role assignment in social business process management: 2014 Science and Information Conference (SAI), 2014; S. 810–818.
- [RLR11] Reggio, G.; Leotta, M.; Ricca, F.: “Precise is better than light” a document analysis study about quality of business process models: 2011 First International Workshop on Empirical Requirements Engineering (EmpiRE), 2011; S. 61–68.
- [Ro10] Rothlauf, J.: Total-quality-Management in Theorie und Praxis. Zum ganzheitlichen Unternehmensverständnis. Oldenbourg, München, 2010.
- [Ro15] Rosemann, M.: The Service Portfolio of a BPM Center of Excellence. In (Vom Brocke, J.; Rosemann, M. Hrsg.): Handbook on Business Process Management 2. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015; S. 381–398.
- [Ro96] Rosemann, M.: Komplexitätsmanagement in Prozessmodellen. Methodenspezifische Gestaltungsempfehlungen für die Informationsmodellierung. Gabler, Wiesbaden, ©1996.
- [Ro99] Roland Kaschek: Was sind eigentlich Modelle? In EMISA Forum, 1999, 9; S. 31–35.
- [RQ12] Rupp, C.; Queins, S.: UML 2 glasklar. Praxiswissen für die UML-Modellierung. Hanser, Carl, München, 2012.
- [RR98] Rahman, S.; Rückert, H.] Rahman, S.; Rückert, H.: Dialogische Logik und Relevanz, 1998.
- [RRH97] Reiß, M.; Rosenstolz von, L.; Hofmann, L. M.: Change management. Programme, Projekte und Prozesse. Schäffer-Poeschel, Stuttgart, 1997.
- [Ru01] Rumpe, B.: Extreme Programming - Back to Basics?
<http://www.ham.nw.schule.de/pub/bscw.cgi/d2370491/Rumpe2001.pdf>, 27.07.2015.
- [Rü01] Rückert, H.: Why Dialogical Logic? In (Wansing, H. Hrsg.): Essays on Non-Classical Logic, 2001; S. 165–185.

-
- [Ru04] Russell, N. et al.: Workflow Data Patterns: QUT Technical report, FIT-TR-2004-01, Queensland University of Technology, Brisbane, 2004.
- [Ru05] Russell, N. et al.: Workflow Resource Patterns: Identification, Representation and Tool Support. In (Pastor, O.; Falcao e Cunha, J Hrsg.): Proceedings of the 17th Conference on Advanced Information Systems Engineering (CAiSE'05). Springer-Verlag, Berlin, 2005; S. 216–232.
- [Ru06] Russell, N. et al.: Workflow ControlFlow Patterns: A Revised View, 2006.
- [Rü08] Rücker, B.: Welche Sprache spricht BPM? Es muss nicht immer BPEL sein. In JavaSpektrum, 2008; S. 17–21.
- [Ru11] Rumpe, B.: Modellierung mit UML. Sprache, Konzepte und Methodik. Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [Rü11] Rückert, H.: Dialogues as a dynamic framework for logic. College Publications, London, 2011.
- [Ru12] Rumpe, B.: Agile Modellierung mit UML. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [RZ12] Rentrop, C.; Zimmermann, S.: Schatten-IT. In HMD – Praxis der Wirtschaftsinformatik, 2012; S. 60–68.
- [RZ96] Riehle, D.; Züllighoven, H.: Understanding and using patterns in software development. In Theory and Practice of Object Systems, 1996, 2; S. 3–13.
- [Sa01a] Saussure, F. de: Grundfragen der allgemeinen Sprachwissenschaft. W. de Gruyter, Berlin, 2001a.
- [Sa01b] Sandmann, C.: Eine Orthosprache Zur Nat\ÜRlichsprachlichen Beschreibung Von Dynamischen Objektmodellen: Modellierung 2001, Workshop Der Gesellschaft F\ÜR Informatik e. V. (GI). GI, 2001b; S. 81–90.
- [Sa03] Sander, T.: Expressive (Rede-)Handlungen. In Divinatio. Studia culturologica series 18, 2003; S. 7–34.

-
- [Sa08] Salger, F. et al.: Comprehensive Architecture Evaluation and Management in Large Software-Systems. In (Becker, S.; Plasil, F.; Reussner, R. Hrsg.): Quality of Software Architectures. Models and Architectures. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008; S. 205–219.
- [Sá13] Sánchez-González, L. et al.: Improving Quality of Business Process Models. In (Maciaszek, L. A.; Zhang, K. Hrsg.): Evaluation of Novel Approaches to Software Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013; S. 130–144.
- [Sa14] Santis de, R.; Fasano, A.; Mignolli, N.; Villa, A.: Smart city: fact and fiction. <http://mpira.ub.uni-muenchen.de/54536/>, 27.07.2015.
- [SB02] Schwaber, K.; Beedle, M.: Agile software development with Scrum. Prentice Hall, Upper Saddle River, NJ, 2002.
- [Sc02] Scheer, A.-W.: ARIS--vom Geschäftsprozess zum Anwendungssystem. Springer, Berlin, 2002.
- [Sc04a] Schmelzer, H. J.: Geschäftsprozessmanagement in der Praxis. Produktivität steigern - Wert erhöhen - Kunden zufrieden stellen. Hanser, München, 2004.
- [Sc04b] Schwaber, K.: Agile project management with Scrum. Microsoft Press, Redmond, Wash., 2004.
- [Sc06] Schulmeister, R.: ELearning: Einsichten und Aussichten. Oldenbourg, München [u.a.], 2006.
- [Sc09a] Scott, M. L.: Programming language pragmatics. Elsevier/Morgan Kaufmann Pub., Amsterdam, Boston, 2009.
- [Sc09b] Scheer, A.-W.: Wirtschaftsinformatik zwischen Wissenschaft und Unternehmertum. In WIRTSCHAFTSINFORMATIK, 2009, 51; S. 88–93.
- [Sc12] Schuster, T.: Modellierung, Integration und Analyse von Ressourcen in Geschäftsprozessen. KIT Scientific Publishing, Karlsruhe, 2012.
- [Sc13a] Schicker, E.: Datenbanken und SQL. Eine praxisorientierte Einführung mit Hinweisen zu Oracle und MS-Access. Vieweg & Teubner, Wiesbaden, 2013.

-
- [Sc13b] Schenke, M.: Logikkalküle in der Informatik. Wie wird Logik vom Rechner genutzt? Imprint: Springer Vieweg, Wiesbaden, 2013.
- [Sc14a] Schwab, A. J.: Management-Techniken. In (Schwab, A. J. Hrsg.): Managementwissen für Ingenieure. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014; S. 427–463.
- [Sc14b] Schleicher, D.: Modellierung regelkonformer Geschäftsprozesse, 2014.
- [Sc15] Schöning, S.: Ein Process Mining-Rahmenwerk für agile, personenbezogene Prozesse, Dissertation, Universität Bayreuth, <https://epub.uni-bayreuth.de/2542/>, 2015.
- [Sc90] Scheer, A.-W.: EDV-orientierte Betriebswirtschaftslehre. Grundlagen für ein effizientes Informationsmanagement. Springer-Verl, Berlin [etc.], 1990.
- [Sc95] Schwaber, K.: SCRUM Development Process: Proceedings of the 10th Annual ACM Conference on Object Oriented Programming Systems, Languages, and Applications (OOPSLA, 1995; S. 117–134.
- [Sc97] Schienmann, B.: Objektorientierter Fachentwurf. Ein terminologiebasierter Ansatz für die Konstruktion von Anwendungssystemen. Teubner, Stuttgart, 1997.
- [Sc98] Schütte, R.: Grundsätze ordnungsmäßiger Referenzmodellierung. Konstruktion konfigurations- und anpassungsorientierter Modelle. Gabler Verlag, Wiesbaden, 1998.
- [Sc99] Scheer, A.-W.: „ARIS — House of Business Engineering“: Konzept zur Beschreibung und Ausführung von Referenzmodellen. In (Becker, J.; Rosemann, M.; Schütte, R. Hrsg.): Referenzmodellierung. Physica-Verlag HD, Heidelberg, 1999; S. 2–21.
- [Se15] Seidlmeier, H.: Prozessmodellierung mit ARIS®. Eine beispielorientierte Einführung für Studium und Praxis in ARIS 9. Springer Vieweg, Wiesbaden, 2015.
- [Se69] Searle, J. R.: Speech acts. An essay in the philosophy of languages. Cambridge University Press, 1969.
- [Se83] Searle, J. R.: Ein sprachphilosophischer Essay. Suhrkamp, Frankfurt, 1983.

-
- [Se16] Seitz, M.: Der angemessene Grad an Prozessunterstützung : ein methodisches Rahmenwerk, Dissertation, <https://epub.uni-bayreuth.de/id/eprint/2897>, 2016.
- [SF04] Smith, H.; Fingar, P.: The Third Wave, 2004.
- [SH11] Stahlknecht, P.; Hasenkamp, U.: Einführung in die Wirtschaftsinformatik. Springer, Berlin [u.a.], 2011.
- [SH14] Simon, C.; Hientzsch, B. Hrsg.: Prozesseigner. Springer Fachmedien Wiesbaden, Wiesbaden, 2014.
- [SH15] Scheer, A.-W.; Hoffmann, M.: The Process of Business Process Management. In (Vom Brocke, J.; Rosemann, M. Hrsg.): Handbook on Business Process Management 2. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015; S. 351–380.
- [Sh48] Shannon, C.: A Mathematical Theory of Communication. In Bell System Technical Journal, 1948, 27; S. 379-423, 623-656.
- [Si07] Silvius, A. J. G.: Exploring Differences in the Perception of Business & IT Alignment, 2007.
- [Si09] Silva, A. R. et al.: AGILIPO: Embedding Social Software Features into Business Process Tools. In (Rinderle-Ma, S.; Sadiq, S.; Leymann, F. Hrsg.): Business Process Management Workshops, 2009; S. 1–6.
- [Si11] Sinclair, M.: Handbook of intuition research. Edward Elgar Pub., Cheltenham, Northampton, MA, 2011.
- [SK10] Schwarzer, B.; Krcmar, H.: Wirtschaftsinformatik. Grundlagen betrieblicher Informationssysteme. Schäffer-Poeschel, Stuttgart, 2010.
- [SL09] Schmalt, H.-D.; Langens, T.: Motivation. Kohlhammer, Stuttgart, 2009.
- [SN09] Schmidt, R.; Nurcan, S.: BPM and social software: Business Process Management Workshops, 2009; S. 649–658.
- [Sp11] Spath, D. Hrsg.: Geschäftsprozesse und IT. Werkzeuge für die effiziente Gestaltung von Business Processes; [Veranstaltung mit Fachausstellung, Stuttgart, 30. März 2011]. Fraunhofer Verlag, Stuttgart, 2011.

-
- [SS05] Schiefer, J.; Seufert, A.: Management and Controlling of Time-Sensitive Business Processes with Sense & Respond: International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06), 2005; S. 77–82.
- [SS12] Sedick, F.; Seymour, L. F.: BPMN usage: An analysis of influencing factors. In (Møller, C.; Chaudhry, S. S. Hrsg.): Advances in enterprise information systems II. Proceedings of the fifth International Conference on Research and Practical Issues of Enterprise Information Systems (CONFENIS 2011), Aalborg, Denmark, October 16-18, 2011. CRC Press, Boca Raton, ©2012.
- [SS13] Schwaber, K.; Sutherland, J.: The Scrum Guide™. The Definitive Guide to Scrum: The Rules of the Game.
<http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf>, 27.07.2015.
- [SSP14] Schall, D.; Satzger, B.; Psailer, H.: Crowdsourcing tasks to social networks in BPEL4People. In World Wide Web, 2014, 17; S. 1–32.
- [ŠT13] Škrinjar, R.; Trkman, P.: Increasing process orientation with business process management: Critical practices'. In International Journal of Information Management, 2013, 33; S. 48–60.
- [St14] Stiehl, V.: Process-driven applications with BPMN, 2014.
- [St73] Stachowiak, H.: Allgemeine Modelltheorie. Springer-Verlag, Wien, New York, ©1973.
- [St86] Stekeler-Weithofer, P.: Grundprobleme der Logik. Elemente einer Kritik der formalen Vernunft. W. de Gruyter, Berlin, New York, 1986.
- [SU77] SUGIMORI, Y. et al.: Toyota production system and Kanban system Materialization of just-in-time and respect-for-human system. In International Journal of Production Research, 1977, 15; S. 553–564.
- [Sy12] Symantec: State of Mobility Survey, 2012.

-
- [SZ15] Schönig, S.; Zeising, M.: The DPIL Framework: Tool Support for Agile and Resource-Aware Business Processes. BPM 2015 Demo Track, Innsbruck, Austria, 2015.
- [Te13] Telefónica Germany GmbH & Co. OHG: Telefónica Deutschland _10 Trendstudie - Mein Digitales Ich - Leben in der Netzwerkgesellschaft, 2013.
- [Th05] Thomas, O.: Das Modellverständnis in der Wirtschaftsinformatik: Historie, Literaturanalyse und Begriffsexplikation. In Veröffentlichungen des Instituts für Wirtschaftsinformatik, 2005.
- [Th10] Thonemann, U.: Operations management. Konzepte, Methoden und Anwendungen. Pearson Studium, München [u.a.], 2010.
- [Th13] Thieme, P.: Entwicklung einer neuen Methode zur Prozessleistungsmessung. Universitätsbibliothek der Universität Stuttgart, Stuttgart, 2013.
- [TN14] Termer, F.; Nissen, V.: Zum Begriff der Agilität - Betrachtungen und Implikationen aus etymologischer Perspektive. Techn. Univ. Inst. für Wirtschaftsinformatik; Univ.-Bibliothek, Ilmenau, Ilmenau, 2014.
- [Tø14] Tønnessen, T.: Business Process Reengineering. In (Tønnessen, T. Hrsg.): Managing Process Innovation through Exploitation and Exploration. Springer Fachmedien Wiesbaden, Wiesbaden, 2014; S. 27–33.
- [TP13] Thiemich, C.; Puhmann, F.: An Agile BPM Project Methodology. In (Hutchison, D. et al. Hrsg.): Business Process Management. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013; S. 291–306.
- [Tr08] Trabant, J.: Was ist Sprache? Beck, München, 2008.
- [Tr12] Trepper, T.: Agil-systemisches Softwareprojektmanagement. Springer Fachmedien Wiesbaden, Wiesbaden, 2012.
- [TW14] Tratt, L.; Welc, A.: Programming Languages. In IEEE Software, 2014, 31; S. 33–34.
- [Uf11] Ufer, J.: Entwicklung workflowbasierter eLearning-Dienste für den Einsatz an Realschulen. Logos-Verl, Berlin, 2011.

-
- [Va07] Vanderfeesten, I. et al.: Quality metrics for business process models. In BPM and Workflow handbook, 2007; S. 179–190.
- [Ve13a] Verginadis, Y. et al.: Addressing agility in collaborative processes: A comparative study: 2013 7th IEEE International Conference on Digital Ecosystems and Technologies (DEST) - Complex Environment Engineering, 2013; S. 120–125.
- [Ve13b] VersionOne: 7th-Annual-State-of-Agile-Development-Survey, 2013.
- [Vi11] Villers, J.: Die performative Wende. Austins Philosophie sprachlicher Medialität. Königshausen & Neumann, Würzburg, 2011.
- [vLS14] van der Aalst, Wil M. P.; La Rosa, M.; Santoro, F. M.: WI – Call for Papers Heft 1/2016. In WIRTSCHAFTSINFORMATIK, 2014, 56; S. 339–340.
- [Vo03a] Vom Brocke, J.: Referenzmodellierung. Gestaltung und Verteilung von Konstruktionsprozessen. Logos-Verl., Berlin, 2003.
- [Vo03b] Vogel-Heuser, B.: Systems software engineering. [angewandte Methoden des Systementwurfs für Ingenieure]. Oldenbourg-Industrieverl., München, 2003.
- [Vo14] Vom Brocke, J. et al.: Ten principles of good business process management. In Business Process Management Journal, 2014, 20; S. 530–548.
- [Vo95] Vossenkuhl, W.: Ludwig Wittgenstein. C.H. Beck, München, ©1995.
- [VS15] Vom Brocke, J.; Schmiedel, T.: BPM - Driving Innovation in a Digital World. Springer International Publishing, Cham, 2015.
- [VTM08] Vergidis, K.; Tiwari, A.; Majeed, B.: Business Process Analysis and Optimization: Beyond Reengineering. In IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 2008, 38; S. 69–82.
- [Wa87] Wahren, H.-K. E.: Zwischenmenschliche Kommunikation und Interaktion in Unternehmen. Grundlagen, Probleme und Ansätze zur Lösung, 1987.
- [We00] Wells, D.: Extreme Programming Project.
<http://www.extremeprogramming.org/map/project.html>, 27.07.2015.

-
- [We03] Weigand, E.: Sprache als Dialog. Sprechakttaxonomie und kommunikative Grammatik, 2003.
- [We07] Weigend, M.: Intuitive Modelle der Informatik. Univ.-Verl., Potsdam, 2007.
- [We12] Weske, M.: Business process management. Concepts, languages, architectures. Springer, Berlin, New York, 2012.
- [We15] Weber, M.: Big Data und Geschäftsmodell - Innovationen in der Praxis: 40+ Beispiele.
https://www.bitkom.org/Publikationen/2015/Leitfaden/Big_Data_und_Geschaeftsmodell_Innovationen/Big_Data_und_Geschaeftsmodell_Innovationen_Leitfaden.pdf, 27.07.2015.
- [Wf12] WfMC: Process Definition Interface --XML Process Definition Language. Document Number WfMC-TC-1025. <http://www.xpdl.org/standards/xpdl-2.2/XPDL%202.2%20%282012-08-30%29.pdf>, 27.07.2015.
- [WGK13] Weitlaner, D.; Guettinger, A.; Kohlbacher, M.: Intuitive Comprehensibility of Process Models. In (Fischer, H.; Schneeberger, J. Hrsg.): S-BPM ONE - Running Processes. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013; S. 52–71.
- [Wh03] White, S. A.: XPDL and BPMN.
http://www.omg.org/bpmn/Documents/XPDL_BPMN.pdf, 27.07.2015.
- [Wh04] White, S. A.: Process Modeling Notations and Workflow Patterns.
http://www.omg.org/bpmn/Documents/Notations_and_Workflow_Patterns.pdf, 27.07.2015.
- [Wi03] Wittgenstein, L.: Logisch-philosophische Abhandlung. Tractatus logico-philosophicus. Suhrkamp, Frankfurt am Main, 2003.
- [Wi05] Wittges, H.: Verbindung von Geschäftsprozessmodellierung und Workflow-Implementierung. Dt. Univ.-Verl., Wiesbaden, 2005.
- [Wi07a] Wilhelm, R.: Prozessorganisation. Oldenbourg, München, 2007a.
- [Wi07b] Williams, L.: A Survey of Agile Development Methodologies.
<http://agile.csc.ncsu.edu/SEMaterials/AgileMethods.pdf>, 27.07.2015.

-
- [Wi09] Wieken, J.-H.: SQL. Einstieg für Anspruchsvolle. Addison-Wesley; Pearson Studium, München/Germany, München/Germany, 2009.
- [Wi15] Wieland, U. et al.: Process performance measurement system – towards a customer-oriented solution. In Business Process Management Journal, 2015, 21; S. 312–331.
- [Wi53] Wittgenstein, L.: Philosophische Untersuchungen, 1953.
- [WJR91] Womack, J. P.; Jones, D. T.; Roos, D.: The machine that changed the world. How Japan's secret weapon in the global auto wars will revolutionize western industry. HarperPerennial, New York, NY, 1991.
- [WK06] Wind, M.; Kröger, D.: Handbuch IT in der Verwaltung. Springer, Berlin, 2006.
- [WK14] Wang, M.; Kumar, K.: Developing Flexible Business Process Management Systems Using Modular Computing Technologies. In (Sushil; Stohr, E. A. Hrsg.): The Flexible Enterprise. Springer India, New Delhi, 2014; S. 239–255.
- [WM08] White, S. A.; Miers, D.: BPMN modeling and reference guide. Understanding and using BPMN develop rigorous yet understandable graphical representations of business processes. Future Strategies Inc., Lighthouse Point, Fla., 2008.
- [Wo05] Wohed, P. et al.: Pattern-based Analysis of BPMN - An extensive evaluation of the Control-flow, the Data and the Resource Perspectives, BPMcenter.org, 2005.
- [Wö10] Wörzberger, R.: Management dynamischer Geschäftsprozesse auf Basis statischer Prozessmanagementsysteme. Aachener Informatik-Berichte, 2010.
- [WOI04] Wedekind, H.; Ortner, E.; Inhetveen, R.: Informatik als Grundbildung. In Informatik-Spektrum, 2004, 27; S. 172–180.
- [WR85] Wingert, B.; Riehm, U.: Computer als Werkzeug. Anmerkungen zu einem verbreiteten Mißverständnis.: Technik und Gesellschaft, Jahrbuch 3, 1985; S. 107–131.

-
- [WW06] Wang, M.; Wang, H.: From process logic to business logic - A cognitive approach to business process management. In *Information & Management*, 2006, 43; S. 179–193.
- [WW04] B. Weber; W. Wild: An Agile Approach to Workflow Management: Modellierung'04, 2004; S. 187–201.
- [Yi08] Yi, K.: Mining a Web2.0 Service for the Discovery of Semantically Similar Terms: A Case Study with Del.icio.us. In (Buchanan, G.; Masoodian, M.; Cunningham, S. J. Hrsg.): *Digital Libraries: Universal and Ubiquitous Access to Information*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008; S. 321–326.
- [YJS05] YeongSeok, L.; JungHyun, B.; Seokkoo, S.: Development of quality evaluation metrics for BPM (business process management) system: Fourth Annual ACIS International Conference on Computer and Information Science (ICIS'05), 2005; S. 424–429.
- [Za12] Zahn, Lisa Dyer. Flournoy Henry. Ines Lehmann. Guy Lipof. Fahad Osmani. Dennis Parrott. Wim Peeters. Jonas: *Scaling BPM Adoption. From Project to Program with IBM Business Process Manager*. IBM Redbooks, [S.l.], 2012.
- [Zi91] Zitterbarth, W.: *Der Erlanger Konstruktivismus in seiner Beziehung zum Konstruktiven Realismus*. In (Peschl, M. F. Hrsg.): *Formen des Konstruktivismus in Diskussion. Materialien zu den "Acht Vorlesungen über den Konstruktiven Realismus"*. WUV, Wien, 1991.
- [ZKJ02] Zoche, P.; Kimpeler, S.; Joepgen, M.: *Virtuelle Mobilität: Ein Phänomen mit physischen Konsequenzen?* Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.
- [ZRF14] Zimmermann, S.; Rentrop, C.; Felden, C.: *Managing Shadow IT Instances- A Method to Control Autonomous IT Solutions in the Business Departments*, 2014.
- [Zü10] Zürn, M.: *Referenzmodell für die Fabrikplanung auf Basis von Quality Gates*. Jost-Jetter, Heimsheim, 2010.

EIGENE PUBLIKATIONEN

2016

1. Design of Interactional Decision Support Applications for E-Participation in Smart Cities, 04/2016, International Journal of Electronic Government Research (IJEGR) 12(2), Florian Kurz, Marco Mevius, Erich Ortner, Peter Wiedmann DOI: 10.4018/IJEGR.2016040102.

2015

2. Wie erleben Anwender ihre Geschäftsprozesse? User Feedback mittels Mobile App, 10/2015, Bonner Köllen Verlag, Peter Wiedmann, Marco Mevius, Michael Gebhart · http://www.gi.de/fileadmin/redaktion/2015_LNI/lni-p-246.pdf.
3. Schlanke Architekturen durch smarte Modellierung, 08/2015, DOAG/SOUG News, Peter Wiedmann, Florian Kurz, Marco Mevius, <http://www.doag.org/wissen/publikationen/fachzeitschriften/doagsoug-news.html>.
4. Design of Interactional End-to-End Web Applications for Smart Cities, 05/2015, www2015, Florence, DOI: 10.1145/2740908.2743906, Peter Wiedmann, Florian Kurz, Marco Mevius, Erich Ortner, <http://dx.doi.org/10.1145/2740908.2743906>.
5. Enhanced Stakeholder Socialization using Common Language in Agile BPM: Living business processes models instead of rigid documentations 02/2015 eKNOW 2015, Lisbon, ISBN: 978-1-61208-386-5, Peter Wiedmann, Erich Ortner, Marco Mevius.

-
6. Business Process Evaluation in Agile Business Process Management Using Quality Models, 01/2015, International Journal on Advances in Life Sciences, Iaria, Peter Wiedmann, Michael Gebhart, Marco Mevius, http://iariajournals.org/life_sciences/tocv6n34.html.

2014

7. Nutzerorientierte Multimedia-Geschäftsprozessmodelle als Basis der Serviceorchestrierung, 11/2014, BSOA 2014, pp. 49-62, ISBN 978-3-8440-2940-6, Peter Wiedmann, Florian Kurz, Marco Mevius.
8. Ausgesprochen hochwertig: Hybride Qualitätskontrolle in agilem BPM 05/2014, pp. 66-71, OBJEKTSpektrum, Peter Wiedmann, Florian Kurz, Michael Gebhart, Marco Mevius, [http://www.sigs-datacom.de/fileadmin/user_upload/zeitschriften/os/2014/05/gebhart et al OS 05_14_Et1G.pdf](http://www.sigs-datacom.de/fileadmin/user_upload/zeitschriften/os/2014/05/gebhart_et_al_OS_05_14_Et1G.pdf).
9. Schlank mit Qualität - Qualitätssicherung im agilen Business Process Management, 02/2014, pp. 70-75, OBJEKTSpektrum, Peter Wiedmann, Michael Gebhart, Marco Mevius.
10. Gebrauchssprachliche Modellierung als Grundlage für agiles Geschäftsprozessmanagement, 03/2014, Modellierung 2014, Vienna, pp. 169-184, ISBN 978-388579-619-0, Peter Wiedmann, Erich Ortner, Marco Mevius.
11. Application of Business Process Quality Models in Agile Business Process Management, 02/2014, eKNOW 2014, Barcelona, pp. 152-158, ISBN 978-1-61208-329-2, Peter Wiedmann, Michael Gebhart, Marco Mevius.

2013

12. BPM(N)Easy1.2 - Gebrauchssprachliche Gestaltung IT-basierter Prozesse, 11/2013, BSOA 2013, ISBN 978-3-8440-2108-0, Peter Wiedmann, Marco Mevius.
13. Collaboration in der Cloud 3.0 - Business Process Management und Social Media, 03/2013, Midrange Magazin, Peter Wiedmann, Horst Schönecker, Marco Mevius.
14. An Innovative Approach of agile Business Process Management, 03/2013, eKNOW 2013, Nice, ISBN 978-1-61208-254-7, Peter Wiedmann, Rolf Stephan, Marco Mevius.
15. Open Innovation Plattform zur Entwicklung und Evaluation eines agilen Geschäftsprozessmanagements 2013, Forum - Das Forschungsmagazin, Peter Wiedmann, Marco Mevius.

2012

16. BPM(N)Easy - Agiles cloud- und servicebasiertes Geschäftsprozessmanagement, 11/2012, BSOA 2012, ISBN 978-3-8440-1411-2, Peter Wiedmann, Rolf Stephan, Marco Mevius.

ANHANG

Die Anhänge stellen die erfassten und implementierten Geschäftsprozessmodelle des Anwendungsbeispiels (Abschnitt 6) dar. Die Modelle wurden mit Hilfe von BPM Touch und der Axon.ivy BPM Suite erstellt und direkt daraus entnommen.

Anhang 1: XML-Auszug der BPMN^{Easy}-Modelle erstellt mit BPM Touch

Die in dieser Arbeit verwendete Version von BPM Touch speichert die annotierten Mediafiles getrennt ab, sodass die XML-Dateien keine direkten „Links“ (DataObjects) zu den einzelnen Mediafiles besitzen.

Angebotsprozess Version 1
<pre><?xml version="1.0" encoding="UTF-8" ?> <definitions xmlns="http://www.omg.org/spec/BPMN/20100524/MODEL" xmlns:xs="http://www.w3.org/2001/XMLSchema-instance" xmlns:bpmndi="http://www.omg.org/spec/BPMN/20100524/DI" xmlns:dc="http://www.omg.org/spec/DD/20100524/DC" xmlns:di="http://www.omg.org/spec/DD/20100524/DI" xmlns:tns="http://www.jboss.org/drools" id="Definition" typeLanguage="http://www.java.com/javaTypes" expressionLanguage="http://www.mvel.org/2.0" xs:schemaLocation="http://www.omg.org/spec/BPMN/20100501/BPMN20.xsd" targetNamespace="http://www.bamero.de/bpmtouch"> <process id="_ae7c29aa-ea4b-412d-869a-683c82c21ee3" name="Angebotsprozess Version 1" isExecutable="false" processType="Private"> <documentation> <bpmtouch author="" /> </documentation> <startEvent id="_02f1f56e-861d-41e6-a723-17897ee1817d" name="startEvent" /> <sequenceFlow id="_4f1d1d0d-281a-4162-b92e-be9ef77cdc88" sourceRef="_02f1f56e-861d-41e6-a723-17897ee1817d" targetRef="_156a6a6b-93aa-4085-90b8-31f03570e8e8" /> <serviceTask id="_156a6a6b-93aa-4085-90b8-31f03570e8e8" name="Daten lesen" /> <sequenceFlow id="_ee2b96f7-e111-48df-817f-27add1f357b5" sourceRef="_156a6a6b-93aa-4085-90b8-31f03570e8e8" targetRef="_23f75b36-9ddd-4ae6-809b-8ed645984a56" /> <userTask id="_23f75b36-9ddd-4ae6-809b-8ed645984a56" name="Angebot erfassen" /> <sequenceFlow id="_f033b9f4-e3c7-48f2-bb64-ffe1b9fcb25d" sourceRef="_23f75b36-9ddd-4ae6-809b-8ed645984a56" targetRef="_e31209c7-daf0-42cd-b258-2ee304e941b3" /> <exclusiveGateway id="_e31209c7-daf0-42cd-b258-2ee304e941b3" name="exclusiveGateway" /> <sequenceFlow id="_7b45c101-60ad-47a3-b521-b09743a0c34f" sourceRef="_e31209c7-daf0-42cd-b258-2ee304e941b3" targetRef="_9a7e78d2-7c9f-412e-98f5-970bc627cfa9" /> <userTask id="_9a7e78d2-7c9f-412e-98f5-970bc627cfa9" name="Angebot prüfen" /></pre>

```

    <sequenceFlow id="_7076d516-c357-4f9b-86bc-ccf6e1544621"
sourceRef="_9a7e78d2-7c9f-412e-98f5-970bc627cfa9"
    targetRef="_68434e2d-ec6e-43d5-a11b-6e5e2e92ad44"/>
    <exclusiveGateway id="_68434e2d-ec6e-43d5-a11b-6e5e2e92ad44"
name="exclusiveGateway"/>
    <sequenceFlow id="_ea2c8183-3050-45b2-8f45-6d3a579f936a"
sourceRef="_68434e2d-ec6e-43d5-a11b-6e5e2e92ad44"
    targetRef="_e31209c7-daf0-42cd-b258-2ee304e941b3"/>
    <sequenceFlow id="_00d48138-5963-4466-b11a-3a0ec0bd75a0"
sourceRef="_68434e2d-ec6e-43d5-a11b-6e5e2e92ad44"
    targetRef="_4cb045ab-11ee-4579-9828-043bd4fa035d"/>
    <serviceTask id="_4cb045ab-11ee-4579-9828-043bd4fa035d"
name="Kunden benachrichtigen"/>
    <sequenceFlow id="_49d6dc68-9740-4950-a17c-072a777fe900"
sourceRef="_4cb045ab-11ee-4579-9828-043bd4fa035d"
    targetRef="_e124db05-23d0-4ada-b8a0-db8dd4df2aa3"/>
    <endEvent id="_e124db05-23d0-4ada-b8a0-db8dd4df2aa3"
name="endEvent"/>
  </process>
</bpmndi:BPMNDiagram>
<bpmndi:BPMNPlane bpmnElement="Minimal">
  <bpmndi:BPMNShape bpmnElement="_02f1f56e-861d-41e6-a723-
17897ee1817d">
    <dc:Bounds height="50" width="50" x="35" y="325"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_4f1d1d0d-281a-4162-b92e-
be9ef77cdc88">
    <di:waypoint y="350" x="85"/>
    <di:waypoint y="350" x="110"/>
    <di:waypoint y="350" x="110"/>
    <di:waypoint y="350" x="135"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_156a6a6b-93aa-4085-90b8-
31f03570e8e8">
    <dc:Bounds height="100" width="150" x="135" y="300"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_ee2b96f7-e111-48df-817f-
27add1f357b5">
    <di:waypoint y="350" x="285"/>
    <di:waypoint y="350" x="310"/>
    <di:waypoint y="350" x="310"/>
    <di:waypoint y="350" x="335"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_156a6a6b-93aa-4085-90b8-
31f03570e8e8">
    <dc:Bounds height="100" width="150" x="135" y="300"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNShape bpmnElement="_23f75b36-9ddd-4ae6-809b-
8ed645984a56">
    <dc:Bounds height="100" width="150" x="335" y="300"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_f033b9f4-e3c7-48f2-bb64-
ffe1b9fcb25d">
    <di:waypoint y="350" x="485"/>
    <di:waypoint y="350" x="510"/>
    <di:waypoint y="350" x="510"/>
    <di:waypoint y="350" x="535"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_23f75b36-9ddd-4ae6-809b-
8ed645984a56">
    <dc:Bounds height="100" width="150" x="335" y="300"/>
  </bpmndi:BPMNShape>

```

```

</bpmndi:BPMNShape>
<bpmndi:BPMNShape bpmnElement="_e31209c7-daf0-42cd-b258-
2ee304e941b3">
  <dc:Bounds height="50" width="50" x="535" y="325"/>
</bpmndi:BPMNShape>
<bpmndi:BPMNEdge bpmnElement="_7b45c101-60ad-47a3-b521-
b09743a0c34f">
  <di:waypoint y="350" x="585"/>
  <di:waypoint y="350" x="637"/>
  <di:waypoint y="213" x="637"/>
  <di:waypoint y="213" x="689"/>
</bpmndi:BPMNEdge>
<bpmndi:BPMNShape bpmnElement="_e31209c7-daf0-42cd-b258-
2ee304e941b3">
  <dc:Bounds height="50" width="50" x="535" y="325"/>
</bpmndi:BPMNShape>
<bpmndi:BPMNShape bpmnElement="_9a7e78d2-7c9f-412e-98f5-
970bc627cfa9">
  <dc:Bounds height="100" width="150" x="689" y="163"/>
</bpmndi:BPMNShape>
<bpmndi:BPMNEdge bpmnElement="_7076d516-c357-4f9b-86bc-
ccf6e1544621">
  <di:waypoint y="213" x="839"/>
  <di:waypoint y="213" x="864"/>
  <di:waypoint y="213" x="864"/>
  <di:waypoint y="213" x="889"/>
</bpmndi:BPMNEdge>
<bpmndi:BPMNShape bpmnElement="_9a7e78d2-7c9f-412e-98f5-
970bc627cfa9">
  <dc:Bounds height="100" width="150" x="689" y="163"/>
</bpmndi:BPMNShape>
<bpmndi:BPMNShape bpmnElement="_68434e2d-ec6e-43d5-a11b-
6e5e2e92ad44">
  <dc:Bounds height="50" width="50" x="889" y="188"/>
</bpmndi:BPMNShape>
<bpmndi:BPMNEdge bpmnElement="_ea2c8183-3050-45b2-8f45-
6d3a579f936a">
  <di:waypoint y="238" x="914"/>
  <di:waypoint y="439" x="914"/>
  <di:waypoint y="439" x="560"/>
  <di:waypoint y="375" x="560"/>
</bpmndi:BPMNEdge>
<bpmndi:BPMNShape bpmnElement="_68434e2d-ec6e-43d5-a11b-
6e5e2e92ad44">
  <dc:Bounds height="50" width="50" x="889" y="188"/>
</bpmndi:BPMNShape>
<bpmndi:BPMNEdge bpmnElement="_00d48138-5963-4466-b11a-
3a0ec0bd75a0">
  <di:waypoint y="213" x="939"/>
  <di:waypoint y="213" x="964"/>
  <di:waypoint y="213" x="964"/>
  <di:waypoint y="213" x="989"/>
</bpmndi:BPMNEdge>
<bpmndi:BPMNShape bpmnElement="_68434e2d-ec6e-43d5-a11b-
6e5e2e92ad44">
  <dc:Bounds height="50" width="50" x="889" y="188"/>
</bpmndi:BPMNShape>
<bpmndi:BPMNShape bpmnElement="_4cb045ab-11ee-4579-9828-
043bd4fa035d">
  <dc:Bounds height="100" width="150" x="989" y="163"/>
</bpmndi:BPMNShape>

```

```

    <bpmndi:BPMNEdge bpmnElement="_49d6dc68-9740-4950-a17c-
072a777fe900">
      <di:waypoint y="213" x="1139"/>
      <di:waypoint y="213" x="1164"/>
      <di:waypoint y="213" x="1164"/>
      <di:waypoint y="213" x="1189"/>
    </bpmndi:BPMNEdge>
    <bpmndi:BPMNShape bpmnElement="_4cb045ab-11ee-4579-9828-
043bd4fa035d">
      <dc:Bounds height="100" width="150" x="989" y="163"/>
    </bpmndi:BPMNShape>
    <bpmndi:BPMNShape bpmnElement="_e124db05-23d0-4ada-b8a0-
db8dd4df2aa3">
      <dc:Bounds height="50" width="50" x="1189" y="188"/>
    </bpmndi:BPMNShape>
  </bpmndi:BPMNPlane>
</bpmndi:BPMNDiagram>
</definitions>

```

Angebotsprozess Version 2

```

<?xml version="1.0" encoding="UTF-8" ?>
<definitions xmlns="http://www.omg.org/spec/BPMN/20100524/MODEL"
xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"
xmlns:bpmndi="http://www.omg.org/spec/BPMN/20100524/DI"
xmlns:dc="http://www.omg.org/spec/DD/20100524/DC"
xmlns:di="http://www.omg.org/spec/DD/20100524/DI"
xmlns:tns="http://www.jboss.org/drools" id="Definition"
typeLanguage="http://www.java.com/javaTypes"
expressionLanguage="http://www.mvel.org/2.0"

xs:schemaLocation="http://www.omg.org/spec/BPMN/20100501/BPMN20.xsd"
targetNamespace="http://www.bamero.de/bpmtouch">
<process id="_ae7c29aa-ea4b-412d-869a-683c82c21ee3"
name="Angebotsprozess Version 2" isExecutable="false"
processType="Private">
  <documentation>
    <bpmtouch author=""/>
  </documentation>
  <startEvent id="_02f1f56e-861d-41e6-a723-17897ee1817d"
name="startEvent"/>
  <sequenceFlow id="_4f1d1d0d-281a-4162-b92e-be9ef77cdc88"
sourceRef="_02f1f56e-861d-41e6-a723-17897ee1817d"
targetRef="_156a6a6b-93aa-4085-90b8-31f03570e8e8"/>
  <serviceTask id="_156a6a6b-93aa-4085-90b8-31f03570e8e8"
name="Daten lesen"/>
  <sequenceFlow id="_ee2b96f7-e111-48df-817f-27add1f357b5"
sourceRef="_156a6a6b-93aa-4085-90b8-31f03570e8e8"
targetRef="_23f75b36-9ddd-4ae6-809b-8ed645984a56"/>
  <userTask id="_23f75b36-9ddd-4ae6-809b-8ed645984a56" name="Angebot
erfassen"/>
  <sequenceFlow id="_f033b9f4-e3c7-48f2-bb64-ffe1b9fcb25d"
sourceRef="_23f75b36-9ddd-4ae6-809b-8ed645984a56"
targetRef="_e31209c7-daf0-42cd-b258-2ee304e941b3"/>
  <exclusiveGateway id="_e31209c7-daf0-42cd-b258-2ee304e941b3"
name="exclusiveGateway"/>
  <sequenceFlow id="_7b45c101-60ad-47a3-b521-b09743a0c34f"
sourceRef="_e31209c7-daf0-42cd-b258-2ee304e941b3"
targetRef="_9a7e78d2-7c9f-412e-98f5-970bc627cfa9"/>

```

```

    <userTask id="_9a7e78d2-7c9f-412e-98f5-970bc627cfa9" name="Angebot
prüfen"/>
    <sequenceFlow id="_7076d516-c357-4f9b-86bc-ccf6e1544621"
sourceRef="_9a7e78d2-7c9f-412e-98f5-970bc627cfa9"
    targetRef="_68434e2d-ec6e-43d5-a11b-6e5e2e92ad44"/>
    <exclusiveGateway id="_68434e2d-ec6e-43d5-a11b-6e5e2e92ad44"
name="exclusiveGateway"/>
    <sequenceFlow id="_57e2a9c8-de5b-4b38-b91b-8746d2720f57"
sourceRef="_68434e2d-ec6e-43d5-a11b-6e5e2e92ad44"
    targetRef="_1d6d1924-55dc-4ec9-bf89-bf3b6a3c0181"/>
    <manualTask id="_1d6d1924-55dc-4ec9-bf89-bf3b6a3c0181"
name="Dialog führen"/>
    <sequenceFlow id="_c1b8350b-c97c-43c8-b1d8-aec9440bcedc"
sourceRef="_1d6d1924-55dc-4ec9-bf89-bf3b6a3c0181"
    targetRef="_e31209c7-daf0-42cd-b258-2ee304e941b3"/>
    <sequenceFlow id="_00d48138-5963-4466-b11a-3a0ec0bd75a0"
sourceRef="_68434e2d-ec6e-43d5-a11b-6e5e2e92ad44"
    targetRef="_4cb045ab-11ee-4579-9828-043bd4fa035d"/>
    <serviceTask id="_4cb045ab-11ee-4579-9828-043bd4fa035d"
name="Kunden benachrichtigen"/>
    <sequenceFlow id="_49d6dc68-9740-4950-a17c-072a777fe900"
sourceRef="_4cb045ab-11ee-4579-9828-043bd4fa035d"
    targetRef="_e124db05-23d0-4ada-b8a0-db8dd4df2aa3"/>
    <endEvent id="_e124db05-23d0-4ada-b8a0-db8dd4df2aa3"
name="endEvent"/>
</process>
<bpmndi:BPMNDiagram>
  <bpmndi:BPMNPlane bpmnElement="Minimal">
    <bpmndi:BPMNShape bpmnElement="_02f1f56e-861d-41e6-a723-
17897ee1817d">
      <dc:Bounds height="50" width="50" x="35" y="325"/>
    </bpmndi:BPMNShape>
    <bpmndi:BPMNEdge bpmnElement="_4f1d1d0d-281a-4162-b92e-
be9ef77cdc88">
      <di:waypoint y="350" x="85"/>
      <di:waypoint y="350" x="110"/>
      <di:waypoint y="350" x="110"/>
      <di:waypoint y="350" x="135"/>
    </bpmndi:BPMNEdge>
    <bpmndi:BPMNShape bpmnElement="_156a6a6b-93aa-4085-90b8-
31f03570e8e8">
      <dc:Bounds height="100" width="150" x="135" y="300"/>
    </bpmndi:BPMNShape>
    <bpmndi:BPMNEdge bpmnElement="_ee2b96f7-e111-48df-817f-
27add1f357b5">
      <di:waypoint y="350" x="285"/>
      <di:waypoint y="350" x="310"/>
      <di:waypoint y="350" x="310"/>
      <di:waypoint y="350" x="335"/>
    </bpmndi:BPMNEdge>
    <bpmndi:BPMNShape bpmnElement="_156a6a6b-93aa-4085-90b8-
31f03570e8e8">
      <dc:Bounds height="100" width="150" x="135" y="300"/>
    </bpmndi:BPMNShape>
    <bpmndi:BPMNShape bpmnElement="_23f75b36-9ddd-4ae6-809b-
8ed645984a56">
      <dc:Bounds height="100" width="150" x="335" y="300"/>
    </bpmndi:BPMNShape>
    <bpmndi:BPMNEdge bpmnElement="_f033b9f4-e3c7-48f2-bb64-
ffe1b9fcb25d">
      <di:waypoint y="350" x="485"/>

```

```

    <di:waypoint y="350" x="510"/>
    <di:waypoint y="350" x="510"/>
    <di:waypoint y="350" x="535"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_23f75b36-9ddd-4ae6-809b-
8ed645984a56">
    <dc:Bounds height="100" width="150" x="335" y="300"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNShape bpmnElement="_e31209c7-daf0-42cd-b258-
2ee304e941b3">
    <dc:Bounds height="50" width="50" x="535" y="325"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_7b45c101-60ad-47a3-b521-
b09743a0c34f">
    <di:waypoint y="350" x="585"/>
    <di:waypoint y="350" x="637"/>
    <di:waypoint y="213" x="637"/>
    <di:waypoint y="213" x="689"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_e31209c7-daf0-42cd-b258-
2ee304e941b3">
    <dc:Bounds height="50" width="50" x="535" y="325"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNShape bpmnElement="_9a7e78d2-7c9f-412e-98f5-
970bc627cfa9">
    <dc:Bounds height="100" width="150" x="689" y="163"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_7076d516-c357-4f9b-86bc-
ccf6e1544621">
    <di:waypoint y="213" x="839"/>
    <di:waypoint y="213" x="864"/>
    <di:waypoint y="213" x="864"/>
    <di:waypoint y="213" x="889"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_9a7e78d2-7c9f-412e-98f5-
970bc627cfa9">
    <dc:Bounds height="100" width="150" x="689" y="163"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNShape bpmnElement="_68434e2d-ec6e-43d5-a11b-
6e5e2e92ad44">
    <dc:Bounds height="50" width="50" x="889" y="188"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_57e2a9c8-de5b-4b38-b91b-
8746d2720f57">
    <di:waypoint y="238" x="914"/>
    <di:waypoint y="443" x="914"/>
    <di:waypoint y="443" x="839"/>
    <di:waypoint y="443" x="839"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_68434e2d-ec6e-43d5-a11b-
6e5e2e92ad44">
    <dc:Bounds height="50" width="50" x="889" y="188"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNShape bpmnElement="_1d6d1924-55dc-4ec9-bf89-
bf3b6a3c0181">
    <dc:Bounds height="100" width="150" x="689" y="389"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_c1b8350b-c97c-43c8-b1d8-
aec9440bcedc">
    <di:waypoint y="439" x="689"/>
    <di:waypoint y="439" x="764"/>

```

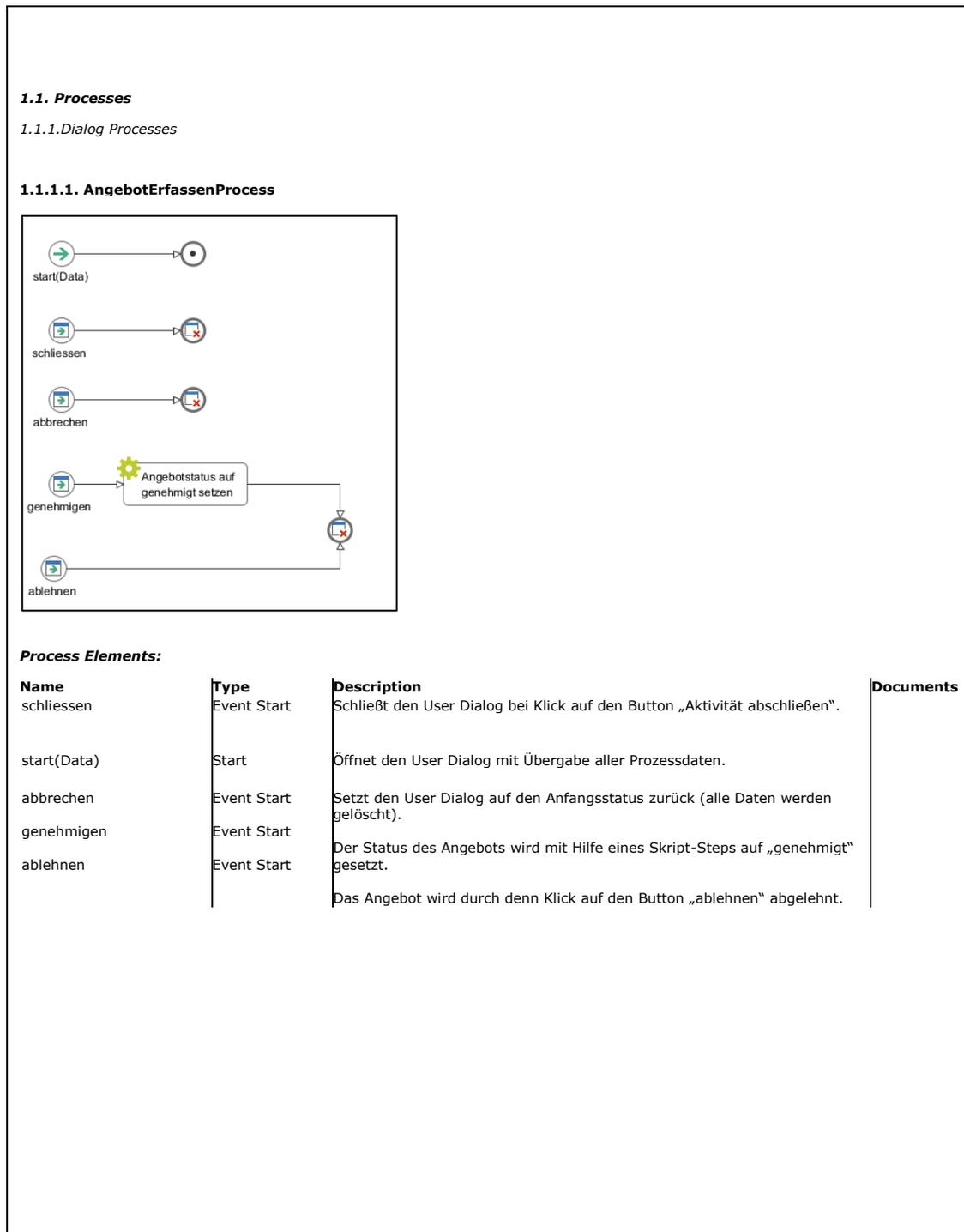
```

    <di:waypoint y="439" x="560"/>
    <di:waypoint y="375" x="560"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_1d6d1924-55dc-4ec9-bf89-
bf3b6a3c0181">
    <dc:Bounds height="100" width="150" x="689" y="389"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_00d48138-5963-4466-b11a-
3a0ec0bd75a0">
    <di:waypoint y="213" x="939"/>
    <di:waypoint y="213" x="961"/>
    <di:waypoint y="213" x="962"/>
    <di:waypoint y="213" x="984"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_68434e2d-ec6e-43d5-a11b-
6e5e2e92ad44">
    <dc:Bounds height="50" width="50" x="889" y="188"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNShape bpmnElement="_4cb045ab-11ee-4579-9828-
043bd4fa035d">
    <dc:Bounds height="100" width="150" x="984" y="163"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNEdge bpmnElement="_49d6dc68-9740-4950-a17c-
072a777fe900">
    <di:waypoint y="213" x="1134"/>
    <di:waypoint y="213" x="1162"/>
    <di:waypoint y="213" x="1162"/>
    <di:waypoint y="213" x="1190"/>
  </bpmndi:BPMNEdge>
  <bpmndi:BPMNShape bpmnElement="_4cb045ab-11ee-4579-9828-
043bd4fa035d">
    <dc:Bounds height="100" width="150" x="984" y="163"/>
  </bpmndi:BPMNShape>
  <bpmndi:BPMNShape bpmnElement="_e124db05-23d0-4ada-b8a0-
db8dd4df2aa3">
    <dc:Bounds height="50" width="50" x="1190" y="188"/>
  </bpmndi:BPMNShape>
</bpmndi:BPMNPlane>
</bpmndi:BPMNDiagram>
</definitions>

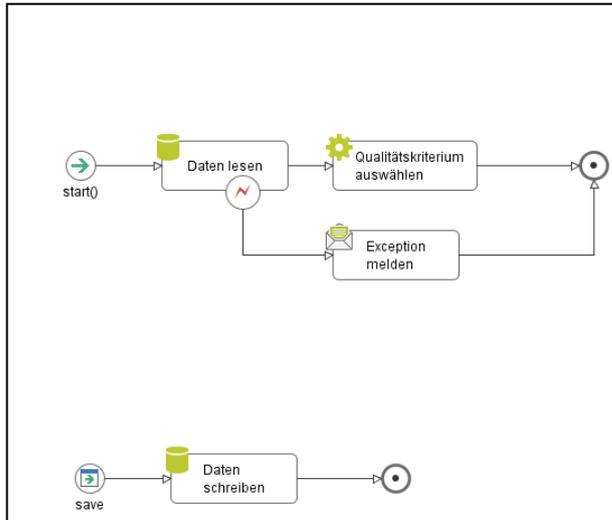
```

Anhang 2: Dokumentation Anwendungsbeispiel Geschäftsprozessapplikation

Die Dokumentation wurde automatisiert mit Hilfe in den Axon.ivy Designer integrierten Report Generators erstellt. Der Anhang 2 beinhaltet das Geschäftsprozessmodell in Version 2, die funktionalen technischen Prozesse (z.B. zum Aufruf eines User Dialogs), Datenobjekte, die Dokumentation der Datenbankverbindung und verwendeten Komponenten.



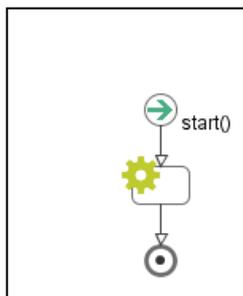
1.1.1.2. FeedbackAppProcess



Process Elements:

Name	Type	Description	Documents
Daten schreiben	Database	Schritt zum Speichern des Feedbacks.	
Daten lesen	Database	Schritt zum Lesen der Qualitätskriterien aus der Datenbank.	
Exception melden	E-Mail	E-Mail-Step zum Versenden des Exceptionberichts.	
Qualitätskriterium auswählen	Script	Logik zur Auswahl des Qualitätskriteriums.	
save	Event Start	Bei Klick auf den Feedback-Button wird das Feedback gespeichert.	
start()	Start	Lädt den Feedback-Dialog.	

1.1.1.3. ProcessChainProcess

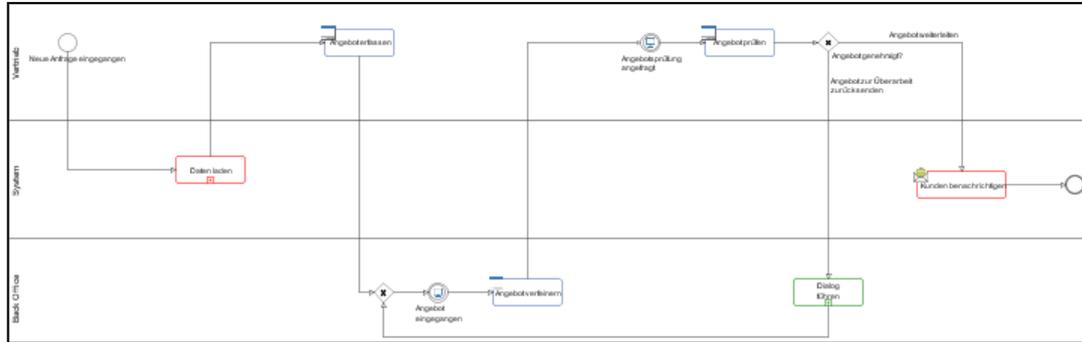


Process Elements:

Name	Type	Description	Documents
start()	Start	Skript-Step zur Initialisierung der ProcessChain-Komponente. Öffnet die ProcessChain zur Anzeige im Dialog.	

1.1.2. Business Processes

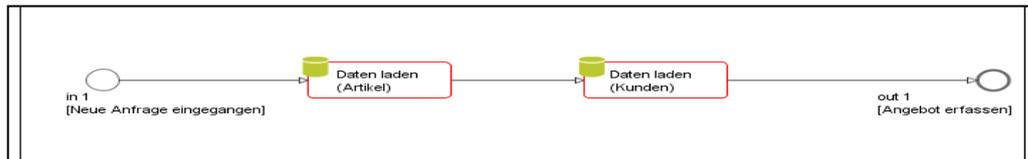
1.1.2.1. Angebotsprozess_v2



Process Elements:

Name	Type	Description	Documents
Angebot eingegangen	Alternative	Dient zur Zusammenführung zweier Sequenzflüsse.	
Angebot erfassen	Simple Task Switch	Erzeugt eine Aufgabe (Task), welche dem Anwender im Axon.ivy Prozessportal angezeigt wird.	
Angebot genehmigt?	JSF Dialog	User Dialog zur Erfassung der Angebotsinformationen.	
Angebot prüfen	Alternative	Auswertung der Variable „AngebotGenehmigt“.	
Angebot verfeinern	JSF Dialog	User Dialog zur Prüfung (genehmigen/ablehnen) des Angebots.	
Angebotsprüfung angefragt	JSF Dialog	User Dialog zur Ergänzung von Informationen durch das Back Office.	
Daten laden	Simple Task Switch	Erzeugt eine Aufgabe (Task), welche dem Anwender im Axon.ivy Prozessportal angezeigt wird.	
Dialog führen	Embedded Sub Process	Funktionaler Prozess zum Lesen von Daten aus der Datenbank (Kunden/Artikel).	
Kunden benachrichtigen	Embedded Sub Process	Dient zur Kapselung des dialogischen Logik-Patterns.	Dokumentation des Patterns
Neue Anfrage eingegangen	E-Mail	E-Mail-Step zum Versenden des Angebots an den Kunden.	
	Request Start	Start einer neuen Instanz.	

1.1.2.1.1. Daten laden



Process Elements:

Name	Type	Description	Documents
Daten laden (Artikel)	Database	Verbindung zur Datenbank und Abruf der Daten (MySQL).	
Daten laden (Kunden)	Database	Verbindung zur Datenbank und Abruf der Daten (MySQL).	

1.2. Data Classes

1.2.1.1. Angebot

Attributes

Name	Type	Comment
SelektierteArtikel	List<angebotsprozess.Artikel>	Liste der selektierten Artikel
SelektierterKunde	String	Speicherung des ausgewählten Kunden
Kommentar	String	
Datum	Date	

1.2.1.2. Artikel

Attributes

Name	Type	Comment
Id	Number	Eindeutige Identifikationsnummer eines Artikels
Artikelbeschreibung	String	Beschreibung eines Artikels

1.2.1.3. Data

Attributes

Name	Type	Comment
Angebot	angebotsprozess.Angebot	
Kundenliste	List<angebotsprozess.Kunde>	
Artikelliste	List<angebotsprozess.Artikel>	
AngebotGenehmigt	Boolean	Hilfsparameter, welcher im XOR Gateway ausgewertet wird.

1.2.1.4. Kunde

Attributes

Name	Type	Comment
Id	Number	
Kundenbeschreibung	String	

1.3. Roles

Identifier	Name	Parent Role
Everybody		Everybody
Backoffice		Everybody
Vertrieb		Everybody

1.4. Databases

Name	Product	Driver	Environment	Max. connections
anwendungsbeispiel	MySQL	mySQL	Default	5

Host: localhost
Database Name: anwendungsbeispiel
User: root