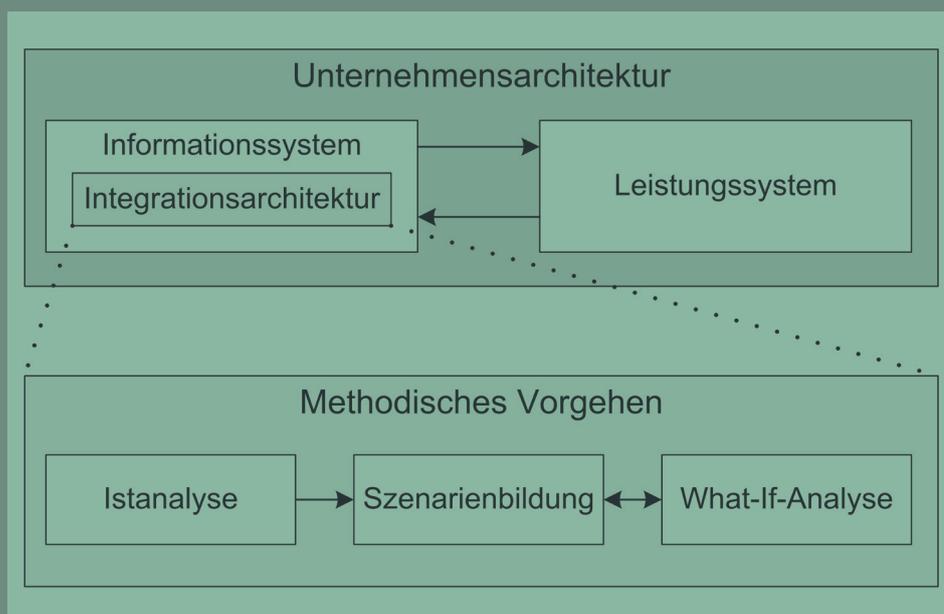


Ermittlung aufgabenbezogener Integrationsbedarfe zur Ableitung von Integrationsarchitekturen für Informationssysteme

Florian Bader



35 Schriften aus der Fakultät Wirtschaftsinformatik
und Angewandte Informatik der Otto-Friedrich-
Universität Bamberg

Contributions of the Faculty Information Systems
and Applied Computer Sciences of the
Otto-Friedrich-University Bamberg

Schriften aus der Fakultät Wirtschaftsinformatik
und Angewandte Informatik der Otto-Friedrich-
Universität Bamberg

Contributions of the Faculty Information Systems
and Applied Computer Sciences of the
Otto-Friedrich-University Bamberg

Band 35

Ermittlung aufgabenbezogener Integrationsbedarfe zur Ableitung von Integrationsarchitekturen für Informationssysteme

Florian Bader

Bibliographische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliographie; detaillierte bibliographische Informationen sind im Internet über <http://dnb.d-nb.de/> abrufbar.

Diese Arbeit hat der Fakultät Wirtschaftsinformatik und Angewandte Informatik der Otto-Friedrich-Universität Bamberg als Dissertation vorgelegen.

1. Gutachter: Prof. Dr. Elmar J. Sinz

2. Gutachter: Prof. Dr. Sven Overhage

Tag der mündlichen Prüfung: 25.07.2018

Dieses Werk ist als freie Onlineversion über den Publikationsserver (OPUS; <http://www.opus-bayern.de/uni-bamberg/>) der Universität Bamberg erreichbar. Das Werk – ausgenommen Cover, Zitate und Abbildungen – steht unter der CC-Lizenz CC-BY.



Lizenzvertrag: Creative Commons Namensnennung 4.0

<http://creativecommons.org/licenses/by/4.0>

Herstellung und Druck: docupoint, Magdeburg

Umschlaggestaltung: University of Bamberg Press, Larissa Günther

© University of Bamberg Press, Bamberg 2019

<http://www.uni-bamberg.de/ubp/>

ISSN: 1867-7401

ISBN: 978-3-86309-650-2 (Druckausgabe)

eISBN: 978-3-86309-651-9 (Online-Ausgabe)

URN: urn:nbn:de:bvb:473-opus4-546510

DOI: <http://dx.doi.org/10.20378/irbo-54651>

Meiner Familie

Kurzzusammenfassung

Die Bedeutung des Entwurfs von Integrationsarchitekturen im Zeitalter von zunehmender IT-Durchdringung steigt. Organisationen stehen vor der Herausforderung für ihre Informationssysteme geeignete Integrationsarchitekturen zu konzipieren und zu implementieren.

Das Informationssystem ist für Organisationen das informationsverarbeitende System. Dieses ist unter anderem für die Planung, Steuerung sowie die Kontrolle von Geschäftsprozessen zuständig. Zur Durchführung von Geschäftsprozessen werden häufig sowohl unterschiedliche maschinelle, insbesondere Anwendungssysteme, als auch personelle Aufgabenträger eingesetzt. Dies führt dazu, dass Ressourcenebenen von Geschäftsprozessen meist heterogen geprägt sind. Damit entsteht das Problem, geeignete Integrationsarchitekturen zu konstruieren und dabei die relevanten Integrationsmerkmale der Ressourcenebene mit den spezifischen Anforderungen der Geschäftsprozesse geeignet zu kombinieren.

Für die Erschließung dieses Problemfeldes bietet die vorliegende Arbeit zunächst einen Überblick über bestehende Ansätze zum Entwurf von Integrationsarchitekturen für Informationssysteme. Bei der ersten Annäherung an das Thema ist schnell festzustellen, dass bei der Integration der Fokus häufig auf die Technologie gerichtet wird, insbesondere auf die Erstellung von Schnittstellen für die beteiligten Anwendungssysteme. Darüber hinaus wird auf Ressourcenebene neben Anwendungssystemen der personelle Anteil bei der Konstruktion von Integrationsarchitekturen oftmals unzureichend berücksichtigt. Zur Vermeidung dieser einengenden Sichtweise wird deshalb im Rahmen dieser Arbeit ein Top-Down-Ansatz entworfen. Bei diesem werden ausgehend vom Geschäftsprozess Integrationsbedarfe identifiziert und mittels Merkmalen die Anforderungen an die Integration aus Sicht der involvierten Aufgaben spezifiziert. Unter geeigneter Kombination mit der Ressourcenebene können verschiedene Szenarien miteinander verglichen werden. Grundlage des entwickelten Top-Down-Ansatzes ist das Semantische Objektmodell. Die Anwendung wird anhand von Fallstudien aus dem Hochschulbereich sowie der Güterdistribution verdeutlicht und mithilfe eines entwickelten Prototypen unterstützt.

Die vorliegende Arbeit richtet sich an Systemgestalter. Sie sollen beim Entwurf von Integrationsarchitekturen für Informationssysteme von Geschäftsprozessen durch den entwickelten Ansatz methodisch unterstützt werden und dabei Kenntnisse über die bestehenden Integrationsbeziehungen sowie Konsequenzen der Integrationsentscheidungen erlangen.

Vorwort

Die vorliegende Arbeit entstand hauptsächlich im Rahmen meiner damaligen Tätigkeit als Projektmitarbeiter der Abteilung 1 am ehemaligen Wissenschaftlichen Institut für Hochschulsoftware der Universität Bamberg (ihb).

Für die persönliche Betreuung, wertvolle Unterstützung sowie konstruktiven Hinweise bei der Erstellung der Arbeit gilt mein besonderer Dank meinem Doktorvater Herrn Prof. Dr. Elmar J. Sinz. Für die Übernahme des Zweitgutachtens danke ich Herrn Prof. Dr. Sven Overhage recht herzlich. Bei Herrn Prof. Dr. Guido Wirtz möchte ich mich für die Begleitung der Arbeit als Mitglied der Promotionskommission bedanken.

Ein besonderer Dank gilt dem FlexNow-Team für die jahrelange erfolgreiche sowie freundschaftliche Zusammenarbeit. Ein Dankeschön gilt in diesem Zusammenhang ebenfalls dem gesamten Team des Lehrstuhls für Wirtschaftsinformatik, insbesondere Systementwicklung und Datenbankanwendung, sowie der Abteilung 2 am ihb für die wertvollen Diskussionsbeiträge. Für die zahlreichen fachlichen Diskussionen sowie konstruktiven Impulse, die zum Gelingen dieser Arbeit beitrugen, möchte ich mich ganz besonders bei Frau Simone Wismer und bei Herrn Tobias Kiehl bedanken. Für die eingehende Durchsicht meines Manuskripts bedanke ich mich recht herzlich bei Frau Silke Schmitz.

An dieser Stelle möchte ich mich auch bei meinen Eltern und Geschwistern bedanken, die mich in meinem Vorhaben stets ermutigt und gefördert haben. Ein besonderer persönlicher und herzlicher Dank gilt meiner Frau Nikolina Bader für das entgegengebrachte Verständnis sowie die Motivation und Unterstützung während der Anfertigung dieser Arbeit. Sie alle gaben mir den erforderlichen Rückhalt, ohne den ein erfolgreicher Abschluss dieser Arbeit nicht möglich gewesen wäre.

Mering, im April 2019

Florian Bader

Inhaltsverzeichnis

Fachliches Abkürzungsverzeichnis	xiii
Abbildungsverzeichnis	xv
Tabellenverzeichnis	xvii
1 Einleitung	1
1.1 Problemstellung und Motivation	2
1.2 Untersuchungssituation	4
1.3 Überblick über Lösungsansätze für die Integration	7
1.4 Charakterisierung des Lösungsansatzes	7
1.5 Entstehung, Umfeld und Adressaten der Arbeit	8
1.6 Aufbau der Arbeit	9
1.7 Konventionen der Arbeit	10
2 Die Unternehmensarchitektur als Ordnungsrahmen	13
2.1 Der Begriff der Unternehmensarchitektur am Beispiel der Methodik des semantischen Objektmodells	13
2.2 Der Begriff der Aufgabe	17
2.2.1 Der Prozessbegriff	18
2.2.2 Automatisierung von Aufgaben	20
2.3 Das betriebliche Informationssystem im Kontext der Unternehmensar- chitektur	21
2.4 Einführung des erläuternden Fallbeispiels	24
3 Integration als Teilaufgabe bei der Gestaltung von Informationssyste- marchitekturen	27
3.1 Integration von Aufgaben über Aufgabenträger	28
3.1.1 Das Analyse-Synthese-Konzept	28
3.1.2 Der Begriff der Heterogenität	30
3.1.3 Aufgabenintegrationsmuster zur Analyse der Aufgabenvernet- zung	31
3.1.4 Das Konzept des betrieblichen Objekts zur Bildung von Auf- gabenkomplexen	33
3.1.5 Kartografie zur Analyse von Abhängigkeiten zwischen An- wendungssystemen	34

3.1.6	Technische Lösungsverfahren zur Herstellung der Interoperabilität	38
3.1.7	Die Integrationsarchitektur als Teil des Informationssystems	39
3.2	Motivation für Integrationsprojekte	42
3.2.1	Auslöser von Integrationsprojekten	42
3.2.2	Ziele bei Integrationsprojekten	43
3.3	Integrationsmerkmale und Lösungskonzepte bei der Gestaltung von Integrationsarchitekturen	46
3.3.1	Integrationsmerkmale bei der Gestaltung von Integrationsarchitekturen	46
3.3.1.1	Redundanz	47
3.3.1.2	Verknüpfung	48
3.3.1.3	Konsistenz	48
3.3.1.4	Zielorientierung	49
3.3.1.5	Bezug der Integrationsziele auf Aufgaben- und Aufgabenträgerebene	49
3.3.2	Integrationskonzepte zur Gestaltung von Integrationsarchitekturen	50
3.3.2.1	Aufgabenträgerorientierte Funktionsintegration	50
3.3.2.2	Datenflussorientierte Funktionsintegration	51
3.3.2.3	Datenintegration	51
3.3.2.4	Objektintegration	52
3.3.2.5	Zielerreichungsgrade der Integrationskonzepte	53
3.4	Konzeption von Integrationsarchitekturen als Systementwicklungsaufgabe	54
3.4.1	Aufgabenmodell der Systementwicklung	54
3.4.2	Beschreibungsebenen bei der Entwicklung von Integrationsarchitekturen	54
4	Ansätze zum Entwurf von Integrationsarchitekturen für Informationssysteme	57
4.1	Strategie und Durchführung der Suche nach Ansätzen	57
4.2	Deskription und Interpretation der Datengrundlage	61
4.3	Kriterien zur Evaluation von Ansätzen	62
4.4	Analyse und Bewertung gefundener Ansätze	64
4.4.1	Quasar Enterprise	64
4.4.2	Beitrag von Gronau	67
4.4.3	Der Ansatz von Jung	69
4.4.4	Der Ansatz von Kattenstroth	73
4.4.5	Der Ansatz von Schwinn	75

4.4.6	Der Ansatz von Vogler	79
4.5	Zusammenfassende Bewertung	83
5	Merkmale als Hinweisgeber zur Gestaltung der Integration von Aufgaben über Aufgabenträger	85
5.1	Untersuchungsstrategie bei der Merkmalsanalyse	86
5.2	Strategie und Durchführung der Suche nach Literatur zu Merkmalen .	86
5.3	Ergebnisse der Literaturanalyse	90
5.4	Bildung des Merkmalskataloges	92
5.4.1	Selektion der durch die Literaturrecherche gefundenen Merkmale	92
5.4.2	Bildung des Kategoriensystems	93
5.4.3	Bildung des initialen Merkmalskataloges	93
5.4.4	Reduktion der Merkmale	93
5.4.4.1	Anforderungen an die Zuverlässigkeit	94
5.4.4.2	Kontext	94
5.4.4.3	Leistungsanforderungen	95
5.4.4.4	Flexibilitätsanforderungen	95
5.4.5	Finaler Merkmalskatalog	96
5.4.5.1	Periodizität des Geschäftsprozesses	97
5.4.5.2	Größe einer zu übertragenden Aufgabenobjekt-Instanz	98
5.4.5.3	Intensität bei der Durchführung einer Transaktion .	99
5.4.5.4	Anpassbarkeit der Transaktion	99
5.4.5.5	Nutzungsdauer der Integrationsarchitektur	101
5.5	Handlungsempfehlungen für die Ausprägungen der Merkmale	102
5.6	Zusammenfassung der Ergebnisse der Merkmalsanalyse	107
6	Entwicklung einer Methode zum Entwurf von Integrationsarchitekturen für Informationssysteme	109
6.1	Metamodell der Methode	110
6.2	Architekturmodell der Methode	113
6.2.1	Metamodell für das Repräsentationsobjekt Geschäftsprozess .	113
6.2.2	Metamodell für das Repräsentationsobjekt Aufgabenträgerzuordnung	114
6.3	Vorgehensmodell der Methode	116
6.3.1	Istanalyse	116
6.3.1.1	Spezifikation des zu analysierenden Geschäftsprozesses	116
6.3.1.2	Erhebung und Analyse des Ist-Zustands	121
6.3.1.3	Hinweise für die Gruppierung von Transaktionen mit Inter-AWS-Kommunikationskanälen	125

6.3.2	Szenarienbildung	126
6.3.2.1	Szenarioorientierte Gestaltung des Geschäftsprozesses	127
6.3.2.2	Erhebung der Freiheitsgrade beim Entwurf der Integrationsarchitektur	128
6.3.2.3	Ermittlung der Soll-Automatisierung für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen	132
6.3.2.4	Ableitung der zu verwendenden Integrationskonzepte für die Inter-AwS-Integrationen	136
6.3.2.5	Optionen bei Unvereinbarkeit zwischen angestrebten und möglichen Integrationskonzept	142
6.4	Techniken und Werkzeuge für die Methodendurchführung	143
6.5	Zusammenfassung und Diskussion	146
7	Konzeption einer Integrationsarchitektur für das Informationssystem eines Güterdistributors	151
7.1	Istanalyse für ein existierendes Güterdistributionssystem	152
7.1.1	Spezifikation des zu analysierenden Geschäftsprozesses	152
7.1.2	Erhebung und Analyse des Ist-Zustands	154
7.2	Szenario: Einführung eines ERP-Systems	160
7.2.1	Szenarioorientierte Gestaltung des Geschäftsprozesses	160
7.2.2	Erhebung der Freiheitsgrade beim Entwurf der Integrationsarchitektur	160
7.2.2.1	Ebene des Unternehmensplans	161
7.2.2.2	Geschäftsprozessebene	161
7.2.2.3	Ressourcenebene	163
7.2.3	Ermittlung der Soll-Automatisierung für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen	164
7.2.4	Ableitung der zu verwendenden Integrationskonzepte für die Inter-AwS-Integrationen	167
7.2.4.1	Festlegung der Integrationsziele für das Szenario	167
7.2.4.2	Ermittlung der Integrationskonzepte für die Inter-AwS-Integrationen	167
7.3	What-If-Analyse für den Güterdistributor	171
7.4	Zusammenfassung und Diskussion	172
8	Zusammenfassung und Ausblick	173
8.1	Zusammenfassung der Ergebnisse	173
8.2	Schlussbetrachtung und Ausblick	175

A	Liste aller gefundenen Qualitätsattribute	179
B	Ergänzende Protokolle für das erläuternde Fallbeispiel	185
B.1	Protokoll der Objekt- und Transaktionszerlegung für das erläuternde Fallbeispiel im Rahmen der Istanalyse	185
B.2	Protokoll der Objekt- und Transaktionszerlegung für das erläuternde Fallbeispiel im Rahmen der Szenariobildung	185
C	Ergänzende Informationen für die Fallstudie Güterdistribution	187
C.1	Protokoll der Objekt- und Transaktionszerlegung für die Fallstudie Güterdistribution	187
C.2	Verhaltenssicht auf den Geschäftsprozess der Fallstudie	188
C.3	Transaktionen, zugeordnete betriebliche Objekte sowie identifizierte AIM und überlappende Aufgabenobjekte	188
C.4	Erfasste Attribute für die Erstellung des Sollkonzepts	194
	Literaturverzeichnis	197

Fachliches Abkürzungsverzeichnis

ADK	Anwendungs-, Datenhaltungs- und Kommunikationsteil
AIM	Aufgabenintegrationsmuster
AIS	Association for Information Systems Research
AO	Aufgabenobjekt
AS	Aktionensteuerung
AT	Aufgabenträger
AwS	Anwendungssystem
BASE	Bielefeld Academic Search Engine
BPMN	Business Process Model and Notation
COCOMO	Constructive Cost Model
CORBA	Common Object Request Broker Architecture
DNB	Deutsche Nationalbibliothek
DV	Datenverarbeitung
EAI	Enterprise Application Integration
EAM	Enterprise Architecture Management
EDI	Electronic Data Interchange
ER	Entity-Relationship
ERP	Enterprise Resource Planning
GP	Geschäftsprozess
HTTP(S)	Hypertext Transfer Protocol (Secure)
IAS	Interaktionsschema
IEEE	Institute of Electrical and Electronics Engineers
ihb	Wissenschaftliches Institut für Hochschulsoftware der Universität Bamberg

IT	Informationstechnologie
JDBC	Java Database Connectivity
KOS	Konzeptuelles Objektschema
KOT	Konzeptueller Objekttyp
LVS	Lagerverwaltungssystem
MDM	Master Data Management
MEMO	Multiperspektivische Unternehmensmodellierung
MVC	Model-View-Controller
OASYS	Offene Anwendungssystem-Architekturen in überbetrieblichen Wertschöpfungsketten
ODBC	Open Database Connectivity
ORM	Object-relational mapping
REST	Representational State Transfer
RFID	Radio-Frequency Identification
RMI	Remote Method Invocation
SERM	Strukturiertes Entity-Relationship-Modell
SOA	Serviceorientierte Architekturen
SOM	Semantische Objektmodell
SQL	Structured Query Language
UML	Unified Modeling Language
VA	Vorgangsauslösung
VES	Vorgangs-Ereignis-Schema
VOS	Vorgangsobjektschema
VOT	Vorgangsobjekttyp
WfMS	Workflow-Management-Systeme
WPF	Windows Presentation Foundation
WSDL	Web Services Description Language

Abbildungsverzeichnis

1.1	Gestaltung der Beziehung Menschen-Aufgaben-Technik	3
1.2	Untersuchungssituation zur Entwicklung einer Methode bei der Identifikation von Integrationsbedarfen	5
1.3	Aufbau der Arbeit	11
2.1	Generischer Architekturrahmen	14
2.2	Unternehmensarchitektur der SOM-Methodik	16
2.3	Struktur einer Aufgabe	17
2.4	Informationsbeziehungen und Kommunikationssysteme im Informationssystem	22
2.5	Haupt- und ausgewählte Serviceprozesse der Hochschule in der Leistungssicht	25
2.6	Eingrenzung der Untersuchung auf die Teilprozesse Semesterausbildung und Prüfung in der ausführlichen Lenkungssicht	26
3.1	Aufgaben und ihre Beziehungen	32
3.2	Schematische Darstellung des objektorientierten Konzepts betrieblicher Objekte	34
3.3	Beispiel einer Anwendungssystemzuordnungssicht	37
3.4	Die Integrationsarchitektur als Bestandteil der Unternehmensarchitektur	40
3.5	Typ- bzw. Instanzbezug der Integrationsziele	49
3.6	Aufgabenmodell bei der Gestaltung der Integration als Systementwicklungsaufgabe	55
3.7	Beschreibungsebenen bei der Gestaltung von Integrationsarchitekturen	55
4.1	Quasar Enterprise Framework	65
4.2	Einfluss von Komponenten der Anwendungsarchitektur auf die Merkmale des qualitativen Informationsbedarfs	70
4.3	Informationsmodell für die Vorstudie	71
4.4	Integriertes Vorgehensmodell für AwS-Integrationsprojekte	76
4.5	Implementierung fachlicher Muster durch DV-konzeptionelle Muster .	77
5.1	Unterstützung bei der Ermittlung der Handlungsempfehlung durch den entwickelten Prototypen	106
6.1	Meta-Metamodell	110
6.2	Aggregiertes Metamodell der Methode	111
6.3	Metamodell der Methode	112
6.4	Metamodell der Geschäftsprozesssicht	114
6.5	Metamodell der Aufgabenträgerzuordnungssicht	115

6.6	Symbole zur Automatisierbarkeit und Automatisierung von Aufgaben und Transaktionen	115
6.7	Schematische Darstellung des Vorgehensmodells der entwickelten Methode zum Entwurf von Integrationsarchitekturen für Informationssysteme	117
6.8	Initiales IAS für den untersuchten Teilprozess	118
6.9	Instanzierte Struktursicht auf den Ist-GP des Fallbeispiels	120
6.10	Instanziertes Modell für die Aufgabenträgerzuordnungssicht	124
6.11	Aufgabenträgerzuordnungssicht auf den GP des Szenarios	129
6.12	Soll-Konzept für das Szenario des Fallbeispiels	135
6.13	Datenmodell des unterstützenden Prototypen als SERM	145
7.1	Initialer GP für das Beispiel der Güterdistribution	152
7.2	Struktursicht auf den Ist-GP der Fallstudie	153
7.3	Abbildung von betrieblichen Objekten, Transaktionen, überlappenden AO sowie identifizierter AIM mithilfe des Prototypen	154
7.4	Zuordnung der AT mithilfe des Prototypen	156
7.5	Gruppierung von Transaktionen mit Inter-AwS-Kommunikationskanälen mithilfe des Prototypen	157
7.6	Aufgabenträgerzuordnung in der Lenkungssicht des untersuchten GP im Rahmen der Istanalyse	159
7.7	Erstellung eines Sollkonzepts für die Transaktionen mit Inter-AwS-Kommunikationskanälen mithilfe des Prototypen	165
7.8	Integrationsarchitektur für das Szenario	166
7.9	Ermittlung der Integrationskonzepte für die Inter-AwS-Integrationen mithilfe des Prototypen	169
7.10	Ergebnis der What-If-Analyse für den Güterdistributor	171
C.1	Verhaltenssicht auf den Ist-GP der Fallstudie	189

Tabellenverzeichnis

2.1	Klassifikation von Aufgaben hinsichtlich ihrer Automatisierbarkeit, Automatisierungsforderung und Realisierung	20
3.1	Integrationsmerkmale	47
3.2	Zielerreichungsgrade der Integrationskonzepte (Ferstl und Sinz 2013, S. 243-251; Mantel et al. 2001, S. 6-9)	53
4.1	Literaturrecherche nach vorhandenen Ansätzen in den verschiedenen Phasen der Untersuchung	60
4.2	Finale Auswahl an Ansätzen	60
4.3	Evaluation des Ansatzes von Engels und Voß (2008)	65
4.4	Evaluation des Ansatzes von Gronau et al. (2008)	68
4.5	Morphologischer Kasten zu Konstruktionsmerkmalen von Integrationsarchitekturen	70
4.6	Evaluation des Ansatzes von Jung (2006)	72
4.7	Evaluation des Ansatzes von Kattenstroth et al. (2013)	74
4.8	Evaluation des Ansatzes von Schwinn (2005)	78
4.9	Evaluation des Ansatzes von Vogler (2006)	81
4.10	Gegenüberstellung der Ansätze	83
5.1	Verwendete Suchbegriffe zur Findung von Qualitätsattributen	88
5.2	Anzahl der Beiträge während den verschiedenen Phasen der Literaturrecherche	89
5.3	Finale Auswahl an Literatur	90
5.4	Kategorien für die Merkmale und deren Herkunft	91
5.5	Übersicht über alle initial gefundenen Merkmale	94
5.6	Merkmale, anhand derer Anforderungen an die Integration aus Sicht der Aufgabe spezifiziert werden können	97
5.7	Eigenschaften von Standard-, Routine- und Nicht-Routinetransaktionen	100
5.8	Vorlage für die Erhebung der Ausprägungen für jedes Merkmal	102
6.1	Überblick über das Architekturmodell	113
6.2	Hilfstabelle zur Dokumentation der Transaktionen, Transaktionsgruppen, zugeordneter betrieblicher Objekte sowie identifizierte AIM und überlappende AO	119
6.3	Hilfstabelle zur Zuordnung der AT zu den betrieblichen Objekten	122
6.4	Hilfstabelle zur Dokumentation der Automatisierungsgrade von Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen	123

6.5	Hilftabelle zur Dokumentation der Inter-AwS-Integrationen und die dabei verwendeten Integrationskonzepte	125
6.6	Betriebliche Objekte und die ihnen zugeordneten AT im Szenario des Fallbeispiels	127
6.7	Hilftabelle zur Dokumentation des Sollkonzepts für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen . .	133
6.8	Zusammenfassung der freiheitsgradbeschränkenden Faktoren	134
6.9	Hilftabelle zur Dokumentation der zu erreichenden Integrationsziele für das Szenario	137
6.10	Hilftabelle zur Wahl des Integrationskonzepts	138
6.11	Integrationskonzepte und ihre Auswirkungen	139
6.12	Abgeleitetes Integrationskonzept für das Szenario des Fallbeispiels . .	141
6.13	Überblick der Bestandteile der entwickelten Methode	147
6.14	Evaluation der entwickelten Methode	147
6.15	Zusammenfassende Bewertung der Ansätze	149
7.1	Betriebliche Objekte und die ihnen zugeordneten AT im Rahmen der Istanalyse	155
7.2	Integrierte AwS und die dabei verwendeten Integrationskonzepte im Rahmen der Istanalyse	158
7.3	Betriebliche Objekte und die ihnen zugeordneten AT im Szenario: Einführung eines ERP-Systems	161
7.4	Ausprägung der Merkmale für die Transaktion <i>V: AuffPrf</i>	163
7.5	Festlegung der Integrationsziele für das Szenario	168
7.6	Abgeleitete Integrationskonzepte für das Szenario	170
A.1	Rohtabelle mit allen Merkmalen	179
B.1	Protokoll der Objekt- und Transaktionszerlegung für das erläuternde Fallbeispiel im Rahmen der Istanalyse	185
B.2	Protokoll der Objekt- und Transaktionszerlegung für das erläuternde Fallbeispiel im Rahmen der Szenariobildung	186
C.1	Protokoll der Objekt- und Transaktionszerlegung für das Beispiel der Güterdistribution	187
C.2	Automatisierungsgrade der Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen	190
C.3	Transaktionen, Transaktionsgruppen, zugeordnete betriebliche Objekte sowie identifizierte AIM und überlappende AO	191
C.4	Sollkonzept für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen im Szenario	195

1 Einleitung

1.1 Problemstellung und Motivation	2
1.2 Untersuchungssituation	4
1.3 Überblick über Lösungsansätze für die Integration	7
1.4 Charakterisierung des Lösungsansatzes	7
1.5 Entstehung, Umfeld und Adressaten der Arbeit	8
1.6 Aufbau der Arbeit	9
1.7 Konventionen der Arbeit	10

Das als Gestaltung von Integrationsarchitekturen für Informationssysteme bezeichnete Teilgebiet der Wirtschaftsinformatik ist ein Forschungsgebiet von zentraler Bedeutung (Mertens et al. 2005, S. 6). Für dieses stellt die Wirtschaftsinformatik konzeptionelle Lösungsansätze, die sich mit der Verknüpfung von Menschen, Aufgaben und Technik zu einer Einheit befassen, bereit (Mertens 2013a, S. 13). Allgemein wird unter dem Begriff der Integration als Vorgang die Herstellung einer Einheit aus Teilen sowie die Einbeziehung eines Teils in ein größeres Ganzes verstanden. Darüber hinaus kann als Integration auch der abschließende Zustand nach einem Integrationsvorgang bezeichnet werden (Dudenredaktion 2008, S. 464). Im Fokus dieser Arbeit steht dabei das betriebliche Informationssystem, welches Geschäftsprozesse, Anwendungssysteme sowie Personen umfasst und dessen Gegenstand die Informationsverarbeitung ist.

Im Rahmen dieser Arbeit wird das Ziel verfolgt, einen pragmatischen Ansatz zur Ermittlung von Integrationsbedarfen zur darauf aufbauenden Ableitung geeigneter Integrationsarchitekturen für Informationssysteme¹ von Geschäftsprozessen zu entwickeln. Im Speziellen werden die Gestaltung der Aufgabenträgerebene² sowie die Abbildung des Übergangs zwischen Aufgaben thematisiert. Dadurch soll dem Systemgestalter eine methodische Unterstützung geboten werden, um ausgehend von den Anforderungen der Aufgaben an die Vernetzung eine geeignete Integrationsarchitektur gestalten zu können. Aus diesem Grund werden im vorliegenden Kapitel zunächst in Abschnitt 1.1 die Problemstellung sowie die mit dieser Arbeit zu untersuchende Hypothese vorgestellt. Darauf aufbauend werden in Abschnitt 1.2 das Forschungsziel

¹Der Begriff Informationssystem und wie er in der vorliegenden Arbeit verstanden werden soll, wird in Abschnitt 2.3 entwickelt.

²Die beiden Begriffe Aufgabenträgerebene sowie Ressourcenebene sind in der vorliegenden Arbeit synonym aufzufassen.

definiert und das zu bearbeitende Themengebiet abgegrenzt. In Abschnitt 1.3 wird ein kurzer Überblick über Arbeiten mit ähnlichen Forschungszielen sowie Forschungsgebiete mit einzelnen Anknüpfungspunkten gegeben. Anschließend wird in Abschnitt 1.4 der zu konstruierende Lösungsansatz charakterisiert sowie eine Prognose zum erwarteten Mehrwert der zu konstruierenden Methode gegeben. In Abschnitt 1.5 wird auf die Entstehung und das Umfeld der Arbeit eingegangen. Abschließend erfolgt in Abschnitt 1.6 ein Überblick über den Aufbau der Arbeit und die ihr zugrundeliegenden Konventionen (Abschnitt 1.7).

1.1 Problemstellung und Motivation

Bei der Ausführung von Geschäftsprozessen (GP) werden häufig unterschiedliche Anwendungssysteme (AwS)³ eingesetzt. Neben AwS sind an der Durchführung auch Personen beteiligt. Dies führt dazu, dass die Ressourcenebene von GP meist heterogen⁴ geprägt ist. Eine daraus resultierende Problemstellung, die im Rahmen dieser Arbeit untersucht wird, wird in Abb. 1.1 beispielhaft veranschaulicht.

Anhand der Abb. 1.1 ist erkennbar, dass im oberen Bereich die Aufgabe *Erstelle und übermittle Lehrveranstaltungsangebote* von einer anderen Person durchgeführt wird, als die Aufgabe *Erstelle Prüfungsangebote auf Grundlage der gemeldeten Lehrveranstaltungsangebote*. Da die beiden Aufgaben jedoch in einer Informationsbeziehung zueinander stehen, ist diese Beziehung auch auf der Ressourcenebene als Kommunikationsbeziehung vorzusehen. Dies wird erstens durch den Kommunikationskanal zwischen den personellen Aufgabenträgern (AT), zweitens durch die Unterstützung beider Aufgaben durch das AwS FlexNow sowie drittens durch die Mensch-Maschine-Kommunikationskanäle (MMK₁, MMK₂) zwischen den personellen AT und dem AwS unterstützt.

Im Gegensatz zum oberen Bereich in Abb. 1.1 soll im unteren das AwS Virtueller Campus zur Unterstützung der Aufgabe *Erstelle und übermittle Lehrveranstaltungsangebote* eingeführt werden, da dieses spezialisierte AwS eine umfangreiche Aufgabenunterstützung für die Aufgabe *Erstelle und übermittle Lehrveranstaltungsangebote* bietet. Durch die Einführung eines weiteren AwS entsteht das Problem, dass der Mensch-Maschine-Kommunikationskanal₁ (MMK₁) zwischen der Organisationseinheit⁵ Lehrstuhl und dem AwS FlexNow entfällt und stattdessen ein neuer MMK₁ zwischen Lehrstuhl und Virtueller Campus etabliert wird. Diese Änderung führt zur Entstehung neuer

³Der Begriff AwS wird in Abschnitt 2.3, GP in Abschnitt 2.2.1 definiert.

⁴Der Begriff der Heterogenität sowie seine Interpretation in der vorliegenden Arbeit wird in Abschnitt 3.1.1 eingeführt.

⁵Unter Organisationseinheiten wird im Rahmen der vorliegenden Arbeit ein Verbund zusammengehörender Stellen verstanden.

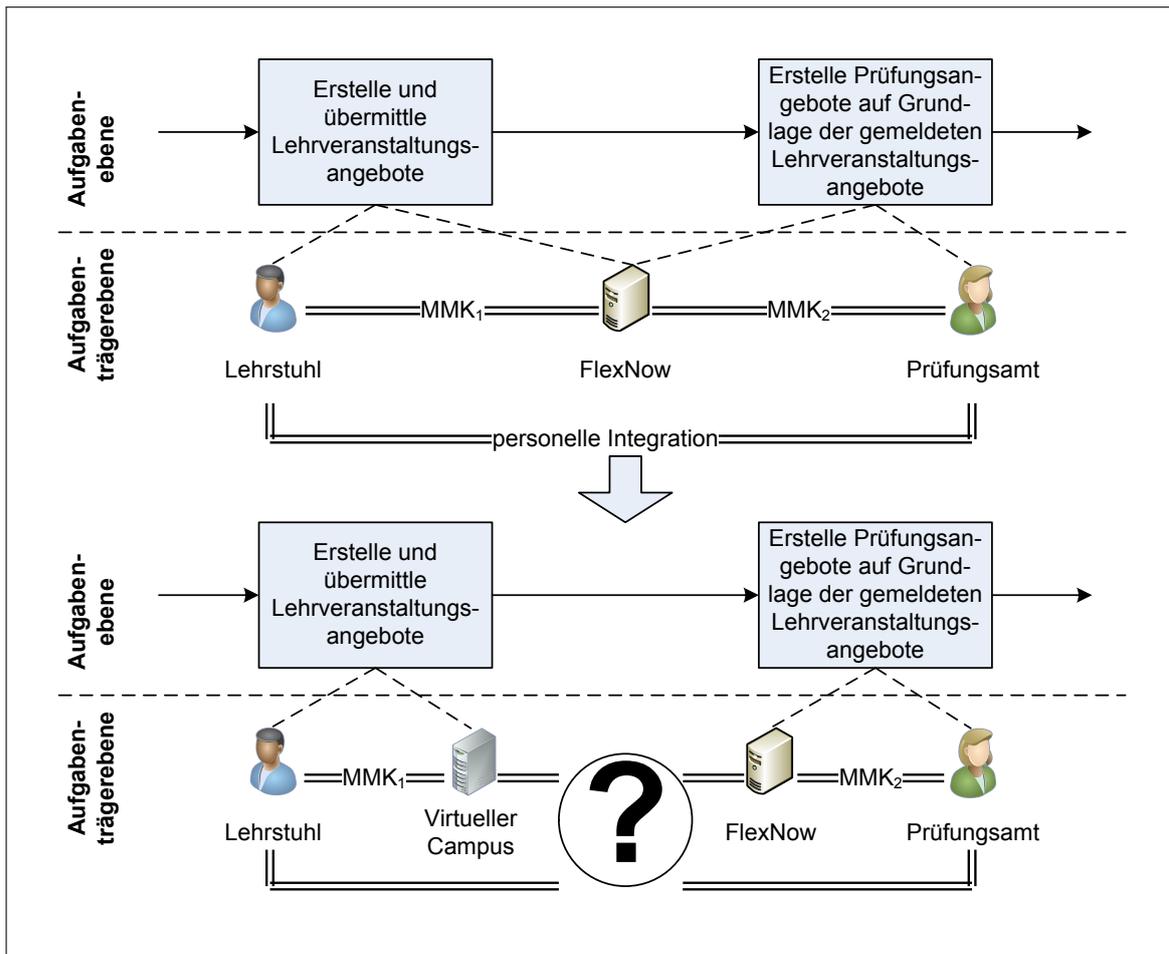


Abbildung 1.1: Gestaltung der Beziehung Menschen-Aufgaben-Technik

Integrationsbedarfe – in der Abb. 1.1 mit einem „?“ gekennzeichnet. Für diese gilt es geeignete Integrationsarchitekturen, insbesondere für die Gestaltung der Beziehung zwischen AwS⁶ zu konstruieren, diese miteinander zu vergleichen und sich für eine zu entscheiden.

Bei der Gestaltung von Integrationsarchitekturen ist häufig die Ressourcenebene der Ausgangspunkt. Der Fokus wird dabei zumeist auf die Technologie, insbesondere auf die Erstellung von Schnittstellen für die beteiligten AwS, gerichtet. Das Konstruktionsproblem hingegen sollte unter Berücksichtigung personeller AT und dem GP als Ausgangspunkt betrachtet werden. Eine Fokussierung auf die Technologie birgt die Gefahr, Schnittstellen zu schaffen, die nicht erforderlich und womöglich überdimensioniert sind, folglich über die eigentlichen Bedarfe hinausgehen und somit mehr Kosten verursachen als nötig wäre.

⁶In der vorliegenden Arbeit ist zu trennen zwischen den Beziehungen innerhalb eines AwS, die Gegenstand der Intra-AwS-Integration sind, sowie den Beziehungen von AwS untereinander, welche Gegenstand der Inter-AwS-Integration sind (Schissler et al. 2001, S. 4). In der vorliegenden Arbeit liegt der Fokus auf der Inter-AwS-Integration.

Um dem Problem der Heterogenität zumindest für den maschinellen Anteil zu begegnen, wird in der betrieblichen Praxis oftmals ein Wechsel von der Best-of-Breed-Strategie mit vielen spezialisierten AwS hin zu wenigen AwS, die viele Aufgabenbereiche unterstützen, verfolgt. Bei der Best-of-Breed-Strategie wird für jeden Aufgabenbereich das AwS ausgewählt, welches die gestellten Anforderungen aus Sichtweise der Fachanwender am geeignetsten erfüllt. Im Gegensatz dazu sollen mit AwS, die viele Aufgabenbereiche unterstützen, die Anzahl der AwS im Informationssystem auf einem beherrschbaren Niveau gehalten werden. In der Vergangenheit wurde jedoch erkannt, dass zum einen selbst derartige AwS häufig nicht vollständig die erforderlichen Aufgabenbereiche in dem Maße abdecken können, wie ursprünglich erwartet und somit weitere Maßnahmen zur Unterstützung unzureichend abgedeckter Aufgabenbereiche notwendig werden (Schelp und Schwinn 2005, S. 1333; Sinz und Wismans 2001, S. 27). Zum anderen hat sich herausgestellt, dass AwS, die viele Aufgabenbereiche eines Informationssystem abdecken können, nicht ohne zeitliche Verzögerungen um die Unterstützung neuer, durch geänderte Umweltbedingungen induzierte Aufgabenbereiche erweitert werden können. Infolgedessen werden dann vielfach spezialisierte AwS zumindest temporär, bis die neu hervorgerufenen Aufgabenbereiche auch durch das integrierte AwS unterstützt werden, eingeführt (Winter und Schelp 2006, S. 23).

Aus den vorausgehenden Ausführungen entsteht die für die vorliegende Arbeit zugrundeliegende Hypothese, dass nicht in allen Fällen integrierte AwS zur Aufgabenvernetzung erforderlich sind. Im Rahmen dieser Hypothese sind Merkmale herauszuarbeiten, die Hinweise auf einen geeigneten Automatisierungsgrad der zu konstruierenden Integrationsarchitekturen geben. Diese Merkmale weisen in konkreten Anwendungsfällen bspw. darauf hin, ob die Aufgaben mittels personeller AT, AwS oder in Kooperation beider zu vernetzen sind. Daneben können die Merkmale über das zu verwendende Integrationskonzept Aufschluss geben. Des Weiteren sind Restriktionen herauszuarbeiten, die die Freiheitsgrade bei der Wahl von Integrationsarchitekturen einschränken. Hierzu zählen unter anderem Vorgaben von der Unternehmensleitung über die zu verwendenden AwS ebenso wie Einschränkungen in Bezug auf die Interoperabilität eingesetzter AwS.

1.2 Untersuchungssituation

In Abb. 1.2 wird die sich aus der vorausgehenden Problemstellung ergebende *Untersuchungssituation* in Anlehnung an den Vorschlag von Ferstl (1979, S. 43) zur Strukturierung einer Problemsituation zusammenfassend dargestellt und nachfolgend beschrieben.

Die *Untersuchungssituation* orientiert sich vor dem Hintergrund des erkenntnistheoretischen Paradigmas an der konstruktionswissenschaftlichen Forschung (Wilde und

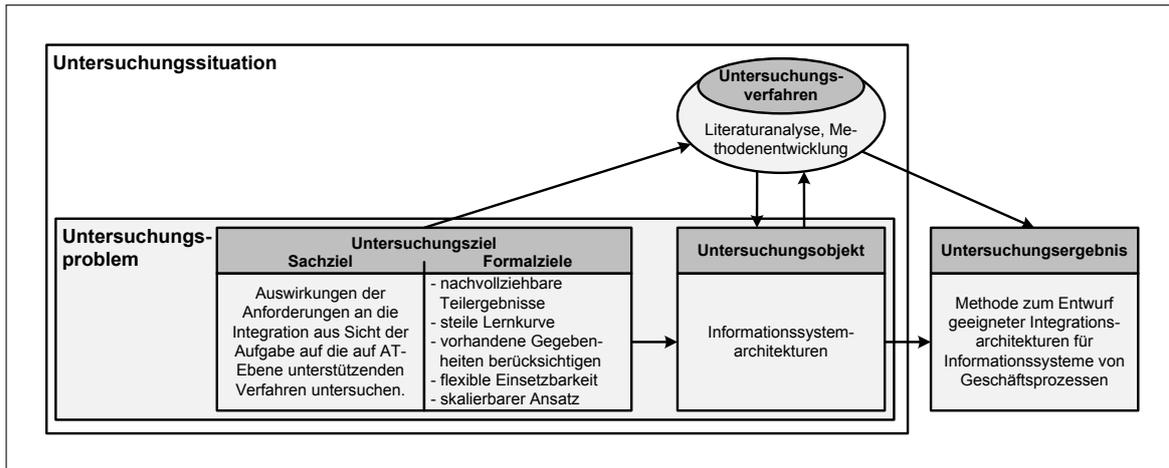


Abbildung 1.2: Untersuchungssituation zur Entwicklung einer Methode bei der Identifikation von Integrationsbedarfen

Hess 2007, S. 281). Ausgangspunkt des Forschungsprozesses stellt dabei zunächst die Wahrnehmung eines aus der betrieblichen Praxis identifizierten Problems, nämlich der Gestaltung geeigneter Integrationsarchitekturen für Informationssysteme, dar (Abschnitt 1.1). Anschließend wird geprüft, inwiefern der gegenwärtige Stand der Technik zur Lösung des identifizierten Problems geeignet ist und dabei die Forschungslücke herausgearbeitet (Abschnitt 4.5). Darauf aufbauend wird ein Entwurfsprozess mit dem Ziel, ein Artefakt in Form einer Methode zu erzeugen, durchlaufen (Kapitel 5, Kapitel 6). Abschließend wird dieses Artefakt hinsichtlich seines Nutzens evaluiert (Abschnitt 6.5). Die Konzepttauglichkeit der Methode wird mittels Fallstudien (Kapitel 6, Kapitel 7) überprüft. Damit wird zum einen der Forderung einer beispielhaften Instanziierung des Artefakts, wie bspw. von Offermann et al. (2010, S. 291) gefordert, nachgekommen und zum anderen die Vermittlung der Forschungsergebnisse unterstützt (Kapitel 7).

Das *Untersuchungsobjekt* der vorliegenden Arbeit sind Architekturen für Informationssysteme, insbesondere Integrationsarchitekturen. Der Ausschnitt, der dabei analysiert werden soll, wird durch das *Untersuchungsziel* spezifiziert. Dieses gliedert sich in Sach- und Formalziele. Das *Sachziel* ist die Untersuchung der Möglichkeiten, die unterschiedlichen Bedarfe der durch die Vernetzung von Aufgaben hervorgerufenen Anforderungen an die Integration auf der AT-Ebene zu lösen. Daraus ergibt sich nachfolgend die im Rahmen dieser Arbeit zu beantwortende Forschungsfrage:

Wie kann der Entwurf geeigneter Integrationsarchitekturen, die die Anforderungen des Geschäftsprozesses erfüllen und neben Anwendungssystemen den personellen Aufgabenträger beim Entwurf einbeziehen, methodisch unterstützt werden?

Ein *Formalziel* bei der Gestaltung des Lösungsansatzes ist es, die durch die Anwendung des Lösungsansatzes getroffenen Entscheidungen für alle Projektbeteiligten offen zu legen und die einzelnen Schritte sowie deren Ergebnisse, die zur Erstellung einer geeigneten Integrationsarchitektur geführt haben, nachvollziehbar zu machen. Da beim Entwurf von Integrationsarchitekturen die subjektiven Einflüsse kaum eliminiert werden können, soll durch dieses Formalziel zumindest die Willkür bei der Entscheidungsfindung durch Offenlegung weitestgehend reduzieren.

Ein weiteres Formalziel besteht darin, dass der zu konstruierende Lösungsansatz leicht erlernbar sein soll. Die steile Lernkurve soll durch Fallstudien, die die Anwendung der einzelnen Schritte ausführlich demonstrieren, unterstützt werden.

Das Formalziel der Berücksichtigung vorhandener Gegebenheiten bedeutet, dass der zu konstruierende Lösungsansatz eine bereits etablierte AT-Ebene bei der Bedarfsermittlung berücksichtigen soll. Dazu muss es ermöglicht werden, in einer von Best-of-Breed geprägten AWS-Landschaft eine flexible Integration heterogener AWS vornehmen zu können. Ferner sollte auch die personelle Integration von Aufgabenkomplexen als mögliche Lösung einbezogen werden. Darüber hinaus gilt es zu berücksichtigen, dass bei der Bedarfsermittlung nicht immer das komplette Informationssystem betroffen ist und aus diesem Grund die Eingrenzung auf die von den Änderungen abhängigen Bereiche zu unterstützen ist.

Das Formalziel der flexiblen Einsetzbarkeit verlangt, dass die Anwendung der Ergebnisse nicht nur für die hier vorgestellten Fallstudien möglich sein soll, sondern dass mit diesen auch Integrationsarchitekturen für Informationssysteme anderer GP entworfen werden sollen.

Das Formalziel der Skalierbarkeit soll erreicht werden, indem bei der Konstruktion des Lösungsansatzes darauf geachtet wird, dass dieser sowohl bei Informationssystemen mit geringerer als auch mit umfangreicherer Extension eingesetzt werden kann.

Die *Untersuchungsverfahren*, die zur Bildung des Artefakts angewendet werden, sind zum einen die strukturierte Literaturanalyse sowie zum anderen die Entwicklung, Verfeinerung und das Testen der Methode an Fallstudien aus der betrieblichen Praxis. In die Methodenentwicklung fließen die durch die Untersuchungen gewonnen Erkenntnisse ein.

Das *Untersuchungsergebnis* drückt sich in einer auf Pragmatik ausgerichteten Methode zum Entwurf geeigneter Integrationsarchitekturen für Informationssysteme von GP aus.

1.3 Überblick über Lösungsansätze für die Integration

Für die Gestaltung der Aufgabenintegration mittels AT bietet sich ein breit gefächertes Repertoire an organisatorischen sowie technologischen Maßnahmen an. Zu den organisatorischen Gestaltungsmöglichkeiten zählen Änderungen an der Aufbau- und Ablauforganisation⁷. Dabei können bspw. Organisationseinheiten so gebildet werden, dass sie alle Aufgaben eines GP ausführen. Darüber hinausgehend können zur Gestaltung in diesem Bereich auch die bereits vorhandenen AwS einbezogen werden. Unter den Instrumenten in diesem Umfeld werden das Enterprise Architecture Management (EAM) mitsamt Strukturierungsmuster⁸, das Master Data Management (MDM)⁹ sowie der Bereich der Softwarekartografie¹⁰ subsumiert.

Zu den Technologien werden die Lösungsverfahren zur Kopplung von AwS gezählt. Hierzu zählen Technologien und Strukturierungsmuster aus dem Umfeld Enterprise Application Integration (EAI)¹¹ sowie Workflow-Management-Systeme (WfMS), mithilfe derer sowohl Arbeitsabläufe gesteuert als auch Dokumente und Arbeitsergebnisse verwaltet werden können (Ferstl und Sinz 2013, S. 455). Weitere Technologien zur Herstellung der Interoperabilität unterschiedlicher AwS werden in Abschnitt 3.1.6 überblicksartig vorgestellt.

1.4 Charakterisierung des Lösungsansatzes

Den Ausgangspunkt der Untersuchung stellen Informationssysteme dar. Da es zum einen zur Erzeugung des in Abschnitt 1.2 definierten Untersuchungsergebnisses nicht zweckmäßig ist, bei der Vielzahl existierender Methoden im Forschungsbereich zu Informationssystemen eine gänzlich neuartige zu entwerfen, wird auf das Semantische Objektmodell (SOM)¹² als Basis für den zu konstruierenden Lösungsansatz zurückgegriffen. Zum anderen erlaubt sie sowohl die Abbildung von GP als auch die Spezifikation der ausführenden AT und ist deshalb geeignet, das Informationssystem von Unternehmen abzubilden. Zudem bietet die SOM-Methodik eine Struktursicht, die im Rahmen dieser Arbeit insbesondere zur Analyse der Auswirkungen alternativ konstruierter Ressourcenebenen intensiv genutzt werden kann. Darüber hinaus bietet sich die SOM-Methodik auch deshalb an, da sie ein objekt- und geschäftsprozes-

⁷Die Begriffe Aufbau- und Ablauforganisation sowie deren Gestaltungsparameter werden in den Abschnitten 2.2 und 3.1.1 eingeführt.

⁸vgl. bspw. Buckl et al. (2008)

⁹vgl. bspw. Conrad (1997) für die möglichen Konzepte in diesem Umfeld, Legner und Otto (2007, S. 566-568) für die Kategorisierung von Stammdatenarchitekturen sowie Schmidt (2010) für eine Methode zur Gestaltung der Stammdatenintegration

¹⁰siehe Abschnitt 3.1.5

¹¹vgl. bspw. Hohpe et al. (2004)

¹²Die SOM-Methodik wird in Abschnitt 2.1 eingeführt.

sorientierter Modellierungsansatz ist (Ferstl und Sinz 2013, S. 194) und somit die spezifischen Vorteile beider Paradigmen vereint.

Die SOM-Methodik soll beim zu entwickelnden Ansatz während der Ermittlung der Integrationsbedarfe sowie der Ableitung geeigneter Integrationsarchitekturen herangezogen werden. Die Idee besteht darin, zunächst durch Änderungen auf GP- oder Ressourcenebene hervorgerufene Integrationsbedarfe mittels der Verzeichnung von AT in Modellen aufzudecken. In der sich anschließenden Delta-Analyse sollen anhand von Merkmalen und unter Berücksichtigung etwaiger Restriktionen Gestaltungsempfehlungen für die geeignete Aufgabenintegration über AT abgeleitet werden. Von der konkreten technologischen Realisierung soll abstrahiert werden, so dass bspw. bei der Verknüpfung von AwS zwischen den Architekturvarianten Punkt-zu-Punkt-Verbindung oder Hub-and-Spoke-Architektur frei gewählt werden kann. Dadurch bleibt die von der Unternehmung getroffene Strategie bezüglich der Verknüpfung unberührt.

1.5 Entstehung, Umfeld und Adressaten der Arbeit

Die Idee zur Bearbeitung dieses Themenfeldes entstand aus der Tätigkeit beim damaligen Wissenschaftlichen Institut für Hochschulsoftware der Universität Bamberg (ihb)¹³ im Rahmen der Weiterentwicklung des AwS FlexNow¹⁴. Den ausschlaggebenden Impuls hat die Kopplung der AwS FlexNow vom ihb und LSF¹⁵ der Firma HIS eG gegeben. So konnte im Rahmen erster Recherchen festgestellt werden, dass die Integration im Informationssystem primär aus technischer Sicht betrachtet wird und dort der Fokus auf der Interoperabilität der AwS, insbesondere auf der Erstellung von Schnittstellen, liegt. Dadurch werden jedoch sowohl die GP, aus denen die Integrationsbedarfe begründet werden, als auch die Rolle des personellen AT zur Unterstützung der Integration vernachlässigt. Aus diesen Gegebenheiten heraus entstand der Gedanke, dieses Themengebiet genauer zu untersuchen und es unter Einbezug der Aufgabenebene umfassender zu beleuchten, anstatt auf rein technischer Ebene lediglich die Realisierung der Kopplung zu betrachten. Zudem bestand durch die vorliegende Fallstudie die Möglichkeit, die im Rahmen der Forschung gewonnenen Erkenntnisse zu überprüfen.

¹³Das ihb bestand seinerzeit aus zwei Geschäftsbereichen. Der Geschäftsbereich aus dessen Forschungstätigkeiten das AwS FlexNow entstand ist heute als eigenständige Firma Institut für Hochschulsoftware Bamberg eG ausgegründet.

¹⁴FlexNow ist ein AwS, das Prüfungsprozesse aus Verwaltungssicht unterstützt (Wismans 1996; Sinz und Wismans 1998).

¹⁵LSF steht für Lehre, Studium, Forschung. Dieses AwS unterstützt die Verwaltung von Ressourcen (Einrichtungen, Personen, Räume) und stellt seine Dienste als Studieninformations-, Studienberatungs- sowie Planungssystem für unterschiedliche Nutzerkreise zur Verfügung.

Die vorliegende Arbeit richtet sich an Systemgestalter. Dies umfasst alle Personen, die mit der Planung und Durchführung von Vorhaben zur Gestaltung betrieblicher Informationssysteme befasst sind, und Forscher, die sich mit der Gestaltung dieser auseinandersetzen. Die Adressaten finden in dieser Arbeit eine dokumentierte Methode einschließlich deren Anwendung im Rahmen einer umfangreichen Fallstudie und können diese in ihrem eigenen Tätigkeitsbereich einsetzen und bewerten. Fach- und IT-Architekten in Organisationen können bspw. mithilfe des Methodenvorschlags aus einem Integrationsbedarf auf Aufgabenebene einen begründeten Handlungsbedarf auf AT-Ebene ableiten, um anschließend eine geeignete Integrationsarchitektur für das Informationssystem herzuleiten. Softwarehersteller sollen bei der strategischen Weiterentwicklung vorhandener AwS unterstützt werden. Zu diesem Zweck kann die Methode eingesetzt werden, um aus Sicht des GP zu prüfen, welche Aufgaben gegenwärtig durch die sich im Portfolio befindenden AwS abgedeckt werden und um welche Aufgabenbereiche die Unterstützung der AwS durch Weiterentwicklung ausgeweitet werden kann. Für Forscher ist die Arbeit als Beitrag zum Forschungsbereich der Gestaltung betrieblicher Informationssysteme anzusehen. Zur Erreichung dieses Vorhabens werden bestehende Methoden weiterentwickelt und eigene Vorschläge zur Lösung von Problemen im Bereich der Integration in Informationssystemen eingebracht.

Bei der Konstruktion der Methode, insbesondere beim Einsatz der gefundenen Merkmale, die Hinweise zur Gestaltung der Integration geben, wurde Wert darauf gelegt, dass deren Anwendung weder einen hohen Einarbeitungsaufwand noch besondere technische Hilfsmittel erfordert.

1.6 Aufbau der Arbeit

In Abb. 1.3 wird der Aufbau der Arbeit dargestellt. Daraus wird ersichtlich, dass nach der Einleitung in Kapitel 2 die Unternehmensarchitektur als übergeordneter Rahmen vorgestellt wird. Daneben wird im zweiten Kapitel ein Ordnungsrahmen, anhand dessen die spezifischen Grundlagen erläutert und in eine Hierarchie gebracht werden sollen, aufgebaut. Zudem wird das in der Arbeit verwendete Fallbeispiel aus dem Hochschulumfeld zur Demonstration der Ausführungen eingeführt.

In Kapitel 3 wird die Gestaltung der Integration in Informationssystemen als Teilaufgabe bei der Systementwicklung aufgefasst. Dazu wird eine hierarchische Terminologie aufgebaut. Ziel ist es, eine umfassende Definition für die beiden in dieser Arbeit zentral verwendeten Begriffe Integrationsbedarf und Integrationsarchitektur zu erstellen.

In Kapitel 4 werden Methoden, die die Gestaltung der Integration in Informationssystemen zum Gegenstand haben und im Rahmen einer strukturierten Literaturrecherche gefunden wurden, anhand von Anforderungen (Abschnitt 4.3) evaluiert.

Gegenstand von Kapitel 5 ist die Identifikation von Merkmalen, mithilfe derer die Entscheidungsfindung für die Gestaltung der Integration in Informationssystemen unterstützt werden kann. Ausgangspunkt hierfür ist ebenfalls die Durchführung einer strukturierten Literaturanalyse, mithilfe derer eine umfassende Basis an Merkmalen gebildet wird.

Aufbauend auf den vorhergehenden Kapiteln wird in Kapitel 6 eine modellgestützte Methode zur Ableitung geeigneter Integrationsarchitekturen aus dem GP entwickelt. Die Ausführungen werden anhand des in Abschnitt 2.4 eingeführten Fallbeispiels aus dem Hochschulbereich verdeutlicht. Abschließend wird die vorgeschlagene Methode bewertet (Abschnitt 6.5, Tab. 6.14). Insgesamt wird mit diesem Kapitel die Erreichung des pragmatischen Erkenntnisziels verfolgt.

In Kapitel 7 wird die Anwendbarkeit der in Kapitel 6 konstruierten Methode demonstriert. Dieses Kapitel dient als Machbarkeitsnachweis. Die Durchführung von diesem wird durch einen im Rahmen der vorliegenden Arbeit entwickelten Prototypen unterstützt. Mithilfe der entwickelten Methode werden an einem Fallbeispiel aus dem Bereich der Güterdistribution die Ergebnisse der Istanalyse mit den Auswirkungen eines entworfenen Szenarios im Rahmen einer Waht-If-Analyse miteinander verglichen.

Abschließend werden in Kapitel 8 die Ergebnisse zusammengefasst und ein Ausblick auf mögliche Weiterentwicklungen gegeben.

1.7 Konventionen der Arbeit

Zum besseren Verständnis der Arbeit werden notwendige Begriffe schrittweise eingeführt. Zentrale Begriffe, die an einer Stelle definiert oder näher beschrieben werden, werden im Text **fett** hervorgehoben. Darüber hinaus können die zentralen Begriffe im Stichwortverzeichnis der Arbeit nachgeschlagen werden. Die verwendeten Abkürzungen werden im Abkürzungsverzeichnis erläutert. Zu Beginn von jedem Kapitel wird ein einleitender Überblick gegeben und am Ende wird selbiges mit den Kernergebnissen zusammengefasst, um einen schnellen Zugang zur Arbeit zu ermöglichen. Besonders zu betonende bzw. hervorzuhebende Textstellen werden *kursiv* dargestellt.

Im gesamten Beitrag wird aus Gründen der besseren Lesbarkeit bei geschlechtsspezifischen Begriffen die maskuline Form verwendet. Diese Form versteht sich explizit als geschlechtsneutral. Gemeint sind selbstverständlich immer beide Geschlechter.

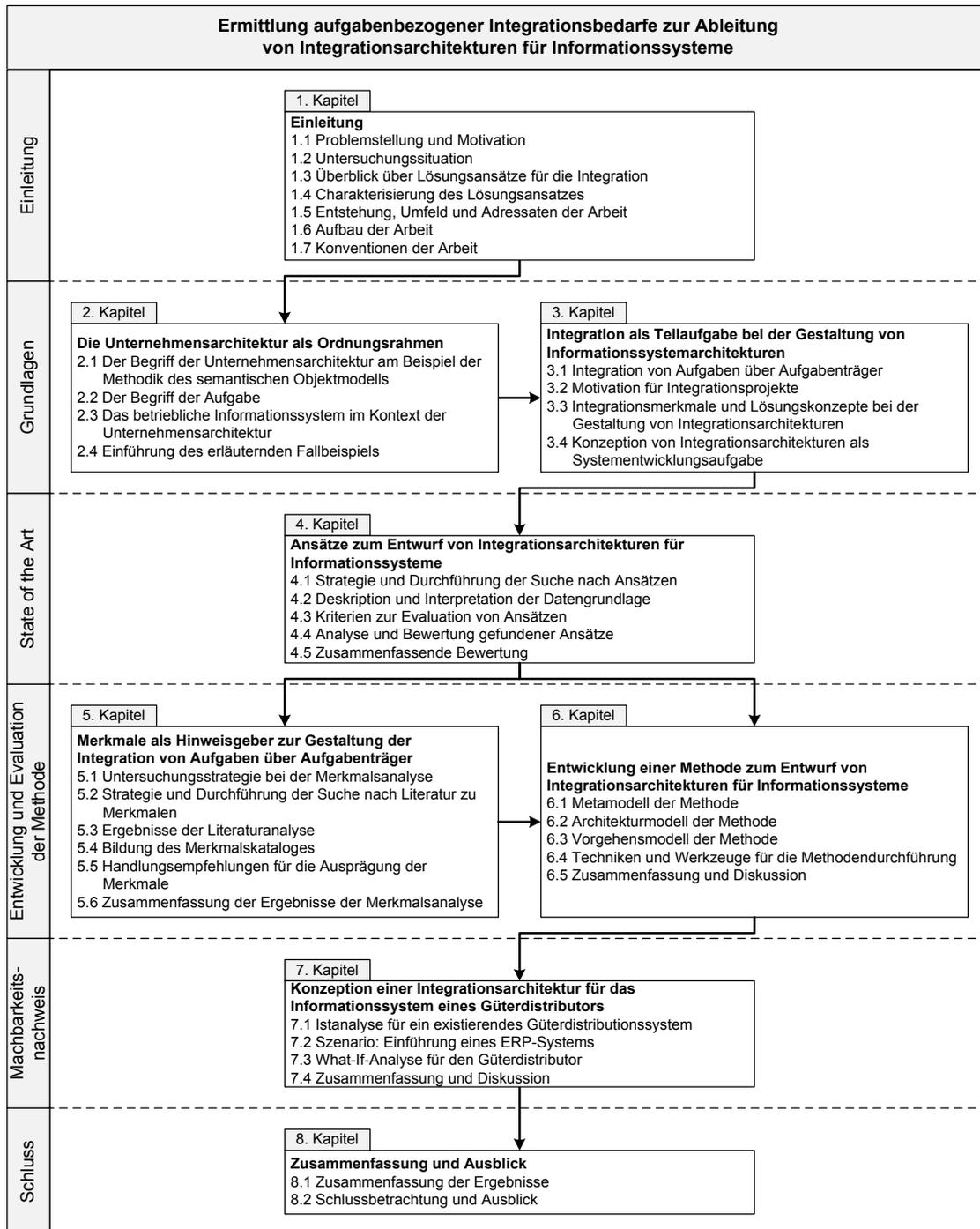


Abbildung 1.3: Aufbau der Arbeit

2 Die Unternehmensarchitektur als Ordnungsrahmen

2.1 Der Begriff der Unternehmensarchitektur am Beispiel der Methodik des semantischen Objektmodells	13
2.2 Der Begriff der Aufgabe	17
2.2.1 Der Prozessbegriff	18
2.2.2 Automatisierung von Aufgaben	20
2.3 Das betriebliche Informationssystem im Kontext der Unternehmensarchitektur	21
2.4 Einführung des erläuternden Fallbeispiels	24

In diesem Kapitel werden die für die vorliegende Arbeit spezifischen Grundlagen eingeführt. Da der Begriff der Unternehmensarchitektur den Ordnungsrahmen bildet, wird dieser zunächst in Abschnitt 2.1 am Beispiel der SOM-Methodik eingeführt. Eine zentrale Rolle darin nimmt die Aufgabenebene ein, deshalb wird auf diese in Abschnitt 2.2 eingegangen. Dazu werden insbesondere der Prozessbegriff (Abschnitt 2.2.1) und die Aufgabenautomatisierung (Abschnitt 2.2.2) beleuchtet. Anschließend wird in Abschnitt 2.3 das für die vorliegende Arbeit gültige Verständnis vom Informationssystem dargelegt. Die Ausführungen der vorhergehenden Abschnitte werden in Abschnitt 2.4 exemplarisch vertieft. Auf das in Abschnitt 2.4 eingeführte Fallbeispiel wird im Verlauf der vorliegenden Arbeit zur Erläuterung der Ausführungen zurückgegriffen.

2.1 Der Begriff der Unternehmensarchitektur am Beispiel der Methodik des semantischen Objektmodells

Der Begriff der **Architektur** wird in der vorliegenden Arbeit analog zur Architekturlehre im Bauwesen verwendet. Zum einen beinhaltet die Architektur den Bauplan im Sinne der Spezifikation und Dokumentation verwendeter Elemente sowie ihrer Beziehungen und Interaktionen (Barroca et al. 2000, S. 2). Diese sind dabei unter allen relevanten Blickwinkeln zu dokumentieren. Zum anderen enthält sie die Konstruktionsregeln in Form von Vorschriften für die Erstellung des Bauplans (Sinz 2002, S. 1055).

Da jedes System notwendigerweise eine Architektur besitzt, gilt dies auch für Unternehmen, deren Bauplan anhand einer Unternehmensarchitektur beschrieben werden kann (Sinz 2004, S. 315). Das Konzept der Unternehmensarchitektur ist ein modell-

theoretisches Rahmenwerk, welches eine umfassende Modellierung des betrieblichen Realsystems ermöglicht. Dabei kann mit einer Unternehmensarchitektur der Ist-Zustand in Form eines Beschreibungsmodells oder der Soll-Zustand in Form eines Gestaltungsmodells spezifiziert werden (Sinz 2004, S. 315).

Indem ein Unternehmen aus verschiedenen Blickwinkeln betrachtet wird, soll die mit der Analyse und Gestaltung von Unternehmensarchitekturen einhergehende Komplexität beherrschbar gemacht werden (Sinz 2002, S. 1056). Dazu hat jeder Blickwinkel eine eigene Modellebene (Ferstl und Sinz 2013, S. 194-196), auf welcher das Unternehmen im Hinblick auf eine bestimmte Zielsetzung vollständig abgebildet werden sollte (Sinz 2002, S. 1057).

Neben der Gliederung in Modellebenen lässt sich die Komplexität einer Unternehmensarchitektur durch die Bildung von Sichten besser beherrschen. Jede Sicht hebt bestimmte Teilausschnitte einer Modellebene hervor, während andere Teilausschnitte bewusst ausgeblendet werden. Dadurch stellt eine Sicht im Allgemeinen eine unvollständige Beschreibung der jeweiligen Modellebene dar (Sinz 2002, S. 1057). Ferstl und Sinz (2013, S. 142) unterscheiden zwischen statischen Sichten auf die Funktionen (**Funktionssicht**, F), Datenstrukturen (**Datensicht**, D) und Kommunikationskanäle (**Interaktionssicht**, I) sowie der dynamischen verhaltensorientierten Sicht auf Vorgänge (**Vorgangssicht**, V).

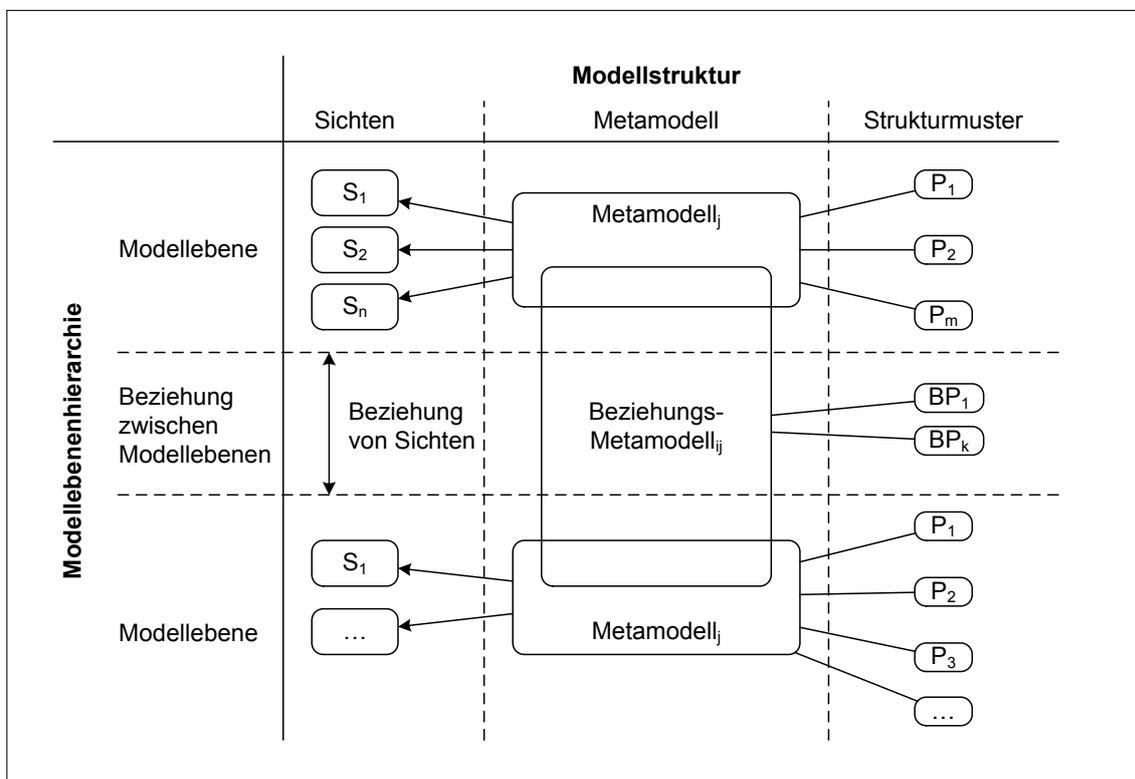


Abbildung 2.1: Generischer Architekturrahmen (Sinz 2002, S. 1056)

2.1 Der Begr. der Unternehmensarchitektur am Bsp. der Methodik des semantischen Objektm.

Durch den in Abb. 2.1 dargestellten generischen Architekturrahmen lässt sich zeigen, wie die Modellebenen, Teilmodellsysteme und Sichten in eine integrierte Gesamtarchitektur überführt werden können. Zum einen enthält dieser die Regeln zur Konstruktion der Teilmodellsysteme und Sichten (Metamodell) und zum anderen die Angaben über die Beziehungen zwischen den Modellebenen (Beziehungs-Metamodell). Aus der Abb. 2.1 wird auch ersichtlich, dass jede Modellebene über ein eigenes Metamodell verfügt und die Sichten einer Modellebene als Projektionen auf das Metamodell spezifiziert sind (Sinz 2002, S. 1056-1058).

Mit **Unternehmensarchitekturen** wird das Ziel verfolgt, „die wichtigsten Artefakte eines Unternehmens und deren Beziehungen in Form von Modellen abzubilden“ (Aier et al. 2008, S. 292). Hintergrund ist, dass die in vielen Unternehmen historisch gewachsenen, heterogenen Prozess- und Informationstechnologie (IT)-Landschaften durch ihre enorme Komplexität zu einer hohen Kostenbelastung und einer geringen Flexibilität führen (Buhl und B. Heinrich 2004, S. 311). Aus diesem Grund soll die Unternehmensarchitektur „helfen, z. B. durch Standardisierung von Prozessen oder IT-Funktionalitäten, dieser oftmals als 'Wildwuchs' bezeichneten Situation entgegen zu wirken, um die Kosten zu drücken bzw. neue fachliche Anforderungen schneller umzusetzen“ (Buhl und B. Heinrich 2004, S. 311). Auf der anderen Seite darf nicht verschwiegen werden, dass die Erstellung und Pflege umfangreicher Modellierungen sowohl kosten- als auch zeitintensiv ist und die Architekturmodelle oftmals schnell veralten. Deshalb wird der Nutzen von Unternehmensarchitekturen in der Praxis teilweise auch in Frage gestellt (Buhl und B. Heinrich 2004, S. 311). Dennoch sind sie in der Wissenschaft und in der Praxis verbreitet und akzeptiert (Aier et al. 2008, S. 292).

Für die vorliegende Arbeit wird als Beispiel für die Gestaltung der Unternehmensarchitektur stellvertretend die SOM-Methodik (Ferstl und Sinz 1990, S. 566-581; Ferstl und Sinz 1991, S. 477-491; Ferstl und Sinz 1995a, S. 209-220; Ferstl und Sinz 2006, S. 349-351; Ferstl und Sinz 2013, S. 194-236) gewählt, da sie sowohl objekt- als auch geschäftsprozessorientiert ist. Dadurch können die Defizite reiner daten- sowie prozessorientierter Ansätze überwunden werden (Ferstl und Sinz 2013, S. 193-194). Da eine vertiefende Untersuchung der Unternehmensarchitektur selber und der Vergleich verschiedener Ansätze zur Gestaltung selbiger nicht Gegenstand dieser Arbeit sind, wird stattdessen auf die einschlägige Literatur verwiesen (bspw. Aier et al. (2008); Leist-Galanos (2006); Winter und R. Fischer (2007)).

Die Objekt- und Geschäftsprozessorientierung in der SOM-Methodik führen dazu, dass sie zur ganzheitlichen Modellierung betrieblicher Systeme eingesetzt werden kann (Ferstl und Sinz 2013, S. 194-236). Kern der SOM-Methodik ist die Unter-

nehmensarchitektur, wie in Abb. 2.2 dargestellt. Daraus ist ersichtlich, dass sie die folgenden drei Modellebenen umfasst (Ferstl und Sinz 2013, S. 196-197):

1. Im **Unternehmensplan** wird die Gesamtaufgabe des Unternehmens aus der Außenperspektive beschrieben. Daneben werden zugehörige Strategien sowie Rahmenbedingungen spezifiziert. Der Unternehmensplan enthält Beschreibungen sowohl für die Aufgaben- als auch für die AT-Ebene. Dazu werden die Leistungserbringung und die dafür erforderlichen Ressourcen erfasst.
2. Das Lösungsverfahren für die Realisierung des Unternehmensplans wird im **Geschäftsprozessmodell** beschrieben. Dieses ist Teil der Innenperspektive und bezieht sich auf die Aufgabenebene. Im Geschäftsprozessmodell werden Struktur und Verhalten der Teilaufgaben des Unternehmens festgelegt.
3. Im **Ressourcenmodell** werden die personellen und maschinellen AT, die zur Durchführung betrieblicher Aufgaben erforderlich sind, definiert. Hierfür werden die Aufbauorganisation, die AwS sowie Maschinen und Anlagen spezifiziert. Das Ressourcenmodell gehört ebenfalls zur Innenperspektive und fokussiert sich auf die AT-Ebene. Für die vorliegende Arbeit sind insbesondere die AwS als maschinelle AT und die Aufbauorganisation als Repräsentation personeller AT von besonderem Interesse.

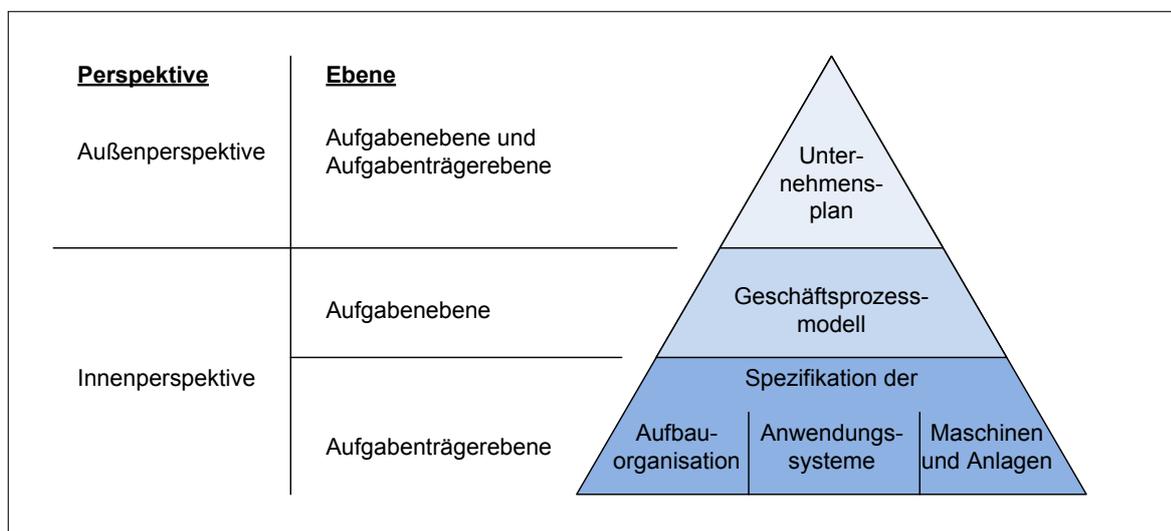


Abbildung 2.2: Unternehmensarchitektur der SOM-Methodik (Ferstl und Sinz 2013, S. 195)

Der Fokus der vorliegenden Arbeit liegt auf der Gestaltung der Innenperspektive der Unternehmensarchitektur (Abb. 2.2), deshalb beschränken sich die weiteren Ausführungen auf die Ebenen dieser Sicht.

2.2 Der Begriff der Aufgabe

In Abschnitt 2.1 wurde der Terminus des Geschäftsprozessmodells eingeführt. Gegenstand dessen sind Teilaufgaben des Unternehmens. Für den Begriff der **Aufgabe** wird die Definition von Kosiol herangezogen. Dieser fasst die Aufgabe als zweckbezogenes menschliches Handeln zusammen (Kosiol 1976, S. 43). Seiner Auffassung nach soll gedanklich von der Ausführung durch den AT abstrahiert (Kosiol 1976, S. 63) und dadurch die Gestaltung der Aufbauorganisation unterstützt werden (Kosiol 1976, S. 185). Die Aufgabe beschreibt somit eine Soll-Leistung, die von einem AT zu erbringen ist. AT können dabei sowohl Personen als auch Maschinen sein (Hentze et al. 2001, S. 153-154). Aus Sicht der Wirtschaftsinformatik sind die maschinellen AT, in Form von Rechner- und Kommunikationssystemen¹⁶, und die personellen AT von Interesse (Ferstl und Sinz 2013, S. 4-5).

Im Rahmen der vorliegenden Arbeit wird die von Ferstl und Sinz erweiterte Auffassung des Aufgabenbegriffes verwendet. Sie differenzieren bei der Aufgabenbeschreibung zwischen Außen- und Innensicht (Ferstl und Sinz 2013, S. 98), um Freiheitsgrade bei der Wahl des Automatisierungsgrades von Aufgaben zu erlangen (Ferstl und Sinz 2013, S. 202-203). Der erweiterte Aufgabenbegriff wird anhand von Abb. 2.3 erläutert (Ferstl und Sinz 2013, S. 103).

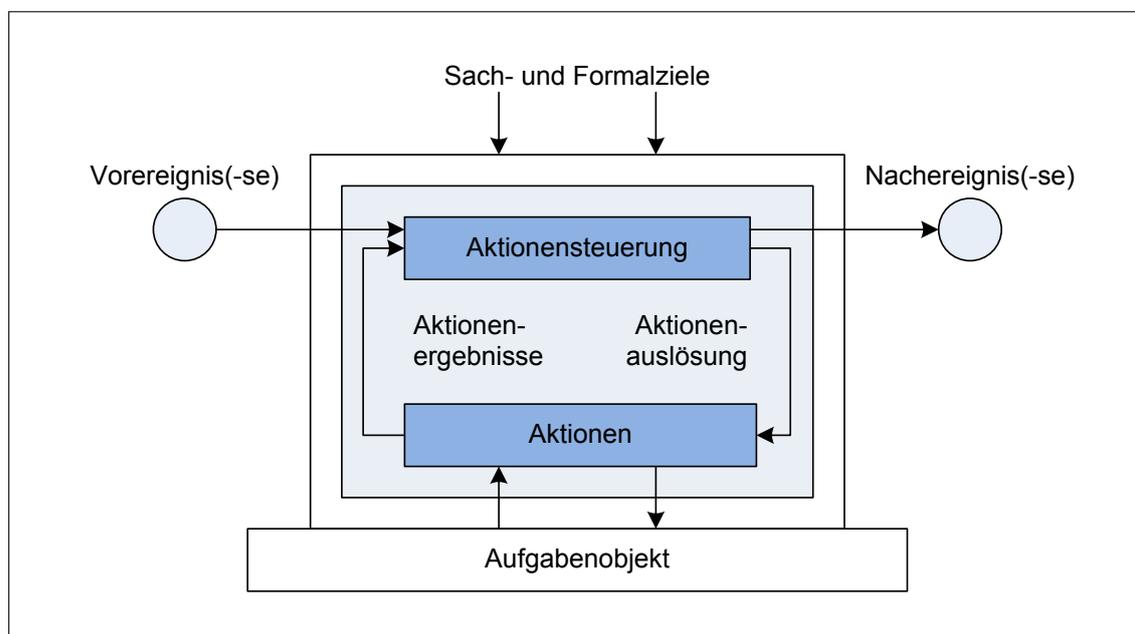


Abbildung 2.3: Struktur einer Aufgabe (Ferstl und Sinz 2013, S. 103)

¹⁶Für Kommunikationssysteme wird die nachfolgende Definition von Ferstl und Sinz (2013, S. 5) zugrunde gelegt: „Kommunikationssysteme sind spezielle maschinelle AT für Übertragungs- und Speicheraufgaben.“

Die **Außensicht einer Aufgabe** ist vom AT unabhängig und legt kein Verfahren zur Durchführung fest. Vor- und Nachereignisse sowie Sach- und Formalziele nehmen Einfluss auf die Aufgabe. Ausgelöst wird die Aufgabendurchführung durch ein oder mehrere Vorereignisse. Sie erzeugt als Ergebnis ein oder mehrere Nachereignisse (Ferstl und Sinz 2013, S. 98). „Sachziele bestimmen Art und Zweck der Leistungserstellung (...). Formalziele bestimmen Art und Umfang der Sachzielerreichung“ (Ferstl und Sinz 2013, S. 65).

In der **Innensicht einer Aufgabe** werden das Lösungsverfahren sowie zusätzliche aufgabenträgerspezifische Formalziele festgelegt (Ferstl und Sinz 2013, S. 102). Das Lösungsverfahren besteht aus einer Aktionensteuerung (AS) und aus einer Menge von Aktionen. Durch ein eintretendes Ereignis löst die AS Aktionen aus. Die Aktionen bearbeiten anschließend das Aufgabenobjekt (AO) entweder als Glieder einer Steuerkette¹⁷, ohne explizite Rückmeldung der Resultate, oder als Regler, um die Ergebnisse der AS mitzuteilen. In Abb. 2.3 wird das Lösungsverfahren als Regelkreis¹⁸, in dem die AS als Regelstrecke fungiert und über die Ergebnisse der Aktionendurchführung informiert wird, dargestellt (Ferstl und Sinz 2013, S. 102-103).

2.2.1 Der Prozessbegriff

Aus den bisherigen Ausführungen wird ersichtlich, dass Aufgaben häufig mit anderen Aufgaben interagieren. Diese Beziehungen können mithilfe des in diesem Abschnitt eingeführten **Prozessbegriffes** beschrieben werden.

Formal kann ein Prozess als Input-Output-System definiert werden, welches Ressourcen als Input verbraucht, um daraus eine Leistung zu erstellen. Die erstellte Leistung wird als Output an einen Empfänger abgegeben (Ferstl und Sinz 1993, S. 590; Ferstl und Sinz 2013, S. 200). Somit kann unter einem Prozess ein „raumzeitlich sich abspielender Vorgang, der zur Zielerreichung hinführt und sie bewirkt“ (Kosiol 1976, S. 185) und zur fortschreitenden Verwirklichung des Betriebsziels dient (Nordsieck 1972, S. 10), verstanden werden. Da sich die Ausführung einer Aufgabe in Form eines Vorgangs darstellen lässt (Ferstl und Sinz 1993, S. 590; Ferstl und Sinz 2013, S. 62), kann ein Prozess auch als Lösungsverfahren einer Aufgabe bezeichnet werden (Nordsieck 1972, S. 10; Kosiol 1976, S. 43).

GP sowie Aufgaben, die zur Durchführung von GP geeignet zusammengesetzt werden, sind Gegenstand auf der Ebene des Geschäftsprozessmodells der Unternehmensarchi-

¹⁷Der Begriff der Steuerkette stammt aus dem Gebiet der Systemtheorie und bezeichnet eine Folge miteinander interagierender Systemkomponenten (Ferstl und Sinz 2013, S. 27).

¹⁸Der Begriff des Regelkreises stammt aus dem Gebiet der Systemtheorie. Ein Regelkreis besteht im einfachsten Fall aus einer passiven, beeinflussbaren Systemkomponente, der Regelstrecke, und einem Regler, welcher aktiv in Erscheinung tritt (Ferstl und Sinz 2013, S. 27).

tektur der SOM-Methodik (Abb. 2.2). Dabei liegt dem SOM-Ansatz ein wesentlich umfassenderes Verständnis von GP zugrunde, welches in den drei Sichten – Leistungssicht, Lenkungsicht, Ablaufsicht – auf die entsprechenden GP zum Ausdruck kommt. Die Sichten nehmen Bezug auf die Systemmerkmale Struktur und Verhalten (Ferstl und Sinz 1995a, S. 213-214; Ferstl und Sinz 2013, S. 200).

Die **Leistungssicht** ist strukturorientiert. Sie bildet betriebliche Leistungsbeziehungen wie Güter, Zahlungen oder Dienstleistungen ab. In der **Lenkungsicht** wird die Koordination der Leistungserstellung anhand von betrieblichen Transaktionen betrachtet. Sie ist ebenfalls strukturorientiert. In der **Ablaufsicht** wird der ereignisgesteuerte Ablauf von Aufgaben dargestellt. Mit ihrer Hilfe wird das Verhalten von GP beschrieben (Ferstl und Sinz 2013, S. 200-201).

Aus Ablaufsicht betrachtet, kann eine Aufgabe durch ein auslösendes Vorereignis, welches das Lösungsverfahren für den zugrundeliegenden Vorgang bereitstellt, angestoßen werden (Ferstl und Sinz 2013, S. 62). Jeder Vorgang kann wiederum ein Nachereignis produzieren und somit die Durchführung einer nachfolgenden Aufgabe veranlassen (Ferstl und Sinz 2013, S. 62-63); vgl. Abb. 2.3.

Die Gestaltung von Prozessen kann als Mittel beim Entwurf der Ablauforganisation eingesetzt werden (Wittlage 1998, S. 173). Dabei wird nicht die Zielsetzung, sondern der Arbeitsprozess zur Zielerreichung in den Mittelpunkt gerückt (Kosiol 1976, S. 185; Gaitanides 2007, S. 22). Somit soll mit einem Prozess beschrieben werden, mit welcher Tätigkeit und mit welchen Arbeitsmitteln ein Gegenstand zu bearbeiten ist (Kosiol 1976, S. 67). Eine besondere Stellung nimmt dabei der Begriff des **GP** ein. Dieser wird häufig als ereignisgesteuerter Ablauf von Aktivitäten, der Nutzen gegenüber seinen nachfragenden Abnehmern stiftet, aufgefasst (Ferstl und Sinz 2013, S. 200).

In der SOM-Methodik werden GP unterteilt in Haupt- und Serviceprozesse. Diese sind durch Leistungsbeziehungen miteinander verbunden. Hauptprozesse geben ihre Leistung an die Umwelt ab, während Serviceprozesse ihre Leistungen entweder an einen Haupt- oder an einen anderen Serviceprozess abgeben können (Ferstl und Sinz 2013, S. 196). Darüber hinaus ist der Begriff des GP vom Begriff des **Workflows** abzugrenzen. GP beschreiben ein Netz von Aufgaben, deren Durchführung eine betriebliche Leistung erzeugt. Workflows hingegen spezifizieren ein Netz von Arbeitsschritten als Lösungsverfahren für betriebliche Aufgaben (Ferstl und Sinz 2013, S. 189). Auf Ebene des Workflows wird über die Automatisierbarkeit von Aufgaben entschieden und der erforderliche Automatisierungsgrad von Aufgaben für die Ausführungsebene festgelegt.

2.2.2 Automatisierung von Aufgaben

Wie bereits in Abschnitt 2.2 beschrieben und in Abb. 2.3 dargestellt, wird die Durchführung eines Lösungsverfahrens durch eine Nachricht an die AS ausgelöst. Das Lösungsverfahren ist als Vorgang zu verstehen, welcher aus einer AS und mindestens einer Aktion besteht. Es ergeben sich die drei Schrittararten Aktionen, AS und Vorgangsauslösung (VA), die jeweils automatisiert (*a*) oder personell (*p*) ausgeführt werden können. Anhand dieser Klassifizierung können 2³ Zielerreichungsgrade der Automatisierung einer Aufgabe differenziert werden. Zwischen der Vollautomatisierung (*a, a, a*) und der vollständig personellen Durchführung (*p, p, p*) können mit dieser Form der Beschreibung, verschiedene Teilautomatisierungsgrade unterschieden werden (Ferstl und Sinz 2013, S. 113-114).

Daneben hat Krumbiegel (1997, S. 129) eine Klassifizierung von Aufgaben hinsichtlich ihrer Automatisierbarkeit (*Kann*) sowie der Automatisierungsforderung (*Soll*) und der Automatisierung bezüglich ihrer Forderung (*Ist*) vorgeschlagen. In Tab. 2.1 wird diese Klassifizierung skizziert. Anhand dieser ist zunächst zu prüfen, ob eine Aufgabe automatisiert werden kann. Ist sie zumindest teilautomatisierbar, entsteht daraus die Forderung, sie teilweise zu automatisieren. Bei der vollständigen Automatisierbarkeit wird analog verfahren. Abschließend wird die realisierte Automatisierung gegen die Automatisierungsforderung geprüft. Abweichungen stellen Handlungsbedarfe in Bezug auf die Automatisierung dar und können durch die Implementierung entsprechend automatisierter Verfahren aufgelöst werden.

Tabelle 2.1: Klassifikation von Aufgaben hinsichtlich ihrer Automatisierbarkeit, Automatisierungsforderung und Realisierung (in Anlehnung an Krumbiegel (1997, S. 129))

Automatisierbarkeit (Kann)	Nicht automatisierbar	Teilautomatisierbar	Vollautomatisierbar
Automatisierungsforderung (Soll)	Nicht zu automatisieren (Darstellung: □)	Teilweise zu automatisieren (Darstellung: ◐)	Vollständig zu automatisieren (Darstellung: ◼)
Automatisierung bzgl. Forderung (Ist)	bzgl. Forderung nicht automatisiert (Darstellung: □)	bzgl. Forderung teilweise automatisiert (Darstellungen: ◐, ◑)	bzgl. Forderung vollständig automatisiert (Darstellungen: ◒, ◓, ◔)

2.3 Das betriebliche Informationssystem im Kontext der Unternehmensarchitektur

Das **betriebliche Informationssystem** bildet den zentralen Bezugsrahmen der vorliegenden Arbeit und wird in diesem Abschnitt eingeführt. Dieses wird in der vorliegenden Arbeit als Mensch-Aufgabe-Technik-System verstanden, in welchem Menschen und Maschinen zur gemeinsamen Zielerreichung kooperieren (L. J. Heinrich et al. 2011, S. 3). Den methodischen Rahmen für das Informationssystem bildet die Systemtheorie.¹⁹

Der Begriff Informationssystem, wie er im Rahmen dieser Arbeit verwendet wird, baut nicht auf der Tätigkeit des Informierens, sondern auf der Objektart Informationsverarbeitung auf (Ferstl und Sinz 2013, S. 11). Daraus ergibt sich, dass das Informationssystem das Teilsystem eines betrieblichen Systems ist, welches Informationen verarbeitet, d. h. erfasst, überträgt, transformiert, speichert und bereitstellt (Ferstl und Sinz 2013, S. 3). Basierend auf diesen Ausführungen kann das Informationssystem wie folgt definiert werden:

Definition 1 (Informationssystem) *Das Informationssystem dient zur Planung, Steuerung und Kontrolle der betrieblichen Leistungserstellung (Ferstl und Sinz 2013, S. 35) und ist damit der informationsverarbeitende Teil der Unternehmensarchitektur. Sie umfasst sowohl die Aufgabenebene (Aufgaben) als auch die AT-Ebene (Menschen und Maschinen) (Ferstl und Sinz 2013, S. 4-5).*

Die Bestandteile eines Informationssystems werden schematisch in Abb. 2.4 dargestellt. Dabei wird zwischen Aufgaben- und AT-Ebene differenziert. Beide Ebenen zusammen dienen zur Beschreibung des Informationssystems (Sinz 2002, S. 1055).

Der für die Abb. 2.4 relevante Aufgabenbegriff auf Aufgabenebene wurde in Abschnitt 2.2 eingeführt. Auf AT-Ebene wird zwischen maschinellen und personellen AT unterschieden. Im Rahmen der vorliegenden Arbeit wird die Betrachtung bei den maschinellen AT auf die AwS reduziert.

¹⁹Für ein umfangreiches Studium des Informationssystem im Umfeld der Systemtheorie vgl. bspw. Ferstl und Sinz (2013, S. 13-35).

Nach diesen Ausführungen kann für den Begriff des **AT** folgende Definition zugrunde gelegt werden:

Definition 2 (Aufgabenträger) Ein Aufgabenträger führt die ihm zugeordneten Aufgaben durch. Dabei können die Aufgaben, im Verständnis der Unternehmung als offenes, sozio-technisches und zielgerichtetes System (Ferstl und Sinz 2013, S. 65), von Menschen (nicht automatisierte Aufgaben), Maschinen (vollautomatisierte Aufgaben) oder in Kooperation beider (teilautomatisierte Aufgaben) durchgeführt werden.

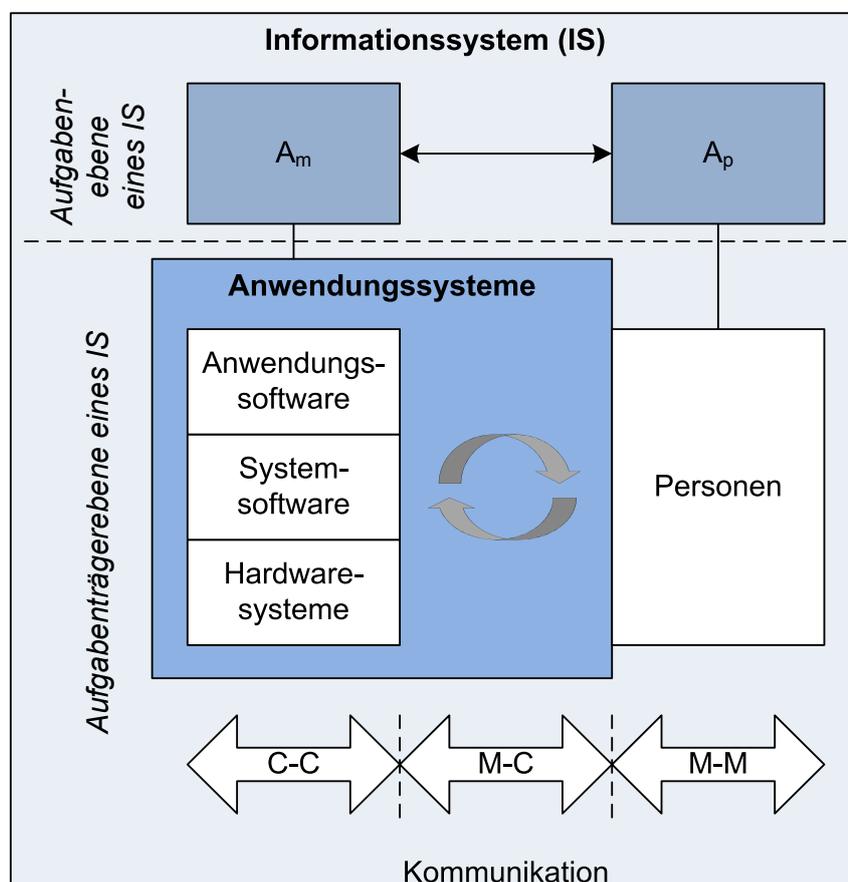


Abbildung 2.4: Informationsbeziehungen und Kommunikationssysteme im Informationssystem (in Anlehnung an Ferstl und Sinz (2013, S. 5))

In Abb. 2.4 werden Aufgaben auf Aufgabenebene durch Informationsbeziehungen und AT durch Kommunikationsbeziehungen miteinander verknüpft. Besteht zwischen Aufgaben mit unterschiedlichen Aufgabenträgern eine Informationsbeziehung, wird die korrespondierende Kommunikationsbeziehung durch einen Kommunikationskanal

zwischen den Aufgabenträgern realisiert (Ferstl und Sinz 2013, S. 5). Kommunikationskanäle werden durch die jeweils verwendeten Kommunikationssysteme bereitgestellt und können, wie in Abb. 2.4 dargestellt, unterschieden werden in: Kommunikation C-C zwischen Rechnern, Kommunikation M-C zwischen Personen und Rechnern und Kommunikation M-M zwischen Personen (Ferstl und Sinz 2013, S. 5-6).

AwS sind auf einzelne Aufgaben oder Aufgabenbereiche eines Informationssystems ausgelegt und bilden zusammen mit Personen und Kommunikationssystemen für die M-M-Kommunikation die eigentliche AT-Ebene eines Informationssystems (Ferstl und Sinz 2013, S. 6).

Wie aus Abb. 2.4 entnommen werden kann, besteht ein **AwS** neben der Anwendungssoftware, die die fachlichen Anforderungen erfüllt, auch aus der Systemsoftware sowie aus dem Hardwaresystem. Diese Trennung lässt sich auch mithilfe des Strukturmodells der **Nutzer- und Basismaschine** erklären (Ferstl und Sinz 2013, S. 318-321). Das AwS kann in diesem Fall als Nutzermaschine verstanden werden (Ferstl und Sinz 2013, S. 321). Die Nutzermaschine stellt die Außensicht dar. In der Innensicht besteht sie aus der Basismaschine und dem zugehörigen Programm (Ferstl und Sinz 2013, S. 318). Das Programm ist als Anwendungssoftware aufzufassen. Zur Realisierung der fachlichen Anforderungen werden die Datenobjekte und Operatoren der Basismaschine verwendet. Die Programmierplattformen Java und .NET können exemplarisch als Basismaschinen angesehen werden. So kann ein Programm, mit welchem Aufträge für die Produktion nach vorgegebenen Parametern eingelastet werden, unter Verwendung der Basismaschine Java erstellt werden. Beim Begriff des AwS werden unter der Systemsoftware sowohl höhere Ausbaustufen der Programmierplattformen als auch das Betriebssystem selbst subsumiert. Die mehrstufige Anordnung von Nutzer- und Basismaschinen erlaubt hier eine Untergliederung, in der auch das Hardwaresystem eingebunden werden kann. Bei der mehrstufigen Anordnung wird die zuvor genannte Basismaschine Java zum Programm. Dieses verwendet zur Realisierung Datenobjekte und Operatoren des Betriebssystems, welches nun als neue Basismaschine zu verstehen ist. Das Betriebssystem als Programm verwendet die Operatoren des Hardwaresystems bspw. den Befehlsvorrat des Prozessors. Dabei ist das Hardwaresystem als untergeordnete Basismaschine zu interpretieren.

Mit dem **ADK-Strukturmodell** kann ein AwS orthogonal zum Schichtenmodell der Nutzer- und Basismaschine in die Teilsysteme Anwendungsfunktionen (A), Datenverwaltung (D) und Kommunikation (K) aufgeteilt werden (Ferstl und Sinz 2013, S. 321-323). Diese beiden methodischen Hilfsmittel, das ADK-Strukturmodell sowie das Schichtenmodell der Nutzer- und Basismaschine, ermöglichen die Spezifikation einer Vielzahl von Architekturformen verteilter und nicht-verteilter AwS (Ferstl und Sinz 2013, S. 485-487). Dabei können diese beiden Hilfsmittel zur Trennung von

Zuständigkeiten und zur Bildung kohärenter Komponenten eingesetzt werden. Anhand einer dreischichtigen Web-Architektur wird dies stellvertretend erläutert (Ferstl und Sinz 2013, S. 493-495): Bei dieser übernimmt der Web-Client die Kommunikation mit Personen (K_P). Zur Kommunikation mit anderen Maschinen (K_M) werden standardisierte Schnittstellen zur Verfügung gestellt, die von maschinellen Clients aufgerufen werden können. Ein dedizierter Web-Server stellt die Anwendungsfunktionen (A) zur Verfügung und vermittelt zwischen Web-Client, externen maschinellen Clients und dem Daten-Server (D). Letzterer verwaltet die persistenten Daten. Die ADK-Teilsysteme werden ihrerseits wiederum in mehrstufige Nutzer- und Basismaschinen, wie oben ausgeführt, angeordnet.

Zur Modellierung der hier ausgeführten Strukturmodelle werden in der SOM-Methodik AwS in Form von Konzeptuellen Objektschemata (KOS) und darauf aufbauender Vorgangsobjektschemata (VOS) spezifiziert.²⁰ Dazu werden im KOS relevante Konzeptuelle Objekttypen (KOT) und ihre Beziehungen abgebildet. Mit einem KOT werden der Name, die Attribute, die Nachrichtendefinitionen sowie die Operatoren (Methoden) für einen Objekttyp festgelegt (Ferstl und Sinz 2013, S. 223-229). Mit dem VOS wird der Ablauf zur Durchführung von Aufgaben, die vom AwS auszuführen sind, beschrieben. Dazu werden die relevanten Vorgangsobjekttypen (VOT) im VOS eingebunden. VOT definieren den Algorithmus unter Verweis auf die jeweils heranzuziehenden KOT. Sie bestehen aus Namen, Attributen, Nachrichtendefinitionen sowie Operatoren (Ferstl und Sinz 2013, S. 230-233).

2.4 Einführung des erläuternden Fallbeispiels

In diesem Abschnitt wird das den Ausführungen der vorliegenden Arbeit zugrundeliegende Fallbeispiel eingeführt. Dieses stammt aus dem Umfeld von Hochschulen und fasst selbige als Dienstleistungsbetriebe auf (Krumbiegel 1997, S. 22-29). In der Leistungssicht, dargestellt in Abb. 2.5, werden die unterschiedlichen Haupt- und Serviceprozesse sichtbar.²¹ Einer der Hauptprozesse ist die Ausbildung von Studierenden – die markierte Transaktion *Hochschulausbildung (L)* in der Abb. 2.5. Im Rahmen dieses Hauptprozesses werden unter anderem Lehrinhalte in Form von Lehrveranstaltungen, Gastvorträgen oder ähnlichem vermittelt (Krumbiegel 1996b). Im Rahmen von Prüfungen wird festgestellt, ob der Studierende die vermittelten Inhalte erfasst hat und anwenden kann bzw. neues Wissen im Rahmen von Seminar- oder Abschlussarbeiten selbstständig erarbeiten kann. Somit soll der Studierende seine Leistungsfähigkeit nachweisen. Hat der Studierende im Rahmen seines von ihm gewählten Studienganges

²⁰Zur Vertiefung von KOS und VOS vgl. Ferstl und Sinz (2013, S. 223-233).

²¹Für eine ausführliche Beschreibung der Haupt- und Serviceprozesse sowie ihre Leistungsbeziehungen untereinander und mit der Umwelt sei stellvertretend auf Sinz (1998, S. 16-17) verwiesen.

die vorgeschriebene Anzahl an Leistungen erbracht, wird dies durch die Verleihung eines abschließenden Grades dokumentiert (Krumbiegel 1996a).

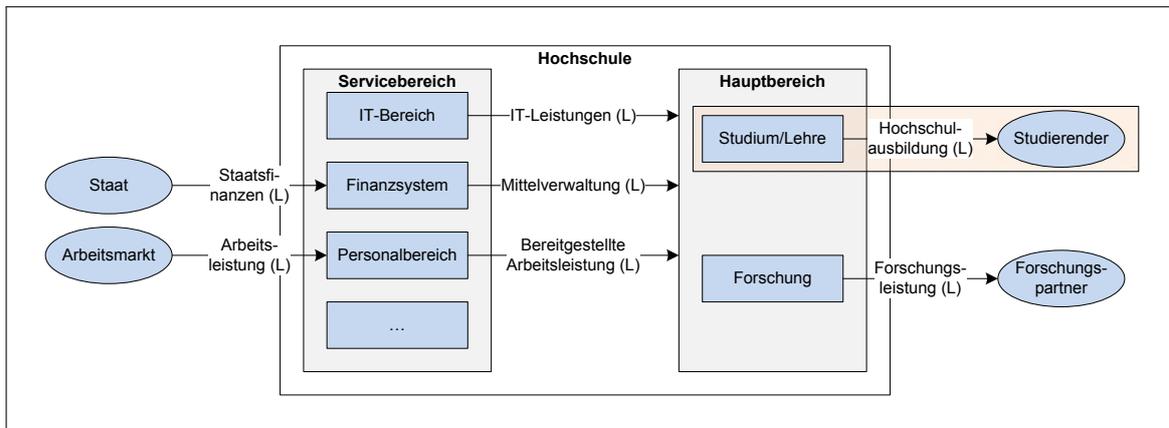


Abbildung 2.5: Haupt- und ausgewählte Serviceprozesse der Hochschule in der Leistungssicht (in Anlehnung an Sinz (1998, S. 16))

Die Hochschulausbildung wird arbeitsteilig durchgeführt, da die individuellen Kapazitäten einer einzelnen Person nicht ausreichen würde, um das für einen Studiengang vorgesehene Lehrangebot anzubieten sowie die Verwaltung der Studienverläufe übernehmen zu können. Aus diesem Grund vermitteln Lehrende die Lehrinhalte und eine in Hochschulen oftmals als Prüfungsamt bezeichnete Organisationseinheit übernimmt die Überwachung des Studienfortschritts von Studierenden. Das bedeutet, dass das Prüfungsamt unter anderem darüber wacht, ob der Studierende eine Leistung gemäß der für ihn gültigen Prüfungsordnung in Abstimmung mit den bislang erbrachten Leistungen zum Studienfortschritt einbringen darf. Dazu kontrolliert das Prüfungsamt bereits die Anmeldungen zu Prüfungen auf Konformität gegenüber der Prüfungsordnung. Somit werden nur diejenigen Studierenden zur Prüfungsteilnahme zugelassen, die aufgrund der für sie gültigen Prüfungsordnung die Leistung in den Studienverlauf einbringen dürfen.

Die Benachrichtigung über die angebotenen Prüfungen ist wiederum der Auslöser der Untersuchung, denn diese werden auf Grundlage von Lehrveranstaltungen erzeugt. Zu diesem Zweck melden die Lehrenden das Angebot an Lehrveranstaltungen sowohl dem Studierenden als auch dem Prüfungsamt.

Um die weiteren Untersuchungen genauer eingrenzen zu können, wird die in der Abb. 2.5 aggregierte Transaktion *Hochschulausbildung (L)* verfeinert. Dazu wird auf die von Sinz (1998, S. 18) veröffentlichte Lenkungssicht zurückgegriffen. Sie wird in Abb. 2.6 dargestellt. Aufgrund bisheriger Ausführungen kann der Bereich auf den in der Abb. 2.6 markierten eingegrenzt werden. Dessen Gegenstand sind die Teilprozesse Semesterausbildung und Prüfung, zusammengefasst in der Transaktion: *D (par): AusbildPrf (L)*. Die weiteren Ausführungen im Rahmen der vorliegenden

Arbeit beschränken sich auf diese Transaktion. Für eine ausführliche Beschreibung der übrigen Leistungen, Transaktionen und betrieblichen Objekte wird auf Sinz (1998, S. 18-19) verwiesen.

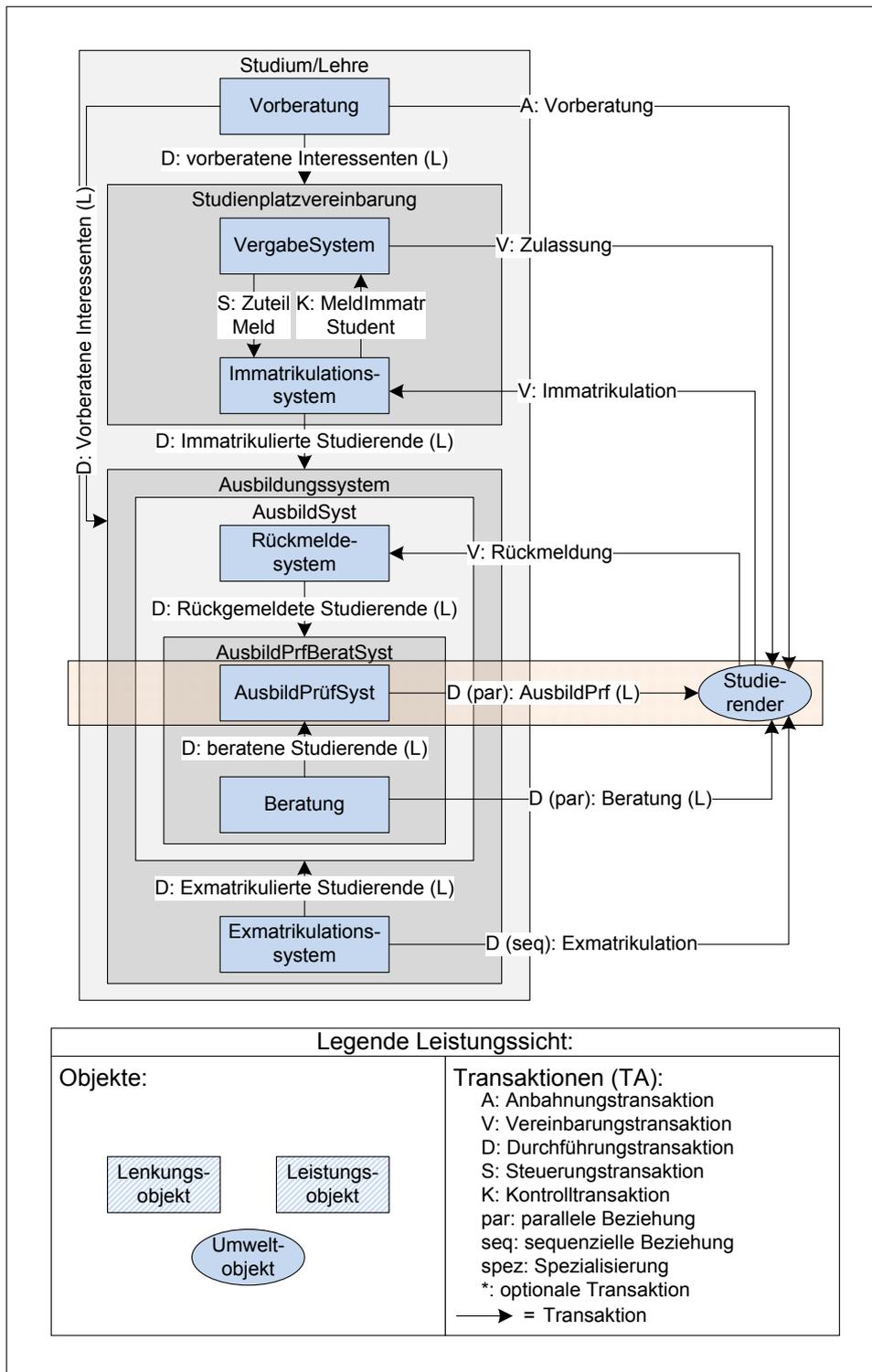


Abbildung 2.6: Eingrenzung der Untersuchung auf die Teilprozesse *Semesterausbildung* und *Prüfung* in der ausführlichen Lenkungs-sicht (in Anlehnung an Sinz (1998, S. 18))

3 Integration als Teilaufgabe bei der Gestaltung von Informationssystemarchitekturen

3.1 Integration von Aufgaben über Aufgabenträger	28
3.1.1 Das Analyse-Synthese-Konzept	28
3.1.2 Der Begriff der Heterogenität	30
3.1.3 Aufgabenintegrationsmuster zur Analyse der Aufgabenvernetzung	31
3.1.4 Das Konzept des betrieblichen Objekts zur Bildung von Aufgabenkomplexen	33
3.1.5 Kartografie zur Analyse von Abhängigkeiten zwischen Anwendungssystemen	34
3.1.6 Technische Lösungsverfahren zur Herstellung der Interoperabilität	38
3.1.7 Die Integrationsarchitektur als Teil des Informationssystems	39
3.2 Motivation für Integrationsprojekte	42
3.2.1 Auslöser von Integrationsprojekten	42
3.2.2 Ziele bei Integrationsprojekten	43
3.3 Integrationsmerkmale und Lösungskonzepte bei der Gestaltung von Integrationsarchitekturen	46
3.3.1 Integrationsmerkmale bei der Gestaltung von Integrationsarchitekturen	46
3.3.1.1 Redundanz	47
3.3.1.2 Verknüpfung	48
3.3.1.3 Konsistenz	48
3.3.1.4 Zielorientierung	49
3.3.1.5 Bezug der Integrationsziele auf Aufgaben- und Aufgabenträgerebene	49
3.3.2 Integrationskonzepte zur Gestaltung von Integrationsarchitekturen	50
3.3.2.1 Aufgabenträgerorientierte Funktionsintegration	50
3.3.2.2 Datenflussorientierte Funktionsintegration	51
3.3.2.3 Datenintegration	51
3.3.2.4 Objektintegration	52
3.3.2.5 Zielerreichungsgrade der Integrationskonzepte	53
3.4 Konzeption von Integrationsarchitekturen als Systementwicklungsaufgabe	54
3.4.1 Aufgabenmodell der Systementwicklung	54
3.4.2 Beschreibungsebenen bei der Entwicklung von Integrationsarchitekturen	54

Durch die Aufgabengliederung und die damit verbundene Arbeitsteilung entsteht bei arbeitsbezogenen Abhängigkeiten (Interdependenzen) der Bedarf nach Koordination zwischen den beteiligten AT (Kieser und Walgenbach 2010, S. 93-94). Unter Koordination wird im Allgemeinen die Abstimmung von Teilaufgaben im Hinblick auf ein übergeordnetes Ziel verstanden (Hentze et al. 2001, S. 160). In der betriebswirtschaftlichen Organisationslehre gibt es Instrumente, mit denen die Koordination zwischen personellen AT bewerkstelligt werden kann (Kieser und Walgenbach 2010, S. 100-127). In der vorliegenden Arbeit sind die jeweiligen Instrumente der betriebswirtschaftlichen Organisationslehre zur Koordination personeller AT jedoch von untergeordnetem Interesse. Hingegen ist auf organisatorischer Gestaltungsebene die Aufgabenintegration durch AT (Abschnitt 3.1) Untersuchungsgegenstand in diesem Kapitel. Darauf aufbauend werden in Abschnitt 3.2 die Beweggründe für die Durchführung von Integrationsprojekten, im Rahmen derer die Gestaltungsmaßnahmen implementiert werden, beschrieben. Anschließend werden in Abschnitt 3.3 Eigenschaften, die beim Entwurf von Integrationsarchitekturen zu berücksichtigen sind, definiert und Lösungskonzepte vorgestellt, die diese in unterschiedlichem Ausmaß unterstützen. Abschließend wird in Abschnitt 3.4 das Vorgehen bei der Konzeption von Integrationsarchitekturen als Systementwicklungsaufgabe vorgestellt.

3.1 Integration von Aufgaben über Aufgabenträger

In Abschnitt 2.3 wurde das Informationssystem als Teilsystem der Unternehmung eingeführt. Darauf aufbauend ist die Gestaltung der Integration von Aufgaben im betrieblichen Informationssystem Gegenstand dieses Abschnitts.

3.1.1 Das Analyse-Synthese-Konzept

Das Analyse-Synthese-Konzept kann als Teil zur Bildung der Aufbauorganisation aufgefasst werden. In der Aufbauorganisation werden die Zielsetzungen der Bereiche, Abteilungen und Stellen eines Unternehmens festgelegt, indem aus der Unternehmensaufgabe entsprechende Teilaufgaben abgeleitet (Gaitanides 1983, S. 2) und anschließend geeigneten AT zugeordnet werden (Kosiol 1976, S. 186-187). Im Gegensatz dazu sind Prozesse (Abschnitt 2.2.1) Teil bei der Gestaltung der Ablauforganisation (Wittlage 1998, S. 173). Erst nachdem der organisatorische Aufbau mittels Aufgabenanalyse und -synthese festgelegt ist, wird darauf aufbauend der organisatorische Ablauf des Unternehmens betrachtet.

Zur Durchführung der Gesamtaufgabe eines Unternehmens müssen i. d. R. mehrere AT zusammenarbeiten, da ab einer bestimmten Größenordnung die Gesamtaufgabe zu komplex wird, um von einem einzelnen AT vollständig durchgeführt werden zu können (Frese et al. 2012, S. 4). Dazu ist es erforderlich, die Gesamtaufgabe in Teilaufgaben

zu gliedern, um sie anschließend verantwortlichen AT zuzuweisen. Kosiol (1976, S. 32-33) hat zu diesem Zweck das Analyse-Synthese-Konzept vorgeschlagen. Dabei wird zunächst im Rahmen der Aufgabenanalyse, ausgehend von der Unternehmensaufgabe, eine fortschreitende Zerlegung in Teilaufgaben forciert. Mit der dadurch entstehenden mehrstufigen Hierarchie wird die Untersuchung des betrieblichen Handelns auf der dafür zweckmäßigen Aggregationsebene ermöglicht (Kosiol 1976, S. 42).

Zur Zerlegung von Aufgaben hat Kosiol (1976, S. 49) folgende sechs Gliederungsprinzipien vorgeschlagen: Objekt, Phase, Rang, Sachmittel, Verrichtung und Zweckbeziehung. Da eine Aufgabe nur anhand ihrer Verrichtungen oder ihrer Aufgabenobjekte beliebig oft hintereinander zerlegt werden kann (Kosiol 1976, S. 65), sind insbesondere für die Aufgabenanalyse und Organisationsgestaltung diese beiden Prinzipien relevant (Ferstl und Sinz 2013, S. 72-73). Beim Verrichtungsprinzip werden die Aufgaben anhand der Aufgabenziele und des Lösungsverfahrens differenziert. Bei der Gliederung nach dem Objektprinzip werden die Aufgaben sukzessive nach der Art der Aufgabenobjekte zerlegt (Kosiol 1976, S. 49-53; Ferstl und Sinz 2013, S. 72-73). Ziel dieser Prinzipien ist es, Aufgaben so lange zu zerlegen, bis sie genau einem gedachten AT zugeordnet werden können (Ferstl und Sinz 1984, S. 28).

Nach der Aufgabenzerlegung erfolgt die Aufgabensynthese. Bei dieser werden zusammenhängende Teilaufgaben einem AT zugeordnet (Kosiol 1976, S. 33). Dabei werden die durch die Aufgabensynthese gruppierten Aufgaben durch Stellen, im Falle der Zuordnung zu personellen AT, oder durch AwS, bei der Zuordnung zu maschinellen AT, zusammengefasst (Ferstl und Sinz 2013, S. 237). Bei der Zusammenstellung von Aufgaben für personelle AT ist darauf zu achten, dass der mögliche Aufgabenumfang durch die Arbeitskapazität des personellen AT begrenzt ist. Im Gegensatz dazu unterstützen maschinelle AT aufgrund ihrer variablen Kapazität Aufgaben sehr unterschiedlichen Umfangs (Ferstl und Sinz 1984, S. 28). Daher können bei der Zusammenstellung der Aufgaben für den maschinellen AT unter dem Gesichtspunkt der Integration andere Kriterien für die Zerlegung von Aufgaben herangezogen werden. So kann bspw. die Reduktion der Anzahl der Informationsbeziehungen zwischen Aufgaben ein wesentliches Kriterium sein (Ferstl und Sinz 1984, S. 28). Bei diesem Prinzip werden die Aufgaben so zerlegt, dass „der Umfang der Informationsbeziehungen zwischen den erzeugten Teilaufgaben kleiner oder gleich dem Umfang jeder anderen Zerlegung gleichen Detaillierungsgrades ist“ (Ferstl und Sinz 1984, S. 28). Durch diese Vorgehensweise können die Kosten der Datenübertragung minimiert, die Durchführung der Aufgaben beschleunigt und die Wahrscheinlichkeit für eine korrekte Aufgabendurchführung erhöht werden (Ferstl und Sinz 1984, S. 28). Insgesamt stellt die Aufgabensynthese als Vorgang eine Form der Integration dar, da Aufgaben in diesem Fall über einen AT integriert werden.

Die beschriebene Strukturierung unterstützt im Wesentlichen den Entwurf der Arbeitsteilung²² (Hentze et al. 2001, S. 154). Die Arbeitsteilung motiviert sich zum einen daraus, dass, wie eingangs geschildert, die Gesamtaufgabe häufig nicht mehr von einem einzelnen AT vollständig durchgeführt werden kann und daher auf mehrere AT aufzuteilen ist. Zum anderen können spezialisierte Stellen geschaffen werden, um dadurch die Steigerung der Produktivität voranzutreiben (A. Picot et al. 2008, S. 2).

3.1.2 Der Begriff der Heterogenität

Durch die Aufgabenverteilung auf unterschiedliche AT entsteht Heterogenität. Heterogenität bezeichnet im Allgemeinen Ungleichartigkeit, Verschiedenartigkeit oder Uneinheitlichkeit (Dudenredaktion 2008, S. 401) und kann bspw. durch die Zuordnung von Aufgaben auf unterschiedliche AT, wie in Abschnitt 3.1.1 beschrieben, entstehen. Im Fokus der vorliegenden Arbeit steht die Heterogenität im Informationssystem, insbesondere von Informationssystemen, in denen unterschiedliche AwS eingesetzt werden. Daneben existieren zahlreiche weitere Arten der Heterogenität im Unternehmen, welche von heterogenen Organisationsformen bis hin zu heterogener Hardware reichen können, jedoch nicht Bestandteil dieser Arbeit sind.

Informationssysteme, in denen heterogene AwS eingesetzt werden, sind häufig historisch gewachsen (Ferstl und Sinz 2013, S. 260). Begünstigt wird diese Entwicklung dadurch, dass integrierte AwS eines Herstellers oftmals nicht vollständig alle erforderlichen Aufgabenbereiche einer Organisation abdecken können (Sinz und Wismans 2001, S. 27) und aus diesem Grund vielfach unterschiedliche AwS im Informationssystem eines Unternehmens anzutreffen sind. Darüber hinaus wird sich die Bildung von Informationssystemen mit heterogenen AwS wohl nie vollständig vermeiden lassen, wenn ein Unternehmen bspw. neue Technologien einsetzen möchte. Entscheidungen, die zulasten von Vereinheitlichung in Kauf genommen werden, bieten jedoch Vorteile bei der Unabhängigkeit gegenüber Herstellern (Winter und Schelp 2006, S. 22-23).

Heterogenität im Informationssystem kann in syntaktische, semantische und strukturelle Heterogenität unterteilt werden:

- **Syntaktische Heterogenität** kann differenziert werden in technische Heterogenität und Heterogenität bei den Schnittstellen (Bauer et al. 2002, S. 4). Technische Heterogenität entsteht aufgrund von vielzähligen parallel existierenden Technologien, wie unterschiedlichen Programmiersprachen (Hagen 2004, S. 68). Heterogenität bei den Schnittstellen entsteht, wenn diese andersartige Sprachen

²²Unter Arbeitsteilung wird im Rahmen dieser Arbeit die „Aufteilung von Aufgaben auf Aktionsträger“, wie sie Grochla (1982, S. 25) definiert, verstanden.

für den Zugriff anbieten. Hierzu zählen bspw. unterschiedliche SQL-Dialekte (Bauer et al. 2002, S. 4).

- **Semantische Heterogenität** bezeichnet das ungleiche Verständnis über die Bedeutung und Interpretation (Hergula und Härder 1999, S. 5-6). Hierbei kann zwischen Intension und Extension eines Begriffswortes unterschieden werden (Ortner 1997, S. 30).²³ Zu den Problembereichen dieser Form zählen Namenskonflikte, insbesondere der Umgang mit Synonymen und Homonymen (Rosemann 1996, S. 187).²⁴
- **Strukturelle Heterogenität** kann durch unterschiedliche Arten der Repräsentation entstehen (Hergula und Härder 1999, S. 6) und verletzt dadurch den Grundsatz der Richtigkeit, insbesondere der Konsistenz (Rosemann 1996, S. 216). Als Beispiel hierzu zählt die Abbildung eines Ausschnittes aus der Realwelt in einem Modell mit verschiedenen Modellierungsansätzen (Bauer et al. 2002, S. 5).

3.1.3 Aufgabenintegrationsmuster zur Analyse der Aufgabenvernetzung

Aufgaben können untereinander in Beziehung stehen. Zur Beschreibung des Ablaufs vernetzter Aufgaben wurde in Abschnitt 2.2.1 der Prozessbegriff eingeführt. Mithilfe von Aufgabenintegrationsmustern (AIM) können Aufgabenbeziehungen zu Analyse Zwecken in ihre Einzelbestandteile zerlegt und dadurch die Aufgabensynthese unterstützt werden.

Die AIM sind ein Teilergebnis des Forschungsprojekts OASYS²⁵ (Mantel et al. 2004, S. 12-13; Eckert et al. 2003, S. 99; Schissler et al. 2002, S. 460). Nachfolgend werden die vier elementaren AIM erklärt und korrespondierend dazu in Abb. 3.1 schematisch dargestellt.

Bei dem AIM *Reihenfolgebeziehung zwischen Aufgabendurchführungen* ist das Nachereignis einer Aufgabe gleichzeitig das Vorereignis einer anderen. In Abb. 3.1 wird die Identifikation dieses Musters durch die Hervorhebung des Ereignisses dargestellt.

²³Begriffe werden durch Begriffsworte beschrieben und haben eine Intension (Inhalt) und Extension (Umfang). Die Intension eines Begriffes enthält Angaben zu verschiedenen Kriterien, anhand derer geprüft werden kann, ob ein Objekt zu einem Begriff gehört oder nicht. Die Extension bildet die Menge der Objekte, die unter dem Begriff subsumiert werden (Ortner 1997, S. 30).

²⁴Unter Synonymen werden unterschiedliche Bezeichner mit der gleichen Bedeutung verstanden. Im Gegensatz dazu sind unter Homonymen gleiche Bezeichner mit verschiedener Bedeutung zu verstehen. Hierzu wird oftmals der Begriff der „Bank“, unter der, je nach Kontext, die Sitzbank oder die Bank als Organisation gemeint ist, als Beispiel herangezogen (Rosemann 1996, S. 188-189).

²⁵Das Projekt „Offene Anwendungssystem-Architekturen in überbetrieblichen Wertschöpfungsketten“ (OASYS) war ein Teilprojekt des bayerischen Forschungsverbundes Wirtschaftsinformatik (FORWIN) und wurde im Zeitraum 2000 bis 2005 an der Universität Bamberg durchgeführt. Die Ergebnisse können unter <https://sedaintra.seda.wiai.uni-bamberg.de/forwin/> eingesehen werden.

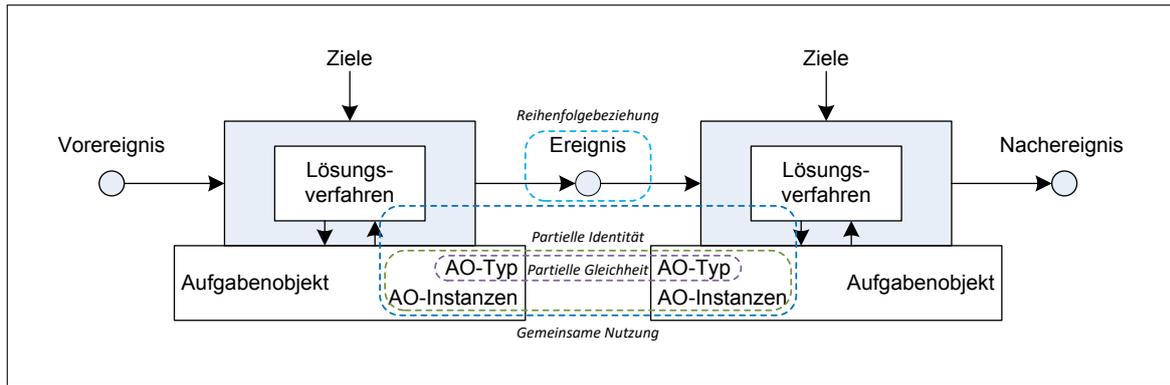


Abbildung 3.1: Aufgaben und ihre Beziehungen (in Anlehnung an Schissler et al. 2002, S. 460)

Das AO beschreibt die zu einer Aufgabe gehörenden Attribute (AO-Typ) und Werte (AO-Instanz). Das Muster *Partielle Gleichheit von AO-Typen* und in weiterer Ausprägung *Partielle Identität von AO-Instanzen* tritt auf, wenn sich die AO zweier Aufgaben auf Typ- oder sogar auf Instanzebene zumindest teilweise überlappen. In Abb. 3.1 wird das Muster *Partielle Gleichheit von AO-Typen* durch die Umrahmung selbiger veranschaulicht. Bei dieser Beziehungsart wurden die AO nicht disjunkt zerlegt (Ferstl 1992, S. 8). Das Muster *Partielle Identität von AO-Instanzen* wird durch die Umrahmung der AO-Instanzen bei beiden AO verdeutlicht. Um die überlappenden Anteile erkenntlich zu machen, schlägt Kiehl (2012, S. 73) vor, die Teilgraphen des für den GP geltenden KOS den jeweiligen betrieblichen Objekten zuzuordnen.

Das Muster *Gemeinsame Nutzung von (Teil-) Lösungsverfahren* kennzeichnet die partielle Gleichheit von Lösungsverfahren einer Aufgabe. Dies kann sich durch gleiche (Teil-) Vorgänge in den zu vergleichenden Aufgaben äußern. Bei diesem AIM überlappen sich auch die AO der beiden Aufgaben – dargestellt in Abb. 3.1 durch die Umrahmung der AO-Typen, AO-Instanzen und den Teilrahmen bei den Lösungsverfahren. Um die gemeinsame Nutzung von (Teil-) Lösungsverfahren herauszufinden, bietet es sich an, den Workflow der betroffenen Aufgaben mithilfe von VOS bspw. in der Business Process Model and Notation (BPMN) zu modellieren und diese anschließend auf redundante Aktivitäten zu vergleichen.

Da Aufgaben prinzipiell auch über mehr als ein AIM miteinander in Beziehung stehen können, dürfen die elementaren AIM zusammengesetzt werden.

Von identifizierten AIM geht ein Integrationsbedarf aus, der über die AT-Ebene entsprechend abzudecken ist.

3.1.4 Das Konzept des betrieblichen Objekts zur Bildung von Aufgabenkomplexen

Mit den in Abschnitt 3.1.3 vorgestellten AIM können Aufgabenbeziehungen analysiert werden. Stellt sich bei der Analyse heraus, dass es Aufgaben gibt, die einen hohen Grad an Überlappung aufweisen, können diese zu einem *betrieblichen Objekt* zusammengefasst werden. Durch das Konzept des betrieblichen Objekts können die nach dem Verrichtungs- oder dem Objektprinzip zerlegten Aufgaben zu Aufgaben mit gleichartigen Arbeitsabläufen oder AO zusammengefasst werden (Ferstl und Sinz 2013, S. 72-73). Mithilfe dieser Zusammenfassung wird die Aufgabenintegration unterstützt und es werden Aufgabenkomplexe gebildet.

Das Konzept des betrieblichen Objekts ermöglicht es, ein verteiltes System autonomer und loser gekoppelter Komponenten zu realisieren. Sie werden mithilfe von Transaktionen in Bezug auf ein gemeinsames Ziel koordiniert (Ferstl und Sinz 2013, S. 202). Mit dem Begriff des betrieblichen Objekts wird das allgemeine Konzept der Objektorientierung auf die Bildung betrieblicher Aufgabenstrukturen übertragen (Ferstl und Sinz 2013, S. 203). Mithilfe betrieblicher Objekte kann den aus der Informatik bekannten Forderungen nach *Kohäsion* (Myers 1978, S. 29-39; Schach 2011, S. 187-192) und *Kopplung* (Myers 1978, S. 41-54; Schach 2011, S. 192-199) nachgekommen werden. Die Forderung nach Kohäsion wird erfüllt, indem zusammenhängende Aufgaben in einem betrieblichen Objekt zusammengefasst werden. Der Forderung nach Kopplung wird nachgekommen, indem betriebliche Objekte lose miteinander gekoppelt werden.

In einem betrieblichen Objekt werden diejenigen Aufgaben zusammengefasst, die zusammengehörige Sach- und Formalziele aufweisen und auf einem gemeinsamen AO operieren (Ferstl und Sinz 2013, S. 203). Zusammengehörige Sach- und Formalziele entstehen durch eine nicht disjunkte Zerlegung der Aufgabenziele (Ferstl und Sinz 2013, S. 237). Dies äußert sich durch das Auftreten des AIM *Gemeinsame Nutzung von (Teil-) Lösungsverfahren*. Die nicht disjunkte Zerlegung des AO macht sich durch überlappende AO bemerkbar (Ferstl und Sinz 2013, S. 237), nachgewiesen durch das AIM *Partielle Gleichheit von AO-Typen*. Da diejenigen Aufgaben in einem betrieblichen Objekt zusammenzufassen sind, die auf gemeinsamen AO operieren, ist über das AIM *Partielle Gleichheit von AO-Typen* hinaus das AIM *Partielle Identität von AO-Instanzen* festzustellen.

In Abb. 3.2 wird das Konzept des betrieblichen Objekts schematisch dargestellt. Dabei wird ersichtlich, dass es auf dem in Abschnitt 2.2 eingeführten Aufgabenbegriff aufbaut; erkennbar sowohl durch das Vor- und Nachereignis als auch die Sach- und Formalziele sowie das Lösungsverfahren. Das AO einer Aufgabe wird im objektinternen Speicher eines betrieblichen Objekts repräsentiert. Betriebliche Objekte werden

durch Transaktionen mit anderen betrieblichen Objekten über Interaktionskanäle lose gekoppelt. Innerhalb eines betrieblichen Objekts liegen Vor- und Nachereignisse als objektinterne Ereignisse vor und koppeln so die objektinternen Aufgaben. Zur Durchführung der in einem betrieblichen Objekt vereinten Aufgabenkomplexe wird diesem ein personeller AT, eine Rolle oder eine Organisationseinheit zugeordnet. Sofern automatisierbare Aufgabenanteile vorhanden sind, wird maximal ein AwS als maschineller AT zugewiesen (Krumbiegel 1997, S. 138).

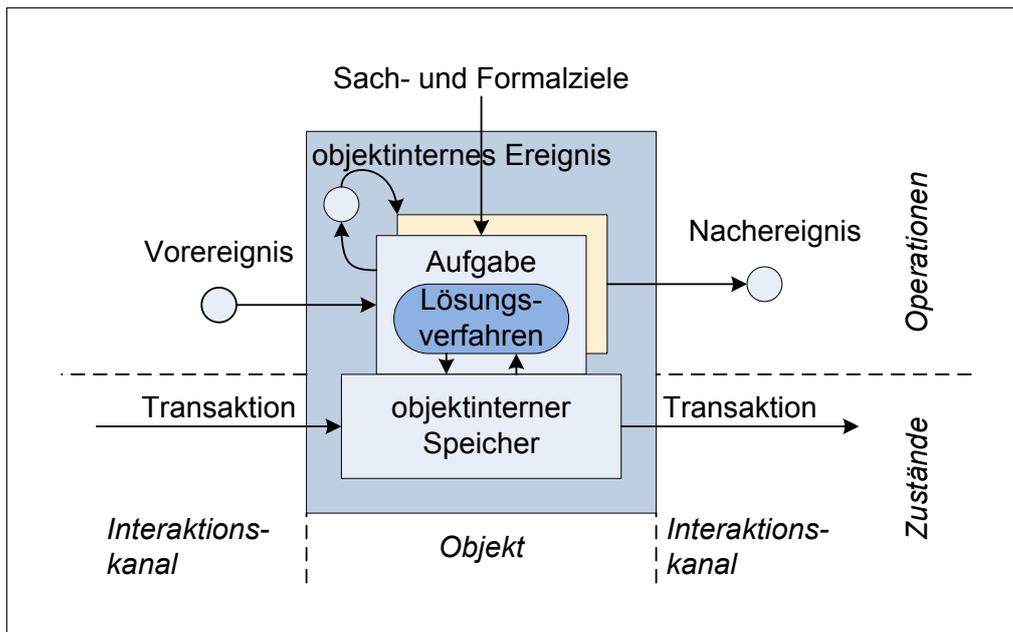


Abbildung 3.2: Schematische Darstellung des objektorientierten Konzepts betrieblicher Objekte (in Anlehnung an Ferstl und Sinz (2013, S. 202))

3.1.5 Kartografie zur Analyse von Abhängigkeiten zwischen Anwendungssystemen

Die AwS-Kartierung kann auf AT-Ebene nutzbringend eingesetzt werden, um Abhängigkeiten zwischen bereits vorhandenen AwS zu analysieren und darauf aufbauend den Integrationsvorgang zu unterstützen. Krumbiegel hat dazu vorgeschlagen, AwS in Geschäftsprozessmodellen zu verzeichnen (Krumbiegel 1997, S. 137-139). Die Erarbeitung eines grundlegenden Begriffsapparates zur Beschreibung von AwS-Landschaften sowie die Entwicklung eines Modells für Softwarekarten und deren adäquate grafische Repräsentation waren Ziele des Forschungsprojekts SoCa²⁶.

²⁶Das Projekt „Softwarekartographie“ (SoCa) war ein Projekt, das von 2003 bis 2007 an der TU München am Lehrstuhl für Software Engineering for Business Information Systems (sebis) durchgeführt wurde. Die Ergebnisse können unter <https://www.matthes.in.tum.de/pages/ae99wcdanm0v/sebis%20Public%20Website/Research/Completed%20Research%20Projects/SoCa> eingesehen werden.

Mithilfe der AwS-Kartografie sollen AwS-Landschaften und die relevanten Aspekte des maschinellen Anteils betrieblicher Informationssysteme visualisiert werden (Matthes und Wittenburg 2004b, S. 71). Die AwS-Kartierung ist folglich eine grafische Verortung der im GP eingesetzten AwS. Sie setzt sich zusammen aus einem Kartengrund und den auf dem Kartengrund aufbauenden Sichten. Sichten sind spezielle Ausprägungen einer AwS-Karte, auf der bestimmte Aspekte visualisiert werden können. Verschiedene Sichten können kombiniert werden, um andere AwS-Karten zu erzeugen (Matthes und Wittenburg 2004a, S. 4).

In AwS-Karten können Beziehungen zwischen verschiedenen AwS hinsichtlich der fachlichen Aufgaben und ihrer technischen Realisierung untersucht werden (Matthes und Wittenburg 2004b, S. 71). Dazu können als Kartengrund die betrieblichen Objekte und ihre Beziehungen untereinander verwendet werden. In einer darauf aufbauenden Anwendungssystemzuordnungssicht sind die jeweils unterstützenden AwS zu verorten (Krumbiegel 1997, S. 137-139). Neben dieser statischen Analyse ist eine dynamische Analyse, die die Evolution der Anwendungslandschaft berücksichtigt, von ähnlicher Bedeutung. Mit der dynamischen Analyse kann untersucht werden, wie sich die Modifikation von Zielen und GP auf das bestehende Informationssystem und insbesondere auf die AwS-Landschaft auswirken. Diese Untersuchungen können hinsichtlich planerischer, wirtschaftlicher, fachlicher, technischer und operativer Merkmale durchgeführt werden (Matthes und Wittenburg 2004a, S. 10; Matthes und Wittenburg 2004b, S. 72; Wittenburg 2007, S. 74).

Planerische Merkmale erfüllen Anforderungen, die sich durch die Evolution der AwS-Landschaft über die Zeit ergeben. So können die Auswirkungen, die Projekte auf die AwS-Landschaft haben, in der AwS-Karte abgebildet werden, um bspw. Abhängigkeiten, die Projekte untereinander haben können, zu erkennen (Matthes und Wittenburg 2004a, S. 10-11).

Mithilfe *wirtschaftlicher Merkmale* werden die verschiedenen Kostenarten erfasst, die bei der Entwicklung, dem Betrieb, der Wartung, dem Einkauf etc. von Informationssystemen entstanden sind bzw. entstehen werden. Dadurch kann das IT-Controlling unterstützt werden (Matthes und Wittenburg 2004a, S. 12).

Fachliche Merkmale beziehen sich auf die Aufbau- und Ablauforganisation. In AwS-Karten werden indirekt durch die Zuordnung der AwS zu den betrieblichen Objekten auch die fachlich verantwortlichen und nutzenden Organisationseinheiten festgelegt. Daneben können AwS in operative, dispositive Systeme etc. differenziert werden (Matthes und Wittenburg 2004a, S. 11).

Technische Merkmale umfassen die Sprache, in der ein AwS implementiert wurde, die Schnittstellen sowie seine Eigenschaften, wie Architektur oder genutzte Middleware²⁷ (Matthes und Wittenburg 2004a, S. 11).

Operative Merkmale beziehen sich auf den Betrieb von AwS. So kann bspw. unter dem Betriebsort die geographische Position des physikalischen Servers, auf dem das AwS gehostet wird, erfasst werden. Dieser Betriebsort kann unter Umständen vom Nutzungsort abweichen. Ein anderer operativer Aspekt ist der Ablauf von Batch-Programmen. Bei diesen gilt es zu erfassen, ob diese voneinander abhängig oder zeitlich versetzt laufen müssen. Derartige Abhängigkeiten und Abläufe in AwS-Landschaften können bspw. unter Nutzung von Gantt-Diagrammen visualisiert werden (Matthes und Wittenburg 2004a, S. 12).

Grundsätzlich ist zwischen den relevanten, den erfassbaren und pflegbaren Merkmalen zu differenzieren. Die Erhebung der für die Merkmale erforderlichen Informationen ist mit Aufwand und Kosten verbunden. Deshalb ist darauf zu achten, dass nur die Informationen erhoben werden, die einen entsprechenden Mehrwert bieten (Matthes und Wittenburg 2004b, S. 72).

Da die AwS eines Unternehmens nicht autark eingesetzt werden, sondern untereinander interagieren, können die vorhandenen Beziehungen der in einer AwS-Landschaft verzeichneten AwS zum Zwecke der Verzeichnung vorhandener Interaktionen herangezogen werden. Durch die Verzeichnung von AwS in GP können darauf aufbauend bislang noch nicht von den AwS unterstützte Transaktionen aufgedeckt werden. Durch den Abgleich mit den an die Transaktionen gestellten Automatisierungsanforderungen können Bedarfe in Bezug auf die Integration ermittelt werden. Daraus lassen sich zu implementierende Schnittstellen ableiten.

In Abb. 3.3 wird eine exemplarische AwS-Zuordnung dargestellt.²⁸ Diese orientiert sich am Vorschlag von Krumbiegel. Zur Darstellung wird als Kartengrund das in SOM erstellte Interaktionsschema (IAS) genutzt (Krumbiegel 1997, S. 137-139). Bei der vorgestellten AwS-Zuordnung werden den betrieblichen Objekten die unterstützenden AwS zugeordnet. Die Aufgaben im Rahmen des Semesterausbildungssystems (*SemAusbildSyst*) werden durch das AwS *Virtueller Campus* unterstützt. Die Aufgaben, die im betrieblichen Objekt Prüfungssystem (*PrüfungSyst*) gekapselt sind, werden vom AwS *FlexNow* unterstützt.

²⁷Der Begriff der Middleware wird in Abschnitt 3.1.6 eingeführt.

²⁸Für die exemplarische AwS-Zuordnung wird das in Abschnitt 2.4 eingeführte Fallbeispiel aufgegriffen. Die Beschreibung des zugrundeliegenden GP ist aus Abschnitt 6.3.2 zu entnehmen. Das korrespondierende Protokoll für die Objekt- und Transaktionszerlegung kann in Tab. B.2 nachgeschlagen werden.

Aus Abb. 3.3 wird ersichtlich, dass die Transaktion *D: MeldLvAngeb* gegenwärtig durch die beteiligten AwS nicht unterstützt wird, obwohl die Anforderungen an die Transaktion einen Bedarf an Unterstützung postulieren. Um dieses Delta zu schließen, können bspw. die involvierten AwS zur Unterstützung der betroffenen Transaktion miteinander gekoppelt werden. Dazu sind gegebenenfalls Schnittstellen in den AwS zu schaffen.

Neben der Visualisierung in Abb. 3.3 sind alternative Kartentypen mit Kartengrund bspw. Clusterkarten, Prozessunterstützungskarten oder Intervallkarten möglich (Lankes et al. 2005, S. 1450-1455; Lankes et al. 2006, S. 318-324). Daneben gibt es auch Karten ohne Kartengrund (Lankes et al. 2005, S. 1455-1457; Lankes et al. 2006, S. 324-325).

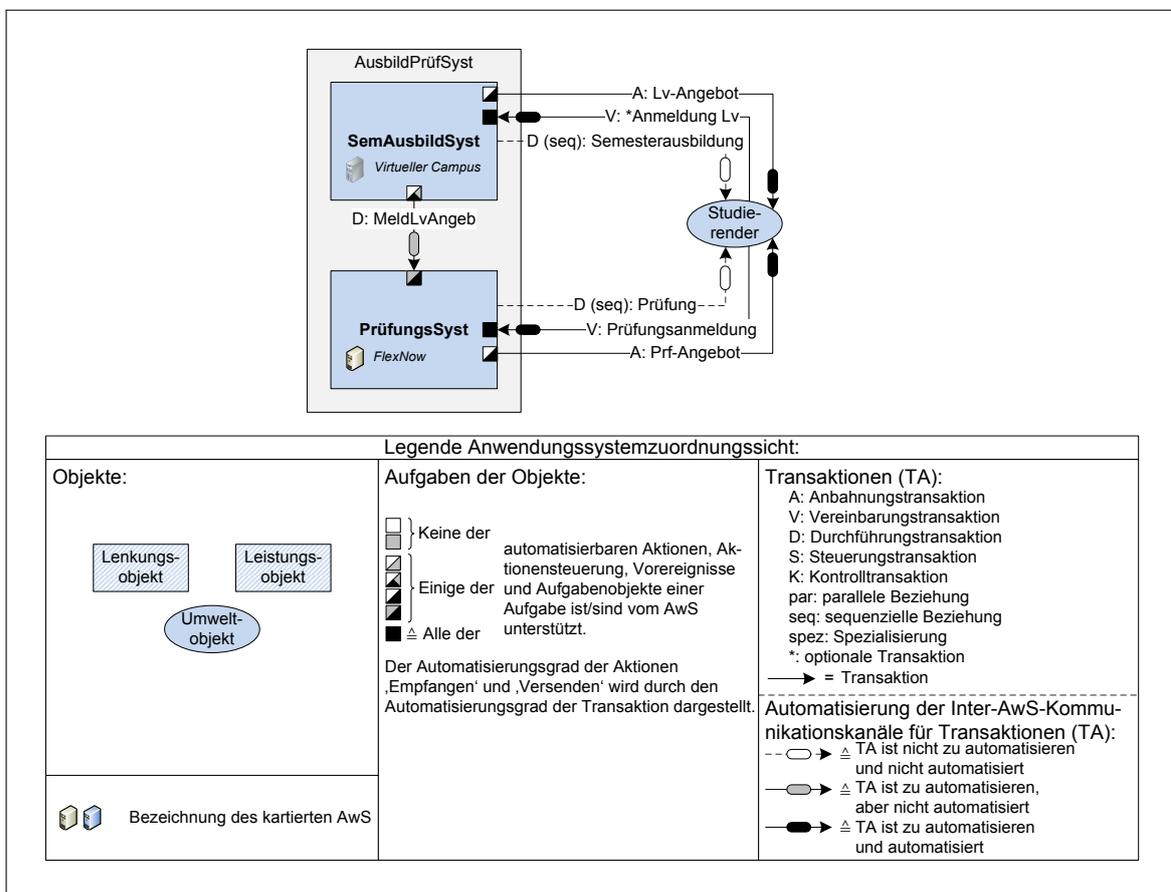


Abbildung 3.3: Beispiel einer Anwendungssystemzuordnungssicht

3.1.6 Technische Lösungsverfahren zur Herstellung der Interoperabilität

Die Überwindung des in Abschnitt 3.1.5 identifizierten Deltas auf technischer Ebene ist Gegenstand dieses Abschnitts.

Situationen, in denen AwS zur Unterstützung der Transaktionen zwischen betrieblichen Objekten zu koppeln sind, erfordern technische Lösungsverfahren zur Herstellung der Interoperabilität. Dabei gilt es zunächst die Kommunikation der beteiligten AwS zu ermöglichen, falls dies die Basismaschinen der betroffenen AwS noch nicht erlauben.²⁹

Kommunikation ist in diesem Zusammenhang aufzufassen als der physische Versand von Daten bzw. von Nachrichten (Riehm 1997, S. 56). Mittels eines Kopplungsmechanismus kann die Kommunikation hergestellt werden. Dies kann bspw. erreicht werden, indem die Programmierplattform, die in diesem Fall die Basismaschine darstellt, um Bibliotheken oder Frameworks, mithilfe derer ein physischer Versand ermöglicht oder vereinfacht werden kann, erweitert wird. Um bspw. mit der Programmierplattform Java einen REST Webservice nutzen zu können, ist eine Bibliothek einzubinden, welche die HTTP(S) Methoden wie GET und POST implementiert. Der Kopplungsmechanismus für die Kommunikation ist hierbei HTTP(S).

Die Klasse systemnaher Software, die als Bindeglied zwischen Systemsoftware und Anwendungsprogrammen fungiert, wird unter dem Begriff Middleware subsumiert (Ferstl und Sinz 2013, S. 430). Unter diesem wird „eine Softwareschicht verstanden, welche Dienstleistungen für die Integration in einer verteilten, heterogenen Umgebung erbringt“ (Riehm und Vogler 1996, S. 27) und hierfür verschiedene Kopplungsmechanismen bereitstellt. Ein standardisierter Kopplungsmechanismus zur Unterstützung bei der Implementierung verteilter AwS ist eine der Kernfunktionen, die von Middleware geleistet wird. Eine weitere mit Middleware assoziierte Kernfunktion besteht darin, Schnittstellen zur Verfügung zu stellen, die es Entwicklern ermöglichen, portierbare Anwendungssoftware zu entwickeln, die unabhängig von Hardware- und Betriebssystemen sowie Netzwerken und technischen Kommunikationsprotokollen ist (Ferstl und Sinz 2013, S. 430-431).

Ziel ist es, Kommunikationskanäle zwischen den zu koppelnden AwS mittels geeigneter Middleware zu etablieren. Hierfür bieten verfügbare Middleware-Produkte unterschiedliche Kopplungsmechanismen, von denen stellvertretend nachfolgend einige genannt werden, an. Für den Aufruf entfernter Daten bieten sich die Technologien ODBC und JDBC an. Sie ermöglichen die Kommunikation über einen gemeinsamen Datenspeicher und führen zur engen Kopplung. Mittels CORBA, Java RMI oder .NET Remoting können AwS lose gekoppelt werden. Für weitere Kopplungsmechanis-

²⁹Das Strukturmodell Nutzer-/Basismaschine wird in Abschnitt 2.3 eingeführt.

men, deren spezifische Einsatzfelder sowie deren Anwendung ist die entsprechende Fachliteratur heranzuziehen (bspw. Tanenbaum und Steen 2008).

Sobald Kommunikationskanäle für die beteiligten AwS mittels eines der Kopplungsmechanismen etabliert wurden, ist in einem darauf aufbauenden Schritt die Interoperabilität in Bezug auf das gemeinsame Verständnis auszutauschender Daten herzustellen. Dies bedeutet, ein gegenseitiges Verständnis der Syntax und Semantik der Daten und Funktionalität des jeweils anderen AwS zu bilden (Riehm 1997, S. 40). Damit wird die in Abschnitt 3.1.2 angesprochene syntaktische Heterogenität adressiert. Mit WSDL bspw. ist die Spezifikation der Syntax von Schnittstellen möglich, da mittels dieser die Datentypen der Attribute definiert werden können. Zur Überwindung der semantischen Heterogenität ist entweder die Dokumentation der Schnittstelle heranzuziehen, die an der Erstellung der Schnittstellen beteiligten Akteure stimmen sich untereinander ab oder mittels Reverse Engineering wird geprüft, ob die Eingabe bestimmter Parameter zum unterstellten Verhalten des AwS führt.

Als Alternative zur Schaffung von Kommunikationskanälen zwischen den beteiligten AwS bietet sich die Integration auf Präsentationsebene, bspw. mittels „Screen-Scraper“ (Noffsinger et al. 1998) oder Portal-Integration, an.

3.1.7 Die Integrationsarchitektur als Teil des Informationssystems

Winter und R. Fischer (2007, S. 2-3) teilen die Unternehmensarchitektur in verschiedene Ebenen mit unterschiedlichen Gestaltungsobjekten ein. Auf der Integrationsebene befinden sich die Artefakte zum Entwurf der Integrationsarchitektur. Diese Ebene legt bei der Gestaltung der Integration den Fokus ausschließlich auf die Kopplung maschineller AT. Die Kommunikation personeller AT wird mit dem Gestaltungsobjekt *Informationsflüsse* auf der Organisationsebene adressiert. Das Informationssystem wird nicht in der vorgestellten Unternehmensarchitektur eingeordnet.

Der Begriff der Integrationsarchitektur wird in der vorliegenden Arbeit, im Gegensatz zu Winter und R. Fischer (2007, S. 2-3), umfassender interpretiert. Dazu wird die Integrationsarchitektur in den Kontext der Unternehmensarchitektur eingeordnet. Ausgangspunkt für die Einordnung ist das Grundmodell der Unternehmung von Grochla (1975, S. 12-13). Dieses Grundmodell besteht zum einen aus einem Informationssystem zur Planung, Steuerung und Kontrolle und zum anderen aus einem Basissystem für die betriebliche Leistungserstellung. Beeinflusst werden beide Teilsysteme von der Umwelt der Unternehmung und zusammengefasst in der Unternehmensarchitektur.

Bei der Gestaltung der Integration von Aufgaben über die AT-Ebene werden sowohl die Integration über maschinelle AT als auch über personelle AT sowie eine Kombination von beidem als Gestaltungsmöglichkeiten berücksichtigt – zur beispielhaften

grafischen Veranschaulichung vgl. Abb. 1.1. Das Ergebnis dieses Entwurfsvorganges für alle Aufgaben im Informationssystem ist die Integrationsarchitektur für dieses. Die Integrationsarchitektur ist folglich eine Sicht auf das Informationssystem und das daraus zu gestaltende Teilsystem. In Abb. 3.4 werden die Beziehungen zwischen Unternehmensarchitektur, Informationssystem, Leistungssystem und Integrationsarchitektur in einem Modell grafisch veranschaulicht.

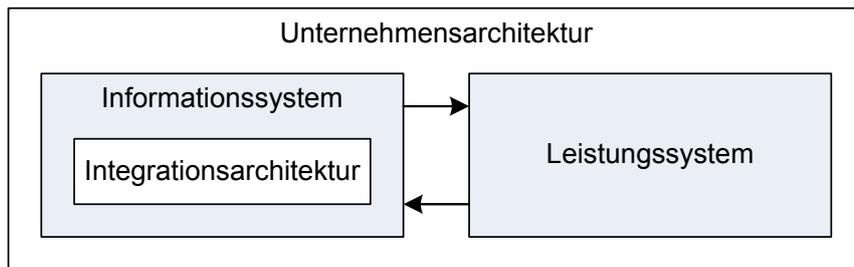


Abbildung 3.4: Die Integrationsarchitektur als Bestandteil der Unternehmensarchitektur

Zur Erstellung der Integrationsarchitektur werden zunächst im Rahmen der Aufgabensynthese Aufgaben zu zusammengehörenden Aufgabenkomplexen zusammengefasst (Abschnitt 3.1.1 und Abschnitt 3.1.4) sowie den entsprechenden AT zugeordnet (Abschnitt 3.1.5). Simultan dazu ist die Reihenfolge der Aufgabendurchführungen festzulegen (Abschnitt 2.2.1) und die entstehenden Aufgabenvernetzungen (Abschnitt 3.1.3) auf AT-Ebene mittels Implementierung von Kommunikationskanälen (Abschnitt 2.3) zu unterstützen. Die Interoperabilität (Abschnitt 3.1.6) ist dabei als Voraussetzung zur Kooperation maschineller AT zu verstehen. Mit ihr werden im Rahmen der Gestaltung der Integrationsarchitektur die in Abb. 2.4 definierten C-C Kommunikationskanäle implementiert. Ein weiterer Bestandteil bei der Integration von Aufgaben über AT ist die Kollaboration involvierter Personen. Zur Abbildung der Informationsbeziehungen werden korrespondierende M-M Kommunikationskanäle etabliert. Der M-C Kommunikationskanal dient zur Ausführung von Aufgaben, die zur Zielerreichung kooperativ sowohl von personellen als auch von maschinellen AT durchzuführen sind. Da M-C Kommunikationskanäle die Durchführung einzelner Aufgaben adressieren, gehören sie nicht zum Betrachtungsgegenstand beim Entwurf von Integrationsarchitekturen. Es ist jedoch die Gestaltung von M-C Kommunikationskanälen von der Verwendung geeigneter AwS zur Durchführung von Aufgaben zu trennen. Auf letzteres kann beim Entwurf von Integrationsarchitekturen Einfluss genommen werden, sollte sich bspw. herausstellen, dass es geeigneter ist, einem personellen AT Zugriff auf ein AwS zur Durchführung der Aufgabe zu geben, anstatt C-C oder M-M Kommunikationskanäle zu implementieren.

Beim Entwurf der Integrationsarchitektur können lediglich Strukturen im Rahmen von Kommunikationskanälen vorgegeben werden. Grenzen bestehen bei der Schaffung idealer Rahmenbedingungen, die zur günstigen Entwicklung von Kommunikationskanälen beitragen, insbesondere bei M-M Kommunikationskanälen. Während auf die Funktionstüchtigkeit von C-C Kommunikationskanälen Einfluss genommen werden kann, ist dies bei M-M Kommunikationskanälen nur bedingt möglich. Deshalb sind bei der Etablierung dieser Kommunikationskanäle psychologische Aspekte, die die zwischenmenschliche Beziehungen der involvierten Personen betreffen, zu berücksichtigen. Bei der Gestaltung der Integrationsarchitektur kann nur soweit Einfluss genommen werden, dass zur Aufrechterhaltung von nicht oder nicht mehr funktionierenden M-M Kommunikationskanälen neue Kommunikationspartner definiert oder diese in C-C Kommunikationskanäle überführt werden. Kommunikationskanäle, die durch Manipulation, Zurückhalten oder die verzögerte Eingabe von Daten wegen fehlender Bereitschaft zur Kooperation entgegen der Arbeitsanweisungen blockiert werden, können unter Verweis auf das bestehende Arbeitsverhältnis und der damit eingegangenen Pflicht, Arbeitsanweisungen zu befolgen, entstört werden. Weiterführende Gestaltungsmöglichkeiten zur Wiederherstellung dysfunktionaler Kommunikationskanäle sind jedoch nicht Gegenstand beim Entwurf von Integrationsarchitekturen.

Zusammengefasst besteht die Hauptaufgabe beim Entwurf der Integrationsarchitektur darin, Vorgaben in Bezug auf die Koordination der AT zur Integration von Aufgaben zu treffen, damit sie, von außen betrachtet, eine Einheit bilden. Die Kooperation der AT ist dabei das Gestaltungsobjekt.

Zum Management der Integrationsarchitektur zählt, diese von Zeit zu Zeit auf Handlungsbedarfe zu untersuchen bzw. an Änderungen anzupassen; bspw. wenn ein GP geändert, die Aufbauorganisation reorganisiert oder die AwS-Unterstützung reformiert werden soll.

3.2 Motivation für Integrationsprojekte

In Abschnitt 3.1 wurde der Begriff Integration, so wie er im Rahmen dieser Arbeit verwendet wird, elaboriert. Darauf aufbauend wird die Durchführung von Integrationsprojekten motiviert. Dazu werden zunächst mögliche Situationen, die zu Integrationsprojekten führen können, in Abschnitt 3.2.1 aufgezeigt. Anschließend werden Ziele, die mit der Durchführung von Integrationsprojekten verfolgt werden, in Abschnitt 3.2.2 vorgestellt.

3.2.1 Auslöser von Integrationsprojekten

Nachfolgend werden Situationen beschrieben, die den Bedarf nach Integrationsprojekten auslösen können. Während Gassner (1996, S. 161) dabei vorrangig die informationstechnische Seite betrachtet, berücksichtigen Österle (1996, S. 3-5) und Winter (2009, S. 53-69) auch unternehmerische und organisatorische Aspekte. Gassner (1996, S. 161) identifiziert nachfolgende Situationen als Auslöser von Integrationsprojekten:

- **Einführung neuer AwS** Kauft oder entwickelt ein Unternehmen neue AwS, dann sollten diese in das bestehende Informationssystem integriert werden.
- **Weiterentwicklung bestehender AwS** AwS können im Laufe ihres Lebenszyklus mehrfach zur Unterstützung neuer Aufgaben weiterentwickelt werden. Jede Änderung bei vorhandenen AwS kann Auswirkungen auf die vernetzten AwS nach sich ziehen und gegebenenfalls die Integrationsbeziehungen verändern.
- **Restrukturierung** Eine Restrukturierung der AwS-Landschaft bei gleichbleibender Funktionalität der AwS.

Zudem identifiziert Winter die *Einführung von Standardsoftware* (Winter 2009, S. 62-65) und die *System-Migration* (Winter 2009, S. 59-61) als Situationen, die zu Integrationsprojekten führen können. Diese können als Ausprägungen zu den bereits aufgeführten Situationen, die von informationstechnischer Seite Integrationsprojekte auslösen können, gerechnet werden. So geht mit der Standardsoftwareeinführung oftmals die Vernetzung mit vorhandenen AwS, insbesondere den im Unternehmen bereits existierenden, einher (Winter 2009, S. 59). Unter System-Migration wird der Umstieg von einem AwS auf ein anderes verstanden (Winter 2009, S. 59). Dies kann durch technische Innovation, grundlegende Neustrukturierung der IT-Landschaft oder Zusammenlegung von Unternehmen ausgelöst werden (Winter 2009, S. 59). Zu den technischen Innovationen kann der Umstieg auf eine Client-Server-Architektur gezählt werden (Winter 2009, S. 59).

Zu den unternehmerischen Gründen, die Integrationsprojekte auslösen können, gehören die *Aufteilung von Unternehmen in operative Einheiten eines Konzerns* sowie die *Auslagerung von Einheiten (Outsourcing)* (Österle 1996, S. 4; Winter 2009, S. 65-67). Ziele, die mit einer Zerlegung und Verteilung von Unternehmen verfolgt werden, sind Konzentration auf Kernkompetenzen sowie Marktmechanismen und Ergebnisverantwortung anstelle von hierarchischer Koordination (Österle 1996, S. 4). In der Folge entstehen vielfältige Kooperationsgeflechte zwischen und innerhalb von Unternehmen (A. Picot et al. 2003, S. 2), mit denen wiederum die Realisierung von Synergieeffekten beabsichtigt wird (Österle 1996, S. 4) und die in der Folge eine *unternehmensübergreifende Integration* erfordern (Winter 2009, S. 57-59). Aufgrund derartiger Änderungen müssen auch involvierte AwS miteinander interagieren, die nicht gemeinsam entworfen oder entwickelt wurden (Österle 1996, S. 5).

Zusammenschlüsse von Unternehmen (Mergers) sowie Unternehmensübernahmen (Acquisitions) können ebenfalls Integrationsprojekte auslösen (Winter 2009, S. 54-57). Dabei ist zwischen *Mergers & Acquisitions*³⁰ und unternehmensübergreifender Integration, bei der die beteiligten Unternehmen erhalten bleiben bzw. bei der die Selbstständigkeit dieser nicht berührt wird, zu differenzieren (Winter 2009, S. 57-59).

Darüber hinaus führt Winter die *unternehmensinterne Reorganisation* als einen weiteren möglichen Grund an, der Integrationsprojekte zur Folge haben kann (Winter 2009, S. 68-69). Ursachen für die unternehmensinterne Reorganisation können fachliche Innovationen, technische Innovationen oder Maßnahmen zur Vereinfachung sein (Winter 2009, S. 68). Zu den fachlichen Innovationen zählen bspw. die Einführung eines neuen Produkts oder die Erschließung eines neuen Marktsegments (Winter 2009, S. 68). Eine technische motivierte Innovation kann bspw. die Einführung von RFID-Etiketten sein (Winter 2009, S. 68). Bei den Maßnahmen zur Vereinfachung sollen Strukturen, die sich historisch entwickelt haben, kritisch analysiert und gegebenenfalls neu gestaltet werden (Winter 2009, S. 69).

3.2.2 Ziele bei Integrationsprojekten

D. Fischer hat nachfolgende, häufig in der Literatur genannten Ziele, die mit der Integration in Informationssystemen verfolgt werden, identifiziert (D. Fischer 2008, S. 161): Zeitersparnis durch Verkürzung von Durchlauf- und Transaktionszeiten, Kostenersparnis durch Verringerung von Bearbeitungs- und Transaktionskosten, Entlastung des Personals von monotonen Routinetätigkeiten, Vermeidung von Datenredundan-

³⁰Für das inzwischen fest in der deutschsprachigen Betriebswirtschaftslehre verankerte Begriffspaar *Mergers & Acquisitions (M&A)* gibt es bisher keine allgemein akzeptierte Übersetzung, es kann aber sinngemäß mit „Unternehmenszusammenschlüsse und Unternehmensübernahmen“ übersetzt werden (G. Picot 2008, S. 26).

zen, Verbesserung der Integrität/Kontrolle von Daten/Vorgängen, Vermeidung von Fehlern bei der Datenerfassung, Zugriff auf eine einheitliche gemeinsame Datenbasis bzw. synchron gehaltene Datenbestände, einmalige Datenerfassung bzw. Verringerung von Doppelerfassungen, Erschließung neuer Geschäftsfelder und Absatzwege, Verbesserung der strategischen Wettbewerbsposition.

Die von D. Fischer genannten Ziele können gruppiert werden in unternehmerische Ziele, Senkung des Ressourcenbedarfs, Reduktion der Redundanzen sowie Erhöhung der Konsistenz (D. Fischer 2008, S. 162-173). Die Einordnung der Ziele in die Kategorien kann aus nachfolgender Auflistung entnommen werden. Das Ziel *einmalige Datenerfassung bzw. Verringerung von Doppelerfassungen* stellt dabei eine Ausnahme dar, da es in die Kategorien Ressourcenbedarf, Redundanz sowie Konsistenz eingeordnet wird.

- **Unternehmerische Ziele** Erschließung neuer Geschäftsfelder und Absatzwege, Verbesserung der strategischen Wettbewerbsposition
- **Senkung des Ressourcenbedarfs** Zeitersparnis durch Verkürzung von Durchlauf- und Transaktionszeiten, Kostenersparnis durch Verringerung von Bearbeitungs- und Transaktionskosten, Entlastung des Personals von monotonen Routinetätigkeiten, einmalige Datenerfassung bzw. Verringerung von Doppelerfassungen
- **Reduktion der Redundanzen** Vermeidung von Datenredundanzen, einmalige Datenerfassung bzw. Verringerung von Doppelerfassungen
- **Erhöhung der Konsistenz** Verbesserung der Integrität/Kontrolle von Daten/Vorgängen, Vermeidung von Fehlern bei der Datenerfassung, Zugriff auf eine einheitliche gemeinsame Datenbasis bzw. synchron gehaltene Datenbestände, einmalige Datenerfassung bzw. Verringerung von Doppelerfassungen

Die Kategorien *unternehmerische Ziele* und *Senkung des Ressourcenbedarfs* können zu den Kategorien mit vorrangig wirtschaftlichen Interessen gezählt werden. Die Erste kann unter anderem durch unternehmensübergreifende Kooperationen, Letztere bspw. durch die Automatisierung von Transaktionen beeinflusst werden. Die Kategorien *Reduktion der Redundanzen* und *Erhöhung der Konsistenz* unterstützen wiederum die beiden vorgenannten Kategorien. Da die Kategorien Redundanz und Konsistenz auch Qualitätsmerkmale einer Integrationsarchitektur darstellen, werden sie in Abschnitt 3.3.1 näher beleuchtet.

Neben den bereits genannten Zielen spielt die *Investitionssicherheit* bei der Implementierung von Gestaltungsvorschlägen eine wichtige Rolle (Noffsinger et al. 1998, S. 81). Sie bezieht sich darauf, eine implementierte Integrationsarchitektur mindestens so lange zu nutzen, bis sich die Kosten der Einführung amortisiert haben. Da zur

Planungszeit oftmals nicht alle Umweltbedingungen berücksichtigt werden können oder eine über den definierten Lebenszyklus hinausgehende Verwendung erwünscht ist, besteht die Anforderung darin, die Integrationsarchitektur anpassbar zu gestalten. *Wiederverwendbarkeit* und *Erweiterbarkeit* sind daher Aspekte, die beim Entwurf von Integrationsarchitekturen zu berücksichtigen sind.

Beim Entwurf von Integrationsarchitekturen ist zu prüfen, inwiefern vorhandene AwS, sofern erwünscht, durch Anpassung wiederverwendet werden können oder ob sie durch neu einzuführende abgelöst sind. Bei der Ablösung vorhandener AwS ist darauf zu achten, dass die von den AwS bislang unterstützen Transaktionen nach Implementierung der Maßnahmen weiterhin zu unterstützen sind. Offene AwS-Architekturen begünstigen die Wiederverwendbarkeit. *Offenheit* wird erreicht, indem einerseits Standards verwendet werden und andererseits die Architektur sowie die Entscheidungsprozesse dokumentiert werden. Das von Thränert (2008, S. 108) vorgeschlagene fünfstufige Reifegradmodell kann dabei als Gradmesser herangezogen werden, um den Reifegrad von Integrationsarchitekturen in Bezug auf die Wiederverwendung zu bewerten. Mit dem Reifegradmodell können bspw. Benchmarks durchgeführt und ein höherer Reifegrad als neues Ziel festgelegt werden. Es ist in folgende fünf Stufen unterteilt: Einmalige Integrationslösung (Ad-hoc-Wiederverwendung), Wiederverwendung verfügbarer Artefakte, Entwicklung für Wiederverwendung (generische Lösung in Form eines Frameworks), Verwendung von Domänenmodellen und der statistischen Steuerung der von den Domänen betroffenen GP sowie eine organisationsweite Ausrichtung auf Wiederverwendung.

Bei der Erweiterbarkeit liegt der Fokus darauf, ob und mit welchem Aufwand eingesetzte AwS um die Unterstützung neuer Aufgaben oder Transaktionen erweitert werden können. Voraussetzungen dafür sind, dass die betroffenen AwS modifizierbar und wartbar sind. Auch hier unterstützen die zuvor angesprochenen offenen AwS-Architekturen die Erweiterbarkeit. Die Gestaltung der Aufgabenintegration hängt dabei maßgeblich davon ab, ob bestehende Kommunikationskanäle oder AwS um die Unterstützung neuer Transaktionen durch Etablierung von Kommunikationskanälen erweitert werden können.

3.3 Integrationsmerkmale und Lösungskonzepte bei der Gestaltung von Integrationsarchitekturen

Sachziel bei der Gestaltung von Informationssystemen ist die Automatisierung der Aufgaben durch AwS.³¹ Aus aufgabenübergreifender Perspektive ist die Integration von Aufgaben durch AwS Gegenstand der übergeordneten Gestaltungsaufgabe, die als Integrationsaufgabe bezeichnet wird (Ferstl 1992, S. 3; Schissler et al. 2001, S. 4). Diese Aufgabe zählt zu den Formalzielen bei der Konzeption von Informationssystemen. Zu den einzelnen Formalzielen zählen die nachfolgend eingeführten Integrationsmerkmale. Sie nehmen auf die Gestaltung der Beziehungen zwischen den Aufgaben Bezug (Ferstl und Sinz 2013, S. 243-244). Dabei ist zu trennen zwischen den Beziehungen von Teilaufgaben eines AwS, die Gegenstand der Intra-AwS-Integration des jeweiligen AwS sind, und den Beziehungen von Teilaufgaben mehrerer AwS, welche Gegenstand der Inter-AwS-Integration zwischen den beteiligten AwS sind (Schissler et al. 2001, S. 4). Obwohl in der vorliegenden Arbeit der Fokus auf der Inter-AwS-Integration liegt, gelten die Integrationsmerkmale für beide Arten von Beziehungen.

3.3.1 Integrationsmerkmale bei der Gestaltung von Integrationsarchitekturen

Bevor Aufgaben zerlegt und AT zugeordnet werden, sind Ziele und Restriktionen, die dabei einzuhalten sind, zu definieren. Die Integrationsmerkmale nehmen auf diese Anforderungen Bezug und werden in ihrer Zielausprägung als Integrationsziele bezeichnet. Integrationsmerkmale und Integrationsziele bilden zusammen den Integrationsgrad eines Aufgabenkomplexes.³² Um den geeigneten Integrationsgrad bestimmen zu können ist für jedes Integrationsmerkmal eine anzustrebende Zielausprägung festzulegen (Ferstl und Sinz 2013, S. 240).

Die Integrationsmerkmale, wie in Tab. 3.1 dargestellt, orientieren sich an den Eigenschaften der Aufgaben und AT. Mit den strukturorientierten Integrationsmerkmalen *Redundanz* und *Verknüpfung* wird berücksichtigt, dass AwS aus Systemkomponenten unterschiedlicher Art, wie Datenstrukturen, Funktionen und Objekten sowie aus Kommunikationskanälen zwischen Komponenten, bestehen können. Mit den verhaltensorientierten Integrationsmerkmalen *Konsistenz* und *Zielorientierung* wird die ganzheitliche Ausrichtung der Aufgaben abgestimmt (Ferstl und Sinz 2013, S. 240-241).

³¹Die Automatisierung von Aufgaben und ihre Ausprägungen werden in Abschnitt 2.2.2 beschrieben.

³²Der Begriff Aufgabenkomplex wird in Abschnitt 3.1.1 erläutert und die Zielerreichungsgrade in Abschnitt 2.2.2.

Tabelle 3.1: Integrationsmerkmale (Ferstl und Sinz 2013, S. 241)

Merkmalsgruppe		Ziel: Einhaltung einer vorgegebenen Ausprägung des Merkmals ...
Struktur	Redundanz	Datenredundanz
		Funktionsredundanz
	Verknüpfung	Kommunikationsstruktur
Verhalten	Konsistenz	semantische Integrität
		operationale Integrität
	Zielorientierung	Vorgangsteuerung

Nachfolgend werden die in der Tab. 3.1 aufgeführten Ausprägungen der Integrationsmerkmale erläutert und auf deren Bedeutung für die anwendungssystemübergreifende Integration (Inter-AwS-Integration) eingegangen.

3.3.1.1 Redundanz

Das Merkmal Redundanz gibt Auskunft darüber, inwieweit Elemente mehrfach vorhanden sind. Redundante Elemente können ohne Beeinträchtigung entfernt werden. Ihr Wegfall würde mögliche Inkonsistenzen zwischen redundanten Elementen, die Notwendigkeit der Prüfung und Korrektur von Inkonsistenzen und eine mangelnde Wirtschaftlichkeit der Ressourcennutzung vermindern. Da diese im Störfall jedoch als Rückfallstufe dienen können, durch parallele Nutzung redundanter Elemente mögliche Leistungssteigerungen zu verzeichnen sind und die Strukturkomplexität möglicherweise reduziert werden könnte, ist abzuwägen, welche Elemente redundant vorgehalten werden sollen (Ferstl 1992, S. 13; Mantel et al. 2001, S. 4; Ferstl und Sinz 2013, S. 241-242).

Redundanz kann differenziert werden in *Daten-* und *Funktionsredundanz*. Die Datenredundanz ist in der Regel die Folge überlappender AO und zeigt sich in der Form redundanter Datenobjekttypen bzw. Datenattribute sowie redundanter Datenobjekte. Die Funktionsredundanz ist häufig das Ergebnis aus nicht disjunkten Zerlegungen von Aufgabenzielen und tritt bei Überlappung von Lösungsverfahren in Form redundanter Aktionen auf (Ferstl 1992, S. 13; Ferstl und Sinz 2013, S. 241-242).

In vorhandenen AwS-Landschaften lassen sich Daten- und Funktionsredundanz nur schwer vermeiden, da bei Verwendung bereits am Markt vorhandener Standardsoftware oftmals nur begrenzte Einflussmöglichkeiten bei deren Weiterentwicklung existieren.

Datenredundanz entsteht in diesen Fällen oftmals dadurch, dass AwS Kopien benötigter Daten anfertigen, um beim Betrieb unabhängig von anderen AwS zu sein. Funktionsredundanz kann bei Einsatz von Standardsoftware ebenfalls nicht vermieden werden, da AwS-Hersteller eingebaute Funktionen in der Regel nicht ohne gewichtige Gründe entfernen. Das wird bereits dadurch erschwert, dass Funktionen, die in einem anwendenden Unternehmen nicht verwendet werden, von anderen Organisationen genutzt werden könnten und deshalb die Entfernung vorhandener Funktionen nicht mehr möglich ist. Folglich besteht das Integrationsziel darin, ungeplante Redundanz zu vermeiden (Ferstl 1992, S. 13).

3.3.1.2 Verknüpfung

Mit dem Merkmal Verknüpfung werden Art und Anzahl der Kommunikationskanäle zwischen den Systemkomponenten definiert. In objektorientierten AwS sind dies die Kommunikationskanäle zwischen Objekten, um Nachrichten auszutauschen oder auf gemeinsamen Datenobjekten operieren zu können. Der Bedarf an Kommunikationskanälen ergibt sich aus der Abgrenzung der Systemkomponenten sowie deren Grad an Redundanz. Bei anwendungssystemübergreifender Integration ist das Ziel dieses Merkmals der Entwurf einer *Kommunikationsstruktur*, die Aussagen zur Anzahl, Leistungsfähigkeit, Stabilität sowie Flexibilität der Verknüpfungen trifft. Darüber hinaus sind aus Entwickler- und Betreibersicht auch Angaben über die Transparenz und Kontrollierbarkeit der Kommunikationsstruktur zu treffen (Ferstl 1992, S. 13; Mantel et al. 2001, S. 5; Ferstl und Sinz 2013, S. 242).

3.3.1.3 Konsistenz

Mit einem AwS wird ein Ausschnitt der realen Welt abgebildet. Zugehörige Werte des abgebildeten Ausschnitts werden als konsistent, nicht zugehörige Werte als inkonsistent bezeichnet. Das Merkmal Konsistenz kann differenziert werden in *semantische Integrität* und *operationale Integrität*. Semantische Integritätsbedingungen beschreiben aus Modellierungssicht, welche Werte erlaubt sind. Die operationalen Integritätsbedingungen definieren in einem System, das parallele Transaktionen erlaubt, legitime Systemzustände vor und nach der Durchführung von Transaktionen (Ferstl 1992, S. 13; Mantel et al. 2001, S. 5-6; Ferstl und Sinz 2013, S. 242-243). Bei der anwendungssystemübergreifenden Integration sind die Ziele dieses Merkmals eine konsistente Datenhaltung, die operationale Integrität nebenläufiger Zustandsübergänge und globaler Transaktionsschutz. Voraussetzungen sind die vollständige und korrekte Spezifikation der zu integrierenden AwS (Ferstl et al. 1997, S. 26).

3.3.1.4 Zielorientierung

Das Merkmal Zielorientierung beschreibt, inwiefern das aus der Aufgabengliederung resultierende Aufgabennetz und die unterstützenden AwS zur Zielerreichung der Gesamtaufgabe beitragen. Dazu gehören der Zielorientierung als Teilmerkmale der potenzielle sowie der tatsächliche Zielbeitrag an. Der potenzielle Zielbeitrag der Teilaufgaben wird durch die Zerlegung der Gesamtaufgabe festgelegt. Der tatsächliche Zielbeitrag hingegen ist von der Interaktion der zur Erreichung der Gesamtaufgabe involvierten AT abhängig. Dazu werden bei einer anwendungssystemübergreifenden Integration mit einer *Vorgangsteuerung* sowohl die automatisierten als auch die teilautomatisierten Vorgänge der Gesamtaufgabe koordiniert. Das daraus resultierende Integrationsziel ist die Lenkung der Aktionenfolge (Ferstl 1992, S. 13-14; Mantel et al. 2001, S. 6; Ferstl und Sinz 2013, S. 243).

3.3.1.5 Bezug der Integrationsziele auf Aufgaben- und Aufgabenträgerebene

In Abb. 3.5 werden die Integrationsziele den Integrationsgegenständen zugeordnet (Ferstl und Sinz 2013, S. 240-241). Aus Abb. 3.5 wird ersichtlich, dass Maßnahmen, mithilfe derer die Integrationsziele erreicht werden können, sowohl auf Aufgaben- als auch auf AT-Ebene und dort jeweils auf der Typ- oder Instanzebene ergriffen werden können. So kann das Integrationsziel *Zielorientierung* lediglich auf den Typebenen der Aufgaben- und AT-Ebene beeinflusst werden. Bei der *Verknüpfung* bieten sich die Typ- oder Instanzebenen der Aufgaben- und AT-Ebene an. Die *Konsistenz* kann bei Aufgaben- und AT-Ebene lediglich auf Instanzebene erreicht werden. Maßnahmen, mit denen die Ziele der *Redundanz* erreicht werden können, sind nur auf AT-Ebene, dafür aber sowohl auf Typ- als auch auf Instanzebene, beeinflussbar.

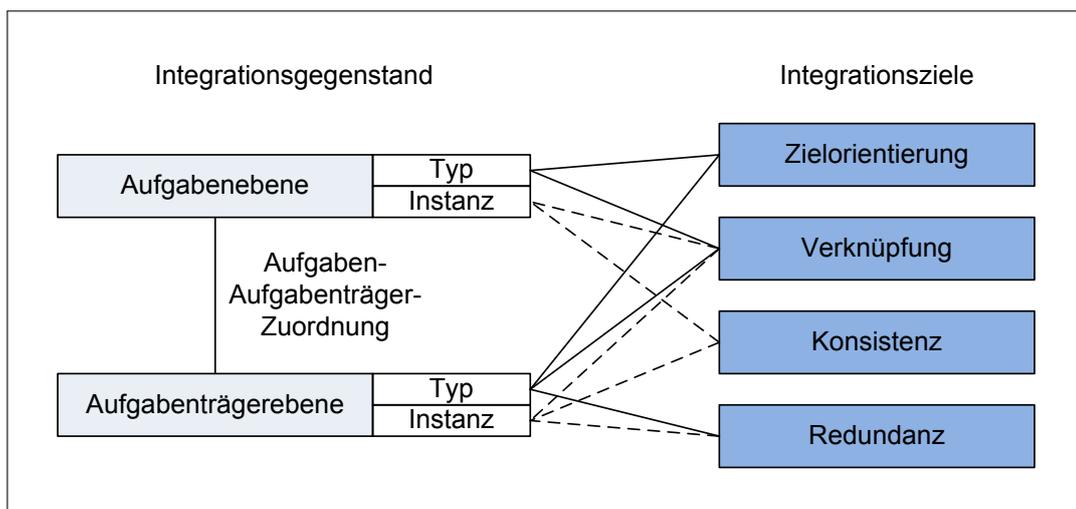


Abbildung 3.5: Typ- bzw. Instanzbezug der Integrationsziele (Ferstl und Sinz 2013, S. 240)

3.3.2 Integrationskonzepte zur Gestaltung von Integrationsarchitekturen

Integrationsmerkmale (Abschnitt 3.3.1) legen die bei der Gestaltung der Integrationsarchitektur zu erreichenden Eigenschaften fest. Integrationskonzepte sind Konzepte zur Gestaltung der Integrationsarchitektur. Mit diesen können die Ausprägungen der Integrationsmerkmale in unterschiedlichem Maß erreicht werden. Grundsätzlich wird unterschieden zwischen Funktion-, Daten- und Objektintegration, die bezüglich der Formalziele der Integration sehr unterschiedliche Zielerreichungsgrade aufweisen. Ziel der *Funktionsintegration* ist die Vernetzung von Aufgaben bzw. Lösungsverfahren durch Kommunikationskanäle. Sie kann gegliedert werden in *aufgabenträgerorientierte* und *datenflussorientierte* Funktionsintegration. Die *Datenintegration* orientiert sich bei der Vernetzung an überlappenden AO. Die *Objektintegration* bezieht entsprechend dem Objektbegriff Kommunikationskanäle zwischen Objekten und Überlappung innerhalb von Objekten ein (Ferstl und Sinz 2013, S. 245).

3.3.2.1 Aufgabenträgerorientierte Funktionsintegration

Sachziel der aufgabenträgerorientierten Funktionsintegration ist die Vernetzung teilautomatisierter Aufgaben. Die Aufgaben werden bei diesem Konzept durch einen gemeinsamen personellen AT integriert. Mit der aufgabenträgerorientierten Funktionsintegration werden die Integrationsziele *Kommunikationsstruktur* und *Vorgangsteuerung* adressiert. Bei der Kommunikationsstruktur berücksichtigt sie personell gestützte Kommunikationskanäle zwischen den Funktionen eines AwS. Bei der Aufgabengliederung ist in Bezug auf das Merkmal Vorgangsteuerung darauf zu achten, dass der personelle Anteil der zu integrierenden Aufgaben von einer Person durchgeführt werden kann (Ferstl 1992, S. 14-15; Schissler et al. 2001, S. 5; Ferstl und Sinz 2013, S. 244-245). Daher eignet sich dieses Konzept zwar prinzipiell für die anwendungssystemübergreifende Integration, bringt jedoch erhebliche potenzielle Fehlerquellen im produktiven Betrieb mit sich, dadurch dass wichtige Integrationsziele, wie *Redundanz* und *Konsistenz*, nicht berücksichtigt werden. Darüber hinaus ist die Vorgangsteuerung auf die von einer Person integrierten Aufgaben ausgerichtet; eine übergreifende Koordination aller Aufgaben zur Erreichung der Gesamtaufgabe wird vernachlässigt.

Bei der aufgabenträgerorientierten Funktionsintegration können AwS isoliert bestehen bleiben. Die Vernetzung von Aufgaben wird durch personelle AT sichergestellt. Eine typische Ausprägung ist die nicht-automatisierte Datenübertragung von einem AwS in ein anderes durch den personellen AT.

3.3.2.2 Datenflussorientierte Funktionsintegration

Bei der datenflussorientierten Funktionsintegration tauschen AwS Daten aus, im Allgemeinen ohne Bezug auf die vom AwS durchzuführende Transaktion (Mantel et al. 2001, S. 7). Dazu werden die Ein-/Ausgabekanäle der AwS gemäß ihrer korrespondierenden Aufgaben verknüpft. Das Konzept stammt aus den 60er Jahren des letzten Jahrhunderts und „entstand aus dem Bemühen, Automationsinseln zu verbinden und die Kommunikationskanäle zu automatisieren“ (Ferstl und Sinz 2013, S. 245). Daher greift dieses Konzept bei der anwendungssystemübergreifenden Integration primär das Integrationsziel *Kommunikationsstruktur* auf, indem ein Kommunikationssystem, über das Nachrichten zwischen den beteiligten AwS ausgetauscht und Informationen bezüglich der Kommunikationsstruktur bereitgestellt werden, aufgebaut wird. Das Integrationsmerkmal *Vorgangssteuerung* wird teilweise durch das Überwachen von Vorgangsketten mithilfe von Monitoren unterstützt (Ferstl 1992, S. 15-16; Mantel et al. 2001, S. 7; Schissler et al. 2001, S. 6; Ferstl und Sinz 2013, S. 245-247). Als stellvertretendes Beispiel sei hier Electronic Data Interchange (EDI) erwähnt, welches einen automatisierten Austausch von Geschäftsdokumenten anstrebt (Ballnus 2000, S. 14). Auch dieses Konzept ist prinzipiell für die anwendungssystemübergreifende Integration geeignet, bringt jedoch durch die unberücksichtigten Integrationsziele *Redundanz* und *Integrität* ähnliche Fehlerquellen wie die aufgabenträgerorientierte Funktionsintegration mit sich. Im Gegensatz zur aufgabenträgerorientierten Funktionsintegration können die potenziellen Fehlerquellen im produktiven Betrieb durch eine entsprechende Gestaltung der AwS-Landschaft teilweise vermieden werden. So kann die Vorgangssteuerung mit einem globalen Transaktionsmechanismus kombiniert werden, welcher bei einem Problem den Rollback der übergeordneten Transaktion auslöst und damit die operationale Integrität gewährleistet. Die Datenredundanz kann mittels Verteilungsmechanismen begrenzt werden.

Auch bei der datenflussorientierten Funktionsintegration können AwS isoliert bestehen bleiben. Die Vernetzung von Aufgaben wird durch maschinelle AT unterstützt, indem die Datenübertragung mittels AwS bewerkstelligt wird.

3.3.2.3 Datenintegration

Bei der Datenintegration operieren die Lösungsverfahren der Aufgaben über externe Sichten (Views) auf ihren jeweiligen AO. Dabei werden die Datenstrukturen der gekoppelten AO in einem konzeptuellen Datenschema eines Datenbanksystems zusammengefasst. Damit wird das Ziel verfolgt, ein sogenanntes unternehmensweites Datenschema aufzubauen (Sinz 1995, S. 1). Mit dem Konzept der Datenintegration werden vordergründig die Integrationsziele *Datenredundanz* und *Konsistenz* erreicht. Kontrolliert wird die Datenredundanz anhand von Methoden der Datenmodellierung.

Definierte semantische Integritätsbedingungen stellen die semantische Integrität sicher und durch die Transaktionsverwaltung des Datenbankverwaltungssystems wird die operative Integrität gewährleistet. Die Kommunikation erfolgt über den überlappenden Bereich von externen Sichten, mit dem damit verbundenen Nachteil, dass das Integrationsziel *Kommunikationsstruktur* aufgrund der engen Kopplung nicht erreichbar ist. Das Integrationsziel *Vorgangsteuerung* wird ebenfalls vernachlässigt (Ferstl 1992, S. 17-21; Mantel et al. 2001, S. 7-8; Schissler et al. 2001, S. 6; Ferstl und Sinz 2013, S. 247-249). Folglich eignet sich die Datenintegration nur bedingt für die anwendungssystemübergreifende Integration.

3.3.2.4 Objektintegration

Bei der Objektintegration können konzeptuelle Objekte und Vorgangsobjekte lose gekoppelt zusammengesetzt werden.³³ Dazu führen Vorgangsobjekte die Aufgaben durch und nutzen die Dienste der konzeptuellen Objekte. Der Nachrichtenaustausch erfolgt in diesem Fall über Kommunikationskanäle. Das Konzept der Objektintegration berücksichtigt dadurch alle in Ferstl und Sinz (2013, S. 240-243) sowie in Ferstl (1992, S. 11-14) eingeführten Integrationsziele (vgl. Abschnitt 3.3.1). So wird das Integrationsziel *Vorgangsteuerung* erreicht, indem jeder VOT mit einer Teilaufgabe des Aufgabennetzes und die Beziehungen zwischen den VOT mit der Zerlegungsstruktur der Gesamtaufgabe korrespondieren. Die *Datenredundanz* wird durch Methoden der Datenmodellierung vermieden. Das Integrationsziel *Funktionsredundanz* wird durch das Konzept der Generalisierung sowie die Nutzung von Vererbungsbeziehungen erreicht. Die *semantische Integrität* wird durch die Kommunikation mithilfe von Nachrichten und die *operative Integrität* durch die Transaktionsverwaltung mithilfe von Transaktionsmanagern sichergestellt. Zur Kommunikation aller Objekte wird ein eigenes Kommunikationssystem verwendet, welches auch eine Kontrolle der *Kommunikationsstruktur* erlaubt (Ferstl 1992, S. 21-24; Mantel et al. 2001, S. 8-9; Schissler et al. 2001, S. 6-7; Ferstl und Sinz 2013, S. 249-251). Aus den genannten Gründen ist von den vorgestellten Konzepten die Objektintegration für die anwendungssystemübergreifende Integration am besten geeignet. Allerdings sollte anstatt einer AWS-übergreifenden Vererbung, die zu starken Abhängigkeiten zwischen den integrierten AWS führt, der Delegation von Funktionen den Vorzug gegeben werden (Schissler et al. 2001, S. 7).

³³Konzeptuelle Objekte und Vorgangsobjekte sind Teil der SOM-Methodik und werden in Abschnitt 2.3 eingeführt.

3.3.2.5 Zielerreichungsgrade der Integrationskonzepte

Zusammenfassend werden in Tab. 3.2 die Zielerreichungsgrade der maschinell unterstützten Integrationskonzepte dargestellt. Die Maßnahmen, mit denen in einem Integrationskonzept ein Integrationsmerkmal erreicht werden kann, können ebenfalls aus der Tab. 3.2 entnommen werden. Dabei ist zu unterscheiden, ob Methoden zur Zielverfolgung verfügbar sind (●), Teilansätze für eine methodische Zielverfolgung vorliegen (◐) oder das Ziel nicht verfolgt wird bzw. keine methodischen Ansätze für eine Zielverfolgung zur Verfügung stehen (○).

Tabelle 3.2: Zielerreichungsgrade der Integrationskonzepte (Ferstl und Sinz 2013, S. 243-251; Mantel et al. 2001, S. 6-9)

Integrationsmerkmal	Integrationskonzept mit Maßnahmen und Zielerreichungsgraden		
	datenflussorientierte Funktionsintegration	Datenintegration	Objektintegration
Datenredundanz	○	● Datenmodellierung	● Daten-/Objektmodellierung
Funktionsredundanz	○	○	● Objektmodellierung
Kommunikationsstruktur	● Kommunikationsnetz, Ein-/Ausgänge jeder Funktion bekannt	○	● Kommunikationsnetz
Integrität	○	● semantische Integritätsbedingungen, Transaktionsverwaltung	● semantische Integritätsbedingungen, Transaktionsverwaltung
Vorgangsteuerung	◐ Monitor zur Überwachung von Vorgangsketten	○	● Vorgangsubjekte

3.4 Konzeption von Integrationsarchitekturen als Systementwicklungsaufgabe

Der Begriff der Systementwicklung wird im Rahmen dieser Arbeit umfassend verstanden. Er beschränkt sich nicht nur auf die ingenieurmäßige Entwicklung eines AwS, sondern bezieht alle Teilnehmer, auch personelle AT ein. Darüber hinaus liegt der Fokus bei der Gestaltung nicht nur auf der Entwicklung eines AwS, sondern auf dem Zusammenwirken aller im Informationssystem beteiligten Akteure.

3.4.1 Aufgabenmodell der Systementwicklung

Die Systementwicklungsaufgabe für Integrationsarchitekturen lässt sich anhand des in der Abb. 3.6 dargestellten Aufgabenmodells beschreiben.

Aufgaben bestehen u. a. aus Sach- und Formalzielen, wie in Abschnitt 2.2 beschrieben. Das Sachziel ist die Entwicklung einer Integrationsarchitektur, die die vorgegebenen Anforderungen inhaltlicher Art – welche Informationen sollen voneinander abhängige Aufgaben untereinander austauschen – und qualitativer Art – wie sind die Aufgaben über die AT zu integrieren – erfüllt. Die Formalziele lassen sich gliedern in Zeit-, Kosten- und Qualitätsziele. Sie beziehen sich auf die Durchführung bei der Entwicklung einer Integrationsarchitektur. Zeitziele schränken den zeitlichen Horizont für die Entwicklung einer Integrationsarchitektur ein. Kostenziele richten sich an die Einhaltung vorgegebener Budgets und Qualitätsziele auf die Qualität des Systementwicklungsprozesses. AO ist die zu gestaltende Integrationsarchitektur. Betroffen sind dabei neben der Gestaltung der Integration von Aufgaben über AT auch alle weiteren Modelle und Spezifikationen. Vorereignis ist die Initiierung des Systementwicklungsprojekts ausgelöst aufgrund einer der in Abschnitt 3.2.1 genannten Situationen. Das Nachereignis ist das Vorliegen einer betriebsbereiten, eingeführten und abgenommenen Integrationsarchitektur. Die phasenorientierte Durchführung von Entwicklungsaktivitäten bildet das Lösungsverfahren von Systementwicklungsaufgaben.

3.4.2 Beschreibungsebenen bei der Entwicklung von Integrationsarchitekturen

Grundsätzlich lassen sich die Beschreibungsebenen bei der Systementwicklung in die vier in Abb. 3.7 dargestellten Ebenen unterteilen. Die Beschreibungsebenen können auch bei der Aufgabe *Gestaltung von Integrationsarchitekturen* angewandt werden.

Auf der Ebene *Modell des betrieblichen Objektsystems* werden die zu untersuchenden betrieblichen Objekte, ihre Beziehungen untereinander sowie der relevante Ausschnitt der Umwelt erfasst. Die Aufgaben eines betrieblichen Objekts werden anhand der Außensicht spezifiziert (Ferstl und Sinz 2013, S. 482). Bei der Entwicklung von Inte-

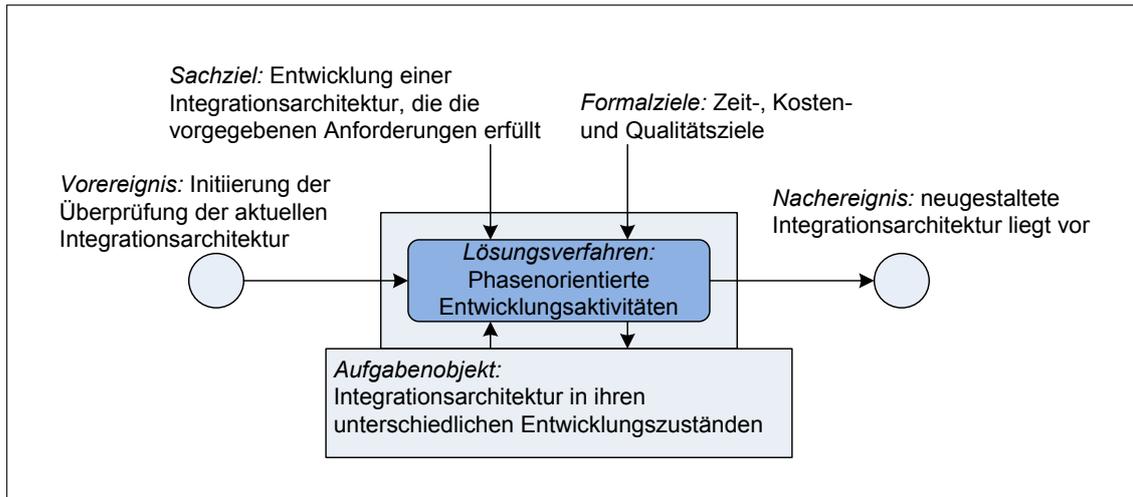


Abbildung 3.6: Aufgabenmodell bei der Gestaltung der Integration als Systementwicklungsaufgabe (in Anlehnung an Ferstl und Sinz (2013, S. 497))

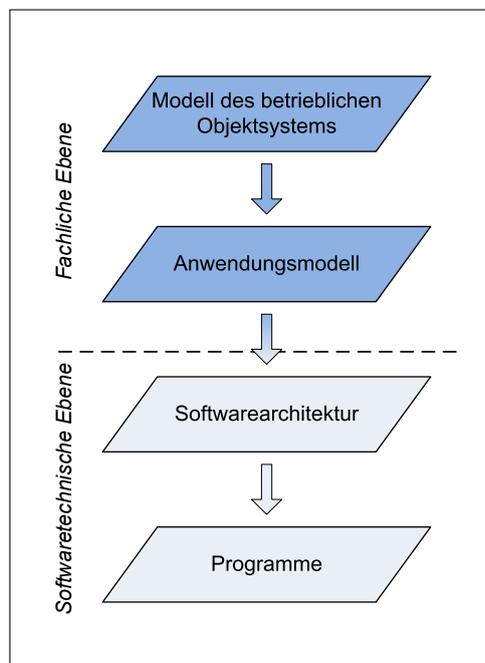


Abbildung 3.7: Beschreibungsebenen bei der Gestaltung von Integrationsarchitekturen (Ferstl und Sinz 2013, S. 482)

Integrationsarchitekturen wird der Ausschnitt des GP, für den eine Integrationsarchitektur erstellt oder bei einer bereits vorhandenen Architektur eine Anpassung durchgeführt werden soll, definiert. Die SOM-Methodik bietet hierfür das IAS und das Vorgangs-Ereignis-Schema (VES) an.

Festgelegte Automatisierungsgrade für Aufgaben und Transaktionen sind die Eingangsparameter für die Erstellung von *Anwendungsmodellen*. Auf dieser Ebene wird ein fachliches Modell für die zu erweiternden oder noch zu entwickelnden bzw. einzu-

führenden AwS, unter Verwendung von Standard-AwS, erstellt. Bei der Konzeption von Integrationsarchitekturen ist auf dieser Beschreibungsebene zu definieren, für welche Fälle welches AwS verantwortlich ist und entsprechend die Datenhoheit besitzt. Darüber hinaus sind Lösungsverfahren für Aufgaben zu definieren, die bislang noch von keinem AwS unterstützt werden. Für beides stehen im Rahmen der SOM-Methodik die Modellklassen KOS und VOS zur Verfügung.

Die Beschreibungsebenen *Modell des betrieblichen Objektsystems* sowie *Anwendungsmodell* betreffen die fachliche Ebene. Sie dienen dazu, Anforderungen für die zu untersuchende Diskurswelt zu definieren und Defizite des Ist-Zustandes im Vergleich zum Soll-Zustand aufzudecken. Die Ergebnisse dieser Analyse stellen die inhaltlichen Vorgaben für das Projekt dar.

Die Ebenen *Softwarearchitektur* und *Programm* sind Gegenstände der softwaretechnischen Beschreibungsebene. Bei der Softwarearchitektur werden Teilsysteme und Komponenten für ein AwS spezifiziert. Die Softwarearchitektur bildet die Grundlage für die darauf aufbauende Programmierung. Da die Konzeption von Integrationsarchitekturen von der softwaretechnischen Implementierung abstrahiert, obliegen beide Beschreibungsebenen den Herstellern der AwS und werden folglich nicht näher betrachtet. Einzig Einschränkungen hinsichtlich der zu verwendenden Hardwaresysteme sowie in Bezug auf die Systemsoftware, wie Betriebssysteme und Middleware, sind vom Auftraggeber zu treffen.

4 Ansätze zum Entwurf von Integrationsarchitekturen für Informationssysteme

4.1 Strategie und Durchführung der Suche nach Ansätzen	57
4.2 Deskription und Interpretation der Datengrundlage	61
4.3 Kriterien zur Evaluation von Ansätzen	62
4.4 Analyse und Bewertung gefundener Ansätze	64
4.4.1 Quasar Enterprise	64
4.4.2 Beitrag von Gronau	67
4.4.3 Der Ansatz von Jung	69
4.4.4 Der Ansatz von Kattenstroth	73
4.4.5 Der Ansatz von Schwinn	75
4.4.6 Der Ansatz von Vogler	79
4.5 Zusammenfassende Bewertung	83

Gegenstand dieses Kapitels ist die Untersuchung ausgewählter Ansätze zum Entwurf von Integrationsarchitekturen. Dazu wird nachfolgend die Untersuchungsstrategie beschrieben und parallel dazu ein Überblick über dieses Kapitel gegeben. Bei der Analyse von Ansätzen zum Entwurf von Integrationsarchitekturen für Informationssysteme soll mittels einer strukturierten Literaturrecherche eine möglichst spezifische Basis vorhandener Ansätze aufgebaut werden (Abschnitt 4.1). Die dabei erzielten Ergebnisse werden anschließend vorgestellt (Abschnitt 4.2). Ziel ist es, gefundene Ansätze mittels definierter Kriterien (Abschnitt 4.3) zu evaluieren (Abschnitt 4.4 und Abschnitt 4.5).

4.1 Strategie und Durchführung der Suche nach Ansätzen

In diesem Abschnitt wird die bei der Literaturrecherche angewandte Strategie beschrieben. Diese orientiert sich an den Richtlinien von Webster und Watson (2002, S. xvi). Ausgangspunkt ist die Suche nach Ansätzen, mithilfe derer Integrationsarchitekturen für Informationssysteme entworfen werden können.

Die Recherche selbst wurde unter Zuhilfenahme einschlägig bekannter Suchmaschinen durchgeführt. Um ein möglichst breites Spektrum abdecken zu können und die Suche interdisziplinär zu halten, wurden Datenbanken mit den Schwerpunkten Wirtschafts-

wissenschaften (WISO³⁴, Business Source Complete via EBSCO Host³⁵), Informatik (IEEE Computer Society Digital Library³⁶, SpringerLink³⁷) sowie Wirtschaftsinformatik bzw. deren „Halbschwester-Disziplin“ (Mertens 2013b) Information Systems (AIS Electronic Library³⁸) herangezogen. Diese Suchmaschinen decken einen Großteil der bekannten Fachkonferenzen und Fachzeitschriften ihrer jeweiligen Schwerpunktbereiche ab. Um darüber hinaus auch Monographien einzubeziehen, wurde die Suche auf die Suchmaschinen Bielefeld Academic Search Engine (BASE)³⁹ und Deutsche Nationalbibliothek (DNB)⁴⁰ ausgeweitet. Für die Recherche wurden folglich sieben Suchmaschinen verwendet.

Die Suche in den Datenbanken wurde ergänzt um eine Sichtung von Konferenzbeiträgen, die nicht durch die oben genannten Datenbanken abgedeckt werden. Dazu wurden Konferenzen berücksichtigt, in denen das Forschungsgebiet Integration schwerpunktmäßig behandelt wurde. Zu den in Betracht gezogenen Konferenzen zählen Data Warehousing (DW), Enterprise modelling and information systems architectures (EMISA), Modellierung, Modellierung betrieblicher Informationssysteme (MobIS), Multikonferenz Wirtschaftsinformatik (MKWI) sowie die Workshops zur EAI der Gesellschaft für Informatik (GI).

Die Suche wurde zuletzt Ende 2015 durchgeführt. Der Zeitraum der Suche berücksichtigt Publikationen, die vom 01. Januar 2004 bis zum 30. November 2015 veröffentlicht wurden. Es wurden sowohl Suchergebnisse in deutscher als auch in englischer Sprache einbezogen. Ferner wurde nur Literatur aufgenommen, die mit vertretbarem Aufwand beziehbar war (Onlinezugriff oder Bestellung über die lokale Bibliothek der Universität Bamberg einschließlich Fernleihe).

Die Recherche wurde sprachlich, wie bereits angedeutet, auf deutsche und englische Beiträge ausgerichtet. Entsprechend wurden auch die verwendeten Suchbegriffe zweisprachig gewählt. Der Suchstring selber wurde sukzessive verfeinert, bis die Anzahl der Treffer über alle Datenbanken hinweg kleiner 500 war. Diese Menge wurde als angemessenes Gleichgewicht zwischen Aufwand bei der Analyse der Ergebnisse einerseits und die damit zu erwartende Anzahl relevanter Ansätze andererseits erachtet.

³⁴<http://www.wiso-net.de/amedien.tin>

³⁵<http://search.ebscohost.com/community.aspx>

³⁶<https://www.computer.org/portal/web/search/advanced>

³⁷<http://link.springer.com/advanced-search>

³⁸<http://aisel.aisnet.org/>

³⁹<http://www.base-search.net/>

⁴⁰http://www.dnb.de/DE/Home/home_node.html

Der finale Suchbegriff lautet:

("Integration Engineering" OR "Enterprise Application Integration" OR EAI) AND (outline AND integration AND architecture AND (application OR information AND system*) AND (approach OR method)) OR ((Gestaltung OR Konzeption OR Entwicklung OR Entwurf OR Konstruktion) AND Integrationsarchitektur* AND (Informationssystem* OR Anwendungssystem*) AND (Methode OR Ansatz))

Dieser Suchstring kann in den verwiesenen Datenbanken ohne Anpassungen verwendet werden. Beim Aufbau des Suchstrings wurden einzelne Begriffe trunziert und mittels Boolescher Operatoren miteinander verknüpft. Darüber hinaus wurden Synonyme aufgenommen.

Die Tab. 4.1 zeigt die Vorgehensweise bei der Literaturrecherche sowie die Anzahl der Treffer bei den jeweils durchgeführten Phasen. Die initiale Recherche wurde mit dem Suchbegriff sowie der Sichtung der Konferenzen und Workshops durchgeführt. In der Spalte *Treffer* werden dazu die Anzahl gefundener Publikationen zu jedem Bereich angegeben. Über die weitere Verwendbarkeit wurde aufgrund der Begutachtung des Titels, der Inhaltsangabe und des Inhaltsverzeichnisses entschieden. Kriterium hierfür war die inhaltliche Auseinandersetzung mit dem Entwurf von Integrationsarchitekturen für Informationssysteme. Berücksichtigt wurden dabei die Beiträge, deren Untersuchungsziel entweder eine hohe Ähnlichkeit zum vorliegenden aufweist oder die einen Teilaspekt der im Vordergrund stehenden Problemstellung betreffen (vgl. Abschnitt 1.2). Die Anzahl übrig gebliebener Beiträge nach dieser Phase ist in der Spalte *Auswahl nach Bearbeitung* aufgeführt. Dieses stellt ein Zwischenergebnis dar. Um der Gefahr zu begegnen, einen isolierten Autoren- und Meinungskreis zu erhalten, wurde in Phase zwei eine vorwärts- und rückwärtsgerichtete Suche auf Basis der Referenzen der in Phase eins ausgewählten Literatur durchgeführt (*Durchsicht der Referenzen*). Im Rahmen der rückwärtsgerichteten Suche wurden die referenzierten Beiträge aus der Literatur von Phase eins nach dem gleichen Muster wie in der vorhergehenden Phase untersucht und anschließend wurde entschieden, ob sie aufgenommen werden sollen. Darüber hinaus wurden bei der vorwärtsgerichteten Suche Beiträge geprüft, die auf die aus Phase eins ausgewählte Literatur verweisen. Die Ergebnisse dieser Phase wurden nach ihrem jeweiligen Literaturtyp gruppiert. Die Gesamtanzahl der zur Analyse verwendeten Quellen setzt sich zusammen aus der Anzahl der Beiträge nach Bearbeitung sowie der in Phase zwei neu hinzugefügten. Da bei der zweiten Phase allerdings keine weiteren relevanten Beiträge identifiziert werden konnten, ist das Ergebnis der dritten Phase deckungsgleich zu dem der ersten und steht in der rechten Spalte.

Tabelle 4.1: Literaturrecherche nach vorhandenen Ansätzen in den verschiedenen Phasen der Untersuchung

Bereich	1. Initiale Recherche			2. Durchsicht der Referenzen				3. Finale Auswahl
	Treffer	Auswahl nach Begutachtung	Auswahl nach Bearbeitung	Normen	Auswahl von Büchern	Auswahl von Zeitschriftenbeiträgen	Auswahl von Konferenzbeiträgen	
Recherche in Datenbanken mit Suchstring	388	6	4	0	0	0	0	4
Sichtung von Konferenzbänden	2	2	2	0	0	0	0	2

Die final verwendeten Ansätze werden in Tab. 4.2 dargestellt.

Tabelle 4.2: Finale Auswahl an Ansätzen

Bereich	Ansätze der finalen Auswahl
Recherche in Datenbanken mit Suchstring	Engels und Voß (2008), Jung (2006), Schwinn (2005), Vogler (2006)
Sichtung von Konferenzbänden	Gronau et al. (2008), Kattenstroth et al. (2013)

4.2 Deskription und Interpretation der Datengrundlage

Bei der finalen Auswahl der Ansätze (Tab. 4.2) fällt auf, dass die Beiträge ausschließlich im deutschsprachigen Raum publiziert wurden. Drei Ansätze (Jung 2006, Schwinn 2005, Vogler 2006) wurden an der Universität St. Gallen entwickelt. Dies kann dadurch erklärt werden, dass an dieser Institution Kompetenzzentren mit dem Schwerpunkt Integration existierten („Prozess- und Systemintegration“ (CC PSI), „Application Integration Management“ (CC AIM)) und gegenwärtig immer noch existieren („Integration Factory“ (CC IF)).

Auch die Veröffentlichungen, die im Rahmen des EAI-Kompetenzzentrums unter Federführung von Prof. Krallmann an der TU Berlin entstanden sind, wurden im Rahmen der Recherche ausfindig gemacht. Es konnte jedoch kein Ansatz identifiziert werden, welcher die im Vordergrund stehende Problemstellung adressiert. Gleiches gilt für die Arbeitsgruppe „Integration Engineering“⁴¹ an der Universität Leipzig unter Leitung von Prof. Fähnrich.

Bei den Beiträgen aus dem nicht-deutschsprachigen Raum steht weniger die Entwicklung von Ansätzen zum Entwurf von Integrationsarchitekturen im Fokus, sondern vielmehr die Identifikation von Anforderungen oder Faktoren, die eine erfolgreiche Einführung von EAI unterstützen. Stellvertretend hierfür zählen Beiträge, die im Rahmen der iterativen Verfeinerung des Suchstrings in der Datenbank AIS Electronic Library gefunden wurden, die größtenteils die „Halbschwester-Disziplin“ Information Systems abdeckt. Es konnte bspw. der Beitrag „Information Systems and Healthcare XXV: Factors and Actors Affecting the EAI Adoption in the Healthcare Sector“ von Mantzana et al. (2008) identifiziert werden, der die Erfolgsfaktoren zur Einführung von EAI im Wirtschaftszweig Gesundheitswesen zum Gegenstand hat. Insbesondere die beiden Mitautoren Themistocleous und Irani beschäftigen sich seit der Jahrtausendwende mit dem Themenfeld Integration von AWS und haben dazu eine Vielzahl von Beiträgen unter anderem zur Erfolgsfaktorenforschung in diesem Bereich veröffentlicht. Ein weiteres Beispiel, welches die Fokussierung auf die Forschung zu Erfolgsfaktoren unterstützt, ist der Beitrag „Success Factors of Application Integration: An Exploratory Analysis“ von Gericke et al. (2010). Daneben ist auch die Auswirkung der Einführung von EAI auf den Firmenwert Gegenstand von Untersuchungen. Stellvertretend hierfür sei auf den Beitrag „Enterprise Application Integration and its Effect on Business Value: A Stock Market View“ von Roztock und Roland (2006) verwiesen.

Weiterhin fällt auf, dass der Großteil der final ausgewählten Ansätze im Zeitraum 2005 bis 2008 publiziert wurde. Dies lässt sich dadurch erklären, dass dem Forschungsbereich EAI, dem diese Arbeit zuzuordnen ist, in diesem Zeitraum verstärktes Interesse

⁴¹<http://www.integration-engineering.de/>

gewidmet wurde. Auch nach diesem Zeitraum wurde in diesem Themengebiet geforscht, jedoch konnten bis auf Kattenstroth et al. (2013) keine weiteren Ansätze, die sich umfassend mit dem Entwurf von Integrationsarchitekturen auseinandersetzen, identifiziert werden. Das Thema scheint jedoch durch das „Cloud-Computing“ in jüngerer Zeit wieder stärker in den Fokus zu rücken. So konnten während des Aufbaus des Suchstrings stellvertretend hierfür die beiden Publikationen „Enterprise Application Integration - The Cloud Perspective“ von Lässig und Ullrich (2013) sowie „Information Systems Integration in the Cloud: Scenarios, Challenges and Technology Trends“ von Kleeberg et al. (2014) identifiziert werden. Diese beiden Beiträge haben die Einbindung von „Cloud-Diensten“ in die bestehende AWS-Landschaft mittels EAI zum Gegenstand. Da in den Beiträgen aber kein Ansatz zum Entwurf von Integrationsarchitekturen entwickelt wurde, waren sie in einer späteren Iteration der sukzessiven Verfeinerung des Suchstrings nicht mehr bei der finalen Auswahl enthalten.

4.3 Kriterien zur Evaluation von Ansätzen

Zur Evaluation der gefundenen Ansätze werden Anforderungen spezifiziert. Ausgangspunkt für diese ist die in Abschnitt 1.1 definierte Problemstellung. Die Anforderungen orientieren sich dabei an den drei Ebenen der SOM-Unternehmensarchitektur, wie in Abschnitt 2.1 eingeführt. Sie nehmen Bezug auf die Unterstützung der Zustandserhebung, der Analyse sowie der Pragmatik und unterscheiden sich in inhaltliche (I) und formale Anforderungen (F).

I1 Unterstützung bei der korrelativen Gestaltung von GP- und AT-Ebene Bei Änderungen an bestehenden GP sind die induzierten Integrationsbedarfe zu ermitteln. Änderungen können sich dabei durch Modifikation auf GP-Ebene oder auf Ressourcenebene ergeben. Hierbei sind die entstehenden Integrationsbedarfe zu ermitteln und mit dem bestehenden Zustand für Prognosezwecke zu vergleichen. Darüber hinaus ist eine Möglichkeit vorzusehen, dass Änderungen an der Ressourcenebene auf die GP-Ebene übertragen werden können, soweit eine Anpassung erforderlich ist.

I2 Berücksichtigung der Vorgaben des Unternehmensplans als Rahmenbedingung bei der Gestaltung der Integrationsarchitektur Beim Entwurf der Integrationsarchitektur ist es erforderlich, auf mögliche Einschränkungen der Freiheitsgrade von Seiten des Unternehmensplans hinzuweisen. Dies betrifft bspw. Vorgaben über einzusetzende AWS. Mit der Anforderung wird folglich definiert, inwiefern vorhandene Ansätze auf etwaige Beschränkungen der Gestaltungsspielräume Bezug nehmen.

- I3 Modellgestützte Ableitung fachlicher Integrationsbedarfe aus dem GP** Integration von Aufgaben über AT kann technologie- oder prozessgetrieben erfolgen. Technologieorientierung birgt das Problem, dass Integration betrieben wird, wo keine erforderlich ist. Bei Prozessorientierung hingegen hat jede Aufgabenintegration über AT ihren Ursprung im GP. Deshalb besteht die Anforderung darin, Integrationsbedarfe für AT, idealerweise modellgestützt, aus der Beziehung zwischen Aufgaben herzuleiten.
- I4 Beachtung der Potenziale der Aufgabenintegration über personelle AT bei Gestaltung der Integrationsarchitektur** Identifizierte fachliche Integrationsbedarfe zwischen Aufgaben sind über AT abzudecken. Hierbei sind die Potenziale zu berücksichtigen, die die personellen AT bei der Aufgabenintegration bieten. Dazu soll geprüft werden, inwiefern vorhandene Ansätze darauf hinweisen, Aufgaben über personelle AT zu integrieren.
- I5 Berücksichtigung der Möglichkeit der Aufgabenintegration durch AWS-Integration bei Gestaltung der Integrationsarchitektur** Die Aufgabenintegration mittels AwS kann neben der Kopplung vorhandener AwS auch durch die Erweiterung der Aufgabenunterstützung eines AwS oder durch die Ersetzung mehrerer vorhandener AwS durch ein einzelnes erreicht werden. Die Anforderung besteht darin, zu untersuchen, inwiefern vorhandene Ansätze die unterschiedlichen Möglichkeiten der Aufgabenintegration mittels AwS differenziert beleuchten und dennoch die vorhandene AwS-Landschaft berücksichtigen.
- F1 Verfügbarkeit von Techniken zur Komplexitätsreduktion** Zur Unterstützung bei der Analyse komplexer Modelle ist es hilfreich, wenn die Ansätze Techniken zur Komplexitätsreduktion bieten. Hierunter fallen bspw. Architekturen zur Untergliederung eines Modellsystems in mehrere Modellebenen unter jeweiligen Blickwinkeln wie Innen- und Außensicht, Sichten innerhalb der Modellebenen (Abschnitt 2.1), Trennung zwischen Nutzer- und Basismaschine (Abschnitt 2.3), Typ- und Attributmodellierung sowie Verknüpfung der Ebenen durch definierte Beziehungen, Beziehungsmetamodelle oder integrierte Metamodelle (Abb. 2.1). Auch die Verfügbarkeit von Modellsichten zählt dazu. Mithilfe dieser kann der Modellnutzer die Ansicht des Modells auf einen Teil reduzieren. Darüber hinaus ist bei dieser Anforderung auch die Möglichkeit der hierarchischen Zerlegung von Modellen oder Modellelementen von zentraler Bedeutung. Diese unterstützt den Modellnutzer bei der Analyse und Gestaltung der Modelle sowohl auf groben als auch auf detaillierten Ebenen.
- F2 Verfügbarkeit grafischer Modellrepräsentationen** Grafische Modellrepräsentationen betrieblicher Aufgaben sowie deren zugeordnete AT sind in der Regel für den Modellnutzer leichter zu erfassen als textuelle Darstellungen, wie Fließtext,

Tabellen oder Matrizen. Darüber hinaus können diese zwischen den verschiedenen Beteiligten besser kommuniziert und damit die Verifikation der Modellabbildung erleichtert werden. Zur Verifikation von Modellabbildungen zählen unter anderem die Überprüfung der Struktur- und Verhaltenstreue auf Vollständigkeit sowie Richtigkeit hinsichtlich der Modellierungsziele. Aus den genannten Gründen ist es hilfreich, wenn die Ansätze grafische Modelldarstellungen, wie in Abschnitt 3.1.5 skizziert, zur Verfügung stellen können.

4.4 Analyse und Bewertung gefundener Ansätze

In diesem Abschnitt werden die in Tab. 4.2 final ausgewählten Ansätze kritisch gewürdigt. Darüber hinaus wird geprüft, ob die Ansätze die in Abschnitt 4.3 aufgestellten Anforderungen erfüllen. Bei der Bewertung der jeweiligen Anforderung wird unterschieden, ob sie erfüllt (●), bedingt erfüllt (◐) oder nicht erfüllt bzw. kein Hinweis auf Erfüllung gegeben wird (○).

Im Rahmen der nachfolgenden Vorstellung wurde die in den jeweiligen Ansätzen verwendete Terminologie auf die in der vorliegenden Arbeit eingeführte übertragen.

4.4.1 Quasar Enterprise

Im Beitrag von Engels und Voß (2008) wird das Framework Quasar Enterprise vorgestellt. Dieses unterstützt die serviceorientierte Gestaltung von AWS-Landschaften. Es besteht aus 20 Verfahrensbausteinen, die sich aus den drei Dimensionen des Frameworks ergeben und das vorgeschlagene Vorgehen unterstützen. Der Beitrag ist ein Ausschnitt aus dem Hauptwerk von Engels et al. (2008), in dem die Verfahrensbausteine ausführlicher beschrieben und konkrete Anwendungsbeispiele gegeben werden.

In Abb. 4.1 wird das Framework Quasar Enterprise visualisiert. Daraus wird ersichtlich, dass in der Horizontalen nach den Schichten Kontext, Konzept, Logik und Physik gegliedert wurde. Die Vertikale wurde aufgeteilt in die Aspekte Geschäft und IT. In der Tiefe werden die Zustände der Anwendungslandschaft nach IST, SOLL und IDEAL differenziert.

Der Ablauf bei der Gestaltung serviceorientierter AWS-Landschaften wird anhand der fünf Pfeile in Abb. 4.1 verdeutlicht. Jeder Schritt im Ablauf wird durch unterschiedlich viele Verfahrensbausteine unterstützt. Zunächst ist die Geschäftsarchitektur zu analysieren. Darauf aufbauend ist die ideale AWS-Landschaft zur Orientierung zu definieren. Im dritten Schritt ist die Evolution der AWS-Landschaft zu planen. Die Integration von Komponenten in einer AWS-Landschaft ist Gegenstand des vierten Schrittes. Abschließend wird eine unterstützende Integrationsplattform ausgewählt.

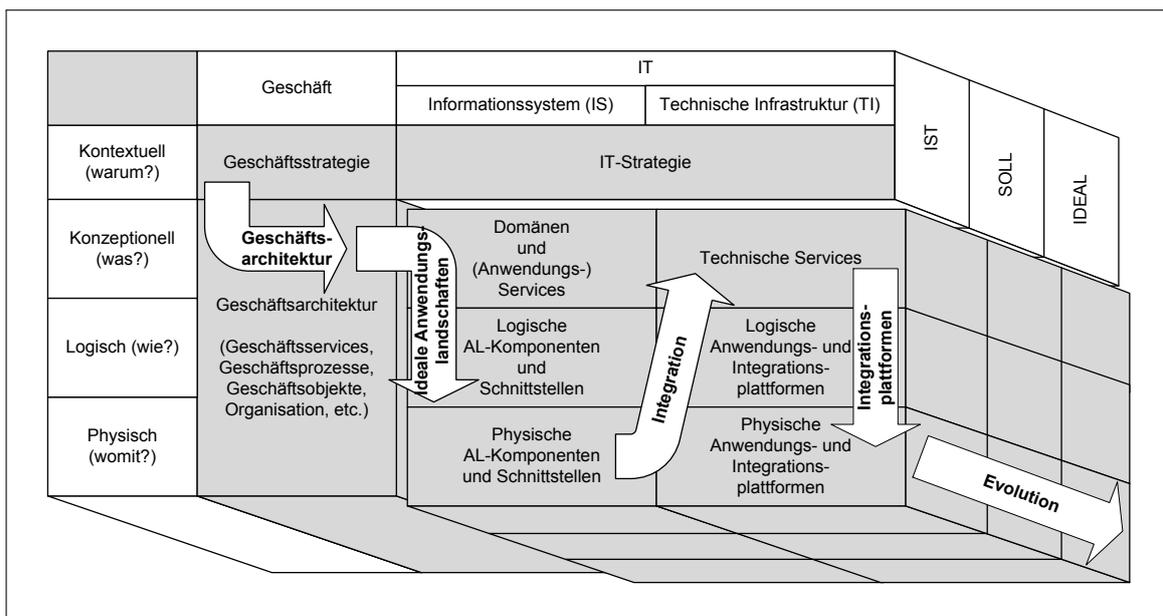


Abbildung 4.1: Quasar Enterprise Framework (Engels et al. 2008, S. 551)

Eine Aufzählung und detaillierte Beschreibung der Verfahrensbausteine ist nicht Gegenstand der Untersuchung, deshalb wird zur Vertiefung auf Engels und Voß (2008) sowie Engels et al. (2008) verwiesen.

Der Ansatz von Engels und Voß (2008) wird in Tab. 4.3 bewertet.

Tabelle 4.3: Evaluation des Ansatzes von Engels und Voß (2008)

Kriterium	Bewertung	Begründung
What-If-Analyse (I1)	●	Ein Vergleich von Szenarien ist durch die Berücksichtigung der Zustände der Anwendungslandschaft unterteilt nach IST, SOLL und IDEAL möglich. Optionen zur Rückübertragung von auf der Anwendungslandschaft ausgehenden Modifikationen auf GP-Ebene werden nicht skizziert.
Unternehmensplan (I2)	●	Unternehmensziele werden zwar angesprochen, mögliche Rahmenbedingungen, die die Freiheitsgrade beschränken, werden nicht definiert.
Integrationsbedarfe (I3)	●	Integrationsbedarfe begründen sich aus den Interaktionen zwischen Geschäftsservices auf Aufgabenebene. Aus diesen leiten sich die zu entwerfenden Schnittstellen und die zu gestaltende Kopplungsarchitektur ab.

Tabelle 4.3 – Fortsetzung von vorhergehender Seite

Kriterium	Bewertung	Begründung
personelle AT (I4)	○	Gestaltungspotenziale bei der Aufgabenintegration über personelle AT werden nicht ausgeschöpft, da sie als Möglichkeit zur Gestaltung der Aufgabenintegration nicht in Erwägung gezogen werden.
AwS (I5)	●	Für die Gestaltung der AwS-Integration werden verschiedene Entwicklungsformen genannt. Neben der Ablösung vorhandener AwS durch neu einzuführende oder zu entwickelnde, die größere Aufgabenkomplexe integrieren, wird auch die Möglichkeit der Neuordnung von Aufgaben zu AwS, die somit größere Aufgabenkomplexe integrieren, in Erwägung gezogen. Darüber hinaus wird auch die Alternative der Kopplung von AwS zur Unterstützung der Aufgabenintegration vorgeschlagen.
Komplexitätsbeherrschung (F1)	●	Es wird zwischen der Aufgaben- (Geschäft) und AT-Ebene (IT) getrennt. Die von den Autoren vorgeschlagene Aufteilung in Domänen fördert die Komplexitätsbeherrschung, da damit die Analyse eines zu untersuchenden Ausschnitts unterstützt wird, und gleichzeitig die von der Analyse ausgeschlossenen Teilbereiche ausgeblendet werden können. Funktions- und Interaktionssicht werden von Komponenten- sowie Anwendungsfalldiagrammen aus der UML unterstützt. Zur Unterstützung der Vorgangssicht werden die Verwendung von Sequenz- und Aktivitätsdiagrammen aus der UML vorgeschlagen. Ein Hinweis für die Unterstützung der Datensicht wird nicht gegeben. Dennoch wird diese Anforderung als erfüllt betrachtet, da zahlreiche Techniken zur Komplexitätsreduktion genannt werden.
Grafisch (F2)	●	Engels et al. (2008) nutzen durchgängig ein jeweils die Ausführungen unterstützendes UML-Diagramm zur visuellen Demonstration der Verfahrensbausteine.

4.4.2 Beitrag von Gronau

Gronau et al. (2008) stellen in ihrem Beitrag einen Ansatz vor, in dem mithilfe von Benutzerbefragungen und Softwarekarten⁴² fehlende AwS-Kopplungen aufgedeckt werden. Sowohl das Vorgehensmodell als auch die Lösungsansätze zur Integration werden anhand eines Fallbeispiels aus dem öffentlichen Sektor demonstriert.

Die drei Schritte *Erhebung*, *Modellierung* und *Auswertung* stellen das Vorgehen bei der Analyse von AwS-Landschaften dar.

Die AwS-Landschaft sowie die Kopplung der AwS untereinander wird mittels schriftlicher Befragung der Nutzer erhoben. Über Telefoninterviews wird bei den Nutzern eine Einschätzung zur Relevanz der Aufgabenintegration mittels AwS-Kopplung bislang noch nicht gekoppelter AwS ermittelt. Daneben wird auch eine Einschätzung zu den Verbesserungspotenzialen durch AwS-Kopplung erfragt (Gronau et al. 2008, S. 427-428).

Die Darstellungen der Modellierung fokussieren die Anwendungsebene und blenden Infrastrukturkomponenten, wie Netzwerkkomponenten, aus. Auf dieser Ebene ist es möglich den Aufgaben und Organisationseinheiten die unterstützenden AwS zuzuordnen. Zur Modellierung schlagen Gronau et al. (2008, S. 428-430) die Verwendung eines der von Lankes et al. (2006, S. 318-325) vorgestellten Typen von Softwarekarten vor.

Die Modelle unterstützen die Auswertung, welche AwS miteinander zu koppeln sind (Gronau et al. 2008, S. 430-434). Dazu werden zunächst zur Visualisierung die AwS mittels Kanten miteinander verbunden, die aufgrund der Aufgabenvernetzung in Beziehung stehen. Dies gilt sowohl für bereits existierende AwS-Kopplungen als auch für die Aufgaben, die mittels personeller AT integriert werden. Durch die Stärke der Kanten kann die Regelmäßigkeit der Nutzung der Kommunikationsbeziehung verdeutlicht werden. Dort wo das Netz aus Kommunikationsbeziehungen besonders dicht ist, wird eine Domäne gebildet. Innerhalb dieser sind die betroffenen AwS zu koppeln, sofern sie noch nicht gekoppelt sind. Dabei sind datenschutzrechtliche Beschränkungen zu berücksichtigen. Wenn bspw. der Austausch personenbezogener Daten nicht erlaubt ist, dann dürfen die AwS nicht gekoppelt werden. Zur Kopplung der AwS selber schlagen die Autoren die Nutzung von Punkt-zu-Punkt-Verbindungen, Hub-and-Spoke-Architekturen oder das Paradigma Serviceorientierter Architekturen (SOA) vor. Im Bedarfsfall können Leistungen einer Domäne anderen Domänen mittels standardisierter Schnittstellen zur Verfügung gestellt werden. Für diese ist entweder ein Zugang für die nutzenden personellen AT einzurichten oder sie sind als „Shared Services“ zu koppeln.

⁴²Der Begriff der Softwarekartografie wird in Abschnitt 3.1.5 eingeführt.

Der Ansatz von Gronau et al. (2008) wird in Tab. 4.4 bewertet.

Tabelle 4.4: Evaluation des Ansatzes von Gronau et al. (2008)

Kriterium	Bewertung	Begründung
What-If-Analyse (I1)	○	Ein Vergleich von Szenarien ist bei diesem Ansatz nicht vorgesehen, da nicht nach AwS-Landschaften, die im zeitlichen Verlauf entstehen können differenziert wird. Eine Möglichkeit zur Rückübertragung von auf der Anwendungslandschaft ausgehenden Modifikationen auf GP-Ebene wird ebenfalls nicht aufgezeigt.
Unternehmensplan (I2)	●	Bis auf den Verweis auf die Einschränkung der Freiheitsgrade beim Entwurf von Integrationsarchitekturen aufgrund datenschutzrechtlicher Beschränkungen werden keine weiteren Hinweise darauf gegeben, welche Rahmenbedingungen darüber hinaus zu beachten sind und wie sie ermittelt werden können.
Integrationsbedarfe (I3)	○	Aufgaben- und AT-Ebene werden bei diesem Ansatz nicht voneinander getrennt. Daher begründen sich Integrationsbedarfe in erster Linie aus den von den AwS benötigten Daten und nicht durch fachliche Anforderungen.
personelle AT (I4)	○	Gestaltungspotenziale bei der Aufgabenintegration mittels personeller AT werden nicht ausgeschöpft, da sie nicht in Erwägung gezogen werden.
AwS (I5)	●	Beim Entwurf der Integrationsarchitektur liegt der Fokus darauf, bislang noch nicht gekoppelte AwS zu koppeln, sofern ein Datenaustausch erforderlich ist. Potenziale durch Neuordnung bei der Aufgabenunterstützung durch AwS werden nicht ausgeschöpft. Hierzu zählt bspw., dass der Bereich der Aufgabenunterstützung eines AwS ausgeweitet wird, vorausgesetzt das AwS unterstützt den zu erweiternden Aufgabenbereich. In diesem Fall wäre keine Kopplung zu anderen AwS erforderlich, da die Aufgaben bereits durch das AwS integriert werden. Der Schwerpunkt liegt somit auf der Umsetzung des Integrationskonzepts datenflussorientierte Funktionsintegration (Abschnitt 3.3.2.2).

Tabelle 4.4 – Fortsetzung von vorhergehender Seite

Kriterium	Bewertung	Begründung
Komplexitätsbeherrschung (F1)	●	Es wird auf die Verwendung eines der in Lankes et al. (2006, S. 318-325) vorgestellten Typen von Softwarekarten verwiesen. Mithilfe dieser Softwarekarten können Sichten auf Funktionen, Interaktionen und Vorgänge gebildet werden. Ein Hinweis für die Unterstützung der Datensicht wird nicht gegeben. Zudem fehlen Verweise auf die hierarchische Zerlegung der Modelle.
Grafisch (F2)	●	Die Methode selbst sieht keine durchgängige grafische Notationen zur Visualisierung vor. Zur Visualisierung von AWS-Landkarten wird auf die Verwendung eines der in Lankes et al. (2006, S. 318-325) vorgestellten Typen von Softwarekarten zu verwiesen.

4.4.3 Der Ansatz von Jung

In der Habilitationsschrift von Jung (2006) wird der „Zusammenhang zwischen Informationsbedarf und Zustand der Anwendungsarchitektur einerseits und der Auswahl geeigneter Integrationsarchitekturtypen andererseits erklärt“ (Jung 2006, S. 267). Dazu werden vier Forschungsfragen beantwortet. Die erzielten Ergebnisse fließen in den Methodenvorschlag für die initialen Phasen eines Integrationsvorhabens, insbesondere der Durchführung einer qualitativen Informationsbedarfsanalyse und einer daran anschließenden Vorstudie zur Ermittlung grundsätzlich geeigneter Architekturtypen zur Synchronisation von Daten, ein (Jung 2006, S. 233).

Zunächst hat Jung (2006, S. 106-126) analysiert, mit welchen qualitativen Merkmalen Informationsbedarfe beschrieben werden können (Forschungsfrage 1). Anschließend wird der Einfluss der Komponenten der Anwendungsarchitektur auf die Erfüllbarkeit der qualitativen Merkmale des Informationsbedarfs untersucht (Forschungsfrage 2; Jung 2006, S. 126-152). Das daraus resultierende Wirkungsnetz wird in Abb. 4.2 dargestellt. Dabei befinden sich in den Rechtecken im unteren Teil die von Jung vorgeschlagenen Komponenten der Anwendungsarchitektur. In der oberen Hälfte werden die identifizierten qualitativen Merkmale eines Informationsbedarfs abgebildet. Die gerichteten Kanten signalisieren einen unterstützenden Einfluss einer Komponente auf ein Merkmal oder eines Merkmals auf ein anderes.

Ferner untersucht Jung, wie von den zur Verfügung stehenden Integrationstechnologien zu abstrahieren ist, so dass Integrationsarchitekturtypen entstehen und anhand welcher

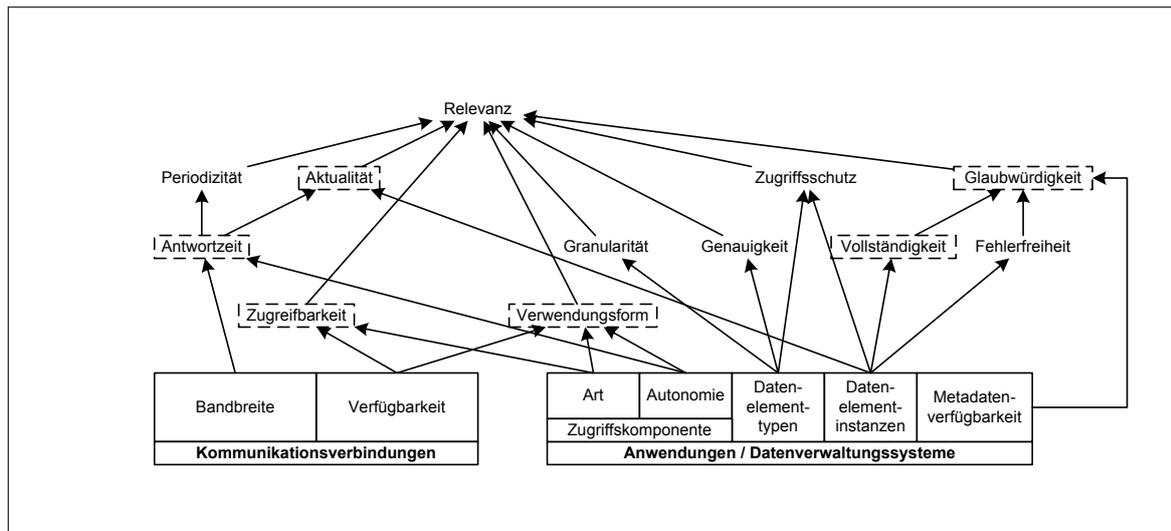


Abbildung 4.2: Einfluss von Komponenten der Anwendungsarchitektur auf die Merkmale des qualitativen Informationsbedarfs (Jung 2006, S. 216)

Merkmale diese abgegrenzt werden können (Forschungsfrage 3; Jung 2006, S. 153-210). Die dabei herausgearbeiteten Merkmale sowie deren mögliche Ausprägungen zur Abgrenzung der Integrationsarchitekturtypen werden in der Tab. 4.5 dargestellt.

Tabelle 4.5: Morphologischer Kasten zu Konstruktionsmerkmalen von Integrationsarchitekturen (Jung 2006, S. 194-201)

Merkmal	Merkmalsausprägung		
	Föderation	Fusion	Multilaterale Kopplung
Architekturtopologie	Föderation	Fusion	Multilaterale Kopplung
Replikation	repliziert	virtuell	teilweise migriert
Transaktionstyp	lesen	lesend und schreibend	
Synchronisierungskontrolle lokal-zu-global	synchron	asynchron	
Synchronisierungskontrolle global-zu-lokal	synchron	asynchron	

Abschließend wird in der Arbeit geprüft, welche relevanten Merkmale des Informationsbedarfs von den unterschiedlichen Integrationsarchitekturtypen unterstützt werden (Forschungsfrage 4; Jung 2006, S. 211-232). Die dabei identifizierten relevanten Merkmale des Informationsbedarfs werden in Abb. 4.2 durch ein gestricheltes Rechteck hervorgehoben.

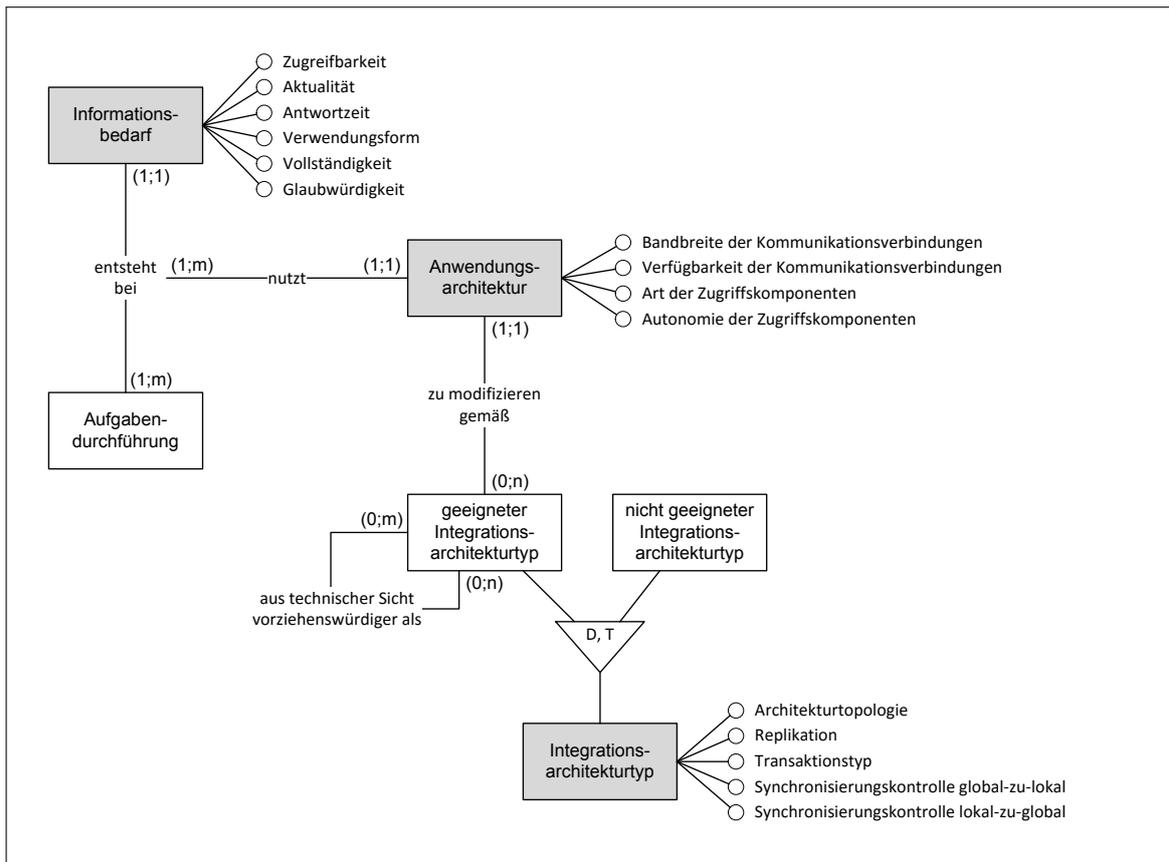


Abbildung 4.3: Informationsmodell für die Vorstudie (Jung 2006, S. 239)

Aus den gewonnenen Erkenntnissen entwickelt Jung (2006, S. 233-265) einen Methodenvorschlag. Dieser orientiert sich am Konzept der Methodenbeschreibung von Gutzwiller (1994, S. 13) und besteht aus einem Vorgehens- und einem Informationsmodell sowie unterstützenden Techniken. Das Vorgehensmodell besteht aus drei Aktivitäten (Jung 2006, S. 240-247). Sie sind im korrespondierenden Informationsmodell in Abb. 4.3 grau hinterlegt. Wie aus dem Informationsmodell ersichtlich ist, sind im Rahmen der ersten Aktivität vom Systemgestalter Ausprägungen für die Merkmale des Informationsbedarfs festzulegen – in Abb. 4.2 die durch ein gestricheltes Rechteck hervorgehobenen Merkmale sowie in Abb. 4.3 links oben. Im zweiten Schritt ist die Anwendungsarchitektur in Bezug auf ihre Merkmale zu bewerten – die Komponenten in den Rechtecken im unteren Teil der Abb. 4.2 sowie der mittlere Teil der Abb. 4.3. Anschließend sind die Ausprägungen für die Konstruktionsmerkmale festzulegen – Tab. 4.5 sowie der untere Teil in Abb. 4.3. Abschließend werden mithilfe einer Regelbasis die Architekturtypen eliminiert, die die Merkmalsausprägungen, die in den drei vorhergehenden Schritten definiert wurden, nicht unterstützen (Jung 2006, S. 241-247).

Der Ansatz von Jung (2006) wird in Tab. 4.6 bewertet.

Tabelle 4.6: Evaluation des Ansatzes von Jung (2006)

Kriterium	Bewertung	Begründung
What-If-Analyse (I1)	○	Diese Methode sieht einen Vergleich von Ist- und Sollzuständen nicht vor, da eine Historie der Anwendungsarchitektur ausgeblendet wird. Darüber hinaus wird eine Möglichkeit zur Rückübertragung von auf der Anwendungsarchitektur ausgehenden Modifikationen auf GP-Ebene ebenfalls nicht aufgezeigt.
Unternehmensplan (I2)	○	Mögliche Einschränkungen beim Entwurf der Integrationsarchitektur von Seiten des Unternehmensplans werden nicht adressiert.
Integrationsbedarfe (I3)	○	Bei der Ermittlung der Integrationsbedarfe wird nicht zwischen Aufgaben- und AT-Ebene getrennt. Die Integrationsbedarfe ergeben sich aus den Datenbeständen, die die AwS benötigen, und nicht durch fachliche Anforderungen.
personelle AT (I4)	○	Gestaltungspotenziale bei der Aufgabenintegration mittels personeller AT werden nicht ausgeschöpft, da sie bei der Aufgabenintegration über AT nicht berücksichtigt werden.
AwS (I5)	●	Bei der Gestaltung der AwS-Unterstützung wird ein umfangreiches Rahmenwerk möglicher Integrationsarchitekturtypen vorgestellt, welches auch die Potenziale durch Neuordnung bei der Aufgabenunterstützung durch AwS ausschöpft.
Komplexitätsbeherrschung (F1)	○	Es werden keine Hinweise auf Möglichkeiten zur Komplexitätsreduktion gegeben. Es fehlen bspw. Verweise auf hierarchische Zerlegung der Modelle, so dass Details, die bei der Analyse nicht von Interesse sind, ausgeblendet werden können. Auf unterstützende Sichten zur Komplexitätsbeherrschung finden sich ebenfalls keine Hinweise.
Grafisch (F2)	○	Der Entwurf wird nicht durch grafische Visualisierungen unterstützt.

4.4.4 Der Ansatz von Kattenstroth

Kattenstroth et al. (2013) stellen in ihrem Beitrag einen Ansatz auf Basis der Multiperspektivischen Unternehmensmodellierung (MEMO) (Frank 2014, S. 941-962) vor, in dem mithilfe grafischer Visualisierung Integrationsbedarfe in Informationssystemen aufgedeckt und analysiert sowie darauf aufbauend Lösungen erarbeitet und bewertet werden können. Der Fokus liegt dabei auf der Vermeidung von Inkonsistenzen verursacht durch redundante Datenbestände.

Kernbestandteile bei der Nutzung des Ansatzes sind das Zielsystem, der zu analysierende GP, ein Ausschnitt aus dem Workflow des zu untersuchenden GP, die unterstützenden AwS sowie die jeweiligen Nutzer- und Basismaschinen (Kattenstroth et al. 2013, S. 92).

Ausgangspunkt für die Gestaltung von Integrationsarchitekturen für Informationssysteme ist das Zielsystem. In diesem werden Ziele für die durchzuführende Analyse vorgegeben. Anschließend wird für den betroffenen Ausschnitt des GP der Ist-Zustand des Workflows modelliert. Darauf aufbauend werden die bestehenden Beziehungen zwischen den betroffenen AwS identifiziert. Dazu wird geprüft, welche Daten in den involvierten AwS redundant gehalten werden und wie sie übertragen werden. An dieser Stelle kann bspw. festgestellt werden, dass die erforderlichen Daten bereits mittels AwS-Kopplung oder durch personelle AT übertragen werden. Im Anschluss daran wird untersucht, ob eine Änderung an den Beziehungen prinzipiell einen Zielbeitrag leisten kann. Zu diesem Zweck wird der Ausbau bestehender AwS-Kopplungen oder die Überführung der Datenübertragung von personellen AT auf maschinell unterstützte bewertet. Für vielversprechende Änderungen sind vom Systemgestalter Lösungsverfahren zur AwS-Kopplung zu erarbeiten. Zu den Lösungsverfahren zählen unter anderem die Implementierung von Batchverarbeitungsprogrammen zur Übertragung von Daten auf Datenverwaltungsebene oder von Schnittstellen in der Anwendungssoftware zur Übertragung von Daten auf Ebene der Anwendungsfunktionen. Diese sind hinsichtlich diverser Aspekte, wie erwartete Auswirkung auf die Datenredundanz, ökonomischer Nutzen etc., zu bewerten. Die vom Lösungsverfahren erwarteten Effekte sind im Zielsystem, im GP-Modell und im Workflow zu annotieren.

In Tab. 4.7 wird der Ansatz von Kattenstroth et al. (2013) bewertet.

Tabelle 4.7: Evaluation des Ansatzes von Kattenstroth et al. (2013)

Kriterium	Bewertung	Begründung
What-If-Analyse (I1)	⦿	Dieser Ansatz berücksichtigt die Möglichkeit eines Ziel-Ist-Vergleiches. Möglichkeiten zur Rücküberführung von auf der Ressourcenebene ausgehenden Modifikationen auf GP-Ebene werden nicht skizziert.
Unternehmensplan (I2)	⦿	Bei diesem Ansatz sind strategische Ziele der Ausgangspunkt für den Entwurf der Integrationsarchitektur. Rahmenbedingungen, die auf Ebene des Unternehmensplans vorgegeben werden und Freiheitsgrade beim Entwurf einschränken können, werden jedoch nicht adressiert.
Integrationsbedarfe (I3)	●	Integrationsbedarfe stammen von fachlichen Beziehungen auf Aufgabenebene. Sie werden mithilfe des Ansatzes modellgestützt abgeleitet.
personelle AT (I4)	⦿	Personelle AT werden bei der Erhebung des Ist-Zustandes berücksichtigt. Dessen Potenziale werden jedoch bei der Gestaltung der Soll-Integrationsarchitektur außer Acht gelassen.
AwS (I5)	⦿	Die Gestaltung der AwS-Integration bewegt sich auf dem Ist-Zustand und zieht die Möglichkeit der Neuordnung von Aufgaben zu AwS nicht in Erwägung. Der Fokus liegt folglich darauf, bislang noch nicht durch AwS unterstützte Transaktionen durch eine korrespondierende AwS-Kopplung zu unterstützen, sofern ein signifikant positiver Zielbeitrag erwartet wird. Schwerpunktmäßig wird dabei die Umsetzung des Integrationskonzepts datenflussorientierte Funktionsintegration (Abschnitt 3.3.2.2) verfolgt.
Komplexitätsbeherrschung (F1)	●	Die Modellierung des Modellsystems wird in mehrere Modellebenen unterteilt und unterstützt somit die Komplexitätsbeherrschung. Darüber hinaus werden Sichten auf Funktionen, Interaktionen sowie Vorgänge vorgestellt. Die Datensicht wird teilweise unterstützt, da mit dieser zu übertragende Nachrichtentypen, nicht aber deren Attribute dargestellt werden. Dennoch wird diese Anforderung als erfüllt betrachtet, da auf zahlreiche Techniken zur Komplexitätsreduktion verwiesen wird.

Tabelle 4.7 – Fortsetzung von vorhergehender Seite

Kriterium	Bewertung	Begründung
Grafisch (F2)	●	Der Entwurf wird durch einen grafischen Ansatz, der in Kattenstroth et al. (2013, S. 93) exemplarisch vorgestellt wird, unterstützt.

4.4.5 Der Ansatz von Schwinn

Schwinn (2005) entwickelt in seiner Dissertation eine Methode zur Gestaltung von Integrationsarchitekturen für Informationssysteme. Der Fokus der Methode liegt darauf, bislang fehlende und dennoch erforderliche AWS-Integrationen aufzudecken und dem Systemgestalter beim Entwurf von AWS-Integrationen zu assistieren. Der komplette Vorgang wird unterstützt durch ein umfangreiches Vorgehensmodell.

Zur Entwicklung der Methode identifiziert Schwinn zunächst typische Anforderungen und Ziele im Rahmen von AWS-Kopplungsprojekten aus der Praxis (Schwinn 2005, S. 47-56). Parallel dazu baut er ein Rahmenkonzept für ein Vorgehensmodell zur Gestaltung von Integrationsarchitekturen auf (Schwinn 2005, S. 57-64). Anschließend werden drei Fallstudien vorgestellt und deren spezifischer Beitrag mit dem entwickelten Rahmenkonzept zusammengebracht (Schwinn 2005, S. 65-126). Die dabei gewonnenen Erkenntnisse fließen in den Methodenentwurf ein (Schwinn 2005, S. 127-238).

Zur Beschreibung der Methode wird Bezug auf das von Gutzwiller (1994, S. 13) vorgeschlagene Konzept der Methodenbeschreibung genommen. Daher sind die Kernbestandteile ein Metamodell, ein Vorgehensmodell, ein Dokumentationsmodell sowie ein Rollenmodell mit typischen Rollen, die aus den herangezogenen Fallstudien extrahiert wurden. Nachfolgend wird das Vorgehensmodell (Abb. 4.4) überblicksartig vorgestellt, da dieses den umfangreichsten Anteil in der Arbeit von Schwinn einnimmt und darüber hinaus als Basis für die Evaluation des Ansatzes in Tab. 4.8 herangezogen wird.

Das von Schwinn vorgeschlagene Vorgehensmodell zur Unterstützung von AWS-Integrationsprojekten wird in Abb. 4.4 abgebildet. Es besteht aus den drei Phasen *Gestaltung der Applikationsebene*, *Gestaltung der Softwarekomponenten- und Datenstrukturebene* sowie dem *Integrationsarchitekturmanagement*. Die Phasen enthalten wiederum Hauptaktivitäten, die in einzelne Aktivitäten gegliedert sind (Schwinn 2005, S. 57-58).

Ausgangspunkt beim Vorgehensmodell ist die Identifikation von Integrationsbedarfen. Bei diesem Schritt werden die Auswirkungen eines neuen oder veränderten GP auf

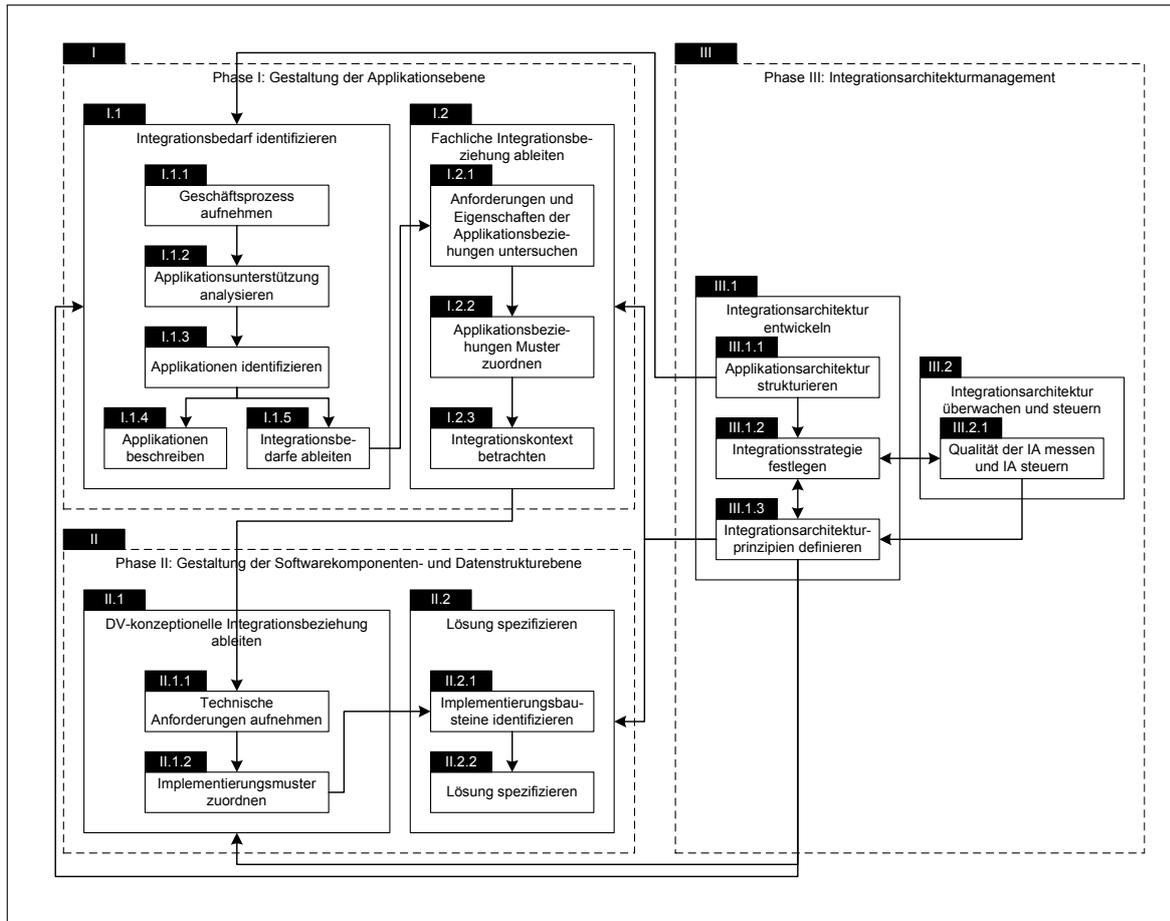


Abbildung 4.4: Integriertes Vorgehensmodell für AwS-Integrationsprojekte (Schwinn 2005, S. 153)

die AwS-Ebene analysiert (Schwinn 2005, S. 58-59). Dazu wird geprüft, ob Aufgabenvernetzungen durch bestehende AwS-Integrationen bereits unterstützt werden. Anschließend sind fachliche Integrationsbedarfe abzuleiten. Darunter werden jedoch die technologischen Rahmenbedingungen für die Integration der AwS verstanden, da bei dieser Aktivität bspw. zu prüfen ist, ob der Nachrichtenaustausch unmittelbar nach Eintreten eines Ereignisses stattfinden soll oder zeitversetzt. Zur Systematisierung werden die im oberen Bereich der Abb. 4.5 dargestellten Entwurfsmuster für fachliche Integrationsbeziehungen vorgeschlagen (Schwinn 2005, S. 170-177; Schwinn 2006, S. 49-57).

Für die aufgedeckten Integrationsbedarfe ist auf AwS-Ebene eine geeignete Integrationsarchitektur zu entwerfen. In Abstimmung zu den technischen Anforderungen und den identifizierten Entwurfsmustern für fachliche Integrationsbeziehungen sind korrespondierende Entwurfsmuster für konzeptionelle Integrationsbeziehungen für die Datenverarbeitung (DV) festzulegen. Die möglichen Muster werden im unteren Bereich der Abb. 4.5 dargestellt. Mögliche Implementierungen eines jeden Entwurfs-

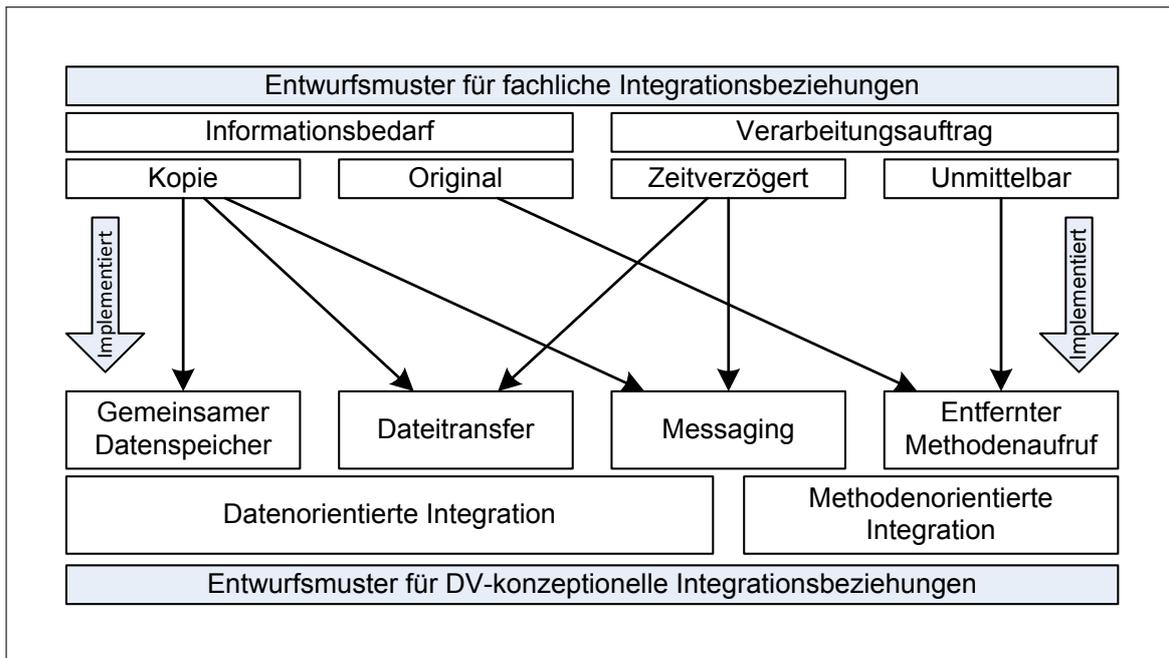


Abbildung 4.5: Implementierung fachlicher Muster durch DV-konzeptionelle Muster (Schwinn 2005, S. 180)

musters für fachliche Integrationsbeziehungen durch ein entsprechendes Pendant auf DV-Ebene werden ebenfalls in Abb. 4.5 dargestellt. Darauf aufbauend wird in Einklang zu den Prinzipien des Integrationsarchitekturmanagements die zu verwendende Middleware festgelegt. Anschließend werden die Technologien, die zur Integration eingesetzt werden dürfen, ausgewählt. Auch dies erfolgt in Abstimmung zu den Prinzipien des Integrationsarchitekturmanagements (Schwinn 2005, S. 60-62).

Das Integrationsarchitekturmanagement ist die übergeordnete Aufgabe, in der Rahmenbedingungen bezüglich der Realisierung eines Integrationsbedarfs definiert werden. Dies beinhaltet zum einen die Definition übergreifender Vorgaben, bspw. darüber, welche Middleware eingesetzt werden darf. Zum anderen adressiert es die Aufgabe der Steuerung und Überwachung der Integrationsarchitektur (Schwinn 2005, S. 62-64).

In Tab. 4.8 wird der Ansatz von Schwinn (2005) bewertet.

Tabelle 4.8: Evaluation des Ansatzes von Schwinn (2005)

Kriterium	Bewertung	Begründung
What-If-Analyse (I1)	⦿	Mit diesem Ansatz ist ein Vergleich von Szenarien möglich, da zwischen Ist-Zustand des GP und Lösung unterschieden wird. Optionen zur simultanen Gestaltung von GP- und Ressourcenebene werden nicht aufgezeigt.
Unternehmensplan (I2)	●	Auf mögliche Einschränkungen beim Entwurf der Integrationsarchitektur auf Ebene des Unternehmensplans wird hingewiesen, da auf die Berücksichtigung der Integrationsstrategie und die Einhaltung derer Teilziele sowie die definierten Integrationsprinzipien beim Entwurf verwiesen wird (Aktivitäten III.1.2 und III.1.3 in Abb. 4.4).
Integrationsbedarfe (I3)	●	Ausgangspunkt der Integrationsbedarfe sind fachliche Beziehungen auf Aufgabenebene (Aktivitäten I.1.1 bis I.1.5 in Abb. 4.4).
personelle AT (I4)	○	Personelle AT zur Aufgabenintegration werden nicht berücksichtigt, da das Verständnis der Integrationsarchitektur den personellen AT ausklammert. Unter Integrationsarchitektur wird die Kopplung verteilter Funktionen und Daten über AwS hinweg subsumiert (Schwinn 2005, S. 27).
AwS (I5)	⦿	Die Gestaltung der AwS-Integration bewegt sich auf dem Ist-Zustand und zieht die Möglichkeit der Neuordnung von Aufgaben zu AwS nicht in Erwägung. Die vier Entwurfsmuster für DV-konzeptionelle Integrationsbeziehungen (Abb. 4.5) decken die drei der in Abschnitt 3.3.2 vorgestellten Integrationskonzepte mit maschineller Beteiligung ab. Dateitransfer und Messaging decken das Integrationskonzept datenflussorientierte Funktionsintegration ab (Abschnitt 3.3.2.2), gemeinsamer Datenspeicher die Datenintegration (Abschnitt 3.3.2.3) und entfernter Methodenaufruf die Objektintegration (Abschnitt 3.3.2.4).

Tabelle 4.8 – Fortsetzung von vorhergehender Seite

Kriterium	Bewertung	Begründung
Komplexitätsbeherrschung (F1)	●	Durch das Vorgehensmodell wird das Modellsystem sukzessive ausgebaut. Zunächst existiert eine Sicht auf einen Ausschnitt des zu untersuchenden GP. Diese wird sowohl mit der bestehenden AwS-Unterstützung als auch mit den verwendeten sowie zu verwendenden Middleware-Technologien verknüpft. Daneben wird auf die Verwendung eines der in Lankes et al. (2005, S. 1450-1457) vorgestellten Typen von Softwarekarten und den Ansatz von Winter (2006) zur systematischen Visualisierung verwiesen. Mithilfe dieser können Sichten auf Funktionen, Interaktionen und Vorgänge gebildet werden. Hinweise für die Unterstützung der Datensicht werden nicht gegeben. Dennoch wird diese Anforderung als erfüllt betrachtet, da auf zahlreiche Techniken zur Komplexitätsreduktion verwiesen wird.
Grafisch (F2)	●	Verschiedene grafische Modellrepräsentationen werden im Rahmen der analysierten Fallstudien demonstriert. In der Methode selber ist jedoch keine durchgängige grafische Notation zur Visualisierung vorgesehen (Schwinn 2005, S. 159), stattdessen wird auf die Verwendung eines der in Lankes et al. (2005, S. 1450-1457) vorgestellten Typen von Softwarekarten und den Ansatz von Winter (2006) zur systematischen Visualisierung verwiesen.

4.4.6 Der Ansatz von Vogler

Vogler (2006) entwickelt in ihrer Habilitationsschrift zwei Methoden: Eine für die Erstellung der Sicht auf die Prozessintegration und eine für die Erstellung der Sicht auf die Systemintegration. Beide bestehen jeweils aus einem Dokumentationsmodell, einem Vorgehensmodell sowie unterstützenden Techniken und orientieren sich an dem Konzept der Methodenbeschreibung von Gutzwiller (1994, S. 13). Der Einsatz der Methoden kann ausgelöst werden durch die Neugestaltung oder die kontinuierliche Weiterentwicklung von Prozessen. Das übergeordnete Ziel, welches mit den Methoden insgesamt verfolgt wird, ist die evolutionäre Weiterentwicklung bestehender Informationssysteme mittels EAI. Dazu setzen die Methoden einen GP-Entwurf mit Beschreibungen der Aufbau- und Ablauforganisation voraus und haben als Schwerpunkt

die Realisierung dessen auf AT-Ebene. Beide Methoden können unabhängig voneinander eingesetzt werden. Mit der Prozessintegration werden Vorgangsteuerungen für Aktionen⁴³ zur Lösung von Aufgaben entworfen. Mithilfe der Systemintegration wird für in Beziehung stehende Aktionen die Integration auf technischer Ebene gestaltet. Das Vorgehen beider Methoden wird nachfolgend kurz skizziert.

Bei der *Prozessintegration* handelt es sich nicht, wie die Bezeichnung suggeriert, um die geeignete Zusammenstellung von Aufgaben, sondern um den Entwurf und die Implementierung von Vorgangsteuerungen für Aktionen zur Lösung von Aufgaben (Vogler 2006, S. 159).⁴⁴ Für den Entwurf einer Vorgangsteuerung wird zunächst auf Basis vorliegender Prozessdokumentationen sowie des vorhandenen Informationssystems der zu planende Workflow abgegrenzt (*Workflowabgrenzung*). Zudem wird die gewünschte Variante der Prozessintegration ausgewählt und dementsprechend das weitere Vorgehen geplant (Vogler 2006, S. 194-203). Vogler differenziert bei den Varianten der Prozessintegration zwischen konzeptionellen und implementierenden. Zu den konzeptionellen Varianten gehören der ad-hoc, der flexible sowie der transaktionsorientierte Workflow (Vogler 2006, S. 168-173). Bei den Varianten zur Implementierung der Prozessintegration wird unterschieden zwischen autonomen und integrierten Steuerungskomponenten (Vogler 2006, S. 173-188).

Im Anschluss an die *Workflowabgrenzung* folgt die *Desktopintegration*, bei der die einzelnen Aktionen einer Aufgabe definiert und über die Automatisierbarkeit der Steuerung mittels der Merkmale Transaktionsvolumen, Potenzial bzw. Nutzen, Strukturierung, Standardisierung, Arbeitsteilung, Wertschöpfungsanteil, Veränderbarkeit sowie Steuerungsvariante⁴⁵ entschieden wird (Vogler 2006, S. 200-201). Für die Ausprägung der Steuerungsvariante im Rahmen der *Desktopintegration* gibt es die vier Möglichkeiten: manueller Workflow ohne IT-Unterstützung der Steuerung, ad-hoc Workflow, flexibler Workflow sowie transaktionsorientierter Workflow.⁴⁶ An dieser Stelle findet auch der Übergang zur Systemintegration statt, die die Gestaltung der Integration einer Aufgabe durch die unterstützenden AwS zum Gegenstand hat (Vogler 2006, S. 203-213). Mit der abschließenden *Workflowplanung* wird das Ziel verfolgt, den endgültigen Ablauf der Aufgaben zu spezifizieren und dabei die Vorgaben für die

⁴³Für die in Vogler (2006) verwendeten Begriffe Aufgabe und Aktivitäten ist kein Unterschied zu erkennen, so dass davon auszugehen ist, dass sie Synonym zueinander verwendet werden. Beide Begriffe entsprechen dem Verständnis der betrieblichen Aufgabe, wie in der vorliegenden Arbeit eingeführt (Abschnitt 2.2). Auf Workflow-Ebene verwendet Vogler für die Aktionen den Begriff Task.

⁴⁴Die Zusammenhänge zwischen Aufgabe, Vorgangs- bzw. AS sowie Aktivitäten werden in Abschnitt 2.2 beschrieben und in Abb. 2.3 grafisch verdeutlicht.

⁴⁵Die Steuerungsvariante entspricht der AS, wie in Abschnitt 2.2.2 bzw. Abb. 2.3 definiert.

⁴⁶Die Kompatibilität zwischen der Ausprägungen der Steuerungsvariante bei der *Desktopintegration* und der Variante der Prozessintegration bei der *Workflowabgrenzung* wird in Vogler (2006, S. 173) definiert.

Aufbauorganisation, wie Berechtigungskonzepte und Regelungen von Verantwortlichkeiten, festzulegen (Vogler 2006, S. 213-221).

Aus der Beschreibung der Methode zur Prozessintegration wird bereits ersichtlich, dass der Ausarbeitung von Vogler (2006) ein anderes Verständnis zugrunde liegt als der vorliegenden. In der vorliegenden Arbeit werden betriebliche Aufgaben in einem betrieblichen Objekt gekapselt und dieses darf maximal von einem AwS unterstützt werden.⁴⁷ Folglich werden Integrationsbedarfe zwischen betrieblichen Aufgaben ermittelt und nicht wie in Vogler (2006) auf Ebene des Workflows zwischen den Aktionen. Somit wird die Aufgabenintegration über AT nicht adressiert. Dennoch können die auf der Workflow-Ebene gewonnenen Erkenntnisse mit Anpassungen auf die Gestaltung der Integration auf Aufgabenebene übertragen werden.

Der erste Schritt der *Systemintegration* ist die *Interaktionsanalyse*. Sie ist als Vorstudie zu betrachten, in deren Rahmen die Integrationsbeziehungen identifiziert werden, welche durch das Projekt zu modifizieren oder neu zu schaffen sind. Dazu werden neben den unmittelbar neu zu erstellenden Integrationsbeziehungen auch die zu bestehenden bzw. umgebenden AwS festgelegt (Vogler 2006, S. 269-277). Beim darauf aufbauenden *Macroentwurf* einer Integrationsbeziehung werden die zu übertragenden Daten, die Übertragungsweise (Batch, synchron, asynchron etc.) sowie die Periodizität oder der Auslöser definiert (Vogler 2006, S. 277-282). Im Rahmen des *Integrationsdesigns* wird ausgehend von der konzeptionellen Variante die Middleware-Technologie zur Realisierung der Integrationsbeziehung festgelegt (Vogler 2006, S. 282-288). Abschließend werden in der *Integrationsspezifikation* die exakten Vorgaben für die Realisierung von neuen Schnittstellen bzw. für die Konfiguration der Schnittstellen von Standard-AwS sowie der Konfiguration erforderlicher Integrationsmechanismen beschlossen (Vogler 2006, S. 288-292).

In Tab. 4.9 wird der Ansatz von Vogler (2006) bewertet.

Tabelle 4.9: Evaluation des Ansatzes von Vogler (2006)

Kriterium	Bewertung	Begründung
What-If-Analyse (I1)	●	Dieser Ansatz erlaubt einen Vergleich von Szenarien, da zwischen vorliegender Architektur und zu planender differenziert wird. Optionen zur Rücküberführung von auf der Ressourcenebene ausgehenden Modifikationen auf die GP-Ebene werden nicht dargestellt.

⁴⁷vgl. Abschnitt 3.1.4

Tabelle 4.9 – Fortsetzung von vorhergehender Seite

Kriterium	Bewertung	Begründung
Unternehmensplan (I2)	⦿	Auf mögliche Einschränkungen bei der Wahl zulässiger Technologien auf Ebene des Unternehmensplans wird hingewiesen. Weitere Hinweise auf Rahmenbedingungen, die darüber hinaus zu beachten sind, und wie sie ermittelt werden können, sind nicht zu finden.
Integrationsbedarfe (I3)	⦿	Integrationsbedarfe stammen von fachlichen Anforderungen. Sie haben jedoch nicht die Aufgabenintegration über AT zum Gegenstand, sondern die Integration von Aktionen auf Workflow-Ebene.
personelle AT (I4)	●	Die Potenziale personeller AT werden bei der Gestaltung der Architektur, insbesondere bei der Wahl der Variante zur Systemintegration, berücksichtigt (Vogler 2006, S. 253-263).
AwS (I5)	⦿	Bei der Wahl der Variante zur Systemintegration werden für den Entwurf der AwS-Integration zur Vernetzung der Aktionen einer Aufgabe die verschiedensten Gestaltungsmöglichkeiten aufgezeigt (Vogler 2006, S. 253-263). Die Möglichkeit der Neuordnung von Aktionen zu AwS wird nicht in Erwägung gezogen.
Komplexitätsbeherrschung (F1)	●	Es existieren die Sichten Prozessintegration, Desktopintegration, Systemintegration und Informationssystem (Vogler 2006, S. 83-90). Durch die Methoden werden die Sichten sukzessive aufgebaut. Darüber hinaus unterstützen die Vorgehensmodelle der Methoden die Komplexitätsbeherrschung, da sie die hierarchische Zerlegung unterstützen. Somit hat jeder Schritt die der jeweiligen Zerlegungsstufe betreffenden Teilaspekte zum Gegenstand. Zusätzlich werden Beispiele zur Unterstützung der Funktions-, Interaktions- sowie Vorgangssicht gegeben. Die Datensicht wird durch ein tabellarisches Modell unterstützt.
Grafisch (F2)	○	Vereinzelt werden Sachverhalte mittels grafischer Repräsentationen verdeutlicht. Die Methoden selber sehen jedoch keine durchgängige grafische Unterstützung vor.

4.5 Zusammenfassende Bewertung

In diesem Kapitel wurden anhand einer strukturierten Literaturrecherche (Abschnitt 4.1) Ansätze identifiziert, mit denen Integrationsarchitekturen entworfen werden können. Die Ergebnisse aus den in Abschnitt 4.4 untersuchten Arbeiten werden in Tab. 4.10 zusammenfassend gegenübergestellt. Wie aus der Tab. 4.10 entnommen werden kann, deckt jedoch kein Ansatz alle der in Abschnitt 4.3 definierten Anforderungen komplett ab. Eine weitere Erkenntnis aus der Analyse der Ansätze liegt darin, dass bei der überwiegenden Anzahl der untersuchten Beiträge der Schwerpunkt auf dem maschinellen Teil bei der Konzeption der Integrationsarchitektur liegt. Indem der personelle Teil des Informationssystems nicht ausreichend berücksichtigt wird, werden jedoch Gestaltungspotenziale verschenkt. Es ist deshalb erforderlich zu prüfen, unter welchen Umständen Aufgaben über den maschinellen oder über den personellen AT zu integrieren sind. Eine Möglichkeit besteht darin, anhand von Merkmalen die Entscheidungsfindung zu unterstützen (Kapitel 5). Darüber hinaus bieten nicht alle analysierten Ansätze eine grafische Unterstützung (F2) für den Systemgestalter und unterstützen auch nur bedingt die in Abschnitt 2.1 vorgeschlagenen Sichten.

Tabelle 4.10: Gegenüberstellung der Ansätze

		Ansätze					
		Engels und Voß (2008)	Gronau et al. (2008)	Jung (2006)	Kattenstroth et al. (2013)	Schwinn (2005)	Vogler (2006)
Kriterien	What-If-Analyse (I1)	◐	○	○	◐	◐	◐
	Unternehmensplan (I2)	◐	◐	○	◐	●	◐
	Integrationsbedarfe (I3)	●	○	○	●	●	◐
	personelle AT (I4)	○	○	○	◐	○	●
	AwS (I5)	●	◐	●	◐	◐	◐
	Komplexitätsbeherrschung (F1)	●	◐	○	●	●	●
	Grafisch (F2)	●	◐	○	●	◐	○

5 Merkmale als Hinweisgeber zur Gestaltung der Integration von Aufgaben über Aufgabenträger

5.1	Untersuchungsstrategie bei der Merkmalsanalyse	86
5.2	Strategie und Durchführung der Suche nach Literatur zu Merkmalen	86
5.3	Ergebnisse der Literaturanalyse	90
5.4	Bildung des Merkmalskataloges	92
5.4.1	Selektion der durch die Literaturrecherche gefundenen Merkmale	92
5.4.2	Bildung des Kategoriensystems	93
5.4.3	Bildung des initialen Merkmalskataloges	93
5.4.4	Reduktion der Merkmale	93
5.4.4.1	Anforderungen an die Zuverlässigkeit	94
5.4.4.2	Kontext	94
5.4.4.3	Leistungsanforderungen	95
5.4.4.4	Flexibilitätsanforderungen	95
5.4.5	Finaler Merkmalskatalog	96
5.4.5.1	Periodizität des Geschäftsprozesses	97
5.4.5.2	Größe einer zu übertragenden Aufgabenobjekt-Instanz	98
5.4.5.3	Intensität bei der Durchführung einer Transaktion	99
5.4.5.4	Anpassbarkeit der Transaktion	99
5.4.5.5	Nutzungsdauer der Integrationsarchitektur	101
5.5	Handlungsempfehlungen für die Ausprägungen der Merkmale	102
5.6	Zusammenfassung der Ergebnisse der Merkmalsanalyse	107

Gegenstand dieses Kapitels ist das Vorgehen sowie die Durchführung der Ermittlung von Merkmalen, die auf Aufgabenebene Hinweise zur Gestaltung der Integration geben können. Dadurch soll dem Anwender bei der Entscheidungsfindung, ob eine Integration maschinell zu unterstützen ist, assistiert werden. Das Kapitel ist wie folgt gegliedert: In Abschnitt 5.1 wird die grundsätzliche Vorgehensweise der Analyse der Merkmale vorgestellt. Anschließend werden Strategie und Durchführung der Literaturrecherche wiedergegeben (Abschnitt 5.2). Darauf aufbauend werden in Abschnitt 5.3 die Ergebnisse präsentiert. In Abschnitt 5.4 wird der Merkmalskatalog gebildet. Im Rahmen der Bildung des Merkmalskataloges wird über die Verwertbarkeit der gefundenen Merkmale diskutiert. Eine pragmatische Verwendungsmöglichkeit der Merkmale zur Entscheidungsfindung wird in Abschnitt 5.5 gezeigt. In Abschnitt 5.6 werden die Ergebnisse zusammengefasst.

5.1 Untersuchungsstrategie bei der Merkmalsanalyse

Bei der Analyse der Merkmale wird die folgende Untersuchungsstrategie verfolgt: Mittels einer strukturierten Literaturrecherche soll zunächst eine nach Möglichkeit breit angelegte Basis an Merkmalen aufgebaut werden (Abschnitt 5.2 und Abschnitt 5.3). Ziel ist es, systematisch eine möglichst vollständige Liste auf Grundlage vorhandener Literatur zu bilden. Anschließend werden aus dieser Liste die Merkmale ausgewählt, mithilfe derer die Anforderungen an die Integration aus Sicht der Aufgabe spezifiziert werden können. Danach werden logisch zusammengehörende Merkmale gruppiert. Das so entstehende System an Merkmalen wurde weiteren Untersuchungen unterzogen sowie einer Gruppe von Fachexperten zur Diskussion vorgestellt. Mit der Diskussion sollen die Relevanz und Vollständigkeit der gefundenen Merkmale sichergestellt werden. Abschließend wurden die Diskussionsergebnisse eingearbeitet.

5.2 Strategie und Durchführung der Suche nach Literatur zu Merkmalen

Die angewandte Strategie orientiert sich, wie die in Abschnitt 4.1 beschriebene Literaturrecherche, an den Richtlinien von Webster und Watson (2002, S. xvi). Ausgangspunkt ist die Suche nach Merkmalen, mithilfe derer die Anforderungen an die Integration aus Sicht der Aufgabe spezifiziert werden können. Diese Merkmale sollen bei ihrer Anwendung Hinweise darüber geben, wie die Integration von Aufgaben über die AT-Ebene zu gestalten ist. Mit der systematischen Literaturrecherche wird dabei das Ziel verfolgt, anhand der gefundenen Literatur eine möglichst breite sowie vollständige Basis von Merkmalen zu bilden, um darauf aufbauend diejenigen Merkmale herauszufiltern, die das Anforderungskriterium erfüllen. So soll sichergestellt werden, dass möglichst viele relevante Merkmale berücksichtigt werden.

Zunächst wurden die Forschungsbereiche in der Wirtschaftsinformatik und den wirtschaftsinformatiknahen Wissenschaften Betriebswirtschaftslehre und Informatik identifiziert, die sich mit einer verwandten Zielsetzung, der Findung von Qualitätsattributen, auseinandersetzen. Die drei als relevant erachteten Bereiche sind nichtfunktionale Anforderungen aus dem Requirements Engineering, Informationsattribute aus dem Bereich der Informationsbedarfsanalyse sowie Integrationsanforderungen aus den Forschungen zur EAI.

Das Forschungsfeld nichtfunktionale Anforderungen aus dem Requirements Engineering wurde aufgenommen, da sich dieses mit der Erhebung von Anforderungen auseinandersetzt und damit eine ähnliche Zielsetzung wie in diesem Kapitel verfolgt wird. Lediglich der Fokus dieses Bereiches unterscheidet sich, da er schwerpunktmäßig die AT-Ebene im Blick hat, die Untersuchungen dieses Kapitels sich jedoch

auf die Anforderungen an die Integration aus Sicht der Aufgabenebene konzentrieren. Das Forschungsfeld nichtfunktionale Anforderungen wird bei der Suche dennoch berücksichtigt, da erwartet wird, dass in diesem Attribut gefunden werden können, die auf den hier vorliegenden Kontext übertragen werden können.

Im Bereich der Informationsbedarfsanalyse werden unter anderem Eigenschaften vorgeschlagen, mithilfe derer Informationsbedarfe beschrieben werden können (L. J. Heinrich et al. 2011, S. 180-181). Da zu erwarten ist, dass ein Teil dieser Eigenschaften auch zur Erhebung der Anforderungen an die Integration aus Sicht der Aufgabe verwendet werden kann, wird dieser Bereich bei der Recherche einbezogen.

Da sich die vorliegende Arbeit im Umfeld der EAI bewegt, ist es naheliegend, auch dieses Themengebiet bei der Recherche einzubeziehen. Hierbei ist insbesondere der Begriff der Integrationsanforderungen relevant.

Für die Recherche wurden dieselben sieben Suchmaschinen wie bei dem in Abschnitt 4.1 beschriebenen Vorgehen verwendet. Zusätzliche Konferenzen oder Workshops wie in Abschnitt 4.1 wurden nicht aufgenommen. Die Suche wurde zuletzt Mitte 2014 durchgeführt. Der Zeitraum der Suche berücksichtigt Publikationen, die vom 01. Januar 2004 bis zum 30. Juni 2014 veröffentlicht wurden. Auch bei dieser Suche wurden Suchergebnisse sowohl in deutscher als auch in englischer Sprache einbezogen. Ferner wurde wiederum nur Literatur aufgenommen, die mit vertretbarem Aufwand beziehbar war (Onlinezugriff oder Bestellung über die lokale Bibliothek der Universität Bamberg einschließlich Fernleihe).

Da die Recherche sprachlich auf deutsche und englische Beiträge ausgerichtet wurde, wurden auch die Suchbegriffe zweisprachig gehalten (Tab. 5.1). Die Suchstrings selbst wurden sukzessive verfeinert, bis die Anzahl an Treffern bei den einzelnen Suchmaschinen kleiner 100 war. Diese Menge wurde als angemessenes Gleichgewicht zwischen Aufwand bei der Analyse der Ergebnisse einerseits und der damit zu erwartenden Anzahl an Qualitätsattributen andererseits erachtet. Es ist zu erwarten, dass die unberücksichtigten Publikationen keine zusätzlichen Qualitätsattribute zu denen in der herangezogenen Literatur liefern werden bzw. dass diese bereits in der gefundenen Literatur aufgegangen sind. Daher ist davon auszugehen, dass die Gefahr, durch diese Vorgehensweise relevante Merkmale zu übersehen, als gering einzustufen ist.

In Tab. 5.1 werden die Bereiche mit ihren jeweiligen finalen Suchbegriffen dargestellt. Wie aus der Tab. 5.1 entnommen werden kann, wurden Begriffe trunziert und mittels Boolescher Operatoren miteinander verknüpft. Darüber hinaus wurden Synonyme aufgenommen und zum Begriff Enterprise Application Integration auch dessen Abkürzung EAI verwendet. Da der initiale Suchstring im Bereich nichtfunktionaler Anforderungen sehr viele Ergebnisse zurückgemeldet hat, die sich mit *Methoden zur*

Erhebung selbiger auseinandersetzen, wurde der Suchstring um den Ausschluss von Methoden mittels „NOT“-Verknüpfung erweitert. Für die anderen beiden Bereiche gilt ähnliches. Bei diesen wurden viele Beiträge präsentiert, die vorrangig *Muster* zum Gegenstand haben. Da jedoch nach Qualitätsattributen gesucht wurde, wurden Beiträge, die sich mit Mustern beschäftigen, ebenfalls mittels „NOT“-Verknüpfung ausgeschlossen. Die in Tab. 5.1 vorgestellten Suchbegriffe können ohne Anpassungen in den aufgezählten Datenbanken verwendet werden.

Tabelle 5.1: Verwendete Suchbegriffe zur Findung von Qualitätsattributen

Bereich	Finale Suchbegriffe
Nichtfunktionale Anforderungen	("nicht*funktionale Anforderung*" OR "non*functional Requirement*") AND ("Requirement* Engineering") NOT (approach OR Ansatz OR method* OR Method*)
Informationsbedarfs"-analyse	(Informationsbedarf* OR ("information demand" OR "information needs")) AND (Informationsattribut* OR Informationsmerkmal* OR "information attribut*") NOT (pattern* OR Muster)
Integrationsanforderungen	(Integrationsanforderung* OR "integration requirement*") AND ("Integration Engineering" OR "Enterprise Application Integration" OR EAI) NOT (pattern* OR Muster)

Tab. 5.2 zeigt die Vorgehensweise bei der Literaturrecherche sowie die Anzahl der Treffer bei den jeweils durchgeführten Phasen. Im Rahmen der ersten Phase wurde mit den Suchbegriffen der Tab. 5.1 die initiale Suche durchgeführt. In der Spalte *Treffer* werden dazu die Anzahl gefundener Publikationen zu jedem Bereich angegeben. Auch hier wurde, wie bereits in Abschnitt 4.1, über die weitere Verwendbarkeit aufgrund der Begutachtung des Titels, der Inhaltsangabe und des Inhaltsverzeichnisses entschieden. Kriterium hierfür war die inhaltliche Auseinandersetzung mit Qualitätsattributen. Die Anzahl der nach dieser Phase übrig gebliebenen Literatur ist in der Spalte *Auswahl nach Begutachtung* aufgeführt. Diese Beiträge wurden anschließend bearbeitet, um daraufhin anhand des oben erwähnten Kriteriums über deren Weiterverwendung zu entscheiden. Die Anzahl übrig gebliebener Beiträge nach dieser Phase steht in der Spalte *Auswahl nach dem Lesen*, welche gleichzeitig ein Zwischenergebnis darstellt. In der zweiten Phase wurden, analog zu Abschnitt 4.1, die Referenzen der gelesenen Literatur nach dem gleichen Muster untersucht und es wurde entschieden, ob sie zur Bildung der Basis aufgenommen werden sollten. Die Ergebnisse dieser Phase wurden nach ihrem jeweiligen Literaturtyp gruppiert. Die Gesamtanzahl der zur Analyse verwendeten Quellen steht zu jedem Bereich in der rechten Spalte *Finale Auswahl*.

Tabelle 5.2: Anzahl der Beiträge während den verschiedenen Phasen der Literaturrecherche

Bereich	1. Initiale Recherche			2. Durchsicht der Referenzen				3. Finale Auswahl
	Treffer	Auswahl nach Begutachtung	dem Lesen	Normen	Auswahl von Büchern	Zeitschriftenbeiträge	Konferenzbeiträge	
Nichtfunktionale Anforderungen	71	5	1	1	0	2	1	5
Informationsbedarfsanalyse	61	1	1	0	3	0	0	4
Integrationsanforderungen	65	7	1	0	0	0	0	1

Die Publikation Balzert (2011) schied aus, da anstatt dem relevanten Kapitel, das nichtfunktionale Anforderungen zum Gegenstand hat (Balzert 2011, S. 109-133), einige der darin enthaltenen Referenzen aufgenommen wurden. Dabei wurde die mittlerweile zurückgezogene Norm International Organization for Standardization und International Electrotechnical Commission (2001) durch die sie ersetzende Norm International Organization for Standardization und International Electrotechnical Commission (2011) ausgetauscht.

Obwohl bekannt ist, dass die International Organization for Standardization und International Electrotechnical Commission (2008) ein Modell für Datenqualität zum Gegenstand hat, schied dieses aus. Zum einen, da sie nicht über die strukturierte Literaturrecherche gefunden wurde und zum anderen, da davon auszugehen ist, dass die ebenda publizierten Merkmale ebenfalls in der in Tab. 5.3 verwiesenen Literatur aufgeführt werden.

Die Quellen der final verwendeten Literatur werden in Tab. 5.3 dargestellt.

Tabelle 5.3: Finale Auswahl an Literatur

Bereich	Literatur der finalen Auswahl
Nichtfunktionale Anforderungen	Chen et al. (2013, S. 43), International Organization for Standardization und International Electrotechnical Commission (2011, S. 10-16), Mairiza et al. (2010, S. 315), Penzenstadler et al. (2014, S. 40), Riebisch und Bode (2009, S. 340-341)
Informationsbedarfsanalyse	Alpar et al. (2014, S. 7-10), Jung (2006, S. 106-126), Horváth (2011, S. 310-312), Küpper et al. (2013, S. 215-220)
Integrationsanforderungen	Lam und Shankararaman (2004, S. 44)

5.3 Ergebnisse der Literaturanalyse

Mit der in Tab. 5.3 aufgeführten Literatur kann eine Basis aus 126 Qualitätsattributen gebildet werden. Die zum großen Teil aus den Qualitätsmerkmalen von Software stammenden Attribute werden im vorliegenden Kapitel auf ihre Verwendbarkeit für die Ermittlung von Integrationsbedarfen hin untersucht. Dabei werden einige Attribute von mehreren Autoren genannt. Merkmale, die identische Ziele verfolgen, können als Synonyme zusammengefasst werden. Auf diese Weise kann die Basis auf 107 Qualitätsattribute reduziert werden. Um die Merkmale zu strukturieren, werden sie gruppiert. Dazu werden weitestgehend die Kategorien und die Zuordnung der Attribute in die Kategorien übernommen, wie sie von der International Organization for Standardization und International Electrotechnical Commission (2011), Mairiza et al. (2010), Penzenstadler et al. (2014) sowie Riebisch und Bode (2009) vorgeschlagen werden (Tab. 5.4). Merkmale aus Publikationen ohne Klassifikationssystem werden entsprechend den Gruppen zugeordnet, deren bereits enthaltene Merkmale ähnliche Ziele verfolgen. Sofern sich dabei Attribute nicht in bestehende Kategorien einordnen lassen, werden neue gebildet. Da einige der 107 zusammengefassten Merkmale in mehrere unterschiedliche Kategorien einsortiert werden können, ist die Summe der Merkmale in Tab. 5.4 über die 14 Kategorien hinweg 128. Die aggregierte Anzahl an Merkmalen pro Kategorie wird ebenfalls in der Tab. 5.4 dargestellt. Die vollständige Liste aller gefundenen Qualitätsattribute kann in Tab. A.1 nachgeschlagen werden.

Tabelle 5.4: Kategorien für die Merkmale und deren Herkunft

Nr.	Kategorien	Anzahl an Merkmalen	Herkunft der Kategorien aus der Literatur
1	Gebrauchstauglichkeit	4	International Organization for Standardization und International Electrotechnical Commission (2011, S. 10)
2	Leistungseffizienz	21	International Organization for Standardization und International Electrotechnical Commission (2011, S. 10), Mairiza et al. (2010, S. 315)
3	Kompatibilität	3	International Organization for Standardization und International Electrotechnical Commission (2011, S. 10)
4	Benutzerfreundlichkeit	19	International Organization for Standardization und International Electrotechnical Commission (2011, S. 10), Mairiza et al. (2010, S. 315)
5	Zuverlässigkeit	16	International Organization for Standardization und International Electrotechnical Commission (2011, S. 10), Mairiza et al. (2010, S. 315)
6	Betriebssicherheit	10	International Organization for Standardization und International Electrotechnical Commission (2011, S. 10), Mairiza et al. (2010, S. 315)
7	Funktionssicherheit	1	Penzenstadler et al. (2014, S. 40)
8	Wartbarkeit	15	International Organization for Standardization und International Electrotechnical Commission (2011, S. 10), Mairiza et al. (2010, S. 315), Riebisch und Bode (2009, S. 340-341)
9	Portabilität	7	International Organization for Standardization und International Electrotechnical Commission (2011, S. 10)
10	Weiterentwickelbarkeit	13	Riebisch und Bode (2009, S. 340-341)
11	Nachhaltigkeit	1	Penzenstadler et al. (2014, S. 40)
12	Kosten	1	eigener Vorschlag

Tabelle 5.4 – Fortsetzung von vorhergehender Seite

Nr.	Kategorien	Anzahl an Merkmalen	Herkunft der Kategorien aus der Literatur
13	Integrationselement	14	eigener Vorschlag
14	Technologie	3	eigener Vorschlag

5.4 Bildung des Merkmalskataloges

In Abschnitt 5.4.1 werden die durch die Literaturrecherche gefundenen Merkmale anhand des in Abschnitt 5.1 genannten Kriteriums gefiltert. Zur Strukturierung wird ein Kategoriensystem gebildet, welches in Abschnitt 5.4.2 aufgebaut wird. Der dadurch entstehende initiale Merkmalskatalog wird in Abschnitt 5.4.3 vorgestellt. Dieser dient als Basis zur Reduktion der Merkmale. Die Selektion relevanter Merkmale wird in Abschnitt 5.4.4 durchgeführt. Zu diesem Zweck soll bei der Selektion der Merkmale nicht über jedes Merkmal einzeln diskutiert werden, sondern zunächst über die grundsätzliche Verwendbarkeit der Klasse, der die Merkmale zugeordnet sind. Abschließend wird der finale Merkmalskatalog in Abschnitt 5.4.5 vorgestellt.

5.4.1 Selektion der durch die Literaturrecherche gefundenen Merkmale

Aus der in Abschnitt 5.3 mittels der Literaturrecherche gebildeten Basis an Merkmalen werden diejenigen ausgewählt, die das in Abschnitt 5.1 genannte Kriterium – Spezifikation der Anforderungen an die Integration aus Sicht der Aufgabe – erfüllen. Somit kann die Anzahl relevanter Qualitätsattribute von 107 auf folgende 21 reduziert werden:

Aktualität, Änderbarkeit, Anpassbarkeit, Erweiterbarkeit, Fehlerfreiheit, Flexibilität, Gegenstand, Größe, Intensität, Modifizierbarkeit, Periodizität, Pünktlichkeit, Relevanz, Skalierbarkeit, Verfügbarkeit, Verwendungszweck, Vollständigkeit, Volumen, Weiterentwickelbarkeit, Wiederverwendbarkeit, Zugreifbarkeit.

Die übrigen Merkmale aus Abschnitt 5.3 haben entweder keinen Bezug zur Integrationsproblematik oder sie beziehen sich, in Anlehnung an die SOM-Unternehmensarchitektur (vgl. Abb. 2.2), auf die Ebene des Unternehmensplans oder die Ressourcenebene.

5.4.2 Bildung des Kategoriensystems

Mithilfe der in Tab. 5.4 genannten Kategorien wird ein Klassifikationsrahmen zur Strukturierung für die gefundenen Merkmale gebildet und die Merkmale den entsprechenden Kategorien zugeordnet (Tab. 5.5). Die Kategorien *Leistungseffizienz*, *Portabilität* und *Zuverlässigkeit* werden aus den in der Tab. 5.4 verwiesenen Quellen übernommen; *Kontext* wird neu gebildet. *Kontext* bezeichnet die Gruppe der Merkmale, die sich ausschließlich auf das AO beziehen. *Leistungseffizienz* wird umbenannt zu *Leistungsanforderungen* und *Portabilität* zu *Flexibilitätsanforderungen*. Das Merkmal *Flexibilität* wird somit als Bezeichnung für die Kategorie verwendet und löst den Begriff *Portabilität* ab, da dieser bei der Interpretation weniger Freiheitsgrade als *Flexibilität* bietet. *Portabilität* beschreibt lediglich die Anforderungen hinsichtlich der Übertragbarkeit vorhandener Lösungen auf neue, unvorhergesehene Situationen. Anforderungen an die Erweiterbarkeit sowie Weiterentwickelbarkeit schließt dieser Begriff aus. *Flexibilität* hingegen deckt die genannten Merkmale ebenfalls ab.

5.4.3 Bildung des initialen Merkmalskataloges

In verschiedenen Diskussionsrunden kamen zu den aus Abschnitt 5.4.1 genannten 21 Merkmalen die Merkmale *Fortschreibungsmodus*, *Einfachheit der Integrationsarchitektur*, *Ortsabhängigkeit* und *Zeitabhängigkeit* hinzu. Auch sie werden in das Kategoriensystem einsortiert.

In Tab. 5.5 werden alle 25 initial gefundenen Merkmale dargestellt. Wie daraus zu erkennen ist, kann das Merkmal *Zugreifbarkeit* sowohl in der Kategorie *Anforderungen an die Zuverlässigkeit* als auch in die Kategorie *Kontext* einsortiert werden. *Flexibilität* wird als Merkmal entfernt, da es die Kategorie als solche kennzeichnet. Somit bleiben 24 Merkmale, deren Verwendbarkeit zu diskutieren ist.

5.4.4 Reduktion der Merkmale

Tab. 5.5 ist die Grundlage zur sukzessiven Selektion der Merkmale. Nachfolgend wird erläutert, aus welchen Gründen Kategorien und Merkmale entfernt werden.

Zum einen hat sich bei der sukzessiven Selektion der Merkmale herausgestellt, dass einige der in Tab. 5.5 gelisteten Merkmale bereits anderweitig durch AIM sowie Entwurfsprinzipien abgedeckt werden. Sie stellen daher keine exklusiven Merkmale dar, mithilfe derer Integrationsbedarfe aus Sicht der Aufgabe aufgedeckt werden können, und werden deshalb ausgeschlossen. Zum anderen wird im weiteren Verlauf der Untersuchung deutlich, dass mit einigen Merkmalen Anforderungen an die AT in

Tabelle 5.5: Übersicht über alle initial gefundenen Merkmale

Kategorie	Zugeordnete Merkmale
Leistungsanforderungen	Größe, Intensität, Periodizität, Skalierbarkeit, Volumen
Flexibilitätsanforderungen	Anpassbarkeit, Änderbarkeit, Einfachheit der Integrationsarchitektur, Erweiterbarkeit, Modifizierbarkeit, Ortsabhängigkeit, Weiterentwickelbarkeit, Wiederverwendbarkeit, Zeitabhängigkeit
Anforderungen an die Zuverlässigkeit	Aktualität, Pünktlichkeit, Verfügbarkeit, Zugreifbarkeit
Kontext	Fortschreibungsmodus, Fehlerfreiheit, Gegenstand, Relevanz, Verwendungszwecke, Vollständigkeit, Zugreifbarkeit

Bezug auf die Integration erhoben werden können. Darüber hinaus werden Merkmale ausgeschlossen, die mehrdeutig sind oder sich mit anderen überschneiden.

5.4.4.1 Anforderungen an die Zuverlässigkeit

Die Kategorie *Anforderungen an die Zuverlässigkeit* sowie deren Merkmale wird entfernt, da nach differenzierter Auseinandersetzung festgestellt werden kann, dass mit den darin enthaltenen Merkmalen ausschließlich aufgabenträgerspezifische Anforderungen an die Integration definiert werden können. Bspw. ist der Vertrieb abhängig von Informationen zum AO Kunde. Diese sollten sowohl aktuell als auch pünktlich zugänglich sein. Auch die generelle Zugreifbarkeit und die Verfügbarkeit zum Zeitpunkt des Aufrufs müssen für diese Organisationseinheit gewährleistet werden. Sichergestellt wird das durch die Infrastruktur der AT-Ebene.

5.4.4.2 Kontext

Daneben wird auch die Kategorie *Kontext* sowie die darin enthaltenen Merkmale entfernt, da sie durch die AIM abgedeckt werden können oder Aggregate anderer Merkmale sind. So wird bspw. der Gegenstand durch das AIM *Überlappendes AO* identifiziert (Abschnitt 3.1.3); z. B. ist der Kunde sowohl Gegenstand der Kreditorenbuchhaltung als auch des Vertriebs. Beide Organisationseinheiten operieren auf Teilmengen identischer AO. Dadurch wird auch ersichtlich, welche AO zur Aufgabenerfüllung relevant sind, so dass der Zugriff gewährleistet werden muss und der überlappende Anteil fehlerfrei zu übertragen ist. Aus dem GP geht hervor, wer das AO modifizieren darf (Verwendungszweck). Fortschreibungsmodus (vollständige Übertra-

gung oder Delta-Übertragung) sowie Vollständigkeit (historisierte Übertragung des AO oder Übertragung des AO ohne Historie) überschneiden sich mit dem atomaren Merkmal *Größe*. So wird mit dem Merkmal *Fortschreibungsmodus* angegeben, ob immer das komplette AO Kunde zu übertragen ist oder nur die überlappenden Anteile. Letztlich geht es um die Größe des AO und diese kann mit dem gleichlautenden Merkmal definiert werden. Analog dazu verhält sich die *Vollständigkeit* (Jung 2006, S. 119-120). Auch hier wird mit dem Merkmal *Größe* der geschätzte Umfang des zu übertragenden AO Kunde angegeben.

5.4.4.3 Leistungsanforderungen

Die Kategorie *Leistungsanforderungen* wird um die Merkmale *Skalierbarkeit* und *Volumen* reduziert. *Skalierbarkeit* wird entfernt, da es mehrdeutig ist und sich folglich mit den Merkmalen *Intensität*, *Anpassbarkeit* sowie *Weiterentwickelbarkeit* überschneidet. Das Merkmal *Skalierbarkeit* ist zukunftsbezogen und bringt zum Ausdruck, dass die Integrationsarchitektur an sinkende oder steigende Intensität anpassungsfähig gehalten werden soll. Damit wird die noch nicht planbare Bandbreite bei der Übertragung von AO-Instanzen zum Ausdruck gebracht. Dies ist bspw. der Fall, wenn noch nicht absehbar ist, ob in Zukunft die Anzahl auszutauschender AO-Instanzen vom Typ Kunde zwischen den Organisationseinheiten gleich bleibt, sinkt oder deutlich mehr wird. *Volumen* wird entfernt, da es in seiner eigentlichen Bedeutung den Wunsch nach möglichst wenig unterschiedlichen AO-Typen ausdrückt (Küpper et al. 2013, S. 217-218). AO-Typen entstehen bereits im Rahmen der Aufgabenzerlegung bei der fachlichen Spezifikation. Somit kann die Zerlegung beim Entwurf von Integrationsarchitekturen nicht beeinflusst werden und deshalb sind die entstehenden AO-Typen als gegeben hinzunehmen. *Periodizität*, *Intensität* sowie *Größe* bleiben erhalten. *Periodizität* beschreibt, wie häufig ein GP in einer Periode ausgeführt wird. So wird bspw. der Teilprozess „Erstellung des Sortiments“ bei einem Konsumgüterhersteller einmal pro Jahr durchgeführt. Mit der *Intensität* wird die durchschnittliche Übertragungshäufigkeit von AO-Instanzen innerhalb einer Periode definiert. Bei dem bekannten Beispiel handelt es sich um ca. 20 Änderungen im Sortiment jährlich. Die dementsprechende Anzahl an AO-Instanzen ist im Unternehmen an die davon abhängigen Organisationseinheiten zu propagieren. Die *Größe* spezifiziert letztlich die durchschnittliche Größe einer zu übertragenden AO-Instanz. In diesem Beispiel macht jedes AO ca. zwei DIN-A4 Seiten aus bzw. ist ca. 4 Kilobyte groß.

5.4.4.4 Flexibilitätsanforderungen

Die Kategorie *Flexibilitätsanforderungen* wird um die Merkmale *Einfachheit der Integrationsarchitektur*, *Erweiterbarkeit*, *Ortsabhängigkeit*, *Wiederverwendbarkeit*

sowie *Zeitabhängigkeit* verringert. *Erweiterbarkeit* und *Wiederverwendbarkeit* (Chen et al. 2013, S. 43; Riebisch und Bode 2009, S. 341) beschreiben Formalziele beim Entwurf von Architekturen und wurden aus diesem Grund entfernt. *Einfachheit der Integrationsarchitektur* wird entfernt, da es zum einen ein Aggregat aus Anpassbarkeit und Nutzungsdauer ist und zum anderen den Fokus auf die Abschätzung des Aufwands zur Implementierung der Integrationsarchitektur auf AT-Ebene richtet. Das Merkmal adressiert dabei vordergründig die Fragestellung, wie einfach eingesetzte AwS mit anderen AwS über Schnittstellen gekoppelt werden können. Mit den Merkmalen *Ortsabhängigkeit* und *Zeitabhängigkeit* werden ebenfalls aufgabenträgerspezifische Anforderungen definiert. So ist bspw. für die eingesetzten AwS zu definieren, ob eine Kopie der AO zur orts- und zeitunabhängigen Offline-Benutzung vorgehalten werden soll. Bei der Synchronisation der AO sind Lösungen für die unterschiedlichen Konfliktsituationen zu implementieren – z. B. wenn das AO sowohl vom mobilen AT in der Kopie als auch von einem stationären AT im Original verändert wird. Daneben überschneiden sich beide Merkmale mit der *Verfügbarkeit* sowie darüber hinaus die *Zeitabhängigkeit* mit der *Pünktlichkeit*.

Änderbarkeit, *Modifizierbarkeit* sowie *Weiterentwickelbarkeit* werden zu *Nutzungsdauer* zusammengefasst. Der graduelle Unterschied dieser drei Merkmale spielt für die Bestimmung der Anforderung an die Integration aus Sicht der Aufgabe eine untergeordnete Rolle. Wichtiger ist vielmehr die Bestimmung der Dauer, wie lange einzelne Aufgaben im GP Bestand haben bzw. der GP als Ganzes Bestand hat. Aus diesem Grund wird das Merkmal *Nutzungsdauer* beibehalten. Ebenfalls beibehalten wird das Merkmal *Anpassbarkeit*, da mit diesem definiert werden kann, ob das Lösungsverfahren für die Durchführung der Transaktion zwischen Aufgaben wiederholt oder einmalig ausgeführt wird. Der Wiederholungscharakter darf dabei nicht zu eng gefasst werden. Damit sind nicht „völlig gleichartige, bis ins Detail übereinstimmende Arbeitsoperationen gemeint“ (Kosiol 1976, S. 31), sondern „vielmehr die Aufgaben, bei denen auch im einzelnen ständig wechselnde Tätigkeiten gewisse Gemeinsamkeiten, Übereinstimmungen und Ähnlichkeiten aufweisen, die immer wiederkehren“ (Kosiol 1976, S. 31).

5.4.5 Finaler Merkmalskatalog

Durch die in Abschnitt 5.4.4 durchgeführte Selektion werden die vormals 24 Merkmale auf die in der Tab. 5.6 aufgelisteten fünf reduziert. In der Tab. 5.6 werden zudem kurze Definitionen zu den einzelnen Merkmalen gegeben. Im weiteren Verlauf dieses Abschnitts werden die Merkmale und deren Implikationen in Bezug auf die Integration von Aufgaben beschrieben.

Tabelle 5.6: Merkmale, anhand derer Anforderungen an die Integration aus Sicht der Aufgabe spezifiziert werden können

Kategorie	Merkmal	Definition
Leistungsanforderungen	Periodizität des GP	Beschreibt, wie häufig ein GP in einer Periode durchgeführt wird.
	Größe einer zu übertragenden AO-Instanz	Spezifiziert die durchschnittliche Größe einer zu übertragenden AO-Instanz.
	Intensität bei der Durchführung einer Transaktion	Legt die durchschnittliche Übertragungshäufigkeit von AO-Instanzen im Rahmen der Ausführung eines GP für jede Transaktion fest.
Flexibilitätsanforderungen	Anpassbarkeit der Transaktion	Legt fest, wo die zu untersuchende Transaktion im Bereich zwischen <i>keine Abweichung vom Standardverhalten</i> bis hin zu <i>vielen Abweichungen vom Standardverhalten</i> anzusiedeln ist.
	Nutzungsdauer der Integrationsarchitektur	Bestimmt den geplanten Zeithorizont der Integrationsarchitektur.

5.4.5.1 Periodizität des Geschäftsprozesses

Mit dem Merkmal *Periodizität des GP* wird beschrieben, wie häufig ein GP in einer bestimmten Periode ausgeführt wird. Jung (2006, S. 108-109) bezieht dieses Merkmal auf Berichte. Das Merkmal kann jedoch auch zur Bestimmung, ob eine Aufgabenintegration maschinell zu unterstützen ist, herangezogen werden. Sinngemäße Synonyme sind *Frequenz* (Lam und Shankararaman 2004, S. 44) sowie *Häufigkeit* (Horváth 2011, S. 310-312; Küpper et al. 2013, S. 218). Entscheider können die Periodizität für die Zukunft anhand von Referenzwerten aus der Vergangenheit prognostizieren, sofern solche vorliegen. Andernfalls ist die erwartete Periodizität durch Domänenexperten zu schätzen. Allgemein gilt, dass Aufgabenvernetzungen bei häufig durchgeführten GP durch maschinelle Integration zu unterstützen ist, da der Aufwand der Integration der Aufgaben durch personelle AT als hoch einzustufen ist. Dabei ist die Einschätzung, ob ein GP häufig oder selten durchgeführt wird in Relation zu anderen in der Organisation vorhandenen GP zu setzen.

Zur Verdeutlichung des Merkmals dient nachfolgendes Beispiel: In einem Konsumgüterunternehmen wird der Teilprozess *Auftragsbearbeitung* mehrere hundert mal am

Tag durchgeführt und als häufig eingestuft. Wenn die Aufgaben des Teilprozesses ausschließlich über personelle AT integriert würden, wäre ein hoher Personalbedarf erforderlich. Hingegen bietet die maschinelle Integration Rationalisierungspotenziale. Bei GP, die wie für das Beispiel „Erstellung des Sortiments“ in Abschnitt 5.4.4.3 zum Merkmal Periodizität einmal jährlich und folglich selten durchgeführt werden, genügt hingegen die Integration der Aufgaben durch den personellen AT, da der Aufwand für die Implementierung der maschinellen Unterstützung den damit einhergehenden Nutzen voraussichtlich übersteigen wird.

5.4.5.2 Größe einer zu übertragenden Aufgabenobjekt-Instanz

Die Ausprägung des Merkmals *Größe einer zu übertragenden AO-Instanz* konkretisiert die durchschnittliche Größe einer zu übertragenden AO-Instanz. Das Merkmal entstammt aus Lam und Shankararaman (2004, S. 44) und wird dort als *Size* bezeichnet. Ausgangsbasis für die Messung der Größe sind die überlappenden Attribute der beteiligten AO-Typen. Diese sind in einer bekannten Einheit zu messen, da sie so greifbar gemacht und besser untereinander verglichen werden können. In Abschnitt 5.4.4.3 wurden diese exemplarisch für das erläuternde Beispiel zum Merkmal Größe in den Einheiten DIN-A4 Seiten bzw. Kilobyte gemessen. Sofern genügend Beispieldatensätze, die bei der Interaktion zwischen Aufgaben anfallen, vorhanden sind, können diese für die Messung herangezogen werden. Falls es sich um einen neu zu gestaltenden GP handelt, ist die Größe durch Domänenexperten zu schätzen. Hierfür kann bspw. das KOS unterstützend herangezogen werden. Dabei sind den Aufgaben die KOT zuzuordnen und die überlappenden Attribute der KOT zwischen Aufgaben ausfindig zu machen. Sowohl die überlappenden Attribute als auch die Werte, die sie annehmen können, stellen die Ausgangsbasis für die Schätzung dar. Die allgemeine Empfehlung bei diesem Merkmal lautet, dass die Übertragung von AO-Instanzen bereits ab mittlerer Größe maschinell zu unterstützen ist, da der Aufwand der Übertragung über personelle AT bereits ab dieser Größe als hoch einzustufen ist. Die Einschätzung, ob die Größe einer zu übertragenden AO-Instanz klein oder hoch ist, ist in Relation zu anderen auszutauschenden AO-Instanzen zu setzen.

Die Anwendung des Merkmals wird beispielhaft anhand eines Versandauftrags in einem Konsumgüterunternehmen verdeutlicht. Beim Versandauftrag teilt der Vertrieb dem Versand die vom Kunden bestellten Waren einschließlich der Mengen mit. Im Durchschnitt nimmt dabei ein Versandauftrag weniger als eine DIN-A4 Seite ein. Die zu übertragende AO-Instanz wird daraufhin als klein eingestuft und somit genügt die Integration über personelle AT. Im Rahmen von Transaktionen, bei denen wie im Beispiel von Abschnitt 5.4.4.3 zum Merkmal Größe ca. 4 Kilobyte große AO-Instanzen ausgetauscht werden, ist die Integration maschinell zu unterstützen, da die personelle Unterstützung viel Personal binden würde.

5.4.5.3 Intensität bei der Durchführung einer Transaktion

Mit dem Merkmal *Intensität bei der Durchführung einer Transaktion* wird die durchschnittliche Übertragungshäufigkeit von AO-Instanzen im Rahmen der Ausführung eines GP für jede Transaktion erhoben. Das Merkmal ist eine Weiterentwicklung des als *Volumen* bezeichneten Merkmals von Lam und Shankararaman (2004, S. 44). Da mit dieser Bezeichnung eine Unterscheidung zum Merkmal *Größe einer zu übertragenden AO-Instanz* erschwert würde, wird der eingangs vorgeschlagene Terminus verwendet. Bei etablierten GP bietet es sich für die Erhebung der tatsächlichen Durchführungen einer Transaktion an, die Ereignisprotokolle der AwS auszuwerten (van der Aalst 2011, S. 95-123). Wird der damit verbundene Aufwand jedoch als unverhältnismäßig hoch angesehen bzw. handelt es sich bei der Transaktion um eine neu im GP zu implementierende, dann sind von Domänenexperten Annahmen zu treffen und Schätzwerte zur Intensität abzugeben. Bei diesem Merkmal lautet die allgemeine Empfehlung, dass intensiv genutzte Transaktionen maschinell zu unterstützen sind, da der Aufwand der Vernetzung der Aufgaben über personelle AT als hoch einzustufen ist. Dabei ist die Einschätzung der Intensität bei der Durchführung der Transaktion in Relation zu anderen Transaktionen zu bringen.

Das Merkmal wird exemplarisch anhand der Transaktion *Fertigungsauftrag* im Teilprozess *Produktionsplanung* bei einem Konsumgüterhersteller demonstriert. Der Teilprozess *Produktionsplanung* wird in diesem Fall immer donnerstags durchgeführt. Das Unternehmen hat ein Sortiment von ca. 100 Artikeln, wovon bei der Durchführung der Transaktion für 60 Fertigungsaufträge erstellt werden. Die Intensität bei der Durchführung dieser Transaktion beträgt somit 60 und wird von dem Domänenexperten als hoch eingestuft. Folglich ist die Vernetzung der beteiligten Aufgaben maschinell zu unterstützen. Würden die an der Transaktion beteiligten Aufgaben ausschließlich über personelle AT integriert, wäre ein hoher Personalbedarf erforderlich. Bei Transaktionen, die wie im erläuternden Beispiel von Abschnitt 5.4.4.3 zum Merkmal Intensität ca. 20 Änderungen im Sortiment jährlich vorgenommen werden und folglich wenig intensiv sind, genügt hingegen die Integration der Aufgaben durch personelle AT, da der Aufwand für die Implementierung der maschinellen Unterstützung den damit einhergehenden Nutzen vermutlich übersteigen wird.

5.4.5.4 Anpassbarkeit der Transaktion

Das Merkmal *Anpassbarkeit der Transaktion* spiegelt wider, wo die zu untersuchende Transaktion im Bereich zwischen *keine Abweichung vom Standardverhalten* bis hin zu *vielen Abweichungen vom Standardverhalten* anzusiedeln ist. Das Merkmal stammt von der International Organization for Standardization und International Electrotechnical Commission (2011, S. 15) und wird dort als *Adaptability* bezeichnet. Ursprünglich

verwendet die International Organization for Standardization und International Electrotechnical Commission (2011, S. 15) das Merkmal, um damit den Grad der Effektivität und Effizienz, mit der ein System, Produkt oder eine Komponente auf eine andere Hardware, Software oder Umgebung portiert werden kann, zu beschreiben. Diese Definition richtet sich jedoch an die AT-Ebene und wurde deshalb für den vorliegenden Zweck angepasst. Chen et al. (2013, S. 43) nennen als Attribut ebenfalls das Merkmal *Adaptability*, geben dazu jedoch keine Definition. Um die Ausprägung der Anpassbarkeit zu bestimmen, bietet es sich an, die Transaktionen anhand von Eigenschaften zu bewerten. Zu diesem Zweck werden die Klassifikationssysteme von A. Picot und Rohrbach (1995, S. 30-32) sowie Lillrank (2003, S. 222) zur Differenzierung unterschiedlicher Prozess- und Aufgabentypen auf die vorliegende Problemstellung übertragen. Lillrank differenziert zwischen Standard, Routine und Nicht-Routine. Eine ähnliche Differenzierung nehmen A. Picot und Rohrbach vor. In Tab. 5.7 werden ausgewählte Eigenschaften vorgestellt, die die Zuordnung von Transaktionen in eine der drei Klassen unterstützen. Dazu werden die von A. Picot und Rohrbach (1995, S. 30-32) und Lillrank (2003, S. 222) vorgestellten Eigenschaften zusammengeführt. Auswahlkriterien für eine Eigenschaft sind die Anwendbarkeit auf die Transaktion sowie der Beitrag zur Differenzierung in Bezug auf die Klassenzuordnung. Anhand der Tab. 5.7 wird ersichtlich, dass die Eigenschaften *Lösungsweg* (A. Picot und Rohrbach 1995, S. 32; Lillrank 2003, S. 222) und *Planbarkeit* (A. Picot und Rohrbach 1995, S. 32) für die Einordnung einer Transaktion in eine der drei Klassen herangezogen werden können.⁴⁸

Tabelle 5.7: Eigenschaften von Standard-, Routine- und Nicht-Routinetransaktionen

	Standard-transaktion	Routine-transaktion	Nicht-Routine-transaktion
Lösungsweg	Algorithmus	Gewohnheit	Heuristik
Planbarkeit	Hoch	Mittel	Niedrig

Allgemein ist zu empfehlen, dass Standard- und Routinetransaktionen maschinell zu unterstützen sind. Für Nicht-Routinetransaktionen wird empfohlen, die Aufgaben personell zu integrieren, da der Implementierungsaufwand von einmalig zu verwendenden ad-hoc Lösungsverfahren für Transaktionen bei personellen AT geringer ausfällt als bei maschinellen.

⁴⁸Wagner et al. stellen einen Ansatz vor, mit dem die Flexibilitätsbedarfe von GP mithilfe der Fuzzy-Set-Theorie klassifiziert werden können (Wagner et al. 2011). Auf die Einbindung dieses Ansatzes wurde jedoch zugunsten der einfachen Erlern- und Anwendbarkeit des Regelwerks mit seinen Handlungsempfehlungen verzichtet.

Beim Beispiel Konsumgüterhersteller sind die Parameter bei der Übermittlung von Kommissionieraufträgen bekannt – der Lösungsweg kann als algorithmisch und die Planbarkeit als hoch eingestuft werden. Somit ist die Transaktion als Standardtransaktion zu kennzeichnen und nach der allgemeinen Empfehlung maschinell zu unterstützen, da die Anpassungsfähigkeiten, die die personellen AT bieten, bei dieser Transaktion nicht erforderlich sind. Eine Sonderbestellung beim Lieferanten ist hingegen personell zu unterstützen. Sie tritt ein, wenn kurz vor Beginn oder während der Produktion einer Charge bekannt wird, dass die vorhandene Menge eines Rohstoffs nicht ausreicht, weil sich die Lieferung verzögert oder ein kurzfristig zu produzierender Sonderauftrag eingegangen ist. Bei dieser Transaktion ist der Lösungsweg heuristisch zu bewältigen und die Planbarkeit als niedrig einzustufen. Folglich ist die Transaktion als Nicht-Routinetransaktion aufzufassen.

5.4.5.5 Nutzungsdauer der Integrationsarchitektur

Mit diesem Merkmal wird der geplante Zeithorizont für die Integrationsarchitektur definiert. Für die Einordnung der geplanten Nutzungsdauer bieten sich für den Systemgestalter die allgemein akzeptierten Klassen *kurzfristig*, *mittelfristig* und *langfristig* an. Dabei ist die Nutzungsdauer der Transaktion zu berücksichtigen. Eine Orientierung an der Nutzungsdauer des GP ist nicht empfehlenswert, da sich die Aufgaben des GP fundamental verändern, die Transaktion und die implementierte Integrationsarchitektur aber trotzdem bestehen bleiben können. Allgemein ist zu empfehlen, bereits ab einer mittelfristigen Nutzungsdauer die Integration der Aufgaben durch maschinelle AT zu unterstützen, da dies Stabilität signalisiert und folglich mit wenig Änderungsbedarfen bei der Integrationsarchitektur zu rechnen ist.

Zur Verdeutlichung des Merkmals wird auf das Beispiel des Konsumgüterherstellers zurückgegriffen. Die Transaktion *Fertigungsauftrag* im Teilprozess *Produktionsplanung* wird langfristig vorhanden sein. An dem für die Integrationsarchitektur zur Verfügung stehenden AT sind ebenfalls langfristig keine Änderungen geplant. Daher ist der allgemeinen Empfehlung folgend diese Transaktion maschinell zu unterstützen. Bei der Transaktion *Kommissionierauftrag* hingegen verhält es sich anders. Diese wird langfristig vorhanden sein, aber eines der eingesetzten AwS soll in Kürze abgelöst werden. Bis zur Ablösung ist die Transaktion folglich personell zu unterstützen.

5.5 Handlungsempfehlungen für die Ausprägungen der Merkmale

Um für Außenstehende nachvollziehbare Entscheidungen treffen zu können, genügt es nicht, lediglich die Ausprägungen zu jedem Merkmal zu erheben und auf dieser Basis zu entscheiden, ob die Integration der Aufgaben maschinell unterstützt werden soll. Die auf diese Weise getroffenen Entscheidungen können je nach Subjekt unterschiedlich ausfallen und einer Willkür obliegen. Deshalb ist ein Schema zur Ableitung von Handlungsempfehlungen zu finden. Mögliche Schemata bewegen sich dabei in der Bandbreite zwischen Willkür und formalen Methoden. Das Ziel besteht darin, ein Schema zu konstruieren, welches von Systemgestaltern intuitiv und ohne größere Einarbeitung verwendet werden kann, trotzdem willkürliche und intransparente Entscheidungen weitestgehend reduziert. Deshalb sollten die Merkmale im Zielschema nicht nur erhoben, sondern zugleich die jeweilige Ausprägung in einer Skala eingeordnet werden. Hierfür bietet sich eine dreistufige Skala, wie in Tab. 5.8 dargestellt, an. Durch diese Einordnung kann zwar die Subjektivität nicht gänzlich eliminiert werden, die Willkürlichkeit bei der Entscheidungsfindung kann jedoch durch Transparenz und Nachvollziehbarkeit, indem die Einordnung in die Skala erklärt werden muss, eingedämmt werden.

Tabelle 5.8: Vorlage für die Erhebung der Ausprägungen für jedes Merkmal

Kategorie	Merkmal	Ausprägung		
		niedrig	moderat	hoch
Leistungsanforderungen	Periodizität des GP			
	Größe einer zu übertragenden AO-Instanz			
	Intensität bei der Durchführung einer Transaktion			
Flexibilitätsanforderungen	Anpassbarkeit der Transaktion			
	Nutzungsdauer der Integrationsarchitektur			

Die Einordnung der Ausprägung in die Skala kann entweder von dem Fachexperten anhand seiner Erfahrungen oder formal mithilfe von Metriken vorgenommen werden. Da die formale Erhebung problematisch ist, wenn keine Referenzwerte vorhanden sind und der Aufwand gegenüber der zu treffenden Entscheidung überwiegt, wird

vorgeschlagen, dass der Fachexperte über die Einordnung der Ausprägung in die Skala entscheiden soll.

Nach der Einordnung der Ausprägungen in die Tab. 5.8 ist zu entscheiden, ob die Integration maschinell zu unterstützen ist. Die in Tab. 5.8 vorgenommene Aufteilung ergibt 243 mögliche Kombinationen ($|Ausprägungen|^{Merkmale} \rightarrow 3^5 = 243$). Um den Systemgestalter bei der Entscheidungsfindung zu unterstützen, sollte ihm ein Regelwerk mit Handlungsempfehlungen, welches alle Kombinationen abdeckt, angeboten werden.

Zum Aufbau des zu erstellenden Regelwerks ist es hilfreich, sich den Lösungsraum als Entscheidungsbaum vorzustellen. Jeder Knoten im Baum repräsentiert dabei ein Merkmal, während jeder vom Knoten ausgehende Zweig eine mögliche Ausprägung dieses Merkmals darstellt. Bei der Regelerstellung sind durch Variation der Anordnung der Merkmale diejenigen Knoten zu finden, die einen hohen *Information Gain*⁴⁹ erzielen und dadurch viele Ausprägungen subsumieren. Hierbei wurde die hierarchische Anordnung der Knoten durch ein heuristisches Verfahren, indem die Anordnung der Merkmale systematisch geändert wurde, ermittelt. Ziel dieser Vorgehensweise ist es, eine überschaubare Anzahl an Regeln mit Handlungsempfehlungen zu erzeugen, um dem Systemgestalter auch bei fehlender Rechnerunterstützung helfen zu können. Die finale hierarchische Anordnung der Merkmale im Entscheidungsbaum lautet: *Anpassbarkeit, Periodizität, Größe, Intensität, Nutzungsdauer*. Sie spiegelt sich in den unten dargestellten Handlungsempfehlungen wider.

Das Ergebnis der Vorgehensweise sind die sieben nachfolgend vorgestellten Handlungsempfehlungen. Diese decken alle 243 Ausprägungen ab. Die den Prämissen zugeordneten Handlungsempfehlungen sind das Ergebnis einer argumentativ-deduktiven Analyse, in der die allgemeinen Empfehlungen zu den jeweiligen Merkmalen aus Abschnitt 5.4.5 eingeflossen sind. Die Handlungsempfehlungen wurden nicht empirisch erhoben. Beim Regelwerk gilt: Sobald eine Prämisse zutrifft, sind die nachfolgenden Regeln nicht mehr zu überprüfen.

1. **Prämisse:** Anpassbarkeit ist hoch

Handlungsempfehlung: Aufgabenintegration nicht maschinell unterstützen

Beschreibung: Ist das Merkmal Anpassbarkeit mit *hoch* ausgeprägt, dann ist eine maschinelle Unterstützung der Integration nicht zu empfehlen. So können sich jedes Mal bei Durchführung der Aufgaben auch die Anforderungen an die Integration der betroffenen Aufgaben ändern. Es ist davon auszugehen, dass der fortwährende Aufwand für die Anpassung der maschinellen Unterstützung den

⁴⁹Der *Information Gain* wurde nicht berechnet, sondern es wurde lediglich eine Approximation vorgenommen. Für die Berechnung des *Information Gains* der unterschiedlichen Anordnungen der Merkmale in der Hierarchie kann das Maß der *Entropie* herangezogen werden.

einhergehenden Nutzen übersteigt. Diese Regel deckt $3^4 = 81$ Ausprägungen ab. Es verbleiben die Ausprägungen niedrig und moderat beim Merkmal Anpassbarkeit sowie drei Ausprägungen bei den Merkmalen Periodizität, Größe, Intensität sowie Nutzungsdauer. Folglich sind $2 * 3^4 = 162$ Ausprägungen durch weitere Regeln abzudecken.

2. **Prämisse:** Anpassbarkeit ist nicht hoch und (Periodizität ist hoch oder Größe ist hoch oder Intensität ist hoch oder Nutzungsdauer ist hoch)

Handlungsempfehlung: Aufgabenintegration maschinell unterstützen

Beschreibung: Ist eines der Merkmale Periodizität, Größe, Intensität oder Nutzungsdauer mit *hoch* und das Merkmal Anpassbarkeit nicht mit *hoch* ausgeprägt, dann ist eine maschinelle Unterstützung der Integration zu empfehlen. So ist bspw. bei einer einmalig durchzuführenden Datenübernahme, bei der das Merkmal Größe mit hoch, aber das Merkmal Nutzungsdauer mit kurzfristig bewertet wird, maschinell zu unterstützen, da davon auszugehen ist, dass der Nutzen der maschinellen Unterstützung gegenüber dem Aufwand und den eventuell damit einhergehenden Problemen überwiegt. Mit dieser Regel werden 130 der verbleibenden 162 Ausprägungen abgedeckt ($2^1 * 3^3 + 2^2 * 3^2 + 2^3 * 3^1 + 2^4 * 3^0 = 130$). Es verbleiben jeweils zwei Ausprägungen bei allen Merkmalen. Folglich sind $2^5 = 32$ Ausprägungen durch weitere Regeln abzudecken.

3. **Prämisse:** (Anpassbarkeit ist nicht hoch oder Periodizität ist nicht hoch oder Größe ist nicht hoch oder Intensität ist nicht hoch) und Nutzungsdauer ist niedrig

Handlungsempfehlung: Aufgabenintegration nicht maschinell unterstützen

Beschreibung: Ist das Merkmal Nutzungsdauer mit *niedrig* und eines der Merkmale Anpassbarkeit, Periodizität, Größe oder Intensität nicht mit *hoch* ausgeprägt, dann ist eine maschinelle Unterstützung der Integration nicht zu empfehlen. Die Anwendung dieser Regel kann bei Übergangssituationen eintreten. Bei diesen wird empfohlen, die davon betroffenen Aufgaben ausschließlich über personelle AT zu integrieren, da davon auszugehen ist, dass der Aufwand für die Implementierung einer maschinellen Unterstützung für die Aufgabenintegration bei kurzfristiger Nutzungsdauer den zu erwartenden Nutzen übersteigt. Diese Regel deckt $2^4 = 16$ der verbleibenden 32 Ausprägungen ab. Es verbleiben jeweils zwei Ausprägungen bei den Merkmalen Anpassbarkeit, Periodizität, Größe und Intensität. Bei Nutzungsdauer verbleibt die Ausprägung mittelfristig. Folglich sind $2^4 * 1 = 16$ Ausprägungen durch weitere Regeln abzudecken.

4. **Prämisse:** Anpassbarkeit ist niedrig und (Periodizität ist moderat oder Größe ist moderat oder Intensität ist moderat) und Nutzungsdauer ist moderat

Handlungsempfehlung: Aufgabenintegration maschinell unterstützen

Beschreibung: Ist das Merkmal Anpassbarkeit mit *niedrig* ausgeprägt und eines

der Merkmale Periodizität, Größe und Intensität mit *moderat* sowie Nutzungsdauer mit *mittelfristig*, dann ist eine maschinelle Unterstützung der Integration zu empfehlen. Bei der Integration von Aufgaben, in der geringe Anforderungen an die Anpassbarkeit gestellt werden und eine mittelfristig Nutzungsdauer vorgesehen wird, ist die maschinelle Unterstützung zu empfehlen, da der Nutzen den Aufwand der Implementierung übersteigt. Diese Regel deckt $2 * 2 + 2 * 1 + 1 * 1 = 7$ der verbleibenden 16 Ausprägungen ab. Folglich sind neun Ausprägungen durch weitere Regeln abzudecken.

5. **Prämisse:** Anpassbarkeit ist niedrig und Periodizität ist niedrig und Größe ist niedrig und Intensität ist niedrig und Nutzungsdauer ist moderat
Handlungsempfehlung: Aufgabenintegration nicht maschinell unterstützen
Beschreibung: Ist das Merkmal Anpassbarkeit mit *niedrig* ausgeprägt und die Merkmale Periodizität, Größe und Intensität mit *niedrig* sowie Nutzungsdauer mit *mittelfristig*, dann ist eine maschinelle Unterstützung der Integration nicht zu empfehlen. Es ist zu erwarten, dass in diesen Fällen der Aufwand für die Implementierung der maschinellen Unterstützung höher ausfällt als der damit einhergehende Nutzen. Regel fünf kann nicht durch eine Erweiterung der Regel sieben ersetzt werden, da sich sonst Regel vier und Regel sieben widersprechen. Diese Regel deckt eine der verbleibenden neun Ausprägungen ab. Folglich sind acht Ausprägungen durch weitere Regeln abzudecken.
6. **Prämisse:** Anpassbarkeit ist moderat und ((Periodizität ist moderat und Größe ist moderat und Intensität ist nicht hoch) oder (Periodizität ist moderat und Größe ist nicht hoch und Intensität ist moderat) oder (Periodizität ist nicht hoch und Größe ist moderat und Intensität ist moderat)) und Nutzungsdauer ist moderat
Handlungsempfehlung: Aufgabenintegration maschinell unterstützen
Beschreibung: Ist das Merkmal Anpassbarkeit mit *moderat* sowie Nutzungsdauer mit *mittelfristig* ausgeprägt und mindestens zwei der Merkmale Periodizität, Größe und Intensität mit *moderat*, dann ist eine maschinelle Unterstützung der Integration zu empfehlen. In diesen Fällen ist zu erwarten, dass der Nutzen, der von der maschinellen Unterstützung ausgeht, den für die Implementierung erforderlichen Aufwand übersteigt. Diese Regel deckt vier der verbleibenden acht Ausprägungen ab. Folglich sind vier Ausprägungen durch weitere Regeln abzudecken.
7. **Prämisse:** Anpassbarkeit ist moderat und ((Periodizität ist niedrig und Größe ist niedrig und Intensität ist nicht hoch) oder (Periodizität ist niedrig und Größe ist nicht hoch und Intensität ist niedrig) oder (Periodizität ist nicht hoch und Größe ist niedrig und Intensität ist niedrig)) und Nutzungsdauer ist moderat
Handlungsempfehlung: Aufgabenintegration nicht maschinell unterstützen

Beschreibung: Ist das Merkmal Anpassbarkeit mit *moderat* sowie Nutzungsdauer mit *mittelfristig* ausgeprägt und mindestens zwei der Merkmale Periodizität, Größe und Intensität mit *niedrig*, dann ist eine maschinelle Unterstützung der Integration nicht zu empfehlen. Es ist zu erwarten, dass in diesen Fällen der Aufwand für die Implementierung der maschinellen Unterstützung höher ausfällt als der damit einhergehende Nutzen. Mit dieser Regel werden die verbleibenden vier Ausprägungen abgedeckt.

Zur Unterstützung und Vermeidung von Fehlern bei der Anwendung der Handlungsempfehlungen wurden diese in einem Hilfswerkzeug prototypisch implementiert. In Abb. 5.1 wird ein exemplarischer Ausschnitt des zur Methodenunterstützung entwickelten Prototyps dargestellt. Daraus wird ersichtlich, dass für jedes Merkmal eine Ausprägung aus der jeweiligen Dropdown-Liste auszuwählen ist. Abschließend bestimmt das Programm, welche Regel auf die gewählte Konstellation zutrifft. Im unteren Bereich wird dann die korrespondierende Handlungsempfehlung ausgegeben.

Abbildung 5.1: Unterstützung bei der Ermittlung der Handlungsempfehlung durch den entwickelten Prototypen

5.6 Zusammenfassung der Ergebnisse der Merkmalsanalyse

Im vorliegenden Kapitel wurden Merkmale, mithilfe derer Anforderungen an die Integration aus Sicht der Aufgaben erhoben werden können, herausgearbeitet. Sie sollen dem Anwender bei der Entscheidungsfindung, ob eine Integration maschinell zu unterstützen ist, assistieren. Sie sollen auch isoliert ohne Einbettung in eine Modellierungsmethode eingesetzt werden können. Mit den gefundenen Merkmalen wird die Forschungsfrage, anhand welcher Kriterien die Gestaltung eines Integrationsbedarfs bewertet werden kann, beantwortet. Dazu wurde eine strukturierte Literaturrecherche durchgeführt. Die weitere grundsätzliche Vorgehensweise bei der Analyse der Merkmale wurde in Abschnitt 5.1 erläutert. Strategie und Durchführung der Literaturrecherche wurden in Abschnitt 5.2 vorgestellt. Die darauf aufbauenden Ergebnisse waren Gegenstand von Abschnitt 5.3. Der final gebildete Merkmalskatalog wurde in Abschnitt 5.4 präsentiert. Nach einer differenzierten Auseinandersetzung mit den Möglichkeiten über die Verwendung der vorgeschlagenen Merkmale wurde eine pragmatische Vorgehensweise, die zwischen den beiden Extremen Willkür und formaler Methode anzusiedeln ist, gewählt. Dazu wurden Regeln mit Handlungsempfehlungen zur Unterstützung der Entscheidungsfindung in Abschnitt 5.5 beschrieben.

Die in Abschnitt 5.5 vorgestellte pragmatische Vorgehensweise lässt dem Systemgestalter Freiheitsgrade bei der Festlegung der Ausprägung. Dies kann unter Umständen zu Problemen bei der Ableitung der Handlungsempfehlung führen, da die Ansichten unterschiedlicher Systemgestalter gegebenenfalls nicht deckungsgleich ausfallen. Es sei jedoch darauf hingewiesen, dass die vorgestellte pragmatische Vorgehensweise bei Bedarf durch eine formale Methode ersetzt werden kann. Zu diesem Zweck sind die Ausprägungen der einzelnen Merkmale sowie die Einordnung der Ausprägungen in die Skala anhand von Metriken durchzuführen.

Sofern einigen Handlungsempfehlungen in dem in Abschnitt 5.5 vorgestellten Regelwerk nicht zugestimmt wird, sind diese anzupassen. Der Beitrag von Abschnitt 5.5 bleibt aber dennoch bestehen, solange das Regelwerk selber unverändert bleibt.

6 Entwicklung einer Methode zum Entwurf von Integrationsarchitekturen für Informationssysteme

6.1 Metamodell der Methode	110
6.2 Architekturmodell der Methode	113
6.2.1 Metamodell für das Repräsentationsobjekt Geschäftsprozess	113
6.2.2 Metamodell für das Repräsentationsobjekt Aufgabenträgerzuordnung	114
6.3 Vorgehensmodell der Methode	116
6.3.1 Istanalyse	116
6.3.1.1 Spezifikation des zu analysierenden Geschäftsprozesses	116
6.3.1.2 Erhebung und Analyse des Ist-Zustands	121
6.3.1.3 Hinweise für die Gruppierung von Transaktionen mit Inter-AwS-Kommunikationskanälen	125
6.3.2 Szenarienbildung	126
6.3.2.1 Szenarioorientierte Gestaltung des Geschäftsprozesses	127
6.3.2.2 Erhebung der Freiheitsgrade beim Entwurf der Integrationsarchitektur	128
6.3.2.3 Ermittlung der Soll-Automatisierung für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen	132
6.3.2.4 Ableitung der zu verwendenden Integrationskonzepte für die Inter-AwS-Integrationen	136
6.3.2.5 Optionen bei Unvereinbarkeit zwischen angestrebten und möglichen Integrationskonzept	142
6.4 Techniken und Werkzeuge für die Methodendurchführung	143
6.5 Zusammenfassung und Diskussion	146

Aufbauend auf den Erkenntnissen der vorhergehenden Kapitel wird in diesem Kapitel ein Vorschlag für eine Methode zur Ableitung geeigneter Integrationsarchitekturen aus GP konstruiert, um damit die in Abschnitt 4.5 identifizierte Forschungslücke zu schließen. Wie in Abschnitt 1.4 geschildert, verwendet der Methodenvorschlag die SOM-Methodik⁵⁰ als Grundlage, da sie sowohl objekt- als auch geschäftsprozessorientiert ist und dadurch Defizite reiner daten- sowie prozessorientierter Ansätze überwunden werden können. Zudem bietet sie neben einer Verhaltenssicht auf GP eine Struktursicht. Weitere Anknüpfungspunkte, deren Forschungsergebnisse teilweise im Methodenvorschlag eingearbeitet werden, bilden die Arbeit von Krumbiegel (1997) sowie das Forschungsprojekt OASYS. Darüber hinaus werden die in Kapitel 5 gefundenen Merk-

⁵⁰Die SOM-Methodik wurde in Abschnitt 2.1 im Kontext zur Definition des Begriffs der Unternehmensarchitektur eingeführt.

male in die Methodenentwicklung zur Entscheidungsunterstützung eingebunden, um bestimmen zu können, wie Aufgaben über die AT-Ebene zu integrieren sind.

Ergebnisse der Anwendung der Methode sind sowohl für jedes Szenario geeignete Integrationsarchitekturen für das Informationssystem des GP als auch das für die Implementierung zu verwendende Integrationskonzept.

Die zu konstruierende Methode wird anhand der in den jeweiligen Abschnitten beschriebenen Bestandteilen aufgebaut. In Abschnitt 6.1 wird das *Metamodell* zur Definition eines Begriffssystems eingeführt. Das *Architekturmodell* stellt einen Rahmen zur Bildung der Architektur des Modellsystems dar und wird in Abschnitt 6.2 vorgestellt. Zur Spezifikation des Vorgehens bei der Modellierung wird das *Vorgehensmodell* in Abschnitt 6.3 aufgebaut. Die zur Unterstützung der Tätigkeit des Modellierens unterstützenden Techniken und Werkzeuge werden in Abschnitt 6.4 vorgestellt. Die Bestandteile der Methode werden anhand eines durchgängigen Beispiels in Abschnitt 6.3 verdeutlicht. Dazu wird das in Abschnitt 2.4 eingeführte aus dem Hochschulumfang stammende Fallbeispiel aufgegriffen. Abschließend wird in Abschnitt 6.5 die entwickelte Methode evaluiert und den in Kapitel 4 untersuchten Methoden gegenübergestellt.

6.1 Metamodell der Methode

Als Beschreibungsrahmen für das Metamodell der Methode wird das in Sinz (1996, S. 129) eingeführte und in Abb. 6.1 abgebildete Meta-Metamodell gewählt und die Definition für die Integrationsarchitektur aus Abschnitt 3.1.7 herangezogen.

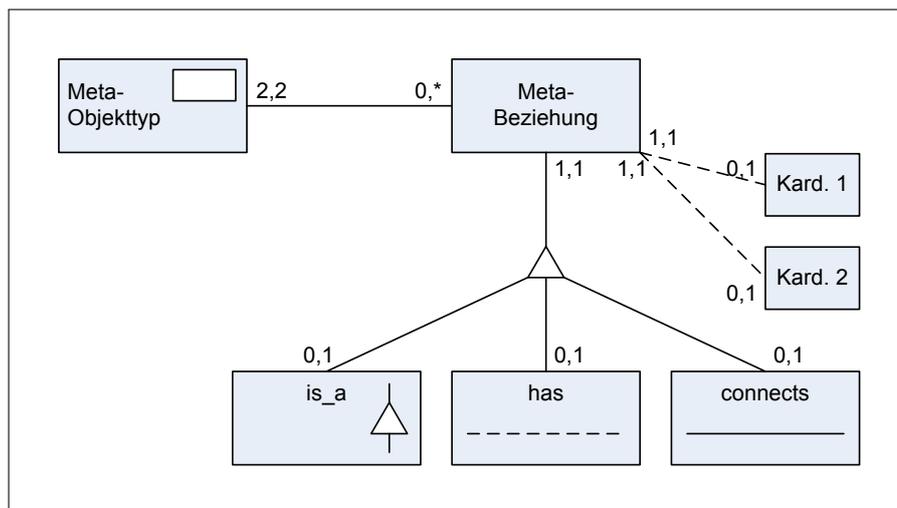


Abbildung 6.1: Meta-Metamodell (Sinz 1996, S. 129)

In Abb. 6.2 wird das aggregierte Metamodell der entwickelten Methode dargestellt. Grundlagen sind das Metamodell der SOM-Methodik für Geschäftsprozessmodelle

(Ferstl und Sinz 2013, S. 218-219) sowie die spezifischen Erweiterungen von Krumbiegel (1997, S. 130-131). Nachfolgend wird zur Erläuterung von Abb. 6.2 nur auf die spezifischen Erweiterungen von Krumbiegel (1997, S. 130-131) eingegangen. Für die ausführliche Erläuterung des Metamodells der SOM-Methodik für Geschäftsprozessmodelle wird auf die Beschreibung in Ferstl und Sinz (2013, S. 218-219) verwiesen.

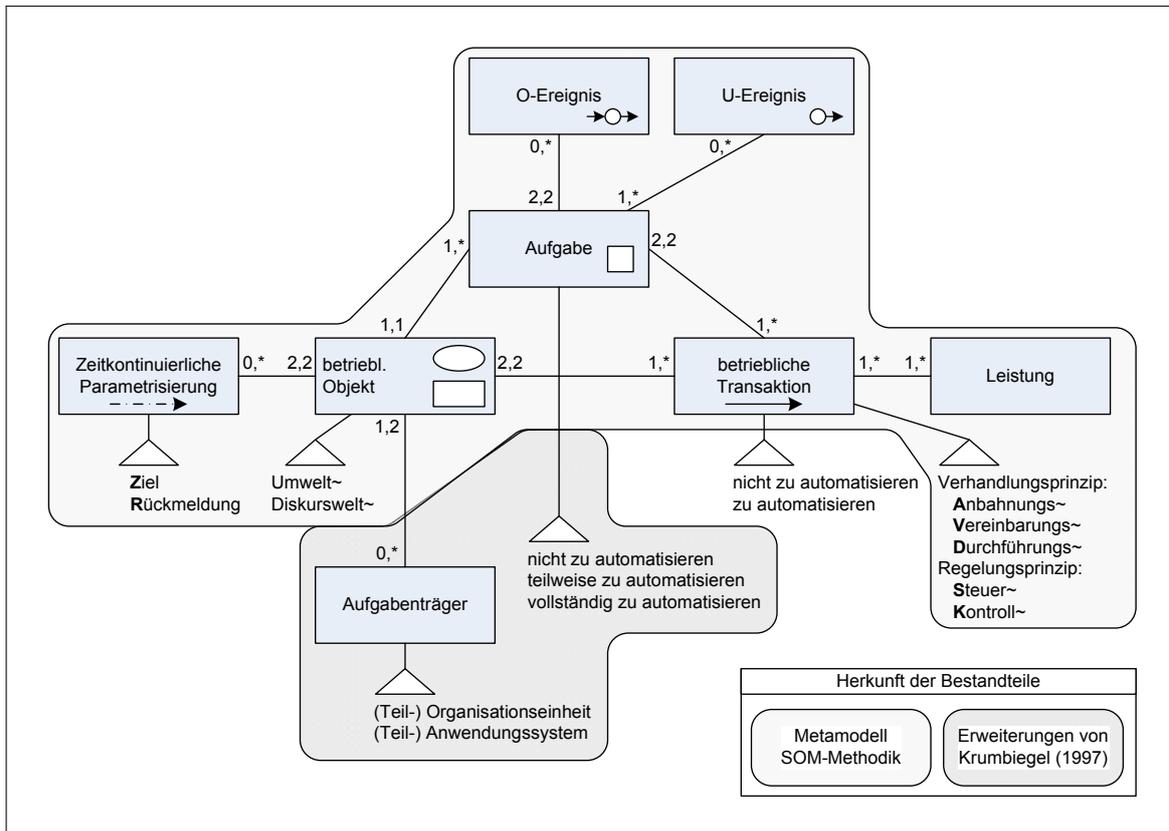


Abbildung 6.2: Aggregiertes Metamodell der Methode (in Anlehnung an Ferstl und Sinz (2013, S. 219) sowie Krumbiegel (1997, S. 131))

Das Metamodell der Methode (Abb. 6.2) erlaubt es, dass betrieblichen Objekten⁵¹ ein personeller und/oder ein maschineller AT zugeordnet werden kann. Aufgaben sind, wie in Abschnitt 2.2.2 beschrieben und in Tab. 2.1 dargestellt, hinsichtlich Automatisierbarkeit, Automatisierungsforderung sowie Realisierung zu unterscheiden. In Bezug auf die Automatisierung gilt Gleiches für betriebliche Transaktionen. Die Herkunft der Bestandteile wird mithilfe von Layern verdeutlicht. Der nicht mit Layern unterlegte Bestandteil kennzeichnet den Erweiterungsvorschlag, der im Rahmen dieser Arbeit ausgebaut wird.

In Abb. 6.3 wird das ausführliche Metamodell dargestellt. Es erweitert das von Krumbiegel (1997, S. 131-133) eingeführte Metamodell. Ausgangspunkt ist das aggregierte Metamodell von Abb. 6.2.

⁵¹Das Konzept des betrieblichen Objekts wird in Abschnitt 3.1.4 eingeführt.

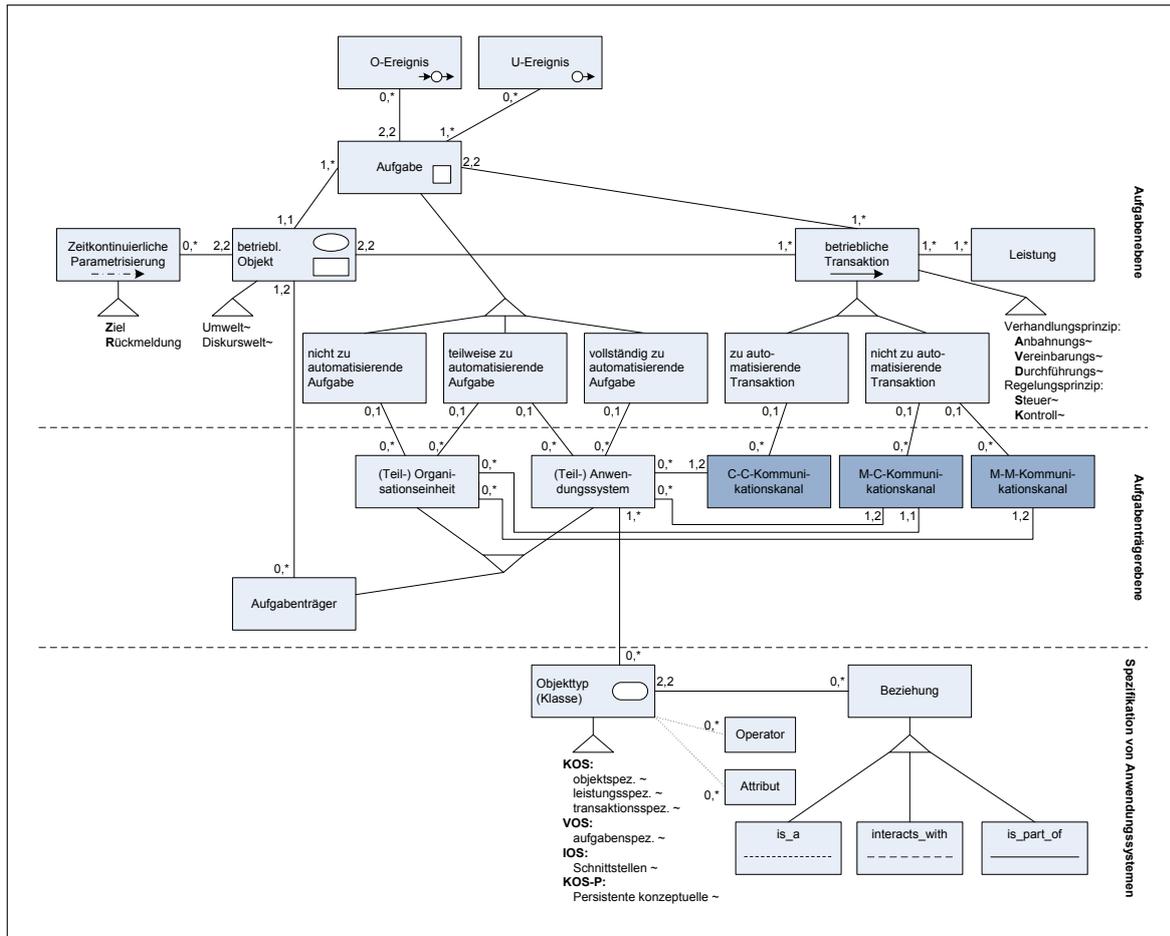


Abbildung 6.3: Metamodell der Methode (in Anlehnung an Krumbiegel (1997, S. 132))

Die Unterscheidung von Aufgaben und betrieblichen Transaktionen hinsichtlich ihrer Automatisierbarkeit, Automatisierungsforderung sowie Realisierung wird in Abb. 6.3 ausführlich dargestellt. Zudem wird die Verbindung zwischen Aufgabe und AT verdeutlicht. So werden *nicht zu automatisierenden Aufgaben* von dem personellen AT der zugeordneten Organisationseinheit durchgeführt. *Vollständig zu automatisierende Aufgaben* werden die entsprechenden AwS als maschinelle AT zugeordnet. Den *teilweise zu automatisierenden Aufgaben* werden sowohl eine Organisationseinheit als auch ein AwS zugeordnet, da Aufgaben dieses Typs in Kooperation beider AT-Arten durchgeführt werden. Daneben werden durch die Einführung der Kommunikationskanäle⁵² die Transaktionen mit den jeweils ausführenden AT verknüpft. Dabei werden *zu automatisierende Transaktionen* von einem oder zwei AwS durch C-C-Kommunikationskanäle unterstützt. *Nicht zu automatisierende Transaktionen* können entweder durch M-M- oder M-C-Kommunikationskanäle realisiert werden. Bei M-M-Kommunikationskanälen werden ein oder zwei Organisationseinheiten zugeordnet.

⁵²Kommunikationskanäle werden in Abschnitt 2.3 definiert.

Bei M-C-Kommunikationskanälen sind zwei Situationen zu unterscheiden. Entweder benötigt eine Organisationseinheit Zugriff auf ein weiteres, erforderliches AwS oder die Organisationseinheit realisiert die betriebliche Transaktion zwischen zwei betrieblichen Objekten, denen das gleiche AwS zugeordnet wurde, mittels eines M-C-Kommunikationskanals.

Im unteren Teil der Abb. 6.3 wird der AT AwS mit dem Metamodell für die Spezifikation von AwS verknüpft (Ferstl und Sinz 2013, S. 233-234). Für die ausführliche Erläuterung des Metamodells zur Spezifikation von AwS wird auf die Beschreibung in Ferstl und Sinz (2013, S. 233-234) verwiesen.

6.2 Architekturmodell der Methode

Nachfolgend werden die Metamodelle der Sichten vorgestellt und auf die Unterstützung der jeweiligen Sichten bei der Entwicklung von Integrationsarchitekturen Bezug genommen (Abschnitt 3.4.2; Abb. 3.7). Zur Beschreibung des Architekturmodells wird der generische Architekturrahmen, wie er in Abschnitt 2.1 eingeführt wurde, herangezogen (Abb. 2.1).

In Tab. 6.1 wird ein Überblick über das Architekturmodell gegeben. Dazu werden das Repräsentationsobjekt mit der Bezeichnung der zugehörigen Sichten sowie den Verweisen, in denen die Metamodelle der jeweiligen Sicht definiert werden, zusammengefasst. Die Darstellung separater Metamodelle für die Beziehungen ist nicht erforderlich, da die Sichten einen Teilausschnitt des in Abschnitt 6.1 vorgestellten Metamodells darstellen. Folglich können aus Abb. 6.3 auch die Beziehungen der Sichten untereinander entnommen werden.

Tabelle 6.1: Überblick über das Architekturmodell

Repräsentationsobjekt	Sichten	
Geschäftsprozess	Interaktions- und Verhaltenssicht	Abschnitt 6.2.1, Abb. 6.4
Aufgabenträgerzuordnung	Aufgabenträgerzuordnungssicht	Abschnitt 6.2.2, Abb. 6.5

6.2.1 Metamodell für das Repräsentationsobjekt Geschäftsprozess

Das Metamodell der SOM-Methodik für Geschäftsprozessmodelle Ferstl und Sinz (2013, S. 218-219) bildet die Grundlage für das Metamodell der Geschäftsprozesssicht, dargestellt in Abb. 6.4. Wie in Abschnitt 6.1 geschildert, wird auf eine Erläuterung

des Metamodells der SOM-Methodik für Geschäftsprozessmodelle verzichtet und stattdessen auf die Beschreibung in Ferstl und Sinz (2013, S. 218-219) verwiesen.

Mit dem in Abb. 6.4 dargestellten Metamodell für die Geschäftsprozessseite werden in der Struktursicht die Interaktionen und in der Verhaltenssicht der chronologische Ablauf der Interaktionen modelliert. Diese Sicht adressiert die fachliche Ebene zur Erstellung eines Modells für das betriebliche Objektsystem.

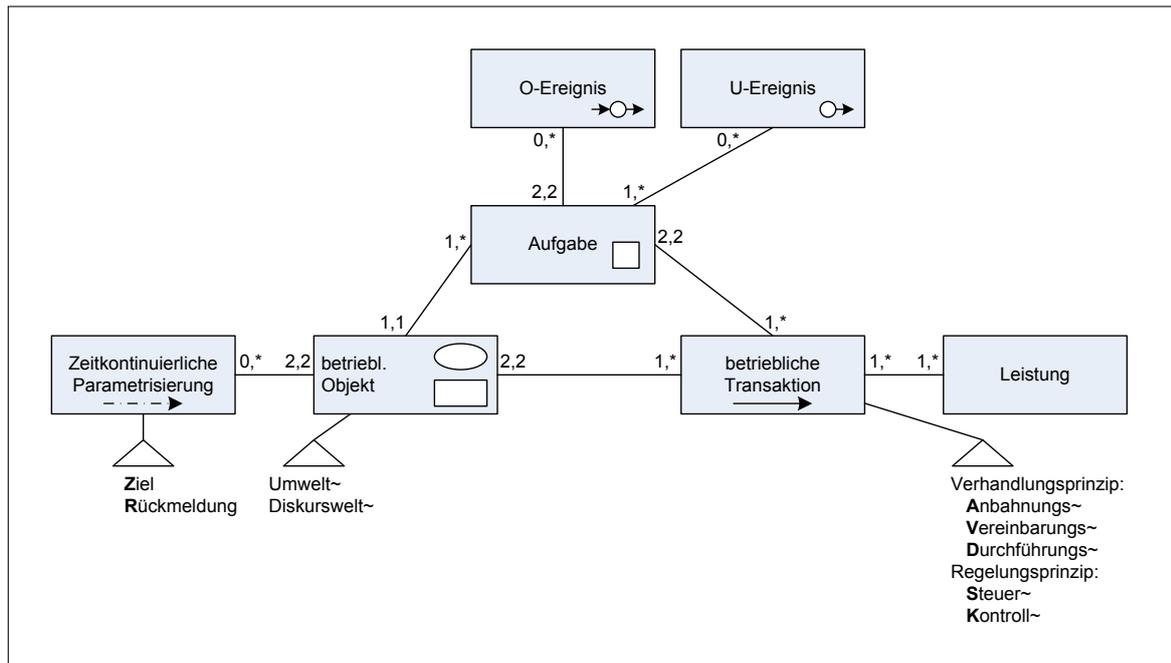


Abbildung 6.4: Metamodell der Geschäftsprozessseite (Ferstl und Sinz 2013, S. 219)

6.2.2 Metamodell für das Repräsentationsobjekt Aufgabenträgerzuordnung

In Abb. 6.5 wird das Metamodell für die Aufgabenträgerzuordnungssicht dargestellt. Sie vereint die von Krumbiegel vorgeschlagene AwS-Zuordnungssicht (Krumbiegel 1997, S. 137-139) sowie die Zuordnungssicht auf Organisationseinheiten (Krumbiegel 1997, S. 141-143). Die Sicht adressiert die fachliche Ebene zur Erstellung eines Modells für das betriebliche Objektsystem. In der Sicht werden im modellierten GP den betrieblichen Objekten die AT zugeordnet. Bei den Transaktionen werden sowohl der gegenwärtige Automatisierungsgrad als auch das mögliche Automatisierungspotenzial erhoben und verzeichnet. Dazu wird aus der Symbolik von Abb. 6.6 der Bereich zur Kennzeichnung der Automatisierbarkeit und Automatisierung von Transaktionen herangezogen. Bei dieser werden für Transaktionen jeweils der mögliche Automatisierungsgrad bei der Automatisierbarkeit sowie die gegenwärtige Automatisierung mittels entsprechender Symbole kombiniert.

Transaktionen werden mittels Kommunikationskanälen realisiert. Dazu werden im Metamodell von Abb. 6.5 die Transaktionen mit den entsprechenden Kommunikationskanälen verbunden. Diese sind wiederum mit den AT verknüpft, da die AT zur Kommunikation untereinander Kommunikationskanäle nutzen. Die Ausprägung der AT wird über den angestrebten Automatisierungsgrad der Aufgabe mit selbiger verknüpft. Darüber hinaus kann die Zuordnung der Aufgaben zu den AT über die Beziehung zwischen betrieblichem Objekt und Aufgabe ermittelt werden.

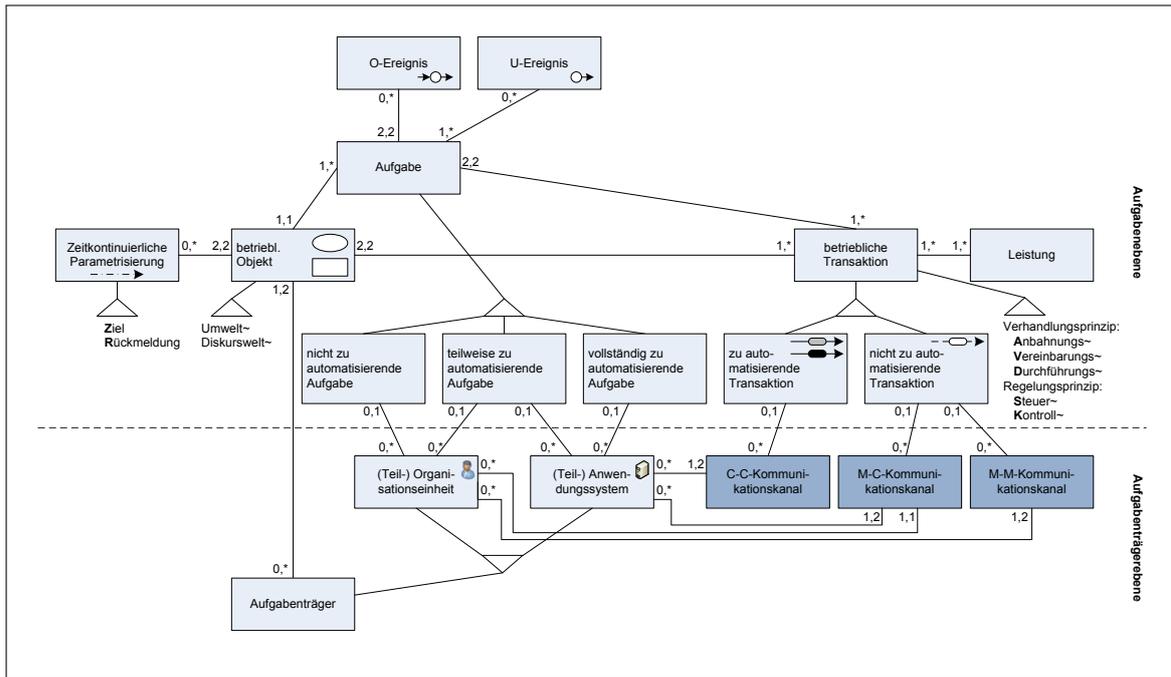


Abbildung 6.5: Metamodell der Aufgabenträgerzuordnungssicht (in Anlehnung an Krumbiegel (1997, S. 137) sowie Krumbiegel (1997, S. 141))

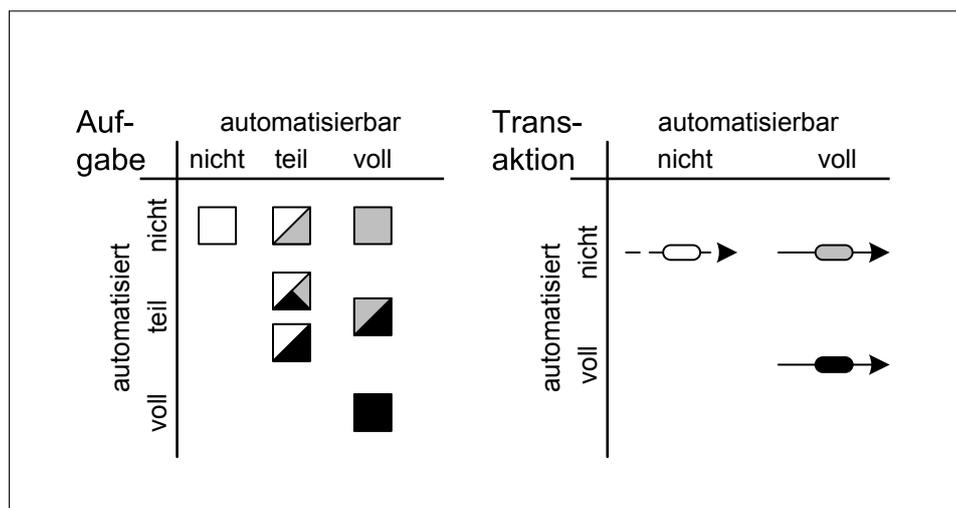


Abbildung 6.6: Symbole zur Automatisierbarkeit und Automatisierung von Aufgaben und Transaktionen (Ferstl und Sinz 2013, S. 221)

6.3 Vorgehensmodell der Methode

In diesem Abschnitt wird das Vorgehensmodell der Methode vorgestellt. Dieses wird zunächst in Abb. 6.7 schematisch dargestellt. Auf der linken Seite der Abb. 6.7 werden die Aktivitäten zu Phasen zusammengefasst, im mittleren Bereich werden die Aktivitäten und auf der rechten Seite die jeweils unterstützenden Elemente aufgeführt. In der nachfolgenden Beschreibung der einzelnen Aktivitäten werden die Zusammenhänge mit den Beschreibungsebenen bei der Entwicklung von Integrationsarchitekturen (Abschnitt 3.4.2 bzw. Abb. 3.7) aufgezeigt.

Die in Abb. 6.7 den Aktivitäten zugeordneten Schritte sind nicht streng in Reihenfolge aufzufassen, sondern dienen als Orientierungshilfe für eine idealtypische Durchführung. Einzelne Aktivitäten sind vielmehr simultan durchzuführen, insbesondere können die Ableitung des Sollkonzepts sowie die Festlegung der Integrationskonzepte eine enge Abstimmung mit dem GP erfordern und durch wechselseitige Beeinflussung zu Modifikationen führen.

Ergebnis des Vorgehens ist zum einen die Integrationsarchitektur für das Informationssystem eines Szenarios, auf Grundlage dessen Kopplungssysteme, sofern AwS beteiligt sind, entworfen werden können. Zum anderen können die unterschiedlichen Szenarien sowie deren mit der Implementierung verbundenen Auswirkungen miteinander verglichen werden und auf dieser Grundlage eine Entscheidung für die Umsetzung eines Szenarios getroffen werden. Die einzelnen Schritte werden in den nachfolgenden Abschnitten detailliert beschrieben und anhand eines durchgängigen Fallbeispiels erläutert. Die Ergebnisse der Istanalyse und die Auswirkungen der entworfenen Szenarien können abschließend im Rahmen einer What-If-Analyse miteinander verglichen werden.

6.3.1 Istanalyse

6.3.1.1 Spezifikation des zu analysierenden Geschäftsprozesses

Ausgangspunkt für den ersten Schritt der Methodenanwendung ist ein vorliegender GP.⁵³ Der zu untersuchende GP ist zunächst zu spezifizieren, um eine Basis für die darauf aufbauenden Analysen zu schaffen. Als Detaillierungsgrad ist die Offenlegung aller Transaktionen zwischen der Diskurswelt und dem Umweltobjekt zu wählen; d. h. aus Sicht des Umweltobjekts sind alle konkreten Transaktionen mit der Diskurswelt im zu untersuchenden Ausschnitt aus dem GP darzustellen – keine aggregierten

⁵³Das Re-Engineering von GP ist nicht Gegenstand dieser Arbeit. Zur Vertiefung wird stattdessen auf den Beitrag von Ferstl und Sinz (1995b) verwiesen, in dem beispielhaft das Re-Engineering mithilfe der SOM-Methodik vorgestellt wird.

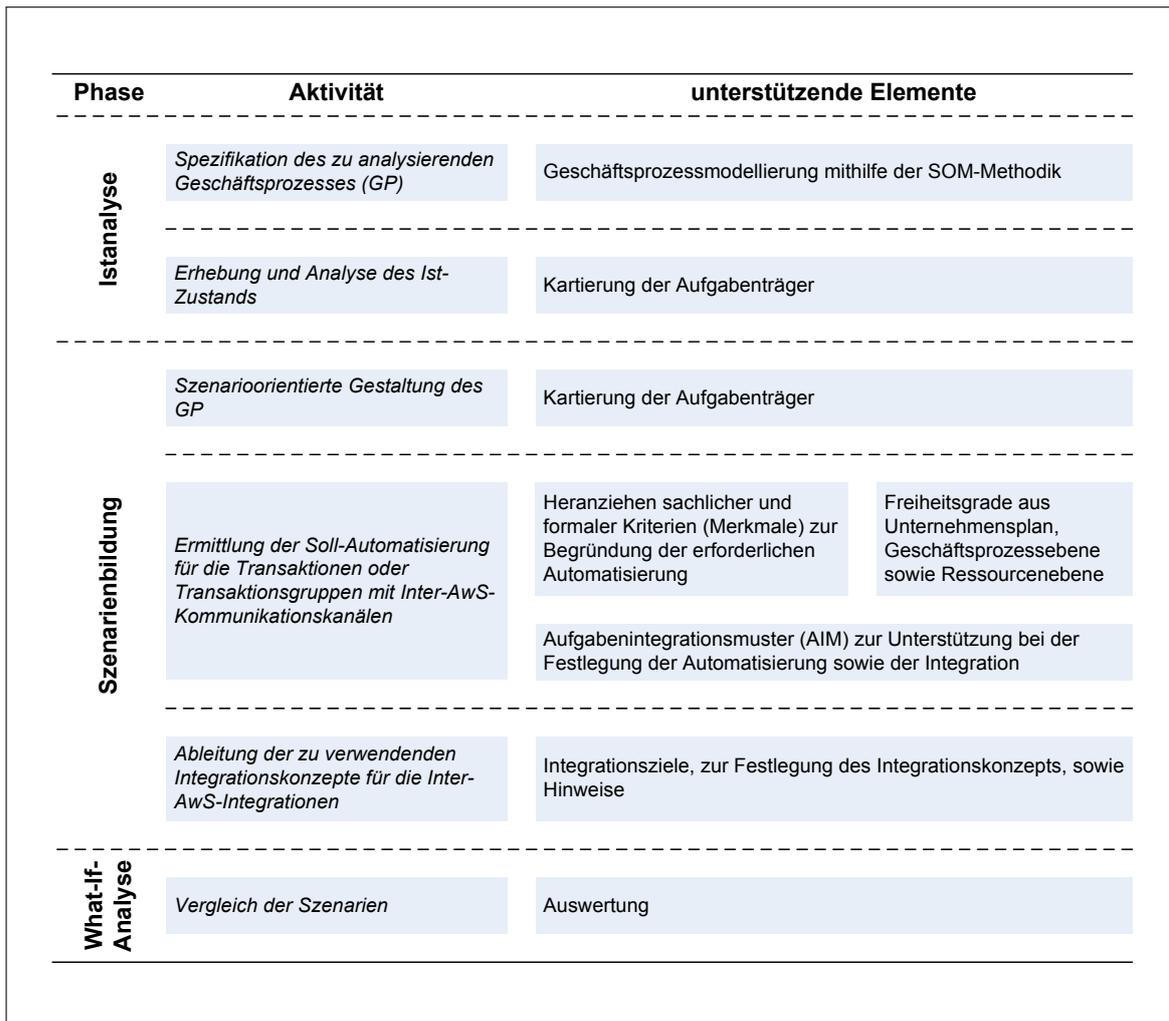


Abbildung 6.7: Schematische Darstellung des Vorgehensmodells der entwickelten Methode zum Entwurf von Integrationsarchitekturen für Informationssysteme

und keine generalisierten Transaktionen. Unterstützt wird diese Aktivität durch die Verwendung von Modellierungsmethoden, die eine grafische Darstellung ermöglichen. Zu diesem Zweck wird als Grundlage die SOM-Methodik verwendet, da diese zum einen zwischen Struktur (IAS) und Verhalten (VES) und zum anderen zwischen Aufgaben- und AT-Ebene trennt (Abschnitt 2.1). Das Metamodell zur Erstellung einer Geschäftsprozesssicht wird in Abschnitt 6.2.1 vorgestellt und in Abb. 6.4 dargestellt. Mit dieser Aktivität wird die fachliche Ebene zur Erstellung eines Modells für das betriebliche Objektsystem adressiert.

Durch die Verwendung der SOM-Methodik wird auch das Konzept des betrieblichen Objekts (Abschnitt 3.1.4) aufgenommen. Dieses liefert bereits erste Hinweise auf die zu verwendende Kopplung der Aufgaben. Aufgaben, die eng zu koppeln sind (hohe Kohäsion innerhalb eines betrieblichen Objekts) sind durch einen AT personell, maschi-

nell oder durch eine Zusammenarbeit beider Typen zu integrieren. Hingegen sind bei Aufgaben, die lose zu koppeln sind, die Beziehungen von mehreren unterschiedlichen AT untereinander durch entsprechende Kollaboration sicherzustellen.

Im Rahmen der Bestandsaufnahme des vorliegenden GP sind für die Transaktionen die überlappenden AO zu ermitteln sowie die Beziehungen zwischen den korrespondierenden Aufgaben zu analysieren. Dabei ist zu prüfen, welche der in Abschnitt 3.1.3 vorgestellten AIM auftreten. Die Ergebnisse werden in darauf aufbauenden Schritten herangezogen. Unterstützt wird die Bestandsaufnahme durch die in Tab. 6.2 vorgestellte Hilfstabelle. In der Tab. 6.2 sind die überlappenden AO sowie die identifizierten AIM in den jeweiligen Spalten aufzunehmen.

Beispiel zur Spezifikation des zu analysierenden Ausschnitts aus dem Geschäftsprozess

Zur Verdeutlichung dieses Schrittes wird das in Abschnitt 2.4 eingeführte Fallbeispiel aufgegriffen. Grundlage ist der in Abb. 2.6 markierte Bereich. Daraus wurden das Umweltobjekt *Studierender* und die ihm gegenüber zu erbringende Leistung übernommen. Diese sind Ausgangspunkt für das initiale IAS von Abb. 6.8.

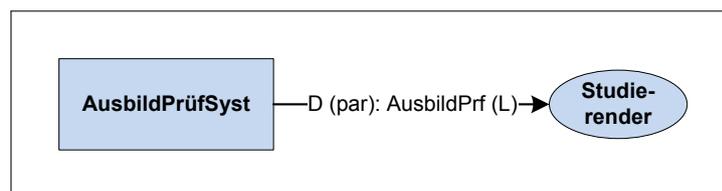


Abbildung 6.8: Initiales IAS für den untersuchten Teilprozess

Das initiale IAS von Abb. 6.8 wird verfeinert. Das korrespondierende Protokoll der Objekt- und Transaktionszerlegung kann aus Tab. B.1 entnommen werden. Das verfeinerte IAS ist in Abb. 6.9 abgebildet und dient zur Erläuterung des in Abb. 6.4 dargestellten sowie in Abschnitt 6.2.1 beschriebenen Metamodells für die Geschäftsprozessseite. In Tab. 6.2 werden die Transaktionen, die sendenden und empfangenden betrieblichen Objekte sowie die überlappenden AO und identifizierten AIM zur Fallstudie dokumentiert.

In Abb. 6.9 wird das betriebliche Objekt Ausbildungsprüfungssystem (*AusbildPrüfSyst*) nach dem Verhandlungsprinzip in die betrieblichen Objekte Semesterausbildungssystem (*SemAusbildSyst*) und Prüfungssystem (*PrüfungsSyst*) zerlegt. Aus Darstellungsgründen werden die beiden gebildeten Objekte untereinander anstatt auf gleicher Höhe angeordnet. Der in Abb. 6.9 dargestellte Ausschnitt der Lenkungssicht ist Teil der Struktursicht und entsteht durch die Modellierung eines GP mittels der SOM-Methodik.

Tabelle 6.2: Hilfstabelle zur Dokumentation der Transaktionen, Transaktionsgruppen, zugeordneter betrieblicher Objekte sowie identifizierte AIM und überlappende AO

Transaktion	Name der Transaktionsgr.	sendendes betr. Objekt	empfangendes betr. Objekt	überlappende AO	Identifizierte AIM
Name der Transaktion	Name der Transaktionsgr.	Name des sendenden betr. Objekts	Name des empfangenden betr. Objekts	AO	Reihenfolgebeziehung zwischen Aufgabendurchführungen / Partielle Gleichheit von AO-Typen / Partielle Identität von AO-Instanzen / Gemeinsame Nutzung von (Teil-) Lösungsverfahren / eine Kombination der AIM
<i>V: AuffLvAngeb</i>		<i>SemAusbild-Syst</i>	<i>PrüfungSys</i>		<i>Reihenfolgebeziehung zwischen Aufgabendurchführungen</i>
<i>D: MeldLvAngeb</i>		<i>PrüfungSys</i>	<i>SemAusbild-Syst</i>	<i>Lehrveranstaltungen</i>	<i>Reihenfolgebeziehung zwischen Aufgabendurchführungen; Partielle Identität von AO-Instanzen</i>
...

Anhand der Abb. 6.9 ist zu erkennen, dass das *SemAusbildSys* dem *Studierenden* die Lehrveranstaltungsangebote anbietet (A: LV-Angebot), dieser sich ggf. zu diesen anmelden muss (V: *Anmeldung Lv) und während des Semesters die Lehrinhalte bspw. im Rahmen von Lehrveranstaltungen vermittelt werden (D (seq): *Semesterausbildung*). Das *PrüfungSys* überprüft, ob der *Studierende* die vermittelten Lehrinhalte verstanden hat. Dazu erstellt es Prüfungsangebote (A: Prf-Angebot), zu denen sich der *Studierende* anmelden kann (V: *Prüfungsanmeldung*). Abschließend hat der *Studierende* einen Leistungsnachweis zu erbringen (D (seq): *Prüfung*). Bei dem untersuchten GP gestaltet es sich gegenwärtig so, dass das *SemAusbildSys* das *PrüfungSys* auffordert, die angebotenen Lehrveranstaltungen mitzuteilen (V: *AuffLvAngeb*). Daraufhin erstellt das *PrüfungSys* die Lehrveranstaltungsangebote und meldet diese dem *SemAusbildSys* (D: *MeldLvAngeb*). Der gegenwärtige GP ist historisch gewachsen.

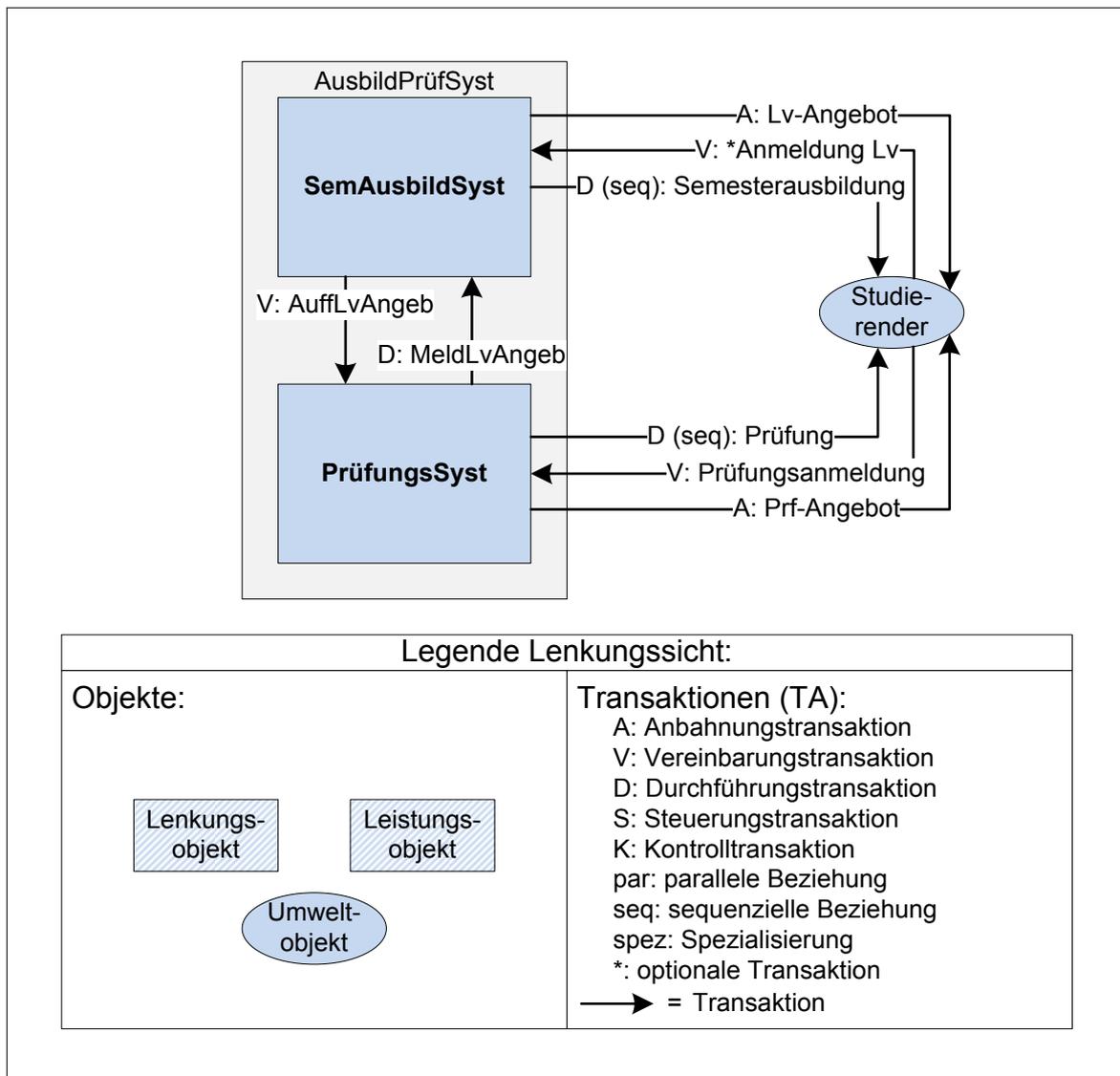


Abbildung 6.9: Instanzierte Struktursicht auf den Ist-GP des Fallbeispiels

6.3.1.2 Erhebung und Analyse des Ist-Zustands

Im Rahmen dieser Aktivität erfolgt die Zuordnung der AT zu den betrieblichen Objekten, die Identifikation der AwS-Grenzen, die Gruppierung der Transaktionen mit Inter-AwS-Kommunikationskanälen, die Erfassung der Transaktionsautomatisierung sowie die Ermittlung der verwendeten Integrationskonzepte. Nachfolgend werden die Teilaktivitäten einzeln beschrieben.

Zuordnung der Aufgabenträger zu den betrieblichen Objekten

In diesem Schritt sind den betrieblichen Objekten sowohl die ausführenden personellen AT in Form von Stellen⁵⁴ bzw. Organisationseinheiten als auch die maschinellen AT in Form von AwS zuzuordnen. Um die Dokumentation bei der Erstellung der Zuordnung zu unterstützen, wird die Verwendung der Tab. 6.3 vorgeschlagen. Im Rahmen der Zuordnung der AT zu den betrieblichen Objekten ist die Regel zu berücksichtigen, dass einem betrieblichen Objekt maximal ein AwS zugeordnet werden darf (Krumbiegel 1997, S. 138). Wird an dieser Stelle erkennbar, dass die Regel verletzt wird, ist der GP solange zu verfeinern (Abschnitt 6.3.1.1), bis diese Regel eingehalten werden kann. Umgekehrt darf ein AwS mehreren betrieblichen Objekten zugeordnet werden.

Die Tab. 6.3 enthält in *kursiver* Formatierung zwei erläuternde Beispieldatensätze. Das Symbol für personelle AT in der Tab. 6.3 und allen folgenden Abbildungen und Tabellen soll auf Angehörige einer zusammenhängenden Organisationseinheit hinweisen. Hinter dem jeweiligen Symbol steht die Rolle⁵⁵, in der der entsprechende personelle AT aktiv wird. Dabei ist es möglich, dass die Aufgaben eines betrieblichen Objekts von einer personellen Ressource oder arbeitsteilig von mehreren personellen Ressourcen durchgeführt werden. Die Bestimmung der personellen Ressource, welche die Aufgaben des jeweiligen betrieblichen Objekts wahrnimmt, obliegt dem Weisungsbefugten der Organisationseinheit, der neben dem potenziell möglichen Arbeitspensum etwaige Vorgaben, wie Zugriffsrechte, berücksichtigen muss.

In der Tab. 6.3 sind die AT-Arten Mensch und Maschine als gleichwertig aufzufassen. Diese Auffassung ist einer der Spezifika der SOM-Methodik.

Identifikation der Anwendungssystem-Grenzen

Mit den Informationen aus Tab. 6.3 können die AwS-Grenzen im GP verzeichnet werden. Hierfür bietet sich die Struktursicht auf den GP an. In dieser sind die jeweils von einem AwS unterstützten betrieblichen Objekte zu kennzeichnen und durch einen

⁵⁴Der Begriff der Stelle wird in Abschnitt 3.1.1 eingeführt.

⁵⁵Der Begriff der Rolle und dessen unterschiedliche Auslegungen *verhaltensorientiert*, *organisationsorientiert*, *aufgabenorientiert*, *kompetenzorientiert* sowie *berechtigungsorientiert* werden in Walther (2005, S. 6-15) erläutert. G. Fischer (2015, S. 156-160) arbeitet den Begriff der Rolle mit Fokus auf die Zugriffskontrolle in das betriebliche Informationssystem ein.

Tabelle 6.3: Hilfstabelle zur Zuordnung der AT zu den betrieblichen Objekten

Betriebliches Objekt	ausführende Organisationseinheit	unterstützendes AwS
Name des betrieblichen Objekts	Symbol und Name der ausführenden Organisationseinheit	Symbol und Name für das unterstützende AwS
<i>SemAusbildSyst</i>	 <i>Lehrstuhl</i>	 <i>UnivIS</i>
<i>PrüfungsSyst</i>	 <i>Lehrstuhl</i>	 <i>FlexNow</i>
...

zusammenhängenden Layer hervorzuheben. Dazu kann die Aufgabenträgerzuordnungssicht verwendet werden. Das Metamodell zu dieser Sicht wird in Abschnitt 6.2.2 vorgestellt und in Abb. 6.5 dargestellt. In Abb. 6.10 wird dieser Schritt beispielhaft ausgeführt. Dabei werden sowohl die betrieblichen Objekte *SemAusbildSyst* als auch *PrüfungsSyst* durch unterschiedliche Layer eingegrenzt, da sie durch sich unterscheidende AwS unterstützt werden. Zwischen diesen beiden betrieblichen Objekten bestehen Inter-AwS-Kommunikationskanäle für Transaktionen.

Gruppierung von Transaktionen mit Inter-AwS-Kommunikationskanälen

Im Rahmen dieser Aktivität sind Transaktionen mit Inter-AwS-Kommunikationskanälen zu gruppieren. Hinweise zur Gruppierung von Transaktionen werden im separaten Abschnitt 6.3.1.3 gegeben, um ihn im weiteren Verlauf der Arbeit gezielt referenzieren zu können. Im Rahmen des Fallbeispiels konnten keine Transaktionen gruppiert werden, entsprechend bleiben die Einträge in der Tab. 6.2 in der Spalte Name der Transaktionsgruppe (*Name der Transaktionsgr.*) leer.

Erfassung der Transaktionsautomatisierung

Neben der Kartierung der AT im IAS sind die gegenwärtigen Automatisierungsgrade der Transaktionen mit Inter-AwS-Kommunikationskanälen zu verzeichnen. Zur Erfassung der Automatisierung können entweder die Ergebnisse aus einer früheren Untersuchung herangezogen werden oder sie wird an dieser Stelle erstmalig ermittelt.

Im Rahmen der erstmaligen Ermittlung ist zunächst für die betroffenen Transaktionen zu definieren, ob die Nutzdaten (Ereignisse oder AO-Instanzen) rechnergestützt übertragen werden können (Ferstl und Sinz 2013, S. 220). Hierfür ist die funktionale Beschreibbarkeit des Lösungsverfahrens als formales Kriterium ausschlaggebend. Kann dieses nicht formal beschrieben werden, ist auch der entsprechende Inter-AwS-Kommunikationskanal nicht automatisierbar und so zu kennzeichnen (Ferstl und Sinz

2013, S. 115-117). Durch diesen Detaillierungsgrad kann binär entschieden werden, ob eine Transaktion automatisierbar ist. Anschließend ist die Ist-Automatisierung für Transaktionen zu ermitteln und entsprechend der Symbolik aus Abb. 6.6 im IAS zu verzeichnen. Eine darüber hinausgehende Bewertung, ob Automatisierungspotenziale auch tatsächlich durch Automatisierung der Transaktionen herbeizuführen sind, ist nicht Gegenstand der Istanalyse. Sollte dies dennoch in Betracht gezogen werden, ist das Ergebnis dieser Aktivität als Ausgangspunkt für ein darauf aufbauendes Szenario heranzuziehen.

Zur Erfassung der Ergebnisse wird die Nutzung der Tab. 6.4 vorgeschlagen. In dieser werden in jeder Zeile zu den Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen sowohl das Automatisierungspotenzial als auch die Ist-Automatisierung, wie beschrieben, erfasst. Die exemplarische Übertragung der Ergebnisse der Tab. 6.4 in das IAS wird für das Fallbeispiel in Abb. 6.10 durchgeführt.

Tabelle 6.4: Hilfstabelle zur Dokumentation der Automatisierungsgrade von Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen

Transaktion oder Transaktionsgruppe	Automatisierungspotenzial	Ist-Automatisierung
Name der Transaktion oder Transaktionsgruppe	--> / →	 /  / 
<i>V: AuffLvAngeb</i>	-->	
<i>D: MeldLvAngeb</i>	→	
...

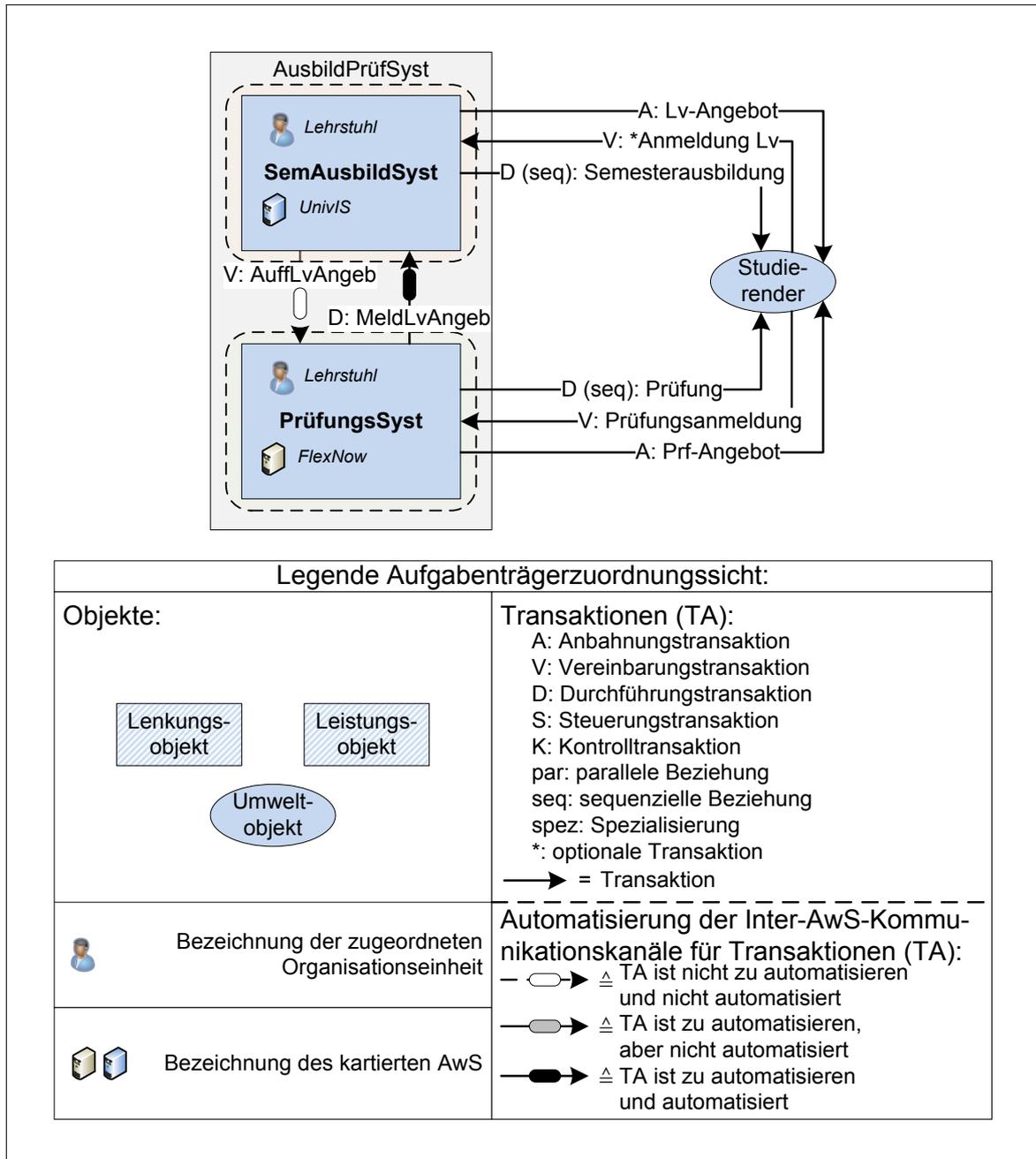
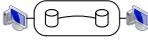


Abbildung 6.10: Instanziiertes Modell für die Aufgabenträgerzuordnungssicht

Ermittlung verwendeter Integrationskonzepte

Im Rahmen dieser Aktivität sind die gegenwärtig verwendeten Integrationskonzepte aus Abschnitt 3.3.2 zu ermitteln. Zur Dokumentation wird die Tab. 6.5 vorgeschlagen. Darin wird jeweils ein Paar bestehend aus dem sendenden sowie empfangenden betrieblichen Objekt in zwei Zeilen untereinander erfasst und das verwendete Integrationskonzept eingetragen. In der Tab. 6.5 sind die in *kursiv* gesetzten Datensätze erläuternde Beispieldatensätze zum vorliegenden Fallbeispiel. Aus der Tab. 6.5 wird ersichtlich, dass zwischen den betrieblichen Objekten *SemAusbildSyst* und *PrüfungsSyst* das Integrationskonzept *datenflussorientierte Funktionsintegration* (Abschnitt 3.3.2.2) verwendet wird.

Tabelle 6.5: Hilfstabelle zur Dokumentation der Inter-AwS-Integrationen und die dabei verwendeten Integrationskonzepte

Betriebliches Objekt	AwS	verwendetes Integrationskonzept
Name des betrieblichen Objekts	Symbol und Name für das unterstützende AwS	Symbol und Name für das verwendete Integrationskonzept
<i>SemAusbildSyst</i>	 <i>UnivIS</i>	 <i>datenflussorientierte Funktionsintegration</i>
<i>PrüfungsSyst</i>	 <i>FlexNow</i>	
...
...

6.3.1.3 Hinweise für die Gruppierung von Transaktionen mit Inter-AwS-Kommunikationskanälen

Nachfolgend werden sowohl argumentativ-deduktiv hergeleitete Hinweise zur Gruppierung von Transaktionen mit Inter-AwS-Kommunikationskanälen als auch die sich für diese daraus ergebenden Konsequenzen gegeben.

Mit der Gruppierung von Transaktionen wird das Ziel verfolgt, Nachrichtentypen und Schnittstellen für Transaktionen mit Inter-AwS-Kommunikationskanälen zu identifizieren, die zur Unterstützung mehrerer solcher Transaktionen herangezogen werden können. Hinweise darauf, wie mit einem Nachrichtentyp oder einer Schnittstelle mehrere Transaktionen mit Inter-AwS-Kommunikationskanälen abgedeckt werden können, geben bereits die Transaktionen selbst. Hierzu sind vor allem die zu übertragenden AO parallel (*par*) ablaufender Transaktionen mit Inter-AwS-Kommunikationskanälen miteinander zu vergleichen. Zum einen können bei diesen die Nachrichtentypen

zu einem Nachrichtentyp zusammengefasst werden, wenn es sich um parallel ablaufende Transaktionen zwischen einem betrieblichen Objekt sowie einem anderen betrieblichen Objekt handelt und die übertragenen AO überlappend sind. Zudem ist für solch gruppierte Transaktionen lediglich eine Schnittstelle zwischen den betroffenen AwS zu implementieren. In der Folge kann dadurch auch der Implementierungsaufwand verringert werden. Sind zum anderen die von einem betrieblichen Objekt zu unterschiedlichen betrieblichen Objekten übertragenen AO überlappend, können sie ebenfalls zu einem Nachrichtentyp zusammengefasst werden. Für die verdichteten Transaktionen mit Inter-AwS-Kommunikationskanälen wird jeweils ein gemeinsames Integrationskonzept gewählt.

Daneben können Transaktionen, die mittels einer Schnittstelle auf Ressourcenebene abgedeckt werden können, zu einer Gruppe zusammengefasst werden. Außerdem sollten bei allen Transaktionen einer Gruppe die AIM identisch ausgeprägt sein.

Konsequenterweise können gruppierte Transaktionen mittels einer zu implementierenden Schnittstelle auf Ressourcenebene abgedeckt werden. Darüber hinaus sind die Merkmale bei allen Transaktionen einer Gruppe gleich auszuprägen. Dies gilt insbesondere bei der Ermittlung der Soll-Automatisierung im Rahmen der Bildung von Szenarien (Abschnitt 6.3.2.3).

6.3.2 Szenarienbildung

Mithilfe von Szenarien sollen mögliche Auswirkungen bei Modifikation vorhandener GP skizziert werden. Ausgangspunkt der zu bildenden Szenarien sind die Ergebnisse der Istanalyse (Abschnitt 6.3.1). Im Rahmen dieser Phase können beliebig viele Szenarien aufgebaut werden, so dass die Effekte unterschiedlicher Modifikationen evaluiert werden können.

Zunächst kann für die unterschiedlichen Szenarien, wie eingangs geschildert, der GP aus Abschnitt 6.3.1.1 als Basis herangezogen werden. Gegebenenfalls ist dieser jeweils für die unterschiedlichen Szenarien anzupassen. Hierfür gelten die gleichen Ausführungen wie aus Abschnitt 6.3.1.1. Darauf aufbauend sind in Anlehnung an Abschnitt 6.3.1.2 in Abschnitt 6.3.2.1 die AIM zu identifizieren, AT den betrieblichen Objekten zuzuordnen, AwS-Grenzen festzustellen sowie ggf. die Transaktionen mit Inter-AwS-Kommunikationskanälen zu gruppieren. Bei diesem darauf aufbauenden Schritt kann der GP des Szenarios modifiziert werden, bspw. wenn die Zuordnung der AT Änderungen auf den GP induzieren. Folglich sind die Ausführungen mit denen die erste Aktivität bei der Phase der Szenarienbildung von Abb. 6.7 abgedeckt werden nicht streng als Sequenz aufzufassen, sondern vielmehr als simultanes Vorgehen.

Für die daran anschließende Ermittlung der Soll-Automatisierung für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen sind die Freiheitsgrade beim Entwurf der Integrationsarchitektur (Abschnitt 6.3.2.2) zu erheben, ein Sollkonzept für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen (Abschnitt 6.3.2.3) zu erstellen sowie die zu verwendenden Integrationskonzepte für die Inter-AwS-Integrationen (Abschnitt 6.3.2.4) abzuleiten.

6.3.2.1 Szenarioorientierte Gestaltung des Geschäftsprozesses

In Anlehnung an Abschnitt 6.3.1.2 werden im Rahmen dieser Aktivität die AT den betrieblichen Objekten zugeordnet, die AwS-Grenzen identifiziert sowie ggf. der GP modifiziert (Tab. 6.6 und Abb. 6.11) und Transaktionen mit Inter-AwS-Kommunikationskanälen gruppiert. Zu diesem Zweck wird in Abb. 6.11 eine Teilmenge aus dem Metamodell der Aufgabenträgerzuordnungssicht genutzt.

Tabelle 6.6: Betriebliche Objekte und die ihnen zugeordneten AT im Szenario des Fallbeispiels

Betriebliches Objekt	ausführende Organisationseinheit	unterstützendes AwS
<i>SemAusbildSyst</i>	 <i>Lehrstuhl</i>	 <i>Virtueller Campus</i>
<i>PrüfungSyst</i>	 <i>Prüfungsamt</i>	 <i>FlexNow</i>

Durch die Berücksichtigung der personellen AT kann auch die Aufgabenintegration über diese als potenzielle Lösung bei der Sollkonzeption für die Szenarien berücksichtigt werden. Dies ist insbesondere in den Situationen von Interesse, in denen zwei betriebliche Objekte zwar von unterschiedlichen AwS, aber vom gleichen personellen AT unterstützt werden. In diesen Fällen ist die Aufgabenintegration über personelle AT als probates Mittel einzubeziehen. Darüber hinaus kann bei der Feststellung, dass zwei betriebliche Objekte durch unterschiedliche personelle AT unterstützt werden, eine Zerlegung des betrieblichen Objekts erwogen werden, so dass einem personellen AT Zugriff auf das jeweils benötigte AwS gewährt wird.

Indem die AwS-Grenzen verzeichnet werden, kann die Identifikation von Inter-AwS-Kommunikationskanälen erleichtert werden, da im Rahmen der Identifikation der AwS-Grenzen die Transaktionen ausfindig zu machen sind, an denen unterschiedliche AwS involviert sind. Sie stellen die Inter-AwS-Kommunikationskanäle dar, für die eine geeignete Integrationsarchitektur zu entwerfen ist. Durch diese Vereinfachung soll die Skalierbarkeit der Methode erhöht werden, da lediglich Inter-AwS-Kommunikationskanäle fokussiert und gestaltet werden. Die Gestaltung von Intra-AwS-Kommunikationskanälen ist dem Bereich der Neuentwicklung oder Einführung

von AWS zur Unterstützung betrieblicher Objekte eines GP zuzuordnen. Sie wird im Rahmen dieser Arbeit nicht betrachtet.

Anhand der Abb. 6.11 ist zu erkennen, dass der GP im Rahmen dieses Fallbeispiels modifiziert wird. Diese Änderung wird aufgrund der geänderten AT-Zuordnung und der damit einhergehenden Änderung im GP erforderlich. Dem *SemAusbildSyst* wird ein anderes AWS zugeordnet. Gleichzeitig soll der GP dahingehend modifiziert werden, dass die im neu zugeordneten AWS verwalteten Lehrveranstaltungen zukünftig an das davon abhängige AWS, welches dem betrieblichen Objekt *PrüfungsSyst* zugeordnet wird, übertragen werden soll. Das dadurch erforderlich gewordene modifizierte Protokoll der Objekt- und Transaktionszerlegung kann aus der Tab. B.2 entnommen werden. In der Tab. 6.2 können für die erhalten gebliebene Transaktion *D: MeldLvAngeb* die zugeordneten betrieblichen Objekte sowie die identifizierten AIM und das überlappende AO nachgeschlagen werden.

6.3.2.2 Erhebung der Freiheitsgrade beim Entwurf der Integrationsarchitektur

In diesem Teilschritt sind die Rahmenbedingungen, die die Restriktionen für die Gestaltung vorgeben, zu erheben. Dadurch sollen zum einen die Festlegung der Soll-Automatisierung für die Transaktionen oder Transaktionsgruppen mit Inter-AWS-Kommunikationskanälen unterstützt und zum anderen die Freiheitsgrade für die Auswahl der zur Verfügung stehenden Integrationskonzepte für einen darauf aufbauenden Schritt ausfindig gemacht werden. Zur Systematisierung werden nachfolgend beschriebene mögliche Einflussfaktoren in die Ebenen der SOM-Unternehmensarchitektur (Abschnitt 2.1, Abb. 2.2) eingeordnet. Dokumentiert werden können die Ausprägungen auf den jeweiligen Ebenen in der in Tab. 6.7 vorgestellten unterstützenden Hilfstabelle. Die Tab. 6.7 enthält in *kursiver* Formatierung bereits Datensätze aus dem erläuternden Beispiel.

Ebene des Unternehmensplans

Rahmenbedingungen auf Ebene des Unternehmensplans sind Vorgaben sowohl hinsichtlich der Adoption und dem Einsatz von Technologien als auch in Bezug auf die einzusetzenden AWS sowie Strategien zur Umsetzung der Ziele.

Bei der Strategie hinsichtlich der Adoption und dem Einsatz von Technologien ist festzulegen, wie das Verhalten beim Gebrauch neuer Technologien ist. Dabei ist zu definieren, ob jede neu am Markt verfügbare Technologie eingesetzt oder ob nur eine bewährte, im Unternehmen bereits eingesetzte Technologie verwendet werden darf. Derartige Prämissen schränken die Freiheitsgrade hinsichtlich der Verwendung von Technologien als Lösungsverfahren und damit die Auswahl der Verfügung stehenden Basismaschinen ein. Falls bspw. gegenwärtig im Unternehmen noch keine Bus-Archi-

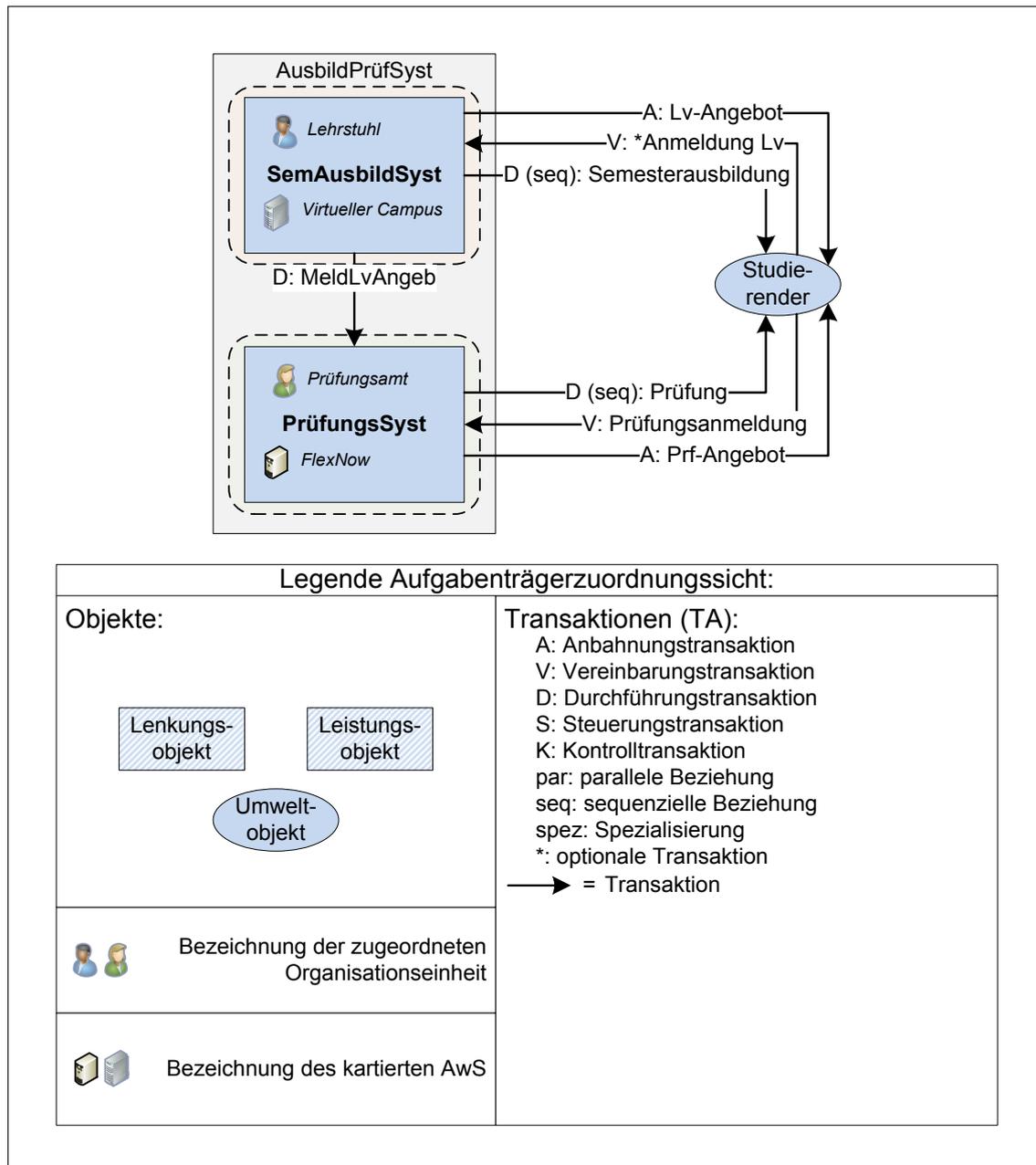


Abbildung 6.11: Aufgabenträgerzuordnungssicht auf den GP des Szenarios

tektur (Ferstl und Sinz 2013, S. 262-264) zur Verknüpfung von AwS eingesetzt wird, sondern sie direkt gekoppelt werden und die Vorgabe lautet, gegenwärtig keine Bus-Architektur einzuführen, dann stehen bei der Kopplung von AwS lediglich technologische Lösungsverfahren zur Verfügung, die im Unternehmen bereits etabliert sind und eine direkte Kopplung gestatten.

Die Vorgaben bezüglich einzusetzender AwS wirken sich ebenfalls auf die zur Auswahl stehenden Basismaschinen und damit auf die möglichen Technologien aus (Balzert 2011, S. 137). Die Anwendungssoftware als der Teil des AwS, mit dem die fachlichen

Anforderungen in der Nutzermaschine umgesetzt werden, wird auf einer bestimmten Programmierplattform realisiert. Je nach Portabilität der Anwendungssoftware kann sie auch auf unterschiedlichen Programmierplattformen realisiert werden. Die Programmierplattform kann dabei als Basismaschine interpretiert werden (vgl. Abschnitt 2.3). Aus diesem Grund stehen als Lösungsverfahren für die Kopplung nur die Technologien zur Verfügung, die von den Programmierplattformen oder einer ihrer Erweiterungen unterstützt werden.

Strategien zur Umsetzung der Unternehmensziele werden im Rahmen von Projekten implementiert. Aus diesem Grund sind beim Entwurf von Integrationsarchitekturen gegenwärtig laufende sowie bereits geplante Projekte zu berücksichtigen, um mögliche Skalen- und Synergieeffekte bei der Umsetzung zu erzielen. Dazu sind Methoden des Multiprojektmanagements, dessen Gegenstand die Auswahl, Priorisierung sowie Koordination mehrerer Projekte ist, heranzuziehen. Daneben sind auch Kostenvorgaben zu berücksichtigen (Balzert 2011, S. 140).

Zudem ist auf Einhaltung von Normen als auch rechtlicher Rahmenbedingungen, wie Gesetzesvorgaben sowie Bestimmungen zu Datenschutz und Datensicherheit zu achten. Falls es Normen gibt, die die Unterstützung von in Beziehung stehenden Aufgaben durch unterschiedliche AwS vorgeben, scheiden Lösungen zur Integration mittels eines AwS aus. Ähnlich verhält es sich, wenn Gesetzesvorgaben oder Regelungen zum Datenschutz und zur Datensicherheit die Integration von Aufgaben über ein AwS untersagen, um bspw. vorsorglich den Benutzern den Zugang zu einer breiteren Datenbasis zu erschweren, insbesondere falls Zugriffsberechtigungen in den AwS bislang noch unzureichend implementiert sind. Dies gilt im gleichen Umfang auch für die als sensibel eingestuften Daten wie Konstruktionspläne.

Geschäftsprozessebene

Auf Aufgabenebene ist für die Inter-AwS-Kommunikationskanäle mithilfe des in Abschnitt 5.4.5 definierten finalen Merkmalskataloges und der in Abschnitt 5.5 vorgeschlagenen Handlungsempfehlungen zu prüfen, ob sie maschinell zu unterstützen sind. Ergänzend zur Tab. 6.7 besteht mit dem in Abb. 5.1 vorgestellten Ausschnitt aus dem Prototypen eine Möglichkeit zur Unterstützung bei der Anwendung der Merkmale.

Ressourcenebene

Rahmenbedingungen auf Ressourcenebene betreffen sowohl die Interoperabilität der betroffenen AwS als auch das vorhandene Know-how für die Realisierung der Integrationsarchitektur.

Fehlende Interoperabilität⁵⁶ der AwS wirkt einschränkend auf die Freiheitsgrade beim Entwurf der Integrationsarchitektur. Bei AwS, die zu koppeln sind, aber bislang lediglich disjunkte Kopplungsmechanismen⁵⁷ unterstützen, ist zu prüfen, ob sie durch Erweiterungen kopplungsfähig gemacht werden können und sich dadurch auf einen gemeinsam unterstützten Kopplungsmechanismus verständigt werden kann. Dabei sind auch die Vorgaben hinsichtlich der Adoption und dem Einsatz von Technologien, die bereits auf Ebene des Unternehmensplans getroffen wurden, zu berücksichtigen. Verfügt eines der zu koppelnden AwS nicht über einen Kopplungsmechanismus, dann ist zu prüfen, ob das AwS um die Fähigkeit, bspw. HTTP(S) zur Kommunikation zu nutzen, mittels einer Bibliothek erweitert werden kann, um so eine Möglichkeit zur Kommunikation mit anderen AwS anbieten zu können. Ist eine solche Erweiterung nicht möglich, dann scheidet die Kopplung der AwS zur Aufgabenintegration als Möglichkeit aus.

Das Vorhandensein von Know-how schränkt ebenfalls die Freiheitsgrade ein (Balzert 2011, S. 140). So können bspw. Integrationsarchitekturen nur mit den Technologien bzw. Basismaschinen realisiert werden, für die ausreichend Know-how zur Verfügung steht. Ausgeweitet werden kann dies intern mittels Fortbildungen oder extern, indem das Know-how über Dienstleister oder neue Mitarbeiter „akquiriert“ wird.

Ein weiterer Aspekt ist die Einfachheit bei der Realisierung der Integrationsarchitektur. Dabei ist insbesondere eine Prognose über den Aufwand für die Implementierung des maschinellen Lösungsverfahrens der identifizierten AIM abzugeben, sofern sie zu automatisieren sind. Die Prognose ist mit der geplanten Nutzungsdauer abzugleichen. Langwierig umzusetzende Lösungen sind für GP mit kurzer Einsatzdauer nicht zu empfehlen. Hingegen ist bei GP und Aufgaben, die stillgelegt oder nur temporär durchgeführt werden sollen, eine Kopplung oder Integration nur dann sinnvoll, sofern sie ohne großen Aufwand umzusetzen ist. Die Prognose über die Einfachheit bei der Realisierung der Integrationsarchitektur unter Berücksichtigung der Einsatzdauer wirkt restriktiv auf die Wahl der möglichen Aufgabenintegration über AT. Sie ist darüber hinaus unter anderem ein Eingangswert für die Kosten der Implementierung auf Ebene des Unternehmensplans. Zur Aufwandsschätzung kann bspw. eine der in Balzert (2009, S. 515-542) vorgestellten Methoden herangezogen werden. Zu den in Balzert vorgestellten Methoden zählen unter anderem die Schätzverfahren, die Function-Point-Methode, die Object Points / Application Points sowie COCOMO II.

Darüber hinaus ist der Kostenrahmen, sofern einer vorgegeben wird, zu berücksichtigen; d. h. die zu erwartenden Kosten der Implementierung dürfen den vorgegebenen Kostenrahmen nicht übersteigen. Sollte absehbar sein, dass dieser durch eine gewählte

⁵⁶Der Begriff *Interoperabilität* wurde in Abschnitt 3.1.6 eingeführt.

⁵⁷Der Begriff *Kopplungsmechanismus* wurde in Abschnitt 3.1.6 eingeführt.

Integrationsarchitektur dennoch überschritten wird, ist eine alternative Integrationsarchitektur zu wählen, mit der die Kostenvorgaben eingehalten werden können.

Zusammenfassung der freiheitsgradbeschränkenden Faktoren

In Tab. 6.8 werden nochmals alle Faktoren, die die Freiheitsgrade beim Entwurf der Architektur für die Integrationsarchitektur beschränken können, zusammengefasst und der jeweiligen Ebene zugeordnet.

6.3.2.3 Ermittlung der Soll-Automatisierung für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen

Ziel dieser Aktivität ist es, ein Sollkonzept für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen zu entwerfen und durch einen Abgleich mit der Soll-Automatisierung Deltas zu ermitteln. Durch die Fokussierung auf diese Art von Transaktionen soll die Skalierbarkeit des Ansatzes unterstützt werden. Als Metamodell wird das in Abschnitt 6.2.2 eingeführte und in Abb. 6.5 dargestellte Metamodell der Aufgabenträgerzuordnungssicht herangezogen. Die Aktivität adressiert ebenfalls die fachliche Ebene zur Erstellung eines Modells für das betriebliche Objektsystem.

Zur Unterstützung bei der Festlegung der Soll-Automatisierung für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen sind die AO, die AIM – beide sind für das Fallbeispiel für die Transaktion *D: MeldLvAngeb* dokumentiert in der Tab. 6.2 – sowie die Freiheitsgrade – diese können für das Fallbeispiel aus der Tab. 6.7 entnommen werden – heranzuziehen. Für die Transaktion *D: MeldLvAngeb* wird anhand der Handlungsempfehlung für die ausgeprägten Merkmale (Abschnitt 5.5) festgelegt, dass sie zu automatisieren ist. Das Ergebnis wird in der Spalte Soll-Automatisierung (*Soll-Automat.*) der Tab. 6.7 eingetragen.

Neben der Festlegung der Soll-Automatisierung ist für die jeweiligen Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen analog zur korrespondierenden Aktivität in Abschnitt 6.3.1.2 der aktuelle Grad der Automatisierung der Inter-AwS-Kommunikationskanäle zu verzeichnen. Dokumentiert werden kann dies in der Spalte Ist-Automatisierung (*Ist-Automat.*) in der Tab. 6.7. Zur Visualisierung der Ist-Automatisierung werden die Symbole auf der rechten Seite von Abb. 6.6 verwendet.

Die in der Tab. 6.7 erhobenen Ergebnisse können auf die Struktursicht des GP übertragen werden. Zur Verdeutlichung wurde in Abb. 6.12 die Visualisierung für das laufende Fallbeispiel exemplarisch vorgenommen. Damit unterstützen sowohl die Tab. 6.7 als auch die Abb. 6.12 die Identifikation der Deltas. Diese sind anhand ihrer

Tabelle 6.7: Hilfstabelle zur Dokumentation des Sollkonzepts für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen

Transaktion od. Transaktionsgr.	ausgepr. Merk. im Unternehm. menspl.	Period.	Größe	Intensität	Anpassb. Anpasst.	Nutzungs- d. Handl.	ausgepr. Merk. auf Ressourcenebene	Soll-Automat.	Ist-Automat.
Name der Transaktion od. Transaktionsgr.	Auflistung ausgepr. Merk. im Unternehm. menspl.	Auspr. der Periode	Auspr. der Größe	Auspr. der Intensität	Auspr. der Anpassbarkeit	Auspr. der Nutzungsdauer	abgel. Handlungsempf. auf Ressourcenebene	--> / →	<input type="checkbox"/> / <input type="checkbox"/> / <input checked="" type="checkbox"/>
<i>D: MeldLvAngeb</i>	<i>Strategie</i>	<i>niedrig</i>	<i>niedrig</i>	<i>hoch</i>	<i>niedrig</i>	<i>langfristig</i>	<i>2 Interop., Know-how, Einfachheit, 5.000 €</i>	<i>→</i>	<input type="checkbox"/>
...

Tabelle 6.8: Zusammenfassung der freiheitsgradbeschränkenden Faktoren

Ebene der SOM-Unternehmensarchitektur	beeinträchtigende Größen
Ebene des Unternehmensplans	<ul style="list-style-type: none"> – Strategie hinsichtlich Adoption und Einsatz von Technologien und folglich die Auswahl der zur Verfügung stehenden Basismaschinen – Einfluss von oder auf andere geplante oder gegenwärtig laufende Projekte – Einhaltung von Normen – rechtliche Rahmenbedingungen, wie Gesetzesvorgaben – Bestimmungen zu Datenschutz und Datensicherheit (Zugriffsberechtigungen auf AwS mit sensiblen Daten)
GP-Ebene	<ul style="list-style-type: none"> – Merkmale aus Abschnitt 5.4.5 sowie Handlungsempfehlungen aus Abschnitt 5.5
Ressourcenebene	<ul style="list-style-type: none"> – Interoperabilität der AwS – Know-how zur Realisierung der Integrationsarchitektur – Einfachheit der Implementierung – Kosten der Implementierung sowie Kostenvorgaben

Potenziale hinsichtlich ihrer Automatisierbarkeit erkennbar (☐). Mittels diesem Symbol wird verdeutlicht, für welche Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen Handlungsbedarf besteht. Die Abb. 6.12 bildet zudem das Gestaltungskonzept für das darauf aufbauende Implementierungsprojekt.

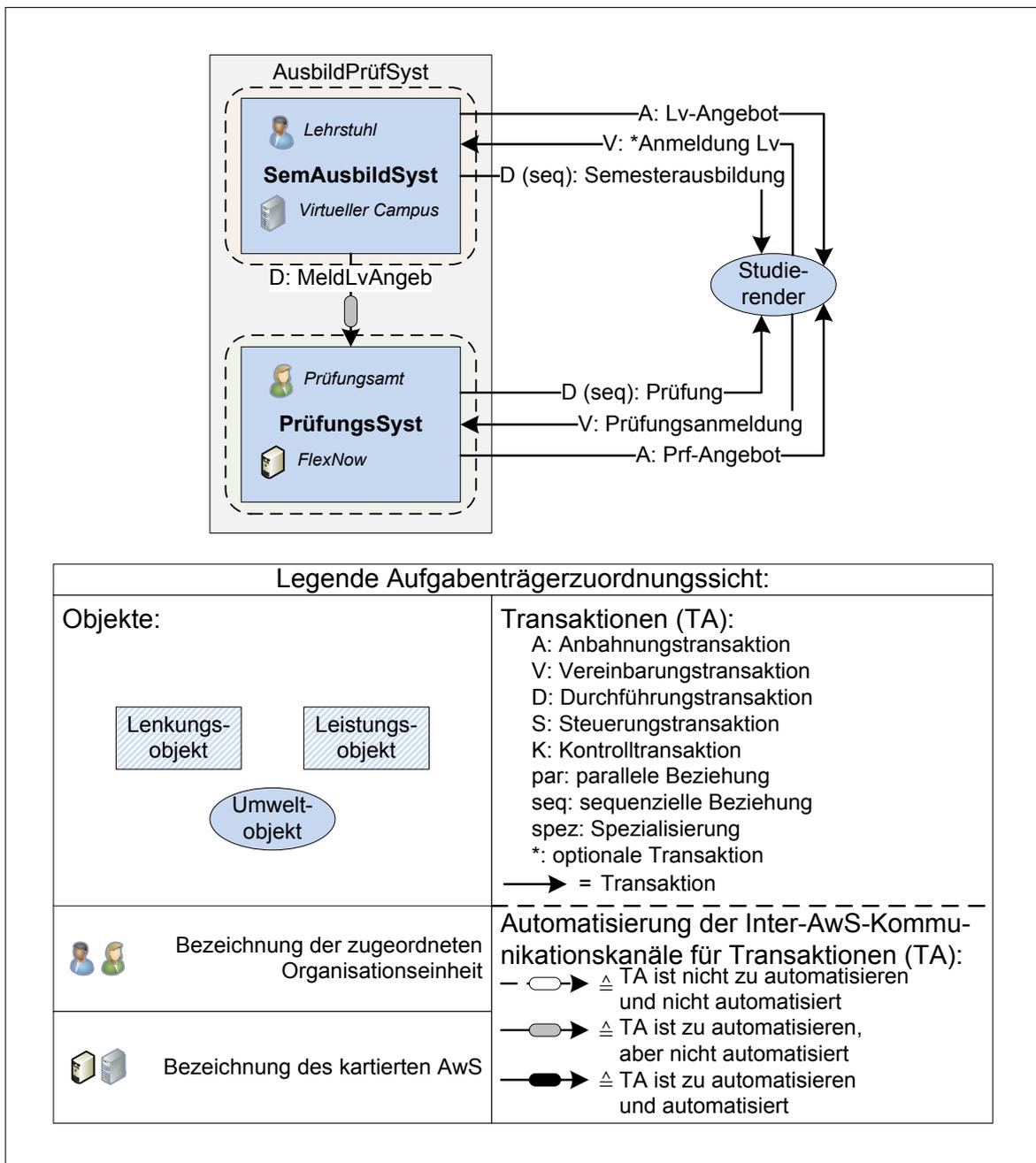


Abbildung 6.12: Soll-Konzept für das Szenario des Fallbeispiels

6.3.2.4 Ableitung der zu verwendenden Integrationskonzepte für die Inter-AWS-Integrationen

Ziel der im Folgenden beschriebenen Aktivität ist unter Abstimmung zu den vorhergehenden Schritten die Ableitung der zu verwendenden Integrationskonzepte für die Inter-AWS-Integrationen der in Abschnitt 6.3.2.3 identifizierten Deltas (□). Dabei sind ausgeprägte Integrationsmerkmale, AIM, erhoben in Abschnitt 6.3.1.1, sowie Freiheitsgrade, ermittelt in Abschnitt 6.3.2.2, Faktoren, die die Auswahl an Integrationskonzepten einschränken. Sofern im Rahmen der Durchführung dieses Schrittes festgestellt wird, dass die im Sollkonzept definierten Vorgaben nicht umsetzbar sind, ist mittels einer Simultanplanung zwischen der Ermittlung des Sollkonzepts sowie der Ableitung der zu verwendenden Integrationskonzepte der Konflikt aufzulösen. Sollte dies eine Änderung des GP nach sich ziehen, sind die durch AT-Restriktionen induzierten Änderungen auf den GP zu übertragen. Die Möglichkeiten, die hierfür zur Verfügung stehen, werden in Abschnitt 6.3.2.5 erläutert. Im Rahmen der aktuellen Aktivität wird die fachliche Ebene zur Erstellung eines Modells für das betriebliche Objektsystem verfeinert.

Zunächst sind die Integrationsmerkmale auszuprägen. Hierfür bietet sich zur Unterstützung die Verwendung der Tab. 6.9 an. Die Tab. 6.9 stützt sich auf die Integrationsmerkmale aus Abschnitt 3.3 sowie auf Tab. 3.1. Exemplarische Ausprägungen für das laufende Beispiel werden in *kursiver* Formatierung dargestellt.

Mit den definierten Integrationszielen sowie unter Berücksichtigung der AIM, die bereits in Abschnitt 6.3.1.1 und in Tab. 6.2 erhoben wurden, kann die Wahl des Integrationskonzepts eingegrenzt werden. Dazu wird vorgeschlagen, die Tab. 3.2 aus Abschnitt 3.3.2.5 mit den Unterstützungspotenzialen für die AIM durch die Integrationskonzepte (Ferstl und Sinz 2013, S. 251-252) zu kombinieren. Das Ergebnis dieser Kombination wird in Tab. 6.10 dargestellt. Aus dieser kann entnommen werden, ob ein Integrationskonzept die Automatisierung eines AIM unterstützt (X) oder nicht (-). Bei den Integrationszielen ist zu unterscheiden, ob Methoden zur Zielverfolgung verfügbar sind (●), Teilansätze für eine methodische Zielverfolgung vorliegen (◐) oder das Ziel nicht verfolgt wird bzw. keine methodischen Ansätze für eine Zielverfolgung zur Verfügung stehen (○).

Mit den ausgeprägten Integrationsmerkmalen, den AIM sowie den Freiheitsgraden, kann ein Integrationskonzept für eine Inter-AWS-Integration abgeleitet werden. Da jedes Integrationskonzept Auswirkungen auf das Informationssystem hat, werden diese in der Tab. 6.11 als Entscheidungshilfe dargestellt.

In der Tab. 6.11 stehen in der ersten Spalte die Integrationskonzepte aus Abschnitt 3.3.2. In den darauf folgenden drei weiteren Spalten werden die erwarteten Folgen der

Tabelle 6.9: Hilfstabelle zur Dokumentation der zu erreichenden Integrationsziele für das Szenario

Gruppe	Integrationsmerkmal	Ausprägung	AwS-Unterstützung
Struktur	Datenredundanz	<i>Für das vorliegende Fallbeispiel wird eine redundante Datenhaltung akzeptiert.</i>	○
	Funktionsredundanz	<i>Mehrfach in den unterschiedlichen AwS implementierte Lösungsverfahren werden akzeptiert.</i>	○
	Kommunikationsstruktur	<i>Dokumentation der Kommunikationsstruktur durch benutzte Ein-/Ausgänge ist erwünscht. Zudem ist diese in den AwS kodifiziert.</i>	●
Verhalten	semantische Integrität	<i>Organisatorische Richtlinien zur Einhaltung dieser sind vorhanden.</i>	○
	operationale Integrität	<i>Organisatorische Richtlinien zur Einhaltung dieser sind vorhanden.</i>	○
	Vorgangsteuerung	<i>Organisatorische Richtlinien zur Einhaltung dieser sind vorhanden.</i>	○

jeweiligen Integrationskonzepte in Bezug auf die AIM beschrieben. Dabei bezieht sich die zweite Spalte auf die *Homogenisierung partiell gleicher AO-Typen*, die dritte auf die *Homogenisierung partiell identischer AO-Instanzen* und die vierte auf partiell gleiche Lösungsverfahren (*Ausw. auf partiell gleiche LV*).

In der fünften und sechsten Spalte werden die erwarteten Auswirkungen auf die Mensch-Maschine-Kommunikation, die mit der Implementierung der jeweiligen AT-Konfiguration einhergehen können, aufgezeigt. Dabei wird getrennt zwischen den prognostizierten Effekten auf die *Mensch-Maschine-Kommunikation₁* und folglich der zeitlich vorgelagerten Aufgabe sowie denen auf die *Mensch-Maschine-Kommunikation₂*, welche die erwarteten Implikationen auf die nachgelagerte Aufgabe aufzeigt.

Voraussetzung bei der Wahl eines Integrationskonzepts mit maschineller Beteiligung ist, dass die involvierten AwS kopplungsfähig sind oder kopplungsfähig gemacht werden können. Die bestehenden Freiheitsgrade hierfür wurden bereits in Abschnitt 6.3.2.2 auf Ressourcenebene erhoben und wirken einschränkend auf die Wahl des Integrationskonzepts. Sollten die beteiligten AwS nicht koppelbar sein, ist das Integrationskonzept aufgabenträgerorientierte Funktionsintegration zu wählen.

Tabelle 6.10: Hilfstabelle zur Wahl des Integrationskonzepts

		Integrationskonzepte			
		Funktions- integration	Daten- integration	Objekt- integration	
AIM	Reihenfolge- beziehung	X	X	X	
	Partielle Gleichheit von AO-Typen	-	X	X	
	Partielle Identität von AO- Instanzen	- ○	X ●	X ●	Datenredundanz
	Partielle Gleichheit von Lösungsverfah- ren	- ○	- ○	X ●	Funktions- redundanz
		●	○	●	Kommunikati- onsstruktur
		○	●	●	Integrität
		◐	○	●	Vorgangs- steuerung

Integrationsziele

Tabelle 6.11: Integrationskonzepte und ihre Auswirkungen

	Homogenisierung	Homogenisierung	Ausw. auf	Auswirkungen auf (sofern vorhanden)
Integrationskonzept	partiell gleicher AO-Typen	partiell identischer AO-Instanzen	partiell gleiche LV	Mensch-Maschine-Kommunikation₁ Mensch-Maschine-Kommunikation₂
 aufgaben-trägerorientierte Funktionsintegration	wird nicht gelöst – Behelfslösung durch Abstimmung der Attribute partiell gleicher AO-Typen	Lösung kann redundante AO-Instanzen erzeugen.	wird nicht adressiert	bleibt unverändert bleibt unverändert; Daten müssen im empfangenden AwS erneut eingegeben werden
 datenflussorientierte Funktionsintegration	wird nicht gelöst – Behelfslösung durch Mapping der Attribute partiell gleicher AO-Typen	Lösung kann redundante AO-Typen erzeugen.	wird nicht adressiert	Bleibt unverändert, da nach einer Änderung nicht mehr Attribute des AO erfasst werden als vorher.
 Datenintegration	trägt zur Bildung einer einheitlichen Datenstruktur bei den involvierten AwS bei	Unterstützt die Vermeidung redundanter AO-Instanzen.	wird nicht adressiert	Sinkt, nicht überlappende Attribute sind nach wie vor zu pflegen.

Tabelle 6.11 – Fortsetzung von vorhergehender Seite

Integrationskonzept	Homogenisierung partiell gleicher AO-Typen	Homogenisierung partiell identischer AO-Instanzen	Ausw. auf partiell gleiche LV	Auswirkungen auf (sofern vorhanden) Mensch-Maschine-Kommunikation₁ Mensch-Maschine-Kommunikation₂
 Objektintegration	wird nicht gelöst – Behelfslösung durch Mapping der Attribute partiell gleicher AO-Typen	Lösung kann redundante AO-Instanzen erzeugen.	wird nicht adressiert	Steigt um die Pflege des Ereignisses; personeller Kommunikationskanal entfällt Sinkt, nicht überlappende Attribute sind nach wie vor zu pflegen.

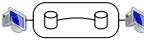
Nachfolgend wird die Wahl des Integrationskonzepts exemplarisch anhand des laufenden Beispiels erläutert. Zur Wahl des Integrationskonzepts sind nur diejenigen Transaktionen von Interesse, die eine maschinelle Unterstützung erfordern (Tab. 6.7). Für diejenigen Transaktionen, die nicht maschinell zu unterstützen sind, gilt das Integrationskonzept aufgabenträgerorientierte Funktionsintegration (Abschnitt 3.3.2.1).

Aus der Tab. 6.10 kann entnommen werden, dass das in Tab. 6.2 identifizierte AIM Reihenfolgebeziehung sowie die Übertragung partiell identischer AO-Instanzen von den Integrationskonzepten Datenintegration und Objektintegration unterstützt wird. Mit dem Integrationskonzept datenflussorientierte Funktionsintegration wird das AIM partielle Identität von AO-Instanzen insoweit nicht unterstützt, als dass damit redundante AO-Instanzen erzeugt werden. Wie in der Tab. 6.9 dargestellt, wird hingegen eine redundante Datenhaltung akzeptiert. Mit dieser Klärung erfolgt ein Übergang von den AIM als einschränkende Faktoren bei der Wahl des Integrationskonzepts zu den Integrationszielen.

Für das laufende Beispiel ist die AwS-Unterstützung für das Integrationsmerkmal Kommunikationsstruktur erforderlich (Tab. 6.9). Dieses Merkmal wird von den beiden Integrationskonzepten datenflussorientierte Funktionsintegration und Objektintegration unterstützt (Tab. 6.10).

Unter Abgleich der Freiheitsgrade aus Abschnitt 6.3.2.2 gilt es, die Strategie hinsichtlich Adoption und Einsatz von Technologien und folglich die Auswahl der zur Verfügung stehenden Basismaschinen zu berücksichtigen. Diese gibt vor, dass gegenwärtig keine EAI-Lösung eingeführt werden darf. Alle Vorgaben, AIM, Integrationsziele sowie Freiheitsgrade, können lediglich mit dem Integrationskonzept datenflussorientierte Funktionsintegration umgesetzt werden, deshalb ist dieses zu wählen, auch wenn damit eine redundante Datenhaltung einher geht. Mithilfe der Tab. 6.12 wird das Ergebnis der Ableitung des Integrationskonzepts für das Szenario dokumentiert.

Tabelle 6.12: Abgeleitetes Integrationskonzept für das Szenario des Fallbeispiels

Betriebliches Objekt	AwS	verwendetes Integrationskonzept
<i>SemAusbildSyst</i>	 <i>Virtueller Campus</i>	 <i>datenflussorientierte Funktionsintegration</i>
<i>PrüfungSyst</i>	 <i>FlexNow</i>	

6.3.2.5 Optionen bei Unvereinbarkeit zwischen angestrebten und möglichen Integrationskonzept

In diesem Abschnitt werden Hinweise in Form von drei Optionen zur Durchführung der Simultanplanung gegeben, sofern sich das aus Abschnitt 6.3.2.4 gewählte Integrationskonzept als nicht realisierbar erweisen sollte.

Im Rahmen der ersten Option sind zur Erreichung der Integrationsziele durch die involvierten AwS Abstriche zu machen und Hilfslösungen für die Defizite vorzusehen. Statt der Automatisierung durch die AwS können bspw. organisatorische Richtlinien, wie Festlegung von Abgabefristen, Einführung von Checklisten oder Verfahrensbeschreibungen, erlassen werden. Diese Option kann bspw. angewendet werden, wenn die Objektintegration gefordert wird, diese sich jedoch als nicht realisierbar erweist. So kann die fehlende Unterstützung bspw. des Integrationsmerkmals Vorgangsteuerung bei dem realisierbaren Integrationskonzept durch organisatorische Richtlinien kompensiert werden. Insgesamt führt dies zu einer Neuwahl beim Integrationskonzept.

Die zweite Option besteht darin, die durch AT-Restriktionen induzierten Änderungen auf den GP zu übertragen (Abschnitt 6.3.2). Im Rahmen dieser Option können bspw. betriebliche Objekte simultan unter Berücksichtigung der AT-Ebene solange verfeinert werden, bis das angestrebte Integrationskonzept verwendet werden kann.

Die dritte Möglichkeit besteht aus Änderungen an der AT-Ebene, so dass das gewählte Integrationskonzept verwendet werden kann. Im Rahmen dieser Möglichkeit werden drei Unteroptionen subsumiert. Mit der ersten Unteroption sind vorhandene AwS durch eines, welches die involvierten betrieblichen Objekte mit dem angestrebten Integrationskonzept unterstützen kann, abzulösen. Die zweite Unteroption besteht, ebenfalls wie die erste Unteroption, darin, vorhandene AwS durch eines abzulösen, welches die Integration der betroffenen betrieblichen Objekte mit dem verfolgten Integrationskonzept erlaubt. Im Gegensatz zur ersten Unteroption sind hierbei nicht mehrere vorhandene AwS durch eines abzulösen, sondern lediglich das AwS durch ein anderes zu ersetzen, mit welchem das angestrebte Integrationskonzept erreicht werden kann. Im Rahmen der dritten Unteroption sind vorhandene AwS so auszubauen, dass damit die Aufgabenbereiche weiterer betrieblicher Objekte übernommen und andere vorhandene AwS abgelöst werden können. Dabei sind die Prinzipien der Kohäsion und Kopplung, wie in Abschnitt 3.1.4 erläutert, zu berücksichtigen.

6.4 Techniken und Werkzeuge für die Methodendurchführung

Unterstützende Werkzeuge für die Methodendurchführung sind Visualisierungsprogramme sowie gängige Bürosoftware (Textverarbeitungsprogramme, Tabellenkalkulationsprogramme). Darüber hinaus können zur Modellierung von GP am Markt verfügbare Tools eingesetzt werden.

Expertenbefragungen, Fragebögen, Interviews sowie Auswertung von Logdateien, insbesondere bei der Erhebung der Leistungsanforderungen (Tab. 5.8), sind hilfreiche Techniken bei der Methodendurchführung.

Im Rahmen der Bearbeitung des Forschungsvorhabens wurde zur Unterstützung bei der Durchführung der Methode ein Prototyp erstellt. Es hat sich herausgestellt, dass es zielführender ist, die bei den jeweiligen Aktivitäten erzielten Ergebnisse durch ein Werkzeug in strukturierter Form zu verwalten anstatt sich ausschließlich auf die genannten Werkzeuge zu verlassen. Zweitens leitet der Prototyp den Anwender durch die einzelnen Aktivitäten der Methode. Drittens können mithilfe des erstellten Prototypen Aktivitäten teilautomatisiert ausgeführt werden. Dies trifft insbesondere auf die Ableitung der Handlungsempfehlungen zu (Abschnitt 5.5). Viertens kann durch in der Software hinterlegte Bedingungen die Einhaltung der Regeln bei der Durchführung der Methode, insbesondere bei der Ableitung der zu verwendenden Integrationskonzepte (Abschnitt 6.3.2.4), gewahrt werden. Fünftens erleichtert der Prototyp die abschließende What-If-Analyse, indem die in den Szenarien erzielten Resultate automatisiert durch die Software ausgewertet und dem Anwender präsentiert werden.

Mit dem Prototypen können keine IAS erstellt werden. Er lässt hingegen einen Import von in anderen Werkzeugen erstellten Modellen als Grafik sowie die Einbindung bei den jeweiligen Aktivitäten zu. Das Programm selber dient zur Verwaltung der in den Modellen verwendeten Bestandteile. Hierzu ist das in Abb. 6.13 dargestellte Datenmodell des Prototypen als SERM⁵⁸ maßgeblich. Ausgangspunkt dieses Datenmodells ist die Entität *Fallstudie*. Im Prototyp werden sowohl für die *Istanalyse* als auch für die *Szenarien* die Entitäten *Aufgabenträger*, *betriebliches Objekt*, *Transaktion* sowie *Aufgabenobjekt* und ihre jeweiligen Attribute verwaltet. Zur Unterstützung der Spezifikation des zu analysierenden GP (Abschnitt 6.3.1.1, Abschnitt 6.3.2) werden die betrieblichen Objekte sowie die Transaktionen und deren Beziehungen in den korrespondierenden Tabellen persistiert. Daneben werden die identifizierten AIM im ER-Typ *Transaktion* sowie die überlappenden AO im korrespondierenden ER-Typ erfasst. Anschließend werden die AT aufgenommen und den betrieblichen Objekten zugeordnet (Abschnitt 6.3.1.2, Abschnitt 6.3.2.1). Die Entität AT ist dabei zu

⁵⁸SERM wird in (Ferstl und Sinz 2013, S. 158-179) eingeführt.

spezialisieren in *AwS* und *personeller AT*. Die *AwS*-Grenzen können algorithmisch vom Prototypen bestimmt werden, da über die Beziehung zwischen den Entitäten *AT* und betriebliches Objekt geprüft werden kann, welche betrieblichen Objekte von einem bestimmten *AT* unterstützt werden bzw. welche betrieblichen Objekte ein *AT* unterstützt. Auch die Selektion der Inter-*AwS*-Kommunikationskanäle ist durch diese Beziehung möglich. Dadurch kann die Ermittlung der erforderlichen Soll-Automatisierung insoweit unterstützt werden, als dass die Inter-*AwS*-Kommunikationskanäle dem Anwender angezeigt werden, dieser die Attribute im ER-Typ *Transaktion* zu den Inter-*AwS*-Kommunikationskanälen erfasst und abschließend die Soll-Automatisierung festlegt (Abschnitt 6.3.2.3). Die Erfassung des Ist-Zustands für die Inter-*AwS*-Kommunikationskanäle wird unterstützt, indem für jeden selbiger im ER-Typ *Transaktion* eingepflegt wird. Abschließend unterstützt der Prototyp mit Hinweisen die Ableitung der zu verwendenden Integrationskonzepte für die Inter-*AwS*-Integrationen (Abschnitt 6.3.2.4). Die gewählten Integrationskonzepte werden ebenfalls im ER-Typ *Transaktion* aufgenommen. Die What-If-Analyse wird durch den Vergleich der Ergebnisse aus der Istanalyse (Abschnitt 6.3.1) sowie der entworfenen Szenarien (Abschnitt 6.3.2) unterstützt.

Der Prototyp wurde in der Programmiersprache C# implementiert und ist auf dem .NET-Framework in der Version 4.5.2 lauffähig. Für die Implementierung der grafischen Oberflächen wurde auf die Windows Presentation Foundation (WPF) zurückgegriffen. Grundlage für die Persistenzschicht ist das in Abb. 6.13 dargestellte SERM. Die Daten werden in einer MSSQLLocalDB in der Version 12.0.2000 persistiert. Zur Objektrelationalen Abbildung (ORM) zwischen der Datenbank und dem Programm wird das Entity Framework in der Version 6.1.3 eingesetzt. Bei der Implementierung wurde auf die Verwendung gängiger Patterns aus der Softwareentwicklung, wie bspw. Model-View-Controller (MVC), geachtet.

Zum Zwecke der Verwendung, der Untersuchung sowie der Weiterentwicklung des Prototypen wird der Quellcode öffentlich⁵⁹ zur Verfügung gestellt. Der Prototyp selber wird zur Unterstützung bei der Durchführung der Fallstudie in Kapitel 7 ergänzend zu den eingeführten Hilfstabellen herangezogen und bei der Ausführung der jeweiligen Aktivitäten detailliert vorgestellt. Die Ergebnisse werden in Form von Screenshots des entwickelten Prototypen bekannt gemacht.

⁵⁹Der Quellcode kann unter <https://bitbucket.org/flobad/integrationsbedarfe> eingesehen und heruntergeladen werden.

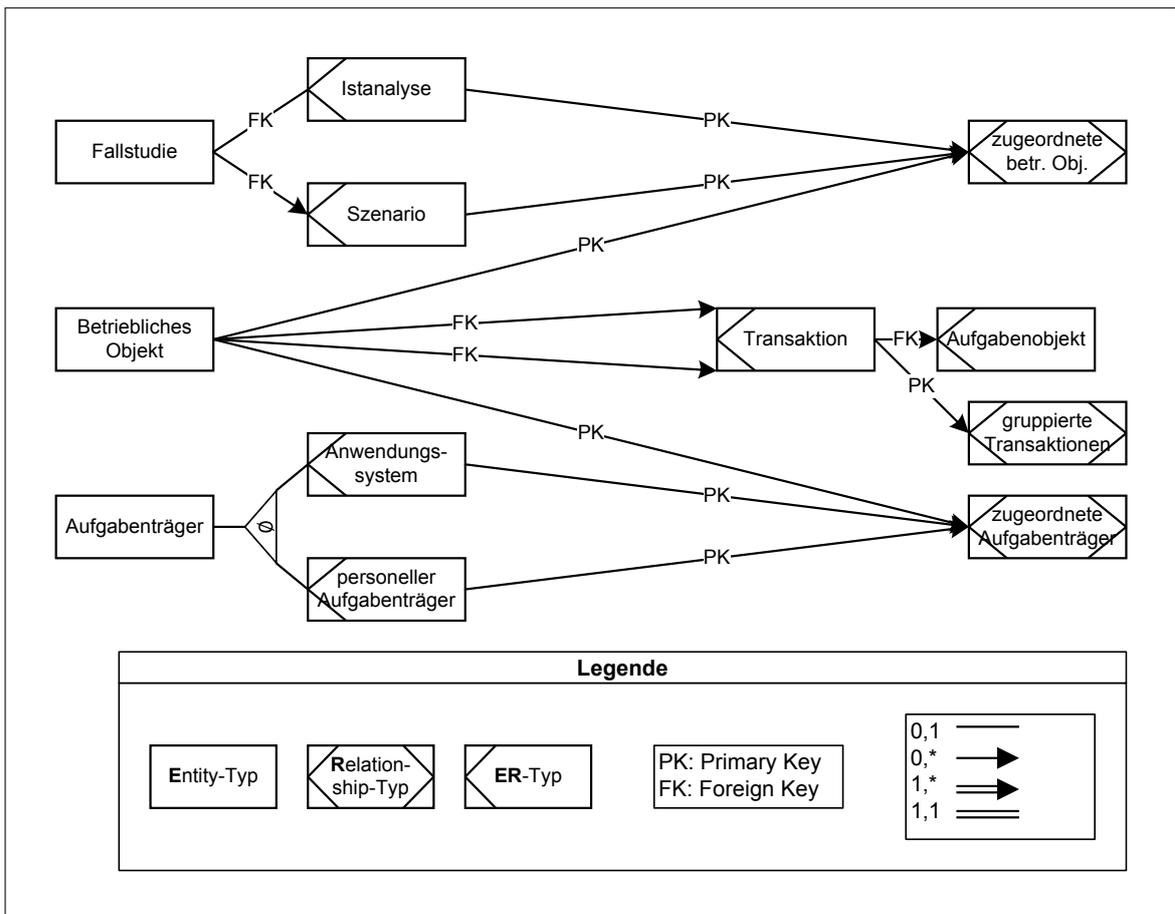


Abbildung 6.13: Datenmodell des unterstützenden Prototypen als SERM

6.5 Zusammenfassung und Diskussion

Im vorliegenden Kapitel wurde eine auf Handlung ausgerichtete Methode entwickelt, mithilfe derer Integrationsarchitekturen aus GP abgeleitet werden können. Damit wird die in Abschnitt 1.2, Abb. 1.2 definierte *Forschungsfrage* nach der methodischen Unterstützung beantwortet. Daneben wird das *Untersuchungsergebnis* in Form von einer auf Pragmatik ausgerichteten Methode vorgestellt.

Basis der entwickelten Methode ist die prozess- und objektorientierte SOM-Methodik. Kern des konstruierten Vorgehensmodells (Abschnitt 6.3, Abb. 6.7) ist die What-If-Analyse. Im Rahmen der Durchführung der Methode wird zunächst der Ist-Zustand aufgenommen (Abschnitt 6.3.1). Anschließend werden Szenarien gebildet (Abschnitt 6.3.2). Für die Erstellung der jeweiligen Sollkonzepte werden die in Abschnitt 5.5 vorgestellten Merkmale und Handlungsempfehlungen eingebunden. Zur Umsetzung der Sollkonzepte werden jeweils passende Integrationskonzepte abgeleitet (Abschnitt 6.3.2.4). Durch die geschilderte Vorgehensweise entsteht unter Berücksichtigung personeller AT die für die Szenarien zu implementierende Integrationsarchitektur. Dadurch wird, wie in Abschnitt 3.1.7 dargelegt, ein umfassendes Verständnis bei der Gestaltung der Integrationsarchitektur zugrunde gelegt. Die Ausführungen wurden anhand eines Fallbeispiels aus dem Hochschulumfeld verdeutlicht.

Das Ergebnis der Durchführung kann zum einen für den Vergleich der Szenarien und zum anderen als Grundlage (Blueprint) für die Implementierung herangezogen werden. Dazu sind mithilfe der SOM-Methodik die zu übertragenden Nachrichtentypen abzuleiten. Diese dienen zur Definition von Nachrichtentypen, die zur Durchführung von Transaktionen eingesetzt werden, bei denen ähnliche AO übertragen werden. Dadurch wird sowohl die Erstellung von Anwendungsmodellen als auch die Konzeption von Softwarearchitekturen unterstützt. Dabei wird der Übergang von der fachlichen Ebene auf die softwaretechnische eingeleitet.

In Tab. 6.13 werden die Bestandteile der entwickelten Methode überblicksartig vorgestellt.

In Tab. 6.14 wird die in diesem Kapitel entwickelte Methode anhand der in Abschnitt 4.3 definierten Kriterien bewertet. Bei der Bewertung der jeweiligen Anforderung wird unterschieden, ob sie erfüllt (●), bedingt erfüllt (◐) oder nicht erfüllt bzw. kein Hinweis auf Erfüllung gegeben wird (○). Aus der Tab. 6.14 wird ersichtlich, dass die in Abschnitt 4.5 identifizierten Defizite – unvollständige Abdeckung der in Abschnitt 4.3 definierten Anforderungen und nicht ausreichende Berücksichtigung der personellen AT bei der Gestaltung der Integrationsarchitektur – durch die im vorliegenden Kapitel entwickelte Methode geschlossen werden können.

Tabelle 6.13: Überblick der Bestandteile der entwickelten Methode

Bestandteil	Ausprägung
Metamodell	Das Metamodell der Methode wird in Abschnitt 6.1 vorgestellt und in Abb. 6.3 dargestellt.
Architektur	Durch die Einbindung der SOM-Methodik als Basis, werden die drei Modellebenen, wie in Abschnitt 2.1 beschrieben und in Abb. 2.2 dargestellt, eingebunden. Es existieren die Sichten Interaktions-, Verhaltens- sowie Aufgabenträgerzuordnungssicht (Abschnitt 6.2). Die Abstimmung zwischen den Sichten wird durch das integrierte Metamodell der Methode gewährleistet (Abb. 6.3).
Vorgehen	Die Reihenfolge bei der Anwendung der Methode wird durch ein Vorgehensmodell (Abschnitt 6.3) vorgegeben.
Werkzeug	Hinweise für unterstützende Werkzeuge werden in Abschnitt 6.4 gegeben.

Tabelle 6.14: Evaluation der entwickelten Methode

Kriterium	Bewertung	Begründung
What-If-Analyse (I1)	●	Dieser Ansatz erlaubt einen Vergleich von Szenarien, da zwischen vorliegendem Ist-Zustand in Abschnitt 6.3.1 und Bildung von Soll-Zuständen in Abschnitt 6.3.2 differenziert wird. Darüber hinaus werden zudem Optionen zur Rücküberführung von auf der Ressourcenebene ausgehenden Modifikationen auf die GP-Ebene in Abschnitt 6.3.2.5 skizziert.
Unternehmensplan (I2)	●	Auf die Erhebung möglicher Einschränkungen beim Entwurf der Integrationsarchitektur auf Ebene des Unternehmensplans wird in Abschnitt 6.3.2.2 sowie in Tab. 6.8 hingewiesen.
Integrationsbedarfe (I3)	●	Integrationsbedarfe stammen von fachlichen Beziehungen auf Aufgabenebene – ersichtlich insbesondere aus der Interaktionssicht im Rahmen der GP-Spezifikation. Aus ihnen werden mithilfe der Methode im Rahmen der Sollkonzeption (Abschnitt 6.3.2.3) modellgestützt Gestaltungsbedarfe abgeleitet.

Tabelle 6.14 – Fortsetzung von vorhergehender Seite

Kriterium	Bewertung	Begründung
personelle AT (I4)	●	Die Potenziale personeller AT werden bei der Gestaltung der Integrationsarchitektur, insbesondere bei der Ableitung der Integrationskonzepte durch die Berücksichtigung des Integrationskonzepts <i>aufgabenträgerorientierte Funktionsintegration</i> berücksichtigt.
AwS (I5)	●	Für die Gestaltung der AwS-Integration werden die Integrationskonzepte aus Abschnitt 3.3.2 herangezogen. Durch die Szenarienbildung kann neben der Ablösung vorhandener AwS durch neu einzuführende oder zu entwickelnde, die größere Aufgabenkomplexe integrieren, auch die Möglichkeit der Neuordnung von Aufgaben zu AwS, die somit größere Aufgabenkomplexe integrieren, in Erwägung gezogen werden. Darüber hinaus kann auf diesem Wege auch die Alternative der Kopplung von AwS zur Unterstützung der Aufgabenintegration berücksichtigt werden.
Komplexitätsbeherrschung (F1)	●	Die Modellierung des Modellsystems wird in die drei Modellebenen, die die SOM-Methodik zur Verfügung stellt (Abschnitt 2.1, Abb. 2.2), unterteilt und fördert somit die Komplexitätsbeherrschung. Darüber hinaus werden durch die Verwendung der SOM-Methodik als Kernbestandteil die Sichten auf Funktionen (VOS), Daten (KOS), Interaktionen (IAS) sowie Vorgänge (VES und VOS) unterstützt.
Grafisch (F2)	●	Der Entwurf wird durch die Verwendung der Modelle, welche die SOM-Methodik bietet, sowie die in Abschnitt 6.2 vorgestellten Sichten grafisch unterstützt.

Die in Tab. 4.10 vorgenommene zusammenfassende Bewertung der Ansätze wird in Tab. 6.15 um die Bewertung der in diesem Kapitel entwickelten und in Tab. 6.14 bewerteten Methode erweitert. Auch in Tab. 6.14 wird bei der Bewertung der jeweiligen Anforderung unterschieden, ob sie erfüllt (●), bedingt erfüllt (◐) oder nicht erfüllt bzw. kein Hinweis auf Erfüllung gegeben wird (○).

Tabelle 6.15: Zusammenfassende Bewertung der Ansätze

		Ansätze						
		Engels und Voß (2008)	Gronau et al. (2008)	Jung (2006)	Katten- stroth et al. 2013	Schwinn (2005)	Vogler (2006)	Me- tho- de
Kriterien	What-If- Analyse (I1)	◐	○	○	◐	◐	◐	●
	Unternehmens- plan (I2)	◐	◐	○	◐	●	◐	●
	Integrationsbe- darfe (I3)	●	○	○	●	●	◐	●
	personelle AT (I4)	○	○	○	◐	○	●	●
	AwS (I5)	●	◐	●	◐	◐	◐	●
	Komplexitätsbe- herr- schung (F1)	●	◐	○	●	●	●	●
	Grafisch (F2)	●	◐	○	●	◐	○	●

7 Konzeption einer Integrationsarchitektur für das Informationssystem eines Güterdistributors

7.1 Istanalyse für ein existierendes Güterdistributionssystem	152
7.1.1 Spezifikation des zu analysierenden Geschäftsprozesses	152
7.1.2 Erhebung und Analyse des Ist-Zustands	154
7.2 Szenario: Einführung eines ERP-Systems	160
7.2.1 Szenarioorientierte Gestaltung des Geschäftsprozesses	160
7.2.2 Erhebung der Freiheitsgrade beim Entwurf der Integrationsarchitektur	160
7.2.2.1 Ebene des Unternehmensplans	161
7.2.2.2 Geschäftsprozessebene	161
7.2.2.3 Ressourcenebene	163
7.2.3 Ermittlung der Soll-Automatisierung für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen	164
7.2.4 Ableitung der zu verwendenden Integrationskonzepte für die Inter-AwS-Integrationen	167
7.2.4.1 Festlegung der Integrationsziele für das Szenario	167
7.2.4.2 Ermittlung der Integrationskonzepte für die Inter-AwS-Integrationen	167
7.3 What-If-Analyse für den Güterdistributor	171
7.4 Zusammenfassung und Diskussion	172

In diesem Kapitel wird die Anwendung der in Kapitel 6 entwickelten Methode exemplarisch an einer Fallstudie aus dem Güterdistributionsbereich verdeutlicht. Dabei wird nach der schematischen Darstellung von Abb. 6.7, erläutert in Abschnitt 6.3, vorgegangen. Im Rahmen der Fallstudie wird mithilfe der Methode untersucht, inwieweit Handlungsbedarfe nach einer Änderung auf AT-Ebene bei der Integration von Aufgaben existieren und wie diese gelöst werden können. Durch die Untersuchung wird einerseits das erzeugte Artefakt beispielhaft instanziiert und andererseits die Anwendbarkeit der entwickelten Methode nachgewiesen, wie bspw. von Offermann et al. (2010, S. 291) gefordert. Dabei wird der in diesem Kapitel durchgeführte Machbarkeitsnachweis durch den in Abschnitt 6.4 eingeführten Prototypen unterstützt. In Abschnitt 7.4 werden die Ergebnisse kritisch gewürdigt. Nicht Gegenstand dieses Kapitels ist die Durchführung der Implementierung der durch die Methodenanwendung identifizierten Gestaltungsmaßnahmen.

7.1 Istanalyse für ein existierendes Güterdistributionsystem

7.1.1 Spezifikation des zu analysierenden Geschäftsprozesses

Gegenstand der Fallstudie ist die Distribution von Konsumgütern. In Abb. 7.1 wird dazu der initiale GP dargestellt. Die für die Durchführung der Fallstudie erforderliche Granularität des GP wird in Abb. 7.2 abgebildet. Das dazugehörige Protokoll der Objekt- und Transaktionszerlegung kann aus Tab. C.1 entnommen werden. Dabei werden in der Tab. C.1 die in der Abb. 7.2 endgültig verwendeten betrieblichen Objekte *kursiv* und die Transaktionen *fett* hervorgehoben. Die übrigen betrieblichen Objekte und Transaktionen sind Zwischenschritte des Zerlegungsprozesses und werden deshalb nicht verwendet. Die zur Abb. 7.2 korrespondierende Verhaltenssicht auf den GP wird in Abb. C.1 dargestellt.

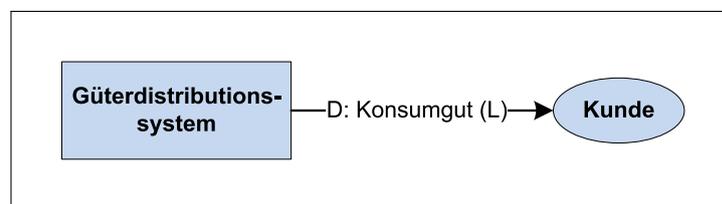


Abbildung 7.1: Initialer GP für das Beispiel der Güterdistribution

Nachfolgend wird der in Abb. 7.2 dargestellte GP beschrieben. Ausgangspunkt sind die dem Kunden angebotenen Produktinformationen. Der Kunde übermittelt anschließend die Auftragsdaten. Diese werden von der Auftragsprüfung zusammen mit der Prüfung der Bonität kontrolliert. Darüber hinaus untersucht die Auftragsprüfung, ob der Auftrag termingerecht bedient werden kann, indem sie beim Lager die Lagerbestände anfordert. Verläuft die Prüfung positiv, erhält der Kunde eine Auftragsbestätigung, andernfalls wird in der Auftragsbestätigung der Kunde über die Ablehnung des Auftrags informiert und der GP terminiert an dieser Stelle. Bei Auftragsannahme erteilt die Vertriebslenkung die Lieferanweisung an die Versandlenkung und die Forderungsanweisung an das Finanzwesen. Das Finanzwesen kümmert sich um die Zahlungsabwicklung der offenen Forderung. Die Versandlenkung stellt die Auslieferung der bestellten Waren sicher. Dazu gibt sie an das Lager eine Bereitstellungsanweisung heraus und schließt mit dem Logistikdienstleister einen Vertrag zur Auslieferung der Waren ab. Die bereitgestellten Waren werden letztlich vom Logistikdienstleister abgerufen und an den Kunden ausgeliefert.

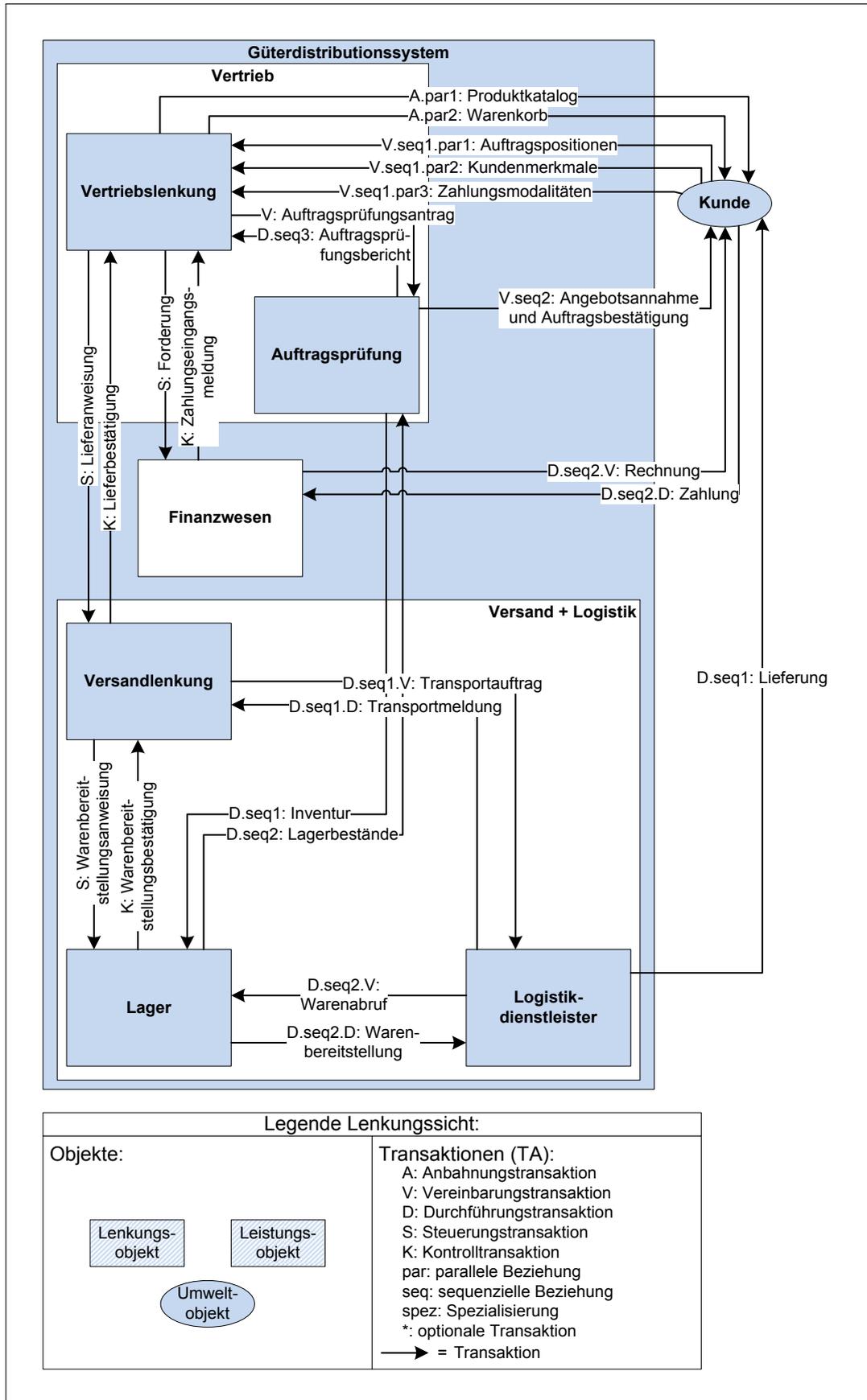


Abbildung 7.2: Struktursicht auf den Ist-GP der Fallstudie

In Abb. 7.3 werden die Abbildung betrieblicher Objekte (linker Bereich in der Abb. 7.3, versehen mit der Ziffer 1), die Zuordnung der Transaktionen (mittlerer Bereich in der Abb. 7.3 für das betriebliche Objekt *Auftragsprüfung*, versehen mit der Ziffer 2) einschließlich überlappender AO (rechter mittlerer Bereich in der Abb. 7.3, versehen mit der Ziffer 3) sowie die Erhebung der AIM (rechter unterer Bereich in der Abb. 7.3, versehen mit der Ziffer 4) mithilfe des Prototypen exemplarisch dargestellt. Der Transaktion *D.seq1: Inventur* in der Abb. 7.3 wird mittels einer Dropdown-Liste das sendende betriebliche Objekt sowie das empfangende betriebliche Objekt zugeordnet (rechter oberer Bereich in der Abb. 7.3, versehen mit der Ziffer 5). Die komplette Zuordnung betrieblicher Objekte zu den Transaktionen sowie die dabei identifizierten AIM und überlappenden AO sind der Tab. C.3 zu entnehmen.

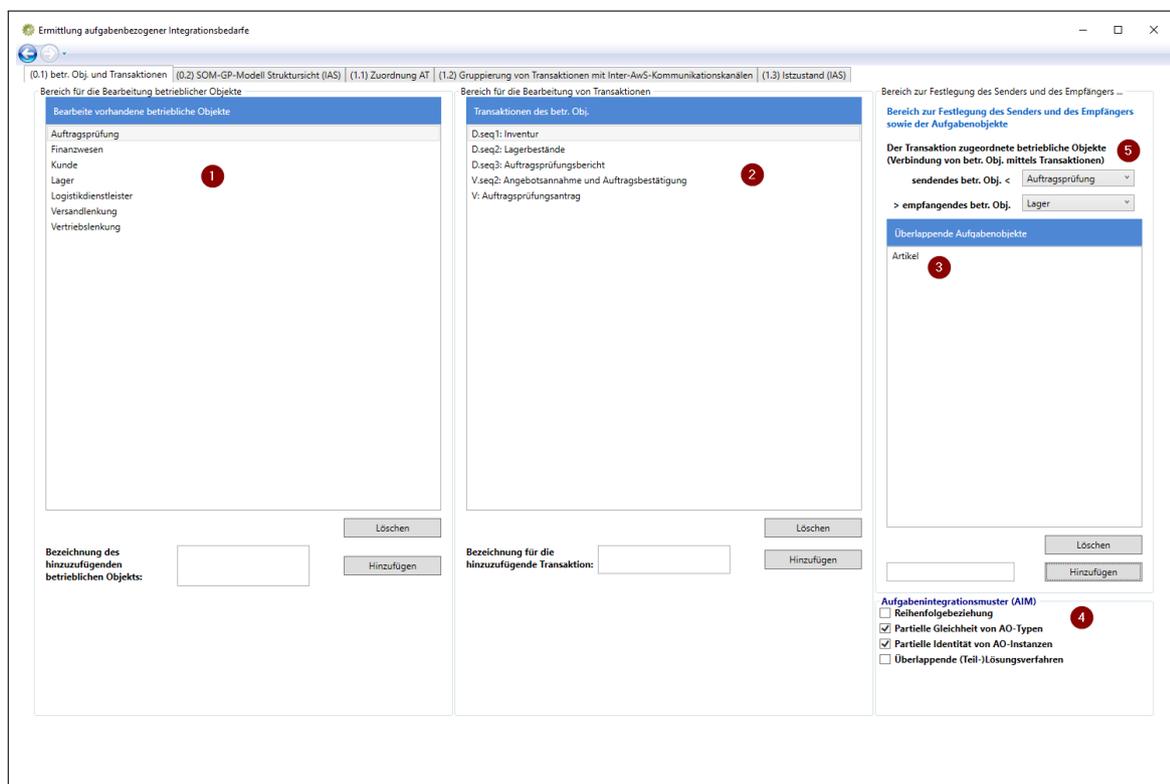


Abbildung 7.3: Abbildung von betrieblichen Objekten, Transaktionen, überlappenden AO sowie identifizierter AIM mithilfe des Prototypen

7.1.2 Erhebung und Analyse des Ist-Zustands

Dieser Schritt wird unterstützt, indem zunächst den betrieblichen Objekten die jeweiligen AT – ausführende Organisationseinheit sowie unterstützendes AWS – zugeordnet werden. In der Tab. 7.1, deren Basis Tab. 6.3 ist, wird das Ergebnis für die vorliegende Fallstudie dargestellt.

In der Fallstudie werden den betrieblichen Objekten *Vertriebslenkung* und *Auftragsprüfung* verschiedene personelle AT in unterschiedlichen Rollen zugeordnet. Die betroffenen personellen AT gehören jedoch der gleichen Organisationseinheit an. Aus diesem Grund wird dasselbe Symbol für die zugeordnete ausführende Organisationseinheit verwendet (vgl. dazu die Ausführungen in Abschnitt 6.3.1.2).

Die in Abschnitt 6.3.1.2 definierte Regel mit der Zuordnung der AwS zu einem betrieblichen Objekt kann eingehalten werden, weshalb eine weitere Verfeinerung des GP nicht erforderlich ist.

Tabelle 7.1: Betriebliche Objekte und die ihnen zugeordneten AT im Rahmen der Istanalyse

Betriebliches Objekt	ausführende Organisationseinheit	unterstützendes AwS
Vertriebslenkung	 Vertrieb	 Auftragsverwaltungssystem
Auftragsprüfung	 Vertrieb	 Auftragsverwaltungssystem
Versandlenkung	 Versand	 Auftragsverwaltungssystem
Lager	 Kommissionierung	 LVS
Logistikdienstleister	 Disponent	 Fremdsystem
Finanzwesen	 Finanzen	 SoftM-Suite FI

Zur Abbildung der zugeordneten AT ist beim entwickelten Prototyp zunächst ein betriebliches Objekt zu selektieren (linker Bereich in Abb. 7.4, versehen mit der Ziffer 1) und der Name der unterstützenden AT einzupflegen (mittlerer Bereich in Abb. 7.4, versehen mit der Ziffer 2). In Abb. 7.4 werden exemplarisch dem betrieblichen Objekt *Auftragsprüfung* der personelle AT *Vertrieb* sowie das AwS *Auftragsverwaltungssystem* hinzugefügt.

In Abb. 7.4 können durch die Selektion der AT die betrieblichen Objekte angezeigt werden, die sie unterstützen (rechter Bereich in Abb. 7.4, versehen mit der Ziffer 3). In der Abb. 7.4 werden dabei beispielhaft für den selektierten personellen AT *Vertrieb* die betrieblichen Objekte angezeigt, denen er zugeordnet wurde – im rechten oberen Bereich in Abb. 7.4 die beiden betrieblichen Objekte *Auftragsprüfung* und *Vertriebslenkung*. Gleiches gilt für die AwS. In Abb. 7.4 im rechten unteren Bereich wurde exemplarisch das AwS *Auftragsverwaltungssystem* ausgewählt. Dieses unterstützt die betrieblichen Objekte *Auftragsprüfung*, *Versandlenkung* sowie *Vertriebslenkung*.

7 Konzeption einer Integrationsarchitektur für das Informationssystem eines Güterdistributors

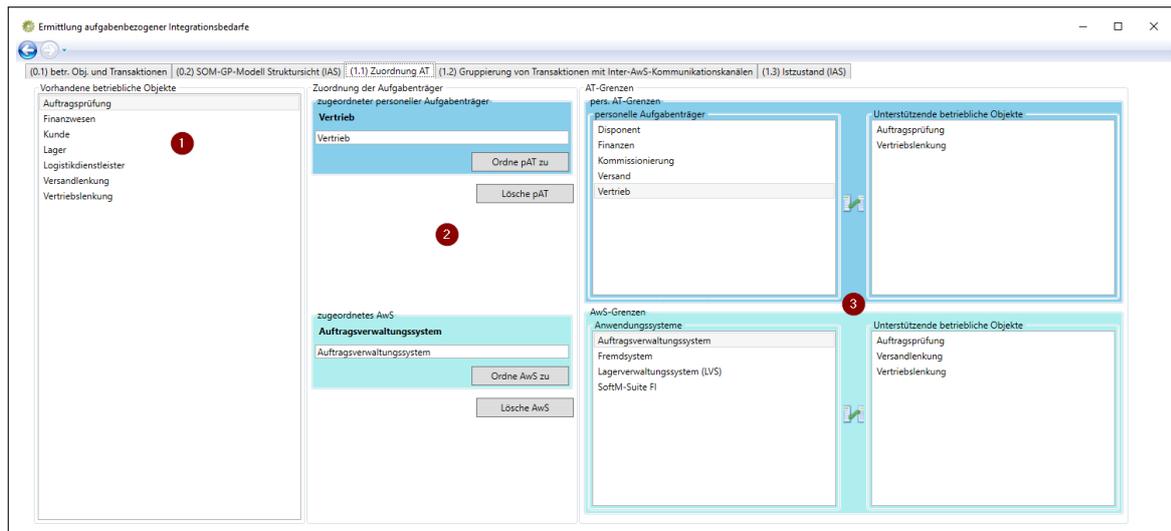


Abbildung 7.4: Zuordnung der AT mithilfe des Prototypen

Darauf aufbauend können Transaktionen nach den Hinweisen aus Abschnitt 6.3.1.3 ggf. gruppiert werden. Die so zusammengefassten Transaktionen lassen sich durch einen Kommunikationskanal abdecken, wie in Abschnitt 6.3.1.2 erläutert. Dadurch kann die Vergleichbarkeit bei der abschließenden Analyse der Szenarien in Abschnitt 7.3 unterstützt werden. Die Unterstützung der Vergleichbarkeit zielt darauf ab, dass bei allen Szenarien die verwendbaren, gruppierten Transaktionen des GP aus der Istanalyse identisch benannt werden, um gleiche Gruppen von Transaktionen über unterschiedliche Szenarien hinweg miteinander vergleichen zu können. Daneben kann durch die Zusammenfassung von Transaktionen die Skalierbarkeit des Ansatzes unterstützt werden.

In Abb. 7.5 wird gezeigt, wie der Prototyp zur Gruppierung von Transaktionen eingesetzt werden kann. Da im Beispiel keine Transaktionsgruppe gebildet werden kann, wird für die nachfolgende Beschreibung exemplarisch die Transaktion *D.seq1: Inventur* gewählt.

Die Anzeige gruppierter Transaktionen sowie von Transaktionen ohne Gruppe werden im linken oberen Bereich in Abb. 7.5, versehen mit der Ziffer 1, aufgelistet. Wird eine Transaktion oder Transaktionsgruppe selektiert, werden die sich in einer Gruppe befindlichen Transaktionen im linken oberen Bereich in Abb. 7.5, versehen mit der Ziffer 2, dargestellt. Gehört dabei die gewählte Transaktion keiner Gruppe an, wird an dieser Stelle in dem Bereich lediglich die Transaktion selber angezeigt.

Im rechten oberen Bereich in Abb. 7.5, versehen mit der Ziffer 3, werden Informationen angezeigt, die für die Gruppierung von Transaktionen hilfreich sein können. Dazu gehören die Anzeige des sendenden und empfangenden betrieblichen Objekts, den betrieblichen Objekten jeweils zugeordnete AT sowie die identifizierten AIM der

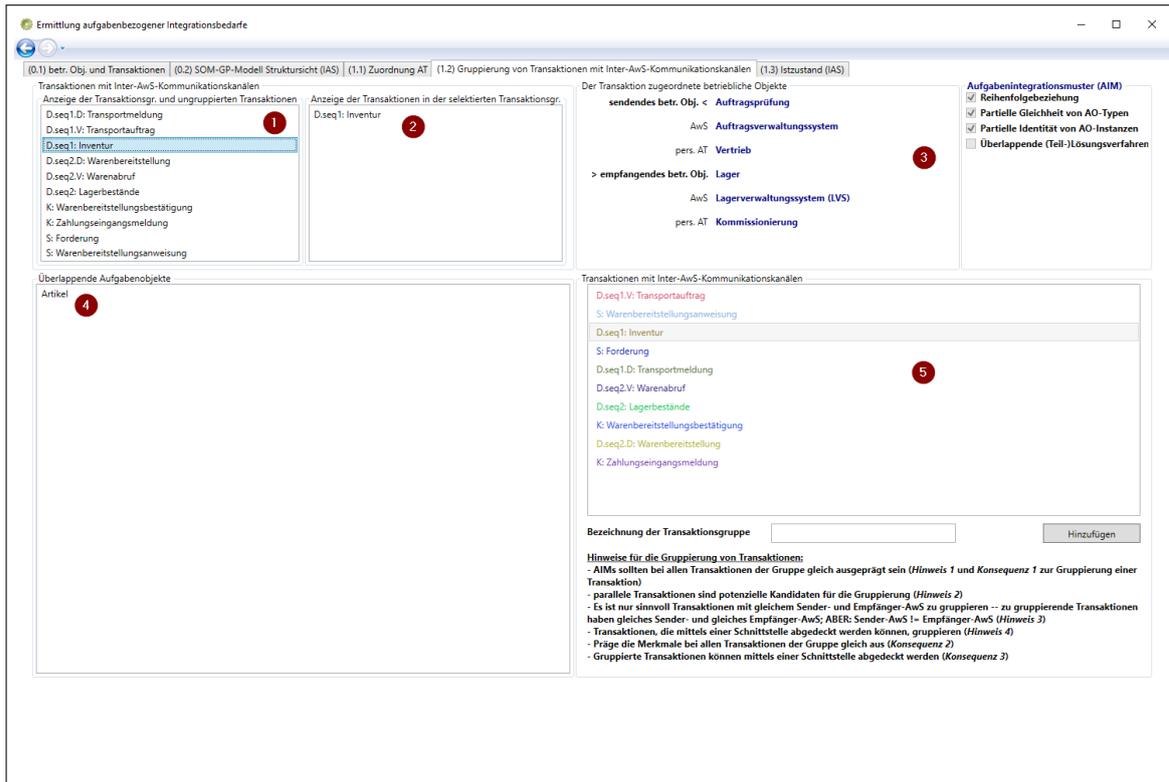


Abbildung 7.5: Gruppierung von Transaktionen mit Inter-AwS-Kommunikationskanälen mithilfe des Prototypen

selektierten Transaktion. Diese Informationen stammen aus Abschnitt 7.1.1 sowie aus der Tab. C.3. Ein weiterer Bereich, der informativen Zwecken dient, wird links unten in Abb. 7.5, versehen mit der Ziffer 4, dargestellt. Dabei werden überlappende AO der selektierten Transaktion oder Transaktionsgruppe aufgelistet.

Die Gruppierung von Transaktionen durch Vergabe eines eindeutigen Namens für alle zur Gruppe gehörenden Transaktionen sowie Vorschläge aus Abschnitt 6.3.1.3 zur Gruppierung von Transaktionen werden im rechten unteren Bereich in Abb. 7.5, versehen mit der Ziffer 5, dargestellt. Zur weiteren Unterstützung werden potenziell zu gruppierende Transaktionen in der gleichen Schriftfarbe gesetzt.

Die Ergebnisse der AT-Zuordnung können wiederum im IAS verzeichnet werden. Das Ergebnis wird in Abb. 7.6 in der Lenkungssicht auf den untersuchten GP dargestellt. Die Transaktionsautomatisierung ist in der Tab. C.2 dokumentiert. Analog zum Vorschlag in Abschnitt 6.3.1.2 werden in der Abb. 7.6 die AwS-Grenzen durch zusammenhängende Layer in unterschiedlichen Farben hervorgehoben. Abschließend können die gegenwärtig verwendeten Integrationskonzepte in einer eigenständigen Sicht, wie der von Krumbiegel (1997, S. 140-141) vorgeschlagenen Integrationssicht, oder, wie im Rahmen der vorliegenden Fallstudie, in einer Tabelle erfasst werden (Tab. 7.2). Dabei ist die Tab. 6.5 die Grundlage für die Tab. 7.2.

Tabelle 7.2: Integrierte AwS und die dabei verwendeten Integrationskonzepte im Rahmen der Istanalyse

Betriebliches Objekt	AwS	verwendetes Integrationskonzept
Vertriebslenkung	 Auftragsverwaltungssystem	 datenflussorientierte Funktionsintegration
Finanzwesen	 SoftM-Suite FI	
Versandlenkung	 Auftragsverwaltungssystem	 datenflussorientierte Funktionsintegration
Logistikdienstleister	 Fremdsystem	
Auftragsprüfung und Versandlenkung	 Auftragsverwaltungssystem	 datenflussorientierte Funktionsintegration
Lager	 LVS	
Logistikdienstleister	 Fremdsystem	 aufgabenträgerorientierte Funktionsintegration
Lager	 LVS	

7.2 Szenario: Einführung eines ERP-Systems

Der Güterdistributor plant, vorhandene AwS durch Einführung eines ERP-Systems zu ersetzen. Zur Prognose der Handlungsbedarfe fließt dies als Szenario in die Fallstudie ein. Ziel dieses Schrittes ist es, entstehende Integrationsbedarfe beim Szenario aufzuzeigen und eine geeignete Integrationsarchitektur für das Szenario zu entwerfen.

7.2.1 Szenarioorientierte Gestaltung des Geschäftsprozesses

Analog zu dem in Abschnitt 6.3 vorgestellten Vorgehensmodell werden für den Aufbau des Szenarios zunächst die AT im GP kartografiert. Ausgangspunkt hierfür ist das IAS aus der Istanalyse (Abb. 7.2). An dieser Stelle kann der GP modifiziert werden, wenn ersichtlich wird, dass sich Abläufe aufgrund der Änderungen auf AT-Ebene auf den GP auswirken, oder, wenn ein solches Szenario auch als Anlass für ein Re-Engineering des GP genutzt werden soll. Darüber hinaus kann der GP ebenso geändert werden, wenn die in Abschnitt 6.3.1.2 definierte Regel mit der Zuordnung der AT zu einem betrieblichen Objekt nicht eingehalten werden kann. Änderungen oder neu einzuführende betriebliche Objekte und Transaktionen sowie die sich daraus ergebenden überlappenden AO und identifizierte AIM sind im Prototypen an der entsprechenden Stelle für das Szenario zu hinterlegen – analog zu Abb. 7.3 im Rahmen der Istanalyse. Da bei dem Szenario am GP von der Istanalyse keine Änderungen vorgenommen werden, können auch die Zuordnung betrieblicher Objekte zu den Transaktionen sowie die dabei identifizierten AIM und überlappenden AO für dieses Szenario der Tab. C.3 entnommen werden.

In der Tab. 7.3 wird das Ergebnis der AT-Zuordnung für das vorliegende Szenario dargestellt. Die AT werden mithilfe des Prototypen für das Szenario erfasst. Dafür wird die in Abb. 7.4 dargestellte Eingabemaske verwendet.

Analog zu Abschnitt 7.1.2 für die Istanalyse werden auch für das Szenario die Transaktionen mit Inter-AwS-Kommunikationskanälen gruppiert. Da, wie eingangs erläutert, am GP keine Änderungen vorgenommen werden, kann die in Abschnitt 7.1.2 für die Abb. 7.5 beschriebene Gruppierung für das Szenario übernommen werden.

7.2.2 Erhebung der Freiheitsgrade beim Entwurf der Integrationsarchitektur

Um die Erstellung des Sollkonzepts unterstützen zu können, sind die Freiheitsgrade zu ermitteln. Dazu sind, wie in Abschnitt 6.3.2.2 beschrieben, sowohl auf Ebene des Unternehmensplans als auch auf Ressourcenebene etwaige Restriktionen ausfindig zu machen. Auf GP-Ebene wird ermittelt, ob Inter-AwS-Kommunikationskanäle für

Tabelle 7.3: Betriebliche Objekte und die ihnen zugeordneten AT im Szenario: Einführung eines ERP-Systems

Betriebliches Objekt	ausführende Organisationseinheit	unterstützendes AwS
Vertriebslenkung	 Vertrieb	 SAP ERP
Auftragsprüfung	 Vertrieb	 SAP ERP
Versandlenkung	 Versand	 SAP ERP
Lager	 Kommissionierung	 LVS
Logistikdienstleister	 Disponent	 Fremdsystem
Finanzwesen	 Finanzen	 SAP ERP

Transaktionen aufgrund der Anforderungen aus Sicht der jeweils involvierten Aufgaben maschinell zu unterstützen sind.

7.2.2.1 Ebene des Unternehmensplans

Von Seiten des Unternehmensplans ist vorgegeben, dass an der Aufbauorganisation und somit an der Zuordnung der Aufgaben zu personellen AT keine Änderungen vorgenommen werden dürfen. Des Weiteren wird in der Organisation bislang keine EAI-Lösung verwendet. Die Einführung einer solchen wird erlaubt.

7.2.2.2 Geschäftsprozessebene

Nachfolgend wird die Ableitung von Handlungsempfehlungen und die damit verbundene Anwendung der Merkmale beispielhaft anhand der Transaktion *D.seq1: Inventur* verdeutlicht. Dazu werden die Merkmale aus Abschnitt 5.4.5 angewendet:

- Periodizität:** Zur Einordnung des untersuchten GP in die Skala wird untersucht, wie oft dieser im Mittel pro Periode durchgeführt wird. Als Periode wird ein Kalendertag definiert. Für die Erhebung werden Erfahrungswerte aus der Vergangenheit herangezogen. Es wurde festgestellt, dass der GP im Durchschnitt täglich fünfzigmal durchgeführt wird ($\bar{p} = 50$ mal pro Tag), d. h. es gehen an jedem Arbeitstag durchschnittlich 50 Aufträge ein. Die Entscheidungsträger ordnen diese Ausprägung der Klasse $P_{(hoch)}$ zu, da dieser GP in der Organisation derjenige ist, der am häufigsten ausgeführt wird.
- Größe:** Zur Einordnung der durchschnittlichen Größe, die der zu übertragende AO-Typ annehmen kann, wird aus den bekannten AO-Instanzen der Mittelwert

gebildet. Ausgangsbasis der Messung für die Größe der zu übertragenden AO-Instanzen sind die überlappenden Attribute der AO der über die AT zu integrierenden Aufgaben. Im Ergebnis wurde festgestellt, dass die AO-Instanzen im Mittel den Wert $\bar{g} = 1$ Kilobyte annehmen. Die Einordnung erfolgt in die Klasse $G_{(niedrig)}$, da andere überlappende AO-Instanzen existieren, die bedeutend größere Werte annehmen können.

- **Intensität:** Zur Einordnung der Intensität der Durchführungen der hier untersuchten Transaktion *D.seq1: Inventur* wird aus den bekannten Vergangenheitswerten der Mittelwert gebildet. Dieser liegt bei $\bar{i} = 5$ Ausführungen pro Durchführung eines GP. Dieser Wert sagt aus, dass ein Auftrag durchschnittlich fünf Positionen enthält und für diese jeweils separate Anfragen beim Lager nach Beständen gestellt werden. Von den Entscheidungsträgern wird dieser Wert der Klasse $I_{(moderat)}$ zugeordnet, da sowohl Transaktionen bekannt sind, die eine bedeutend höhere Intensität aufweisen als auch solche, die eine niedrigere Intensität aufweisen.
- **Anpassbarkeit:** Um die Transaktion *D.seq1: Inventur* in die Skala einordnen zu können, wurden die Eigenschaften von Standard-, Routine- und Nicht-Routine-Transaktionen und deren Ausprägung für diese Transaktion erhoben (Tab. 5.7). Bei der Aufnahme der Eigenschaften wurde festgestellt, dass der Lösungsweg algorithmisch beschrieben und die Planbarkeit als hoch eingestuft werden kann. Somit ist die Transaktion als Standardtransaktion zu kennzeichnen und der Klasse $A_{(niedrig)}$ zuzuordnen.
- **Nutzungsdauer:** Für die Transaktion der vorliegenden Fallstudie ist gegenwärtig kein Ende der Nutzungsdauer absehbar. Er soll in seiner vorgestellten Form dauerhaft genutzt werden. Für die zu implementierende Integrationsarchitektur liegt die geplante Nutzungsdauer bei $n = 10$ Jahren. Die Nutzungsdauer ist aufgrund der Empfehlung von Abschnitt 5.4.5.5 in die Klasse $N_{(hoch)}$ einzuordnen.

In der Tab. 7.4 werden die Ausprägungen der Merkmale, basierend auf dem Vorschlag von Tab. 5.8, für die Transaktion *D.seq1: Inventur* im Überblick dargestellt.

Für die Transaktion ist nun zu ermitteln, ob sie maschinell unterstützt werden soll oder nicht. Dazu werden die Regeln von Abschnitt 5.5 herangezogen. Auf die hier vorgestellten Ausprägungen trifft die *Regel 2* aus Abschnitt 5.5 zu. Die dazugehörige Empfehlung lautet, die Transaktion maschinell zu unterstützen.

Für die anderen Transaktionen mit Inter-AwS-Kommunikationskanälen sind analog zur beispielhaft untersuchten Transaktion Unterstützungsbedarfe abzuleiten. In Tab. C.4 werden die Ausprägungen der Merkmale nach Durchführung der Analyse einschließlich der aus Abschnitt 5.5 zutreffenden Regeln dargestellt (die Spalten unterhalb von

Tabelle 7.4: Ausprägung der Merkmale für die Transaktion *V: AuffPrf*

Kategorie	Merkmal	Ausprägung		
		niedrig	moderat	hoch
Leistungsanforderungen	Periodizität des GP			50 mal pro Tag
	Größe einer zu übertragenden AO-Instanz	1 Kilobyte		
	Intensität bei der Durchführung einer Transaktion		5	
Flexibilitätsanforderungen	Anpassbarkeit der Transaktion	niedrig		
	Nutzungsdauer der Integrationsarchitektur			langfristig

Merkmalsausprägung auf GP-Ebene). Die Regeln wurden nach Erhebung der Ausprägungen für jede Transaktion mit einem Inter-AwS-Kommunikationskanal mithilfe des in Abb. 5.1 vorgestellten Ausschnitt aus dem entwickelten Prototypen abgeleitet.

7.2.2.3 Ressourcenebene

Auf AT-Ebene hat sich herausgestellt, dass die von der Kopplung betroffenen AwS gegenwärtig noch keine Schnittstellen zueinander zur Integration der korrespondierenden Aufgaben haben. Die Softwarehersteller können ihre AwS jedoch um die geforderte Integrationsfähigkeit erweitern, indem sie Schnittstellen in den AwS implementieren. Das Know-how zur Realisierung der Schnittstellen ist ebenfalls vorhanden. Die Implementierungskosten werden auf 3.000 € geschätzt.

7.2.3 Ermittlung der Soll-Automatisierung für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen

Für die aus Abschnitt 7.2.1 hervorgehenden Transaktionen und gruppierten Transaktionen mit Inter-AwS-Kommunikationskanälen wird die erforderliche Soll-Automatisierung, wie in Abschnitt 6.3.2.3 vorgeschlagen, ermittelt. Dieser Schritt wird exemplarisch anhand der Transaktion *D.seq1: Inventur* sowie der dazugehörigen Sendeaufgabe aus Sicht des betrieblichen Objekts *Auftragsprüfung* unter Zuhilfenahme des Prototypen erläutert (Abb. 7.7).

In Abb. 7.7 wird der Ausschnitt aus dem Prototypen zur Unterstützung bei der Erstellung des Sollkonzepts für die Transaktionen mit Inter-AwS-Kommunikationskanälen dargestellt. Im linken oberen Bereich werden die Transaktionen und Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen aufgelistet (Ziffer 1). Im linken mittleren Bereich werden die bei einer Transaktion überlappenden AO dargestellt (Ziffer 2). Sie wurden in Abschnitt 7.2.1 ermittelt und sind aus der Tab. C.3 zu entnehmen. Im rechten oberen Bereich in Abb. 7.7 werden zu Informationszwecken das sendende und das empfangende betriebliche Objekt, den betrieblichen Objekten jeweils zugeordnete AT sowie die identifizierten AIM der selektierten Transaktion dargestellt (Ziffer 3). Im mittleren Bereich in Abb. 7.7 stehen die freiheitsgradbeschränkenden Faktoren, wie sie in Tab. 6.8 zusammengefasst und für die vorliegende Fallstudie in Abschnitt 7.2.2 erhoben wurden (Ziffer 4).

Der Ist-Zustand der Automatisierung von den Transaktionen mit Inter-AwS-Kommunikationskanälen wird analog zum Vorschlag in Abschnitt 6.3.2.3 mithilfe des entwickelten Prototypen erfasst. Dieser unterstützt die Aktion, indem, wie in Abb. 7.7 dargestellt, im linken unteren Bereich der Ist-Zustand eingepflegt wird (Ziffer 5). Die exemplarische Transaktion *D.seq1: Inventur* ist nicht automatisiert, weshalb bei der Ist-Automatisierung „nicht automatisiert“ vermerkt wird. Daneben wird auch das Ergebnis für die Soll-Automatisierung der Transaktion erfasst (Ziffer 5).

Die weiteren in Abb. 7.7 dargestellten Transaktionen und Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen sind nach dem gleichen Verfahren zu bewerten. Die im Rahmen der Sollkonzeption erhobenen Werte sind der Tab. C.4 zu entnehmen.

Die Ergebnisse dieses Schrittes können in die Lenkungssicht des untersuchten GP übertragen werden. Somit entsteht die Integrationsarchitektur, wie in Abb. 7.8 dargestellt. Sie bildet das Gestaltungskonzept für das darauf aufbauende Implementierungsprojekt.

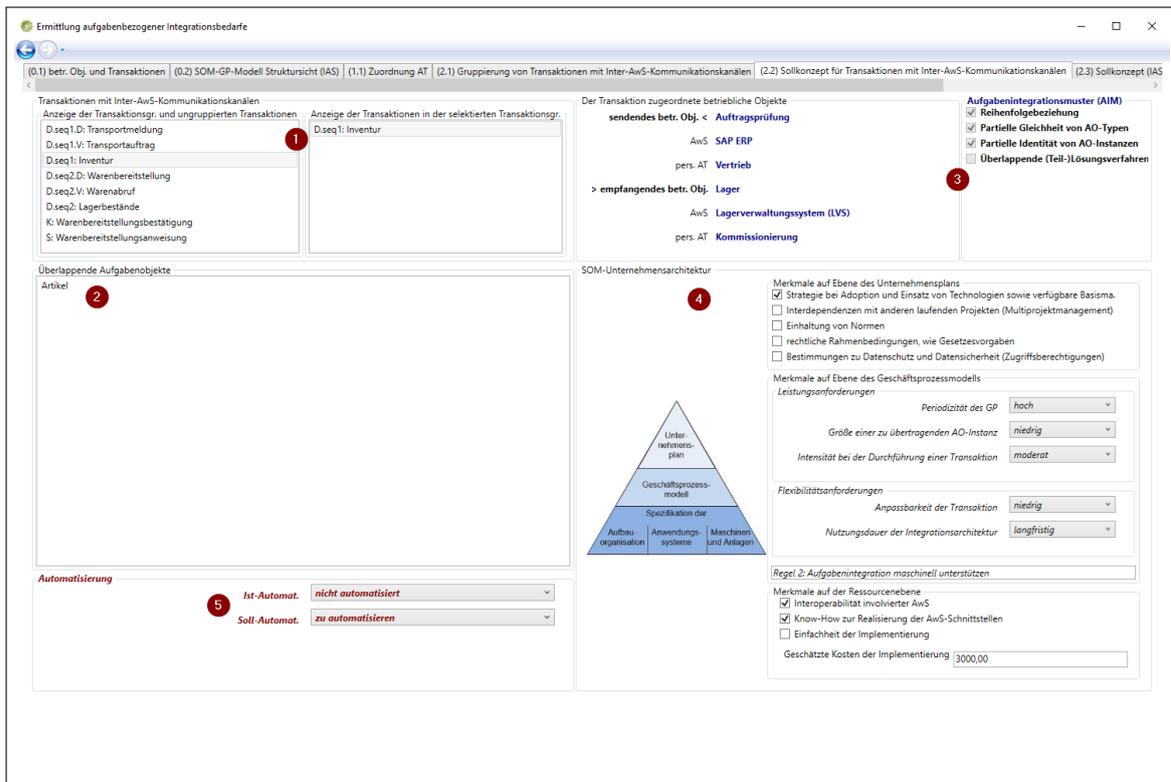


Abbildung 7.7: Erstellung eines Sollkonzepts für die Transaktionen mit Inter-AwS-Kommunikationskanälen mithilfe des Prototypen

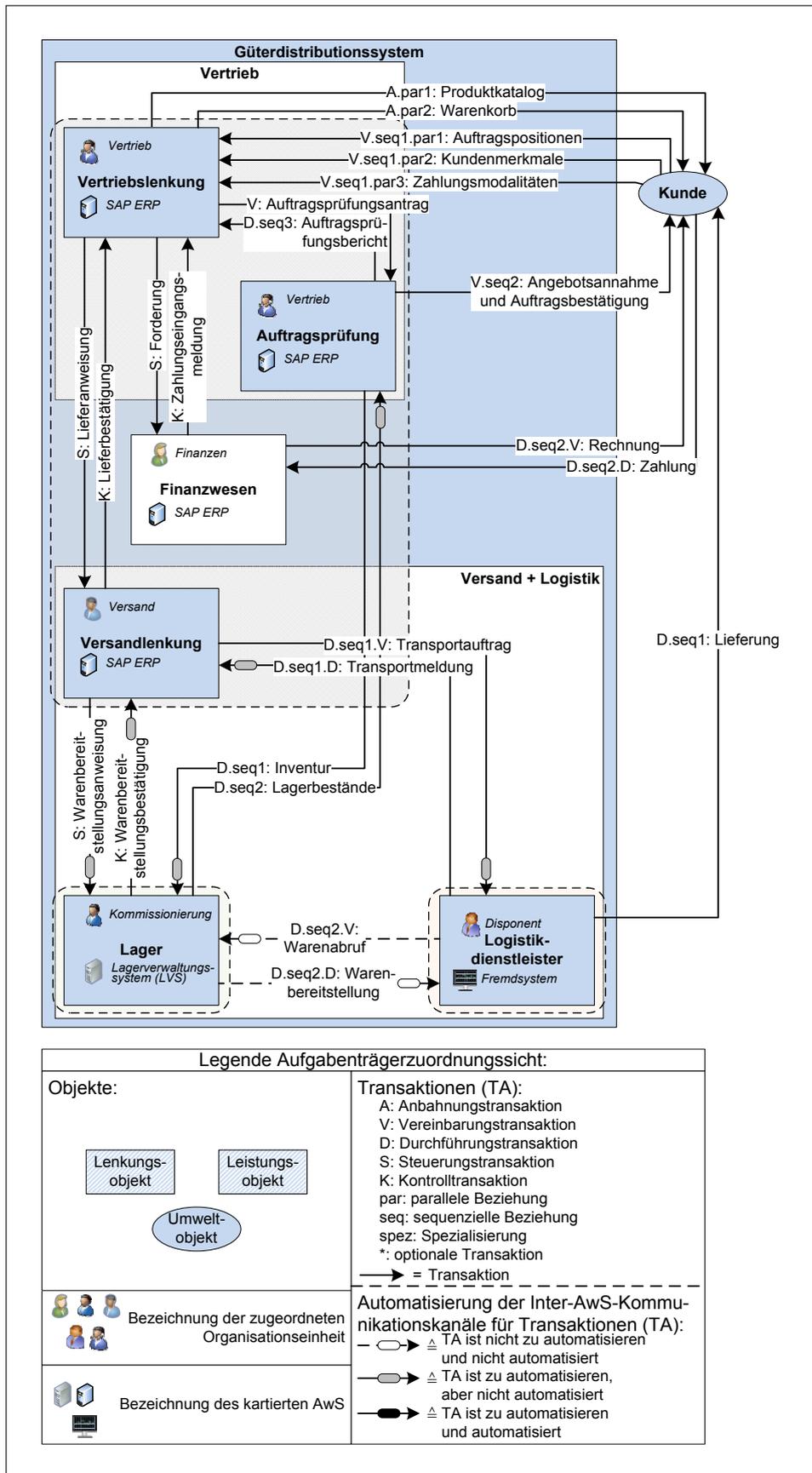


Abbildung 7.8: Integrationsarchitektur für das Szenario

7.2.4 Ableitung der zu verwendenden Integrationskonzepte für die Inter-AwS-Integrationen

Im letzten Schritt der Gestaltung des Szenarios werden die Integrationskonzepte für die AwS-Integrationen abgeleitet. Dazu werden zunächst die Integrationsziele festgelegt (Abschnitt 7.2.4.1) und anschließend die den Anforderungen gerecht werdenden Integrationskonzepte ermittelt (Abschnitt 7.2.4.2).

7.2.4.1 Festlegung der Integrationsziele für das Szenario

Vorab werden die für das Szenario gültigen Integrationsziele definiert. Diese sind insbesondere bei der Ermittlung der Integrationskonzepte für die Inter-AwS-Integrationen in den Szenarien heranzuziehen. Dabei müssen die nachfolgend ermittelten Integrationskonzepte die in der Tab. 7.5 festgelegten Integrationsziele einhalten. Die Tab. 7.5, stellt eine Variante der Tab. 6.9 dar.

7.2.4.2 Ermittlung der Integrationskonzepte für die Inter-AwS-Integrationen

Zur Umsetzung der Integrationsarchitektur sind für die Inter-AwS-Integrationen geeignete Integrationskonzepte zu wählen. Wie in Abschnitt 6.3.2.4 erläutert, sind dabei die Integrationsziele aus Tab. 7.5, die AIM aus Abschnitt 7.1.1 (Abb. 7.3) sowie die Freiheitsgrade aus Abschnitt 7.2.2 zu berücksichtigen.

In Abb. 7.9 wird dargestellt, wie der entwickelte Prototyp bei der Ableitung der Integrationskonzepte für die Inter-AwS-Integrationen im vorliegenden Szenario unterstützend eingesetzt werden kann. Dabei werden in Abb. 7.9 links oben die Inter-AwS-Kommunikationskanäle, für die ein Delta besteht, aufgelistet (Ziffer 1). In Abhängigkeit zur ausgewählten Transaktion werden im oberen mittleren Bereich Details zu den sendenden und empfangenden betrieblichen Objekten dargestellt (Ziffer 2). Darunter werden zu Informationszwecken die in Abb. 7.3 zu jeder Transaktion oder Transaktionsgruppe mit Inter-AwS-Kommunikationskanälen hinzugefügten überlappenden AO aufgelistet (Ziffer 3). Im linken unteren Bereich in Abb. 7.9 wird die Tab. 6.10 eingebunden (Ziffer 4). Dabei werden die AIM so, wie sie in Abb. 7.3 bei den jeweiligen Transaktionen identifiziert wurden, markiert (Ziffer 5). Daneben ist im rechten Bereich die Beherrschung der Integrationsmerkmale durch die AwS-Unterstützung so, wie sie in Tab. 7.5 erhoben wurde, anzugeben (Ziffer 6). Bei der Analyse der in Tab. 7.5 ausgeprägten Integrationsmerkmale kann festgestellt werden, dass lediglich das Integrationsmerkmal Kommunikationsstruktur durch die beteiligten AwS unterstützt werden soll, daher wird im Prototyp das Integrationsmerkmal Kommunikationsstruktur selektiert (Ziffer 6 in Abb. 7.9).

Tabelle 7.5: Festlegung der Integrationsziele für das Szenario

Gruppe	Integrationsmerkmal	Ausprägung	AwS-Unterstützung
Struktur	Datenredundanz	<i>Redundante Datenhaltung wird akzeptiert.</i>	○
	Funktionsredundanz	<i>Redundant verwendete Funktionen sind zu ermitteln, die zukünftig zu verwenden festzulegen und die übrigen zu deaktivieren bzw. deren Einsatz durch Richtlinien zu unterbinden.</i>	○
	Kommunikationsstruktur	<i>AT sind bei Bedarf durch Schaffung von Kommunikationskanälen miteinander zu verknüpfen. Die Erfordernis von Kommunikationskanälen und den darüber auszutauschenden Nachrichtentypen sind aus dem GP heraus abzuleiten.</i>	●
Verhalten	semantische Integrität	<i>AT dürfen nur mit gültigen Tupeln einer Systemrelation arbeiten. Aus diesem Grund sind gültige Tupel zu definieren.</i>	○
	operationale Integrität	<i>Inkonsistente Datenbestände sind durch die Implementierung geeigneter Maßnahmen zu unterbinden.</i>	○
	Vorgangsteuerung	<i>Die Vorgangsteuerung ist durch die Implementierung geeigneter Maßnahmen zu unterstützen.</i>	○

In Abhängigkeit der AIM und der Beherrschung der Integrationsmerkmale durch die AwS-Unterstützung werden links unten Hinweise darüber ausgegeben, welche Integrationskonzepte sich anbieten und welche Einschränkungen sowie welche Besonderheiten bei den gewählten Integrationskonzepten gegebenenfalls zu beachten sind (Ziffer 7). Unter Abgleich der Tab. 6.10 bleibt folglich Wahlfreiheit zwischen den beiden Integrationskonzepten datenflussorientierte Funktionsintegration und Objektintegration. In der Organisation wird bislang keine EAI-Lösung verwendet. Bei den in Abschnitt 7.2.2 erhobenen Freiheitsgraden wird im Unternehmensplan vorgegeben, dass mit diesem Projekt auch die Einführung eines neuen IT-Systems erlaubt wird. Durch diese Erlaubnis wird das Integrationskonzept Objektintegration gewählt (Ziffer 8), die durch eine einzuführende EAI-Lösung zu realisieren ist und mit der damit einhergehend vorhandene Punkt-zu-Punkt-Verbindungen abgelöst werden sollen.

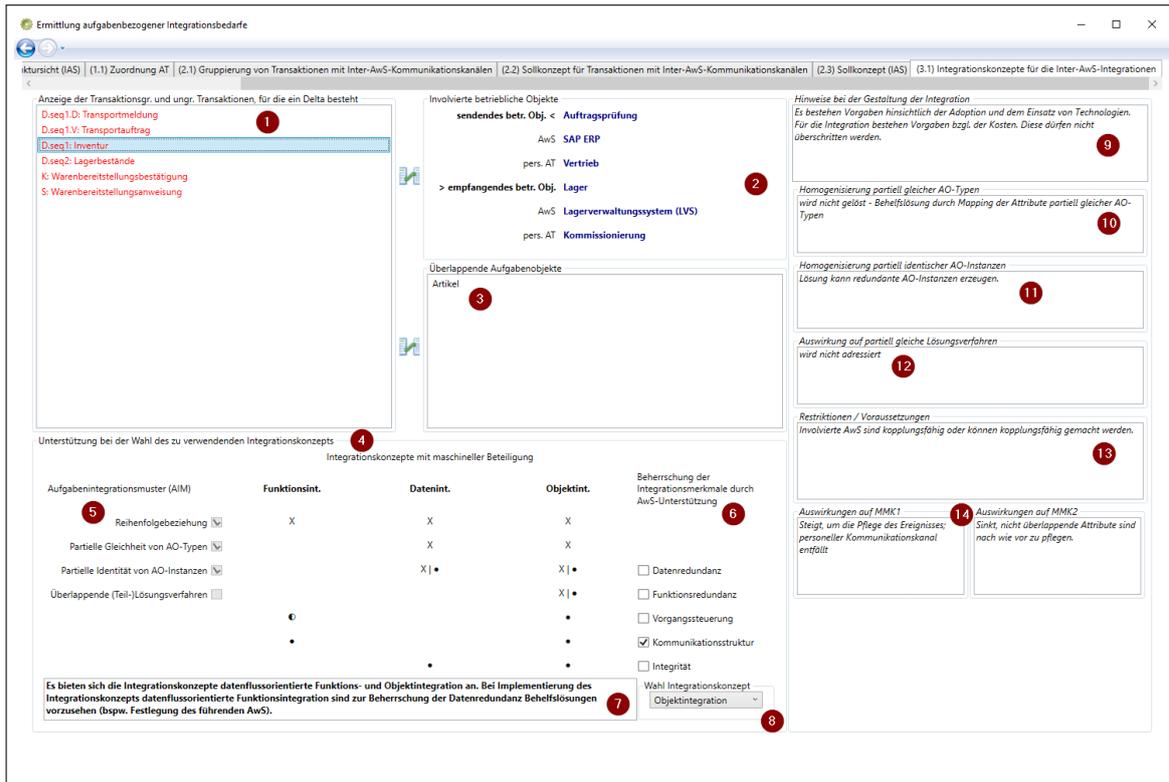


Abbildung 7.9: Ermittlung der Integrationskonzepte für die Inter-AwS-Integrationen mithilfe des Prototypen

Im rechten oberen Bereich in Abb. 7.9 werden die Restriktionen aus Abb. 7.7 ausgegeben (Ziffer 9). So wurde bspw. für die Transaktion *D.seq1: Inventur* in Abb. 7.7 festgelegt, dass Vorgaben hinsichtlich der einzusetzenden Technologien sowie der Kosten bestehen.

Im darunterliegenden Bereich in Abb. 7.9 werden mit der Wahl des Integrationskonzepts (Ziffer 8) die in der Tab. 6.11 (Abschnitt 6.3.2.4) in den jeweiligen Spalten hinterlegten Hinweise ausgegeben. Auf Grundlage dieser Hinweise kann der Systemgestalter gegebenenfalls die Wahl des Integrationskonzepts revidieren oder den GP mithilfe der in Abschnitt 6.3.2.5 beschriebenen Optionen anpassen.

Für die im Beispielfall betroffene Inter-AwS-Integration (Ziffer 2) wurde die Objektintegration (Ziffer 8) gewählt. In dem entsprechenden Bereich der Abb. 7.9 werden deshalb die in den jeweiligen Spalten zum Integrationskonzept Objektintegration hinterlegten Hinweise in Tab. 6.11 ausgegeben. Dies betrifft die Homogenisierung partiell gleicher AO-Typen (Ziffer 10 in Abb. 7.9), Homogenisierung partiell identischer AO-Instanzen (Ziffer 11 in Abb. 7.9), Auswirkung auf partiell gleiche Lösungsverfahren (Ziffer 12 in Abb. 7.9), Restriktionen / Voraussetzungen (Ziffer 13 in Abb. 7.9; Hinweise stammen aus dem Abschnitt 6.3.2.4), Auswirkung auf Mensch-Maschine-

Kommunikation₁ sowie Mensch-Maschine-Kommunikation₂ (Ziffer 14 in Abb. 7.9; Hinweise stammen aus Tab. 6.11).

Nach diesem exemplarisch für die Transaktion *D.seq1: Inventur* beschriebenen Vorgehen, sind für die restlichen Deltas (Ziffer 1 in Abb. 7.9) geeignete Integrationskonzepte abzuleiten. Die für das Szenario abgeleiteten Integrationskonzepte für die AWS-Integrationen sind der Tab. 7.6 zu entnehmen.

Für die Transaktionen, die in der Abb. 7.8 als nicht maschinell zu unterstützen gekennzeichnet wurden (*D.seq2.V: Warenabruf*, *D.seq2.D: Warenbereitstellung*), gilt das Integrationskonzept aufgabenträgerorientierte Funktionsintegration (Abschnitt 3.3.2.1).

Tabelle 7.6: Abgeleitete Integrationskonzepte für das Szenario

Betriebliches Objekt	AwS	verwendetes Integrationskonzept
Versandlenkung	 SAP ERP	 Objektintegration
Logistikdienstleister	 Fremdsystem	
Auftragsprüfung und Versandlenkung	 SAP ERP	 Objektintegration
Lager	 LVS	
Logistikdienstleister	 Fremdsystem	 aufgabenträgerorientierte Funktionsintegration
Lager	 LVS	

7.3 What-If-Analyse für den Güterdistributor

In der Abb. 7.10 wird die Auswertung der What-If-Analyse für den Güterdistributor dargestellt. Dazu werden im oberen Bereich für die Istanalyse und die Szenarien die Informationssystemarchitekturen mit der Automatisierung der Transaktionen mit Inter-AwS-Kommunikationskanälen dargestellt. Im unteren Bereich werden diese aufgelistet. Darüber hinaus werden bei den Szenarien die Summen der geschätzten Kosten der Implementierung angezeigt. Im Rahmen der vorliegenden Fallstudie wird im rechten Bereich in Abb. 7.10 das in Abschnitt 7.2 vorgestellte Szenario dargestellt. Der rechte Bereich in Abb. 7.10 würde im Prototypen erweitert, wenn weitere Szenarien aufgenommen werden würden.

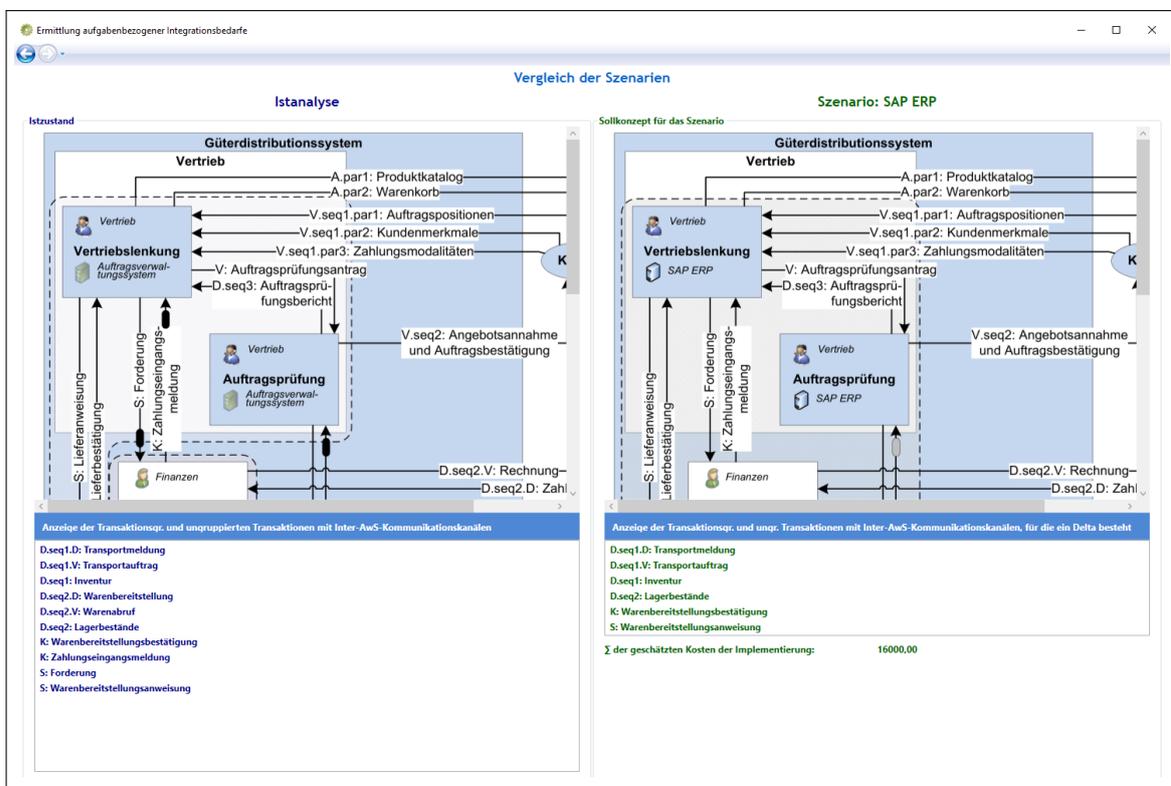


Abbildung 7.10: Ergebnis der What-If-Analyse für den Güterdistributor

7.4 Zusammenfassung und Diskussion

Im vorliegenden Kapitel wurde die in Kapitel 6 entwickelte Methode beispielhaft anhand einer Fallstudie verdeutlicht. Mithilfe der Methode wurde zunächst der Ist-Zustand aufgenommen (Abschnitt 7.1). Dabei war der Ausgangspunkt der Methoden-anwendung der GP (Abschnitt 7.1.1). Anschließend wurde in Abschnitt 7.2 ein exemplarisches Szenario gebildet. Für dieses wurde mithilfe der Methode eine Integrationsarchitektur zur Unterstützung der Integrationsbedarfe abgeleitet. Dazu wurde die Soll-Automatisierung für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen ermittelt – Ergebnis von Abschnitt 7.2.3. Darauf aufbauend wurden in Abschnitt 7.2.4 die Integrationskonzepte für die Inter-AwS-Integrationen unter Berücksichtigung der Freiheitsgrade (Abschnitt 7.2.2) und der in Abschnitt 7.2.4.1 festgelegten Integrationsziele abgeleitet. Abschließend können, wie in Abschnitt 7.3 aufgezeigt, die Ergebnisse der Istanalyse sowie der Szenarien miteinander verglichen werden. Nach der Auswahl eines Szenarios ist selbiges als Blueprint, mithilfe dessen unter anderem Nachrichtentypen als Grundlage für ein Implementierungsprojekt abgeleitet werden können, heranzuziehen. In einem darauf aufbauenden Integrationsprojekt ist das Ziel, die abgeleiteten Gestaltungsmaßnahmen zu implementieren – im Rahmen der Fallstudie werden diese für das vorgestellte Szenario in Abschnitt 7.2.3 und Abschnitt 7.2.4.2 definiert. Zur Unterstützung der Inter-AwS-Integrationen sind Kopplungssystemarchitekturen erforderlich. Beispielhafte Entwürfe von Kopplungssystemen sind nicht Gegenstand dieses Kapitels, können aber beispielsweise mittels der OASYS-Methodik konstruiert werden. Die Konstruktion von Kopplungssystemen wird in Eckert et al. (2004) sowie Eckert et al. (2005) demonstriert. Weiterhin ist auch die konkrete Implementierung der Gestaltungsmaßnahmen nicht Gegenstand dieses Kapitels. Wie diese beispielhaft umgesetzt und dokumentiert werden können, wird in Kiehl (2012, S. 79-94) ausführlich dargestellt.

8 Zusammenfassung und Ausblick

8.1 Zusammenfassung der Ergebnisse	173
8.2 Schlussbetrachtung und Ausblick	175

In folgendem Kapitel werden in Abschnitt 8.1 die Kernergebnisse des vorliegenden Beitrags zusammengefasst und in Abschnitt 8.2 ein Ausblick auf weitere Forschungsbedarfe sowie mögliche Weiterentwicklungen in Bezug auf aktuelle Entwicklungen gegeben.

8.1 Zusammenfassung der Ergebnisse

Gegenstand dieser Arbeit ist die Fragestellung, wie Integrationsbedarfe durch Änderungen auf GP- oder AwS-Ebene methodisch aufgedeckt und durch die Findung einer geeigneten Integrationsarchitektur überwunden werden können (Abschnitt 1.2). Wie in der Einleitung in Abschnitt 1.1 beschrieben, ist der Auslöser der Untersuchung das in der betrieblichen Praxis häufig auftretende Problem, dass Modifikationen auf GP- oder AT-Ebene vorgenommen werden sollen und dabei geprüft werden soll, wie sich diese auf die Integration auswirken.

Zur Beantwortung der Fragestellung wurden zunächst in den Kapiteln 2 und 3 die in der Arbeit verwendeten Begrifflichkeiten eingeführt. Dazu wurde in Kapitel 2 die Unternehmensarchitektur als übergeordneter Rahmen vorgestellt und anhand dieser die spezifischen Grundlagen erläutert. In Kapitel 3 wurde die Gestaltung der Integration in Informationssystemen als Teilaufgabe bei der Systementwicklung aufgefasst und dazu eine Terminologie aufgebaut, die sich auf die beiden in dieser Arbeit zentral verwendeten Begriffe Integrationsbedarf und Integrationsarchitektur bezieht.

In Kapitel 4 wurden Methoden anhand von Anforderungen (Abschnitt 4.3) evaluiert. Dabei handelt es sich um Methoden, die die Gestaltung der Integration in Informationssystemen zum Gegenstand haben und im Rahmen einer strukturierten Literaturrecherche gefunden wurden. Dabei wurde festgestellt, dass Methoden, mithilfe derer Integrationsarchitekturen entworfen werden können, sich vorrangig auf die Gestaltung der AwS-Integration fokussieren und deshalb Gestaltungspotenziale verschenken, da sie den personellen AT häufig nur unzureichend und nicht immer als Element zur Gestaltung geeigneter Integrationsarchitekturen berücksichtigen. Darüber hinaus ist

nicht bei allen untersuchten Ansätzen der GP der Ausgangspunkt bei der Gestaltung der Integrationsarchitektur (Abschnitt 4.5, Tab. 4.10). Stattdessen ist bei diesen oftmals die AT-Ebene der Ursprung für Integrationsbedarfe. Daher ist eine Integration oft technisch anstatt fachlich motiviert und kann somit an den eigentlichen Bedarfen vorbei führen.

Von dieser identifizierten Forschungslücke ausgehend wurden in Kapitel 5 Merkmale herausgearbeitet, anhand derer die Entscheidungsfindung bei der Gestaltung der Aufgabenintegration über AT unterstützt werden kann. Ausgangspunkt hierfür war ebenfalls die Durchführung einer strukturierten Literaturanalyse, mithilfe derer eine umfangreiche Basis an Merkmalen gebildet wurde. Ausgehend von dieser Basis wurden die relevanten Merkmale mittels argumentativ-deduktiver Selektion reduziert. Der final gebildete Merkmalskatalog wurde in Abschnitt 5.4 präsentiert. Darauf aufbauend wurde eine auf Pragmatik ausgerichtete Vorgehensweise vorgestellt, anhand derer mittels eines Regelwerks Handlungsempfehlungen zur Unterstützung der Entscheidungsfindung bei der Aufgabenintegration über AT abgeleitet werden können (Abschnitt 5.5).

In Kapitel 6 wurde eine modellgestützte Methode zur korrelativen Konstruktion der Integration von Aufgaben über die Ressourcenebene durch simultane Gestaltung entwickelt. Grundlage dieser ist die SOM-Methodik. Kern des konstruierten Vorgehensmodells (Abschnitt 6.3, Abb. 6.7) ist die What-If-Analyse. Zur Durchführung der Methode wird zunächst der Ist-Zustand aufgenommen (Abschnitt 6.3.1). Anschließend werden Szenarien gebildet (Abschnitt 6.3.2). Dabei wird jeweils unter Zuhilfenahme der in Abschnitt 5.5 vorgestellten Merkmale und Handlungsempfehlungen der Bedarf nach maschineller Unterstützung bei der Aufgabenintegration bewertet und anschließend passende Integrationskonzepte abgeleitet (Abschnitt 6.3.2.4). Durch die geschilderte Vorgehensweise entsteht unter Berücksichtigung personeller AT für jedes entwickelte Szenario eine geeignete Integrationsarchitektur. Die Ausführungen in Kapitel 6 wurden anhand eines Fallbeispiels aus dem Hochschulbereich verdeutlicht. Durch die entwickelte Methode konnte die in Abschnitt 4.5 identifizierte Forschungslücke geschlossen und das in Abschnitt 1.2 definierte pragmatische Erkenntnisziel dieser Arbeit in Form der entwickelten Methode erreicht werden.

Gegenstand von Kapitel 7 war die exemplarische Anwendung der Methode im Rahmen einer Fallstudie für einen Güterdistributor. Dabei wurde durch Unterstützung der Methode und mithilfe eines im Rahmen der vorliegenden Arbeit entwickelten Prototypen untersucht, inwieweit Handlungsbedarfe nach einer Änderung auf AT-Ebene bei der Integration von Aufgaben existieren und wie diese überwunden werden können. Dadurch wurde zum einen die in Kapitel 6 vorgestellte Methode beispielhaft instanziiert und zum anderen die Anwendbarkeit demonstriert.

Zusammenfassend bietet die vorliegende Arbeit einen Lösungsvorschlag für einige der von Vogler (2006, S. 24-29) im Rahmen der Integration identifizierten Probleme. So wird ihrer Forderung nach einem methodischen Vorgehen beim Entwurf der Integrationsarchitektur durch die Methodenentwicklung nachgekommen (Kapitel 6). Die Einbindung der Kartografie ermöglicht die Visualisierung gegenwärtiger und geplanter Integrationsbeziehungen (Abschnitt 6.2.2) und somit das Aufdecken von Konsequenzen aus den Änderungen auf GP- oder AT-Ebene. Durch die Orientierung am GP werden mithilfe der Methode die Integrationsbedarfe fachlich motiviert und den Bedarfen gerecht werdende Integrationsarchitekturen konzipiert.

Das Problem der Skalierung wird durch Vereinfachung gelöst, indem lediglich Inter-AwS-Kommunikationskanäle betrachtet und gestaltet werden. Dem liegt die Denkweise zugrunde, dass die Intra-AwS-Kommunikationskanäle bereits im Rahmen der Einführung des jeweiligen AwS betrachtet werden. Daneben werden Transaktionen klassifiziert und verdichtet (Abschnitt 6.3.1.2 für die Istanalyse und Abschnitt 6.3.2.1 für die Szenarienbildung). Dazu werden die zu übertragenden AO sequenziell (*seq*) oder parallel (*par*) ablaufender Transaktionen miteinander verglichen und bei Bedarf gruppiert. Der Kerngedanke hierzu ist, dass überlappende AO zu einem Nachrichtentyp zusammengefasst werden können, infolge dessen weniger Schnittstellen auf Ressourcenebene zu implementieren sind und deshalb der Implementierungsaufwand verringert werden kann. Darüber hinaus werden weitere Hinweise zur Gruppierung von Transaktionen in Abschnitt 6.3.1.3 gegeben.

Anwender, definiert in Abschnitt 1.5, erhalten ein handlungsorientiertes Hilfsmittel in Form der entwickelten Methode. Durch ihre Anwendung können die individuellen Bedürfnisse und Ressourcen der jeweiligen Organisation sowie die Anforderungen an die Integration aus Sicht der Aufgaben berücksichtigt werden. Zudem erhalten sie Blueprints, die als Diskussionsgrundlage für die Ableitung der zu übertragenden Nachrichtentypen herangezogen werden können.

8.2 Schlussbetrachtung und Ausblick

Die im Rahmen dieser Arbeit gefundenen Merkmale (Kapitel 5), die entwickelte Methode (Kapitel 6) sowie die vorgeschlagene Verwendung der Merkmale in Bezug auf die Ableitung von Handlungsempfehlungen (Abschnitt 6.3.2.2) bieten verschiedene Anknüpfungspunkte.

Forschungsbedarf besteht in der Validierung der in dieser Arbeit gewonnenen Erkenntnisse zu den Merkmalen und den Handlungsempfehlungen. So kann die Signifikanz der in Abschnitt 5.5 vorgestellten Merkmale und Handlungsempfehlungen in Rahmen empirischer Arbeiten untersucht werden. Dabei ist zu prüfen, ob in Integrationspro-

jekten die Gültigkeit der Handlungsempfehlungen bestätigt werden kann. Auch ein Entscheidungsbaumlernen, das von den rückwirkend erhobenen Ausprägungen aus prüft, ob die Regel mit der dazugehörenden Handlungsempfehlung bestätigt werden kann, kann dazu unterstützend eingesetzt werden.

Des Weiteren muss angemerkt werden, dass das Vorgehen der einzelnen Fallstudien als erfolgreich angesehen werden kann, eine Validierung der entwickelten Methode in einem größeren Umfang jedoch noch nicht vorgenommen wurde. Dies könnte im Rahmen weiterer Fallstudien erfolgen. Dem potenziellen Kritikpunkt der Nachvollziehbarkeit der Ergebnisse wird dahingehend begegnet, dass die Fallstudien detailliert dokumentiert wurden, um eine Reproduzierbarkeit der erzielten Ergebnisse zu gewährleisten.

Wie in Abschnitt 4.2 festgestellt wurde, zeigt sich, dass das Thema Integration durch das *Cloud-Computing* in jüngerer Zeit stärker in den Fokus zu rücken scheint. Der Schwerpunkt hierbei besteht in der Einbindung externer Dienste zur Unterstützung neuartiger durch den GP induzierter Anforderungen, die bislang noch nicht abgedeckt werden, in die bestehende Systemlandschaft. In solchen Fällen könnte der zu unterstützende GP partitioniert und ein Teil dessen zur Durchführung an einen externen Dienstleister, bspw. den Cloud-Service-Anbieter, übertragen werden. Dabei kann untersucht werden, inwiefern der in dieser Arbeit vorgestellte Lösungsansatz eine methodische Unterstützung bei der Identifikation der durch die Einbindung externer Dienste entstehenden Integrationsbedarfe sowie deren Lösung bietet.

Zur Unterstützung bei der Lösung dieser Problematik kann auch untersucht werden, inwiefern die Methode dazu verwendet werden kann, um alternative externe Dienste in Bezug auf die Aufgabenintegration und die Einbindung in die bestehende Systemlandschaft zu vergleichen. Diese Gegenüberstellung lässt sich auch dazu verwenden, die gegenwärtig eingesetzten AwS zu katalogisieren und mit alternativen AwS zu vergleichen. Dabei kann geprüft werden, ob bereits vorhandene AwS mehr Aufgaben integrieren können als alternative AwS und ob die vorhandenen AwS weitere Aufgaben integrieren können als bislang. Mithilfe dieser Katalogisierung könnte ein Überblick über die Einsatzpotenziale vorhandener AwS in Bezug auf die Aufgabenintegration mittels der AwS geschaffen werden.

Ferner kann untersucht werden, inwieweit die Katalogisierung bei der Ablösung vorhandener AwS durch ein neu einzuführendes unterstützend eingesetzt werden kann. Hierbei bietet es sich zum einen an, mithilfe der Katalogisierung neu entstehende Integrationsbedarfe aufzudecken. Zum anderen kann die Katalogisierung eingesetzt werden, um zu prüfen, welche Aufgaben bereits durch die vorhandenen AwS integriert werden und aus diesem Grund auch vom neu einzuführenden zu integrieren sind.

Ein Beispiel für die Anwendung wäre im Rahmen der vom Bundeskabinett beschlossenen Konsolidierung der IT des Bundes (CIO Bund 2017). Im Rahmen dieser sollen Doppelstrukturen abgeschafft werden, indem bspw. Individuallösungen durch Standardlösungen ersetzt und diese an zentraler Stelle betrieben werden. Mithilfe der entwickelten Methode könnten z.B. bei Behörden in abgegrenzten, handhabbaren Teil-Informationssystemen untersucht werden, wie sich Änderungen auf GP- oder AT-Ebenen auf die Integration auswirken (What-If-Analyse). Die dadurch ggf. entstehenden Anpassungen bei der jeweils anderen Ebene sind durch Rücküberführungen zu überwinden. Zudem können mithilfe der Methode für die Teil-Informationssysteme passende Integrationskonzepte abgeleitet werden. Dadurch, dass die Methode im Zuge der Skalierbarkeit einen lokalen Einsatz erlaubt, kann ein dezentraler Einsatz bei den Behörden und eine sukzessive Untersuchung von abgrenzbaren Teilbereichen, im Gegensatz zu einem globalen Ansatz, unterstützt werden.

Weiterer Forschungsbedarf besteht darin, zu analysieren, inwiefern der vorliegende Beitrag auch auf die Integration auf Leistungsebene eines Unternehmens angewendet werden kann. Insbesondere im Umfeld von *Industrie 4.0* sowie den damit assoziierten Schlagworten *Internet der Dienste* und *Internet der Dinge* besteht die Herausforderung z. B. für Maschinenbauunternehmen darin, neue Geschäftsfelder zu erschließen. Dies betrifft bspw. einerseits die Erweiterung des Leistungsangebots durch das Anbieten von Wartungsdienstleistungen und andererseits die Kollaboration von Maschinen sowie AwS. Im Bereich der Zusammenarbeit von Maschinen könnte untersucht werden, wie die Methode auf Entwurfsebene dazu eingesetzt werden kann, um auf Leistungsebene Integrationsbedarfe im Bereich der zu automatisierenden Kollaboration aufzudecken und angemessene Lösungsvorschläge abzuleiten. Im Bereich der Wartungsdienstleistungen könnte mithilfe der Methode auf konzeptioneller Ebene untersucht werden, inwiefern die durch die Sensoren erzeugten Daten der sich in Betrieb befindlichen Maschinen bedarfsgerecht in ein Monitoringsystem des Herstellers einbinden lassen. Die entwickelte Methode ermöglicht es dabei, Datenschutzaspekte im Sinne von sensiblen Daten, die Rückschlüsse auf den Geschäftsbetrieb erlauben, einzubeziehen.

Die skizzierten Forschungsbedarfe und die von ihnen umrissenen Themenkomplexe sind untersuchenswert, da mit ihrer Behandlung weitere Anknüpfungspunkte für die umfassende Durchführung der Systementwicklungsaufgabe erschlossen werden könnten.

A Liste aller gefundenen Qualitätsattribute

In der Tab. A.1 werden alle gefundenen Merkmale zur Bildung des in Abschnitt 5.3 vorgestellten aggregierten Merkmalskataloges aufgelistet. Sofern in der in Tab. 5.4 verwiesenen Literatur englische Attribute ins deutsche und umgekehrt übersetzt wurden, wurden die in der Literatur vorgeschlagenen Übersetzungen verwendet. Merkmale, die nicht übersetzt wurden, wurden durch eigene Vorschläge ergänzt. Diese werden durch *kursive* Formatierung deutlich gemacht.

Tabelle A.1: Rohtabelle mit allen Merkmalen

Nr.	Kategorien	Merkmale	Synonyme	Englische Bezeichnung
1	Gebrauchstauglichkeit	Gebrauchstauglichkeit		functional suitability
		Funktionale Vollständigkeit		functional completeness
		Funktionale Korrektheit		functional correctness
		Funktionale Angemessenheit		functional appropriateness
2	Leistungseffizienz	Leistungseffizienz		performance efficiency
		Zeitverhalten		time behaviour
		Antwortzeit	Termindringlichkeit	response time
		Wartezeit		latency
		Reaktionsfreudigkeit		responsiveness
		Ausführungsgeschwindigkeit		execution speed
		Übergangsverzögerung		transit delay
		Verzögerung	Verspätung	delay
		Skalierbarkeit		scalability
		Verbrauchsverhalten		resource utilization

Tabelle A.1 – Fortsetzung von vorhergehender Seite

Nr.	Kategorien	Merkmale	Synonyme	Englische Bezeichnung
		Speicherplatz		space
		Durchsatz		throughput
		Rechenzeit		computation
		Speicher- Inanspruchnahme		memory usage
		Auslastung		workload
		Kapazität		capacity
		Nebenläufige Trans- aktionsverarbeitung		concurrent transaction processing
		Genauigkeit		accuracy
		Modi		modes
		Fehlerraten		miss rates
		Datenverlust		data loss
3	Kompatibili- tät	Kompatibilität		compatibility
		Koexistenz		co-existence
		Interoperabilität		interoperability
4	Benutzer- freundlich- keit	Benutzerfreundlichkeit		usability
		Leistungsfähigkeit		efficiency
		Benutzerreaktionszeit		user reaction time
		Benutzerproduktivität		user productivity
		Brauchbarkeit		usefulness
		Angemessene Begreifbarkeit		appropriateness recognizability
		Verständlichkeit		understandability
		Erlernbarkeit		learnability
		Zugänglichkeit		accessibility

Tabelle A.1 – Fortsetzung von vorhergehender Seite

Nr.	Kategorien	Merkmale	Synonyme	Englische Bezeichnung
		Bedienbarkeit		operability
		Bedienungskomfort		ease of use
		Handhabbarkeit		manageability
		Ästhetische Benutzeroberfläche	Attraktivität	user interface aesthetics
		Darstellungsform	Darstellung, Präsentation	<i>representation</i>
		Ergonomie		human engineering
		Benutzungsfreundlichkeit		user friendliness
		Einprägsamkeit		memorability
		Schutz vor Falscheingaben		user error protection
		Erwartungskonformität		likeability
5	Zuverlässigkeit	Zuverlässigkeit		reliability
		Vollständigkeit		completeness
		Genauigkeit		accuracy
		Konsistenz		consistency
		Integrität		integrity
		Korrektheit	Fehlerfreiheit	correctness
		Ordnungsmäßigkeit		compliance
		Rechtzeitigkeit	Pünktlichkeit	timeliness
		Aktualität	Alter	currency
		Reife		maturity
		Qualität		<i>quality</i>
		Fehlertoleranz		fault tolerance
		Ausfallsicherheit		resilience
		Verfügbarkeit		availability
		Wiederherstellbarkeit		recoverability

Tabelle A.1 – Fortsetzung von vorhergehender Seite

Nr.	Kategorien	Merkmale	Synonyme	Englische Bezeichnung
		Quittierungsüberwachungsprotokoll		handshaking protocol
6	Betriebs-sicherheit	Betriebssicherheit		security
		Vertraulichkeit		confidentiality
		Integrität		integrity
		Nichtabstreitbarkeit		non-repudiation
		Haftung		accountability
		Glaubwürdigkeit		authenticity
		Verfügbarkeit		availability
		Authentifizierung		authentication
		Zugriffsschutz	Sicherheit	<i>access protection</i>
		Zugriffssteuerung		access control
7	Funktions-sicherheit	Funktionssicherheit		<i>functional reliability</i>
8	Wartbarkeit	Wartbarkeit		maintainability
		Instandhaltungsvermögen		supportability
		Analysierbarkeit		analysability
		Modifizierbarkeit		modifiability
		Änderbarkeit		changeability
		Handhabbarkeit		manageability
		Stabilität		stability
		Testbarkeit		testability
		Prüffähigkeit		auditability
		Messbarkeit		<i>measurability</i>
		Modularität		modularity
		Wiederverwendbarkeit		reusability
		Verständlichkeit		understandability

Tabelle A.1 – Fortsetzung von vorhergehender Seite

Nr.	Kategorien	Merkmale	Synonyme	Englische Bezeichnung
		Nachvollziehbarkeit		traceability
		Standardkonformität	Datenformat	data format standard
9	Portabilität	Portabilität		portability
		Anpassbarkeit		adaptability
		Flexibilität		flexibility
		Installierbarkeit		installability
		Austauschbarkeit		replaceability
		Handhabbarkeit		manageability
		Konfigurationsmaß		configurability
10	Weiterentwickelbarkeit	Weiterentwickelbarkeit		evolvability
		Nachvollziehbarkeit		traceability
		Analysierbarkeit		analyzability
		Änderbarkeit		changeability
		Modifizierbarkeit		modifiability
		Integrität		integrity
		Erweiterbarkeit		extensibility
		Wiederverwendbarkeit		reusability
		Handhabbarkeit		manageability
		Portabilität		portability
		Testbarkeit		testability
		Prüffähigkeit		auditability
		Messbarkeit		<i>measurability</i>
11	Nachhaltigkeit	Nachhaltigkeit		sustainability
12	Kosten	Kosten		<i>cost</i>
13	Integrations-element	Gegenstand		<i>object</i>

Tabelle A.1 – Fortsetzung von vorhergehender Seite

Nr.	Kategorien	Merkmale	Synonyme	Englische Bezeichnung
		Zugreifbarkeit		<i>accessibility</i>
		Aktualität	Alter	<i>currency</i>
		Granularität	Aggregationsgrad, Verdichtungsgrad	<i>granularity</i>
		Relevanz	Verwendbarkeit, Wert	<i>relevance</i>
		Abbildungsdimension		<i>mapping dimension</i>
		Bedeutung		<i>significance</i>
		Informationsart	Art	<i>type of information</i>
		Volumen	Menge	<i>amount</i>
		Intensität		<i>volume</i>
		Skalierbarkeit		<i>scalability</i>
		Größe		<i>size</i>
		Verwendungsform	Verwendungszweck	<i>purpose of use</i>
		Periodizität	Häufigkeit, Frequenz	<i>frequency</i>
14	Technologie	Kommunikationsinfrastruktur Verfügbarkeit		communication infrastructure availability
		Standardkonformität	Datenformat	data format standard
Σ 128				

B Ergänzende Protokolle für das erläuternde Fallbeispiel

B.1 Protokoll der Objekt- und Transaktionszerlegung für das erläuternde Fallbeispiel im Rahmen der Istanalyse

Das zu Abb. 6.9 korrespondierende Protokoll der Objekt- und Transaktionszerlegung kann aus Tab. B.1 entnommen werden. In der Tab. B.1 werden die im IAS in Abb. 6.9 verwendeten betrieblichen Objekte *kursiv* und die Transaktionen **fett** hervorgehoben.

Tabelle B.1: Protokoll der Objekt- und Transaktionszerlegung für das erläuternde Fallbeispiel im Rahmen der Istanalyse

Objektzerlegung	Transaktionszerlegung
AusbildPrüfSyst	D (par): AusbildPrf (L)
<i>SemAusbildSyst</i>	D (par): SemAusbildg (L)
<i>PrüfungsSyst</i>	A: Lv-Angebot
V: Lehrinhalte	V: *AnmeldungLv
D: ÜberprüfAusbilderfolg (L)	D: Semesterausbildung
V: AuffLvAngeb	D (par): Prüfung (L)
D: MeldLvAngeb	A: Prf-Angebot
<i>Studierender</i>	V: Prüfungsanmeldung
	D: Prüfung

B.2 Protokoll der Objekt- und Transaktionszerlegung für das erläuternde Fallbeispiel im Rahmen der Szenariobildung

Das zu Abb. 6.11 korrespondierende Protokoll der Objekt- und Transaktionszerlegung kann aus Tab. B.2 entnommen werden. In der Tab. B.2 werden die im IAS in Abb. 6.11 verwendeten betrieblichen Objekte *kursiv* und die Transaktionen **fett** hervorgehoben.

Tabelle B.2: Protokoll der Objekt- und Transaktionszerlegung für das erläuternde Fallbeispiel im Rahmen der Szenariobildung

Objektzerlegung	Transaktionszerlegung
AusbildPrüfSyst	D (par): AusbildPrf (L)
<i>SemAusbildSyst</i>	D (par): SemAusbildg (L)
<i>PrüfungsSyst</i>	A: Lv-Angebot
V: Lehrinhalte	V: *AnmeldungLv
D: ÜberprüfAusbilderfolg (L)	D: Semesterausbildung
D: MeldLvAngeb	D (par): Prüfung (L)
<i>Studierender</i>	A: Prf-Angebot
	V: Prüfungsanmeldung
	D: Prüfung

C Ergänzende Informationen für die Fallstudie Güterdistribution

C.1 Protokoll der Objekt- und Transaktionszerlegung für die Fallstudie Güterdistribution

Das zu Abb. 7.2 korrespondierende Protokoll der Objekt- und Transaktionszerlegung kann aus Tab. C.1 entnommen werden. In der Tab. C.1 werden die im IAS in Abb. 7.2 verwendeten betrieblichen Objekte *kursiv* und die Transaktionen **fett** hervorgehoben.

Tabelle C.1: Protokoll der Objekt- und Transaktionszerlegung für das Beispiel der Güterdistribution

Objektzerlegung	Transaktionszerlegung
Güterdistributionssystem	D: Konsumgut (L)
Vertrieb	A: Produktinformation
<i>Vertriebslenkung</i>	A.par1: Produktkatalog
<i>Auftragsprüfung</i>	A.par2: Warenkorb
V: Auftragsprüfungsantrag	V: Auftrag
D: Auftragsprüfung	V.seq1: Kaufangebot
D.seq1: Inventur	V.seq1.par1: Auftragspositionen
D.seq2: Lagerbestände	V.seq1.par2: Kundenmerkmale
D.seq3: Auftragsprüfungsbericht	V.seq1.par3: Zahlungsmodalitäten
Versand + Logistik	V.seq2: Angebotsannahme und Auftragsbestätigung
Versand	D: Lieferabwicklung
<i>Versandlenkung</i>	D.seq1: Lieferung
<i>Lager</i>	D.seq2: Abrechnung
S: Warenbereitstellung	D.seq2.V: Rechnung
K:	D.seq2.D: Zahlung
Warenbereitstellungsbestätigung	

Tabelle C.1 – Fortsetzung von vorhergehender Seite

Objektzerlegung	Transaktionszerlegung
<i>Logistikdienstleister</i>	
A: Informationen und Konditionen	
V: Annahme Rahmenliefervertrag	
D: Transport	
D.seq1: Transportanweisung	
D.seq1.V: Transportauftrag	
D.seq1.D: Transportmeldung	
D.seq2: Warenfluss	
D.seq2.V: Warenabruf	
D.seq2.D: Warenbereitstellung	
S: Lieferanweisung	
K: Lieferbestätigung	
<i>Finanzwesen</i>	
S: Forderung	
K: Zahlungseingangsmeldung	
<i>Kunde</i>	

C.2 Verhaltenssicht auf den Geschäftsprozess der Fallstudie

In Abb. C.1 wird die Verhaltenssicht des GP der Fallstudie dargestellt. Abb. 7.2 bildet die dazu korrespondierende Struktursicht.

C.3 Transaktionen, zugeordnete betriebliche Objekte sowie identifizierte AIM und überlappende Aufgabenobjekte

Die Tab. C.2 basiert auf der Tab. 6.4 und enthält die Automatisierungsgrade der Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen.

In der Tab. C.3, Grundlage für diese ist die Tab. 6.2, werden sowohl die Transaktionen und Namen der Transaktionsgruppen, sofern die Transaktion einer Gruppe zugeordnet wurde, als auch die jeweils zugeordneten sendenden und empfangenden betrieblichen

Tabelle C.2: Automatisierungsgrade der Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen

Transaktion oder Transaktionsgruppe	Automatisierungspotenzial	Ist-Automatisierung
<i>D.seq1.D: Transportmeldung</i>	→	■
<i>D.seq1.V: Transportauftrag</i>	→	■
<i>D.seq1: Inventur</i>	→	■
<i>D.seq2.D: Warenbereitstellung</i>	--→	□
<i>D.seq2.V: Warenabruf</i>	--→	□
<i>D.seq2: Lagerbestände</i>	→	■
<i>K: Warenbereitstellungsbestätigung</i>	→	■
<i>K: Zahlungseingangsmeldung</i>	→	■
<i>S: Forderung</i>	→	■
<i>S: Warenbereitstellungsanweisung</i>	→	■

Objekte sowie die identifizierten AIM katalogisiert, sofern es sich um Transaktionen ohne Beteiligung von Umweltobjekten handelt. Die AIM werden anhand der in Abschnitt 3.1.3 vorgestellten AIM analysiert.

Tabelle C.3: Transaktionen, Transaktionsgruppen, zugeordnete betriebliche Objekte sowie identifizierte AIM und überlappende AO

Transaktion	Name der Transaktionsgr.	sendendes betr. Objekt	empfangendes betr. Objekt	überlappende AO	Identifizierte AIM
V: Auftragsprüfungsantrag		Vertriebslenkung	Auftragsprüfung	Auftragspositionen, Kundenmerkmale, Zahlungsmodalitäten	Reihenfolgebeziehung zwischen Aufgabendurchführungen; Partielle Identität von AO-Instanzen
D.seq1: Inventur		Auftragsprüfung	Lager	Artikel	Reihenfolgebeziehung zwischen Aufgabendurchführungen; Partielle Identität von AO-Instanzen
D.seq2: Lagerbestände		Lager	Auftragsprüfung	Artikel, Lagerbestand	Reihenfolgebeziehung zwischen Aufgabendurchführungen; Partielle Identität von AO-Instanzen
D.seq3: Auftragsprüfungsbericht		Auftragsprüfung	Vertriebslenkung	Auftragsannahme	Reihenfolgebeziehung zwischen Aufgabendurchführungen; Partielle Identität von AO-Instanzen

Tabelle C.3 – Fortsetzung von vorhergehender Seite

Transaktion	Name der Transaktionsgr.	sendendes betr. Objekt	empfangendes betr. Objekt	überlappende AO	Identifizierte AIM
S: Warenbereitstellungsanweisung		Versandlenkung	Lager	Kommissionierauftrag	Partielle Identität von AO-Instanzen
K: Warenbereitstellungsbestätigung		Lager	Versandlenkung	Kommissionierauftrag, Abschlusskennzeichnung	Partielle Identität von AO-Instanzen
D.seq1.V: Transportauftrag		Versandlenkung	Logistikdienstleister	Abholort, Anzahl Paletten, Bestimmungsort, Liefertermin	Reihenfolgebeziehung zwischen Aufgabendurchführungen; Partielle Identität von AO-Instanzen
D.seq1.D: Transportmeldung		Logistikdienstleister	Versandlenkung	Lieferschein	Reihenfolgebeziehung zwischen Aufgabendurchführungen; Partielle Identität von AO-Instanzen
D.seq2.V: Warenabruf		Logistikdienstleister	Lager	Auftrag	Reihenfolgebeziehung zwischen Aufgabendurchführungen; Partielle Identität von AO-Instanzen

Tabelle C.3 – Fortsetzung von vorhergehender Seite

Transaktion	Name der Transaktionsgr.	sendendes betr. Objekt	empfangendes betr. Objekt	überlappende AO	Identifizierte AIM
D.seq2.D: Warenbereitstellung		Lager	Logistikdienstleister	kommissionierte Ware	Reihenfolgebeziehung zwischen Aufgabendurchführungen; Partielle Identität von AO-Instanzen
S: Lieferanweisung		Vertriebslenkung	Versandlenkung	Lieferauftrag	Partielle Identität von AO-Instanzen
K: Lieferbestätigung		Versandlenkung	Vertriebslenkung	Lieferauftrag, Abschlusskennzeichnung	Partielle Identität von AO-Instanzen
S: Forderung		Vertriebslenkung	Finanzwesen	Auftrag	Partielle Identität von AO-Instanzen
K: Zahlungseingangsmeldung	Finanzwesen	Vertriebslenkung	Begleichung offener Posten	Partielle Identität von AO-Instanzen	

C.4 Erfasste Attribute für die Erstellung des Sollkonzepts

Nachfolgend werden die Spalten der Tab. C.4, Basis ist die Tab. 6.7, erläutert. In der ersten Spalte sind die Namen der Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen aufgelistet. Die Details zu diesen, können der Tab. C.3 entnommen werden. In der Spalte ausgeprägte Merkmale im Unternehmensplan, sind die Ausprägungen in eben jenem aufgelistet. Bei den Spalten zur Merkmalsausprägung auf GP-Ebene werden die Ausprägungen der in Abschnitt 5.4.5 vorgestellten Merkmale für das Szenario dargestellt. In der Spalte Regel wird die zu den ausgeprägten Merkmalen zutreffende Regel aus Abschnitt 5.5 angezeigt. Ausgeprägte Merkmale auf Ressourcenebene werden in der korrespondierenden Spalte aufgelistet. Die Soll-Automatisierung der Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen wird in der vorletzten, die Ist-Automatisierung in der letzten Spalte dargestellt.

Tabelle C.4: Sollkonzept für die Transaktionen oder Transaktionsgruppen mit Inter-AwS-Kommunikationskanälen im Szenario

Transaktion od. Transak- tionsgr.	ausgepr. Merkm.		Merkmalsausprägung auf GP-Ebene		ausgepr. Merkm.		Ist-Au- tomat.		
	im Unterneh- menspl.	Period.	Größe	Intensität	Handl. auf Nutzungsd.	Soll- Automat.			
<i>D.seq1.D:</i> <i>Transport- meldung</i>	Strategie	hoch	niedrig	niedrig	langfristig	2	Interop., Know- how, 1.000 €	→	<input checked="" type="checkbox"/>
<i>D.seq1.V:</i> <i>Transport- auftrag</i>	Strategie	hoch	niedrig	niedrig	langfristig	2	Interop., Know- how, 3.000 €	→	<input checked="" type="checkbox"/>
<i>D.seq1:</i> <i>Inventur</i>	Strategie	hoch	niedrig	moderat	langfristig	2	Interop., Know- how, 3.000 €	→	<input checked="" type="checkbox"/>
<i>D.seq2.D:</i> <i>Warenbereit- stellung</i>	Strategie	hoch	mode- rat	niedrig	langfristig	2		-->	<input type="checkbox"/>
<i>D.seq2.V:</i> <i>Warenabruf</i>	Strategie	hoch	mode- rat	niedrig	langfristig	2		-->	<input type="checkbox"/>

Tabelle C.4 – Fortsetzung von vorhergehender Seite

Transaktion od. Transak- tionsgr.	ausgepr. Merkm. im Unterneh- menspl.	Period.	Merkmalsausprägung auf GP-Ebene			ausgepr. Merkm. Handl. auf Ressour- cenebene	Soll- Automat.	Ist-Au- tomat.		
			Größe	Intensität	Anpassb.				Nutzungs- d.	
<i>D.seq2: La- gerbestände</i>	Strategie	hoch	niedrig	niedrig	niedrig	langfristig	2	Interop., Know- how, 3.000 €	→	
<i>K: Warenbe- reitstellungs- bestätigung</i>	Strategie	hoch	niedrig	niedrig	niedrig	langfristig	2	Interop., Know- how, 3.000 €	→	
<i>S: Warenbe- reitstellungs- anweisung</i>	Strategie	hoch	niedrig	niedrig	niedrig	langfristig	2	Interop., Know- how, 3.000 €	→	

Literaturverzeichnis

- Aier, Stephan, Christian Riege und Robert Winter (2008): „Unternehmensarchitektur - Literaturüberblick und Stand der Praxis“. In: *WIRTSCHAFTSINFORMATIK* 50.4, Seiten 292–304 (siehe Seite 15).
- Alpar, Paul, Rainer Alt, Frank Bensberg, Heinz Lothar Grob, Peter Weimann und Robert Winter (2014): *Anwendungsorientierte Wirtschaftsinformatik: Strategische Planung, Entwicklung und Nutzung von Informationssystemen*. 7., aktualisierte und erweiterte Auflage. Wiesbaden: Springer Vieweg. URL: <http://dx.doi.org/10.1007/978-3-658-00521-4> (siehe Seite 90).
- Ballnus, Rainer (2000): *Erfolg mit EDI und E-Commerce: Handlungsempfehlungen für die Abstimmung und Organisation interorganisationaler Netzwerke*. Marburg: Tectum-Verlag (siehe Seite 51).
- Balzert, Helmut (2009): *Lehrbuch der Software-Technik*. 3. Auflage. Heidelberg [u.a.]: Spektrum Akademischer Verlag (siehe Seite 131).
- Balzert, Helmut (2011): *Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb*. 3. Auflage. Heidelberg: Spektrum Akademischer Verlag (siehe Seiten 89, 129 ff.).
- Barroca, Leonor, Jon Hall und Patrick Hall (2000): „An Introduction and History of Software Architectures, Components, and Reuse“. In: *Software architectures*. Herausgegeben von Leonor Barroca, Jon Hall und Patrick Hall. London: Springer, Seiten 1–11 (siehe Seite 13).
- Bauer, Andreas, Wolfgang Hümmer und Lutz Schlesinger (2002): „Einführung in die Problematiken Heterogener Informationssysteme“. In: *Heterogene Informationssysteme*. Arbeitsberichte des Instituts für Informatik. Erlangen, Seiten 1–15 (siehe Seiten 30 f.).
- Buckl, Sabine, Alexander M. Ernst, Josef Lankes und Florian Matthes (2008): *Enterprise Architecture Management Pattern Catalog: Version 1.0, February 2008*. URL: <https://www.matthes.in.tum.de/file/16knkokcwzfwj/Projects/EAMPC/Download/members%20only/EAMPatternCatalogV1.0.pdf> (Abruf am: 17. 03. 2016) (siehe Seite 7).
- Buhl, Hans Ulrich und Bernd Heinrich (2004): „Unternehmensarchitekturen in der Praxis - Architekturdesign am Reißbrett vs. situationsbedingte Realisierung von Informationssystemen“. In: *WIRTSCHAFTSINFORMATIK* 46.4, Seite 311. URL:

- http://www.wirtschaftsinformatik.de/pdf/wi2004_4_311_322.pdf (Abruf am: 29. 11. 2011) (siehe Seite 15).
- Chen, Lianping, Muhammad Ali Babar und Bashar Nuseibeh (2013): „Characterizing Architecturally Significant Requirements“. In: *IEEE Software* 30.2, Seiten 38–45 (siehe Seiten 90, 96, 100).
- CIO Bund (2017): *IT-Konsolidierung Bund: Das Projekt "IT-Konsolidierung Bund"*. URL: https://www.cio.bund.de/Web/DE/Innovative-Vorhaben/IT-Konsolidierung%20Bund/it_konsolidierung_bund_node.html (Abruf am: 17. 12. 2017) (siehe Seite 177).
- Conrad, Stefan (1997): *Föderierte Datenbanksysteme: Konzepte der Datenintegration*. Berlin: Springer (siehe Seite 7).
- Dudenredaktion (2008): „Fremdwörterbuch 9.A“. In: *DUDEN Doppelpack; Die deutsche Rechtschreibung 24.A. / Fremdwörterbuch 9.A*. Herausgegeben von Dudenredaktion. Band 5. Mannheim u. a.: Bibliographisches Institut & F.A. Brockhaus AG (siehe Seiten 1, 30).
- Eckert, Sven, Martin Schissler und Otto K. Ferstl (2004): „Einsatz einer Entwicklungsmethodik für die überbetriebliche Integration von Anwendungssystemen im Rahmen einer Fallstudie aus der Automobilzulieferindustrie“. In: *Überbetriebliche Integration von Anwendungssystemen*. Herausgegeben von Dieter Bartmann, Peter Mertens und Elmar J. Sinz. Aachen: Shaker, Seiten 41–59. URL: <https://sedaintra.seda.wiai.uni-bamberg.de/forwin/publikationen/abschlussstagung-supplyon.zip> (Abruf am: 13. 03. 2011) (siehe Seite 172).
- Eckert, Sven, Martin Schissler, Stephan Mantel und Claus Schäffner (2003): „Entwicklung von Kopplungsarchitekturen - Evaluierung einer Methodik anhand eines Beispiels aus der Automobilzulieferindustrie“. In: *Modellierung betrieblicher Informationssysteme - MobIS 2003*. Herausgegeben von Elmar J. Sinz, Markus Plaha und Peter Neckel. Proceedings. Bonn: Gesellschaft für Informatik, Seiten 87–107. URL: <http://subs.emis.de/LNI/Proceedings/Proceedings38/article1129.html> (Abruf am: 27. 06. 2012) (siehe Seite 31).
- Eckert, Sven, Christian Suchan, Otto K. Ferstl und Martin Schissler (2005): „Integration von Anwendungssystemen für die Materialwirtschaft - Anwendung einer Entwicklungsmethodik im Bereich des Kraftwerkbaus“. In: *Wirtschaftsinformatik 2005. eEconomy, eGovernment, eSociety*. Herausgegeben von Otto K. Ferstl, Elmar J. Sinz, Sven Eckert und Tilman Isselhorst. Heidelberg: Physica-Verlag, Seiten 667–686. URL: <http://141.13.6.53:8080/wi2005/de/paper/Paper/WI05-Beitrag287.pdf> (Abruf am: 05. 07. 2012) (siehe Seite 172).

-
- Engels, Gregor, Andreas Hess, Bernhard Humm, Oliver Juwig, Marc Lohmann, Jan-Peter Richter, Markus Voß und Johannes Willkomm (2008): *Quasar Enterprise: Anwendungslandschaften serviceorientiert gestalten*. 1. Aufl. Heidelberg: dpunkt (siehe Seiten 64 ff.).
- Engels, Gregor und Markus Voß (2008): „Quasar Enterprise: Anwendungslandschaften serviceorientiert gestalten“. In: *Informatik Spektrum* 31.6, Seiten 548–555. URL: 10.1007/s00287-008-0287-4 (Abruf am: 28. 07. 2015) (siehe Seiten 60, 64 f., 83, 149).
- Ferstl, Otto K. (1979): *Konstruktion und Analyse von Simulationsmodellen*. Band 22. Beiträge zur Datenverarbeitung und Unternehmensforschung. Königstein/Ts und Regensburg: Hain (siehe Seite 4).
- Ferstl, Otto K. (1992): *Integrationskonzepte betrieblicher Anwendungssysteme*. Koblenz (siehe Seiten 32, 46–52).
- Ferstl, Otto K. und Elmar J. Sinz (1984): *Software-Konzepte der Wirtschaftsinformatik*. De-Gruyter-Lehrbuch. Berlin: de Gruyter (siehe Seite 29).
- Ferstl, Otto K. und Elmar J. Sinz (1990): „Objektmodellierung betrieblicher Informationssysteme im Semantischen Objektmodell (SOM)“. In: *WIRTSCHAFTSINFORMATIK* 32.6, Seiten 566–581 (siehe Seite 15).
- Ferstl, Otto K. und Elmar J. Sinz (1991): „Ein Vorgehensmodell zur Objektmodellierung betrieblicher Informationssysteme im Semantischen Objektmodell (SOM)“. In: *WIRTSCHAFTSINFORMATIK* 33.6, Seiten 477–491 (siehe Seite 15).
- Ferstl, Otto K. und Elmar J. Sinz (1993): „Geschäftsprozeßmodellierung“. In: *WIRTSCHAFTSINFORMATIK* 35.6, Seiten 589–592 (siehe Seite 18).
- Ferstl, Otto K. und Elmar J. Sinz (1995a): „Der Ansatz des Semantischen Objektmodells (SOM) zur Modellierung von Geschäftsprozessen“. In: *WIRTSCHAFTSINFORMATIK* 37.3, Seiten 209–220 (siehe Seiten 15, 19).
- Ferstl, Otto K. und Elmar J. Sinz (1995b): *Re-Engineering von Geschäftsprozessen auf der Grundlage des SOM-Ansatzes*. Bamberg. URL: <https://sedaintra.seda.wiai.uni-bamberg.de/downloads/no26.pdf> (Abruf am: 22. 11. 2013) (siehe Seite 116).
- Ferstl, Otto K. und Elmar J. Sinz (2006): „Modeling of Business Systems Using the Semantic Object Model (SOM): A Methodological Framework“. In: *Handbook on Architectures of Information Systems*. Herausgegeben von Peter Bernus, Kai Mertins und Günter Schmidt. International handbooks on information systems. Berlin: Springer, Seiten 347–367 (siehe Seite 15).
- Ferstl, Otto K. und Elmar J. Sinz (2013): *Grundlagen der Wirtschaftsinformatik*. 7. Auflage. München: Oldenbourg (siehe Seiten 7 f., 14–24, 29 f., 33 f., 38, 46–55, 111, 113 ff., 122, 129, 136, 143).

- Ferstl, Otto K., Elmar J. Sinz, Christoph Hammel, Michael Schlitt und Stefan Wolf (1997): „Bausteine für komponentenbasierte Anwendungssysteme“. In: *HMD* 34.197, Seiten 24–46 (siehe Seite 48).
- Fischer, Daniel (2008): *Unternehmensübergreifende Integration von Informationssystemen: Bestimmung des Integrationsgrades auf elektronischen Marktplätzen*. 1. Auflage. Gabler Edition Wissenschaft. Wiesbaden: Gabler (siehe Seiten 43 f.).
- Fischer, Gerlinde (2015): „Ein Zugriffskontrollmodell für aufgabenbezogene Rollen“. Dissertation. Bamberg: Otto-Friedrich-Universität Bamberg. URL: <https://opus4.kobv.de/opus4-bamberg/frontdoor/index/index/docId/45674> (Abruf am: 12. 04. 2016) (siehe Seite 121).
- Frank, Ulrich (2014): „Multi-perspective enterprise modeling: foundational concepts, prospects and future research challenges“. In: *Software & Systems Modeling* 13.3, Seiten 941–962. URL: <http://dx.doi.org/10.1007/s10270-012-0273-9> (siehe Seite 73).
- Frese, Erich, Matthias Graumann und Ludwig Theuvsen (2012): *Grundlagen der Organisation: Entscheidungsorientiertes Konzept der Organisationsgestaltung*. 10., überarb. und erw. Aufl. Lehrbuch. Wiesbaden: Gabler (siehe Seite 28).
- Gaitanides, Michael (1983): *Prozeßorganisation: Entwicklung, Ansätze und Programme prozeßorientierter Organisationsgestaltung*. WiSo-Kurzlehrbücher. München: Vahlen (siehe Seite 28).
- Gaitanides, Michael (2007): *Prozessorganisation: Entwicklung, Ansätze und Programme des Managements von Geschäftsprozessen*. 2., vollst. überarb. Aufl. Vahlens Handbücher der Wirtschafts- und Sozialwissenschaften. München: Vahlen (siehe Seite 19).
- Gassner, Christian (1996): „Konzeptionelle Integration heterogener Transaktionssysteme“. Dissertation. St. Gallen: Universität St. Gallen (siehe Seite 42).
- Gericke, Anke, Mario Klesse, Robert Winter und Felix Wortmann (2010): „Success Factors of Application Integration: An Exploratory Analysis“. In: *Communications of the Association for Information Systems* 27.1. URL: <http://aisel.aisnet.org/cais/vol27/iss1/37> (Abruf am: 01. 12. 2015) (siehe Seite 61).
- Grochla, Erwin (1975): *Betriebliche Planung und Informationssysteme: Entwicklung und aktuelle Aspekte*. Band 373. Rowohlt's deutsche Enzyklopädie. Reinbek: Rowohlt (siehe Seite 39).
- Grochla, Erwin (1982): *Grundlagen der organisatorischen Gestaltung*. Band 100. Sammlung Poeschel. Stuttgart: Poeschel (siehe Seite 30).

-
- Gronau, Norbert, Moreen Stein, Sandy Eggert und Anne Lämmer (2008): „Integration öffentlicher Anwendungslandschaften“. In: *DW 2008*. Herausgegeben von Barbara Dinter, Robert Winter, Peter Chamoni, Norbert Gronau und Klaus Turowski. Lecture Notes in Informatics. Bonn: Gesellschaft für Informatik, Seiten 425–435 (siehe Seiten 60, 67 f., 83, 149).
- Gutzwiller, Thomas (1994): *Das CC-RIM-Referenzmodell für den Entwurf von betrieblichen, transaktionsorientierten Informationssystemen*. Band 54. Betriebs- und Wirtschaftsinformatik. Heidelberg: Physica-Verlag (siehe Seiten 71, 75, 79).
- Hagen, Claus (2004): „Integrationsarchitektur der Credit Suisse“. In: *Enterprise Application Integration*. Herausgegeben von Stephan Aier und Marten Schönherr. Band 1. Enterprise Architecture. Berlin: GITO-Verl., Seiten 61–83. URL: http://www.sysedv.tu-berlin.de/eai/eaibook/leseprobe_beitrag2.pdf (Abruf am: 19. 12. 2011) (siehe Seite 30).
- Heinrich, Lutz Jürgen, Armin Heinzl und René Riedl (2011): *Wirtschaftsinformatik: Einführung und Grundlegung*. 4., überarbeitete und erweiterte Auflage. Berlin und Heidelberg: Springer Berlin Heidelberg (siehe Seiten 21, 87).
- Hentze, Joachim, Albert Heinecke und Andreas Kammel (2001): *Allgemeine Betriebswirtschaftslehre aus Sicht des Managements*. Band 2040. UTB für Wissenschaft Uni-Taschenbücher Mittlere Reihe Betriebswirtschaft. Bern: Haupt (siehe Seiten 17, 28, 30).
- Hergula, Klaudia und Theo Härder (1999): „Eine Abbildungsbeschreibung zur Funktionsintegration in heterogenen Anwendungssystemen“. In: *4. Workshop Föderierte Datenbanken*. Herausgegeben von Ralf-Detlef Kutsche und Johann-Christoph Freytag. URL: http://cis.cs.tu-berlin.de/Forschung/FDBS-99/hergula_final.ps (Abruf am: 03.02.2011) (siehe Seite 31).
- Hohpe, Gregor, Bobby Woolf und Kyle Brown (2004): *Enterprise integration patterns: Designing, building, and deploying messaging solutions*. 4. Auflage. A Martin Fowler signature book. Boston und Mass: Addison-Wesley (siehe Seite 7).
- Horváth, Péter (2011): *Controlling*. 12. Auflage. Vahlens Handbücher der Wirtschafts- und Sozialwissenschaften. München: Vahlen (siehe Seiten 90, 97).
- International Organization for Standardization und International Electrotechnical Commission (2001): *ISO/IEC 9126-1: Software engineering - Product quality - Part 1: Quality model*. Geneva (siehe Seite 89).
- International Organization for Standardization und International Electrotechnical Commission (2008): *ISO/IEC 25012: Software engineering - Software product Quality Requirements and Evaluation (SQaRE) - Data quality model*. Geneva (siehe Seite 89).

- International Organization for Standardization und International Electrotechnical Commission (2011): *ISO/IEC 25010: Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQaRE) - System and software quality models*. Geneva (siehe Seiten 89 ff., 99 f.).
- Jung, Reinhard (2006): *Architekturen zur Datenintegration: Gestaltungsempfehlungen auf der Basis fachkonzeptueller Anforderungen*. 1. Auflage. s.l.: DUV Deutscher Universitäts-Verlag (siehe Seiten 60 f., 69–72, 83, 90, 95, 97, 149).
- Kattenstroth, Heiko, Ulrich Frank und David Heise (2013): „Towards a Modelling Method in Support of Evaluating Information Systems Integration“. In: *Fifth International Workshop on Enterprise Modelling and Information Systems Architectures*. Herausgegeben von Reinhard Jung und Manfred Reichert. Lecture Notes in Informatics (LNI) - Proceedings. Bonn: Köllen Druck+Verlag GmbH, Seiten 85–99. URL: <http://subs.emis.de/LNI/Proceedings/Proceedings222/85.pdf> (Abruf am: 21. 01. 2015) (siehe Seiten 60, 62, 73 ff., 83, 149).
- Kiehl, Tobias (2012): „Integration von Anwendungssystemen im Hochschulbereich am Beispiel der Campus-Management-Systeme UnivIS und FlexNow: Master Thesis“. Master Thesis. Bamberg (siehe Seiten 32, 172).
- Kieser, Alfred und Peter Walgenbach (2010): *Organisation*. 6. Auflage. Stuttgart: Schäffer-Poeschel (siehe Seite 28).
- Kleeberg, Michael, Christian Zirpins und Holger Kirchner (2014): „Information Systems Integration in the Cloud: Scenarios, Challenges and Technology Trends“. In: *Future Business Software*. Herausgegeben von Gino Brunetti, Thomas Feld, Lutz Heuser, Joachim Schnitter und Christian Webel. Progress in IS. Cham: Springer, Seiten 39–54 (siehe Seite 62).
- Kosiol, Erich (1976): *Organisation der Unternehmung*. 2., durchges. Band 6. Reihe A, Betriebswirtschaftslehre. Wiesbaden: Gabler (siehe Seiten 17 ff., 28 f., 96).
- Krumbiegel, Jörg (1996a): *Universitätsprozeß "Prüfung"*. Bamberg. URL: <https://sedaintra.seda.wiai.uni-bamberg.de:443/forschung/kumi/lehre/pruefung/pruefung.htm> (Abruf am: 16. 10. 2013) (siehe Seite 25).
- Krumbiegel, Jörg (1996b): *Universitätsprozeß "Semesterausbildung"*. Bamberg. URL: <https://sedaintra.seda.wiai.uni-bamberg.de:443/forschung/kumi/lehre/semausb/semausb.htm> (Abruf am: 16. 10. 2013) (siehe Seite 24).
- Krumbiegel, Jörg (1997): *Integrale Gestaltung von Geschäftsprozessen und Anwendungssystemen in Dienstleistungsbetrieben: Univ., Diss.–Bamberg, 1996*. DUV. Wiesbaden: DUV Deutscher Universitäts-Verlag (siehe Seiten 20, 24, 34 ff., 109, 111 f., 114 f., 121, 157).

-
- Küpper, Hans-Ulrich, Gunther Friedl, Christian Hofmann, Yvette Hofmann und Burkhard Pedell (2013): *Controlling: Konzeption, Aufgaben, Instrumente*. 6., überarb. Aufl. Stuttgart: Schäffer-Poeschel (siehe Seiten 90, 95, 97).
- Lam, Wing und Venky Shankararaman (2004): „An Enterprise Integration Methodology“. In: *IT Professional* 6.2, Seiten 40–48 (siehe Seiten 90, 97 ff.).
- Lankes, Josef, Florian Matthes und André Wittenburg (2005): „Softwarekartographie: Systematische Darstellung von Anwendungslandschaften“. In: *Wirtschaftsinformatik 2005. eEconomy, eGovernment, eSociety*. Herausgegeben von Otto K. Ferstl, Elmar J. Sinz, Sven Eckert und Tilman Isselhorst. Heidelberg: Physica-Verlag, Seiten 1443–1462. URL: <http://141.13.6.53:8080/wi2005/de/paper/Paper/WI05-Beitrag192.pdf> (Abruf am: 20. 01. 2011) (siehe Seiten 37, 79).
- Lankes, Josef, Florian Matthes und André Wittenburg (2006): „Softwarekartographie als Beitrag zum Architekturmanagement“. In: *Unternehmensarchitekturen und Systemintegration*. Herausgegeben von Stephan Aier und Marten Schönherr. Band 3. Enterprise Architecture. Berlin: GITO-Verl., Seiten 305–333. URL: <http://www.matthes.in.tum.de/file/Publications/2005/LMW05e/LMW05e.pdf> (Abruf am: 26. 07. 2011) (siehe Seiten 37, 67, 69).
- Lässig, Jörg und Markus Ullrich (2013): „Enterprise Application Integration - The Cloud Perspective“. In: *Web Engineering*. Herausgegeben von Florian Daniel, Peter Dolog und Qing Li. Band 7977. Lecture Notes in Computer Science. Berlin: Springer, Seiten 522–525 (siehe Seite 62).
- Legner, Christine und Boris Otto (2007): „Stammdaten-Management“. In: *Das Wirtschaftsstudium* 36.4, Seiten 562–568 (siehe Seite 7).
- Leist-Galanos, Susanne (2006): *Methoden zur Unternehmensmodellierung: Vergleich, Anwendungen und Diskussion der Integrationspotenziale: Univ., Habil.-Schr.–St. Gallen, 2004*. Berlin: Logos-Verl. (siehe Seite 15).
- Lillrank, Paul (2003): „The Quality of Standard, Routine and Nonroutine Processes“. In: *Organization Studies* 24.2, Seiten 215–233 (siehe Seite 100).
- Mairiza, Dewi, Didar Zowghi und Nurie Nurmuliani (2010): „An Investigation into the Notion of Non-Functional Requirements“. In: *Proceedings of the 25th Annual ACM Symposium on Applied Computing 2010*. Herausgegeben von Sung Y. Shin, Sascha Ossowski, Michael Schumacher, Mathew J. Palakal und Chih-Cheng Hung. New York: ACM, Seiten 311–317 (siehe Seiten 90 f.).
- Mantel, Stephan, Sven Eckert, Martin Schissler, Claus Schäffner, Otto K. Ferstl und Elmar J. Sinz (2004): „Eine Entwicklungsmethodik für die überbetriebliche Integration von Anwendungssystemen“. In: *Überbetriebliche Integration von Anwendungssystemen*. Herausgegeben von Dieter Bartmann, Peter Mertens und Elmar J. Sinz. Aachen: Shaker, Seiten 21–39. URL: <https://sedaintra.seda.wiai.uni->

- bamberg.de/forwin/publikationen/abschlusstagung-methodik.zip (Abruf am: 16. 10. 2013) (siehe Seite 31).
- Mantel, Stephan, Bernd Knobloch, Thorsten Ruffer, Martin Schissler, Klaus Schmitz, Otto K. Ferstl und Elmar J. Sinz (2001): *Analyse der Integrationspotenziale von Kommunikationsplattformen für verteilte Anwendungssysteme: Bericht*. Bamberg. URL: https://sedaintra.seda.wiai.uni-bamberg.de/forwin/publikationen/fwn_2000-009.zip (Abruf am: 16. 10. 2013) (siehe Seiten 47 ff., 51 ff.).
- Mantzana, Vasiliki, Marinos Themistocleous, Zahir Irani und Khalil Khoubati (2008): „Information Systems and Healthcare XXV: Factors and Actors Affecting the EAI Adoption in the Healthcare Sector“. In: *Communications of the Association for Information Systems* 22.1, Seiten 104–115. URL: <http://aisel.aisnet.org/cais/vol22/iss1/6> (Abruf am: 01. 12. 2015) (siehe Seite 61).
- Matthes, Florian und André Wittenburg (2004a): *Softwarekarten zur Visualisierung von Anwendungslandschaften und ihren Aspekten - Eine Bestandsaufnahme*. München. URL: <http://www.matthes.in.tum.de/file/Publications/2004/MaWi04a/040326-MaWi-Statusbericht-Softwarekartographie.pdf> (Abruf am: 24. 01. 2011) (siehe Seiten 35 f.).
- Matthes, Florian und André Wittenburg (2004b): „Softwarekartographie: Visualisierung von Anwendungslandschaften und ihrer Schnittstellen“. In: *Informatik 2004*. Herausgegeben von Peter Dadam und Manfred Reichert. Proceedings. Bonn: Gesellschaft für Informatik, Seiten 71–75. URL: <http://subs.emis.de/LNI/Proceedings/Proceedings51/GI-Proceedings.51-18.pdf> (Abruf am: 26. 07. 2011) (siehe Seiten 35 f.).
- Mertens, Peter (2013a): *Integrierte Informationsverarbeitung 1: Operative Systeme in der Industrie*. 18., überarbeitete und aktualisierte Auflage. Band 1. Wiesbaden: Springer Gabler. URL: <http://dx.doi.org/10.1007/978-3-8349-4395-8> (siehe Seite 1).
- Mertens, Peter (2013b): „Wirtschaftsinformatik“. In: *Enzyklopädie der Wirtschaftsinformatik*. Herausgegeben von Karl Kurbel, Jörg Becker, Norbert Gronau, Elmar J. Sinz und Leena Suhl. München: Oldenbourg. URL: <http://www.enzyklopaedie-der-wirtschaftsinformatik.de/wi-enzyklopaedie/lexikon/uebergreifendes/Kerndisziplinen/Wirtschaftsinformatik/> (Abruf am: 17. 07. 2014) (siehe Seite 58).
- Mertens, Peter, Freimut Bodendorf, Wolfgang König, Arnold Picot, Matthias Schumann und Thomas Hess (2005): *Grundzüge der Wirtschaftsinformatik*. Neunte, überarbeitete Auflage. Springer-Lehrbuch. Berlin und Heidelberg: Springer-Verlag Berlin Heidelberg (siehe Seite 1).
- Myers, Glenford J. (1978): *Composite/structured design*. New York: Van Nostrand Reinhold (siehe Seite 33).

-
- Noffsinger, W. B., Robert Niedbalski, Michael Blanks und Niall Emmart (1998): „Legacy object modeling speeds software integration“. In: *Communications of the ACM* 41.12, Seiten 80–89 (siehe Seiten 39, 44).
- Nordsieck, Fritz (1972): *Betriebsorganisation: Betriebsaufbau und Betriebsablauf*. 4. Aufl. Band 3. Sammlung Poeschel / Betriebswirtschaftliche Studienbücher. Stuttgart: Poeschel (siehe Seite 18).
- Offermann, Philipp, Sören Blom, Olga Levina und Udo Bub (2010): „Vorschlag für Komponenten von Methodendesigntheorien“. In: *WIRTSCHAFTSINFORMATIK* 52.5, Seiten 287–297 (siehe Seiten 5, 151).
- Ortner, Erich (1997): *Methodenneutraler Fachentwurf: Zu den Grundlagen einer anwendungsorientierten Informatik*. Teubner-Reihe Wirtschaftsinformatik. Stuttgart und Leipzig: Teubner (siehe Seite 31).
- Österle, Hubert (1996): „Integration: Schlüssel zur Informationsgesellschaft“. In: *Middleware*. Herausgegeben von Hubert Österle, Rainer Riehm und Petra Vogler. Business Computing. Braunschweig: Vieweg, Seiten 1–23 (siehe Seiten 42 f.).
- Penzenstadler, Birgit, Ankita Raturi, Debra Richardson und Bill Tomlinson (2014): „Safety, Security, Now Sustainability: The Nonfunctional Requirement for the 21st Century“. In: *IEEE Software* 31.3, Seiten 40–47 (siehe Seiten 90 f.).
- Picot, Arnold, Helmut Dietl und Egon Franck (2008): *Organisation: Eine ökonomische Perspektive*. 5. Auflage. Stuttgart: Schäffer-Poeschel (siehe Seite 30).
- Picot, Arnold, Ralf Reichwald und Rolf T. Wigand (2003): *Die grenzenlose Unternehmung: Information, Organisation und Management ; Lehrbuch zur Unternehmensführung im Informationszeitalter*. 5. Auflage. Wiesbaden: Gabler (siehe Seite 43).
- Picot, Arnold und Peter Rohrbach (1995): „Organisatorische Aspekte von Workflow-Management-Systemen“. In: *Information Management* 1, Seiten 28–35 (siehe Seite 100).
- Picot, Gerhard (2008): „Wirtschaftliche und wirtschaftsrechtliche Aspekte bei der Planung der Mergers & Acquisitions“. In: *Handbuch Mergers & Acquisitions*. Herausgegeben von Gerhard Picot. Stuttgart: Schäffer-Poeschel, Seiten 2–47 (siehe Seite 43).
- Riebisch, Matthias und Stephan Bode (2009): „Software-Evolvability“. In: *Informatik Spektrum* 32.4, Seiten 339–343 (siehe Seiten 90 f., 96).
- Riehm, Rainer (1997): „Integration von heterogenen Applikationen“. Dissertation. St. Gallen: Universität St. Gallen (siehe Seiten 38 f.).

- Riehm, Rainer und Petra Vogler (1996): „Middleware: Infrastruktur für die Integration“. In: *Middleware*. Herausgegeben von Hubert Österle, Rainer Riehm und Petra Vogler. Business Computing. Braunschweig: Vieweg, Seiten 25–135 (siehe Seite 38).
- Rosemann, Michael (1996): *Komplexitätsmanagement in Prozeßmodellen: Methodenspezifische Gestaltungsempfehlungen für die Informationsmodellierung*. Schriften zur EDV-orientierten Betriebswirtschaft. Wiesbaden und Münster: Gabler Verlag (siehe Seite 31).
- Roztocki, Narcyz und Heinz Roland (2006): „Enterprise Application Integration and its Effect on Business Value: A Stock Market View“. In: *AMCIS 2006 Proceedings*. URL: <http://aisel.aisnet.org/amcis2006/64> (Abruf am: 01. 12. 2015) (siehe Seite 61).
- Schach, Stephen R. (2011): *Object oriented and classical software engineering: 4. Aufl. u.d.T.: Schach, Stephen R.: Classical and object-oriented software engineering with UML and C++*. 8. ed., global ed. Boston: McGraw-Hill (siehe Seite 33).
- Schelp, Joachim und Alexander Schwinn (2005): „Extending the Business Engineering Framework for Application Integration Purposes“. In: *Proceedings of the 2005 ACM symposium on Applied computing*. Herausgegeben von Hisham M. Haddad, Lorie M. Liebrock, Andrea Omicini und Roger L. Wainwright. New York: ACM, Seiten 1333–1337 (siehe Seite 4).
- Schissler, Martin, Stephan Mantel, Otto K. Ferstl und Elmar J. Sinz (2001): *Unterstützung von Kopplungsarchitekturen durch SAP R/3: Bericht*. Bamberg. URL: https://sedaintra.seda.wiai.uni-bamberg.de/forwin/publikationen/fwn_2001-008.zip (Abruf am: 16. 10. 2013) (siehe Seiten 3, 46, 50 ff.).
- Schissler, Martin, Stephan Mantel, Otto K. Ferstl und Elmar J. Sinz (2002): „Kopplungsarchitekturen zur überbetrieblichen Integration von Anwendungssystemen und ihre Realisierung mit SAP R/3“. In: *WIRTSCHAFTSINFORMATIK* 44.5, Seiten 459–468 (siehe Seiten 31 f.).
- Schmidt, Alexander (2010): „Entwicklung einer Methode zur Stammdatenintegration“. Dissertation. St. Gallen: Universität St. Gallen. URL: [http://www1.unisg.ch/www/edis.nsf/SysLkpByIdentifier/3848/\\$FILE/dis3848.pdf](http://www1.unisg.ch/www/edis.nsf/SysLkpByIdentifier/3848/$FILE/dis3848.pdf) (Abruf am: 09. 05. 2012) (siehe Seite 7).
- Schwinn, Alexander (2005): „Entwicklung einer Methode zur Gestaltung von Integrationsarchitekturen für Informationssysteme“. Dissertation. Bamberg: Universität St. Gallen. URL: <http://212.227.178.170/pdf/diss.pdf> (Abruf am: 31. 05. 2012) (siehe Seiten 60 f., 75–79, 83, 149).

-
- Schwinn, Alexander (2006): „Entwurfsmusterbasierter Ansatz zur Systematisierung von Applikationsbeziehungen im Business Engineering“. In: *Integrationsmanagement*. Herausgegeben von Joachim Schelp und Robert Winter. Berlin und Heidelberg: Springer-Verlag, Seiten 31–59 (siehe Seite 76).
- Sinz, Elmar J. (1995): *Kann das Geschäftsprozessmodell der Unternehmung das unternehmensweite Datenschema ablösen? Bamberger Beiträge zur Wirtschaftsinformatik*. Bamberg. URL: <http://141.13.6.53:8080/downloads/no33.pdf> (Abruf am: 26.01.2012) (siehe Seite 51).
- Sinz, Elmar J. (1996): „Ansätze zur fachlichen Modellierung betrieblicher Informationssysteme: Entwicklung, aktueller Stand und Trends“. In: *Information Engineering*. Herausgegeben von Heidi Heilmann, Lutz Jürgen Heinrich und Friedrich Roithmayr. Lehrbuchreihe Wirtschaftsinformatik. München: Oldenbourg, Seiten 123–143 (siehe Seite 110).
- Sinz, Elmar J. (1998): „Universitätsprozesse“. In: *Gestaltungskonzepte für Hochschulen*. Herausgegeben von Hans-Ulrich Küpper und Elmar J. Sinz. Stuttgart: Schäffer-Poeschel, Seiten 13–57 (siehe Seiten 24 ff.).
- Sinz, Elmar J. (2002): „Konstruktion von Informationssystemen“. In: *Informatik-Handbuch*. Herausgegeben von Peter Rechenberg und Gustav Pomberger. München: Hanser, Seiten 1069–1084 (siehe Seiten 13 ff., 21).
- Sinz, Elmar J. (2004): „Unternehmensarchitekturen in der Praxis - Architekturdesign am Reißbrett vs. situationsbedingte Realisierung von Informationssystemen“. In: *WIRTSCHAFTSINFORMATIK* 46.4, Seiten 315–316 (siehe Seiten 13 f.).
- Sinz, Elmar J. und Benedikt Wismans (1998): *Das "Elektronische Prüfungsamt"*. Bamberg. URL: <https://sedaintra.seda.wiai.uni-bamberg.de/downloads/no47.pdf> (Abruf am: 22.11.2013) (siehe Seite 8).
- Sinz, Elmar J. und Benedikt Wismans (2001): „Anforderungen an die IV-Infrastruktur von Hochschulen“. In: *Unternehmen Hochschule 2001*. Herausgegeben von Hans-Jürgen Appelrath, Rolf Beyer, Uwe Marquardt, Heinrich C. Mayr und Claudia Steinberger. Proceedings. Bonn: Gesellschaft für Informatik, Seiten 17–31. URL: http://subs.emis.de/LNI/Proceedings/Proceedings06/AnfordandieIV-INfrastr_2.pdf (Abruf am: 09.03.2011) (siehe Seiten 4, 30).
- Tanenbaum, Andrew S. und Maarten van Steen (2008): *Verteilte Systeme: Prinzipien und Paradigmen*. 2. Auflage. München [u.a.]: Pearson Studium (siehe Seite 39).
- Thranert, Maik (2008): „Integration-Engineering: Grundlagen, Vorgehen und Fallstudien“. Dissertation. Leipzig: Universität Leipzig (siehe Seite 45).
- van der Aalst, Wil M. P. (2011): *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Berlin und Heidelberg: Springer-Verlag Berlin Heidelberg (siehe Seite 99).

- Vogler, Petra (2006): *Prozess- und Systemintegration: Evolutionäre Weiterentwicklung bestehender Informationssysteme mit Hilfe von Enterprise Application Integration*. Wiesbaden: DUV Deutscher Universitäts-Verlag (siehe Seiten 60 f., 79–83, 149, 175).
- Wagner, Daniel, Benjamin Leunig, Christian Suchan, Jochen Frank und Otto K. Ferstl (2011): „Klassifikation von Geschäftsprozessen anhand ihres Flexibilitätsbedarfs“. In: *Dienstorientierte IT-Systeme für hochflexible Geschäftsprozesse*. Herausgegeben von Elmar J. Sinz, Dieter Bartmann, Freimut Bodendorf und Otto K. Ferstl. Schriften aus der Fakultät Wirtschaftsinformatik und Angewandte Informatik der Otto-Friedrich-Universität Bamberg. Bamberg: Univ. of Bamberg Press, Seiten 53–78. URL: <http://www.opus-bayern.de/uni-bamberg/volltexte/2011/317/pdf/forFlexopusfinaleB1.pdf> (Abruf am: 21. 01. 2014) (siehe Seite 100).
- Walther, Irene (2005): „Rollen- und Situationsmodellierung bei betrieblichen Dispositions- und Planungssystemen“. Dissertation. Erlangen-Nürnberg: Friedrich-Alexander-Universität Erlangen-Nürnberg. URL: <https://opus4.kobv.de/opus4-fau/frontdoor/index/index/year/2005/docId/115> (Abruf am: 08. 04. 2016) (siehe Seite 121).
- Webster, Jane und Richard T. Watson (2002): „Analyzing the past to prepare for the future: writing a literature review“. In: *MIS Quarterly* 2.26, Seiten xiii–xxiii (siehe Seiten 57, 86).
- Wilde, Thomas und Thomas Hess (2007): „Forschungsmethoden der Wirtschaftsinformatik: Eine empirische Untersuchung“. In: *WIRTSCHAFTSINFORMATIK* 49.4, Seiten 280–287 (siehe Seite 4).
- Winter, Robert (2006): „Ein Modell zur Visualisierung der Anwendungslandschaft als Grundlage der Informationssystem-Architekturplanung“. In: *Integrationsmanagement*. Herausgegeben von Joachim Schelp und Robert Winter. Berlin und Heidelberg: Springer-Verlag, Seiten 1–29 (siehe Seite 79).
- Winter, Robert (2009): *Management von Integrationsprojekten: Konzeptionelle Grundlagen und Fallstudien aus fachlicher und IT-Sicht*. Business Engineering. Berlin und Heidelberg: Springer Berlin Heidelberg (siehe Seiten 42 f.).
- Winter, Robert und Ronny Fischer (2007): „Essential Layers, Artifacts, and Dependencies of Enterprise Architecture“. In: *Journal of Enterprise Architecture* 3.2, Seiten 7–18 (siehe Seiten 15, 39).
- Winter, Robert und Joachim Schelp (2006): „Integrationsmanagement - Notwendigkeit und Gestaltung aus Controlling-Sicht“. In: *Controlling & Management* 50.8, Seiten 22–32. URL: <http://link.springer.com/article/10.1365/s12176-006-0598-8> (siehe Seiten 4, 30).

-
- Wismans, Benedikt (1996): „FlexNow! - Ein Anwendungssystem zur Unterstützung des flexiblen Prüfungssystems an der Universität Bamberg“. In: *WIRTSCHAFTSINFORMATIK* 38.4, Seiten 441–444 (siehe Seite 8).
- Wittenburg, André (2007): „Softwarekartographie: Modelle und Methoden zur systematischen Visualisierung von Anwendungslandschaften“. Dissertation. München: Technische Universität München. URL: <http://nbn-resolving.de/urn/resolver.pl?urn:nbn:de:bvb:91-diss-20070721-625378-1-0> (Abruf am: 17. 03. 2011) (siehe Seite 35).
- Wittlage, Helmut (1998): *Unternehmensorganisation: Eine Einführung mit Fallstudien*. 6., neubearbeitete und erweiterte Auflage. NWB-Studienbücher Wirtschaftswissenschaften. Herne: Verl. Neue Wirtschafts-Briefe (siehe Seiten 19, 28).



Das Informationssystem ist für Organisationen das informationsverarbeitende System. Dieses ist unter anderem für die Planung, Steuerung sowie die Kontrolle von Geschäftsprozessen zuständig. Zur Durchführung von Geschäftsprozessen werden häufig sowohl unterschiedliche maschinelle, insbesondere Anwendungssysteme, als auch personelle Aufgabenträger eingesetzt. Dies führt dazu, dass Ressourcenebenen von Geschäftsprozessen meist heterogen geprägt sind. Damit entsteht das Problem, geeignete Integrationsarchitekturen zu konstruieren und dabei die relevanten Integrationsmerkmale der Ressourcenebene mit den spezifischen Anforderungen der Geschäftsprozesse geeignet zu kombinieren.

Für die Erschließung dieses Problemfeldes bietet die vorliegende Arbeit zunächst einen Überblick über bestehende Ansätze zum Entwurf von Integrationsarchitekturen für Informationssysteme. Bei der ersten Annäherung an das Thema ist schnell festzustellen, dass bei der Integration der Fokus häufig auf die Technologie gerichtet wird, insbesondere auf die Erstellung von Schnittstellen für die beteiligten Anwendungssysteme. Darüber hinaus wird auf Ressourcenebene neben Anwendungssystemen der personelle Anteil bei der Konstruktion von Integrationsarchitekturen oftmals unzureichend berücksichtigt. Zur Vermeidung dieser einengenden Sichtweise wird deshalb im Rahmen dieser Arbeit ein Top-Down-Ansatz entworfen. Bei diesem werden ausgehend vom Geschäftsprozess Integrationsbedarfe identifiziert und mittels Merkmalen die Anforderungen an die Integration aus Sicht der involvierten Aufgaben spezifiziert. Unter geeigneter Kombination mit der Ressourcenebene können verschiedene Szenarien miteinander verglichen werden. Grundlage des entwickelten Top-Down-Ansatzes ist das Semantische Objektmodell. Die Anwendung wird anhand von Fallstudien aus dem Hochschulbereich sowie der Güterdistribution verdeutlicht und mithilfe eines entwickelten Prototypen unterstützt.

eISBN: 978-3-86309-651-9



9 783863 096519

www.uni-bamberg.de/ubp