

publishes this series in order to make available to a broad public recent findings in informatics (i.e. computer science and information systems), to document conferences that are organized in co-operation with GI and to publish the annual GI Award dissertation.

Broken down into the fields of

- Seminars
- Proceedings
- Dissertations
- Thematics

current topics are dealt with from the fields of research and development, teaching and further training in theory and practice. The Editorial Committee uses an intensive review process in order to ensure the high level of the contributions.

The volumes are published in German or English.

Information: <http://www.gi-ev.de/service/publikationen/lni/>

ISSN 1617-5468

ISBN 978-3-88579-265-9

This volume contains contributions from the refereed "MobIS 2010" main program and selected papers of its tracks. The conference on modeling business information systems was held in Dresden September 15-17, 2010. The guiding theme for MobIS 2010 focused on modeling topics between model-based management and component and service engineering.



GI-Edition

Lecture Notes in Informatics

**Werner Esswein, Klaus Turowski,
Martin Juhrisch (Hrsg.)**

Modellierung betrieblicher Informationssysteme (MobIS 2010)

Modellgestütztes Management

**15.-17. September 2010
Dresden**

Proceedings



Modellgestütztes Management

Werner Esswein, Klaus Turowski,
Martin Juhrisch (Hrsg.)

MobIS 2010

Modellierung betrieblicher Informationssysteme

Modellgestütztes Management

**15.-17. September 2010
in Dresden, Germany**

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Proceedings
Series of the Gesellschaft für Informatik (GI)

Volume P-171

ISBN 978-3-88579-265-9

ISSN 1617-5468

Volume Editors

Prof. Dr. Werner Esswein

Technische Universität Dresden, Wirtschaftsinformatik, insb. Systementwicklung
01062 Dresden, Germany

Email: werner.esswein@tu-dresden.de

Prof. Dr. Klaus Turowski

Universität Augsburg, Wirtschaftsinformatik und Systems Engineering
Universitätsstr. 16, 86159 Augsburg, Germany

Email: klaus.turowski@wiwi.uni-augsburg.de

Dr. Martin Juhrisch

Technische Universität Dresden, Wirtschaftsinformatik, insb. Systementwicklung
01062 Dresden, Germany

Email: martin.juhrisch@tu-dresden.de

Series Editorial Board

Heinrich C. Mayr, Universität Klagenfurt, Austria (Chairman, mayr@ifit.uni-klu.ac.at)

Hinrich Bonin, Leuphana-Universität Lüneburg, Germany

Dieter Fellner, Technische Universität Darmstadt, Germany

Ulrich Flegel, SAP Research, Germany

Ulrich Frank, Universität Duisburg-Essen, Germany

Johann-Christoph Freytag, Humboldt-Universität Berlin, Germany

Thomas Roth-Berghofer, DFKI

Michael Goedicke, Universität Duisburg-Essen

Ralf Hofestädt, Universität Bielefeld

Michael Koch, Universität der Bundeswehr, München, Germany

Axel Lehmann, Universität der Bundeswehr München, Germany

Ernst W. Mayr, Technische Universität München, Germany

Sigrid Schubert, Universität Siegen, Germany

Martin Warnke, Leuphana-Universität Lüneburg, German

Dissertations

Dorothea Wagner, Universität Karlsruhe, Germany

Seminars

Reinhard Wilhelm, Universität des Saarlandes, Germany

Thematics

Andreas Oberweis, Universität Karlsruhe (TH)

Gesellschaft für Informatik, Bonn 2010

printed by Köllen Druck+Verlag GmbH, Bonn

Veranstalter

Technische Universität Dresden
Fakultät Wirtschaftswissenschaften
Lehrstuhl für Wirtschaftsinformatik, insb. Systementwicklung
01062 Dresden, Germany

Prof. Dr. Werner Esswein (Tagungsleitung und Vorsitzender des Programmkomitees)
Dr. Martin Juhrisch (Organisation)

Die Tagung wurde von der Gesellschaft für Informatik, Fachgruppe Informationssystem-Architekturen: Modellierung betrieblicher Informationssysteme (WI-MobIS) gemeinsam mit dem Arbeitskreisen Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten (WI-EPK) und Komponentenorientierte betriebliche Anwendungssysteme (WI-KobAS) sowie dem Lehrstuhl für Wirtschaftsinformatik, insb. Systementwicklung der Technischen Universität Dresden organisiert.

Programmkomitee

Prof. Dr. Jörg Becker, Universität Münster
Dr. Patrick Delfmann, Universität Münster
Dr. Gunnar Dietz, Dongbei University
Prof. Dr. Werner Esswein, TU Dresden
Prof. Dr. Ulrich Frank, Universität Duisburg-Essen
Prof. Dr. Martin Gaedke, TU Chemnitz
Prof. Dr. Steffen Greiffenberg, Universität Cottbus
Prof. Dr. Norbert Gronau, Universität Potsdam
Prof. Dr. Andreas Hilbert, TU Dresden
Dr. Martin Juhrisch, TU Dresden
Prof. Dr. Reinhard Jung, Universität St. Gallen
Prof. Dr. Stefan Kirn, Universität Hohenheim
Prof. Dr. Gerhard Knolmayer, Universität Bern
Prof. Dr. Helmut Krcmar, TU München
Prof. Dr. Peter Loos, Universität des Saarlandes
Prof. Dr. Jan Mendling, HU Berlin
Prof. Dr. Volker Nissen, TU Ilmenau
Prof. Dr. Markus Nüttgens, Universität Hamburg
Prof. Dr. Andreas Oberweis, Universität Karlsruhe
Dr. Sven Overhage, Universität Augsburg
Prof. Dr. Michael Rebstock, Hochschule Darmstadt
Prof. Dr. Frank Rump, Fachhochschule Emden/Leer
Prof. Dr. Carlo Simon, Provadis Hochschule
Prof. Dr. Elmar Sinz, Universität Bamberg
Prof. Dr. Susanne Strahringer, TU Dresden
Dr. Oliver Thomas, Universität Osnabrück
Prof. Dr. Klaus Turowski, Universität Augsburg
Prof. Dr. Jan vom Brocke, Universität Lichtenstein
Prof. Dr. Mathias Weske, Universität Potsdam
Prof. Dr. Robert Winter, Universität St. Gallen

Vorwort

Die MobIS-Jahrestagung kann dieses Jahr bereits auf eine Historie von 12 Vorgänger- tagungen zurückblicken. Als bedeutende Arbeitstagung zum Thema Modellierung im Bereich der Wirtschaftsinformatik führt sie die Wissenschaftler und Anwender in über- schaubarem Kreis zusammen. Aus dieser kreativen Atmosphäre heraus sind immer wie- der neue Ideen und Zusammenarbeiten entstanden – wir hoffen, dass dies bei dieser Tagung wieder der Fall sein wird.

Die Schwerpunkte der MobIS 2010 sind die Themen „Modellgestütztes Management“ sowie die aktuellen Ergebnisse der Arbeitsgruppe „Komponentenorientierte betriebliche Anwendungssysteme“.

Im ersten Schwerpunkt finden sich sowohl Beiträge, welche sich mit allgemeinen Mo- dellierungsfragen beschäftigen (Trojahnner et al., Weller et al.), sowie Arbeiten, die sich dem Modellvergleich, Operationen auf Modellen sowie Modellen zur Überwindung der semantischen Lücke widmen (Fengel u. Rebstock, Fellmann et al.), als auch Beiträge mit starkem Bezug zur Anwendungsdomäne (Strecker et al., Deindl et al.).

Hierbei fällt auf, dass auch technische Anwendungsdomänen in den Fokus der Wirt- schaftsinformatik gelangen. Hierzu passt der Beitrag von Sadek und Meuris aus dem zweiten Schwerpunkt. Die weiteren Beiträge dieses Schwerpunkts zeigen, dass Kompo- nenten und die moderneren Services enge Verwandtschaftsbeziehungen besitzen.

Ingesamt wurden 32 Beiträge zur Tagung eingereicht. Von diesen wurden 13 Beiträge zur Publikation im Tagungsband akzeptiert, was zu einer Annahmequote von knapp 40% führte.

Wir danken an dieser Stelle allen Autoren für ihre Einreichungen. Dank gebührt ebenso den Mitgliedern der Programmkomitees, die teilweise sehr umfangreich in die Begutach- tung von Beiträgen eingebunden waren. Ganz herzlich bedanken möchten wir uns insbes-ondere bei den Personen, die an der Organisation der Tagung und an der Erstellung dieses Buchs unmittelbar mitgewirkt haben. Zu erwähnen sind hier Frau Jeannette Stark und Herr Hannes Schlieter sowie alle beteiligten studentischen Mitarbeiter am Lehrstuhl für Wirtschaftsinformatik, insb. Systementwicklung der TU Dresden.

Inhaltsverzeichnis

Vorwort.....	7
Modellgestütztes Management	
Esswein Werner, Stark Jeannette, Schlieter Hannes	
The selection of modeling grammars.....	13
Trojahnner Iris, Weller Jens, Fürstenau Bärbel, Esswein Werner	
Der Einfluss von Modellierungswerkzeugen auf Qualität und Quantität von Modellen – Eine empirische Untersuchung.....	29
Weller Jens, Helbig Michaela, Großmann Knut	
Eine Methode für den praktischen Einsatz von Konfigurationsmanagement in verteilten Modellierungsprojekten.....	47
Fengel Janina, Rebstock Michael	
Domänensemantik-orientierte Integration heterogener konzeptueller Modelle.....	63
Bögel Stephan, Esswein Werner	
Vertikale Modellintegration in Rahmenwerken – Evaluation von Integrationsansätzen und Untersuchung der Implementierbarkeit anhand eines Fallbeispiels.....	79
Fellmann Michael, Hogreve Frank, Thomas Oliver, Nüttgens Markus	
An ontology-driven approach to support semantic verification in business process modeling.....	99
Juhrisch Martin, Dietz Gunnar	
Context-based Modeling: Introducing a Novel Modeling Approach.....	111
Strecker Stefan, Heise David, Frank Ulrich	
Toward modeling constructs for audit risk assessment: Reflections on internal controls modeling.....	131
Schlieter Hannes, Bürger Maik, Esswein Werner	
Konstruktion eines adaptiven Referenzmodells für den ambulanten Sektor.....	149

Deindl Matthias, Naß Eric, Beckhoff Tim Anwendung der enhanced Telecom Operations Map auf Dienstleistungsunternehmen in der Stromwirtschaft am Beispiel eines Prozesses im Bereich Messwesen.....	171
Komponentenorientierte betriebliche Anwendungssysteme	
Schrödl Holger Service- und komponentenorientierte Informationssystemarchitekturen für die strategische Beschaffung von hybriden Produkten – ein Vergleichsrahmen.....	195
Houy Constantin, Reiter Markus, Fettke Peter, Loos Peter Potentiale serviceorientierter Architekturen für Software-Werkzeuge des Geschäftsprozessmanagements.....	211
Sadek Tim, Meuris Daniel Datentechnische Integration und Visualisierung von Anforderungen innerhalb von CAD-Systemen.....	229

The selection of modeling grammars

Werner Esswein, Jeannette Stark, Hannes Schlieter

Lehrstuhl für Wirtschaftsinformatik, insb. Systementwicklung

Technische Universität Dresden

Schumann-Bau, Münchner Platz

01062, Dresden

{werner.esswein, jeannette.stark, hannes.schlieter}@tu-dresden.de

Abstract: Information systems developers such as business process engineers use Conceptual Modeling to understand the information system's application domain and to derive such a system's requirements. For Conceptual Modeling an appropriate modeling grammar needs to be selected. Approaches have been identified to evaluate and select a modeling grammar. It is supposed that theory-based approaches contribute to a more effective selection of modeling grammars than non-theory-based approaches. However, only two approaches that are theory-based identified lead to a single modeling grammar: Cognitive Fit and Ontology. Using both approaches as complements results more effective to solve a specific (conceptual) modeling task than the individual approaches do. Therefore, we combine both approaches to a procedure that facilitates a theory-based approach to select a modeling grammar.

1 Introduction

The information system's (IS) purpose to manage information about its application domain can only be fulfilled if such a system appropriately reflects this information ([JMSV92]). IS developers use Conceptual Modeling to represent this information by composing the modeling grammar's constructs to a conceptual model ([WW02]). Depending on which type of information they intend to represent, IS developers solve a specific modeling task. To solve this task, a conceptual modeling grammar needs to be selected. However, a variety of grammars has emerged ([AF06]; [MKK02]). Some of these grammars, such as Business Process Modeling Language (BPML), claim to be advantageous to model processes. Other modeling grammars, such as the Unified Modeling Language (UML) do not specialize but propose to be advantageous to model every phenomenon of an application domain ([ES07]). [SR98] have identified in [RB96] seven approaches to evaluate and select a modeling grammar: *Feature Comparison*, *Metamodeling*, *Metrics*, *Contingency Identification*, *Paradigmatic Analyses*, *Ontology* and *Cognitive Fit*.

Each of the seven approaches contributes to the selection of a modeling grammar, but also reveals certain disadvantages. For Feature Comparison metrics have to be developed and applied. Both activities are very subjective ([RB96]). Metamodeling allows a less subjective evaluation of modeling grammars, but is limited to a pair wise comparison of a modeling grammar's constructs ([RB96]). Paradigmatic Analysis enables IS developers to reduce the number of applicable modeling grammars, but not to select a single one. To apply metrics or heuristics of Contingency Identification much empirical work is needed ([RB96]). Selecting a modeling grammar with Ontology is characterized by two gaps. Firstly, evaluations of modeling grammars with Ontology do not base on the same ontological constructs. Selecting a modeling grammar requires to compare these evaluations. However, if evaluations are not based on the same ontological constructs, they can't be compared ([GE07]). Secondly, IS development and consequently conceptual modeling are activities to solve problems ([VG98]). Therefore, the selection of a modeling grammar needs to be based on a modeling problem. The problem, however, is not yet considered in prior research of Ontology. Cognitive Fit allows combining the selection of a modeling grammar with the problem the modeling grammar is intended to solve. Cognitive Fit, though, lacks empirical evaluations of modeling grammars that can be used to select a modeling grammar.

Prior research supposes that theory-based approaches contribute to a more effective selection of modeling grammars than non-theory-based approaches ([Wa95]). Paradigmatic Analyses, Ontology and Cognitive Fit are theory-based. Yet, Paradigmatic Analysis does not allow selecting a single modeling grammar. Therefore, we base our procedure on Ontology and Cognitive Fit, using both approaches as complements to facilitate a more effective solution of a specific conceptual modeling task than the individual approaches do.

This study contributes to theory and practice. To theory, this study relates two different fields of research to facilitate a theory-based selection of a single modeling grammar. To practice the approach contributes a mechanism to select a modeling grammar to solve a specific modeling task, whose application does not require practitioners to be confident with Ontology or Cognitive Fit. Once this framework is established, practitioners just need to consider their modeling task to be given a hierarchy of modeling grammars.

This paper is organized as follows. We first introduce the theoretical background of the framework. The second section, will give an overview about Cognitive Fit and Ontology. In the third section, we explain the framework in detail and derive tasks to operationalize the approach. The results of an exemplary operationalization are presented in section four. Section five summarises the research contributions.

2 Theory

Our approach bases on the theory of Cognitive Fit and is operationalized through Ontology. We first explain how to apply Cognitive Fit to Conceptual Modeling and will continue how to use Ontology to operationalize the approach.

2.1 Cognitive Fit

System development is a problem solving activity ([VG98]). These activities are researched in Cognitive Psychology. [NS72] have developed a theory, in which humans are considered to be information-processing systems. Yet, people's information-processing capacity is limited. Therefore people seek to reduce the effort that is necessary to solve a problem. One way to reduce this effort is to simplify the problem-solving process. This process is simplified if the problem corresponds to the task that is intended to be solved, a state that Vessey calls Cognitive Fit. Within the problem-solving process the problem solver develops a mental representation of a possible problem solution. This representation is then transformed into a real problem solution. As shown within the basic model of figure 1 the degree of how information of the problem representation and of the problem-solving task corresponds, determines how efficient the problem solver can develop a mental representation, and thus can solve the problem ([VG98]).

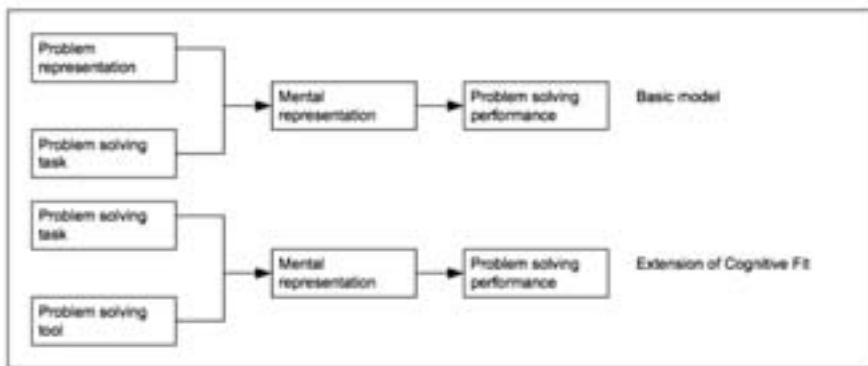


Figure 1: Basis model and extension of Cognitive Fit

[AS96] have extended the basic model for the problem-solving tool. The effort to solve a task is reduced if a Cognitive Fit occurs between the task and the method used to solve the task. We apply the extension of [AS96] to Conceptual Modeling. Correspondingly, the problem-solving task is specified to a modeling task and the problem-solving tool to a conceptual modeling grammar as shown in figure 2.

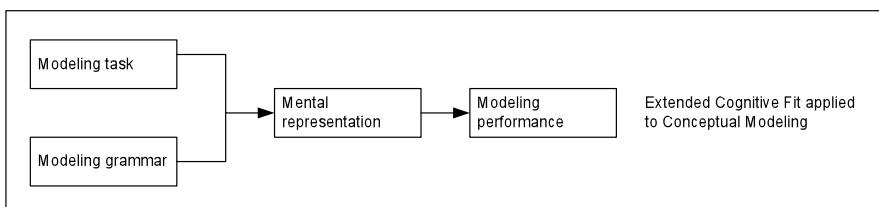


Figure 2: Extended Cognitive Fit applied to Conceptual Modeling

We use the extension of the Cognitive Fit of [AS96] applied to Conceptual Modeling as a theoretical basis for the framework. The framework will be operationalized through Ontology.

2.2 Ontology

Like conceptual modeling grammars, Ontology provides constructs and rules to describe real world phenomena ([MKK02]). Yet, Ontology and conceptual modeling grammars differ in intensity of prior research. Scientists have studied Ontology since Aristotle's time while research on Conceptual Modeling is comparably young. Prior research has used experience of Ontology for Conceptual Modeling by relating grammatical to ontological constructs. We use the evaluations of modeling grammars with the ontology of Bunge for our framework. This ontology is selected because a wide range of conceptual modeling grammars has already been evaluated with this ontology ([DGR04]) while other ontologies such as the ontology of Chisholm are only used to derive rules for Conceptual Modeling but not to evaluate its grammars.

According to the ontology of BUNGE, the world is made of Things. Things exist individually or in Composition. Things possess Properties. While intrinsic Properties characterize one Thing, mutual Properties characterize a Composition. Things and Properties are the bases to derive further static and dynamic constructs ([Wa95]). Static constructs such as Kind and Class can be defined by mutual Properties. While Classes are defined by one mutual Property, Kinds need at least two Mutual Properties. Dynamic constructs are, for example, State, Event and Interaction. The State is determined by the values of all Properties. The change of a Property of a Thing, and thus the change of a State, is called Event. An Event is ruled by Laws, which are specialized Properties of a Thing. Presented constructs of the ontology of BUNGE are summarized in a metamodel by ([GRIR06]).

[WW02] identified interfaces between the ontology of Bunge and Conceptual Modeling. Their achievements are now referenced with the Bunge-Wand-Weber model (hereafter BWW model). The BWW model consists of three models: The State Tracking-, the Representation-, and the Good Decomposition Model (GDM). In more than 25 studies the BWW model is used to evaluate and to improve modeling grammars ([RRI06]). However, most of these studies have focused on the Representation Model and so far only little attention has been paid to the GDM and the State Tracking Model ([GP07]). Due to the strong focus on the Representation Model this study is limited to this model, too. This model relates ontological to grammatical constructs through ontological analysis.

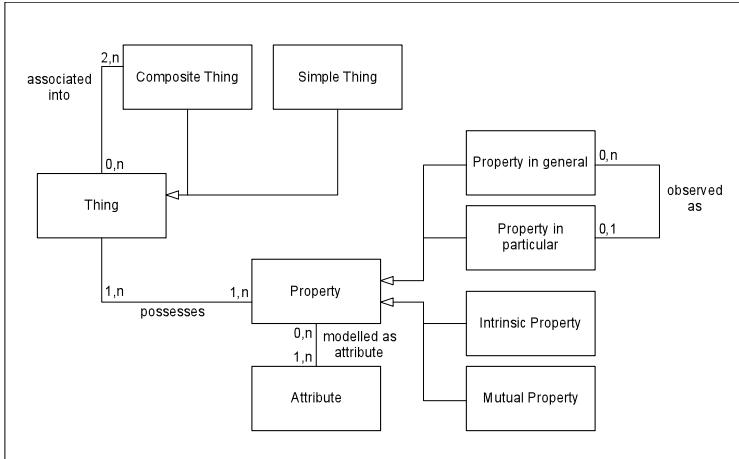


Figure 3: Extract from the Bunge-Metamodel (vgl. [GRIR06])

Ontological analysis focuses on revealing ontological deficits of conceptual modeling grammars. Four ontological deficits can be found ([We97]):

1. Ontological Incompleteness exists if at least one ontological construct does not have a grammatical equivalent.
2. Construct Redundancy exists if at least two grammatical constructs have one or more mutual ontological equivalents.
3. Construct Overload exists if at least one grammatical construct has more than one ontological equivalent.
4. Construct Excess exists if at least one grammatical construct does not have an ontological equivalent.

If ontological incompleteness is identified, the modeling grammar does not have sufficient constructs to model every possible phenomenon of a domain. If constructs of a modeling grammar are overloaded, redundant or do not have an ontological equivalent, the grammar's ability to create comprehensive models is affected. In this case, the modeling grammar is not ontologically clear ([GRIR06]). We use eight ontological analyses to operationalize our framework. Of these analyses, one focuses on the Architecture of integrated Information Systems (ARIS) ([GR00]) three on the Entity Relationship Diagram (ERD) ([We97]; [WSW99]; [Sh03]), one on the Open Modeling Language (OML) ([OH01]) and three on the Unified Modeling Language (UML) ([WSW99]; [IT05]; [OH02]).

3 Framework

Our framework bases on the theory of Cognitive Fit and is operationalized through Ontology. For its operationalization required grammatical constructs to solve the modeling task (operationalized from the modeling objective) and those provided by the modeling grammars are mapped on BWW constructs. Mapping provided and required constructs on BWW constructs allows determining their intersection. Constructs of the intersection will be assessed according to two criteria:

1. Maximum number of provided and needed constructs within the intersection.
2. Maximum degree of ontological clarity of the constructs within the intersection.

The modeling grammar that best fulfills both criteria is most likely to allow modeling relevant real world phenomena in a comprehensive way. Figure 4 provides an overview of the framework and derives three tasks to its development.

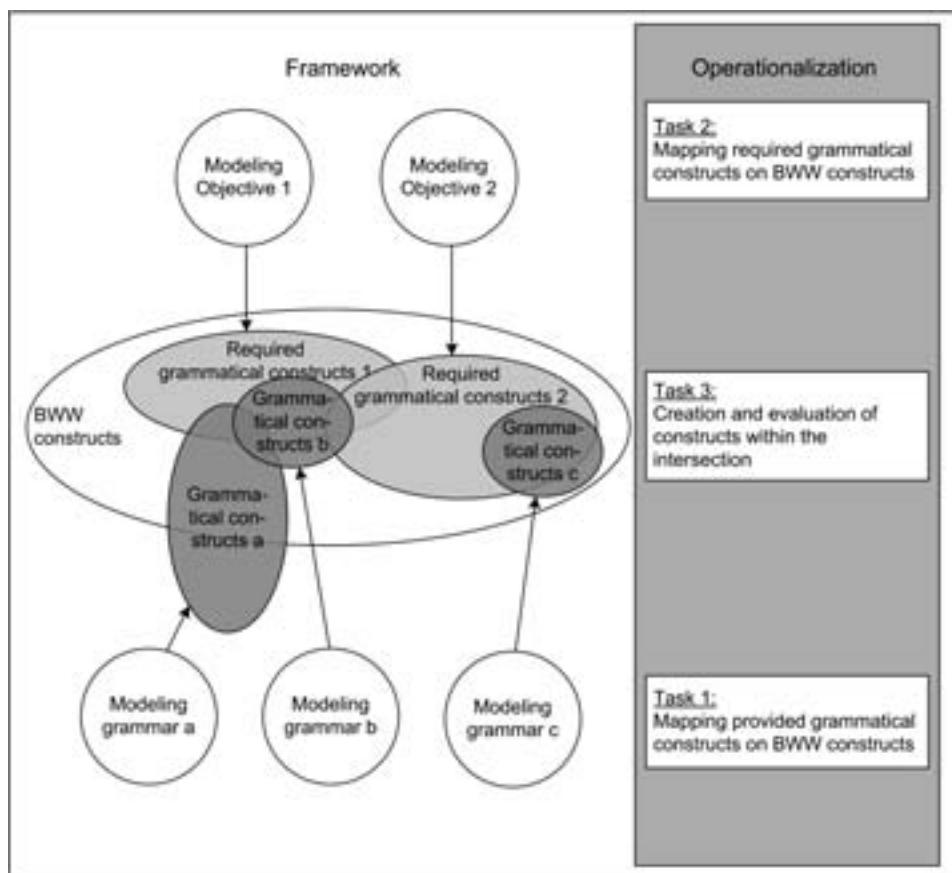


Figure 4: Framework and derivation of tasks to create the framework

Firstly, provided and required grammatical constructs are mapped on BWW constructs. Therefore, ontological analyses will be used, as they already provide relations between grammatical and ontological constructs. For that reason, this study is limited to modeling grammars presented in section 2. Secondly, required constructs to solve a modeling goal are extracted. Thereto, modeling tasks will be researched within a content analysis using the framework of ([Kr04]). The sample is limited to the IS-world magazines, since latest research primarily used this medium to publish research results. Having researched modeling tasks, these tasks will be accumulated to a more comprehensive framework of tasks. As not every modeling task might be relevant, their practice relevance will be assessed. Therefore, the classification of reference models by ([FL03]) is used. Reference models accelerate conceptual modeling by serving as a pattern to create specific models ([RA07]). Correspondingly, a modeling task is relevant if this task is performed as many times, that the creation of a reference model is useful. After limiting modeling tasks to relevant modeling tasks, reference models classified by [FL03] are used to derive grammatical constructs necessary to solve the modeling task. These constructs are then mapped on BWW constructs. Thirdly, the intersection of provided and required ontological constructs is created. The intersection of these constructs indicates to which degree modeling grammars offer constructs to solve the modeling tasks. These constructs will be assessed towards ontological clarity. The degree to which modeling constructs are available and ontologically clear indicates how appropriate a modeling grammar is for a task. Since the appropriateness of several modeling grammars is assessed a hierarchy will be established. For the subsequent exemplification of the framework for modeling grammars are used. These grammars include Elements of the ARIS-Framework [Sch94], ERD [Ch76], UML [RJB04] and OML [FHG98] and are selected as they are considered in available ontological analyses that serve as a basis for this exemplary operationalisation.

4 Results of exemplary operationalisation

Solving the first task relations between grammatical and ontological evaluations are extracted from ontological analyses. To solve the second task a content analysis was conducted. The content analysis reveals three different frameworks of modeling tasks. [JMSV92] distinguish four worlds of modeling task: Subject-, System-, Usage- and Development world modeling. Subject world modeling describes the information the IS maintains. How this information is implemented is described with System world modeling. The IS is implemented in the system's environment. This environment, such as agents, activities, and users is described in Usage world modeling. [JMSV92] do not only focus on the IS system and its environment but also on how it is developed. Therefore, they use Development World modeling.

[Ly87] distinguishes three main categories: IS-, IS environment- and IS context modeling. IS modeling describes the structure and behavior of an IS. As the second main category, IS environment modeling describes the interaction of an IS with its environment. IS environment modeling includes modeling IS architecture, information need from the organizational point of view, success factors of the IS project, Sociotechnical issues and an evaluation. Lyytinen's third category, IS context modeling focuses on contingencies and their impact on the IS development strategy. [AAGS06] distinguish static and dynamic information modeling. Static information modeling describes structural information such as concepts, properties and constraints. They call this modeling declarative information modeling. Dynamic information modeling describes behavior. It includes strategic and tactical information modeling. While strategic information modeling describes problem-solving activities such as what is to do, and in what order, tactical information modeling focuses on describing how and when to include declarative information. To apply the modeling tasks of [JMSV92], [Ly87] and [AAGS06] to the framework, the modeling tasks are first aggregated and subsequently assessed regarding their relevance in practice. Relevance in practice of modeling tasks is indicated through the number of reference models allocated to the tasks.

Table 1: Modeling Task and allocation of reference models to these tasks

Modeling task [JMSV92]	Modeling task [Ly87] and [AAGS06]	Explanation	Reference Modeling Literature (author, pages)
Subject World Modeling	Static Information Modeling	Describing the structural information such as concepts, properties and constraints	[Me00], 31 [Sche94], 91, 163 [Jo92], 131, 209 [Lo97], 106 [GDV08], 7, 49, 52, 69
	Dynamic information modeling	Describing the behavior that takes place in the domain such as functionality, processes, inferences, calculations and action sequences	[BS96], 150 [Re97], 143, 183, 189 [Me00], 88, 129 [Jo92], 69, 131 [GDV08], 11, 12, 13, 45
System World Modeling	IS-Architecture Modeling	Describing the overall architecture for an organization by identifying major applications	
	Formalistic IS Modeling	Describing the static structure and IS dynamics. Static structure is expressing in terms of entities, properties, and relationships. IS dynamics is expressed as triggers and event that change the data	[BS96], 127, 192 [Me00], 20 [Sche94], 150 [Lo92], 121 [Ku99], 204 [Schw99], 193 [Lo97], 116 [We93], 102
	Functional IS Modeling	Describing IS in terms of connected activities that process, transform, store, access, and modify data. Functional IS models are therefore process orientated and originate from the design of the systems for processing	[BS96], 164 [Sche94], 160 [KT98], 300-302 [Schw99], 195

		documents presented in files of records	[Lo97], 112
Usage World Modeling	Socio-technical Modeling	Describing the work organization by incorporating features of the IS, user, and organizational environments	
Development World Modeling	Success Factor Modeling	Describing organizational characteristics such as computer experience to predict IS implementation risk	
	Evaluation Modeling	Describing technical and cost-efficiency criteria	
	Contingency Modeling	Describing contingencies and impact on the IS development strategy	
	Strategical Information Modeling	Describing problem-solving activities such as what to do, when, and in what order	[Sche94], 609
	Tactical Information Modeling	Describing how and when to include the new declarative information about the problem	[Sche94], 683, 684

According to table 1 only four modeling task result as practice relevant: Static Information Modeling, Dynamic Information Modeling, Formalistic IS Modeling, and Functional IS Modeling. Of these modeling tasks constructs are extracted from the reference models allocated to the modeling tasks as seen in table 2. Therefore, grammatical constructs were first extracted and then translated to ontological constructs with the help of already established relations between grammatical and ontological constructs in existing ontological analyses. Figures in Table 2 stand for the number of reference models that include a certain BWW-Construct.

Table 2: Allocation of ontological constructs to modeling tasks

BWW Constructs	Static Information Modeling	Dynamic Information Modeling	Formalistic IS Modeling	Functional IS Modeling	Strategic Information Modeling	Tactical Information Modeling
Binding Mutual Property			4			
Class	6	1	6	1	1	
Coupling	2		6		1	
Event		4	1	4		

External Event	1	5	1	4		1
Internal Event		4	1	4		
Intrinsic Property	3		6		1	
Intrinsic State Law	2	1	1			
Kind	9	5	7	1		
Lawful State Space	1		4			
Level Structure	2	4		1		1
Mutual Property	3	2	6			
Natural Kind/Subkind Rel.		2	1			
Non-binding Mutual Property	2	2	1			
Process	2	10	1	5		1
Property	5	7	3	5		2
Property in General	3	2	5	1		
Stable State	1	5	1	4		1
State		4	1	4		1
State Law	3	2	8	4		1
Subkind			1			
Subsystem	2	3				1
System						1
System Environment		1				
Thing	3	4	5			2
Transformation	2	8	2	5		1
Well-defined Event		4	1	4		
Whole-Part Relationship	3		1			

Having extracted required and provided constructs the intersection of these constructs will be determined and evaluated. Therefore a scoring model is used with scores ranging from 0-5. If the required ontological construct has an ontologically clear equivalent within the grammar the correspondent grammatical construct is given a 0. Score 1 is given if the required ontological construct is characterized by construct overload or redundancy. If both, construct overload and redundancy occur, the correspondent construct is given a 2. Since the model is more affected if there is no grammatical equivalent for a required ontological construct, the modeling grammar is given a 5 in that case. Some ontological constructs are more important to solve a modeling task than others. That way, to solve a Static Information Modeling task it is more important that a modeling grammar has a grammatical equivalent for classes than for Lawful State Spaces or State Spaces. To solve this problem the points given to a certain construct is weighted by the number of how many time the construct was needed in the reference models.

Table 3: Scoring model

BWW-Construct	Static Information Modeling					Dynamic Information Modeling					Formalistic Information Modeling					Functional Information Modeling				
	W E I G H T	E R D I S	A R I G H T	U M L	O M L	W E I G H T	E R D I S	A R I G H T	U M L	O M L	W E I G H T	E R D I S	A R I G H T	U M L	O M L	W E I G H T	E R D I S	A R I G H T	U M L	O M L
Modeling grammar																				
B. Mutual Property											4	2	5	1	2					
Class	6	2	1	2	5	1	2	1	2	5	6	2	1	2	5					
Coupling	2	2	0	0	5						6	2	2	0	5	1	2	0	0	5
Event						4	5	0	2	2	1	5	0	2	2	4	5	0	2	2
External Event	1	5	0	2	5	5	5	0	2	5	1	5	0	2	5	4	5	0	1	5
Internal Event						4	5	0	1	5	1	5	0	1	5	1	0	5	2	5
Intrinsic Property	3	2	1	1	2						6	2	1	1	2					
Intrinsic State Law	2	5	5	1	5	1	5	5	1	5	1	5	5	1	5					
Kind	9	0	5	2	5	5	0	5	2	5	7	0	5	2	5	1	0	5	2	5
Lawful State Space	1	2	5	5	5						4	2	5	5	5					
Level Structure	2	5	0	5	5	4	5	0	5	5						1	5	0	5	5
Mutual Property	3	2	2	2	2	2	2	2	2	2	6	2	2	2	2					
Natural K./Subk. R.						2	5	5	0	0	1	5	5	0	0					

N.-b. Mut. Property	2	5	5	0	0	2	5	5	0	0	1	5	5	5	0	0						
Process	2	5	2	2	1	10	5	2	2	1	1	5	2	2	1	5	5	2	2	1		
Property	5	2	1	2	2	7	2	1	2	2	3	2	1	2	2	5	2	1	2	2		
Property in General	3	2	2	5	5	2	2	2	5	5	5	2	2	5	5	1	2	2	5	5		
Stable State	1	5	0	0	5	5	5	0	0	5	1	5	0	0	5	4	5	0	0	5		
State						4	5	0	1	0	1	5	0	1	0	4	5	0	1	0		
State Law	3	2	1	0	2	2	2	1	0	1	8	2	1	0	2	4	2	1	0	2		
Subkind											1	5	5	0	0							
Subsystem	2	5	1	5	5	3	5	1	5	5												
System Environment						1	5	0	1	2												
Thing	3	2	1	1	2	4	2	1	1	2	5	2	1	1	2							
Transformation	2	5	2	0	2	8	5	2	0	2	2	5	2	0	2	5	5	2	0	2		
Well-defined Event						4	5	0	5	5	1	5	5	5	5	4	5	5	1	2		
Whole-Part Relation	3	5	5	1	2						1	5	5	1	2							
Results	55	26	23	1,9	3,5	80	3,5	1,4	1,7	3,0	70	2,4	2,4	1,8	3,3	47	4,2	1,2	1,3	1,3		27

According to the Scoring Model, modeling grammars with lower scores offer more required construct that lead to more comprehensive models while modeling grammars that exhibit higher scores are characterized by a lack of constructs needed to solve the task and/or by an abundance of ontologically unclear constructs. The allocation of scores to the modeling grammars for each modeling task allows deriving a hierarchy of modeling grammars. The resulting hierarchy is presented in table 4.

Table 4: Hierarchy of modeling tasks

Modeling tasks / Hierarchy	Static Information Modeling	Dynamic Information Modeling	Formalistic IS Modeling	Functional IS Modeling
1.	UML	ARIS	UML	ARIS
2.	ARIS	UML	ERD	UML
3.	ERD	OML	ARIS	OML
4.	OML	ERD	OML	ERD

Except for one modeling task, Formalistic IS Modeling, the modeling grammars of the ARIS-Framework and UML alternate within the first two ranks of the hierarchy while ERD and OML alternate within rank three and four.

5 Conclusion

Prior research does not allow selecting a modeling grammar in a theory-based way. In this study a framework is developed that enables a theory-based selection of a modeling grammar. Cognitive Fit is used as a theoretical basis. If a Cognitive Fit occurs between a modeling grammar and a modeling task, the grammar allows creating the same type of information that the modeling task requires. This way, the modeling task can be solved efficiently. Using Cognitive Fit as a theoretical basis we benefited from relating Conceptual Modeling to a modeling task. We further benefited in that we did not need to compare ontological analyses directly but used them as a tool to combine required constructs of a modeling task with the provided constructs of a modeling grammar. Directly comparing ontological analyses so far has not been possible since each analysis bases on different constructs. For the operationalization of the framework Ontology is used. Using Ontology, we benefited from already existing ontological analyses, which provide relations between ontological and grammatical constructs. In addition, Ontology enabled us to assess modeling grammars for their capability to trace real world phenomena. By using Cognitive Fit as a theoretical basis and Ontology for its operationalization we could include benefits and avoid gaps inherent to each of the theory. The conception of the framework is only done exemplarily and we don't claim it to be complete. Therefore, a content analysis was used to determine modeling task and reference models were used to assess their relevance in practice. According to the framework, modeling grammars were allocated to the modeling task and a hierarchy of these grammars could be established. Further research still need to be done to assess the concepts of the Bunge-Ontology to evaluate modeling grammars, since there are still concepts that are hardly possible to map onto this ontology. The operationalization of the framework serves as a basis to test the framework empirically with a laboratory experiment within future research.

References

- [AAGS06] Andrade, J., Ares, J., Garcia, R., Pazos, J., Silva, A. (2006) Definition of a problem-sensitive conceptual modeling language: foundations and application to software engineering. *Information and Technology* 48, 517-531.
- [AF06] Avison, D.E., Fitzgerald, G. (2003) Where Now for Development Methodologies? *Communications of the ACM* 46, 79-82.
- [AS96] Agarwal, R., Sinha, A.P. (1996) The role of prior experience and task characteristic in object-oriented modeling: an empirical study. *Human-Computer Studies* 45, 639-667.
- [BRU00] Becker, J., Rosemann, M., v. Uthmann, C. (2000) Guidelines of Business process modeling. In: v.d. Aalst, W., Desel, J., Oberweis, A. (eds): Business Process Management – Models, Techniques, and Empirical Studies. Springer, London 30-49.
- [BS96] Becker, J., Schütte, R. (1996) Handels-Informations-Systeme. Verlag Moderne Industrie, Landsberg.
- [CH76] Chen, Ps.S.: The Entity-Relationship Model – Toward a Unified View of Data. In: Transaction on Database Systems 1, 1976.
- [DGR04] Davies, I., Green, P., Rosemann, M. (2004) Exploring proposed Ontological Issues of ARIS with Four different Types of Modellers. *Proceedings of the Australasian Conference on Information Systems*.
- [ES07] Erickson, J., Siau, K. (2007) Theoretical and Practical Complexity of Modeling Methods. *Communications of the ACM* 50, 46-51.
- [FKPT03] Fahrmeier, L., Künstler, R., Pigeot, I., Tutz, G. (2003) Der Weg zur Datenanalyse. vol. 4, Berlin, Springer.
- [FL03] Fettke, P., Loos, P. (2003) Classification of reference models: a methodology and its application. *Information Systems and e-Business Management* 1, 35-53.
- [GDV08] GDV (2008) Die Anwendungsarchitektur der deutschen Versicherungswirtschaft. <http://www.gdv-online.de/vaa/>.
- [GE07] Gehlert, A., Esswein, W. (2007) Toward a formal research framework for ontological analyses. *Advanced Engineering Informatics* 21, 119-131.
- [GP07] Gehlert, A., Pfeiffer, D. (2007) The BWW-Model as Method Engineering Theory. *Proceedings of the Americas Conference on Information Systems*.
- [GR00] Green, P., Rosemann, M. (2000) Integrated Process Modeling: An ontological Evaluation. *Information Systems* 25, 73-87.
- [GRIR06] Green, P., Rosemann, M., Indulska, M., Recker, J. (2006) Improving Representational Analysis: An Example from the Enterprise Systems Interoperability Domain. *Proceedings of the 17th Australasian Conference on Information Systems*.
- [IT05] Irwin, G., Turk, D. (2005) An Ontological Analysis of Use Case Modeling Grammars. *J. of the Association for Information Systems* 6, 1-36.
- [FHG98] Firesmith, F. G., Henderson-Sellers, B., Graham, I.: The Open Modeling Language (OML) reference manual, Cambridge University Press, 1998.
- [JMSV92] Jarke, M., Mylopoulos, J., Schmidt, J., Vassilou, Y. (1992) DAIDA: An environment for evolving information systems. *ACM Transaction on Information Systems* 10, 1-50.
- [Jo92] Jost, W. (1992) EDV-gestützte CIM-Rahmenplanung. Gabler, Wiesbaden.

- [Kr04] Krippendorff, K. (2004) Content analysis: An Introduction to its Methodology. Sage
- [KT98] Keller, G., Teufel, T. (1998) SAP R/3 Process Oriented Implementation – Iterative Process Prototyping. Harlow.
- [Ku99] Kurbel, K. (1999) Produktionsplanung und –steuerung. Methodische Grundlagen von PPS-Systemen und Erweiterungen. vol. 4, Oldenbourg, München.
- [Lo92] Loos, P. (1992) Datenstrukturierung in der Fertigung. Oldenbourg, München.
- [Lo97] Loos, P. (1997): Produktionslogistik in der chemischen Industrie – Betriebstypologische Merkmale und Informationsstrukturen. Wiesbaden, Gabler.
- [LSS94] Lindland O.I., Sindre, G., Solvberg, A. (1994) Understanding quality in conceptual modeling. *IEEE Software* 11, 42-49.
- [Ly87] Lyytinen, K. (1987) Different Perspectives on Information Systems: Problems and Solutions. *ACM Computer Surveys* 19, 5-46
- [Me00] Mertens, P. (2000): Integrierte Informationsverarbeitung – 1. Administrations und Dispositionssysteme in der Industrie. vol. 12, Gabler, Wiesbaden.
- [MKK02] Milton, S., Kazmierczak, E., Keen, C. (2002): On the Study of Data Modelling Languages using Chisholm’s Ontology. *Proceedings of the Information Modelling and Knowledge Bases XIII*, Australia
- [NS72] Newell A., Simon, H.A. (1972) Human Problem-Solving. New Jersey, Englewood Cliffs.
- [OH01] Opdahl, A.L., Henderson-Sellers, B. (2001) Grounding the OML metamodels in ontology. *J. of Systems and Software* 57, 119-143.
- [OH02] Opdahl, A.L., Henderson-Sellers, B. (2002) Ontological Evaluation of the UML Using the Bunge-Wand-Weber Model. *Software and Systems Modeling* 1, 43-67.
- [RA07] Rosemann, M., van der Aalst, W.M.P. (2007) A configurable reference modeling language. *Information Systems* 32, 1-23
- [RB96] Rossi, M., Brinkkemper, S (1996) Complexity Metrics for Systems Development Methods and Techniques. *Information Systems* 21, 209-227.
- [RJB04] Rumbaugh, J., Jacobson, I., Booch, G.: Unified Modeling Language Reference Manual. 2. Aufl. Pearson Hicher Education, 2004.
- [Re97] Remme, M. (1997) Konstruktion von Geschäftsprozessen – Ein modellgestützter Ansatz durch Montage generischer Prozessartikel. Gabler, Wiesbaden.
- [RRI06] Rosemann, M., Recker, J., Indulska, M., Green, P. (2006) A study of the Evolution of the Representational capabilities of Process Modelling Grammars. *Proceedings of the CAiSE*, 447-461.
- [Sche94] Scheer, A.-W. (1994) Business Process Engineering – Reference Models for Industrial Companies. vol. 2, Berlin, Gabler.
- [Schw99] Schwegmann, A. (1999) Objektorientierte Referenzmodellierung – Theoretische Grundlagen und praktische Anwendung. Gabler, Wiesbaden.
- [Sh03] Shanks, G., Nuredini, J., Tobin, D., Moody, D., Weber, R. (2003) Representing Things and Properties in Conceptual Modeling: An empirical Evaluation. *Proceedings of the European Conference on Information Systems*.
- [SR98] Siau, K., Rossi, M. (1998) Evaluation of Information Modeling Methods – A Review. *Proceedings of Hawaii International Conference on System Sciences*.

- [VG98] Vessey, I., Glass, R. (1998): Strong vs. Week Approaches to Systems Development. *Communications of the ACM* 41, 99-103.
- [Wa95] Wand, Y., Monarchi, D.E., Parsons, J., Woo, C.C. (1995) Theoretical foundations for conceptual modelling in information systems development. *Decision Support Systems* 15, 285-304.
- [We93] Wedekind, H. (1993) Kaufmännische Datenbanken. Mannheim.
- [We97] Weber, R. (1997) Ontological Foundations of Information Systems. Cooper&Lybrand, Melbourne.
- [WSW99] Wand, Y., Storey, V.C., Weber, R. (1999) An Ontological Analysis of the Relationship Construct in Conceptual Modeling. *ACM Transactions on Database Systems* 24, 495-528.
- [WW02] Wand, Y., Weber, R. (2002) Research Commentary: Information Systems and Conceptual Modeling – A Research Agenda. *J. of Information Systems Research* 13, 363-376.

Der Einfluss von Modellierungswerkzeugen auf Qualität und Quantität von Modellen – Eine empirische Untersuchung

Iris Trojahn¹, Jens Weller², Bärbel Fürstenau¹, Werner Esswein³

¹Technische Universität Dresden

Lehrstuhl Wirtschaftspädagogik

01062 Dresden

{iris.trojahn|baerbel.fuerstenau}@tu-dresden.de

²Technische Universität Dresden

Institut für Werkzeugmaschinen und Steuerungstechnik

01062 Dresden

jens.weller@tu-dresden.de

³Technische Universität Dresden

Lehrstuhl für Wirtschaftsinformatik insb. Systementwicklung

01062 Dresden

werner.esswein@tu-dresden.de

Abstract: Modellierungswerkzeuge haben sich in den letzten Jahren als Hilfsmittel für die Modellerstellung etabliert. Dabei wird unterstellt, dass deren Nutzung keinen Einfluss auf den Inhalt der Modelle hat. Anderslautende Hinweise aus der Literatur werden, vor allem aufgrund der Vorteile dieser Werkzeuge, ignoriert. Um für weitere wissenschaftliche Arbeiten Klarheit zu schaffen, untersucht der vorliegende Beitrag die Fragestellung, ob sich Modelle, welche mit Hilfe von Modellierungswerkzeugen unter Verwendung der Modellierungssprache Conceptual Graphs erstellt werden, von Modellen, die mit Papier und Bleistift entstehen, unterscheiden. Mit Hilfe einer empirischen Studie wird gezeigt, dass die computergestützte Modellierung keine Nachteile gegenüber der Modellierung mit Papier und Bleistift aufweist.

1 Einleitung

Grafische Modelle haben sich in den letzten Jahren als Hilfsmittel in Wissenschaft und Praxis etabliert [Fr03, We03, Fe09]. Für die Erzeugung der Modelle kommen dabei häufig Modellierungswerkzeuge zum Einsatz. Dadurch werden auf der einen Seite eine elektronische Syntaxprüfung sowie die verteilte Modellierung unterstützt [De08]. Auf der anderen Seite ermöglicht die elektronische Ablage von Modellen auch deren Transformation und Nutzung durch technische Systeme [Es09].

Bei der Verwendung von Modellierungswerkzeugen wird angenommen, dass deren Einsatz im Vergleich zur Verwendung von Papier und Bleistift keine Auswirkung auf den Inhalt der Modelle hat. In der Literatur finden sich jedoch durchaus Hinweise, dass zwischen beiden Explikationen Unterschiede bezüglich der (inhaltlichen) Qualität der Modelle zu verzeichnen sind [TRR02]. Insbesondere wenn Modelle nicht für die bloße Kommunikation, sondern auch für die Bewertung der internen Wissensstrukturen des Modellierers verwendet werden sollen [FT05], muss ein solcher Unterschied jedoch ausgeschlossen oder aber genau definierbar sein, um verwertbare Erkenntnisse zu erzielen. Der vorliegende Beitrag widmet sich dieser Problematik. Anhand einer empirischen Studie wird geklärt, ob sich die Ergebnisse computergestützter Modellierung von denen, die auf herkömmliche Weise im Papier und Bleistift-Verfahren gewonnen wurden, in Umfang und/oder Qualität unterscheiden. Innerhalb der Untersuchung kommt dabei die Modellierungssprache der Conceptual Graphs [So06] zum Einsatz.¹

Der Beitrag ist wie folgt gegliedert. Zunächst werden die Einflussfaktoren aus der Literatur aufgearbeitet und so die Studie vorbereitet. Danach wird der Aufbau der Studie vorgestellt. Anschließend werden die Ergebnisse der Untersuchung präsentiert und im Kontext der obigen Fragestellung diskutiert. Der Artikel schließt mit einer Zusammenfassung und einem Ausblick auf zukünftige Forschung.

2 Einflussfaktoren auf die Explikation bei der Modellbildung

Modelle werden in der Wirtschaftsinformatik heute zumeist als Repräsentationen eines Originals verstanden. Sie sind das Ergebnis eines Konstruktionsprozesses [Th05]. Während dieses Prozesses expliziert der Modellierer seine interne Vorstellung vom Original, auch als mentales Modell bezeichnet, mit Hilfe einer Modellierungssprache [Sc98]. Die Explikation erfolgt dabei unter Zuhilfenahme verschiedener Werkzeuge, die sich u. a. hinsichtlich des verwendeten Mediums unterscheiden können.

Die Qualität des erzeugten Modells unterliegt somit verschiedenen Einflussfaktoren. Dabei sind Faktoren, die aus individuellen Merkmalen des Modellierers resultieren, von den Einflussfaktoren, die in den verwendeten Werkzeugen und Medien zur Explikation begründet sind, zu unterscheiden.

¹ Die Entscheidung für die Modellierungssprache Conceptual Graphs fiel im Zusammenhang mit dem wirtschaftspädagogischen Hintergrund der durchgeführten Studie. So wird diese Modellierungssprache seit den 1980er Jahren zur Darstellung und Diagnose komplexer Begriffs- bzw. Wissensstrukturen genutzt [NG84]. Im Zusammenhang mit dieser als ‚Concept Mapping‘ bekannten Explikationsmethode kommen in jüngster Zeit vielfach computergestützte Werkzeuge zum Einsatz, deren Wirkung auf die explizierten Strukturen bislang kaum hinterfragt wurde.

2.1 Personbezogene Einflussfaktoren des Modellierers

Die Literatur nennt als personbezogene Einflussfaktoren insbesondere das Vorwissen über das explizierte Original, das mediale Vorwissen [Cr06, JG93], das räumliche Vorstellungsvermögen des Modellierers [Ha02] sowie das thematische Interesse am Original [St04], wobei den ersten beiden Einflussfaktoren eine besondere Bedeutung beigemessen wird [Ma07]. Mediales Vorwissen, häufig auch als kognitive Komponente der Computer Literacy bezeichnet, umfasst „die Gesamtheit von prozeduralen und deklarativen Wissensbeständen, die für einen kompetenten Umgang mit dem Computer zielführend sind“ [RNG01]. Darüber hinaus ist davon auszugehen, dass auch die Sicherheit und Vertrautheit mit dem einzusetzenden Werkzeug sowie die Einstellung gegenüber dem Erhebungswerkzeug einen Einfluss auf die Modellerstellung ausüben [RNG01].

2.2 Werkzeugabhängige Einflussfaktoren

Zur Beschreibung des Ausmaßes, in dem computergestützte Werkzeuge dem Menschen bei der Erfüllung seiner Aufgaben hilfreich sind, hat sich der Begriff der Usability etabliert. Usability ist das „...Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und mit Zufriedenheit zu erreichen“ [De98]. Usability beschreibt demnach die adäquate Handhabung eines Produkts bzw. dessen Anpassung an die Bedürfnisse, Fähigkeiten und Fertigkeiten sowie Wünsche des Nutzers [Ni08], wobei die Messung der Usability anhand der Merkmale Effektivität, Effizienz und Zufriedenheit erfolgen kann.

3 Studie zur Bestimmung des Einflusses des Mediums auf die Explikation mentaler Modelle

3.1 Forschungsfrage

Ausgehend von der im Kapitel 1 beschriebenen Motivation, soll geklärt werden, ob sich die Ergebnisse computergestützter Modellierung von denen, die auf herkömmliche Weise im Papier und Bleistift-Verfahren gewonnen wurden, in Quantität und/oder Qualität unterscheiden. Unter Berücksichtigung des skizzierten Modellierungsprozesses kann die Forschungsfrage für die Studie daher folgendermaßen formuliert werden:

Welche Auswirkungen hat das bei der Explikation von mentalen Modellen verwendete Werkzeug auf die Qualität und Quantität der erstellten Modelle?

Als Explikationswerkzeuge kommen dabei ein nicht-IT-basiertes Werkzeug (in Form einer Explikation mit Papier und Bleistift) sowie ein Software-Modellierungswerkzeug zum Einsatz. Alle Modelle werden mit Hilfe der Modellierungssprache Conceptual Graphs [So06] erstellt.

3.2 Aufbau und Durchführung der Untersuchung

An der Studie beteiligten sich 40 Studenten der Wirtschaftswissenschaften (vgl. Tabelle 1), welche in einem Laborexperiment unter Kontrolle potentieller Störvariablen jeweils ein Modell zum Thema Inflation erstellen sollten. Die Modelle wurden anschließend in Bezug auf ihre Qualität und Quantität bewertet. Die Probanden wurden zufällig auf zwei Versuchsgruppen aufgeteilt, wobei die erste Gruppe als Werkzeug Papier und Bleistift, die zweite Gruppe das Modellierungswerkzeug Cubetto Toolset [Cu10] verwendete.

Merkmale der Probanden	Gruppe Cubetto	Gruppe Papier&Bleistift	gesamt
Probanden insgesamt:	21	19	40
Geschlecht: männlich/weiblich:	5/16	5/14	10/30
Durchschnittsalter:	21,3	22,4	21,8

Tabelle 1: Übersicht über die Probanden und die Zusammensetzung der Gruppen

Um die im Abschnitt 2.1 skizzierten personenbezogenen Einflüsse auf die Explikation zu kontrollieren, war es notwendig, das themenbezogene Vorwissen sowie die Erfahrungen und Einstellungen der Probanden im Zusammenhang mit Computern zu ermitteln. Hierfür absolvierten alle Probanden im Vorfeld der Studie zunächst einen themenbezogenen Wissenstest und beantworteten mehrere Fragebögen zur Computerbildung, welche insbesondere die Vertrautheit/Sicherheit im Umgang mit Computern (VECA und SUCA) sowie die Einstellung zum Computer im Allgemeinen (FIDEC) abfragten und dem Inventar zur Computerbildung (INCOBI) [RNG01] entnommen wurden. Der Wissenstest umfasste fünf Fragen zum Thema Inflation für deren Beantwortung im Essay-Format den Probanden jeweils 5 Minuten Zeit gewährt wurde [Gö07, Kö07, Ka08].

Um ungewollte Einflüsse durch unterschiedliche Kenntnis der eingesetzten Werkzeuge zu vermeiden, erhielten alle Probanden außerdem ein Training zur Anwendung der Modellierungssprache, das für beide Gruppen an das jeweils verwendete Explikationswerkzeug (Cubetto bzw. Papier und Bleistift) angepasst wurde.

Nach diesen vorbereitenden Aktivitäten erfolgte die eigentliche Modellierung. Da aufgrund der Auswahl der Probanden eine gewisse Affinität zu wirtschaftsbezogenen Themen unterstellt werden konnte, erhielten die Teilnehmer die Aufgabe, ihr Wissen bzw. ihre mentalen Modelle zum Thema Inflation in Form eines entsprechenden Modells zu explizieren. Dabei sollten sie insbesondere auf die verschiedenen Ursachen einer Inflation sowie auf die zwischen diesen Ursachen bestehenden Zusammenhänge eingehen.² Die Bearbeitungszeit war für beide Gruppen identisch und betrug 30 Minuten.

² Die exakte Aufgabenstellung lautete dabei wie folgt: „Was verstehen Sie unter dem Begriff der ‚Inflation‘? Erklären Sie die Entstehung einer Inflation! Beziehen Sie dabei verschiedene mögliche Ursachen in Ihre Überlegungen ein! Zeigen Sie auch auf, welche Beziehungen zwischen den Ursachen bestehen!“

Den Probanden der Gruppe Papier&Bleistift wurden für die Explikation ihrer mentalen Modelle ein Blatt der Größe DIN A1, kleine selbstklebende Zettel sowie ein Bleistift zur Verfügung gestellt. Sie wurden angewiesen, sich zuerst die zur Explikation ihres mentalen Modells benötigten Modellelement-Knoten (bei Conceptual Graphs als Konzepte bezeichnet) zu überlegen und auf die Klebezettel zu schreiben. Die beschrifteten Zettel waren dann auf dem DIN A1-Blatt sinnvoll zu platzieren und durch passende Relationen miteinander zu verbinden. Durch die Verwendung selbstklebender Zettel war es jederzeit möglich, die Anordnung der Konzepte zu verändern. Bereits festgelegte Relationen durften gestrichen oder ausgeradiert oder von anderen Relationen gekreuzt werden. Außerdem konnte die für die Modellerstellung zur Verfügung stehende Fläche jederzeit durch das Anfügen zusätzlicher Papierbögen erweitert werden.

Um eine größtmögliche Vergleichbarkeit des Modellierungswerkszeugs mit dem Papier und Bleistift-Verfahren zu erhalten, wurden bei der Gruppe Cubetto die Funktionen des Werkzeugs entsprechend eingeschränkt. So wurde das Cubetto Toolset derart konfiguriert, dass mit nur wenigen Mouse-Klicks Konzepte erzeugt und beschriftet werden konnten. Ebenso einfach war es, zwei Konzepte durch eine gerichtete und benannte Relation miteinander zu verbinden oder die Position der Konzepte und Relationen durch einfaches Ziehen zu verändern. Die für die Modellierung zur Verfügung stehende Fläche war im Werkzeug prinzipiell unendlich groß, wobei eine gewisse Einschränkung durch den sichtbaren Bildschirmrand gegeben war. Diese ließ sich jedoch durch den Gebrauch der Zoom-Funktion überwinden. Über die genannten Funktionen hinaus wurde das Modellierungswerkzeug so eingeschränkt, dass, wie auch bei der Modellierung mit Papier und Bleistift, keine weiteren Darstellungsmöglichkeiten, wie farbliche Hervorhebungen o. Ä., gegeben waren.

Im Anschluss an die Erhebung wurden die Probanden gebeten, ihre Erfahrungen mit den verwendeten Werkzeugen zu bewerten. Auf diese Weise konnten Aussagen der Probanden zur Usability der eingesetzten Werkzeuge (Papier und Bleistift bzw. Cubetto) gewonnen werden. Hierfür wurde für beide Gruppen jeweils ein eigener Fragebogen entwickelt, der die zentralen Merkmale der Usability (Effizienz, Effektivität und Zufriedenheit) zugrunde legt und unter Berücksichtigung der DIN EN ISO 9241-11 [De98] operationalisiert wurde.

Die Items der Fragebögen waren jeweils auf einer 5-stufigen Skala (von „stimme zu“ bis „stimme nicht zu“) einzuschätzen. Darüber hinaus beinhaltete der Fragebogen offene Fragen, die Probleme der Probanden bei der Modellerstellung sowie deren Wünsche für zusätzliche Funktionen und Darstellungsmöglichkeiten des Werkzeugs erfassten. Abbildung 1 fasst den Ablauf der Studie noch einmal zusammen.

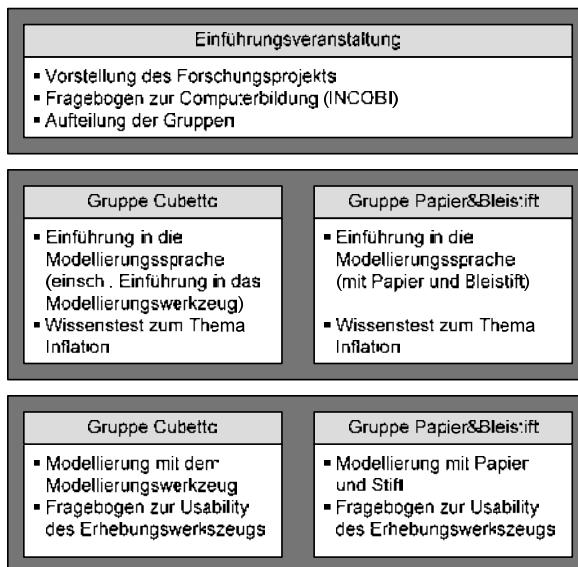


Abbildung 1: Ablauf der Untersuchung

3.3 Auswertungsmethoden

Die Auswertung des Wissenstests sowie der eingesetzten Fragebögen erfolgte mit Hilfe von Verfahren der deskriptiven und induktiven Statistik. Die qualitative Auswertung der Modelle erfolgte auf Grundlage der Modalnetze der beiden Versuchsgruppen. Modale Netzwerke sind künstlich generierte Modelle, welche die am häufigsten in den untersuchten Modellen genannten Propositionen (Konzept-Relation-Konzept Tripel) beinhalten und damit zusammenfassende Aussagen über die gesamte Gruppe ermöglichen [FT05]. Die quantitative Auswertung der Modelle erfolgte hingegen auf Grundlage der Individualmodelle. Insbesondere wurden die quantitativen Kennzahlen Umfang, Dichte und Zerklüftetheit bestimmt und Zusammenhänge zwischen diesen und den möglichen Einflussfaktoren (vgl. Kapitel 2) aufgezeigt. Um die Vergleichbarkeit der erzeugten Modelle in inhaltlicher und struktureller Hinsicht zu erhöhen, wurden zuvor die Modelle der Probanden einer kategorialen und einer strukturellen Inhaltsanalyse unterzogen, wodurch u. a. Synonyme erkannt und beseitigt werden konnten [GR95, FT05].³

³ Bei der kategorialen Inhaltsanalyse bilden nicht die gesamten Propositionen, sondern die Konzepte und Relationen getrennt voneinander die jeweilige Analyseeinheit. Im Zuge der Zusammenfassung einzelner Konzepte bzw. Relationen zu einheitlichen Kategorien wurde insbesondere darauf geachtet, dass die Aussagekraft der individuellen Modelle, zugunsten einer erhöhten Vergleichbarkeit der Modelle untereinander, nicht zu stark geschmälert wurde. Bis zur endgültigen Festlegung des Auflösungsgrades der Kategorien wurden aus diesem Grund mehrere Iterationen durchgeführt [FT05].

4 Ergebnisse und Diskussion

4.1 Ergebnisse der Voruntersuchung

Die Auswertung des Wissenstests erfolgte anhand der Musterlösungen der Arbeiten von Göldner [Gö07] und Kahle [Ka08]. Insgesamt konnten 18 Punkte erreicht werden. Davon erreichten die Probanden der Gruppe Cubetto im Durchschnitt 8,1 Punkte ($SD = 2,36$), während die Probanden der Gruppe Papier&Bleistift durchschnittlich 7,3 Punkte erzielten ($SD = 2,57$). Der Vergleich der Mittelwerte beider Gruppen zeigt keinen signifikanten Unterschied ($t[38] = 1,003; p = 0,322$). Es ist daher davon auszugehen, dass beide Versuchgruppen in etwa über dieselbe, wenn auch gering ausgeprägte, Wissensbasis zum Thema Inflation verfügen.

Die Ermittlung der Computerbildung erfolgte mithilfe ausgewählter Fragebögen des INCOBI (vgl. Kapitel 3 sowie [RNG01]). Wie Tabelle 2 zu entnehmen ist, schätzen die Probanden beider Gruppen ihre Sicherheit und Vertrautheit im Umgang mit Computern und Computeranwendungen im Durchschnitt eher neutral bzw. durchschnittlich mit positiver Tendenz ein. Darüber hinaus weisen beiden Gruppen eine tendenziell positive Einstellung zum Computer als Lern- und Arbeitsmittel auf.

	Gruppe	N	Mittelwert	Standardabweichung
Computer Literacy gesamt	Cubetto	21	2,22	0,393
	Papier&Bleistift	19	2,34	0,484
Computerbezogene Einstellungen gesamt	Cubetto	21	2,62	0,427
	Papier&Bleistift	19	2,88	0,452
Computerbildung gesamt	Cubetto	21	2,35	0,340
	Papier&Bleistift	19	2,52	0,427

Tabelle 2: Aggregierte Ergebnisse der Fragebögen zur Computerbildung

Als Gesamtaussage lässt sich festhalten, dass die Computerbildung als Summe aus Computer Literacy und computerbezogenen Einstellungen bei beiden Gruppen eher positiv bewertet werden kann. Der Mittelwertvergleich ergibt keinen signifikanten Unterschied ($t[38] = -1,423; p = 0,163$), sodass von einer homogenen Computerbildung beider Gruppen ausgegangen werden kann.

4.2 Ergebnisse der Hauptuntersuchung

4.2.1 Auswertung der Fragebögen zur Usability

Die im Anschluss an die Modellerstellung eingesetzten Fragebögen dienten der Einschätzung der verschiedenen Merkmale der Usability des jeweils eingesetzten Modellierungswerkzeugs durch die Probanden. Tabelle 3 zeigt die jeweiligen Mittelwerte und Standardabweichungen für die verschiedenen Gruppen.⁴ Für das Merkmal Effizienz ergab der Mittelwertvergleich keinen signifikanten Unterschied ($t[38] = 0,555$; $p = 0,582$). Demzufolge kann davon ausgegangen werden, dass den Probanden der Umgang mit dem jeweiligen Medium insgesamt keine großen Schwierigkeiten bereitete. Insbesondere zeigt sich, dass bei den Teilnehmern der Gruppe Cubetto kaum Probleme bei der Aneignung und Bedienung der Software auftraten.

	Gruppe	N	Mittelwert	Standardabweichung
Effizienz	Cubetto	21	,98	,521
	Papier&Bleistift	19	,89	,538
Effektivität	Cubetto	21	-,14	,761
	Papier&Bleistift	19	-,16	,817
Zufriedenheit	Cubetto	21	1,29	,845
	Papier&Bleistift	19	,53	1,504
Usability gesamt	Cubetto	21	,71	,470
	Papier&Bleistift	19	,42	,745

Tabelle 3: Bewertung der Merkmale der Usability

Interessant ist die Betrachtung einzelner Items der Effizienz im Vergleich. So empfanden die Probanden der Gruppe Papier&Bleistift die zeitliche Restriktion bei der Modellerstellung als signifikant weniger problematisch als die Probanden der Gruppe Cubetto (vgl. Tabelle 4). Die Frage, ob die Probanden die Erstellung des Modells mit dem jeweiligen Medium insgesamt als schwierig empfanden, wurde von beiden Gruppen eher verneint. Den Korrekturaufwand schätzte die Gruppe Cubetto als signifikant geringer ein als die Gruppe Papier&Bleistift. Die Mittelwerte der Fragen bezüglich der grafischen Gestaltung der Modelle zeigen, dass sich die Gruppe Papier&Bleistift weniger Gedanken um die optische Gestaltung der Modelle machte. So zeigten sich bei der Betrachtung der Modelle dieser Gruppe auch signifikant mehr Überschneidungen der einzelnen Propositionen als dies bei der Gruppe Cubetto der Fall war ($z = -2,894$; $p = 0,004$).

⁴ Die Fragebögen zur Einschätzung der Usability der Explikationswerkzeuge umfassten insgesamt 16 Fragen (Gruppe Cubetto) bzw. 11 Fragen (Gruppe Papier&Bleistift). Die Antworten der Probanden wurden entsprechend der 5-stufigen Skalen mit „-2 = stimme zu“ bis „+2 = stimme nicht zu“ bewertet.

Ausgewählte Items zur Beurteilung der Effizienz	Mittelwerte	
	Gruppe Cubetto	Gruppe Papier& Bleistift
Für die Erstellung des Modells hätte ich gerne mehr Zeit gehabt.	-,19**	,89**
Die Erstellung des Modells mit Papier und Bleistift/am Computer ist mir insgesamt schwer gefallen.	1,57	1,32
Ich habe mehr Zeit für die grafische Gestaltung meines Modells verwendet als mir über den Inhalt Gedanken zu machen.	,71	1,21
Ich habe sehr viel Zeit damit verbracht, das Modell optisch schön zu gestalten.	,67	1,21
Die Korrektur von Fehlern empfand ich als mühsam (z. B. bei falsch verbundenen Modellelementen).	1,05*	0,21*

* Die Mittelwerte unterscheiden sich voneinander signifikant ($\alpha \leq 0,05$)

** Die Mittelwerte unterscheiden sich voneinander signifikant ($\alpha \leq 0,01$)

(Bewertung der Urteile: stimme zu = -2 bis stimme nicht zu = +2)

Tabelle 4: Ausgewählte Items zur Beurteilung der Effizienz

Wie Tabelle 3 zeigt, wurde der Grad der Zielerreichung, welcher durch das Merkmal Effektivität ausgedrückt wird, von beiden Gruppen als eher gering eingeschätzt, wobei zwischen beiden Gruppen kein signifikanter Unterschied festgestellt werden konnte ($t[38] = 0,60$; $p = 0,952$).

Interessant ist an dieser Stelle jedoch die nähere Betrachtung des Zusammenhangs zwischen der empfundenen Effektivität und der computerbezogenen Einstellung der Probanden. Eine Korrelationsanalyse innerhalb der Gruppe Cubetto zeigt, dass sich eine positive Einstellung gegenüber dem Medium Computer positiv auf die Einschätzung der Effektivität auswirkt ($r = 0,485$; $p = 0,026$). Bei der Gruppe Papier&Bleistift hingegen besteht ein negativer Zusammenhang zwischen der Einstellung zum Medium Computer und dem Grad der Zielerreichung ($r = -0,469$; $p = 0,043$). Demzufolge schätzen Probanden mit einer positiven Computereinstellung die Effektivität des Werkzeugs Papier und Bleistift als geringer ein.

Tabelle 3 zeigt weiterhin, dass der Mittelwert des Merkmals Zufriedenheit als Zeichen für die Akzeptanz des Mediums der Gruppe Cubetto deutlich über dem der Gruppe Papier&Bleistift liegt ($t[38] = 1,993$, $p = 0,053$). Hinzu kommt, dass deutlich mehr Teilnehmer der Gruppe Papier&Bleistift sich eine Erstellung des Modells am Computer gewünscht hätten als Probanden der Gruppe Cubetto eine Darstellung des Modells mit Papier und Bleistift bevorzugt hätten ($t[27,71] = 1,941$; $p = 0,063$). Diese Feststellung wird durch die Überprüfung des Zusammenhangs zwischen der Computer Literacy und der Zufriedenheit mit dem Medium bei der Gruppe Papier&Bleistift unterstützt. Die Korrelationsanalyse ergibt eine signifikante negative Korrelation von $r = -0,655$ bei einer Signifikanz von 0,002. Demnach sinkt die Akzeptanz des Mediums Papier mit zunehmendem Computerwissen.

Insgesamt kann die Usability als Mittelwert der Merkmale Effizienz, Effektivität und Zufriedenheit in beiden Gruppen als tendenziell positiv interpretiert werden (vgl. Tabelle 3). Die Mittelwerte zeigen, dass die Gruppe Papier&Bleistift die Usability des Werkzeugs Papier und Bleistift etwas geringer einschätzt als die Gruppe Cubetto die Usability des Modellierungswerkzeugs Cubetto. Allerdings ist dieser Unterschied nicht signifikant ($t[38] = 1,482$; $p = 0,146$). Korrelationsanalysen zeigen zusätzlich negative Zusammenhänge zwischen der Einschätzung der Effizienz und der Computer Literacy ($r = -0,408$; $p = 0,083$) sowie zwischen der Zufriedenheit mit dem jeweiligen Werkzeug und der computerbezogenen Einstellung ($r = -0,424$; $p = 0,070$). Auch wenn diese Ergebnisse nicht signifikant sind, zeigt sich zumindest für diese Untersuchungsgruppe, dass sich eine positive Einstellung zum Medium Computer sowie ein höheres Computerwissen negativ auf die eingeschätzte Benutzerfreundlichkeit des Mediums Papier auswirken.

Bei der Auswertung der offenen Fragen des Fragebogens konnte festgestellt werden, dass die Probanden vor allem Schwierigkeiten mit der Thematik Inflation hatten. Dies verdeutlichen Aussagen auf die Frage „Welche Probleme hatten Sie während der Erstellung ihres Modells?“. Die Probanden lieferten Aussagen wie „Unsicherheiten zum Thema Inflation haben mich in der Gestaltung eingeschränkt...“, „Inhaltliche Lücken“, „Unzureichendes Wissen“, „die Thematik Inflation an sich zu erklären...“. Diese Feststellung kann zusätzlich zur Erklärung des relativ niedrig eingeschätzten Grades der Zielerreichung herangezogen werden. Ebenso spiegeln die Aussagen der Probanden die Ergebnisse des Wissenstests wieder. Auf die Frage „Welche Funktionen/Darstellungsmöglichkeiten hätten Sie sich zusätzlich gewünscht?“ wurde in beiden Gruppen die Möglichkeit zur besseren optischen Gestaltung genannt. So äußerten Teilnehmer der Gruppe Papier&Bleistift z. B. den Wunsch nach verschiedenfarbigen Klebezetteln oder die Möglichkeit zur Verwendung verschiedener Stiftfarben. Ähnliche Vorschläge wurden in der Gruppe Cubetto genannt. Auch hier wünschten sich Probanden mehr farbliche Gestaltungsmöglichkeiten.

4.2.2 Qualitative Auswertung der Modelle

Die qualitative Auswertung der Modelle erfolgt auf Grundlage der modalen Netzwerke. Diese sind das Ergebnis der strukturellen Inhaltsanalyse und enthalten die von den Probanden am häufigsten genannten Propositionen (Konzept-Relation-Konzept Tripel). Sie stellen somit eine durchschnittliche Modellstruktur über alle Probanden der jeweiligen Gruppe dar [FT05].

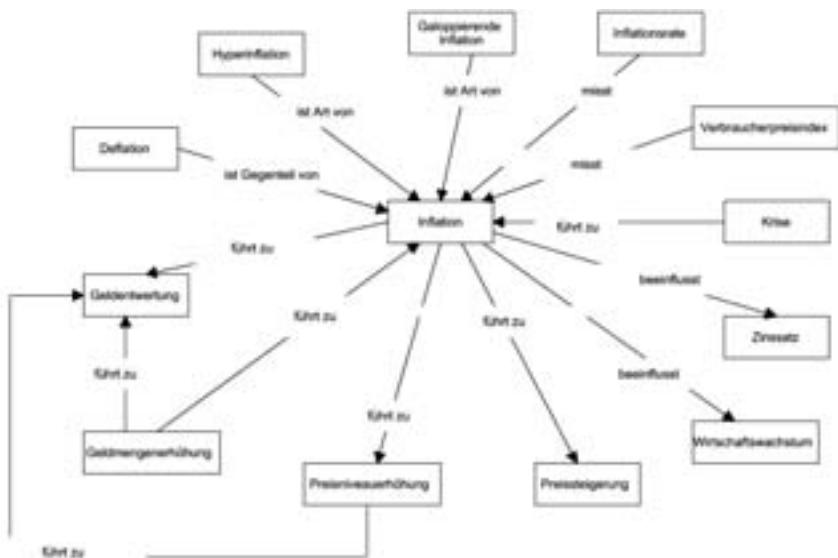


Abbildung 2: Modalnetz der Gruppe Cubetto

Das modale Netzwerk der Gruppe Cubetto (vgl. Abbildung 2) enthält 14 Propositionen und repräsentiert damit 16,05 % aller innerhalb der Gruppe Cubetto genannten Propositionen. Strukturell zeichnet sich das Modalnetz dieser Gruppe dadurch aus, dass es lediglich aus einer Zusammenhangskomponente besteht, d. h. es besteht innerhalb der Gruppe Cubetto ein gewisses Maß an Einigkeit über die Relevanz der angesprochenen Zusammenhänge. Inhaltlich werden verstärkt Bereiche angesprochen, die einen direkten Zusammenhang mit dem Thema Inflation aufweisen, wofür auch die sternförmige Anordnung der Propositionen mit dem Konzept ‚Inflation‘ als Mittelpunkt hindeutet. Insbesondere werden zentrale Begriffe, wie Preisniveauerhöhung, Geldentwertung, Preissteigerung oder Geldmengenerhöhung, genannt [BI06, Ma04]. Auffällig ist weiterhin die Einbeziehung der unterschiedlichen Inflationsarten.

Das Modalnetz der Gruppe Papier&Bleistift besteht aus 22 Propositionen (vgl. Abbildung 3). Die Abbildungsleistung beträgt 15,9 % und ist damit annähernd so groß wie die der Gruppe Cubetto. Auch dieses Modalnetz weist eine sternförmige Struktur auf, jedoch fällt auf, dass zusätzlich zu den oben genannten Themenbereichen auch Konzepte genannt werden, die nur im weiteren Sinne mit der Thematik Inflation zusammenhängen [BI06, Ma04]. Insbesondere werden die Rolle der EZB sowie die der Geschäftsbanken thematisiert. Außerdem werden verstärkt die Folgen einer Inflation herausgestellt, was nicht explizit Teil der Aufgabenstellung war. Dazu passt das Vorhandensein von drei Zusammenhangskomponenten. Offenbar ist der Konsens zwischen den Probanden der Gruppe Papier&Bleistift über die elementaren Begriffe im Zusammenhang mit dem Thema Inflation nicht so groß wie bei den Probanden der Gruppe Cubetto.

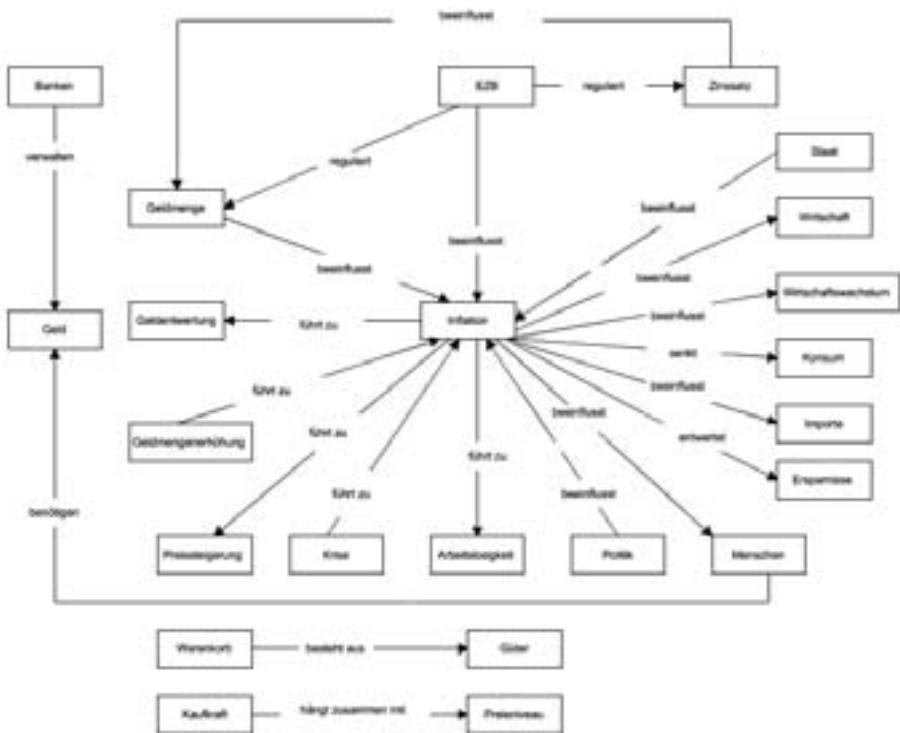


Abbildung 3: Modalnetz der Gruppe Papier&Bleistift

4.2.3 Quantitative Auswertung der Modelle

Die quantitative Auswertung der Modelle erfolgt anhand verschiedener Kennzahlen. Der Umfang eines Modells ergibt sich direkt aus der Anzahl der Propositionen. Die Modelle der Gruppe Cubetto umfassen durchschnittlich 20,8 Propositionen ($SD = 7,79$). Im Vergleich dazu weisen die Modelle der Gruppe Papier&Bleistift im Mittel einen Umfang von 32,2 Propositionen ($SD = 11,32$) auf. Damit unterscheiden sich die Mittelwerte der beiden Gruppen signifikant voneinander ($t[38] = -3,757; p = 0,001$).

Der Grad der Vernetzung wird über die Dichte operationalisiert. Die Dichte beschreibt das Verhältnis der Anzahl der im Modell genannten Propositionen zur Anzahl der maximal möglichen Verknüpfungen [WFG85]. Bei den Modellen der Gruppe Cubetto liegt die Dichte im Durchschnitt bei einem Wert von 0,09 ($SD = 0,03$), während sie in der Gruppe Papier&Bleistift durchschnittlich 0,11 ($SD = 0,04$) beträgt. Der Mittelwertvergleich zeigt, dass die Modelle der Gruppe Papier&Bleistift im Durchschnitt einen signifikant höheren Grad der Vernetzung aufweisen ($t[38] = -2,415; p = 0,021$).

Die Zerklüftetheit einer Modellstruktur bestimmt sich aus der Anzahl der einzelnen Zusammenhangskomponenten eines Modells [We94]. Die Modelle der Gruppe Papier&Bleistift bestanden jeweils nur aus einer Zusammenhangskomponente, d. h. es existierten keine isolierten Propositionen oder Teilmodelle. Bei der Gruppe Cubetto hingegen wiesen die Modelle von fünf Probanden jeweils zwei und das Modell eines Probanden sogar drei Zusammenhangskomponenten auf. Der Unterschied ist statistisch signifikant ($z = -2,491$; $p = 0,013$).

Insgesamt lässt sich festhalten, dass die Modelle der Gruppe Papier&Bleistift, bei nahezu gleicher Anzahl von Konzepten, einen signifikant größeren Umfang aufweisen und die Modellelemente untereinander signifikant stärker vernetzt sind. Ebenso sind die Modelle der Gruppe Papier&Bleistift signifikant weniger zerklüftet als die der Gruppe Cubetto. Ob diese Ergebnisse tatsächlich auf eine umfangreichere und vernetztere Wissensbasis im Sinne stärker elaborierter mentaler Modelle innerhalb der Gruppe Papier&Bleistift hindeuten [We94] oder sich auf andere Faktoren, wie Computerbildung oder Usability der eingesetzten Werkzeuge, zurückführen lassen, zeigen die folgenden Korrelationsanalysen.

	Wissen		Computer Literacy		Computer Einstellung		Computerbildung gesamt	
Gruppe	Cub N=21	P&B n=19	Cub N=21	P&B n=19	Cub N=21	P&B n=19	Cub N=21	P&B n=19
Umfang	,543*	,208	-,297	,161	,219	-,266	-,137	,026
Dichte	,039	-,172	-,067	,009	-,127	,323	-,105	,121
Zerklüftetheit	,087	†	-,026	†	-,127	†	,016	†

* Die Korrelation ist auf dem Niveau von 0,05 (2-seitig) signifikant.

† Für das Merkmal Zerklüftetheit konnten im Zusammenhang mit der Gruppe Papier&Bleistift keine Korrelationen bestimmt werden, da die Anzahl der Zusammenhangskomponenten bei allen Probanden 1 betrug.

Tabelle 5: Korrelationen zwischen personenbezogenen Einflussfaktoren auf die Modellbildung und quantitativen Modellmerkmalen

Wie Tabelle 5 sowie Tabelle 6 zeigen, konnten insbesondere für die Gruppe Cubetto einige signifikante Zusammenhänge aufgedeckt werden. So hat der Faktor Wissen einen signifikanten Einfluss auf den Umfang der Modelle. Probanden mit einem höheren themenbezogenen Vorwissen erstellten somit Modelle mit einer höheren Anzahl an Propositionen. Bei der Gruppe Papier&Bleistift konnte ein solcher Zusammenhang jedoch nicht festgestellt werden. Interessant erscheint weiterhin, dass im Bereich der Computerbildung offenbar weder die Computer Literacy noch die computerbezogenen Einstellungen einen Einfluss auf die Modellerstellung hatten. Es kann somit davon ausgegangen werden, dass für den Gebrauch des Modellierungswerkzeugs Cubetto zum Zwecke der Modellexplikation ebenso wie bei der Verwendung von Papier und Bleistift keine besonderen computerbezogenen Vorkenntnisse oder Einstellungen erforderlich sind.

Gruppe	Effizienz		Effektivität		Zufriedenheit		Usability gesamt	
	Cub N=21	P&B n=19	Cub N=21	P&B n=19	Cub N=21	P&B n=19	Cub N=21	P&B n=19
Umfang	-,242	-,231	,336	,310	-,103	,071	,030	,106
Dichte	,436*	-,016	,143	-,537*	,136	-,336	,320	-,427
Zerklüfttheit	-,226	†	-,405	†	-,233	†	-,512*	†

* Die Korrelation ist auf dem Niveau von 0,05 (2-seitig) signifikant.

† Für das Merkmal Zerklüfttheit konnten im Zusammenhang mit der Gruppe Papier&Bleistift keine Korrelationen bestimmt werden, da die Anzahl der Zusammenhangskomponenten bei allen Probanden 1 betrug.

Tabelle 6: Korrelationen zwischen werkzeugbezogenen Einflussfaktoren auf die Modellbildung und quantitativen Modellmerkmalen

Der Faktor Usability scheint hingegen durchaus Einfluss auf die Beschaffenheit der Modelle zu haben. So ergibt sich für die Gruppe Cubetto zwischen dem Gesamtkonstrukt und dem Maß der Zerklüfttheit eine negative Korrelation von $r = -0,512$. Dies bedeutet, dass die Anzahl der Zusammenhangskomponenten steigt, je geringer die Usability des Modellierungswerkzeugs eingeschätzt wird. Außerdem kann ein Zusammenhang zwischen den Merkmalen Effizienz bzw. Effektivität und dem Grad der Vernetzung (Dichte) festgestellt werden. Bei der Gruppe Cubetto wirkt sich eine positive Einschätzung der Effizienz positiv auf den Grad der Vernetzung aus. Offenbar führt also eine positive Einschätzung des Aufwand-Nutzen-Verhältnisses zu einer stärkeren Vernetzung der Konzepte untereinander. Hingegen ergibt sich bei der Gruppe Papier&Bleistift eine negative Korrelation zwischen der Einschätzung der Effektivität und der Dichte. Dies kann so interpretiert werden, dass die Probanden der Gruppe Papier&Bleistift ihre Konzepte zwar vielfältig miteinander vernetzten, dies aber nicht als zielführend erachteten.

Über die genannten Zusammenhänge hinaus, deuten auch die Ergebnisse des Fragebogens zur Modellbildung darauf hin, dass die Usability der eingesetzten Werkzeuge bei der Modellerstellung eine bedeutende Rolle spielt. Wie bereits gezeigt, wird die Modellerstellung mit Hilfe des Modellierungswerkzeugs Cubetto als zeitlich aufwändiger eingeschätzt als dies bei der Papier&Bleistift-Methode der Fall ist. Des Weiteren zeigt die Auswertung des Fragebogens, dass sich die Probanden der Gruppe Cubetto eher auf eine übersichtliche und optisch ansprechende Gestaltung der Modelle konzentrierten als auf inhaltliche Aspekte.

Über die im Fragebogen erfassten Faktoren hinaus kann weiterhin vermutet werden, dass insbesondere der begrenzte Bildschirmausschnitt Auswirkungen auf die Modellierung am Computer ausübt. Ein Überblick über die Gesamtstruktur kann nur über die Verkleinerungsfunktion des Programms erreicht werden, was jedoch eine Verringerung der Lesbarkeit nach sich zieht. Die Betrachtung der Modelle der Gruppe Cubetto lassen vermuten, dass diese Funktion nicht im vollen Umfang ausgeschöpft wurde, da sich die Größe der Modelle auf den sichtbaren Ausschnitt des Bildschirms beschränkt (zum „Guckkasten-Effekt“ vgl. [Kn95]).⁵

⁵ Die Einführung in den Gebrauch der Zoom-Funktion war expliziter Bestandteil des vorangegangenen Trainings zum Gebrauch des Explikationswerkzeugs Cubetto.

Die Gruppe Papier&Bleistift hingegen nutzte den zur Verfügung stehenden Platz des A1-Blattes überwiegend aus. Somit können der geringere Zeitaufwand, der offenbar geringere Anreiz zur ansprechenden Gestaltung und der bessere Überblick verbunden mit einem subjektiv empfundenen größerem Platzangebot über die genannten Faktoren hinaus dazu geführt haben, dass die Probanden der Gruppe Papier&Bleistift umfangreichere und stärker vernetzte Modelle erstellten, die zugleich weniger Zusammenhangskomponenten aufwiesen.

5 Zusammenfassung und Ausblick

Der Vergleich des computergestützten Erhebungswerkzeugs Cubetto mit dem traditionellen Papier und Bleistift-Verfahren führte insgesamt zu uneinheitlichen Ergebnissen. Zwar scheint auf den ersten Blick die herkömmliche Erhebung mit Papier und Bleistift der computergestützten Erhebung überlegen zu sein, da sie zu umfangreicheren, vernetzteren und strukturell kohärenteren Modellen führt. Jedoch ist die Interpretation dieser quantitativen Kennzahlen im Sinne „je mehr, desto besser“, wie Weber sie vorschlägt [We94] nicht unumstritten [Al02]. Ebenso wichtig wie die quantitative Beschreibung der Ergebnisse ist daher die qualitative Beurteilung der Modelle. Diese zeigt, dass die Probanden der Gruppe Cubetto sich stärker an der Aufgabenstellung orientierten und sich auf die Darstellung der wesentlichen Zusammenhänge konzentrierten, worin ein großer Vorteil gegenüber dem traditionellen Explikationsverfahren gesehen werden kann. Um zu einem differenzierteren Urteil über die Vorteilhaftigkeit eines der beiden Werkzeuge zu gelangen, bietet es sich für zukünftige Untersuchungen an, die Modelle mit einem externen Referenzmodell zu vergleichen und so die Angemessenheit der explizierten mentalen Modelle zu beurteilen [We94, Fü01, TF08].

Insgesamt zeigt sich, dass die computergestützte Explikation mentaler Modelle keine Nachteile gegenüber der Explikation mit Papier und Bleistift aufweist. Zusätzlich wurde eine Präferenz aller Probanden für die computergestützte Erhebung geäußert, sodass eine motivationssteigernde Wirkung beim Einsatz computergestützter Erhebungswerkzeuge vermutet werden kann und in zukünftigen Studien auch experimentell überprüft werden sollte. Positiv hervorzuheben ist weiterhin, dass die Modellerstellung mit Hilfe des Modellierungswerkzeugs unabhängig von computerbezogenem Vorwissen oder Einstellungen gelingt.

Aufgrund des nachgewiesenen Einflusses der Usability ist diesem Faktor bei zukünftigen Untersuchungen verstärkte Aufmerksamkeit zu widmen. Zur verbesserten Wahrnehmung der Merkmale Effektivität und Zufriedenheit sollte zudem eine Domäne gewählt werden, die den Probanden bereits gut vertraut ist. Außerdem ist der Freiraum in Bezug auf die Modellierungssprachen zu überdenken, sodass die Probanden auf weitere Gestaltungsmöglichkeiten zugreifen können. Dies könnte die motivierende Wirkung, und damit die Qualität der Modelle, weiter erhöhen [St04].

Literaturverzeichnis

- [Al02] Al-Diban, S.: Diagnose mentaler Modelle. Dissertation. Verlag Dr. Kovac, Hamburg, 2002.
- [Bl06] Blanchard, O.; Illing, G.: Makroökonomie. Pearson Studium, München, 2006.
- [Cr06] Creß, U.: Lernorientierungen, Lernstile, Lerntypen und kognitive Stile. In (Mandl, H.; Friedrich, H. F., Hrsg.): Handbuch Lernstrategien, 2006, S. 365-377.
- [Cu10] Semture GmbH: Cubetto Toolset, <http://www.semture.de/cubetto>, Download: 20.04.2010.
- [De98] Deutsches Institut für Normung: DIN EN ISO 9241-11: Ergonomische Anforderungen für Bürotätigkeiten mit Bildschirmgeräten, Teil 11: Anforderungen an die Gebrauchstauglichkeit, 1998.
- [De08] Delfmann, P. et al.: Ein konfiguratives Metamodellierungswerkzeug, In: Tagungsband Modellierung betrieblicher Informationssysteme, 2008. S. 109-127.
- [Es09] Esswein, W. et al.: Identifikation von Services aus Geschäftsprozessmodellen durch automatisierte Modellanalyse, Tagungsband Internationale Tagung Wirtschaftsinformatik, 2009, S. 513-522
- [Fe09] Fettke, P.: Ansätze der Informationsmodellierung und ihre betriebswirtschaftliche Bedeutung: Eine Untersuchung der Modellierungspraxis in Deutschland, In: Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung, 61, 2009.
- [Fr03] Frank, U.: Conceptual Modelling as the Core of the Information Systems Discipline: Perspectives and Epistemological Challenges, In (Haseman, D. W.; Nazareth, D.; Goodhue, D., Hrsg.): Proc of the Fifth America's Conf. on Information Systems, 1999, S. 695-697
- [FT05] Fürstenau, B.; Trojahner, I.: Prototypische Netzwerke als Ergebnis struktureller Inhaltsanalysen, In (Gonon, P.; Klauser, F.; Nickolaus, R.; Huisenga, R., Hrsg.): Kompetenz, Kognition und neue Konzepte der beruflichen Bildung, Verlag für Sozialwissenschaften, 2005, S. 191-202.
- [Fü01] Fürstenau, B.: Empirische Prüfung zweier Lehr-Lern-Arrangements für Industriekaufleute: Evaluation von Wissen mit Hilfe von Netzwerken. In: Zeitschrift für Berufs- und Wirtschaftspädagogik, 97 (2), S. 247-270.
- [Gö07] Göldner, S.: Studie zur Eignung von Netzwerken zum Inhaltslernen am Beispiel der Inflation. Diplomarbeit, Technische Universität Dresden, Lehrstuhl für Wirtschaftspädagogik, 2007.
- [Gr95] Groeben, N.; Rustemeyer, R.: Inhaltsanalyse. In (König, E.; Zedler, P., Hrsg.): Bilanz qualitativer Forschung. Band II: Methoden. Weinheim: Deutscher Studien-Verlag, 1995.
- [Ha02] Haller, H.: Mappingverfahren zur Wissensorganisation. Diplomarbeit. Download: <http://heikohaller.de/literatur/diplomarbeit/>, 2002.
- [JG93] Jonassen, D. H.; Grabowski, B. L.: Handbook of Individual Differences, Learning, and Instruction. Mahwah, 1993.
- [Ka08] Kahle, J.: Entwicklung eines Constructed-Response-Tests zur Messung der Lernwirksamkeit pädagogisch-didaktischer Interventionen. In (Fürstenau, B., Hrsg.): Dresdner Beiträge zur Wirtschaftspädagogik, (1), 2008.
- [Kn95] Knorr, D.: Elektronische Medien im wissenschaftlichen Alltag. Auswirkungen des Einsatzes von Computern auf die persönliche Literaturverwaltung und -nutzung. In (Jakobs, E.-M.; Knorr, D.; Molitor-Lübbert, S., Hrsg.): Wissenschaftliche Textproduktion. Mit und ohne Computer, Lang, Frankfurt/Main, S. 53-71.
- [Kö07] Köhler, R.: Ableitung und Operationalisierung von Kriterien zum Nachweis von Begriffsveränderungen in Netzwerken: Eine Untersuchung zum Thema Preisniveau. Diplomarbeit, Technische Universität Dresden, Lehrstuhl für Wirtschaftspädagogik, 2007.
- [Ma04] Mankiw, N. G.: Grundzüge der Volkswirtschaftslehre. Schäffer-Poeschel Verlag, Stuttgart, 2004.

- [Ma07] Manzel, S.: Kompetenzzuwachs im Politikunterricht. Ergebnisse einer Interventionsstudie zum Kernkonzept Europa, Waxmann. 2007.
- [Ni08] Niegemann, H.-M.; Domagk, S.; Hessel, S.; Hein, A.: Kompendium Multimediales Lernen, Springer, 2008.
- [NG84] Novak, J. D.; Gowin, D. B.: Learning how to learn. Cambridge University Press, 1984.
- [RNG01] Richter, T.; Naumann, J; Groeben, N.: Das Inventar zur Computerbildung (INCOBI): Ein Instrument zur Erfassung von Computer Literacy und computerbezogenen Einstellungen bei Studierenden der Geistes- und Sozialwissenschaften. In: Psychologie in Erziehung und Unterricht, (48), 2001, S. 1-13.
- [Sc98] Schütte, R.: Grundsätze ordnungsmäßiger Referenzmodellierung, Gabler Verlag, 1998.
- [So06] Sowa, J., F.: Conceptual Graphs, In: Handbook on Architectures of Information Systems, Springer Verlag, 2006, S. 295-319
- [St04] Stracke, I.: Einsatz computerbasierter Concept Maps zur Wissensdiagnose in der Chemie. Empirische Untersuchungen am Beispiel des Chemischen Gleichgewichts. Waxmann, Münster, 2004.
- [TF08] Trojahner, I.; Fürstenau, B.: Vorwissen von Studierenden im Bereich der Unternehmensgründung. In (Münk, D.; Breuer, K.; Deißinger, T., Hrsg.): Berufs- und Wirtschaftspädagogik: Probleme und Perspektiven aus nationaler und internationaler Sicht. Verlag Barbara Budrich, 2008.
- [Th05] Thomas, O.: Das Modellverständnis in der Wirtschaftsinformatik: Historie, Literaturanalyse und Begriffsexplikation, In (Scheer, A.-W., Hrsg.): Arbeitsbericht des Instituts für Wirtschaftsinformatik, 184, 2005.
- [TRR02] Troche, S.; Rammstedt, B.; Rammseyer, T.: Vergleich einer Papier-Bleistift- und einer computergestützten Version des Leistungsprüfsystems (LPS), In: Diagnostica, 48 (3), 2002, S. 115-120.
- [We94] Weber, S.: Vorwissen in der betriebswirtschaftlichen Ausbildung. Deutscher Universitäts-Verlag, Wiesbaden, 1994.
- [We03] Weber, R.: Still Desperately Seeking the IT Artifact, In: MIS Quarterly, 27 (2), 2003, S. iii-xi.
- [WFG85] West, L. H. T.; Fensham, P. J.; Garrard, J. E.: Describing the cognitive structure of learners following instruction in chemistry. In (West, L. H. T.; Pines, A. L., Hrsg.): Cognitive structure and conceptual change, 1995, S. 61-90.

Eine Methode für den praktischen Einsatz von Konfigurationsmanagement in verteilten Modellierungsprojekten

Jens Weller, Michaela Helbig, Knut Großmann

Technische Universität Dresden
Institut für Werkzeugmaschinen und Steuerungstechnik
01062 Dresden
{weller|helbig|groszman}@iwm.mw.tu-dresden.de

Abstract: Modellierungsprojekte werden heute aufgrund zunehmendem Umfang und steigender Komplexität kaum noch von einem einzigen Modellierer durchgeführt. Vielmehr werden Modelle parallel durch unterschiedliche Teilnehmer erstellt und anschließend zu einem Gesamtmodell integriert. Für die Verwaltung und das Zusammenführen der Teilmodelle ist das Konzept des Modell-Konfigurationsmanagements (Modell-KM) grundsätzlich geeignet. In der Praxis erweist sich die Anwendung von Modell-KM, insbesondere durch Domänenexperten, jedoch als kompliziert. Der vorliegende Beitrag stellt eine Methode vor, die die Nutzung von Modell-KM anleitet und so dessen praxisorientierten Einsatz unterstützt.

1 Einleitung

Modelle haben sich seit langem als Medium der Unternehmensdokumentation etabliert [Fe09]. Je nach Art und Umfang der betrachteten Domäne kann es sinnvoll sein, die Modellierung nicht zentral, sondern dezentral innerhalb einzelner Teilbereiche durchzuführen. Durch die Integration der Teilmodelle entsteht schließlich das Gesamtmodell des betrachteten Originals. Diese verteilte Modellierung wird in der Praxis zunehmend eingesetzt [St04]. Vor allem in umfangreichen Projekten im industriellen Umfeld oder in interdisziplinärer Forschung ermöglicht dieses Vorgehen die aktive Mitwirkung der jeweiligen Domänenexperten an der Modellierung, so dass das Fachwissen direkt in die Modelle einfließen kann [Be03].

Die verteilte Modellierung ist jedoch mit zahlreichen Herausforderungen verbunden. Aufgrund der Modellmodifikationen im Zuge der Unternehmensentwicklung und durch die Schwierigkeiten bei der Administration verteilter Modelle ergibt sich die Notwendigkeit, technische und organisatorische Ansätze für die Modellverwaltung zu finden [Br03b]. Diese Forderung wird verstärkt, wenn nicht nur die Modelle verändert werden, sondern auch die Modellierungssprache einer kontinuierlichen Anpassung unterliegt, wie dies z. B. bei Verwendung situationspezifischer Methoden eintreten kann [Sa06, EW07].

In der Literatur wird das Modell-Konfigurationsmanagement (Modell-KM) als ein möglicher Lösungsansatz zur Modellverwaltung genannt [Th07, Br10]. Das methodische Vorgehen bei einer praktischen Umsetzung dieses Ansatzes bleibt dabei jedoch offen. Insbesondere im industriellen Umfeld ist dies jedoch von besonderer Bedeutung, da hier sehr häufig Nicht-Modellierungsexperten an der Modellerstellung mitwirken.

Der vorliegende Beitrag widmet sich diesem Defizit. Hierfür wird eine Methode für den praktischen Einsatz von Modell-Konfigurationsmanagement vorgestellt, welche in einem industriellen Modellierungsprojekt entstanden ist. Die Methode berücksichtigt dabei parallele Änderungen am Modell, an der Sprache sowie an ggf. eingesetzten Modellierungswerkzeugen. Sie adressiert dabei explizit auch Nicht-Modellierungsexperten, so dass weder durch die Auswirkungen von Änderungen an den komplexen Beziehungen zwischen Modell, Sprache und Werkzeug noch durch die zusätzliche Verwaltungsebene des Modell-KM eine Verminderung der Akzeptanz der Modellierer hervorruft wird.

Unsere Forschung wird dem Design Science zugeordnet [He04]. Nach einer Einführung grundlegender Begriffe wird, angelehnt an die Forschungsmethodik von Verschuren und Hartog [VH05], zunächst eine Anforderungsanalyse durchgeführt. Danach wird die erstellte Methode, die bei der Umsetzung des KM-Ansatzes helfen soll, vorgestellt. Wesentlicher Schwerpunkt ist dabei die Berücksichtigung von Nicht-Modellierungsexperten als Zielgruppe der Methode. Abschließend wird gezeigt, wie und mit welchen Erfahrungen die Methode in einem industriellen Modellierungsprojekt eingesetzt wurde. Der Beitrag schließt mit einer Zusammenfassung der gewonnenen Erkenntnisse und einer Diskussion über zukünftige Forschungspotenziale.

2 Begriffliche Grundlagen

„Ein *Modell* ist das Ergebnis einer Konstruktion eines Modellierers, der für Modellnutzer eine Repräsentation eines Originals zu einer Zeit als relevant mit Hilfe einer Sprache deklariert [...]. Ein Modell setzt sich somit aus der Konstruktion des Modellierers, dem Modellnutzer, einem Original, der Zeit und einer Sprache zusammen“ [Sc98]. Ausgehend von dem konstruktivistischen Modellbegriff wird deutlich, dass jedes Modell einem direkten Zweckbezug unterliegt. Alle Tätigkeiten in Bezug auf die Modellerstellung, sowie die Modellmodifikation werden unter dem Begriff des *Modellierens* zusammengefasst. Im weiteren Sinn umfasst dieser Begriff nicht nur alle Aktivitäten in Zusammenhang mit der Modellkonstruktion, sondern auch die Gestaltung und Nutzung der Modelle [Br03a]. Die Tätigkeit des Modellierens wird durch den *Modellierer* übernommen.

Wie der Modelldefinition zu entnehmen ist, steht ein Modell immer in direktem Bezug zu einer *Modellierungssprache*. In der Wirtschaftsinformatik werden vorrangig semiformale Sprachen verwendet, d. h. die Sprachen besitzt sowohl Elemente aus einer natürlichen, als auch einer formalen Sprache. Die Syntax einer semiformalen Sprache stellt hierbei den formalen Teil dar, während die Semantik nicht oder nur unzureichend spezifiziert ist [Fr99].

Gemäß [Fr00] hat die Modellierungssprache direkten Einfluss auf die Qualität der Modelle. Dies liegt darin begründet, dass nicht jede Sprache in der Lage ist alle Sachverhalte optimal darzustellen. Dem Ansatz der situationsspezifischen Methodenentwicklung folgend, entstehen aus diesem Grund stetig neue bzw. modifizierte Modellierungssprachen. Da diese für einen speziellen Modellierungszweck konzipiert sind, können sie die Qualität der Modelle stark verbessern. Da dieser Zweck jedoch zu Beginn des Modellierungsprojektes nicht immer klar umrissen ist und sich während großer Modellierungsprojekte im Projektverlauf ändern kann, unterliegt auch die Modellierungssprache während der Modellierung einer möglichen kontinuierlichen Anpassung und Änderung.

Werden im Rahmen der Modellierung *Modellierungswerzeuge* eingesetzt, müssen diese den möglichen Sprachänderungen ebenfalls angepasst werden. Für die gängigen Modellierungssprachen wird eine Reihe von sprachbasierten Tools auf dem Markt angeboten. Hierzu zählt beispielsweise das ARIS-Toolset, welches direkt auf die Modellierung mit ARIS abgestimmt ist. Diese Werkzeuge erlauben jedoch nur bedingt Anpassungen an Sprachänderungen. Im Gegensatz dazu erlauben generischen Tools (auch Meta-Modellierungswerzeuge genannt) den *Methoden- und Software-Entwicklern* auch umfangreich Anpassungen des Werkzeugs, um Modifikationen an der Modellierungssprachen umzusetzen. Abbildung 1 visualisiert die Zusammenhänge der vorgestellten Begrifflichkeiten noch einmal.

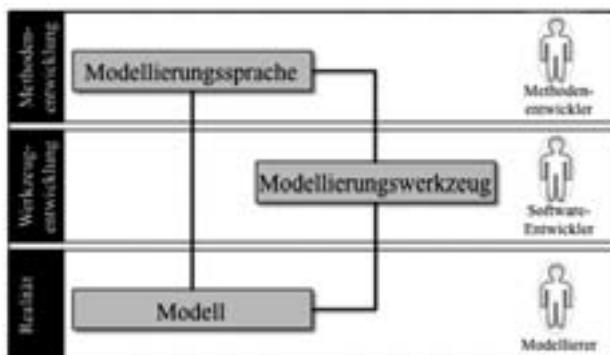


Abbildung 1: Zusammenhang der vorgestellten Begriffe

3 Anforderungsanalyse

Für die Ermittlung von Anforderungen an die zu erstellende Methode werden im Folgenden zunächst Szenarien identifiziert, welche im Umfeld der verteilten Modellierung auftreten können. Der Schwerpunkt liegt hierbei auf möglichen Änderungen an den verteilt erstellten Modellen. Darüber hinaus wird jedoch auch von einer Weiterentwicklung von Werkzeug und Sprache parallel zur Modellierung ausgegangen.

Damit lassen sich insgesamt drei Anwendungsfälle identifizieren (vgl. Abbildung 2). Dem ersten Anwendungsfall (Modell ändern) liegt in den meisten Fällen ein inhaltliches Defizit zugrunde. Dies tritt ein, wenn festgestellt wird, dass Sachverhalte nicht vollständig dargestellt wurden, nicht detailliert genug modelliert wurden oder gänzlich falsch erfasst wurden. Ebenso kann dieser Fall eintreten, wenn sich das betrachtete Original ändert und demnach das (Teil-)Modell angepasst werden muss oder das Modell in eine neue Sprachversion überführt werden muss. Die Behebung dieses Defizits hat in der Regel keinen Einfluss auf Sprache oder Werkzeug, sondern ruft lediglich eine neue Modellversion hervor. Obwohl kein Einfluss auf die weiteren Komponenten besteht, können doch während der Modellierung Defizite festgestellt werden, die eine Modifikation an Sprache oder Werkzeug verlangen.

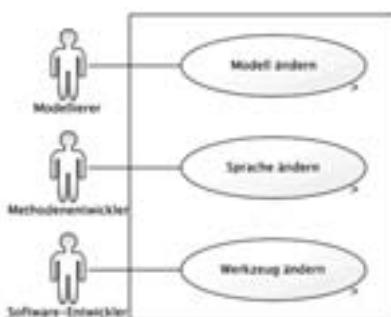


Abbildung 2: Anwendungsfälle

Anwendungsfall 2 (Sprache ändern) tritt dann ein, wenn aus der Entwicklung oder aus der Modellierung heraus Defizite an der Beschreibbarkeit des Originals festgestellt werden. Diese Feststellung kann einerseits durch den Methodenentwickler erfolgen, der an einer Verbesserung der Qualität der Sprache arbeitet oder durch den Modellierer selbst, der mit Hilfe der bestehenden Sprache an die Grenzen der Modellierung stößt, jedoch noch nicht alle Sachverhalte abdecken kann. Behoben wird dieses Defizit durch den Methodenentwickler. Dies kann durch die Erstellung neuer Sprachkonzepte, sowie durch die Anpassung oder Zusammenlegung alter Sprachkonzepte realisiert werden. Aus dem vorhergehenden Abschnitt ist bereits bekannt, dass ein direkter Bezug zwischen Modellierungssprache und Modellierungswerkzeug besteht, da das Werkzeug auf Basis der Sprache eingesetzt wird. Es ergibt sich somit für die meisten Sprachänderungen ebenfalls eine Änderung am Werkzeug (vgl. Anwendungsfall 3), damit die Konformität zwischen beiden wieder hergestellt wird. Die Notwendigkeit einer nachfolgenden Werkzeuganpassung wird bei jeder Änderung an der Sprache geprüft. Des Weiteren ist der Bezug zwischen einem Modell und dem Werkzeug mit dem es erstellt wurde bzw. der Bezug zu der Sprache in der es erstellt wurde ebenfalls aus Abschnitt 2 bekannt. Dies führt dazu, dass nach einer Sprachänderung (evtl. gefolgt von einer Werkzeugänderung) auch auf eine Notwendigkeit zur Änderung der Modelle geprüft werden muss. Um weiterhin alle drei Komponenten konform zu halten, wird dies in den meisten Fällen notwendig sein. Ausnahmen ergeben sich beispielsweise bei abgeschlossenen Projekten oder wenn eine Modellanpassung nicht gewünscht ist [EW07].

Der dritte Anwendungsfall (Werkzeug ändern) tritt beispielsweise als Folge des zweiten Anwendungsfalls auf, in dem das Werkzeug an eine neue Sprache angepasst werden muss. Darüber hinaus handelt es sich bei Werkzeuganpassungen meist um die Behebung von Bedienungsdefiziten. Eine schlechte Bedienbarkeit des Werkzeugs kann indirekt Einfluss auf die Qualität der Modelle haben, daher sollte eine Optimierung diesbezüglich angestrebt werden [Nu02]. Die Modifikation wird durch den Software-Entwickler vorgenommen. Die Identifikation der Defizite kann jedoch durch alle Beteiligten erfolgen. Ebenso wie bei Anwendungsfall 2 wird anschließend die Auswirkung auf die anderen Komponenten geprüft und diese ggf. ebenfalls wieder angepasst. Bei der Nutzung generischer Modellierungswerkzeuge sind Sprach- und Werkzeugänderungen zudem nicht immer eindeutig voneinander abgrenzbar.

Im Kontext der getroffenen Annahmen sind Modifikationen der verschiedenen Artefakte sowohl einzeln, als auch parallel zueinander denkbar. Somit ist ein Szenario, bei dem, sich Modelle einer Modellierungssprache verändern, gleichzeitig jedoch auch eine Weiterentwicklung der Sprache stattfindet, nicht ausgeschlossen. Vor allem beim Einsatz in verteilten Projekten ist eine solche parallele Weiterentwicklung häufig gegeben. Aufgrund der beschriebenen Wechselwirkungen ergibt sich somit eine ständige Generierung neuer Anforderungen, die sich aus Änderungen ergeben. Diese Zusammenhänge sind in Tabelle 1 zusammengefasst. Ziel der weiterführenden Betrachtung soll es sein diese Wechselwirkungen handhabbar zu verwalten, so dass jederzeit auf eine konforme Kombination von Modellierungssprache, -werkzeug und Modell verfügt werden kann. Eine mögliche Lösung wird im folgenden Abschnitt skizziert.

Einfluss auf			
	Modell	Werkzeug	Sprache
Modell ändern	-	Kein direkter Einfluss; Identifikation von Defiziten möglich	Kein direkter Einfluss; Identifikation von Defiziten möglich
Werkzeug ändern	Anpassung meist notwendig	-	Kein direkter Einfluss; Identifikation von Defiziten möglich
Sprache ändern	Anpassung meist notwendig	Anpassung meist notwendig	-

Tabelle 1 Zusammenfassung der Wechselwirkungen

4 Methode für die Umsetzung des Modell-KM

In den vorangegangen Abschnitten wurden die Ausgangssituation und die Motivation der Verwaltung von Modell-, Sprach- und Werkzeugversionen bei der verteilten Modellierung betrachtet. Um hierbei auftretende Konflikte zu vermeiden oder zumindest kontrollierbar zu machen, bedarf es einer Lösung zur Verwaltung und Handhabung der entstehenden Artefaktversionen. In der Literatur wird das Modell-KM als Lösungsansatz genannt [Sa06, Th07]. Im Folgenden wird zunächst dieser Ansatz diskutiert und aufbauend auf den skizzierten Anleitungsdefiziten anschließend die entwickelte Methode vorgestellt.

4.1 Grundlagen des Modell-KM

Das zentrale Element des KM ist die Version. Eine *Version* bezieht sich immer auf einen speziellen Zustand eines Objektes oder Systems [CW98, Ze97, Th07]. Bei der Betrachtung von Modellen kann demnach jedes einzelne Modellelement, aber auch das Modell selbst versioniert werden. Darüber hinaus unterliegt die Sprache bzw. jedes Sprachkonstrukt und das Werkzeug bzw. dessen Komponenten einer Versionierung. Bei den genannten Artefakten handelt es sich um sogenannte *Konfigurationseinheiten* (KE). Eine *Konfiguration* beschreibt gemäß [Di04] miteinander verbundene physische und funktionelle Merkmale, welche ein Produkt näher klassifizieren. Eine Konfiguration ist somit eine Zusammenfassung von Konfigurationseinheiten in den jeweiligen miteinander konformen Versionen. Die Konfiguration ist ihrerseits ebenfalls eine KE und unterliegt somit ebenso der Versionierung.

Sowohl technische, als auch organisatorische Aspekte der Verwaltung von Konfigurationen werden durch das *Konfigurationsmanagement* (KM) übernommen, welches allgemein in der [Di04] beschrieben ist. Für diese Verwaltungsaufgaben in Bezug auf die Konfigurationseinheiten und ihrer Änderungen werden im KM folgende Aktivitäten durchgeführt [Di04]:

- Im Rahmen der **KM-Planung** werden die organisatorischen Rahmenbedingungen für den KM-Einsatz geschaffen.
- Die **Konfigurationsidentifizierung** legt die Granularität der verwalteten Konfigurationseinheiten fest.
- Die **Änderungslenkung** umfasst Aktivitäten zur Dokumentation, Bewertung und Umsetzung von Änderungswünschen an den verwalteten Konfigurationseinheiten.
- Im Rahmen der **Konfigurationsbuchführung** werden Änderungen an den Konfigurationseinheiten dokumentiert.
- **Konfigurationsaudits** sollen schließlich sicherstellen, dass dokumentierten Änderungen auch tatsächlich realisiert wurden.

Das Modell-KM ist eine Technologie, die auf dem allgemeinen Ansatz des Konfigurationsmanagements aufbaut, als Produkt jedoch Modelle betrachtet [Br10]. Im Kontext der Meta-Modellierung können darüber hinaus auch die verwendete Modellierungssprache [EW07] sowie Konfigurationen am Modellierungswerkzeug [Cu10] abgelegt werden.

Für die Verwaltung der Konfigurationseinheiten kommen zumeist KM-Systeme zum Einsatz [Es02]. Diese unterstützen nicht nur die Änderungsverwaltung von Modellen und Modellierungssprachen, sondern häufig auch die verteilte Modellierung [Th07]. Derartige KM-Systeme bilden den Schwerpunkt der bisherigen Forschung im Bereich von Modell-KM [Gr04, Sa06].

Dabei wird zumeist die technische Verwaltung der Konfigurationseinheiten im Rahmen der Konfigurationsbuchführung betrachtet, eine Beschreibung der restlichen KM-Aktivitäten erfolgt hingegen nicht. Bei einer gleichzeitigen Verwaltung aller Versionen der KE ist darüber hinaus zu beachten, dass jede Änderung an einem Artefakt zu einer neuen Version der gesamten Konfiguration führt. Konkrete Beschreibungen, wie in der Praxis mit Änderungen umgegangen werden muss und welche Aufgaben mit welchen Werkzeugen von welchen Aufgabenträgern durchzuführen sind, fehlen jedoch. Diese Lücke soll die im Folgenden vorgestellte Methode schließen.

4.2 Methode für die Umsetzung des KM

Eine Methode dient der Anleitung für den Umgang mit gewissen Sachverhalten. Im Rahmen dieses Beitrages soll eine Methode vorgestellt werden, die die Umsetzung der Verwaltung von Modell-, Sprach- und Werkzeugversionen dient. Die Methode ist dabei im Wesentlichen in drei Teile gegliedert. Der strukturelle Teil beschreibt Objekte, die für die Umsetzung benötigt werden. Im Rollenkonzept werden konkrete Aufgabenblöcke generiert und den jeweiligen Rollen zugewiesen. Abschließend gilt es das Vorgehen zu beschreiben.

4.2.1 Struktur

Strukturell ergibt sich der Aufbau der Methode aus einer Ableitung der in der [Di04] beschriebenen Aktivitäten des KM. Des Weiteren basiert die Methode auf der Annahme einer zentralen Modellverwaltung [Gr04, Th07]. Der Transfer zwischen der zentralen Datenbasis und den einzelnen Workspaces der Bearbeiter erfolgt dabei durch CheckIn-/CheckOut-Sequenzen [EGK02]. Die vollständige Methode besteht aus mehreren Diagrammen, wobei das detaillierte Vorgehen der beschriebenen Aktivitäten innerhalb von Verfeinerungen dargestellt ist. Die wesentlichen Objekte der Vorgehensbeschreibung sind die Konfiguration und die Änderungsanträge (CR; vgl. Abbildung 3).

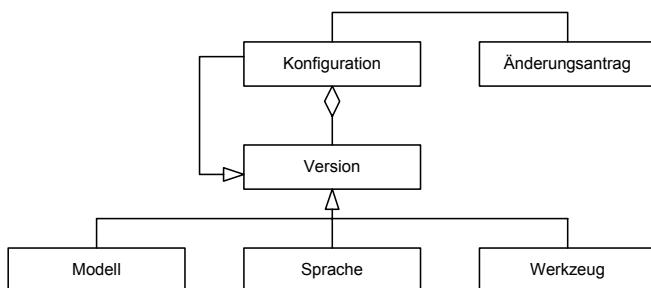


Abbildung 3: Elemente des Modell-KM (Ausschnitt; in Anlehnung an [Br07])

4.2.2 Rollenkonzept

Als Akteure bei der Umsetzung des Modell-KM werden die bereits eingeführten Rollen des Modellierers, des Methodenentwicklers und des Software-Entwicklers erneut aufgegriffen. Zudem ergeben sich aus den Anforderungen der verteilten Modellierung, sowie denen des KM zwei weitere Rollen.

Aufgrund der Annahme eines verteilten Modellierungsprojektes und der dabei entstehenden Konfliktpunkte kann die Rolle eines *Modellverantwortlichen* abgeleitet werden. Seine Aufgabe besteht im Management, der Schnittstellen zwischen den Teilmodellen und der Zusammenführung der Modelle. Eine zweite Aufgabe besteht in der Anpassung der Modelle an bestehende Änderungen der Sprache und des Werkzeugs. Der Modellverantwortliche übernimmt somit Aufgaben des Modellierers. Die Reduzierung der Hemmnisse kann durch ein möglichst einfaches und klares Vorgehen erreicht werden. Verfügt der Modellierer nicht über ausreichend Modellierungskenntnis, kann es daher sinnvoll sein die Zusammenführung der Teilmodelle, sowie der Umgang mit Konfigurationen in einer Rolle zu zentralisieren. Der Modellierer übernimmt demnach lediglich die Modellierung seines Teilbereichs.

Aus den Anforderungen der [Di04] lässt sich die Rolle der *Verfügungsstelle* als zentrales Organ zur Verwaltung von Änderungswünschen ableiten. Die Hauptaufgaben liegen in der Prüfung, Verifizierung und Initiierung eines Änderungsvorhabens. Dieser wird von der Verfügungsstelle über den Änderungsprozess überwacht und abschließend auf seinen Erfüllungsgrad erneut geprüft. Tabelle 2 fasst die Rollen und ihre Aufgaben noch einmal zusammen.

Rolle	Aufgaben
Modellierer	<ul style="list-style-type: none"> - Unternehmensdokumentation - Identifikation von Defiziten an Sprache oder Werkzeug
Modellverantwortlicher	<ul style="list-style-type: none"> - Zusammenführung von Teilmodellen - Überführung von Modellen in neue Sprachversionen - Sicherstellung der Konsistenz - Schnittstelle zwischen Modellierer und Entwicklern
Methodenentwickler	<ul style="list-style-type: none"> - Entwicklung und Änderung der Sprache
Software-Entwickler	<ul style="list-style-type: none"> - Entwicklung und Änderung des Werkzeugs - Anpassung an Sprachmodifikationen
Verfügungsstelle	<ul style="list-style-type: none"> - Verwaltung der Änderungsanträge - Prüfung, Verifizierung und Initiierung von Änderungsaktivitäten - Änderungsprozessüberwachung - Kontrolle

Tabelle 2: Rollenkonzept

4.2.3 Vorgehen

Das Vorgehen der Methode basiert auf den Aktivitäten des KM gemäß [Di04]. Dabei kann die strukturelle Zuweisung der Rollen, die im vorangegangenen Abschnitt vorgenommen wurde, als Teil der Konfigurationsplanung angesehen werden. Darüber hinaus gilt es, genaue Richtlinien aufzustellen, Verantwortlichkeiten eindeutig zu regeln und technische Ressourcen zu benennen. Die Einführung der Methode als Anleitung für das weitere Vorgehen wird ebenfalls innerhalb der Konfigurationsplanung den Rollen entsprechend durchgeführt.

Zur Erfüllung des Anspruches an Deutlichkeit und Einfachheit wurden die Methodenabläufe mittels Aktivitätsdiagrammen der UML [Ob09] dokumentiert. Basierend auf bestehenden Projekterfahrungen kann diese Darstellungsform auch von Nicht-Modellierungs-experten schnell erfasst und korrekt interpretiert werden. Die Anzahl der Aktivitätsdiagramme der obersten Hierarchieebene lässt sich aus den Anwendungsfällen ableiten. Diese Aktivitäten gilt es in der Methode genauer zu beschreiben, so dass die Durchführung eines solchen Projektes angeleitet wird. Im Folgenden werden die Aktivitäten auszugsweise vorgestellt.

Bei der Verwaltung von Sprachänderungen (vgl. Abbildung 4) ist der Modellierer selbst nicht betroffen. Die Initiierung kann, wie im Anwendungsfall beschrieben, durch ihn erfolgen, an der Durchführung hat er jedoch keinen Anteil. Die Initiierung ist durch eine Änderungsanfrage (Change Request CR) gekennzeichnet. Ausgehend von vorliegenden CR wird eine Prüfung durch die Verfügungsstelle vorgenommen. Sofern die Änderung als notwendig freigegeben wird, können die Methodenentwickler an der Beseitigung der Defizite bzw. der Erfüllung der Anforderungen arbeiten, andernfalls wird der Antrag verworfen bzw. mit einer neuen Priorität verschoben.

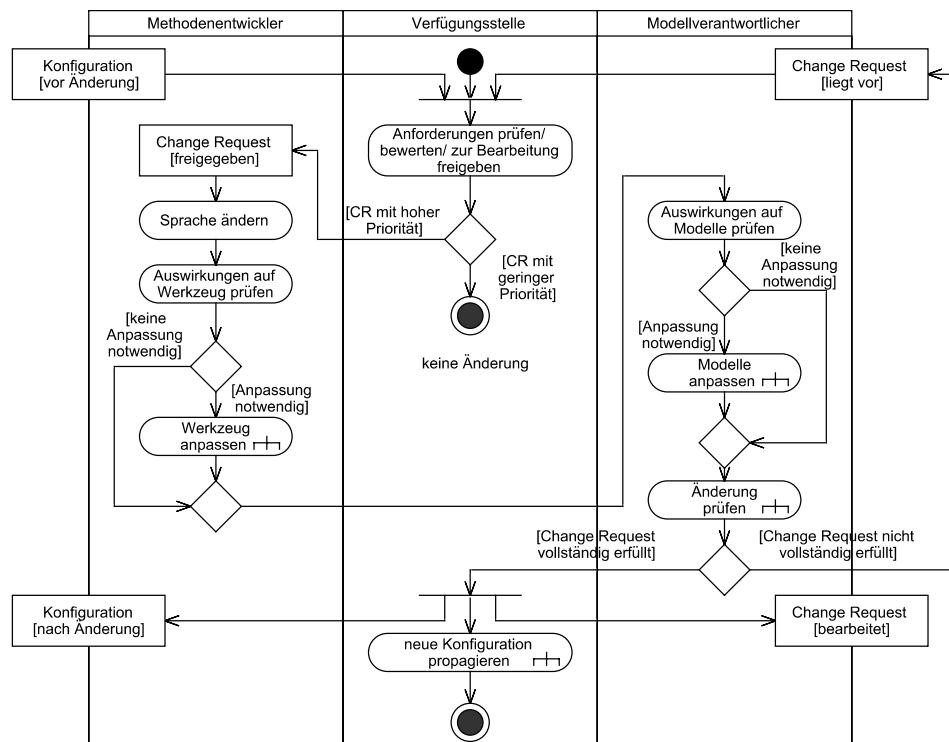


Abbildung 4: Änderungslenkung bei der Modifikation der Sprache (Darstellung gestaucht)

Im Anschluss an eine solche Modifikation der Sprache erfolgt eine Prüfung hinsichtlich der möglichen Auswirkungen, wie sie bereits im Anwendungsfall diskutiert wurde. Dabei wird zuerst durch den Methodenentwickler geprüft, ob eine Veränderung am Werkzeug notwendig geworden ist. Ist dieser Fall eingetreten, wird die Änderung vorgenommen, damit eine Konformität zwischen Sprache und Werkzeug wiederhergestellt ist. Darüber hinaus sieht die Methode das Überspringen dieses Schrittes vor, falls dieser für unnötig erachtet wird. Als Abschluss der Auswirkungsprüfung werden die Folgen auf die Modelle durch den Modellverantwortlichen geprüft. Wie bereits bei dem Werkzeug bietet die Methode an dieser Stelle ebenfalls die Möglichkeit, den Schritt der Änderung bei Nichtbedarf zu Überspringen.

Die so entstandenen Konfigurationen der einzelnen KE werden zu einer Gesamtkonfiguration zusammengefasst und abschließend durch die Verfügungsstelle hinsichtlich des Erfüllungsgrades der CR bewertet. Bei positiver Prüfung kann die Modifikation propagiert werden, bei negativer Prüfung gilt die Anforderung als weiterhin bestehend, so dass die Methode erneut abgearbeitet werden muss. Diese Prüfung entspricht in Teilen dem Konfigurationsaudit.

Die Richtlinien für die Konfigurationsbuchführung [Di04] gehören zu den Bestandteilen der Konfigurationsplanung. Wie oben beschrieben, kann die technische Umsetzung in einem KM-System erfolgen. Dieses System muss einerseits zur Verwaltung der KE eingesetzt werden können, andererseits besteht die Anforderung, dass das Änderungsmanagement ebenfalls unterstützt werden muss. Bei einer Realisierung der Kombination beider Ansprüche ist die werkzeugbedingte teilautomatisierte Konfigurationsbuchführung vorbereitet und kann durch weitere Dokumentationen ergänzt werden.

Die Publikation der Konfiguration als Ergebnis einer Änderung ist sowohl Bestandteil dieses Szenarios, als auch des zweiten Beispiels aus der Methode. Das Zugänglichmachen der Konfiguration für eine weiterführende Bearbeitung ist somit wesentlicher Bestandteil der Umsetzung des KM. Den Modellierer selbst interessiert das Vorgehen bei der Änderung von Sprache oder Werkzeug nur dann, wenn er direkt beteiligt ist. Aufgrund der Annahme von Domänenexperten als Modellierern und dem Bestreben nach einer Vereinfachung des gesamten Verwaltungsprozesses ergibt sich die Modifikation von Modellen als einzige Teilaktivität, die vom Modellierer durchgeführt wird. Die Beschreibung über den Umgang mit Modellen bei einer verteilten Modellierung muss daher sowohl einfach, als auch detailliert sein und ist damit auch Bestandteil der Methode (vgl. Abbildung 5).

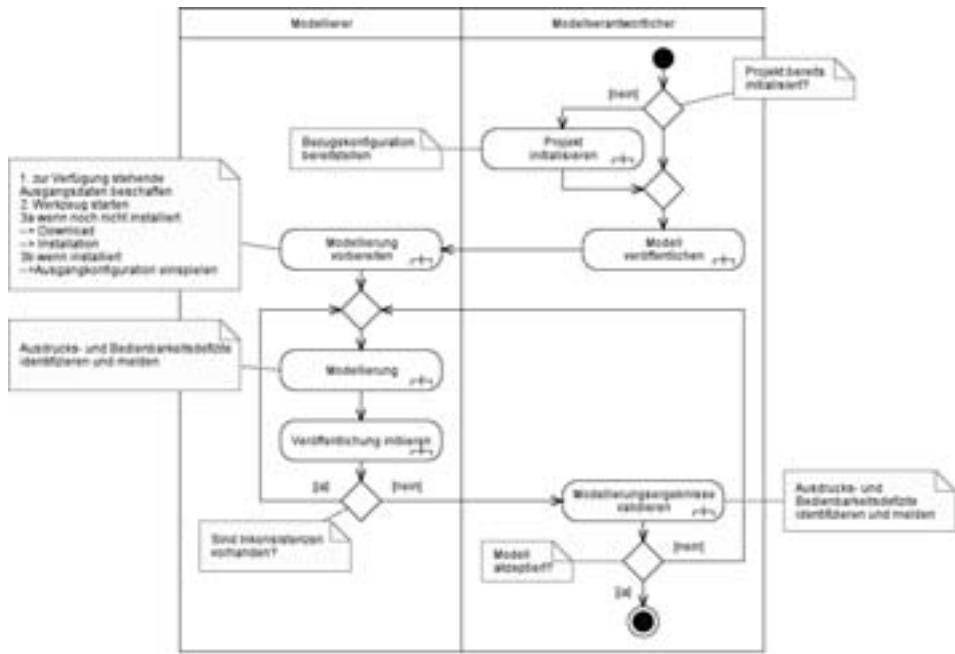


Abbildung 5: Änderungslenkung bei der Modellmodifikation (Darstellung gestaucht)

Dieses Szenario beschränkt sich auf die Interaktion zwischen Modellierer und Modellverantwortlichen. Der Modellverantwortliche ist dabei in der Pflicht eine aktuelle Konfiguration für die Bearbeitung zur Verfügung zu stellen. Der Modellierer wiederum ist verpflichtet dazu, die aktuelle Version der Konfiguration zur Bearbeitung heranzuziehen, diese nach der Bearbeitung dem Modellverantwortlichen zukommen zu lassen und vor einer weiteren Bearbeitung die erneute Bereitstellung der neuen Gesamtkonfiguration abzuwarten. Auf diese Weise kann sichergestellt werden, dass der Modellierer immer mit der aktuellsten Version von Sprache und Werkzeug am Modell weiterarbeitet, so dass Konformitätsprobleme bei der Zusammenführung der Teilmodelle reduziert werden.

5 Anwendung der Methode

Die Anwendbarkeit der vorgestellten Methode konnte bereits in einem industriellen Forschungsprojekt [Hu06] nachgewiesen werden. Die Vorbereitung für den Einsatz in weiteren Projekten wird derzeit getroffen. Im Folgenden wird die Umsetzung in diesem praktischen Umfeld skizziert und die Erfahrungen mit dem Methodeneinsatz beschrieben. Abschließend werden mögliche Potenziale für den weiteren Einsatz abgeleitet.

5.1 Projektbeschreibung

Im involvierten Projekt werden innovative Verbundwerkstoffe und entsprechende Technologien und Verfahren für deren Produktion entwickelt. Aufgrund der Neuartigkeit der Werkstoffe und der Technologie sind Abhängigkeit zwischen Eigenschaften der Produkte und Einstellungen an den zum Einsatz kommenden Maschinen bisher kaum bekannt. Für eine reproduzierbare Fertigung der Verbundwerkstoffe ist es jedoch zwingend notwendig, zu wissen, welche Einstellungen an den Maschinen welche Auswirkungen auf die Qualität der erzeugten Produkte aufweisen.

Um diese Auswirkungen zu ermitteln, werden im vorgestellten Forschungsprojekt Prozessen und konkrete Prozessdaten aus experimentellen Prozessdurchläufen gesammelt und anschließend analysiert [WJG10]. Hierfür musste zunächst die gesamte Prozesskette dokumentiert werden. Dabei waren die betrachteten Prozesse anfänglich nicht feststehend, sondern wurden und werden im Laufe des Projektes erst entwickelt. Entsprechend unterliegen auch die Prozessmodelle einer stetigen Veränderung. Um diese Veränderungen dauerhaft zu erfassen, wurde im Projekt eine dezentrale Modellierung durch die Fachbereiche selbst angestrebt.

Da auch die Methodik zur Analyse und Steuerung der Prozesse erst innerhalb des Projektes erstellt wurde, war zu Beginn der Modellierung folglich auch die Sprache noch nicht endgültig bekannt. Gleichzeitig existierten jedoch in den Fachbereichen bereits grobe Vorstellungen über die Darstellung von Prozessen. Aus diesen Gründen wurde ein iterativer Modellierungsansatz auf Basis der situationsspezifischen Methodenentwicklung gewählt, was jedoch zu kontinuierlichen Änderungen an der Sprache und folglich auch am Modellierungswerkzeug sowie an den Modell führte. Da die Mitarbeiter der Fachbereiche jedoch keine Modellierungsexperten sind und folglich auch die komplexen Zusammenhänge zwischen Modell, Sprache und Modellierungswerkzeug nicht überblicken, war eine einfache Methode zur Änderungsverwaltung notwendig.

5.2 Erfahrungen

Um unnötige Hemmnisse bei der verteilten Modellierung zu vermeiden, war folglich eine Trennung in die Rollen des Domänenexperten als Modellierer und des zentralen Modellverantwortlichen notwendig. Den Nicht-Modellierungsexperten wird somit der Zugang zum Modellieren als Mittel der Prozessbeschreibung so einfach wie möglich gemacht, indem alle administrativen Aufgaben anderweitig vergeben werden.

Zu Beginn des Projektes wurde zunächst die Konfigurationsplanung durchgeführt. Hierbei wurde die entwickelte Methode eingeführt. Die Rollen wurden konkreten Aufgabenträgern zugewiesen. Als Verfügungsstelle wurde ein Team von drei Mitarbeitern eingeteilt, die ebenfalls die Aufgaben der Methoden- und Software-Entwickler, sowie die des Modellverantwortlichen wahrnehmen. Bei den Modellierern handelt es sich um Experten der jeweiligen Fachbereiche ohne Vorkenntnis auf dem Gebiet der (Meta-)Modellierung. Des Weiteren wurden im Rahmen der Planung die zu verwendenden Werkzeuge ausgewählt. Als KM-System kam das Modellierungswerkzeug Cubetto Toolset zum Einsatz [Cu10]. Dieses Werkzeug ist als generisches Modellierungswerkzeug in der Lage sowohl Änderungen an Modellen, an der Modellierungssprache als auch am Werkzeug zu verwalten. Da es jedoch keine Verwaltung von Änderungsanträgen beherrscht, wurde zusätzlich ein Bugtracking-System eingeführt [Bu10].

Die Identifikation der KE konnte im Projekt entfallen, da diese bereits durch die Definition des Modell-KM gegeben sind. Die Konfigurationsbuchführung muss ebenfalls nicht gesondert betrachtet werden, da diese von den KM-Systemen übernommen wird. Die Aktivitäten der Änderungslenkung und der Konfigurationsaudits sind durch die beschriebene Methode und ihre Verfeinerungen vorgegeben. Für die Sicherstellung einer korrekten Umsetzung der Aktivitäten wurden Schulungen mit den Projektpartnern durchgeführt. Zuzüglich dazu wurde die Methode in Form der Aktivitätsdiagramme im Intranet veröffentlicht, so dass sie für die Modellierer jederzeit einzusehen ist.

Die Erfahrungen im Umgang mit der Methode waren überaus positiv. Im Rahmen von qualitativen Interviews wurde von Seiten der Projektpartner vor allem die Entlastung der Modellierer der einzelnen Fachbereiche gelobt. Der empfundene Aufwand für die Domänenexperten hielt sich somit wie erhofft sehr gering, so dass die Akzeptanz der Modelle als nützliches Hilfsmittel deutlich stieg. Ein weiterer Akzeptanz fördernder Aspekt ist die Geschwindigkeit, mit der Änderungsanträge bearbeitet werden. Dies wurde vor allem durch Kommunikation auf kurzem Dienstweg realisiert und gilt sowohl für Änderungen, als auch für Ablehnungen von Änderungswünschen.

Die Entlastung der Domänenexperten wurde jedoch mit starken Belastungen der Verfügungsstelle sowie des Modellverantwortlichen erkauft. Um diese zu verringern, wurde eine zusätzliche Konsistenzprüfung im Modellierungswerkzeug integriert. Hierdurch erfolgt bereits beim Modellierer selbst eine Vorprüfung der erstellten Modelle. Dies hatte den positiven Nebeneffekt, dass die Modellierer für mögliche Inkonsistenzen sensibilisiert wurden.

Durch die Nutzung der Methode können im Rahmen des Projektes Änderungen an der Modellierungssprache, am Werkzeug und an den Modellen parallel vorgenommen werden. Das Konfliktpotenzial wird dabei so niedrig wie möglich gehalten. Die Beziehungen zwischen Modell, Modellierungssprache und Modellierungswerkzeug werden durch das Modell-KM verwaltet. Somit werden die zugrundeliegenden Abhängigkeiten kontrolliert. Das CheckIn-/CheckOut-Prinzip führt darüber hinaus dazu, dass jede Modifikation auf Basis der aktuellen Version vorgenommen wird.

Die Dokumentation in Form von UML-Aktivitätsdiagrammen und die Bereitstellung des Modells über das Intranet (siehe Abbildung 6) hat sich ebenfalls als vorteilhaft erwiesen. Die Darstellungsform führt wie erhofft zum Verständnis der Methode bei Nicht-Modellierungsexperten. Ein weiterer Vorteil liegt in der schnellen Erfassung. Diese ermöglicht im Vertretungsfall eine zügige Einarbeitung der Mitarbeiter.

Kritisch bleibt anzumerken, dass die iterative Entwicklung der eingesetzten Modellierungssprache zu Beginn des Projekts zu zahlreichen Anpassungen des Werkzeugs führte, wodurch die Verfügungsstelle sowie der Methoden- und Softwareentwickler stark belastet wurden. Dies ist dem Vorgehen bei einer kontinuierlichen Anpassung im Rahmen der situationsspezifischen Methodenentwicklung jedoch immanent und daher bei der Personal- und Aufwandsplanung zu berücksichtigen.

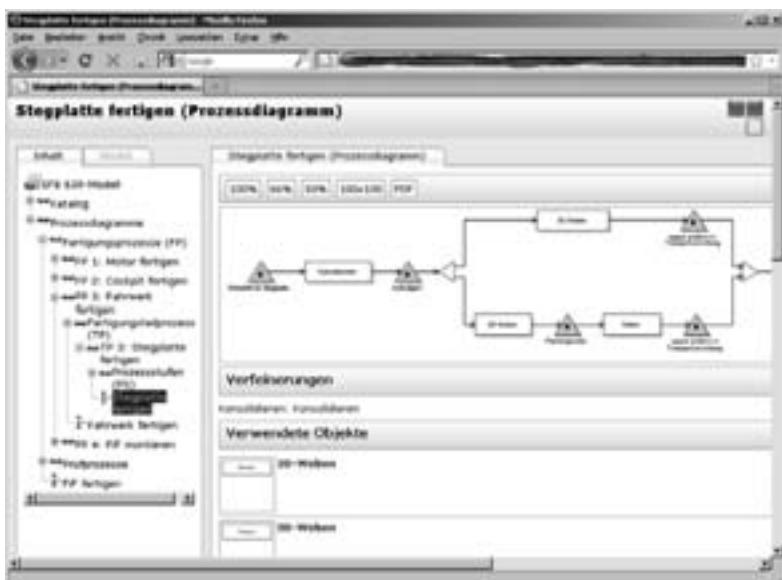


Abbildung 6: Zentrale Bereitstellung der aktuellen Konfiguration im Intranet

6 Zusammenfassung und Ausblick

Im vorliegenden Beitrag wurde eine Methode vorgestellt, die die Umsetzung von Modell-KM im praktischen Umfeld verteilter Modellierungsprojekte vereinfacht. Dabei wurde berücksichtigt, dass Modelle oft nicht das primäre Ziel eines Projektes sind, sondern lediglich als Kommunikations- und Dokumentationsmittel genutzt werden. Entsprechend wurde die Methode derart gestaltet, dass auch Nicht-Modellierungsexperten in die Lage versetzt werden schnell und einfach in verteilten Teams zu Modellieren.

Die Anwendung der Methode im Projekt hat gezeigt, dass die vorgestellte Methode dazu beiträgt, Hemmnisse in komplexen Modellierungsprojekten abzubauen. Die Methode kann damit helfen, die Technologie des Modell-KM in der Praxis umzusetzen, wodurch sich wertvolle Impulse für die Entwicklung der Technologie selbst sowie für die eingesetzten KM-Systeme ergeben können. Gleichzeitig werden durch die Anwendung der Methode aber auch die Methoden- und Modellierungsexperten stärker belastet. Diese können ihre Arbeitsplanung jedoch unabhängig vom Terminplan des Kunden durchführen und verfügen darüber hinaus über mehr Modellierungs-Knowhow, wodurch die Abarbeitung der Aufgaben schnelle als beim Domänenexperten erfolgen sollte. Die Entwicklung entsprechender KM-Werkzeuge sollte die durch die Anwendung der Methode entstehende Aufgabentrennung aber auch die spezifischen Anforderungen an die beteiligten Rollen zukünftig stärker beachten.

Unsere eigene Forschung wird sich verstärkt diesem Thema widmen und konkrete Hinweise für die Weiterentwicklung von KM-Systemen ermitteln. Darüber hinaus wird die vorgestellte Methode weiterentwickelt. So ist geplant, ausgehend von den erstellten Prozessmodellen ein Datenerfassungswerkzeug zu konfigurieren [WJG10]. Damit ergeben sich weitere Abhängigkeiten, da sich das Erfassungswerkzeug in Abhängigkeit der Modelle ebenfalls verändert. Die resultierenden Versionen des Erfassungswerkzeugs sowie die Abhängigkeiten und Gültigkeit der erfassten Daten sind Gegenstand unserer weiteren Forschung.

Literaturverzeichnis

- [Be03] Becker, J.; Knackstedt, R.; Kuropka, D.; Delfmann, P.: Konfiguration fachkonzeptioneller Referenzmodelle; In (Uhr, W.; Esswein, W.; Schoop, E., Hrsg.): Wirtschaftsinformatik 2003: Medien – Märkte – Mobilität, 2 Bde., Heidelberg, Physica-Verlag, 2003
- [Br07] Braun, R.; Esswein, W.; Gehlert, A.; Weller, J.: Configuration Management for Reference Models. In: Reference Modeling for Business Systems Analysis, 2007, S. 310-336.
- [Br10] Braun, R.: Referenzmodellierung: Grundlegung und Evaluation der Technik des Modell-Konfigurationsmanagements, Logos Verlag, Berlin, 2010.
- [Br03a] vom Brocke, J.: Referenzmodellierung: Gestaltung und Verteilung von Konstruktionsprozessen, In (Becker, J.; Grob, H.L.; Klein, S.; Kuchen, H.; Müller-Funk, U.; Vossen, G., Hrsg): Advances in Information Systems and Management Science, Bd. 4, Logos Verlag, Berlin, 2003
- [Br03b] vom Brocke, J.: Verteilte Referenzmodellierung: Gestaltung multipersoneller Konstruktionsprozesse, In Lecture Notes in Informatics, P-35, 2003, S. 238-242.
- [Bu10] Mozilla Foundation: Bugzilla, <http://www.mozilla.org>. Download: 12.04.2010
- [Cu10] Semture GmbH: Cubetto Toolset, 2010, <http://www.semture.de/cubetto>, Download: 12.04.2010
- [CW98] Reidar Conradi and Bernhard Westfechtel. Version models for software configuration management. ACM Computing Surveys, 30 (2), 1998, S. 232-282.
- [Di04] DIN Deutsches Institut für Normung e.V.: Qualitätsmanagement: Leitfaden für Konfigurationsmanagement (DIN ISO 10007:2003), Beuth Verlag, Berlin, 2004.
- [Di08] DIN Deutsches Institut für Normung e.V.: Qualitätsmanagementsysteme: Anforderungen (DIN EN ISO 9001:2008), Beuth Verlag, Berlin, 2008.

- [Es02] Estublier, J. et al.: Impact of the research community on the field of software configuration management: summary of an impact project report, In: SIGSOFT Software Engineering Notes, 27/5, 2002, S. 31-39.
- [EGK02] Esswein, W.; Greiffenberg, S.; Kluge, C.: Konfigurationsmanagement von Modellen, In: Modellierung betrieblicher Informationssysteme - MobIS 2002, Proceedings zur Tagung, 10, Nürnberg, 2002, S. 93-112.
- [EW07] Esswein, W.; Weller, J.: Method modifications in a configuration management environment, In: Proc. 15th Europ. Conf. on Information Systems, St. Gallen, 2007, S. 2002-2013.
- [Fe09] Fettke, P.: Ansätze der Informationsmodellierung und ihre betriebswirtschaftliche Bedeutung: Eine Untersuchung der Modellierungspraxis in Deutschland, In: Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung, 61, 2009.
- [Fr99] Frank, U.: Zur Verwendung formaler Sprachen in der Wirtschaftsinformatik: Notwendiges Merkmal eines wissenschaftlichen Anspruchs oder Ausdruck eines übertriebenen Szientismus?, In (Becker, J.; König, W.; Schütte, R.; Wendt, O.; Zelewski, S., Hrsg.): Bestandsaufnahme und Perspektiven. Wiesbaden: Gabler 1999, S. 127-160
- [Fr00] Frank, U.: Modelle als Evaluationsobjekt: Einführung und Grundlegung, In (Häntschel, I.; Heinrich, I.J., Hrsg.): Evaluation und Evaluationsforschung in der Wirtschaftsinformatik. München, Wien; Oldenbourg, 2000, S. 339-352
- [Gr04] Greiffenberg, S.: Methodenentwicklung in Wirtschaft und Verwaltung, Verlag Dr. Kovac, 2004.
- [He04] Hevner, A.R.; March, S.T.; Park, J.; Sam, S.: Design Science in Information Systems Research. In MIS Quarterly 28 (1), 2004, S. 75-105.
- [Hu06] Hufenbach, W.: Textilverstärkte Verbundkomponenten für funktionsintegrierende Mischbauweisen bei komplexen Leichtbauanwendungen, Sonderforschungsbereich 639 der DFG an der TU Dresden, 10. Dresdner Leichtbausymposium, Dresden, 2006.
- [Nu02] Nüttgens, M.: Rahmenkonzept zur Evaluierung von Modellierungswerkzeugen zum Geschäftsprozessmanagement. In (Gesellschaft für Informatik (GI) e.V.): Informationssystem-Architekturen, Wirtschaftsinformatik Rundbrief der GI Fachgruppe WI-MobIS, 9, 2002.
- [Ob09] Object Management Group: Unified Modeling Language: Superstructure, v2.2, 2009
- [Sa06] Saeki, M.: Configuration Management in a Method Engineering Context, In (Dubois, E.; Pohl, K., Hrsg.): Lecture Notes in Computer Science, 4001, 2006, S. 384-398.
- [Sc98] Schütte, R.: Grundsätze ordnungsmäßiger Referenzmodellierung: Konstruktion konfigurations- und anpassungsorientierter Modelle, Gabler, 1998.
- [St04] Sturm, T.: Mannschaftssport – UML-Modellierung in verteilten Teams. In: OBJEKT-Spektrum, April 2004
- [Th05] Thomas, O.: Das Modellverständnis in der Wirtschaftsinformatik: Historie, Literaturanalyse und Begriffsexplikation, In: Veröffentlichungen des Instituts für Wirtschaftsinformatik: Heft 184, Mai 2005
- [Th07] Thomas, O.: Version Management for Reference Models: Design and Implementation, In (Becker, J.; Delfmann, P., Hrsg.): Reference Modeling, 2007.
- [VH05] Verschuren, P.; Hartog, R.: Evaluation in Design-Oriented Research. In Quality & Quantity 39, 2005, S. 733-762.
- [WJG10] Weller, J.; Juhrisch, M.; Großmann K.: Information Modeling: The Need of Semi-Automatic Model Analysis and Transformation. In: Proceedings of the 16th Americas Conference on Information Systems, Peru, 2010.
- [Ze97] Andreas Zeller. Configuration Management with Version Sets: A Unified Software Versioning Model and its Applications. PhD thesis, Braunschweig Technical University, 1997.

Domänensemantik-orientierte Integration heterogener konzeptueller Modelle

Janina Fengel, Michael Rebstock

Fachbereich Wirtschaft
Hochschule Darmstadt
Haardtring 100
64295 Darmstadt
vorname.nachname@h-da.de

Abstract: In Integrationsprojekten ist eine der Herausforderungen das Vergleichen und Verknüpfen heterogener Modelle. Dies erfordert nicht nur einen strukturellen, sondern insbesondere einen semantischen Abgleich. In diesem Beitrag stellen wir unsere dafür entwickelte Methode vor. Grundlage ist ein Reengineering von Modellen zu webbasierte Ontologien. Für die dafür erforderliche Modelldekomposition wurde eine unterstützende Brückenontologie entwickelt, die hier vorgestellt wird. Zur Unterstützung der anschließenden semantischen Integration nutzen wir Ontology-Matching-Verfahren zur automatisierten Ermittlung von Entsprechungen zwischen Modellen und ihren Elementen bezüglich der zu ihrer Bezeichnung verwendeten DomänenSprache. Die prototypische Implementierung eines entsprechenden Systems zeigt die Machbarkeit unserer Methode.

1 Motivation

Eine der Kernaufgaben des Managements in Unternehmen ist die Gestaltung und laufende Anpassung des geschäftlichen Geschehens an aktuelle Erfordernisse und die Allokation der dafür erforderlichen Ressourcen. Grundlage der Gestaltung von Unternehmen und B2B-Kooperationen ist ihre umfassende, korrekte Beschreibung. Die Beschreibung geschäftlicher Abläufe und der dabei genutzten Informationen und Ressourcen erfolgt zumeist mit Hilfe von Modellen, insbesondere zur Darstellung der Unterstützung des geschäftlichen Geschehens durch IT. Die Modellierung von Organisationsstrukturen, Daten und Prozessen zur Beschreibung und Gestaltung betrieblichen Geschehens hat daher stark an Bedeutung gewonnen. Gegenwärtig sind verschiedene Modellierungssprachen in Verwendung. Zumeist sind sie fokussiert auf bestimmte Aspekte und daher anwendbar zur Erstellung unverbunden nutzbarer Modelle. Ansätze im Bereich der Unternehmensarchitektur thematisieren die Notwendigkeit der Erfassung von Zusammenhängen [La09; Fr04]. Architekturframeworks wie ARIS oder das Zachmann-Framework bieten einen Rahmen zur Integration von Modellen unterschiedlicher Sichten, wie beispielsweise für die Beschreibung der Beziehung zwischen Daten und Prozessen [SZ92; Sc96].

Allerdings wird dabei keine Möglichkeit geboten, Modelle derselben Sicht, ausgedrückt in verschiedenen Modellierungssprachen, miteinander und damit auch zu Modellen anderer Sichten in Bezug zu bringen. Indes ist zur Herstellung durchgängiger Prozessabwicklung eine konsistente Verknüpfung und Verwaltung von Modellen unabdingbarer Bestandteil. Eine der dabei anfallenden Aufgaben ist der Abgleich und die Zusammenführung bereits existierender, in Nutzung befindlicher Modelle. Dieser Bedarf entsteht in Geschäftsintegrationsprojekten wie im Falle innerbetrieblicher Geschäftsprozessoptimierungen oder Business Reengineering und in unternehmensübergreifenden Zusammenhängen bei Fusionen, der Umsetzung von Supply-Chain-Management oder der Gestaltung von B2B-Kooperationen. Für eine umfassende Geschäftsanalyse sind dabei in jedem dieser Fälle konzeptuelle Modelle in Bezug zu setzen, sowohl im Sinne einer vertikalen Integration von Modellen verschiedener Sichten als auch horizontaler Integration von modellsprachlich unterschiedlichen Modellen der gleichen Sicht. Zur Zusammenführung sind diese zu vergleichen bezüglich der Inhaltsbedeutung ihrer Elemente und ihrer Struktur, wobei erst der Abgleich der DomänenSprache anschließend strukturelle Vergleiche erlaubt [SM07].

Das Vergleichen und Verknüpfen heterogener Modelle ist indes eine nicht-triviale Aufgabe. Prinzipiell explizieren Modelle aller Art unterschiedliche Sichten auf eine Organisation und damit Wissen. Allerdings unterscheiden sich häufig selbst Modelle gleichen Typs semantisch [BP08]. Dabei tritt semantische Heterogenität nicht nur im Bereich der Modellierungssprachen auf, sondern typischerweise bei der Auswahl der natürlichsprachlichen Fachbegriffe, die zur Benennung der Modellelemente verwendet werden [TF07]. Besondere die frei wählbare Fachterminologie behindert eine automatisierte Unterstützung semantischer Abgleiche und damit der zugrunde liegenden Daten und Prozesse. Die Modellelementbezeichnungen spiegeln die unternehmensspezifische Geschäftssprache wider. Existiert kein verbindlich definiertes Vokabular oder Regeln bezüglich deren Anwendung, können sich Modelle darin erheblich unterscheiden. Liegen gar Namenskonflikte vor, sind Modelle weder manuell noch automatisiert direkt vergleich- und damit integrierbar [BRS96; TF06]. Daher gilt es, semantische Ungleichheiten aufzulösen, um die Aussagen von Modellen inhaltlich in Bezug bringen und abgleichen zu können, besonders bei unterschiedlicher Herkunft der Modelle. Dabei handelt es sich um eine verknüpfende Modellintegration, die Modelle verbindet, ohne sie zu verschmelzen.

Zur Reduktion dieser Arbeitslast wird hier die Anwendung von Semantic-Web-Technologien vorgeschlagen und beschrieben. Es folgt die Vorstellung der Vorgehensweise zur Ontologisierung von Modellen und der dazu benötigten Brückenontologie UMCO in Abschnitt zwei. In Abschnitt drei wird die Nutzung von Ontology-Matching-Verfahren geschildert, sodass die Anwendung unserer Methode in Abschnitt vier gezeigt werden kann. Der Beitrag endet mit der Vorstellung und der Verbindung zu verwandten Arbeiten und einer kurzen Diskussion mit Ausblick auf zukünftige Arbeiten.

2 Ontologisierung von Modellen

Die Allgegenwärtigkeit des World-Wide-Web als globale Infrastruktur hat zur hohen Akzeptanz webbasierter Unterstützung elektronischer Geschäftsabwicklung beigetragen. Die Entwicklung der Idee des Semantic-Web und seiner spezifischen Technologien bietet nun weiterführend die Möglichkeit der Nutzung webbasierter Ontologien als Mittel zur Wissensstrukturierung und Herstellung semantischer Interoperabilität. Das Prinzip der Annotation von Information mit Metadaten erlaubt die Repräsentation von Wissen in strukturierter, maschinenzugänglicher Form basierend auf Internettechnologien, lesbar sowohl für Maschinen als auch von Menschen, nutzbar zum weiteren Ausbau der Informationsverarbeitung [SBH06].

Da die Entwicklung einer Ontologie ein komplexes Unterfangen sein kann, schlagen wir zur Unterstützung die Nutzung semantischer Technologien vor, mittels derer sich die Inhaltsbedeutung von Information formal erfassen lässt zwecks Herstellung von Maschinenverarbeitbarkeit. Durch Erschließung und Wiederverwendung des in Modellen aller Art enthaltenen Domänenwissens lassen sich Ontologien der Modelle ableiten. Diesen Vorgang der Ermittlung und semantischen Formalisierung nennen wir Ontologisierung. Anschließend können diese Ontologien mit automatisierter Unterstützung semantisch integriert und dadurch verknüpft werden. Über die Zeit kann so eine zusammenhängende, terminologische Domänen- oder auch spezifische Unternehmensontologie aus bereits vorliegenden Modellen automatisiert unterstützt abgeleitet werden. Durch die dabei bottom-up erfolgende Erfassung der verwendeten Domänterminologie kann diese wiederum zur Auflösung semantischer Ambiguität in Modellen eingesetzt werden.

2.1 Dekomposition anhand der Sprachräume

Eine Wiederverwendung von Modellen und Konvertierung in Ontologien erlaubt eine Verwertung existierender Modelle, während sie weiterhin unverändert aktiver Nutzung zur Verfügung stehen. Ein Reengineering dieser nicht-ontologischen Ressourcen bietet die Möglichkeit die Bedeutung der Modellaussage semantisch zu formalisieren. Dabei wird genutzt, dass bei der Erstellung von Modellen sowohl Kenntnis der Modellierungssprache als auch Kenntnis der Fachsprache erforderlich ist. Dies wurde bereits für die Modellierung von EPK gezeigt [TF06], lässt sich aber auf Modelle generell übertragen. Dabei enthalten Modelle Fakten aus zwei Wissensbereichen, wie in Abbildung 1 anhand eines Ausschnitts einer EPK namens „Travel Reservation“ aus der Domäne Touristik gezeigt.

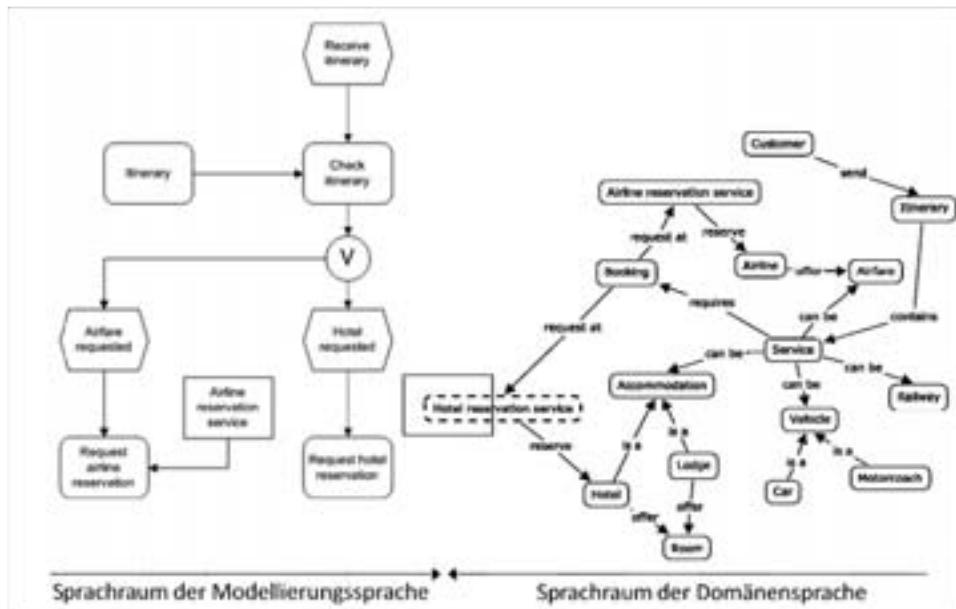


Abbildung 1: Sprachräume von Modellen

Aus dem Sprachraum der Domänen sprache werden Konzepte zur Benennung von Modellen und ihren Elementen genutzt, während die Konzepte der Modellierungssprache zur Beschreibung im Sinne einer Typisierung und Anordnung dieser Konzepte herangezogen werden. In Umkehrung dieses Vorgangs lassen sich Modelle zerlegen, um die jeweils verwendeten Konzepte der Sprachräume zu extrahieren und in Form semantischer Modelle zu erfassen [FR09]. Modelle können so in eine Ontologie mit Modellname und Modelltyp und eine mit Modellementnamen und Modellementtypen zerlegt werden.

2.2 Konvertierung in Ontologien

Modelle repräsentieren in der Regel abgestimmtes Fachwissen. Durch Dekomposition und Überführung in Ontologien wird Maschinenzugang zu diesem Wissen hergestellt. Ontologien im informatiktechnischen Sinne sind Artefakte und können als konzeptuelle Schemata verstanden werden, ähnlich Datenbankschemata [AF05]. Die Elemente einer Ontologie sind Klassen oder Konzepte, die eine Menge individueller Objekte repräsentieren, mit ihren Instanzen, die diese individuellen Objekte repräsentieren sowie Relationen und Axiome zur Beschreibung der Beziehungen der enthaltenen Konzepte. Im Prinzip sind Ontologien Sammlungen von Definitionen von Elementen und ihren Beziehungen und enthalten ein abgestimmtes Vokabular [DOS03]. Bei der Entwicklung wird in der Regel Konsens hergestellt und eine eindeutige Inhaltsbedeutung der enthaltenen Begriffe definiert [SSS01].

Im Zuge einer Konvertierung von Modellen in Ontologien lässt sich der bei der Modellerstellung bereits erreichte Konsens wieder ableiten und für jedes Modell in Form einer spezifischen Ontologie des Modells formalisieren.

Eine Ontologie bietet die Möglichkeit Wissen eindeutig darzustellen und damit zu teilen und wiederzuverwenden, denn sie beschreibt formal eine vereinfachte abstrahierte Sicht auf einen bestimmten Weltausschnitt [Gr93]. Dabei können Ontologien unterschiedlich stark formalisiert sein. Zur Repräsentation webbasierter Ontologien gibt es verschiedene Sprachen. Für einfache, so genannte leichtgewichtige, schwach formalisierte Ontologien bietet sich die XML-basierte Beschreibungssprache RDF-Schema an. Für stärker formalisierte, schwergewichtigere Ontologien eignet sich die Web Ontology Language OWL. Diese vom World Wide Web Consortium publizierte formale Ontologiebeschreibungssprache kann in drei verschiedenen Varianten genutzt werden [SH05]. Eine ausreichend formale Beschreibung ermöglicht automatisierte Weiterverarbeitung, nicht nur durch Ontology-Matching, wie nachfolgend beschrieben, sondern auch durch logisches Schließen [BP06]. Daher wird hier die beschreibungslogikbasierte Sprachvariante OWL DL genutzt, da diese entscheidbar ist [SWM04]. Konzepte, Relationen und Instanzen können dadurch unabhängig voneinander beschrieben werden [Br04]. Bei Verwendung von OWL DL ist es weiterhin möglich, Inferenzregeln basierend auf den in der Ontologie enthaltenen Relationen zu formulieren [SK06]. Somit kann durch logisches Schließen aus vorhandenem Wissen neues abgeleitet [AH08] und eine durch Modellkonvertierung erstellte Modellontologie weiterentwickelt werden.

Eine Ontologie in OWL DL beinhaltet zwei Bereiche. Die TBox, die terminological box, enthält die Definitionen der Domänenkonzepte mit ihren Eigenschaften und Beziehungen. Die ABox, die assertional box, enthält die Fakten und Assertionen bezüglich der beschreibenden Eigenschaften von Konzepten und Instanzen und ihrer Beziehungen. Abbildung 2 zeigt den Bezug zwischen konzeptuellen Modellen und Ontologien in OWL DL über die verschiedenen Modellierungsebenen.

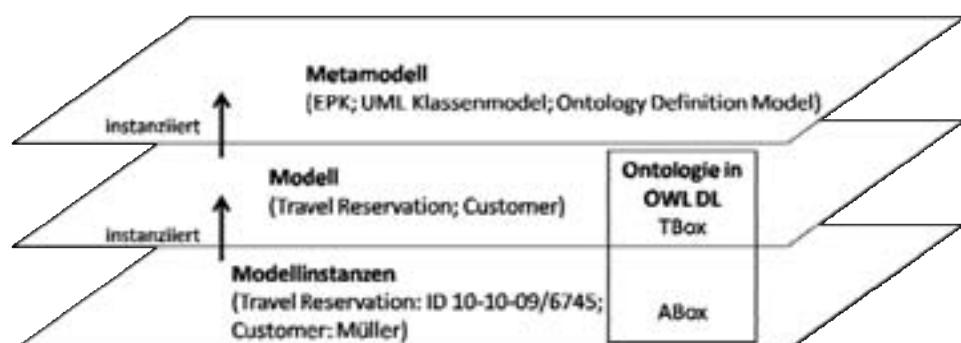


Abbildung 2: Modellebenen konzeptueller Modelle und Ontologien

Bei der Dekomposition von Modellen werden die Modellelemente in zwei Ontologien in OWL DL konvertiert, die Modellontologie und die Modelltypontologie. Zusammen beschreiben sie das Modell mit seinem Namen und Modelltyp sowie die Modellelemente mit ihren Namen und ihrem Modellemententyp. Dabei werden sowohl die Benennungen als auch die Typen als Konzepte in die jeweilige Ontologie aufgenommen. Für den oben gezeigten Ausschnitt der EPK „Travel Reservation“ entstehen aus dem Ereignis „Receive itinerary“ zwei Ontologiekonzepte, nämlich das Konzept „Receive itinerary“ und das Konzept „Ereignis“. Auf diese Weise kann in die entstehende Modellontologie, die die Konzepte der DomänenSprache aufnimmt, auch zukünftig anfallende Instanzinformation aufgenommen werden. Im genannten Beispiel könnten dies Kundendaten oder Informationen aus einem Workflowmanagementsystem sein. Bei der Konvertierung werden alle Modellnamen und Modellementbezeichnungen ohne weitere Verarbeitung transferiert. Auf diese Weise können vollständige Ausdrücke zur Weiterverarbeitung übernommen werden, denn das Fachwissen bei der Modellierung zeigt sich oft erst in der Kombination von Worten zu häufig genutzten Formulierungen.

2.3 Entwicklung der Brückenontologie UMCO

Zur Abbildung der domänenspezifischen Aussage von Modellen werden diese wie oben beschrieben zerlegt. Dadurch werden die Konzepte der DomänenSprache getrennt von den Konstrukten der Modellierungssprache, die sie typisieren. Die Verbindung einer solchen Typinformation mit dem jeweiligen DomänenSprachenkonzept bleibt allerdings erhalten. Zur Darstellung dessen wird hier die Idee der Indexierung in Form von Topic Maps aus dem Bibliothekswesen übernommen. Das Domänenkonzept wird über die Beziehung „occurs as“ mit dem Typkonzept verbunden. Abbildung 3 zeigt das Metamodell eines dekomponierten Modells.

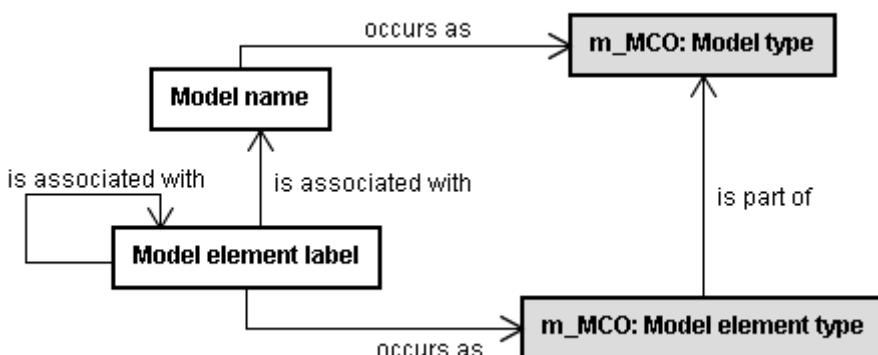


Abbildung 3: Metamodell eines dekomponierten Modells

Die Modellontologie, links im Bild in weiß gezeigt, umfasst das Domänenwissen ausgedrückt in den Elementbezeichnungen in natürlicher Sprache als Ontologiekonzepte und die bestehenden Beziehungen als Relationen, die je nach Bedarf mit Restriktionen versehen sind. Die Modellontologie importiert die Modelltypontologie, wie im Bild rechts in grau dargestellt. Für verschiedene Modellierungssprachen wurde von uns jeweils eine spezifische Modeling Concepts Ontology (MCO) entwickelt. Aktuell liegen MCOs vor für EPK, UML-Aktivitätsmodelle, BPMN, ERM, UML-Klassenmodelle und UML-Use-Case-Modelle, wobei für weitere beliebige Modellierungssprachen entsprechende MCOs entwickelbar sind. Jede MCO enthält als Konzepte die Modellierungssprachkonstrukte und damit Teile des jeweiligen Metamodells, die über Teil-Ganzes-Beziehungen verknüpft sind. Das Fachwissen, das in der Formulierung der logischen Beziehungen zwischen den Modellelementen im Modell enthalten ist, wird zusammen mit den Domänenkonzepten in die Modellontologie überführt. Bei dem gezeigten Beispiel wird entsprechend das Ereignis „Receive itinerary“ als Klasse der Modellontologie über die Relation „occurs as“ mit der Klasse „Event“ der EPC_MCO verbunden.

Zur Verknüpfung der entstehenden Modellontologien können über die MCO Referenzen zwischen Modellen desselben Typs in derselben Modellierungssprache direkt hergestellt werden. Um nun auch Verknüpfungen mit Modellen anderen Typs oder anderer Art zu ermöglichen, wurde die Unifying Modeling Concepts Ontology (UMCO) entwickelt. Sie verknüpft die verschiedenen MCOs und überbrückt so verschiedene Sprachräume, denn sie enthält Konzepte, die Modellierungskonzepte verschiedener Modellierungssprachen mit ähnlicher Intention zusammenfassen. Beispielsweise enthalten Geschäftsprozessmodellierungssprachen das Konzept von Aktivitäten, genannt Aktivität, Funktion, Aufgabe oder Aktion, begonnen und beendet durch Ereignisse und verbunden über Kontrollflüsse, beschrieben mit Hilfe logischer Konnektoren [MK07]. Im Detail ist die Semantik dieser Modellierungssprachen nicht identisch, und Modelle können nicht ohne weitere Bearbeitung verlustfrei von einer Sprache in eine andere transformiert werden [KQC07]. Allerdings sind die grundlegenden Bedeutungen der enthaltenen Sprachkonstrukte vergleichbar. Diese Beobachtung gilt ebenso für die Sprachen zur Modellierung statischer Modelle. Konzeptuelle Datenmodelle können als UML Klassenmodelle, Entity-Relationship-Modelle oder direkt als semantische Modelle in OWL repräsentiert werden. Dabei entsprechen die Bedeutungen einer Klasse bzw. Entität den Klassen einer Ontologie, während die Attribute und Beziehungen den Relationen bzw. Properties in den meisten Ontologiesprachen entsprechen [UG04]. Daher sind UML Klassenmodelle auch direkt zur Ontologiemodellierung nutzbar [GDD04; BVE04].

Zum Zweck der semantischen Integration wurden nun auf diesen Überlegungen basierend in der UMCO generalisierende Modellierungskonzepte definiert und zu den entsprechenden Modellierungskonzepten der MCOs gleichgesetzt. Dies bietet die Möglichkeit logischen Schließens über verknüpfte Modelle hinweg. Abbildung 4 zeigt das Metamodell der UMCO in Erweiterung des in Abbildung 3 vorgestellten Metamodells.

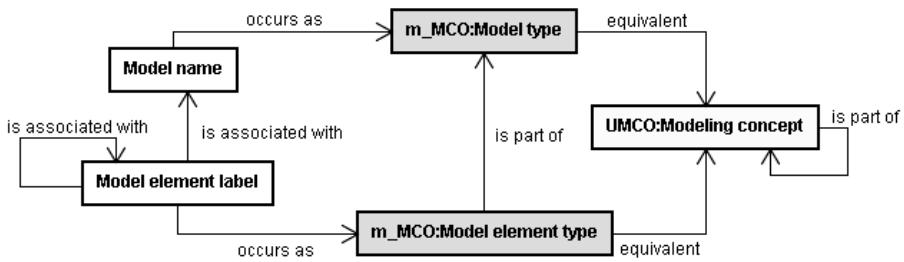


Abbildung 4: Metamodell eines dekomponierten Modells mit Beziehungen zur UMCO

Die UMCO enthält wie die MCOs neben den über Teil-Ganze-Beziehungen verbundenen vereinigenden Modellierungskonzepte keine weiteren definierenden Beziehungen. Das Konzept der UMCO ist generisch und beliebig erweiterbar mit zusätzlichen MCOs. Die Entwicklung von MCOs für Prozess-, Daten-, Organisations- oder auch technischen Modellen erlaubt die Verknüpfung von Domänenwissen nicht nur innerhalb des Sprachraums einer einzelnen Modellierungssprache, sondern sprachraumübergreifend mittels der Konzepte der UMCO auch mit Domänenwissen ausgedrückt in anderen Modellierungssprachen. Beispielsweise sind die Konzepte einer EPK mit dem eines UML Aktivitätsmodells und eines Modells in BPMN gleichgesetzt zum Konzept UMCO:Process. EPC_MCO:Function in der EPK, UML_AM_MCO:Action im UML Aktivitätsmodell und BPMN:Task im BPMN-Modell entsprechen dem Konzept UMCO:Activity in der UMCO. Auf dieselbe Weise lassen sich ebenso Konzepte einer Modellontologie zu Konzepten in Modellontologien entstanden aus anderen Modellarten in Bezug bringen, z.B. lassen sich in Prozessmodellen enthaltene Konstrukte zur Beschreibung von Ressourcen wie Dokumente oder Teilnehmer zu den sie beschreibenden Daten- oder Organisationsmodellen in Bezug bringen. So erfüllt die UMCO eine Brückenfunktion zur Verbindung der Modellontologien anhand der Typinformation der eigentlichen Domänenkonzepte.

3 Semantische Modellintegration

Um nun die entstandenen Modellontologien, die die Domänensemantik der konvertierten Modelle enthalten, automatisiert miteinander in Bezug zu bringen, können Ontology-Matching-Verfahren angewendet werden. Für ansonsten manuell auszuführende Modellabgleiche kann so automatisierte Unterstützung geboten werden und die Modelllemente, die die Domänensemantik widerspiegeln, können so unabhängig von der ursprünglich genutzten Modellierungssprache verglichen werden.

3.1 Anwendung von Ontology-Matching-Verfahren

Ontology-Matching sind Verfahren zur Ermittlung semantischer Entsprechungen zwischen Elementen zweier Ontologien. Es werden Paare von Entitäten gesucht, die vermutlich dieselbe Inhaltsbedeutung haben [EMS08]. Die Ontologien werden dabei abgeglichen, um semantische Korrespondenzen zu entdecken. Diese können als semantische Mappings verstanden werden. Es sind Aussagen der Form “A in Ontologie X entspricht B in Ontologie Y”. Die gefundenen Korrespondenzen beschreiben Äquivalenz oder Entsprechungen. Für weiterführende Nutzung können sie persistiert werden [ES07]. Dadurch können die verknüpften Ontologien bestehen bleiben, ohne zusammengeführt werden zu müssen. Dies ist besonders im Hinblick darauf nützlich, dass die zugrunde liegenden Modelle nicht ohne weiteres geändert werden können, sondern aktiv genutzt werden. Bewahrte Korrespondenzen bieten stattdessen die Möglichkeit einer virtuellen semantischen Integration.

Die Verfahren zum Ontology-Matching lassen sich in element- und strukturprüfende unterscheiden [ES07]. Elementprüfende Verfahren analysieren die Elemente einer Ontologie ohne Betrachtung der Beziehungen, die sie miteinander verknüpfen. Es werden Namen bzw. Zeichenketten und Mengen davon verglichen, wobei linguistische Ressourcen als Hintergrundinformation eingebunden werden können. Editierdistanzbasierte Methoden wie die Levenshtein-Distanz messen die Ähnlichkeit zweier Zeichenketten ausgedrückt als Anzahl der benötigten Änderungsoperationen zur Umwandlung des einen zum anderen Ausdruck. Berechnungsverfahren basierend auf der Jaro-Metrik messen die Anzahl und Anordnung gleicher Zeichen zweier Zeichenketten. Tokenbasierte Methoden führen diese Art von Messungen für Mengen von Zeichenketten aus [CRF03]. Strukturprüfende Verfahren dagegen analysieren die Struktur einer Ontologie und betrachten die Beziehungen der Elemente. In der Regel wird dazu das Vorliegen hierarchischer Beziehungen vorausgesetzt, da solche zumeist das Rückgrat einer Ontologie bilden [ES07].

Erste empirische Evaluationen von Matchings konvertierter Testmodelle wurden durchgeführt, wobei die Qualität der von verschiedenen Matchern automatisiert ermittelten Mappings durch Vergleich mit einem von Domänenexperten erstellten Referenzmapping gemessen wurde. Dabei zeigte sich, dass die Konvertierung von Modellen in Modellontologien wie beschrieben für Prozessmodelle zu Ontologien führt, die keine hierarchischen oder mereologischen Beziehungen enthalten, sodass nur elementprüfende Verfahren Ergebnisse liefern können. Weiterhin zeigte sich, dass die in Prozessmodellen übliche Benennung von Elementen mit mehreren Termen in einem Ausdruck durch Name-Matching-Verfahren allein zu qualitativ minderwertige Ergebnissen führt. Die Zerlegung solcher Multiterm-Ausdrücke und daran anschließendes tokenbasiertes Name-Matching führt dagegen zu höherwertigen Resultaten. Weitere Qualitätssteigerung ist durch zusätzliche Nutzung von WordNet als Hintergrundressource zu erreichen [Mi09]. Es bleibt zu eruieren, wie spezielle Glossare oder Lexika mit hinzugezogen werden können.

Die Konvertierung von Klassen- oder ER-Modellen bringt in den meisten Fällen Ontologien hervor, die auch mit strukturprüfenden Verfahren untersucht werden können. Unsere empirischen Evaluationen haben gezeigt, dass hier auf der Jaro-Winkler-Metrik basierende Vergleiche in Kombination mit Strukturvergleichen Ergebnisse erbringen, da Datenmodelle in der Regel Elementbezeichnungen bestehend aus ein bis zwei Termen enthalten.

3.2 Ermittlung von Korrespondenzen anhand der Domänensemantik

Da das Domänenwissen nicht als Instanz der jeweiligen Modellierungssprachkonstrukte konvertiert wird, sondern als Ontologieklassen, wird verhindert, dass Matching Entsprechungen der Modellierungssprachkonstrukte untereinander liefert. Diese sind dank der Nutzung der MCOs und der UMCO direkt verfügbar, ohne Aufwand für Matcher zu produzieren. Weiterhin werden durch das Nichtvorhandensein einer solchen unerwünschten Struktur die Modelle ausschließlich anhand der Domänsprache abgeglichen. Abbildung 5 zeigt beispielhaft die Verknüpfung von Modellen in Form einer Ontologie, dargestellt als semantisches Netz. Es handelt sich um den bereits bekannten Ausschnitt aus der EPK „Travel Reservation“ mit einem Ausschnitt aus einem UML Aktivitätsmodells namens „Travel Booking“, die abgeglichen und miteinander in Bezug gebracht wurden.

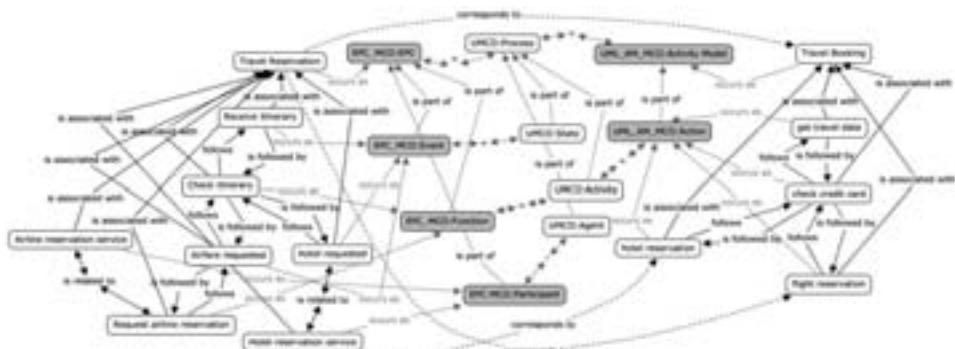


Abbildung 5: Semantisches Netz von Ausschnitten verknüpfter heterogener Modelle

Die konvertierten Modelle entstammen derselben Anwendungsdomäne, nutzen aber unterschiedliche Terminologie und Modellierungssprachen. Die Modellontologien, dargestellt in weiß, bilden die Domänsprache ab. Die Elemente referenzieren ihren Modellelementtyp, enthalten in der in grau dargestellten MCO. Die MCOs sind über die UMCO, hier in hell, verbunden durch die Definition der Entsprechungen der Elemente. Die durch Ontology-Matching ermittelten semantischen Korrespondenzen, im Bild durch gestrichelte Beziehungen dargestellt, verknüpfen die Modellontologien untereinander und damit die Konzepte der Domäne.

Matchings können nach Bedarf schrittweise vorgenommen werden zwischen ontologisierten Modellen, unabhängig welchen Typs und in welcher Sprache. Über die Zeit wird mit zunehmender Zahl an Modellen die Tendenz zu häufig genutzten Begriffen sichtbar. Langfristig kann so eine erste einfache Unternehmensontologie aus den verknüpften Modellen abgeleitet werden. Diese kann genutzt werden, um Suchen nach Korrespondenz auszuführen. Es können Vergleiche zwischen Modellen und Abgleiche mit Referenzmodellen, Standards oder bereits vorliegenden Unternehmensglossaren vorgenommen werden. Eine generelle Suche über alle verknüpften Modelle nach Okkurenz liefert Information zur Häufigkeit des Vorkommens eines Begriffs sowie dem Zusammenhang und Auftreten in bestimmten Modellen.

3.3 Vernetzung von Modellen

In Realisierung unserer Methode wird unser MODI-System entwickelt. Es ist in Java implementiert und über ein Web-Service-Interface nutzbar. Es besteht aus einem Kernsystem, an das über Adapter beliebige Ontology-Matching-Tools angekoppelt werden können. Zusätzlich ist ein Web-Service verfügbar, mit dem Modellkonvertierungen durchgeführt werden können. Viele Modellierungswerzeuge bieten die Möglichkeit Modelle in XMI-Format zu exportieren, sodass eine Konvertierung mit Hilfe von XSLT vorgenommen werden kann. Allerdings hat sich gezeigt, dass aufgrund unterschiedlicher Verwendungen von XMI für jeden Modelltyp abhängig vom Werkzeug ein spezifisches XSLT-Sheet erforderlich ist.

Der Fokus der Entwicklungsarbeiten liegt auf der Realisierung von Matching und Mapping. Die Anwendung von Ontology-Matching-Verfahren liefert automatisiert eine intiale Menge semantischer Korrespondenzen. Diese dient als Ausgangsbasis für nutzergetriebene Weiterentwicklung. Nutzer können neue Korrespondenzen und neue Modelle hinzufügen. Unser System bietet verschiedene Abfragemöglichkeiten. Ein Nutzer kann gezielt zwei ausgewählte Modelle vergleichen oder generell nach vorhandenen Korrespondenzen zu einem bestimmten Begriff suchen. Das System liefert eine Liste mit Vorschlägen möglicher Entsprechungen, aus der der Nutzer auswählen kann. Obwohl die Anwendung von Matching-Werkzeugen die manuell zu erbringende Arbeit verringert, sind die Ergebnisse nicht perfekt, sondern können mehrdeutig oder inkorrekt sein. Zur Disambiguierung oder Korrektur ist menschliches Eingreifen erforderlich [Zh04]. Daher bietet unser System aktive Nutzerpartizipation und ermöglicht das Hinzufügen und Editieren von Korrespondenzen. Abbildung 6 zeigt die Vorschlagsliste für den Vergleich der beiden Beispielmodelle in textueller Form, der als Ausschnitt in Abbildung 5 zu sehen ist.

The screenshot shows a user interface for a semantic matching tool. At the top, there are tabs for 'Importing Models', 'New Correspondence', 'View Model Ontology', and 'Merge Models'. Below this is a search bar with dropdown menus for 'Source Model' and 'Destination Model', both set to 'Travel: Reservation'. There are also checkboxes for 'Show Element Type' and 'Show MCO Type'. A table below lists the results:

Source Expression	Source Model	Destination Expression	Destination Model	Confidence	Comment	Facts
Travel: Reservation	Travel: Reservation	Travel: Booking	Travel: Booking	99%	Travel	Travel: Reservation: Booking
Wally: consumer	Travel: Reservation	Wally: consumer	Travel: Booking	99%		Wally: consumer: Wally
Travel: Reservation	Travel: Reservation	Right: reservation	Travel: Booking	99%		Travel: Reservation: Right
Detail: credit: card	Travel: Reservation	Detail: credit: card	Travel: Booking	99%		Detail: credit: card: Detail
Detail: credit: card	Travel: Reservation	credit: card: model	Travel: Booking	73%		Detail: credit: card: invalid
Detail: reservation: name	Travel: Reservation	Detail: reservation	Travel: Booking	73%		Detail: reservation: invalid
Request: hotel: reservation	Travel: Reservation	hotel: reservation	Travel: Booking	99%		Request: hotel: reservation

At the bottom left, it says 'Found 7 results' and 'Time: 0.000000'. On the right, there is a link 'View all flagged in as invalid_models'.

Abbildung 6: Vorschlagsliste semantischer Korrespondenzen eines Modellvergleichs

Die Kombination der Modellontologien mit ihren MCOs bietet weiterhin die Möglichkeit der Filterung anhand der Typinformation und damit der Suche nach oder in bestimmten Modelltypen, beispielweise nach allen EPKs, oder nach allen Modelltypen einer Art anhand des entsprechenden Konzepts in der UMCO, beispielsweise UMCO:Process. Umgekehrt liefert die Suche nach allen UMCO:Activity eine Liste aller Geschäftsaktivitäten als Ergebnis. So lässt sich sowohl modellsemantik- als auch domänensemantik-orientiert die entstehende Unternehmensontologie abfragen.

4 Verwandte Arbeiten

In der Praxis liegen häufig unterschiedliche unverbundene Modelle vor. Einige Arbeiten zur Modellintegration konzentrieren sich auf die Modellierungssprachen und Möglichkeiten der Migration oder Integration basierend auf der Übertragung der Modelle von einer Modellierungssprache in eine andere [Ge07; MK07; KQC07], dem Matching von Modellen über ihre Metamodelle [Ka05] oder dem Management von Modellen gleichen Typs [MRB03]. Dabei wird der Aspekt heterogen verwendeter DomänenSprache nicht betrachtet. Die Erweiterung von Geschäftsprozessmodellen mit semantischen Annotationen wurde vorgeschlagen. Dabei werden die Domänenkonzepte zu Instanzen in ihrer Modellsprachenontologie [BCD07; TF09]. Andere Vorschläge zur Integration von Prozessmodellen setzen das Vorliegen eines separat erstellten Domänenmodells oder einer Domänenontologie voraus [BEK06; We07; EKO07; Hö07]. Die Nutzung einer manuell erstellten Domänenontologie bei der Prozessmodellierung wird in [HLD05; TF07] vorgeschlagen und für die Softwaremodellierung in [SK06]. Das Problem des damit verbundenen zeit- und kostenintensiven Erstellungsaufwands sowie die Problematik der korrekten Erhebung wird im Software Engineering bei der Erstellung von Domänenmodellen und formalen domänenspezifischen Programmiersprachen [Am08; Hi08] als auch im Knowledge Engineering für die Konstruktion von Ontologien betrachtet [AMS08]. Bestehende Vorschläge zur automatisierten Domänenmodellerstellung konzentrieren sich bisher auf die Wiederverwendung von UML-Klassenmodellen [GGD04].

Aktuell liegen keine Ansätze vor, die eine Lösung der Frage der semantischen Integration von Modellen unterschiedlicher Art und verschiedenen Typs sowohl für den Bereich der Modellierungssprache als auch für den Bereich der DomänenSprache zusammen vorschlagen. Hier kann unsere Methode als Unterstützung und Ergänzung der vorliegenden Ansätze dienen.

5 Schlussbetrachtung

Im vorliegenden Beitrag wurde eine Methode zur domänensemantik-orientierten Integration von Modellen mit Hilfe von Semantic-Web-Technologien vorgestellt und anhand von Beispielmodelle gezeigt. Eine Vorgehensweise zur Wiederverwendung von Modellen durch Konvertierung in Ontologien wurde beschrieben. Die entstehenden Modellontologien lassen sich modellierungssprachlich-orientiert mit Hilfe der dafür entwickelten Brückenontologie UMCO mit ihren erweiternden MCOs verknüpfen. Die Modellontologien, die die Domänensemantik der Modelle enthalten, lassen sich weiterführend ohne manuelle Vorarbeiten zur Erstellung einer Domänenontologie abgleichen und in Bezug setzen. Dafür werden Ontology-Matching-Verfahren eingesetzt. Die auf diese Weise ermittelten semantischen Korrespondenzen können gesammelt werden, um langfristig die Ausgangsbasis für die Entstehung einer einfachen terminologischen Unternehmensontologie zu bilden. Gegeben durch die Ontologisierung der Modelle sind keine Vorarbeiten zur Erstellung einer Gesamtontologie erforderlich.

Durch die Einbindung von Nutzern kann die Qualität der automatisiert ermittelten Korrespondenzen evaluiert und verbessert werden. Unser dafür entwickeltes MODI-System dient somit als Mediator und bietet Unterstützung zur Auflösung semantischer Heterogenität in Modellen. Das hier vorgestellte System wurde prototypisch implementiert und dient als Machbarkeitsnachweis der entwickelten Methode. Die entsprechende Evaluation ist noch zu Ende zu führen. Da das System die Kopplung externer Ontology-Matching-Werkzeug vorsieht, ist eine weiterführende Evaluation bezüglich deren Effizienz erforderlich, um geeignete Kombinationen zu ermitteln. Insgesamt hoffen wir, mit unserem Vorschlag die Nützlichkeit der Anwendung von Semantic-Web-Technologien zur Unterstützung in der Modellierung gezeigt zu haben, insbesondere, da die Brückenontologie beliebig erweiterbar ist mit weiteren MCOs, auch aus technischem oder anderen Anwendungsbereichen.

6 Danksagung

Diese Arbeiten wurden gefördert durch das BMBF im Rahmen des FHProfUnt-Programms im Projekt MODI, Förderkennzeichen 1728X07.

Literaturverzeichnis

- [AF05] Antoniou, G.; Franconi, E.; van Harmelen, F.: Introduction to Semantic Web Ontology Languages. In (Eisinger, N.; Maluszynski, J. Hrsg.): Reasoning Web. First International Summer School 2005, Msida, Malta, July 25-29, 2005, Revised Lectures. Springer, Berlin Heidelberg, 2005; S. 1–21.
- [AH08] Antoniou, G.; van Harmelen, F.: A semantic Web primer. MIT Press, Cambridge, Mass., 2008.
- [Am08] Ambler, S. W.: The object primer. Agile model-driven development with UML 2.0. Cambridge Univ. Press, Cambridge, 2008.
- [AMS08] d'Aquin, M. et al.: Toward a New Generation of Semantic Web Applications. In IEEE Intelligent Systems, 2008, 23; S. 20–28.
- [BCD07] Belecheanu, R. et al.: Business Process Ontology Framework, Super Project, 2007.
- [BEK06] Brockmans, S. et al.: Semantic Alignment of Business Processes. In (Manolopoulos, Y.; Filipe, J.; Constantopoulos, P.; Cordeiro, J. Hrsg.): Proc. of the Eighth Intern. Conf. on Enterprise Information Systems (ICEIS 2006). Information Systems Analysis and Specification. May 2006, Paphos, Cyprus. INSTICC, Setúbal, 2006; S. 197–203.
- [BP06] Blumauer, A.; Pellegrini, T.: Semantic Web und semantische Technologien: Zentrale Begriffe und Unterscheidungen. In (Pellegrini, T.; Blumauer, A. Hrsg.): Semantic Web Wege zur vernetzten Wissensgesellschaft. Springer, Berlin, 2006; S. 9–25.
- [BP08] Becker, J.; Pfeiffer, D.: Solving the Conflicts of Distributed Process Modelling – Towards an Integrated Approach. In (Golden, W.; Acton, T.; Conboy, K.; van der Heijden, H.; Tuunainen, V. K. Hrsg.): 16th European Conf. on Information Systems (ECIS 2008), 2008; S. 1555–1568.
- [Br04] Bruijn, J. de: Semantic Information Integration Inside and Across Organizational Boundaries, Deliverable. DERI, Innsbruck, 2004.
- [BRS96] Becker, J.; Rosemann, M.; Schütte, R.: Prozeßintegration zwischen Industrie- und Handelsunternehmen - eine inhaltlich-funktionale und methodische Analyse. In Wirtschaftsinformatik, 1996, 39; S. 309–316.
- [BVE04] Brockmans, S. et al.: Visual Modeling of OWL DL Ontologies Using UML. In (McIlraith, S. A. Hrsg.): The semantic Web - ISWC 2004. Third International Semantic Web Conference, Hiroshima, Japan, November 7 - 11, 2004. Springer, Berlin, 2004; S. 198–213.
- [CRF03] Cohen, W.; Ravikumar, P.; Fienberg, S.: A Comparison of String Distance Metrics for Name-Matching Tasks. In (Kambhampati, S.; Knoblock, C. A. Hrsg.): Proc. of IJCAI-03 Workshop on Information Integration on the Web (IIWeb-03), 2003; S. 73–78.
- [DOS03] Daconta, M. C.; Obrst L. J.; Smith K. T.: The Semantic Web. A Guide to the Future of XML, Web Services, and Knowledge Management. Wiley, Indianapolis, 2003.
- [EKO07] Ehrig, M.; Koschmider, A.; Oberweis, A.: Measuring Similarity between Business Process Models. In (Roddick, J.F.; Hinze, A. Hrsg.): Proc.of the Fourth Asia-

- Pacific Conf. on Conceptual Modelling (APCCM 2007), Australian Computer Science Communications, Ballarat, 2007; S. 71-80
- [EMS08] Euzenat, J.; Mocan, A.; Scharffe, F.: Ontology Alignments An ontology management perspective. In (Hepp, M.; Leenheer, P.; Moor, A.; Sure, Y. Hrsg.): Ontology Management. Semantic Web, Semantic Web Services, and Business Applications. Springer, Boston, MA, 2008; S. 177–206.
- [ES07] Euzenat, J.; Shvaiko, P.: Ontology Matching. Springer, Berlin, 2007.
- [FR09] Fengel, J.; Rebstock, M.: Model-Based Domain Ontology Engineering. In (Hepp, M.; Hinkelmann, K.; Stojanovic, N. Hrsg.): Proc. of the 4th Intern. Workshop on Semantic Business Process Management (SBPM2009). ACM Press, New York, 2009.
- [Fr04] Frank, U.: Informationstechnologie und Organisation. In (Schreyögg, G., Werder, A. Hrsg.) Handwörterbuch Unternehmensführung und Organisation HWO. 4. Aufl., Schäffer-Poeschel, Stuttgart, 2004; S. 472-480.
- [GDD04] Gašević, D. et al.: Converting UML to OWL ontologies: The thirteenth International World Wide Web Conference. Alternate track papers & posters. ACM Press, New York, 2004; S. 488–489.
- [Ge07] Gehlert, A.: Migration fachkonzeptueller Modelle. Logos-Verl., Berlin, 2007.
- [Gr93] Gruber, T.: A translation approach to portable ontology specifications. In Knowledge Acquisition, 1993, 5; S. 199–220.
- [Hi08] Hillairet, G.: Conception et utilisation d'ontologies pour la conception et l'exploitation des systèmes d'information, Deliverable. 2008.
- [HLD05] Hepp, M. et al.: Semantic Business Process Management: A Vision Towards Using Semantic Web Services for Business Process Management. In (Lau, F. C. M.; Lei, H.; Meng, X.; Wang, M. Hrsg.): Proc. of the IEEE Intern. Conf. on e-Business Engineering. ICEBE 2005, October 2005, Beijing, China. IEEE, Los Alamitos, Calif., 2005; S. 535–540.
- [Hö07] Höfferer, P.: Achieving Business Process Model Interoperability Using Metamodels and Ontologies. In Proc. of the 15th European Conference on Information Systems (ECIS 2007), 2007.
- [Ka05] Kappel, G. et al.: ModelCVS - A Semantic Infrastructure for Model-based Tool Integration, Technical Report, Wien, 2005.
- [KQC07] Kensche, D. et al.: GeRoMe: A Generic Role Based Metamodel for Model Management. In (Spaccapietra, S.; Atzeni, P.; Fages, F.; Hacid, M.-S.; Kifer, M.; Mylopoulos, J.; Pernici, B.; Shvaiko, P.; Trujillo, J.; Zaihrayeu, I. Hrsg.): Journal on Data Semantics VIII. Springer, Berlin, Heidelberg, 2007; S. 82–117.
- [La09] Lankhorst, M.: Enterprise Architecture at Work. Modelling, Communication and Analysis. Second Edition. Springer, Berlin, Heidelberg, 2009.
- [Mi09] Miller, G.A.: WordNet - About Us. WordNet. Princeton University. 2009. <http://wordnet.princeton.edu>, 12.5.2010.
- [MK07] Murzek, M.; Kramler, G.: The Model Morphing Approach – Horizontal Transformations between Business Process Models. In (Nummenmaa, J.;

- Söderström, E. Hrsg.): Proc. of the 6th Intern. Conf. on Perspectives in Business Information Research - BIR'2007, Tampere, Finland, 2007; S. 88–103.
- [MRB03] Melnik, S.; Rahm, E.; Bernstein, P. A.: Rondo: A Programming Platform for Generic Model Management. In (Halevy, A. Y. Hrsg.): Proc. of the ACM SIGMOD Intern. Conf. on Management of Data. 2003, San Diego. ACM Press, New York, 2003; S. 193–204.
- [SBH06] Shadbolt, N.; Berners-Lee, T.; Hall, W.: Semantic Web Revisited. In IEEE Intelligent Systems, 2006, 21; S. 96–101.
- [Sc96] Scheer, A.W.: ARIS – House of Business Engineering. Heft 133. Universität des Saarlandes, Institut für Wirtschaftsinformatik, Saarbrücken, 1996.
- [SH05] Stuckenschmidt, H.; Harmelen, F.: Information Sharing on the Semantic Web. Springer, Berlin Heidelberg, 2005.
- [SK06] Saeki, M.; Kaiya, H.: On Relationships among Models, Meta Models and Ontologies. In (Gray, J.; Tolvanen, J. P.; Sprinkle, J. Hrsg.): Proc. of the 6th OOPSLA Workshop on Domain-Specific Modeling, Jyväskylä, Finland, 2006.
- [SM07] Simon, C.; Mendling, J.: Integration of Conceptual Process Models by the Example of Event-driven Process Chains. In (Oberweis, A.; Weinhardt, C.; Gimpel, H. Hrsg.): eOrganisation. Service-, Prozess-, Market-Engineering. Bd. 1. 8. Intern. Tagung Wirtschaftsinformatik (WI 2007). Univ.-Verl, Karlsruhe, 2007; S. 677–694.
- [SWM04] Smith, M. K.; Welty, C.; McGuinness, D. L.: OWL Web Ontology Language Guide. <http://www.w3.org/TR/owl-guide/>, 27.12.2009.
- [SSS01] Staab, S. et al.: Knowledge Processes and Ontologies. In IEEE Intelligent Systems, 2001, 16; S. 26–34.
- [SZ92] Sowa, J.; Zachmann, J.: Extending and formalizing the framework for information systems architecture. In IBM Systems Journal, 1992, 31; 3, S. 590–616.
- [TF06] Thomas, O.; Fellmann, M.: Semantische Integration von Ontologien und Ereignisgesteuerten Prozessketten. In (Nüttgens, M.; Rump, F. J.; Mendling, J. Hrsg.): EPK 2006 - Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten. CEUR-WS.org, Wien, 2006; S. 7–23.
- [TF07] Thomas, O.; Fellmann, M.: Semantic Business Process Management: Ontology-Based Process Modeling Using Event-Driven Process Chains. In IBIS Interoperability in Business Information Systems, 2007, 2; S. 29–44.
- [TF09] Thomas, O.; Fellmann, M.: Semantische Prozessmodellierung – Konzeption und informationstechnische Unterstützung einer ontologiebasierten Repräsentation von Geschäftsprozessen. In Wirtschaftsinformatik, 2009, 51; S. 506–518.
- [UG04] Uschold, M.; Gruninger, M.: Ontologies and Semantics for Seamless Connectivity. In ACM SIGMOD Record, 2004, 33; S. 58–64.
- [We07] Weske, M.: Business Process Management. Concepts, Languages, Architectures. Springer, Berlin Heidelberg, 2007.
- [Zh04] Zhdanova, A. et al.: Ontology Alignment Solution v2.0., Deliverable D1.4, Esperonto project. 2004.

Vertikale Modellintegration in Rahmenwerken – Evaluation von Integrationsansätzen und Untersuchung der Implementierbarkeit anhand eines Fallbeispiels

Stephan Bögel, Werner Esswein

Technische Universität Dresden

Lehrstuhl für Wirtschaftsinformatik, insb. Systementwicklung

stephan.boegel@tu-dresden.de; werner.esswein@tu-dresden.de

Abstract: Die modellgestützte Simulation mobiler Baumaschinen wird in dem vom BMBF geförderten Projekt INPROVY untersucht. Zur Problemstrukturierung wurde das INPROVY-Rahmenwerk vorgestellt, dass das Simulationsproblem in unterschiedliche Modellebenen aufteilt. Die vertikale Integration der verschiedenen Modellebenen ist Gegenstand dieses Artikels. Es wird eine Evaluation nach der Methode des merkmalsbasierten Vergleichs der Integrationsansätze durchgeführt, die den Stand der Forschung repräsentieren. Die daraus gewonnen Erkenntnisse werden im Rahmen eines Fallbeispiels auf ihre Implementierbarkeit hin geprüft.

1 Einleitung und Motivation

In vielen technischen Disziplinen ist die Simulation von Maschinen oder Abläufen ein wichtiges Mittel um Erkenntnisse zu gewinnen und Kosten zu sparen. Häufig erfolgt die Simulationserstellung am Einzelfall orientiert und wenig systematisch. Um eine systematischere Vorgehensweise zu etablieren, wird die modellgestützte Erstellung von Simulationsmodellen mobiler Arbeitsmaschinen im Rahmen des vom BMBF geförderten Forschungsprojektes INPROVY¹ untersucht.

Da mobile Baumaschinen, wie Radlader oder Betonspritzpumpen, aus Komponenten unterschiedlicher Zulieferer bestehen, die in die Geschäftsprozesse der beteiligten Unternehmen integriert sind, ist die systematische Erstellung wiederverwendbarer Simulationsmodelle eine komplexe Aufgabe. Zur Komplexitätsreduktion wurde daher ein Rahmenwerk erstellt, welches das Problem in handhabbare Betrachtungseinheiten unterteilt [EGL09].

Das Rahmenwerk besteht u.a. aus den Ebenen [EGL09]:

- Fachebene: Beschreibt die Struktur der mobilen Baumaschine aus der Perspektive der Geschäftsprozesse. Hier findet sich beispielsweise das Konzept „Produktkom-

¹ „Integrative Produktentwicklung mit virtuellen Prototypen“ (INPROVY), Gefördert im Rahmenkonzept „Forschung für die Produktion von morgen“ <http://www.inprovye.de>

ponente“. Diese ist durch eine Artikelnummer charakterisiert, die in Bestell- oder Lagerprozessen genutzt wird.

- Simulationsebene: Betrachtet die problemorientierte Aufarbeitung der Maschinen-simulation. Hier findet sich beispielsweise das Konzept „Simulationskomponente“.

Um eine Änderung an einer Ebene an alle betroffenen Ebenen zu propagieren und Inkonsistenzen zwischen den Ebenen zu erkennen, müssen die Ebenen integriert werden.

Insbesondere die Abstimmung der Fach- und der Simulationsebene ist für die Simulation entscheidend. Nur wenn das Unternehmensmodell mit den Simulationsmodellen integriert wird können Informationen aus den Geschäftsprozessen zur Simulationserstellung genutzt und die Simulationsergebnisse schnell in die Geschäftsprozesse integriert werden. Hierbei handelt es sich um eine *vertikale* Integration von Rahmenwerksebenen. Eine durchgängige, modellgestützte Nutzung und Erstellung von Simulationsmodellen erfordert Lösungs-ansätze für die vertikale Integration der Ebenen.

Damit diese Ziele erreicht werden können, müssen die Abhängigkeiten zwischen den einzelnen Betrachtungsebenen beschrieben – modelliert – werden [ARW08]. Es stellt sich die Frage, *wie* diese Abhängigkeiten modelliert werden können.

Im Rahmen der vorliegenden Arbeit wird eine Evaluation von Modellintegrationsansätzen durchgeführt, die den Stand der Forschung repräsentieren. Neben der Bewertung dieser Ansätze werden wesentliche Gestaltungsdimensionen identifiziert, die notwendig sind, um einen Ansatz auf das Problem der vertikalen Integration in Rahmenwerken anzupassen.

Sowohl der Betrachtungsgegenstand der Evaluation als auch das Ziel der vorliegenden Arbeit legen eine konstruktivistische Perspektive nahe. Diese Forschung wird dem Design Science Ansatz zugeordnet [HMPR04]. Angelehnt an die Forschungsmethodik von [VH05] wird in Kapitel 2 das Problem theoriegeleitet untersucht. Anschließend erfolgt die Anforderungsanalyse, die Formulierung des Evaluationsrahmens sowie die Durchführung der Evaluation in Kapitel 3. Das Fallbeispiel wird in Kapitel 4 untersucht und in Kapitel 5 wird weiterer Forschungsbedarf aufgezeigt.

2 Theorie der vertikalen Modellintegration

Das Problem der vertikalen Modellintegration von Ebenen in Rahmenwerken wird im Folgenden anhand der Integrationsdimensionen Integrationsrichtung (vertikal vs. horizontal), Integrationsart (verbindend vs. vereinigend) und Integrationsansatz (Transformation vs. Zuordnung) eingegrenzt. Die Eingrenzung des Problems erfolgt durch Betrachtung verschiedener Merkmale des Integrationsbegriffs und den Ausschluss bestimmter Merkmalsausprägungen aus der Betrachtung. Eine Übersicht ist in Tabelle 1 gegeben.

Integrationsmerkmal	Merkmalsausprägungen			
	Verbindung		Vereinigung	
Integrationsart				
Integrationsrichtung	vertikal	horizontal	hybrid	
Integrationsrealisierung	Organisationsebene	Technikebene	Modellebene	
Integrationsgegenstand	Daten, Prozesse, Funktionen, Objekte, Modelle, Methoden			
Integrationsreichweite	Aufgabe innerbetrieblich	Individuum zwischenbetrieblich	Organisationseinheit überbetrieblich	Unternehmen
Automatisierbarkeit	automatisch	semi-automatisch	manuell	
Sprachdimension	Syntax	Semantik	Pragmatik	
Integrationsansatz	Transformation		Zuordnung	

Tabelle 1: Übersicht Integrationsmerkmale und Ausprägungen

2.1 Vertikale vs. Horizontale Modellintegration

Als mögliche *Integrationsrichtungen* werden die vertikale und die horizontale Modellintegration unterschieden [Ros02]. In Anlehnung an das generische Rahmenwerk von SINZ [Sin99] sind in Abbildung 1 Ausschnitte aus zwei exemplarischen Rahmenwerken dargestellt. Diese werden von zwei unterschiedlichen Unternehmen eingesetzt.

In Rahmenwerk A benutzen Unternehmen 1 und Unternehmen 2 den gleichen Architekturrahmen mit identischen Modellierungssprachen. In Rahmenwerk B verwenden die beiden Unternehmen unterschiedliche Modellierungssprachen. Dies resultiert in unterschiedlichen Metamodellen $MM_{k,x}$ innerhalb einer Ebene.

Weiterhin unterscheiden sich zwei Modelle $M_{i,1}$ und $M_{j,1}$, die auf zwei unterschiedlichen Ebenen liegen, im Betrachtungswinkel [Sin99]. Da die Modelle dasselbe Unternehmen beschreiben, beziehen sie sich auf denselben Betrachtungsgegenstand. Zwei Modelle $M_{i,1}$ und $M_{i,2}$ aus zwei unterschiedlichen Unternehmen oder Organisationseinheiten, die aber auf der gleichen Ebene liegen, unterscheiden sich dann im Betrachtungsgegenstand, nicht aber im Betrachtungswinkel.

Das Szenario einer horizontalen Integration tritt beispielsweise auf, wenn die beiden Unternehmen 1 und 2 fusionieren oder eine enge Kooperation anstreben (etwa mit dem Ziel

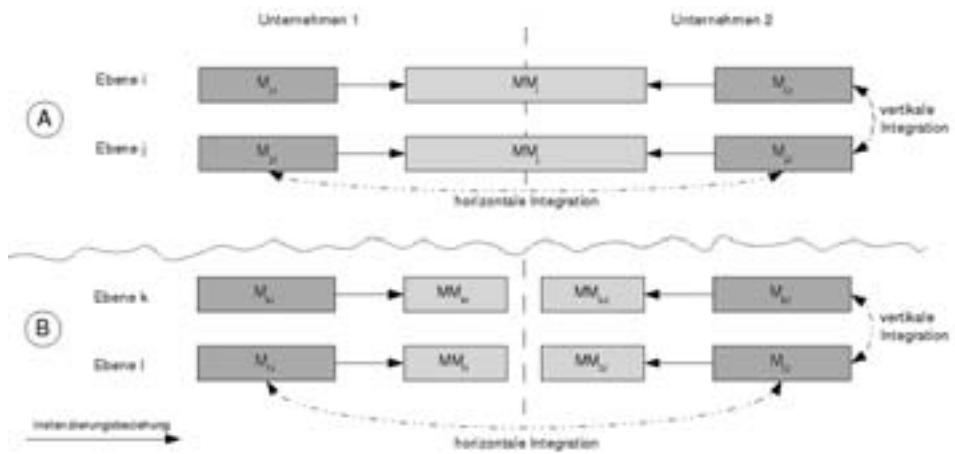


Abbildung 1: Vertikale vs. horizontale Modellintegration in Rahmenwerken

einer integrierten Versorgungskette). Eine solche Integration umfasst beispielsweise die Modelle $M_{i,1}$ und $M_{i,2}$ in Rahmenwerk A. Verwenden beide Unternehmen unterschiedliche Modellierungssprachen, wie in Rahmenwerk B angedeutet, so müssen diese bei der Integration berücksichtigt werden. Die Überwindung von Sprachgrenzen ist also bei der vertikalen Integration immer zu berücksichtigen, bei der horizontalen Integration nur im Fall B. Im Rahmen von INPROVY wird ein Rahmenwerk zur modellgestützten Simulation entwickelt, das die verwendeten Modellierungssprachen vorschreibt. Es liegt hier daher Fall A vor.

2.2 Verbindende vs. Vereinigende Modellintegration

Bezüglich der *Integrationsart* kann eine Integration durch verbinden oder vereinigen erfolgen. Bei einer Integration durch Verbindung werden vorher unverbundene Elemente, die aber logische Beziehungen zueinander aufweisen, zu einem System verbunden [Ros02]. Bei einer Integration durch Vereinigung geht ein neues, zusammengeführtes Modell hervor, welches das alte Modell ersetzt [Ros02].

In Bezug auf die vertikale Integration von Rahmenwerksebenen widerspricht eine Integration durch Vereinigen dem Zweck der Komplexitätsreduktion durch die Aufteilung in einzelne Ebenen. Die vertikale Integration setzt daher eine Integration durch Verbinden voraus.

2.3 Transformation vs. Zuordnung

Die Integration von Modellebenen kann durch Zuordnung oder Transformation erfolgen [Sin99]. Ein Beispiel für einen transformationsorientierten Ansatz ist die Model Driven Architecture (im Folgenden MDA) [MM03]. FRANKEL ET AL. zeigen, wie sich die MDA zur Integration von Modellebenen in dem Rahmenwerk von ZACHMAN [Zac87] einsetzen lässt [FHM⁺03]. Jede Rahmenwerksebene wird einem MDA-Modelltyp zugeordnet, zwischen denen eine top-down Transformationsbeziehung besteht.

Eine solche Transformationsbeziehung besteht in dem INPROVY-Rahmenwerk im Sinne eines Vorgehensmodells, bei dem die Ebenen von oben nach unten durchlaufen werden [EGL09]. Dieses Vorgehen ist sinnvoll bei der (initialen) Erstellung von Simulationsmodellen, stößt aber bei der Rückführung von Ergebnissen von unteren Ebenen in die höherliegenden Ebenen an seine Grenzen.

Bei den hier betrachteten semi-formalen Modellen kann nicht davon ausgegangen werden, dass diese automatisch neu generiert werden können [Geh07]. Eine manuelle Transformation ist aufgrund des Aufwandes nicht nach jeder Änderung möglich. Im Sinne eines *round-trip model engineering* müssen Wege gefunden werden die Modellebenen so zu integrieren, dass eine Änderung an einer Ebene nicht dazu führt, dass die darunter liegenden Ebenen komplett neu generiert werden müssen [SK04].

In diesem Fall ist eine Integration durch Zuordnung sinnvoll. Zuordnungen sind darüber hinaus Voraussetzung für Transformationen, bei denen Transformationsregeln auf Basis der Metamodellelemente beschrieben werden [MM03]. Es wird daher die Integration durch Zuordnungen untersucht.

2.4 Konflikte bei der Modellintegration

Die Schwierigkeit bei der Integration der Modellebenen besteht in der Überwindung von Integrationskonflikten. Ursprünglich aus der Schemaintegration von Datenbanken entstanden [BLN86] und dann auf die konzeptuelle Modellierung übertragen [Ros02], werden Namens-, Struktur- und Typkonflikte unterschieden.

Ein Typkonflikt liegt vor, wenn derselbe Sachverhalt in zwei Modellen durch unterschiedliche Sprachkonstrukte modelliert wird [Ros02]. Es liegt damit eine unterschiedliche Verwendung der Modellierungsgrammatiken vor. Beispielsweise kann in einem Entity-Relationship-Modell ein Sachverhalt als Entity-Typ oder als Attribut modelliert werden. Ein spezieller Typkonflikt liegt vor, wenn Wertebereiche unterschiedlich verwendet werden. In diesem Fall wird von einem Wertebereichskonflikt gesprochen [Geh07].

Wird der gleiche Sachverhalt in einem Modell (semantisch) unterschiedlich beschrieben, so wird dies als Strukturkonflikt bezeichnet. Es wird zwischen Abhängigkeits-, Abstraktions- und Detaillierungskonflikten differenziert [Geh07].

Ein Abhängigkeitskonflikt liegt dann vor, wenn gleiche Strukturen durch sich widersprechende Beziehungsarten modelliert werden. Beispiele hierfür sind unterschiedliche Mul-

tiplizitäten oder eine unterschiedliche Modellierung von Generalisierungs-/Spezialisierungsbeziehungen [Geh07]. Zu einem Abstraktionskonflikt kommt es, wenn der gleiche Sachverhalt unterschiedlich abstrakt bzw. generalisiert modelliert wird [Geh07]. Beispielsweise könnte in einem Modell der Sachverhalt als Buch und in einem anderen Modell als Publikation bezeichnet werden. Ein Detaillierungskonflikt dagegen tritt auf, wenn ein Realitätsausschnitt in unterschiedlicher Ausführlichkeit dargestellt wird [Geh07].

Bei einer unterschiedlichen Verwendung der Fachsprache wird ein Konflikt als Namenskonflikt bezeichnet. Es lassen sich Synonymkonflikte (unterschiedliche Wörter werden gleich verwendet) und Homonymkonflikte (gleiche Wörter werden unterschiedlich verwendet) unterscheiden [Geh07].

Da die beschriebenen Konflikte aus der Schemaentwicklung von Datenbanken bzw. der Prozessmodellintegration stammen, wurden diese bisher nur bei der horizontalen Integration identifiziert. Es stellt sich damit die Frage, ob diese Konflikte auch bei der vertikalen Integration auftreten.

In dem Fallbeispiel in Kapitel 4 wird gezeigt, dass die beschriebenen Konflikte auch bei der vertikalen Integration auftreten.

3 Evaluation

Für die Evaluation wird die Methode des merkmalsbasierten Vergleichs verwendet. Hierzu wird eine Liste von idealen Eigenschaften eines Integrationsansatzes aufgestellt, anhand derer die Ansätze evaluiert werden [SR98].

Problematisch an dieser Methode ist ein hohes Maß an Subjektivität. Zur intersubjektiven Nachvollziehbarkeit der Ergebnisse muss sowohl die Auswahl als auch die Interpretation ausreichend dokumentiert werden [SR98].

3.1 Evaluationsrahmen

Das Vorgehen bei der Erstellung des Evaluationsrahmens orientiert sich an den von HEINRICH vorgeschlagenen Schritten [Hei00]. Die Evaluationsobjekte wurden durch eine systematische Literaturrecherche gefunden [Die08]. Als Filter wird Tabelle 1 herangezogen. Um die Anforderung der Evaluationsmethode nach einer guten Dokumentation der Messung erfüllen zu können, müssen die betrachteten Ansätze selbst ausreichend dokumentiert sein [SR98]. Es werden daher nur Ansätze untersucht, die zwei oder mehr Publikationen aufweisen.

In Tabelle 2 sind die Evaluationsmerkmale zusammen mit den verwendeten Metriken aufgeführt.

Im Folgenden werden die Merkmale kurz erläutert. Zentrale Anforderung an einen Ansatz zur Modellintegration ist die *Überwindung der Integrationskonflikte*. Nur wenn diese

Kriterium		Metrik
A1	Überwindung der Integrationskonflikte	
	1 Überwindung Typkonflikt	Erfüllt, nicht erfüllt
	2 Überwindung Strukturkonflikte	Erfüllt, teilweise erfüllt, nicht erfüllt
A2	Werkzeugunterstützung (Automatisierbarkeit)	Automatisch, halb automatisch, manuell
A3	Implementierungsaufwand	Gering, mittel, hoch
A4	Unterstützung der	
	1 Wiederverwendbarkeit	Wird unterstützt, wird nicht unterstützt
	2 Anpassbarkeit	
der Metamodelle		

Tabelle 2: Evaluationsmerkmale und Metriken

Konflikte überwunden wurden kann von einem integrierten System gesprochen werden. Wird ein Konflikt sowohl auf Ebene der Syntax als auch der Semantik überwunden, so gilt die Anforderung als vollständig erfüllt. Wird ein Konflikt nur auf Ebene der Syntax überwunden, so wird die Anforderung als teilweise erfüllt bewertet. Typkonflikte können schon auf Ebene der abstrakten Syntax aufgelöst werden [Geh07]. Die Anforderung ist damit erfüllt, wenn der Typkonflikt auf Ebene der Syntax oder der Semantik gelöst wird.

Aufgrund der enormen Anzahl können bei komplexen Modellen nicht alle Beziehungen auf *Modellebene* manuell erstellt werden. Es müssen also Ansätze gefunden werden, die eine automatische oder halb-automatische Definition der Zuordnungen auf Modellebene erlauben [WBMW09]. Es wird daher bewertet, inwieweit ein Ansatz eine *Automatisierbarkeit* der Verbindungen auf Modellebene zulässt. Dieses Merkmal bezieht sich damit auf das Integrationsziel Aktualität [Ros02].

Bei der Integration müssen die Autonomiekosten dem Integrationsaufwand gegenübergestellt werden [Ros02]. Der *Aufwand*, der für die Implementierung eines Ansatzes erforderlich ist, findet daher Eingang in die Bewertung, auch wenn Kosten und Nutzen hier nicht quantitativ bewertet werden können. Einfluss haben alle Voraussetzungen, die für die Implementierung eines Ansatzes erfüllt werden müssen. Es wird hierzu geprüft, ob zusätzliche Artefakte, wie Beziehungs-Metamodelle oder Ontologien, erstellt oder bestimmte Maßnahmen durchgeführt werden müssen.

Zentrale Qualitätsmerkmale von Rahmenwerken sind *Wiederverwendbarkeit* und *Anpassbarkeit* des Rahmenwerks [Sin99]. Die Integration soll den Vorteil einer Aufteilung allerdings nicht dadurch zunichte machen, dass die Teile eng gekoppelt werden. Die Anforderung Wiederverwendbarkeit und Anpassbarkeit bezieht sich daher auf die *Metamodelle*. Diese sollten, zumindest in ähnlichen Projekten, wiederverwendet werden können und an-

passbar sein. Müssen die Metamodelle verändert werden, damit eine Integration stattfinden kann, so wird die Anforderung mit „schränkt ein“ bewertet.

3.2 Ansätze der Modellintegration – Stand der Forschung

Im Folgenden wird der Stand der Forschung zu Modellintegrationsansätzen dargestellt und diese evaluiert. Die Ansätze lassen sich zunächst in Metamodell-basierte (syntaktische) und Ontologie-basierte (semantische) Integrationsansätze aufteilen [ABH⁺06].

Folgende Ansätze wurden durch die Literaturrecherche gefunden, aber aus oben erläuterten Gründen nicht in die Evaluation aufgenommen:

- FELLMANN ET AL. untersuchen Modellbeziehungen mit semantischen Wikis. Die Beziehungen werden allerdings nicht auf Ebene der Modellelemente betrachtet, sondern die Modelle als geschlossene Artefakte. Der Ansatz wird nur durch eine Publikation beschrieben [FT09].
- SIMON ET AL. beschreiben einen horizontalen Ansatz zur Integration von Prozessmodellen. Im Fokus steht allerdings die vereinigende Integration der Modelle [SM07].
- FENGEL ET AL. stellen einen Ansatz zur semantischen Verlinkung heterogener Modelle vor. Hierzu werden die Modellelemente der zu integrierenden Modelle mit Schlagworten versehen. Der Ansatz wird im Rahmen eines Forschungsprojektes verfolgt und ist noch nicht durch eine ausreichende Anzahl an Publikationen beschrieben [FRN08].
- HAHN untersucht die Integration verteilter Produktmodelle mit Hilfe von Semantic-Web-Technologien. Die Dokumentation des Ansatzes ist für die Evaluation ebenfalls nicht ausreichend [Hah05].

3.2.1 Metamodell-basierte Integrationsansätze

Metamodell-basierte Integrationsansätze ermöglichen eine Integration auf Ebene der abstrakten Syntax.

Mulitperspektivische Unternehmensmodellierung (MEMO) Der Ansatz behandelt die vertikale Integration der Ebenen „Strategie“, „Organisation“ und „Informationssystem“ [Fra02]. Die Integration der Sprachen erfolgt über gemeinsame Konzepte der Metamodelle, d.h. ein Konzept wird in mehreren Metamodellen verwendet [Fra94]. Bei der MEMO SML wird beispielsweise das Konzept Organisationseinheit aus der MEMO OrgML in Zusammenhang mit den Konzepten „StrategicBusUnit“ und „HumanRessource“ verwendet [Fra02].

Es handelt sich folglich um eine verbindende Integration durch Zuordnung. Im Kern entspricht dieser Ansatz der Integration von Sichten innerhalb eines Modells. In der Konsequenz entsteht eine „Super“-Sprache, die sich über alle Modellebenen erstreckt.

Enterprise Model Integration (EMI) Zentrale Konzepte des EMI Ansatzes sind Mappings und Integrationsregeln [KBJK03]. Mappings beschreiben, welche Teile der Metamodelle integriert werden. Integrationsregeln definieren, wie diese Mappings umgesetzt werden [ZS07].

Mappings stellen eine Integration über Zuordnungen her. Integrationsregeln können entweder eine verbindende Integration (durch Angleichungsregeln) oder eine vereinigende Integration (durch Verbindungsregeln) realisieren. Integrationsmodelle enthalten die Mappings zwischen den Sprachkonzepten der zu integrierenden Modelle [ZS07]. Dies entspricht den von SINZ beschriebenen Beziehungs-Metamodellen [Sin99].

3.2.2 Ontologie-basierte Integrationsansätze

Ontologie-basiert Ansätze streben eine Integration auf der Ebene der Semantik an.

Integration über natürlichsprachliche Ausdrücke nach HÖFFERER Der Ansatz von HÖFFERER ordnet natürlichsprachlichen Ausdrücken, die ein Modellelement enthält, Konstrukten einer Ontologie zu [Hö07]. HÖFFERER betrachtet die horizontale Integration von Modellen, wie sie bei der Integration von Prozessen unterschiedlicher Unternehmen auftritt [KH08].

Integration über semantische Annotationen nach LIN Ebenso wie bei HÖFFERER wird bei LIN die horizontale Integration von Prozessmodellen betrachtet [LSH⁺06]. Im Unterschied zu HÖFFERER werden allerdings dem *Modellelement* Konzepte aus der Ontologie zugeordnet und nicht den natürlichsprachlichen Ausdrücken, die das Modellelement beschreiben [LK09].

3.3 Evaluationsergebnis

Tabelle 3 fasst das Evaluationsergebnis zusammen.

		Metamodell-basiert		Ontologie-basiert	
		MEMO	EMI	HÖFFERER	LIN
Int. Richtung		vertikal	vertikal/horizontal	horizontal	horizontal
A1	1	+	+	+	+
	2		o	+	+
	3	o	-	+	+
A2			o	o	-
A3		+	o	-	-
A4	1		+	+	+
	2		+	+	+

Legende

- + (A1: erfüllt / A2: automatisch / A3: gering / A4: unterstützt)
- o (A1: teilweise erfüllt / A2: halb automatisch / A3: mittel)
- (A1: nicht erfüllt / A2: manuell / A3: hoch / A4: schränkt ein)

Tabelle 3: Evaluationsergebnis

3.3.1 Vergleichende Bewertung der Ansätze (Evaluationsziel 1)

Ein Vergleich der Integrationsrichtung zeigt, dass unter den untersuchten Ansätzen nur die Metamodell-basierten zur vertikalen Integration eingesetzt werden. Die Ontologie-basierten werden bisher nicht zur Integration in vertikaler Richtung eingesetzt. Hier besteht eine Forschungslücke, die in dem Fallbeispiel in Kapitel 4 näher untersucht wird.

Mit allen Ansätzen ist es möglich, Typkonflikte zu überwinden. Strukturkonflikte lassen sich mit den ontologiebasierten Ansätzen vollständig überwinden. Mit dem EMI-Ansatz ist durch die verfeinerten Beziehungstypen eine eingeschränkte Explikation von Strukturkonflikten auf der Ebene der abstrakten Syntax möglich [ZS07]. Auf Ebene der Semantik ist dies nicht möglich, da bei materialen Domänen die Semantik nicht auf die Syntax zurückgeführt werden kann [Geh07].

Namenskonflikte können vollständig nur mit Hilfe der Ontologie-basierten Ansätze überwunden werden. Mit dem MEMO-Ansatz können diese teilweise vermieden werden, indem fachsprachliche Ausdrücke in das Metamodell übernommen werden [Fra02]. Ein einheitliches Verständnis der übernommenen fachsprachlichen Ausdrücke ist allerdings nicht sichergestellt.

Die Anforderung Automatisierbarkeit wird von keinem Ansatz vollständig erfüllt. EMI und der Ansatz nach HÖFFERER erlauben, durch ihre Teil-Automatisierbarkeit, eine Unterstützung des Modellerstellers durch ein Werkzeug. Die Ansätze nach LIN und MEMO erlauben nur eine manuelle Integration und sind für die Integration komplexer Rahmenwerke nicht geeignet.

Den geringsten Implementierungsaufwand weist der Integrationsansatz MEMO auf. Die Ontologie-basierten Ansätze erfordern den höchsten Aufwand, allerdings erfüllen sie auch die meisten Anforderungen.

Der MEMO-Ansatz stellt bezüglich der Wiederverwendbarkeit den schlechtesten Ansatz dar, da die Modellierungssprachen der unterschiedlichen Ebenen nicht getrennt voneinander genutzt werden können. Durch die Nutzung von getrennten Integrationsmodellen im EMI-Ansatz kann dieser als wesentlich besser in Bezug auf die Wiederverwendbarkeit angesehen werden. Die Ontologie-basierten Ansätze erfordern keine Veränderung der Metamodellebene und sind daher in Bezug auf die Wiederverwendbarkeit ebenfalls als positiv zu beurteilen. Das gleiche gilt für die Anpassbarkeit der Metamodelle.

In Bezug auf die Integration der INPROVY Rahmenwerksebenen konnten die Anforderungen A1, A2 und A4 als Sachziele bei der Implementierung identifiziert werden. Demnach ist nur der Ansatz nach HÖFFERER für die Lösung der Integrationsproblematik geeignet. Tabelle 3 zeigt allerdings auch, dass eine geschickte Kombination der Eigenschaften der Metamodell-basierten Ansätze die Sachziele erfüllen könnte. Hierzu sind die relevanten Gestaltungsdimensionen zu identifizieren.

3.3.2 Identifizierte Gestaltungsdimensionen (Evaluationsziel 2)

Die vergleichende Betrachtung der Ansätze zeigt die folgenden Gestaltungsdimensionen auf. Bei den Metamodell-basierten Ansätzen besteht Gestaltungsspielraum hinsichtlich der *Detailliertheit des Metamodells*. Ein höheres Maß an Detailliertheit bedeutet, dass die Zeichenklassen, deren Beziehungen im Metamodell beschrieben werden, feiner aufgelöst werden. Je detaillierter das Metamodell ist, desto besser lassen sich auch die Beziehungen zwischen zwei Metamodellen beschreiben.

Ein grobgranulares Metamodell enthält beispielsweise die Konstrukte „Stelle“ und „Organisationseinheit“, wohingegen ein feingranulares Metamodell beispielsweise die Stellentypen „Controller“ und „Finanzbuchhalter“ unterscheidet [Win09].

Ein hohes Maß an Detailliertheit steht allerdings in Konflikt mit der Wiederverwendbarkeit [Fra02]. Werden Begriffe aus der Fachsprache in das Metamodell aufgenommen, so kann nicht davon ausgegangen werden, dass diese in einem anderen Unternehmen gleichermaßen verwendet werden.

Eine weitere Gestaltungsdimension wird von den *Integrationsbeziehungen* aufgespannt. Die Ansätze unterscheiden sich darin, welche Typen von Beziehungen zugelassen werden. Diese Gestaltungsdimension wird auch von WEIDLICH ET AL. als wichtig erachtet [WBMW09]. Integrationsbeziehungen können einfacher oder komplexer Art, wie Generalisierungen oder Aggregationen, sein.

Interessant ist ein Unterschied zwischen den beiden Ontologie-basierten Ansätzen. Komplexe Beziehungen werden entweder nur innerhalb der Ontologie angewendet (HÖFFERER) oder zwischen Ontologie und Modell (LIN). Der Konstruktionsaufwand verschiebt sich damit von der Ontologiekonstruktion hin zur Verbindung von Modell und Ontologie. Auch wenn die Wiederverwendbarkeit der Ontologien nicht untersucht wurde, da diese nicht Bestandteil von Rahmenwerken sind, so ist die erste Möglichkeit wesentlich besser. Die

Beziehungen innerhalb der Ontologie können wiederverwendet werden, die Beziehungen zwischen Modell und Ontologie nicht.

Der Einfluss der beiden Integrationsdimensionen wird in dem nachfolgenden Fallbeispiel untersucht.

4 Fallbeispiel

Das Beispiel stammt aus dem Forschungsprojekt INPROVY und betrachtet eine Betonspritzpumpe (vgl. Abbildung 2). In Abschnitt 4.1 des Fallbeispiels wird der Einfluss der identifizierten Gestaltungsdimensionen untersucht. In Abschnitt 4.2 wird die Anwendung von Ontologien auf die vertikale Integration anhand des Fallbeispiels geprüft.

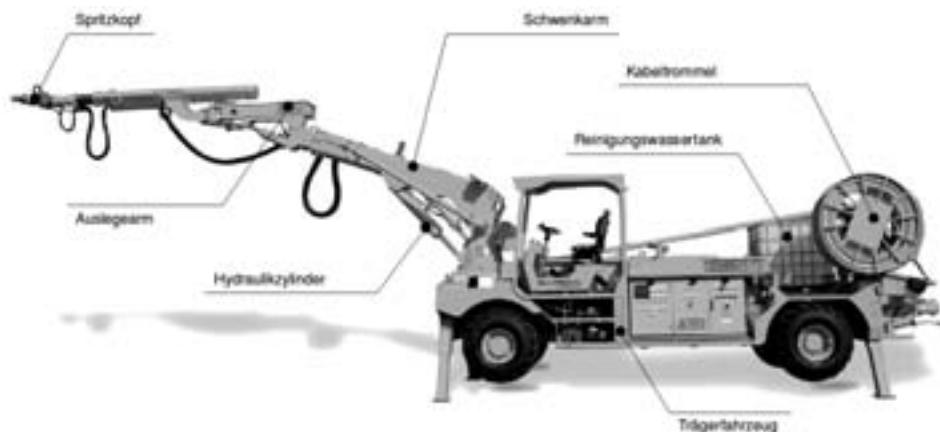


Abbildung 2: Betonspritzpumpe

Auf der Fachebene des INPROVY-Rahmenwerks befindet sich ein konzeptuelles Produktmodell, das die Produktkomponenten dieser Betonspritzpumpe darstellt. Eine Produktkomponente ist etwa dadurch charakterisiert, dass sie eine Artikelnummer besitzt und mit Hilfe dieser Nummer von einem Zulieferer bestellt oder im Lager aufgefunden werden kann. Die Beschreibung der Produktkomponenten ist eine statische Sicht auf die Baumaschine, auf die in einer Prozesssicht referenziert werden kann, um etwa einen Bestellprozess zu beschreiben.

Auf der konzeptuellen Simulationsebene werden die Simulationskomponenten der Betonspritzpumpe beschrieben, die den Ausgangspunkt für verschiedene Simulationsanwendungen darstellen. Beispiele für solche Anwendungen sind die Echtzeitsimulation oder die Detailsimulation einzelner Komponenten zur Vorausentwicklung [EGL09]. Dieses konzeptuelle Simulationsmodell ist ebenfalls eine statische Sicht auf die Betonspritzpumpe, allerdings aus einem anderen Blickwinkel.

In dem Produktmodell werden Komponenten wie die Kabeltrommel oder der Reinigungswassertank, aufgrund verschiedener Artikelnummern, einzeln abgebildet. Im konzeptuellen Simulationsmodell sind, wenn der Schwerpunkt der Simulation etwa auf dem Spritzarm liegt, die Kabeltrommel und der Reinigungswassertank zu dem Modellelement „Trägerfahrzeug“ zusammengefasst.

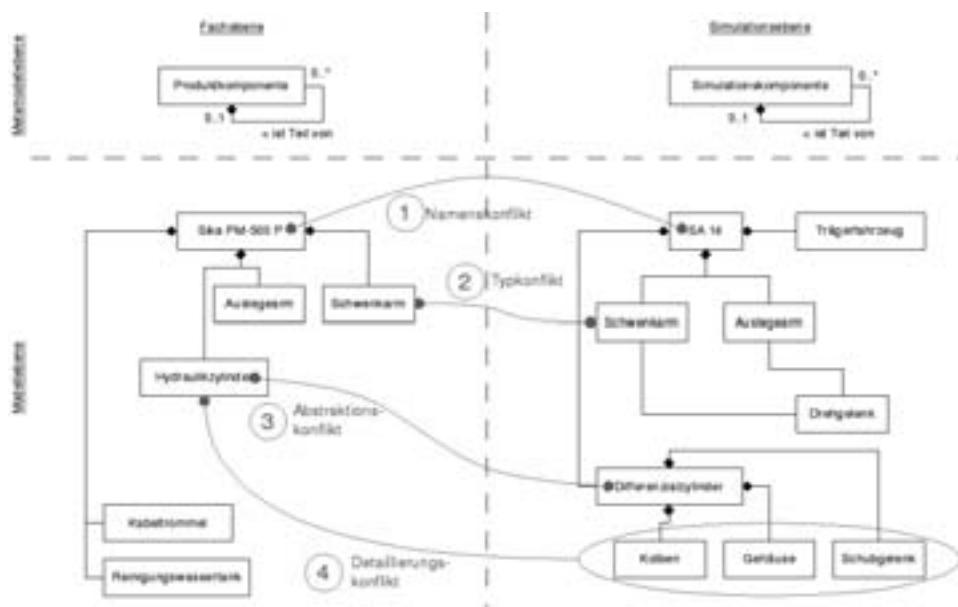


Abbildung 3: Modelle des Fallbeispiels

In Abbildung 3 ist ein Beispielmodell der Fachebene sowie ein Beispielmodell der Simulationsebene dargestellt. Eine Betonspritzpumpe besteht aus mehreren hundert Teilen. Beide Modelle stellen daher nur einen möglichen Ausschnitt dar. Über den Beispielmodellen sind (ebenfalls stark verkürzt) die Metamodelle der Beispielmodelle dargestellt. Zwischen den beiden Modellen bestehen mehrere Integrationskonflikte, von denen exemplarisch vier rot dargestellt sind.

1. Namenskonflikt: In dem Fachmodell wird die Betonspritzpumpe mit dem Produktnamen bezeichnet, während in dem Simulationsmodell die firmeninterne Typenbezeichnung verwendet wird.
2. Typkonflikt: Der Schwenkkarm ist in dem Fachmodell vom Typ „Produktkomponente“, in dem Simulationsmodell allerdings vom Typ „Simulationskomponente“.
3. Abstraktionskonflikt: Die unterschiedlichen Teile des Spritzarms sind durch Hydraulikelemente verbunden. Im Fachmodell wird ein solches Element als „Hydraulikzylinder“ modelliert. Im Simulationsmodell wird dies als „Differentialzylinder“ modelliert. Letzterer ist allerdings ein spezieller Hydraulikzylinder.
4. Detailierungs-konflikt: Im Bereich der Kabeltromme und des Reinigungswassertanks sowie im Differenzialzylinder.

4. Detaillierungskonflikt: Da die unterschiedlichen Elemente eines solchen Differentialzylinders für die Simulation wichtig sind, da sie das Verhalten beeinflussen, werden Kolben, Gehäuse und Schubgelenk im Simulationsmodell gesondert dargestellt. Im Fachmodell hingegen wird der Zylinder als ein Element modelliert, da nur das Teil in seiner Gesamtheit eine Artikelnummer besitzt.

4.1 Metamodell-basierte Lösung

Die erste Möglichkeit besteht in der Integration auf Basis der Metamodelle. Hierzu werden die Beziehungen zwischen den Elementen der beiden Metamodelle modelliert. In einem ersten Schritt kann eine Beziehung zwischen dem Element „Produktkomponente“ und dem Element „Simulationskomponente“ modelliert werden. Dies ist in Abbildung 4 dargestellt.

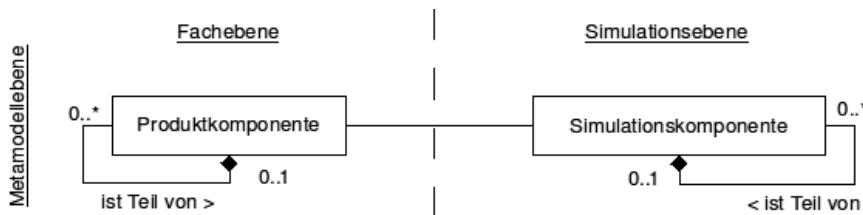


Abbildung 4: Beziehungen zwischen Metamodellen der Fach- und Simulationsebene

Durch die Modellierung einer solchen Beziehung auf Metamodellebene ist es möglich den oben beschriebenen Typkonflikt zu explizieren, indem eine Verbindung auf Modellebene zwischen den beiden Elementen „Produktkomponente:Schwenkarm“ und „Simulationskomponente:Schwenkarm“ modelliert wird.

Die übrigen Integrationskonflikte gehen über die reine Zuordnung von Metamodellelementen hinaus. Namens- und Strukturkonflikte können hierdurch nicht dargestellt werden. Die vergleichende Evaluation der Integrationsansätze in Abschnitt 3.3.2 hat zwei Gestaltungsdimensionen identifiziert. Zum einen die semantische Verfeinerung der Beziehungen zwischen den Metamodellelementen und zum anderen die Verfeinerung der Zeichenklassen, die in den Metamodellen verwendet werden.

Eine Verfeinerung des Metamodells bedeutet, dass fachsprachliche Begriffe in das Metamodell aufgenommen werden. Die Zeichenklassen, deren Beziehungen im Metamodell beschrieben werden, zu spezialisieren, führt dazu, dass das Metamodell umfangreicher wird. Der Namenskonflikt kann durch das in Abbildung 5 dargestellte Metamodell gelöst werden. Das Beispiel zeigt zugleich den gravierenden Nachteil dieses Ansatzes auf. Eine solche Verfeinerung der Zeichenklassen führt zu einem sehr komplexen Metamodell, dessen Wiederverwendbarkeit stark eingeschränkt ist.

Eine weitere Gestaltungsmöglichkeit besteht in der Verfeinerung der Beziehungen zwischen den Metamodellelementen. In der vergleichenden Evaluation wurde beispielsweise

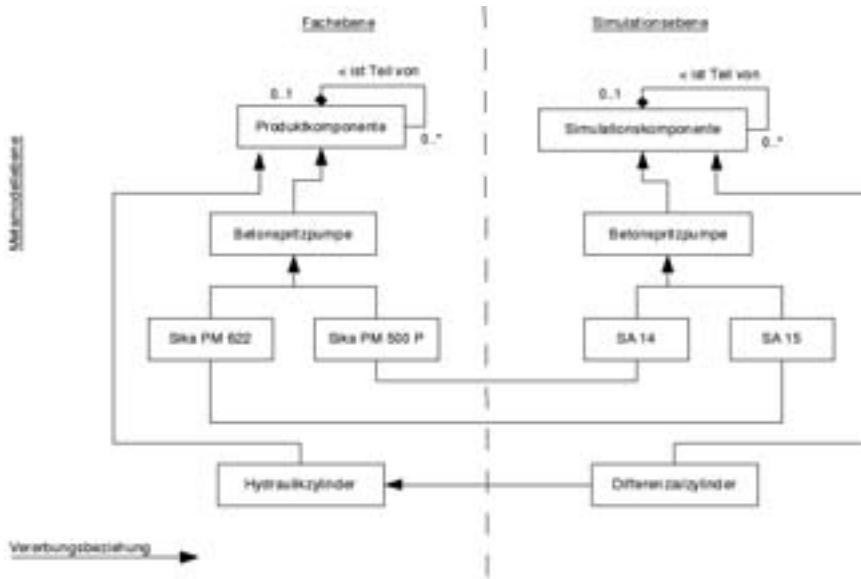


Abbildung 5: Verfeinerte Beziehungen zwischen Metamodellen der Fach- und Simulationsebene

eine Spezialisierungsbeziehung genannt. Diese könnte auf Modellebene verwendet werden, um den Abstraktionskonflikt zwischen dem Modellelement „Hydraulikzylinder“ und dem Element „Differentialzylinder“ zu beschreiben. Dies muss für jedes Modell manuell erfolgen und ist bei großen Modellen daher nur unter unverhältnismäßig hohem Aufwand zu bewerkstelligen.

Die Verwendung dieser Spezialisierungsbeziehung auf Metamodellebene würde dagegen ebenfalls ein verfeinertes Metamodell voraussetzen, damit diese Beziehung zwischen der Zeichenklasse „Hydraulikzylinder“ und „Differentialzylinder“ modelliert werden kann. Folglich entsteht ein ebenso schlecht wiederverwendbares Metamodell.

Mit Hilfe der Metamodell-basierten Ansätze können Typkonflikte expliziert werden. Abstraktions- und Strukturkonflikte können damit nur auf Ebene der abstrakten Syntax überwunden werden. Bei materialen Domänen kann die Semantik nicht auf die Syntax zurückgeführt werden [Geh07]. Damit ist die einheitliche Verwendung von Konstrukten verfeineter Metamodelle nicht sichergestellt. Trotz dieser Einschränkungen konnte gezeigt werden, dass sich verfeinerte Beziehungen und ein detaillierteres Metamodell gegenseitig bedingen. Bei der Überwindung der Integrationskonflikte sind diese, aufgrund des überschaubaren Implementierungsaufwands, als positiv zu beurteilen. Der Widerspruch zwischen Wiederverwendbarkeit und Nutzbarkeit verfeineter Metamodelle ist durch weitere Forschung zu adressieren. Denkbar ist die Trennung zwischen einem Kern-Metamodell und Fachsprachen-Metamodell-Erweiterungen [KR98], [KÖ6].

4.2 Ontologie-basierte Lösung

Die beschriebenen Integrationskonflikte könnten auch mit Hilfe von Ontologien expliziert und eine indirekte Integration der Modelle erreicht werden. In Abbildung 6 (oben) sind zwei Beispielontologien dargestellt.

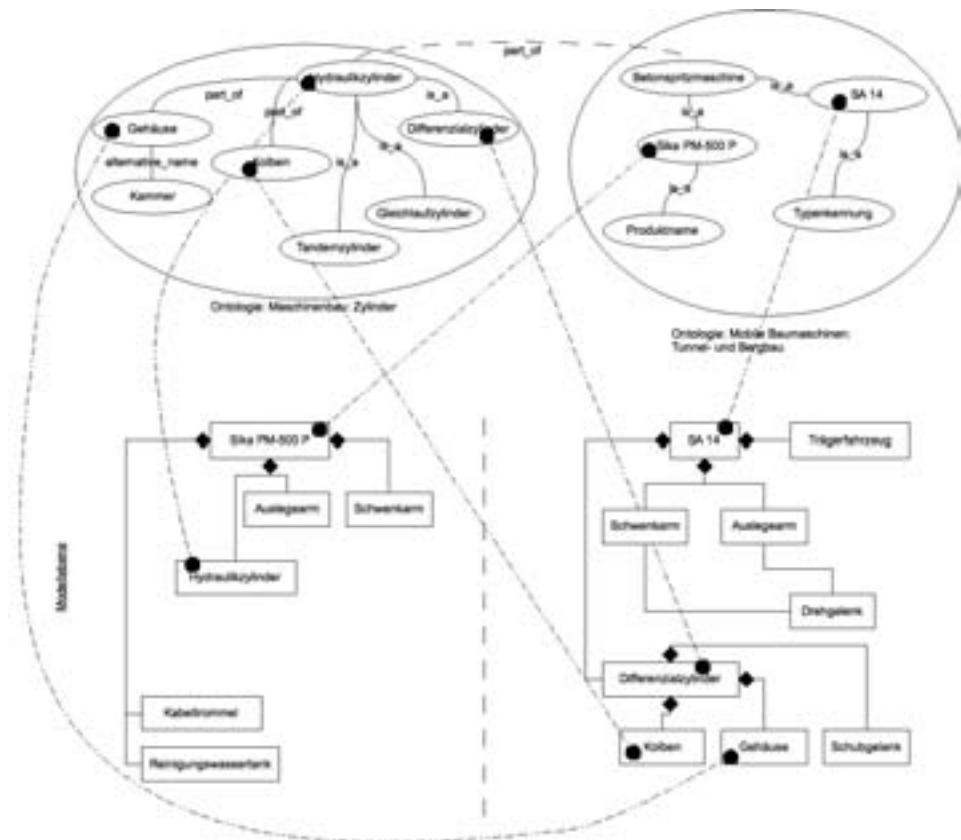


Abbildung 6: Ontologie-basierte Lösung

Die erste Ontologie beschreibt den Zusammenhang zwischen fachsprachlichen Ausdrücken aus dem Bereich Maschinenbau. Die Zweite beschreibt Begriffe aus der Domäne der mobilen Baumaschinen.

Auf Modellebene werden die einzelnen Elemente den Konzepten der Ontologien zugeordnet (in Abbildung 6 als gestrichelte Linie mit Punkten dargestellt). Dies könnte beispielsweise halb-automatisch mit Hilfe von Ähnlichkeitsmaßen durch das Modellierungswerzeug erfolgen [HÖ7].

Der Namenskonflikt kann dadurch überwunden werden, dass die Ontologien ausgewertet

werden. Beide Begriffe stellen eine Betonspritzpumpe dar und unterscheiden sich dadurch, dass sie ein Produktnamen bzw. eine Typenkennung sind. Alternativ wäre es auch möglich eine Synonymbeziehung zwischen „SA 14“ und „Sika PM-500 P“ zu modellieren. Der Abstraktionskonflikt zwischen „Hydraulikzylinder“ und „Differenzialzylinder“ wird dadurch überwunden, dass die beiden Begriffe mit einer Spezialisierungsbeziehung verbunden sind. Der Detaillierungskonflikt wird dadurch expliziert, dass Kolben und Gehäuse bzw. Kammer Teile eines jeden Hydraulikzylinders sind.

In dem Beispiel können alle Konflikte aufgelöst werden. Für die Integration werden jedoch zwei Ontologien benötigt. Diese müssen wiederum integriert werden. In dem Beispiel wird eine Teil-Ganzes-Beziehung zwischen zwei Konzepten manuell modelliert (in Abbildung 6 als grob gestrichelte Linie dargestellt). Bei umfangreichen Ontologien erreichen solche Beziehungen schnell eine große Zahl und verursachen einen hohen Integrationsaufwand. Durch das Beispiel wurde die Anwendbarkeit Ontologie-basierter Ansätze auf die vertikale Integration nachgewiesen.

5 Fazit und weiterer Forschungsbedarf

Die Evaluation hat gezeigt, dass der Ansatz von HÖFFERER als einziger alle Integrationskonflikte überwindet, Wiederverwendbarkeit und Anpassbarkeit unterstützt und eine zumindest halb automatische Integration zulässt. Negativ zu bewerten ist der hohe Implementierungsaufwand.

Darüber hinaus muss das insgesamt positive Abschneiden der Ontologie-basierten Ansätze relativiert werden. Wird davon ausgegangen, dass die Semantik von Zeichen durch ihren Gebrauch in der Sprache zustande kommt, so ist nicht sichergestellt, dass eine Ontologie diesen Gebrauch korrekt und vollständig abbildet [Geh07].

Ein weiteres Problem entsteht durch die Verwendung unterschiedlicher Ontologien. Da auf unterschiedlichen Ebenen des Rahmenwerkes verschiedene Fachsprachen verwendet werden, ist es wenig wahrscheinlich, dass alle Fachausdrücke für alle Modellebenen in einer Ontologie definiert werden können. Alle betrachteten Ansätze nutzen daher auch unterschiedliche Ontologien. Diese Ontologien müssen ihrerseits wieder integriert werden. Hierbei treten ähnliche Probleme auf wie bei der Integration der Modellebenen [VJBS98]. Das Integrationsproblem verschiebt sich daher nur auf eine andere Ebene.

Eine Lösung könnte darin bestehen, die in der Evaluation identifizierten Gestaltungsdimensionen auszuschöpfen. Unter Nutzung von detaillierten Metamodellen und verfeinerten Integrationsbeziehungen werden die Abhängigkeiten zwischen den Modellebenen durch ein Integrationsmodell beschrieben, das von einem Modellierungstool ausgewertet wird.

Die Anreicherung des Metamodells mit fachsprachlichen Ausdrücken erhöht die Komplexität und verringert die Wiederverwendbarkeit. Hierfür muss eine Lösung gefunden werden, welche die Erweiterbarkeit eines Kern-Metamodells sicherstellt.

Es ist eine Integrationssprache zu entwickeln, in der das Integrationsmodell erstellt wird.

Diese steht im Spannungsfeld zwischen den semi-formalen Sprachen, die integriert werden sollen und der Anforderung, durch eine Modellierungswerkzeug auswertbar zu sein.

Weiterhin hat die Evaluation gezeigt, dass die bisher nur bei der horizontalen Integration untersuchten Konflikte auch bei der vertikalen Integration zu überwinden sind.

Allen betrachteten Ansätzen liegt die Integration von Sprachen zugrunde. Sei es eine Integration der Modellierungssprachen durch die Metamodell-basierten Ansätze oder eine Integration der Fachsprachen durch Ontologie-basierte Ansätze. Hinter der Überwindung der Sprachbarrieren steht die Vorstellung, dass eine Spracheinigkeit zur Integration führt.

Eine Integration ohne Integrationsziel ist allerdings wenig realistisch. Eine Integration wird sich nur für eine Auswahl festgelegter Aspekte durch zweckgebundene Artefakte implementieren lassen. Hierzu wird ein entsprechender Prototyp entwickelt.

6 Danksagung

Das Forschungs- und Entwicklungsprojekt „Integrative Produktentwicklung mit virtuellen Prototypen (INPROVY)“ wird mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) innerhalb des Rahmenkonzepts „Forschung für die Produktion von morgen“ gefördert und vom Projektträger Karlsruhe (PTKA) betreut. Weitere Informationen unter <http://www.inprov.y.de>.

Literatur

- [ABH⁺06] K. Arnarsdóttir, A. J. Berre, A. Hahn, M. Missikoff und F. Taglino. Semantic mapping: ontology-based vs. model-based approach Alternative or complementary approaches? In *Proceedings of the Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability, Luxembourg*, 2006.
- [ARW08] Stephan Aier, Christian Riege und Robert Winter. Unternehmensarchitektur – Literaturüberblick und Stand der Praxis. *WIRTSCHAFTSINFORMATIK*, 50(4):292–304, 2008.
- [BLN86] C. Batini, M. Lenzerini und S. B. Navathe. A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys (CSUR)*, 18(4):364, 1986.
- [Die08] Die Sprecher der Wissenschaftlichen Kommission Wirtschaftsinformatik im Verband der Hochschullehrer für Betriebswirtschaft (WKWI) und des Fachbereichs Wirtschaftsinformatik der Gesellschaft für Informatik (GI-FB WI). WI-Orientierungslisten. *WIRTSCHAFTSINFORMATIK*, 50(2):155–163, April 2008.
- [EGL09] Werner Esswein, Steffen Greiffenberg und Sina Lehrmann. Framework zur modellgestützten Simulation. *Wissensportal baumaschine.de*, (2), 2009.
- [FHM⁺03] D. S Frankel, P. Harmon, J. Mukerji, J. Odell, M. Owen, P. Rivitt, M. Rosen und R. M Soley. The Zachman Framework and the OMG's Model Driven Architecture. *Business Process Trends*, (9):1–14, 2003.

- [Fra94] Ulrich Frank. *Multiperspektivische Unternehmensmodellierung. Theoretischer Hintergrund und Entwurf einer objektorientierten Entwicklungsumgebung*. Number 225 in GMD-Bericht. Oldenbourg, München u.a., 1994. Zugl.: Marburg, Univ., Habil.-Schr., 1993.
- [Fra02] Ulrich Frank. Multi-Perspective Enterprise Modeling (MEMO) - Conceptual Framework and Modeling Languages. *PROCEEDINGS OF THE 35TH HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES, HAWAII*, Seiten 1258—1267, 2002.
- [FRN08] J. Fengel, M. Rebstock und M. Nüttgens. Modell-Tagging zur semantischen Verlinkung heterogener Modelle. In *Proceedings der EMISA 2008, Bonn/St. Augustin 18.-19.09.2008*, Bonn u.a., 2008.
- [FT09] Michael Fellmann und Oliver Thomas. Management von Modellbeziehungen mit Semantischen Wikis. In *Proceedings der 9. Internationalen Tagung Wirtschaftsinformatik (WI 2009), 25.-27. Februar in Wien*, Seiten 673–683, Wien, 2009. Österreichische Computer Gesellschaft.
- [Geh07] Andreas Gehlert. *Migration fachkonzeptueller Modelle*. Logos, Berlin, 2007.
- [Hö07] P. Höfferer. Achieving business process model interoperability using metamodels and ontologies. In *Proceedings of 15th European Conference on Information Systems*, Seite 1620–1631, 2007.
- [Hah05] A. Hahn. Integration verteilter Produktmodelle durch Semantic-Web-Technologien. *Wirtschaftsinformatik*, 47(4):278–284, 2005.
- [Hei00] Lutz J. Heinrich. Bedeutung von Evaluation und Evaluationsforschung in der Wirtschaftsinformatik. In Lutz J. Heinrich und Irene Häntsche, Hrsg., *Evaluation und Evaluationsforschung in der Wirtschaftsinformatik*, Seiten 7–22. Oldenbourg, München u.a., 2000.
- [HMPR04] A. R Hevner, S. T March, J. Park und S. Ram. Design science in information systems research. *Management information systems quarterly*, 28(1):75–106, 2004.
- [Kö06] Thomas Kühne. Matters of (Meta-) Modeling. *Software and Systems Modeling*, 5(4):369–385, 2006.
- [KBJK03] Harald Kühn, Franz Bayer, Stefan Junginger und Dimitris Karagiannis. Enterprise Model Integration. In *4th International Conference, EC-Web Prague, Czech Republic, September 2-5, 2003 Proceedings*, Jgg. 2738/2003 of *Lecture Notes in Computer Science*, Seiten 379–392, Berlin, 2003. Springer.
- [KH08] Dimitris Karagiannis und Peter Höfferer. Metamodeling as an Integration Concept. In Joaquim Filipe, Boris Shishkov und Markus Helfert, Hrsg., *Software and Data Technologies*, Jgg. 1 of *Communications in Computer and Information Science*, Seiten 37–50. Springer, Berlin u.a., 2008.
- [KR98] M. Kugeler und M. Rosemann. Fachbegriffsmodellierung für betriebliche Informationssysteme und zur Unterstützung der Unternehmenskommunikation. *Fachausschuss 5.2 der Gesellschaft fu?r Informatik e. V. (GI): Informationssystem-Architekturen*, 5:8–15, 1998.
- [LK09] Yun Lin und John Krogstie. Quality Evaluation of a Business Process Semantic Annotations Approach. *International Journal of Interoperability in Business Information Systems (IBIS)*, 1(3), 2009.

- [LSH⁺06] Yun Lin, Darius Strasunskas, Sari Hakkarainen, John Krogstie und Arne Solvberg. Semantic Annotation Framework to Manage Semantic Heterogeneity of Process Models. In *Advanced Information Systems Engineering*, number 4001/2006 in Lecture Note in Computer Science, Seiten 433–446. Springer, Berlin, 2006.
- [MM03] Jishnu Mukerji und Joaquin Miller. MDA Guide Version 1.0.1, 2003.
- [Ros02] Michael Rosemann. *Komplexitätsmanagement in Prozeßmodellen*. Gabler, Wiesbaden, März 2002.
- [Sin99] E. J. Sinz. Architektur von Informationssystemen. In Peter Rechenberg und Gustav Pomberger, Hrsg., *Informatik-Handbuch*, Seiten 1035–1046. Hanser, München, 2. Auflage, 1999.
- [SK04] S Sendall und JM Köster. Taming Model Round-Trip Engineering. In *Proceedings of Workshop on Best Practices for Model-Driven Software Development*, 2004.
- [SM07] Carlo Simon und Jan Mendling. Integration of Conceptual Process Models by the Example of Event-driven Process Chains. In *8. Internationale Tagung Wirtschaftsinformatik WI, Karlsruhe, Germany, February 28 - March 2, 2007*, Seiten 677–694, Karlsruhe, 2007. Universitätsverlag Karlsruhe.
- [SR98] K. Siau und M. Rossi. Evaluation of information modeling methods-a review. In *Proceedings of the Thirty-First Annual Hawaii International Conference on System Sciences*, Jgg. 5, Seite 314–322, 1998.
- [VH05] Piet Verschuren und Rob Hartog. Evaluation in Design-Oriented Research. *Quality and Quantity*, 39(6):733–762, Dezember 2005.
- [VJBS98] P. R.S Visser, D. M Jones, T. J. M. Bench-Capon und M. J. R. Shave. Assessing heterogeneity by classifying ontology mismatches. In *Formal ontology in information systems: proceedings of the First International Conference (FOIS'98), June 6-8, Trento, Italy*, Seiten 148–162, 1998.
- [WBMW09] M. Weidlich, A. Barros, J. Mendling und M. Weske. Vertical Alignment of Process Models-How can we get there? In *CAiSE 2009 Workshop Proceedings-10th Workshop on Business Process Modeling, Development, and Support (BPMDS)*, ser. LNBIP, S. Nurcan, R. Schmidt, P. Soffer, and R. Ukor, Eds, Jgg. 29, Seite 71–84, 2009.
- [Win09] Robert Winter. *Management von Integrationsprojekten: Konzeptionelle Grundlagen und Fallstudien aus fachlicher und IT-Sicht*. Springer, Berlin, 1. Auflage, 2009.
- [Zac87] J. A. Zachman. A framework for information systems architecture. *IBM systems journal*, 26(3):276–292, 1987.
- [ZS07] Karagiannis D Zivkovic S, Kühn H. Facilitate Modelling Using Method Integration: An Approach Using Mappings and Integration Rules. In *Proceedings of the 15th European Conference on Information Systems (ECIS2007)*, Seiten 2038–2050, St. Gallen, 2007.

An ontology-driven approach to support semantic verification in business process modeling

Michael Fellmann¹, Frank Hogreve², Oliver Thomas¹, Markus Nüttgens²

¹Universität Osnabrück,
Institut für Informationsmanagement und Unternehmensführung,
Fachgebiet Informationsmanagement und Wirtschaftsinformatik,
Katharinenstraße 3, 49069 Osnabrück
{Michael.Fellmann|Oliver.Thomas}@uni-osnabrueck.de

²Universität Hamburg,
Fakultät Wirtschafts- und Sozialwissenschaften,
Lehrstuhl für Wirtschaftsinformatik,
Von-Melle-Park 5, 20146 Hamburg
{frank.hogreve|markus.nuettgens}@wiso.uni-hamburg.de

Abstract: This paper presents an ontology-driven approach that aims at supporting semantic verification of semi-formal process models. Despite the widespread use of these models in research and practice, the verification of process model information is still a challenging issue. We suggest an ontology-driven approach making use of background knowledge encoded in formal ontologies and rules. In the first step, we develop a model for ontology-based representation of process models. In the second step, we use this model in conjunction with rules and machine reasoning for process model verification. We apply our approach using real-life administrative process models taken from a capital city.

1 Introduction

1.1 Motivation

Models are important to manage complexity. They provide a means for understanding the business process, and understanding already is a benefit. This is indicated by a study from Gartner revealing an increase in efficiency of 12 percent gained solely by documenting actions and organizational responsibilities in process models [Me05, p. 4]. Moreover, process models serve for optimization, reengineering and implementation of supporting IT systems. Due to the importance of process models, model quality and correctness is important. According to ISO 8402, quality is “the totality of characteristics of an entity that bear on its ability to satisfy stated and implied needs”. Facets of quality are – amongst others – appropriateness in respect to the abstraction level of the representation (scale), detail of representation (granularity), compliance (conformance to rules

and regulations), adequate coverage of the model object, usefulness and correctness. We concentrate on correctness as the most fundamental quality aspect. Among the aspects of correctness are fundamentally: (a) syntax and formal semantics (structure and grammar), (b) linguistic aspects (labels) and (c) semantics (content). There is much research on (a) e.g. to detect deadlocks in works such as [MeAa07]. Aspects of (b) are increasingly focused in the scientific community to ensure compliance to naming conventions, see e.g. [PeWe09], but (c) has been neglected so far: semantic correctness means that the facts captured in the model about an object are (assumed to be) true. We call the latter aspect “semantic verification”. A major problem regarding semantic verification is how to automate it. This problem is rooted in natural language being used for labeling model elements, thus introducing terminological problems such as ambiguity (homonyms, synonyms) and other linguistic phenomena. Model creators and readers do not necessarily share the same understanding as the concepts they use are usually not documented and mix both discipline-specific terminology and informal, ordinary language. Therefore it is hard for humans to judge if a model is semantically correct and almost impossible for machines (apart from using heuristics) because the model element labels are not backed with machine processable semantics. The result is that the machine cannot interpret the contents of model elements, i.e. what is “inside the box” (rectangle, shape). Our solution approach is to encode the model element semantics in a precise, machine readable form using ontologies. Further, we then use rules to encode constraints used for verifying aspects of semantic correctness.

1.2 Prospects of Semantic Verification

The proposed approach of semantic verification allows performing additional checks on process models. Such checks are possible by annotating process models with instances of a formal ontology containing terminological knowledge of the domain under consideration. The ontology in conjunction with an inference engine can then be used to automatically verify several aspects of models based on the semantics of the individual model elements. This decoupling from human labor makes semantic verification scalable even in incremental approaches to model construction where a model has to be re-verified repeatedly. An important additional benefit thereby is that the semantic verification rules can be formalized on a more abstract and generic level and the inference engine interprets them with the help of both explicitly encoded and inferred knowledge from the ontology. Therefore, it is possible to formulate semantic verification rules in a more natural and understandable way that accommodates to the nature of generic rules such as guidelines, best practices, conventions, recommendations or laws being rather abstract in order to ensure broad applicability.

The paper is organized as follows. In section 2, we provide an overview of tools and approaches in the state-of-the-art of model validation and verification. In section 3, we present a case study that motivates our approach. We present our approach of semantic verification, a rule classification and examples illustrating the application of such rules to the real-world problems of the case study in section 4. In section 5, we describe the limitations of semantic verification and in section 6, we look at future research.

2 State-of-the-Art

The verification of models has focused mainly the syntax and formal semantics so far. In this sense, verification abstracts from the individual semantics of model elements which is given by natural language and concentrates on formal procedures. Such procedures partly originate from software engineering [Gr91] where they are discussed under the terms “model checking” and “theorem proving” [ChBr08]. These approaches concern dynamic aspects of model execution which are verified using finite state automata (FSM). In the area of process modeling, independent formal criteria have been developed such as „soundness“, „relaxed soundness“ or „well-structuredness“ which are used to detect shortcomings such as deadlocks, missing synchronizations and other defects regarding the formal semantics [Me09]. These criteria clearly go beyond merely checking the conformance of a model to its Meta model or grammar of the modeling language. There are some tools supporting these verifications such as the bflow* toolbox (www.bflow.org) or the EPC-Tools ([wwwcs.uni-paderborn.de/cs/kindler/research/EPC Tools](http://wwwcs.uni-paderborn.de/cs/kindler/research/EPC%20Tools)). Research concerning formal verification is still an active field; new approaches consider e.g. the verification of access constraints in semi formal models [WMM09], verification in the context of hierarchical models [SCA07] or workflows [TBB08].

However, a major problem still is that verification rules are exposed to frequent changes due to the dynamics of the contemporary legal and economic world. Some efforts address this problem area of rule dynamics and suggest graphical modeling languages such as BPSL (Business Property Specification Language) [LMX07] or suggest capturing the required rules implicitly by providing negative examples [SiMe06] or by patterns [SPH04]. Nonetheless, a fundamental problem is still, that most approaches require a rather fine grained specification of rules conflicting with the rather abstract nature of rules required for semantic verification in the sense of guidelines, best practices or general principles. First approaches in this direction therefore explore the use of rules together with semantic process descriptions [ThFe09] and describe frameworks for semantic verification related to compliance [ElSt08]. These approaches therefore rely on more formally defined semantics in comparison to e.g. glossaries or technical term models [KuRo98].

We extend the state-of-the-art by showing that ontology-based representations of process models enable the formulation of generic verification rules which are then applied to concrete process models using an inference engine in order to automate semantic verification. We apply our approach to real-world problems and therefore demonstrate that semantic verification is not only feasible, but also proves to be useful for solving real-life problems.

3 Case study

The municipality we took as our case is one of the biggest cities in the country we accomplished our research (region capital city). It has about 580,000 inhabitants and the public administrative authorities are employing about 9,100 employees, distributed over about 440 administration buildings. The structure is decentralized and subdivided into seven departments, each with 48 assigned offices and institutes. Based on Fat Client Server architecture, the 6,000 IT-jobs are workplace-based and completely linked to each other via a communication system throughout the city. In view of the increasing international competition, the city is requested to rearrange its product and process organization, particularly as the support of enterprise-related activities becomes increasingly an essential position factor in the international competition. In the city, about 99% of the enterprises have less than 500 employees and can be considered as small or medium-sized enterprises, these are about 40,000 enterprises.

The strategic goal of the city is to make the place even more attractive for enterprises in terms of their competitiveness with a long-lasting effect. This shall be achieved by making the enterprise-related offers and services of the city even easier to access for enterprises, in terms of a One-Stop eGovernment. To reach this goal, the city has to model about 550 enterprise-related administrative processes. The process setting is highly relevant for the capital city, because several of the procedures are used about 15,000 to 25,000 times per year by the companies. After having started the project we detected several inconsistencies in the collected data. Subsequently, we describe the modeling problems that we lay open.

3.1 Terminological problems

- (T1) Due to the fact that laws and regulations are regularly made by jurists and not by IT-experts, terms and facts of cases are often differently named although the meaning of two terms is the same. For example, the terms “admission” and “permission” were found in 334 administrative process models, but the terms always had the same meaning and the same process-related consequence.
- (T2) Another terminology problem occurs concerning the fact that in the municipality we have examined no rules were arranged to allow only one preferred term for one correspondent meaning. For example, some modelers (14) used “address” and some (8) “mailing address”, or modeler used abbreviations, like “doc” instead of “document”.

So, there is a lack of terminological modeling rules. These terminology problems hindered the identification, comparison and further use of the administrative process models (e.g. in process automation) in the city we focused on.

3.2 Verification problems

The administrative process models had also several errors regarding the correct sequence processing. Subsequently, we show the core modeling errors of process sequence conflicts (V1-V4) and in process sequence conformance (V5):

- (V1) In 64 process models, the event “admission free of charge” was followed by the (wrong!) function “start payment process”.
- (V2) As part of a preliminary check, which is executed in every application process at the beginning, the civil servants check the completeness of the submitted documents. In 41 of these process models, we found after the event “documents uncompleted” the (wrong!) event “preliminary check complete”, although documents were still missing.
- (V3) In 32 process models we found after the event “application is not licensable” the (wrong!) function “send admission”.
- (V4) The next step after the preliminary check is an in-depth check of the admission case. This row is strictly followed. But in 13 of the administrative process models, we found the two checks reversed.

So, there is a lack of element flow rules like: *After X must (must not) follow Y.*

- (V5) If a process contains the event “procedure is billable”, the same process must also contain a function “calculate charge”. But in 21 of the relevant process models, no such function was found.

So, there is a lack of element occurrence rules like: *If a process contains X, the process must (must not) contain Y.*

4 Ontology-driven approach for semantic verification

4.1 Ontology-based representation of process models

A first step towards semantic verification of semiformal process models is the representation of the process models using a formal ontology language such as the Web Ontology Language (OWL) standardized by the World Wide Web Consortium (w3.org/2004/OWL). We use this ontology language, as it has gained a broad acceptance both inside and outside the AI and Semantic Web community. The use of the ontology-based representation is twofold. On the one hand, it allows the connection of process models with domain knowledge in order to improve the interpretation and derive new facts not explicitly specified by the modeler but relevant for verification. On the other hand, it provides for a machine processable representation enabling the automation of such derivations and therefore using logic and reasoning to automate verification tasks. The ontology-based representation of process models consists of creating a model representation in the ontology (step 1) and the annotation of domain knowledge to that representation (step 2) (cf. Fig. 1) which are described subsequently.

The creation of a process model representation in the ontology is done by considering its graph structure. For each node, an instance is created in the ontology and for each arc, a property is created in the ontology connecting the two nodes which are at the end of the arc. This step can be executed automatically using the capabilities of a transformation language such as XSLT. The instances created in the ontology are instances of the classes shown in the left part of Fig. 1 which reflects the well-known Workflow Patterns. The properties having their domain and range on the `p:ProcessGraphNodeClass` are used to represent direct connections between model elements (property `p:connectsTo` being a sub-property of `p:flow`) as well as the set of following elements which can be reached without traversing an exclusive decision point such as an XOR-Gate (transitive property `p:followedBy`) or which can be reached by an arbitrary path along the flow in the process model (transitive property `p:flow`). We use the namespace-prefix `p:` for indicating the process space in general and `ex:` for indicating example data that is strongly intertwined with the concrete process fragment being used for illustrative purposes. Due to space limitations, we have omitted the translation of BPMN-lanes into organizational units in the ontology which can be represented by properties `p:assignedTo` which are added to each node in a lane. Currently, we also omit pools for the sake of simplicity.

The annotation of the process model representation with domain knowledge via the `p:equivalentTo`-properties shown in the right part of Fig. 1 provides for the semantic specification of the model elements with machine processable semantics. Domain ontologies can be built by leveraging existing ontologies (cf. section on the state-of-the-art), using reference models, ontologizing industrial standards or extracting structures out of IT-systems such as database schemas. Also, top-level or upper ontologies may be used as a basic backbone structure that helps bootstrap ontology development and reaching ontological commitment on how to think about the world in the sense of a shared “contract” between the different involved stakeholders. In the example of Fig. 1, we have used the SUMO-ontology as a backbone structure providing basic distinctions such as between abstract and physical entities forming the basis of the subsumption hierarchy. This hierarchy not only serves for disambiguation purposes (e.g. `Service` as subclass of `ComputerProcess` vs. subclass of `Product`). It also provides for the specification of semantic verification rules on varying levels of generality. This enables the specification of rather generic verification rules such as guidelines and best practices and letting an inference engine do the work of verifying whether a specific model is compliant or not. So, for example, a government agency could have the guideline that immediate feedback should be given on each application. If a process model starts with a citizen having filed her tax return and contains an activity “send feedback via e-mail”, then the inference machine can prove that this process complies with the guideline as “tax return” is subsumed by “application” and “feedback via e-mail” is subsumed by “feedback”.

Beyond such simple subsumption reasoning, an inference engine can also be used to automatically derive more complex conclusions. Automatic classification of instances for example could be achieved by using class expressions composed of intersection, union and complement which are available in OWL and which rely on propositional logic. Also, automatic classification can leverage existential restrictions of properties on classes as well as restrictions on their domain and ranges thus relying on a fragment of

first order logic. Moreover, OWL and most of the current ontology languages also provide for specific characteristics of properties such as symmetry, transitivity, reflexivity etc. leading to additional conclusions in regard to the structure of a process graph represented in the ontology. While we use ontology for both, representing a process graph and inferring new facts about it, we use rules to express constraints for verification. Before we show the application of such rules to solve the case problems described in section 3, we will introduce them in the next section.

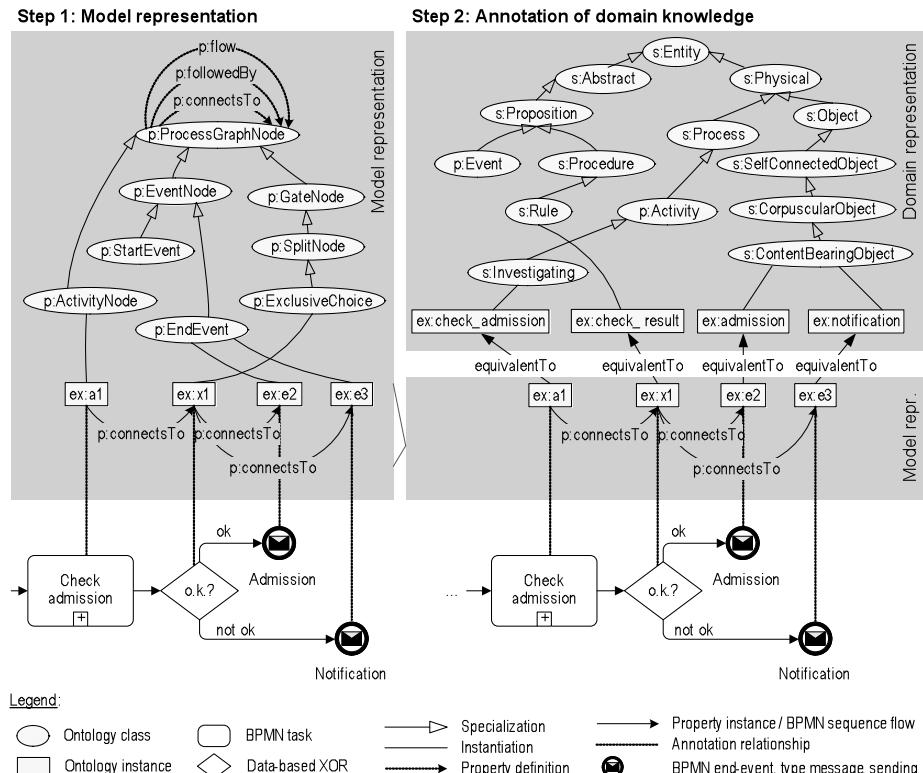


Fig. 1: Ontology-based representation of process models

4.2 Semantic verification rules

According to the IEEE 1012-1998 definition [IEEE98] “verification” means to check whether an artifact and/or its creation comply to a set of given requirements hence focusing on artifact-internal aspects. This understanding of verification is in contrast to validation which means ensuring that an artifact is eligible for the intended purpose [De02] thus focusing on artifact-external aspects and human judgment and experience. Intuitively, verification of process models is more amenable to machine processing than validation, given the fact that the model elements are annotated with machine processable

semantics and the verification rules are formalized. Due to the fact that our approach suggests also using terminological and domain knowledge (encoded in ontologies) for verification, we call our verification rules “semantic verification”. If the focus of a rule is the structure of a process (e.g. the sequence of actions), then we call such a rule *element flow rule*. If the focus of the rule is the occurrence of model elements in arbitrary positions in the model, then we call such a rule *element occurrence rule*. These basic rule types may be mixed in practical applications, such that any combination of two types may be combined to a single rule. For example, if an organizational unit “government representative” is present anywhere in the process (element occurrence rule), then an additional sequence of activities such as “report results to head of administration” has to be performed (element flow rule) involving at least one information system for archiving the results.

4.3 Application to the case problems

In this section, we provide practical examples for each of the basic semantic verification rule types introduced in the previous section illustrating how our approach of semantic verification can be applied to the case problems given in section 3. Fig. 2 illustrates the application of an element flow rule (on the left side) and an element occurrence rule (on the right side). At the bottom layer, fragments of a process described by using BPMN are displayed. Model elements targeted by the verification rules are highlighted (dark-red filling with white labels). Above the model layer, the ontology is displayed consisting of a model representation part and a domain representation part. The semantic verification rules using the classes and instances of the ontology are displayed above the ontology. The rules are displayed in an informal notation with variables prefixed by question marks, class memberships written as functions with one argument and predicates (properties in the OWL-terminology and edges in the graph-terminology) as functions with two arguments. To improve comprehensibility, the rules have additionally been paraphrased using natural language at the topmost layer.

Regarding rules, there are a number of non-web-based ontology languages, such as OCML and Ontolingua, which make it possible to formulate rules without an extension. The ontology language OWL, used in this article, only supports the formulation of rules via extensions (apart from simple property chains in OWL 2.0). Such an extension is the Semantic Web Rule Language (SWRL) [HPB04] which extends OWL with IF-THEN-rules in the form of a logical implication. The rules presented in the examples are of this nature and can be formalized using SWRL. They have the general form of antecedent \rightarrow consequent – i.e. if the antecedent (body) of the rule is true, then the consequent (head) must also be true. Since the consequent consists of error messages, it will not be true in a literal sense, it rather will be generated if the antecedent matches and the rule is executed (fired).

In the following, we elaborate on some of the abstractions and inferences possible by using terminological and domain knowledge. They are an important merit of our approach as they provide for the formulation of rather generic semantic verification rules applicable to concrete models by automated machine reasoning:

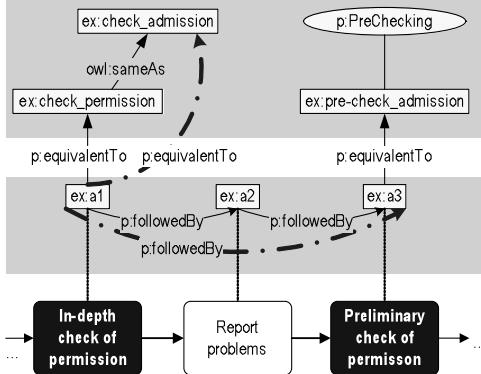
- **Element flow rule:** Terminological knowledge is used in stating that `ex:check_permission` is the same as `ex:check_admission`. Hence, using this terminological knowledge, the `p:equivalentTo` property between `ex:a1` and `ex:check_admission` can be inferred. Moreover, as `p:followedBy` is a transitive property, the triple `ex:a1 p:followedBy ex:a3` can be inferred. As `ex:a3` is annotated with an ontology instance that belongs to the class of `p:PreChecking` activities, the antecedent of the rule is satisfied and the rule fires.
- **Element occurrence rule:** The example makes use of a class definition by enumeration resulting in `ex:receipt_child_benefit_app` being classified as an individual of `p:UnbillableProcStartEvent`. The rule fires because there is another node in the process that is annotated with an individual belonging to the class `p:FeeCalculation`. Obviously, the rule is specified rather generic and will fire if two nodes are annotated with instances classified as members of the two classes `p:UnbillableProcStartEvent` and `p:FeeCalculation`.

Element flow rule

Example: A preliminary check of a permission must not follow after an in-depth check.

```

p:equivalentTo(?node1, ex:check_admission)
^ p:followedBy(?node1, ?node2)
^ p:equivalentTo(?node2, ?activity)
^ p:PreChecking(?activity) => error!
  
```



Key – addition to figure 1

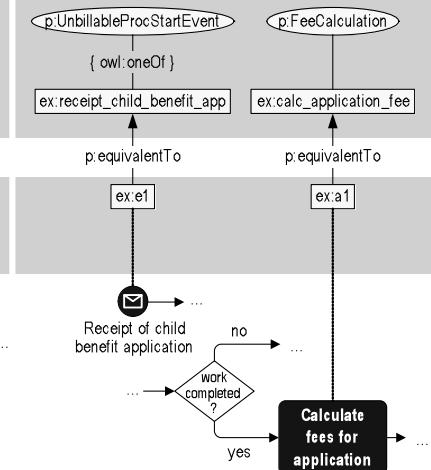
—{ owl:oneOf }— Class membership by enumeration
— — — Inferred property

Element occurrence rule

Example: If a process contains a start event indicating that it's free of charge, then no fee calculation can occur in this process.

```

p:equivalentTo(?node1, ?event) ^
p:UnbillableProcStartEvent(?event) ^
p:equivalentTo(?node2, ?activity) ^
p:FeeCalculation(?activity) => error!
  
```



Rules

Domain represent.

Model represent.

Fig. 2: Element flow rule and element occurrence rule

The examples presented to exemplify the rule types have in common, that they use facts that are explicitly known (either declared or inferred). The general pattern of this is `a ^ b -> error`. However, both rules can also be modified to the form of `a ^ ~b -> error`, i.e.

if some facts a (fragments of a process graph) are known, some other facts b (again fragments of a process graph) should not be present in the knowledge base and the failure to derive them should be treated as a form of (weak) negation. This implies closed world reasoning (as opposed to open world reasoning) and negation as failure (NAF). Ontologies in the Semantic Web adhere to the open world assumption (OWA), which makes sense in an open and networked environment such as the web.

According to the OWA, facts that are not explicitly stated in the knowledge base are not false but instead unknown or undefined. In contrast to that, to verify process models it would be useful to at least temporarily assume to know all the facts and hence switch to closed world reasoning. This sort of reasoning requires negation as failure (NAF) and can be introduced using the Jena built-in rule engine ARQ (jena.sourceforge.net/ARQ/) which provides this feature using procedural built-in primitives which can be invoked by the rules. Each primitive is implemented by a Java object and additional primitives can be created and registered by the user. To achieve closed world reasoning using NAF, the primitive `noValue(?subject, ?predicate, ?object)` can be embedded in a rule which will cause a rule to fire if no matching triple can be found. With closed world reasoning, semantic verification rules such as the following examples would be possible:

- *Element flow rule*: If there is a preliminary check, the in-depth check always has to be performed afterwards.
- *Element occurrence rule*: If the process starts with an event indicating that this process is billable, then somewhere in the process there must be an activity “calculate fee”.

Furthermore, tools such as Jena or the SQWRL query language implemented in the Protégé-editor also provide built-ins for counting, geo-related reasoning and many other possibilities which enhance the power of semantic verification rules.

4 Limitations of semantic verification

Clearly, semantic verification rules have some limitations. To begin with, they should not be regarded as a surrogate for verifications related to the meta-model or the grammar of the used modeling language. They are rather complementary to such verifications and correct models form the basis for additional semantic verification checks. Also, aspects regarding the execution semantics of models such as soundness, relaxed soundness etc. dealing mainly with the absence of deadlocks and livelocks are not covered by our approach due to its complimentary nature.

Further limitations of semantic verification rules are that they depend on the availability of an ontology and the annotation of process models. While in other areas such as the life sciences huge ontologies have been developed and standardized, the field of administration still lacks authorities who develop and standardize ontologies. However, this problem may partly disappear if the terminology problem will be solved, e.g. by defining structured vocabularies which bootstrap the development of ontologies. Also, current tools for process model annotation are mostly in the state of research prototypes. In par-

ticular, functionalities for semi-automated annotations and annotation suggestions based e.g. on annotations previously made in the current model or the whole model repository etc. have to be developed in the future in order to enable comfortable and cost-effective semantic verification. Also, aspects of ontology management have to be considered since ontologies are not automatically a shared knowledge representation. A major limitation of the current approach is that it is agnostic to the control flow of process models. At the moment, the only exception of that is the property `p:followedBy` connecting only nodes which form a sequence when the model will be executed and so it provides for rules such as “b should not be executed after a”.

5 Conclusion and further Research

The approach presented in this paper showed how to use ontologies, rules and reasoning for the semantic verification of process models. Future versions of our approach will tackle some of the described limitations. As a next step, we plan to integrate a further pre-processing step which will mark the nodes in the graph according to their succession of logical connectors such as AND, XOR and OR. The capturing of information on such local contexts of parallelism or exclusivities to the ontology based representation of process models will allow advanced semantic verification rules such as “resource x must not be used in parallel branches” or “activity x and activity y should always be executed exclusively”.

References

- [ChBr08] Chapurlat, V., Braesch, C. (2008): Verification, validation, qualification and certification of enterprise models: Statements and opportunities. In: Computers in Industry 59, Nr. 7, pp. 711–721
- [De02] Desel, J. (2002): Model Validation - A Theoretical Issue? In: Esparza, J., Lakos, C. (Eds.): Application and Theory of Petri Nets 2002: 23rd International Conference, ICATPN 2002, Adelaide, Australia, June 24–30, 2002, Proceedings. Springer, Berlin, (2002), p. 23–43
- [ElSt08] El Kharbili, M., Stein, S. (2008): Policy-Based Semantic Compliance Checking for Business Process Management. In: Loos, P., Nüttgens, M., Turowski, K., Werth, D. (Eds.): Modellierung betrieblicher Informationssysteme (MobIS 2008) - Modellierung zwischen SOA und Compliance Management. 420. RWTH Aachen (CEUR Workshop Proceedings), pp. 165–177
- [Gr91] Gruhn, V. (1991): Validation and verification of software process models. In: Proceedings of the European Symposium on Software Development Environments and CASE Technology. Berlin, Springer
- [HPB04] Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosof, B., Dean, M. (2004): SWRL: A Semantic Web Rule Language: Combining OWL and RuleML, W3C Member Submission 21 May 2004. W3C. <http://www.w3.org/Submission/SWRL/>. Accessed at 2010-05-01
- [IEE98] IEEE 1012-1998. http://www.techstreet.com/standards/IEEE/1012_1998?product_id=31920. Accessed at 2010-05-01

- [KuRo98] Kugeler, M.; Rosemann, M. (1998): Fachbegriffsmodellierung für betriebliche Informationssysteme und zur Unterstützung der Unternehmenskommunikation. In: Fachausschuss 5.2 der Gesellschaft für Informatik e. V. (GI) (Ed.): Informationssystem-Architekturen, 5, pp. 8–15
- [LMX07] Liu, Y., Müller, S., Xu, K. (2007): A static compliance-checking framework for business process models. In: IBM Systems Journal 46, Nr. 2, pp. 335–361
- [Me05] Melenovsky, M.J. (2005): Business Process Management's Success Hinges on Business-Led Initiatives. Stamford, CT: Gartner Research
- [Me09] Mendling, J. (2009): Empirical Studies in Process Model Verification. In: Jensen, K., van der Aalst, W. M. P. (Eds.): Transactions on Petri Nets and Other Models of Concurrency II. 5460. Berlin, Springer, pp. 208–224
- [MeAa07] Mendling, J., van der Aalst, W.M.P. (2007): Formalization and Verification of EPCs with OR-Joins Based on State and Context. In: Krogstie, J., Opdahl, A.L., Sindre, G. (Eds.): Proc. of the 19th International Conference on Advanced Information Systems Engineering (CAiSE 2007), 11–15 June 2007, Trondheim, Norway. Berlin: Springer (LNCS), pp. 439–453
- [PeWe09] Peters, N., Weidlich, M. (2009): Using Glossaries to Enhance the Label Quality in Business Process Models. In: Nüttgens, M., Rump, F. J., Mendling, J., Gehrke, N. (Eds.): 8. Workshop der Gesellschaft für Informatik e.V. (GI) und Treffen ihres Arbeitskreises "Geschäftsprozessmanagement mit Ereignisgesteuerten Prozessketten (WI-EPK)" Berlin, 26.11–27.11.2009, pp. 75–90
- [SiMe06] Simon, C., Mendling, J. (2006): Verification of Forbidden Behavior in EPCs. In: Mayr, H. C., Breu, R. (Eds.): Proceedings of the GI Conference Modellierung (MOD2006), March, 22 - 24, 2006, Innsbruck, Austria, pp. 233–242
- [SCA07] Salomie, I., Cioara, T., Anghel, I., Dinsoreanu, M., Salomie, T. I. (2007): A Layered Workflow Model Enhanced with Process Algebra Verification for Industrial Processes. In: Proceedings of the 2007 IEEE International Conference on Intelligent Computer Communication and Processing, September 6-8, Cluj-Napoca, Romania, pp. 185–191
- [SPH04] Speck, A., Pulvermüller, E., Heuzeroth, D. (2004): Validation of business process models. In: Proceedings of the 17th European Conference on Object-oriented Programming (ECOOP), July 21-25 2003, Darmstadt, Germany. Berlin et al., Springer
- [ThFe09] Thomas, O., Fellmann, M.: (2009) Semantische Prozessmodellierung – Konzeption und informationstechnische Unterstützung einer ontologiebasierten Repräsentation von Geschäftsprozessen. In: Wirtschaftsinformatik 51, Nr. 6, 2009, pp. 506–518
- [TBB08] Touré, F., Baïna, K., Benali, K. (2008): An Efficient Algorithm for Workflow Graph Structural Verification. In: On the Move to Meaningful Internet Systems: OTM 2008: Proceedings of the OTM 2008 Confederated International Conferences, CoopIS, DOA, GADA, IS, and ODBASE 2008. Part I. Berlin, Springer, pp. 392–408
- [WeLa08] Wecker, G., van Laak, H. (2008) (Eds.): Compliance in der Unternehmenspraxis: Grundlagen, Organisation und Umsetzung. Wiesbaden : Gabler
- [WMM09] Wolter, C., Miseldine, P., Meinel, C. (2009): Verification of Business Process Entailment Constraints Using SPIN. In: Proceedings of the 1st International Symposium on Engineering Secure Software and Systems. Berlin, Springer, pp. 1–15

Context-based Modeling: Introducing a Novel Modeling Approach

Martin Juhrisch¹, Gunnar Dietz²

¹Technische Universität Dresden
Lehrstuhl für Wirtschaftsinformatik, insb. Systementwicklung
martin.juhrisch@tu-dresden.de

²Dongbei University of Finance and Economics
International Education Centre
mail@gditz.de

Abstract: Despite the fact that researchers agree on the importance of enterprise models to an organization's success, the knowledge about how to handle problems where models have to be compared or integrated is still fuzzy and vague, and there is little agreement regarding compositional facets. Highly interesting is the interaction between models in shared modeling projects — e. g. between requirement models and service implementations in a service-oriented architecture (SOA). This article highlights an approach that allows to prevent integration conflicts in conceptual models already during the modeling phase. The influence of this approach on conceptual modeling and its use in intra-organizational collaborations is investigated. We show the inherent complexities of model-mediated interactions between domain experts and IT-service developers. It is suggested that at an early stage of the modeling process the use of guidelines has an substantial benefit for avoiding integration conflicts in conceptual models. Furthermore, due to the way how the approach bridges the semantic gap, changes of business requirements as well as technical implementation restrictions influence each other. This results in an ongoing system development process that can be interpreted as a permanent management of application systems. Our results contribute to model-based management theories that have so far neglected the distributed construction of conceptual models.

1 Introduction

In [23] a new approach was introduced, called the Description Kit Approach (DKA), that allows the development and configuration of service compositions. One focus of that article was the development of the inter-level architecture of the DKA that can't be established using classical methods of meta-modeling. Even if the modeling of Description Kits in principle quite similar as the creation of meta-models, and therefore fits technically into the field of meta-modeling, the “philosophical” background is quite different. In contrast to meta-modeling the goal is not the creation of new modeling languages.

One new aspect of the DKA is the embedding of guidelines that at least influence the pro-

cedure model of the modeling method. Therefore guidelines — restrictions that the DKA establishes for the modeling methods — an influence on the ordinary modeling process by influencing or even determining the procedure model. Since the DKA is a generic approach, it can be even used as a method for (indirectly) creating procedure models. These result in annotated models that used domain spanning concepts, which again allow an automated evaluation. A second aspect is the inclusion of the adaption process of the domain specific modeling language into the problem solving process by weakening the barrier between language creation and language use. This is of great importance, since the guidelines result from a labile consensus, which simplifies the model comparability and only by this allows the establishment of problem solving methods.

Another focus was the description of the different model types that can be used and the derivation of language concepts for each of the three modeling levels. The abstract syntax has been discussed, since this is the foundation for an automated model use. The fundamental notation of the elements has been described and shown in examples. A problem that should be solved in a concrete situation can be translated for model-based problem solving methods into the combination or comparison of different domains. Examples are here analysis models and design models, or to compare actual (as-is) domains with target domains, or to combine or compare models from a distributed modeling project. The model for describing guidelines — the description kit language (DKL) — is generic, so that each way of describing a certain situation can be restricted relatedly to the aim of solving the concrete problem. Therefore a general methodology has been created that simultaneously meets the requirements of a certain domain.

2 Theoretical Background

2.1 Method Engineering

The following chapter explains some fundamental definitions from the areas of the present area of research. Starting with the dimensions of a language, classes of languages are discussed. This discussion is necessary, since languages plays a major role when reformulating conceptual content with the intention of creating service compositions.

2.1.1 Classification of Language

A language that is characterized by a precise syntax, a formal semantic and a well-defined usage of the linguistic terms is called a formal language [10, pp. 9]. A precise syntax only allows a limited set of valid and well-formed terms. Semantically formal means that each linguistic term has a unique interpretation [10, pp. 9]. Therefore also the usage of that term is well-defined.

Natural languages are the opposite of formal languages. Their linguistic terms come from a real-world (material) domain, and their syntax and semantic have been evolved historically within a linguistic community [28, pp. 56]. Typically for natural languages is the diversity

of linguistic terms, which results in a high semantic power. However, this also results in several linguistic defects due to the contrariness of their lexicographical and structural possibilities [33], [32].

All previous attempts to describe a material semantic in the sense of a final set of syntactical rules in a complete formal way have been fruitless [37, pp. 97]. Therefore it remains a big challenge, to formally specify a material domain with sufficient preciseness, such that models within that domain can be used for an automatic transformation [35, p. 6].

Another way of classifying languages is to differentiate the phases language creation and language use. Language creation means to the description of the syntax and semantic of a language, e.g. a modeling language, by referring to a meta-language. In this case one refers to the object language that is the language used to describe real-world phenomena within a certain domain, and the meta-language that describes the object language [11]. If the meta-language is again a modeling language, one refers to a meta-modeling language [11, p. 696]. The term language use refers to the usage or application of the object language to describe a certain domain. One application field of Business Informatics (german “Wirtschaftsinformatik”) research that derives from distinguishing object languages and meta-languages is the development of methodologies.

2.1.2 Conceptual Modeling Languages

Since formal languages are not appropriate for complete descriptions of a material semantic, their applicability for analyzing social systems is doubted [10, p. 10], [31, pp. 26]. Semiformal languages have been established in the field of systems development for an understanding of material domains by modeling problems that are not well-structured, cannot be structured objectively, but that are bound to subjects and goals [17, pp. 7], [18, p. 125]. Their linguistic instruments has to cover informal aspects that allow a deep understanding of the operational domain and its potential for reorganization by information and communication technology (ICT). Furthermore, the implementation of future information systems has to be prepared by aspects of formal language [11, p. 696]. In this context one speaks of conceptual modeling languages. A dominant property of these languages is a combination of a formally defined syntax and an extensively material semantic of the linguistic terms [34, p. 44], [12, p. 34]. The semantic is normally derived from a concept of a technical language. This is done by annotating linguistic terms by natural language terms. Typically the terms of the application domain are not contained in the language, for which a grammar is described with the help of a meta-language [26, p. 53]. A conceptual modeling language only comes to live by combining the modeling grammar and a set of technical linguistic terms. Conceptual modeling grammars have a high relevance within the communication process between the individuals involved in the analysis phase of system development [48, pp. 104]. By combining a technical language with a modeling grammars one gets a conceptual modeling language, which can be used to create conceptual models, see Figure 1.

In the following we assume that modeling grammars always base on a diagrammatic notation, whose application results in graphical representations [40, pp. 159], [15, pp. 510], [11, p. 696], [47, p. 363], [12, p. 84].

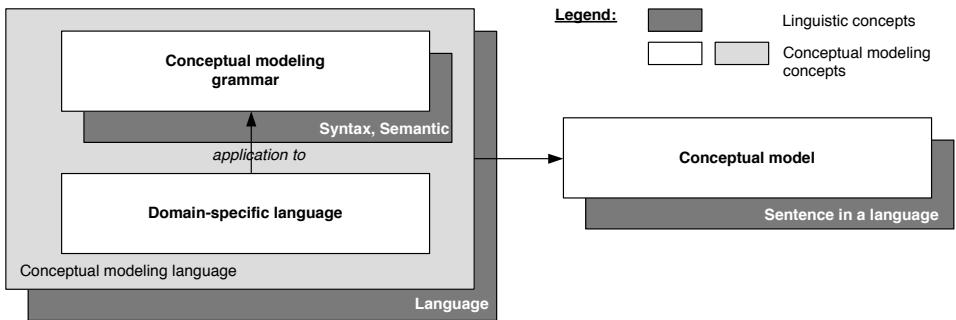


Figure 1: Conceptual modeling

One primary goal of business informatics (“Wirtschaftsinformatik”) is the “... analysis, development and use of information systems” [49, p. 2]. An information system (IS) is a system that can process information, where “process” means collect, transfer, transform, save and supply [9, p. 65]. The utilization of models as an instrument for handling complexity by abstraction allows a well-planned procedure when designing information systems, if one succeeds in systemizing the analytic and creational character of the modeling [9, p. 119], [3, pp. 27], [36, p. 63].

For this reason, methods on the base of using models have been established [43, p. 33], [50, p. 371], [39, p. 27] — so-called model-based methods. In the context of information systems design a method is normally understood as a specification in accordance to aims and available resources, which describes a well-planned procedure for the solution of a certain type of problem [5, p. 5], [27, p. 876], [41, p. 239], [51, p. 34], [7, p. 36], [13, p. 32]. The aim is to successfully achieve the required goals. The method should be inter-subjectively comprehensible, which means it should systematically lead to the solution of a problem [51, p. 34]. Methods have to be designed for the problem solution [13, p. 33]. To derive concrete tasks and stakeholders, the method must declare certain elements as obligatory and restrict the users’ freedom of decisions in this area, e.g. by giving detailed demands and specifications [43, p. 33], [13, p. 34]. On the other hand, a method can also leave a certain freedom for deciding how to use the method.

GUTZWILLER analyzes approaches from information systems research in the domain of method development and approaches for a standardization in this area and extracts some commonly used concepts [16, pp. 15]. From this he derives general method components [16, p. 12]: The procedure for the problem solution is determined by a series of activities, whose course of action has to be described by the procedure model. Within each activity some method products (results) are generated or used, e.g. documents and artifacts. How to perform an activity is determined by techniques. Roles describe stakeholders who use these techniques to generate results. A meta-model is specified as a conceptual data model for all method products. A modeling method contains rules that describes how to use the linguistic concepts of a modeling language and in which order to do that [39, p. 27] (and similar [50, p. 371], [43, p. 33]). Furthermore the modeling method contains minimal

requirements for the models to create. If one transfers GUTZWILLER's method components to modeling methods, the following components are essential: The meta-model of the method, the activities within the modeling process and the result of the model creation. A focus clearly is on the meta-model for the modeling grammar [25, p. 377]. If modeling methods again are developed in a model based way, one gets models for the linguistic concepts and models for the procedure of the model creation.

Language Model: Conceptual data model of the method [16, pp. 12–14].

Procedure Model: Model of the activities of the model creation [21, p. 116].

2.2 Service-oriented System Design

For the development of a process oriented information architecture the paradigm of service orientation has been established. The literature contains several variants of the definition of a service-oriented architecture (SOA). All these definition have in common that a SOA describes a system architecture that offers business services in combination with the application system functionality in the form of (electronic) services [8, p. 3 and p. 50].

In contrast to component orientation the service orientation underlines a high autonomy of a service and the extent of the offered functionality [2, pp. 634]; [29, p. 19]. The conceptual change from objects or components to services is driven by the functional requirements to a service, its technical implementation, and its economic importance.

A service encapsulates a set of functions that encompass a closed (self-contained) process activity. The data a service processes corresponds to complete business objects.

In contrast, object orientation means to encapsulate data and functionality in the concept of a class [4, p. 156]. A component contains the functionality of several classes and therefore a larger amount of data and functions (or better said methods) [42, p. 153]. The functions, however, are still fine-granular. They serve the purpose of the integration of components into an application [19, p. 50]. An example for this are getter methods. They just return the value of an attribute of an object to another object that is associated to the first object. This functionality does not correspond to a process step in a business process, the exchanged data does not correspond to business objects, as they are depicted within the conceptual modeling of cross-application processes [19, p. 50]; [1, p. 24].

Services encapsulate functions that correspond to functional requirements [8, pp. 280]. A service interface has to handle complete business objects instead of single attributes [30, p. 50]. Therefore services have to adapt their granularity on the level of the application system to the process requirements [42, p. 153]; [1, p. 24].

Services and service users are developed independently and at different time. Any composition of service functions is done after developing the services first. Different to objects or components, services and service users are developed by organizationally distributed teams, at different times, and using different technologies [38, p. 45]; [19, p. 50]; [45, p. 16].

When considering reusability of services, the focus lies on a process oriented composition. In object orientation a system is composed into objects, which are created mirroring real world objects and concepts. In contrast, services are created by bundling tasks that follow common actual and formal goals and operate on common business objects. An example would be the service “Inventory” with functions like “ordering spare parts”, “check availability”, “reservation”, etc. The aim is a high cohesion of the functions within one service and a weak coupling between different services. The reuse of a service function and the creation of complex functions is done by composition [24, p. 327 and p. 74]; [52]. This distinguishes services from objects, where the reuse of classes is done on the level of the source code by inheritance [14, p. 32].

Services are, in contrast to objects, deployed. That means they are installed externally and made available externally. A service user binds a service during runtime [6, p. 10]. Service and service user are compiled independently. During runtime an application system constitutes of a certain set of service functions as a so-called Composite Application [19, p. 191]. This distinguishes services from components as well as classes, which are typically made available as libraries. A library must be integrated into a program and must be compiled together with the software. On the other hand, the communication between a service and the service user has to be stateless [8, p. 332]; [46, p. 62]). There is no mechanism for object state or life cycle management for services [19, p. 51]; [46, pp. 61].

3 Context-based Modeling

This section presents the procedure model of the DKA, which guides systematically through the modeling on every modeling layer and describes algorithms for the model comparison. In [22] the concept of the different modeling layers M^1, M^{*k}, M^0 have been described, as well as the language artifacts for the different layers and their relationships. These relationships include different kinds of instantiation relationships, which implicitly describe how to navigate between models within the method.

The DKA combines two different progress models: First for the methodology of the DKA itself, second for specific problem solutions done by using a model-based approach following the DKA. The following section describes the procedure how to put information methodically into models in the sense of the DKA (see Figure 2).

The method engineer has to develop a recommendation for a set of Description Kit Types (DescKitTypes). This has to be done in a consensus with the domain expert based on the concrete problem to be solved. In the progress model this happens directly after the definition of the meta-models for the modeling language(s) that should be used, i.e. as soon as the object language has been determined. For this the necessary modeling language has to be derived from the problem and the meta-model on M^1M has to be described. The development of a recommended set of DescKitTypes is a creative task for the method engineer that cannot be automated. Several DescKitTypes, however, may be fixed for certain scenarios. For example, in the case of service identification or composition the DescKitTypes *Interface*, *Input*, *Output*, *AttributeType*, *Attribute*, *Object*, and *Service* are

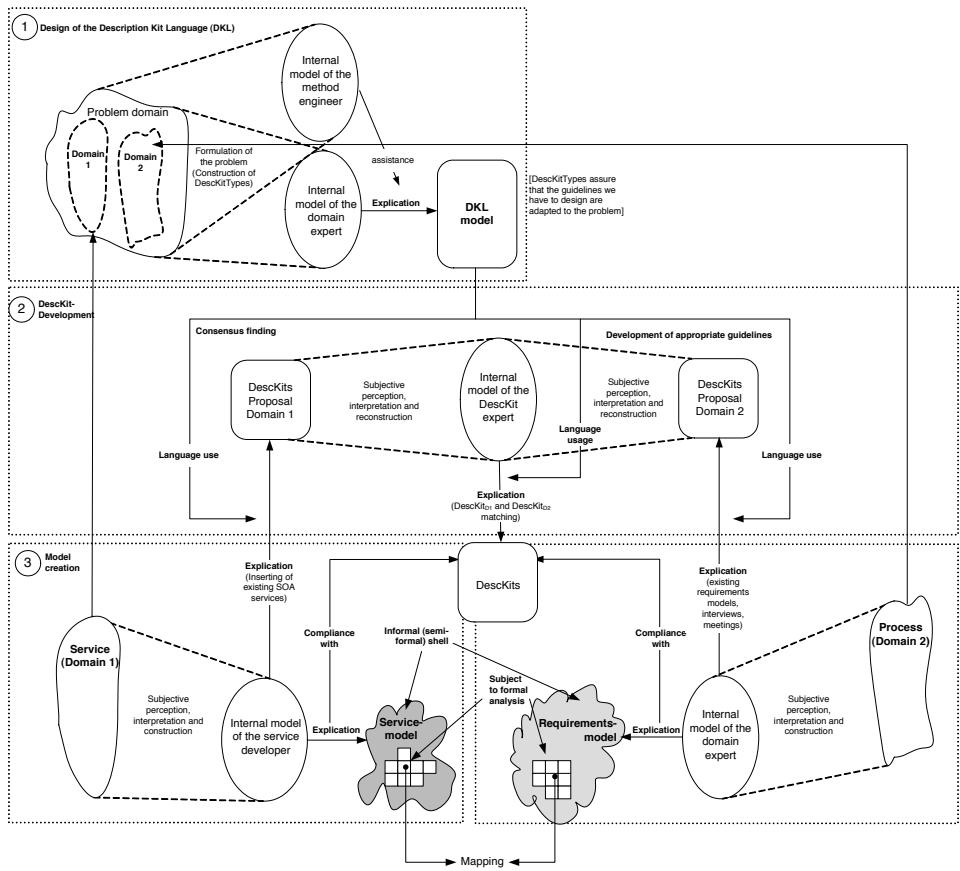


Figure 2: Role model of the DKA

predefined. They are the result of an analysis of the domain conflict [22]. Their aim is the description of objects and object states within analysis and service models. As a result of this activity one gets a set of DescKitTypes, that has to be used for the model creation in both domains simultaneously (see Figure 3).

The set of DescKitTypes and the set of Relation Types has to be added to the meta model layer of the DKL model. The creation of the DKL model includes the hierarchization of the different DescKitTypes and the definition how to use them. The latter means to create links from the DescKitTypes to the concepts of the modeling language. By this, the DKA establishes two generic ways for a restriction of the freedom of modeling:

- The first possibility is to integrate additional information into analysis models in a restricted way. For this a link between the meta-model of a conceptual modeling language (which has to be determined previously) and the meta-model for the DKL. This link determines, which DescKits are obligatory or optional for which model

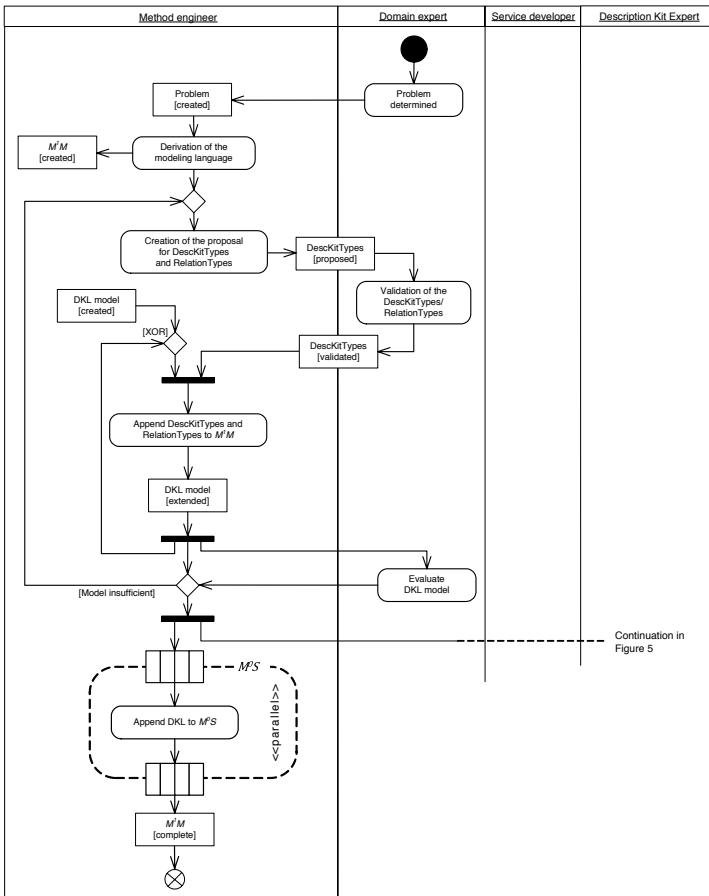


Figure 3: Procedure model for the model creation

elements.

- The second possibility is to combine several DescKitTypes to a complete modeling language. The model creation is then done by only using DescKits and Relations.

A restricted or enforced use of certain DescKits for certain modeling elements not only means a restriction of the modeling freedom with regards to the DescKits, but also a restriction of the (ordinary) modeling process itself. The meta-model of the DKA then is complete and allows the method engineer with help of the model of the DKL to ensure that the created guidelines are comply with the problem to be solved.

The tasks for the determination of DescKits on the layer M^{0*} and their instances in the models (Descriptions on the layer M^0) are done by different persons. The determinations on layer M^{0*} require consolidation processes that must involve both domain experts and

technical experts who want to model in a functional and technical way, respectively. The establishment of a consensus between those persons and the modeling of the result of the consensus are tasks that are not included in classical modeling projects. For the engineering of the DescKits therefore a new role *description kit expert* has to be established as a specialization of the role modeling expert. An approval process starts with the recommendation of a set of DescKitTypes by the description kit expert. Iteratively this recommendation has to be approved or declined by the domain experts and technical experts (service developers) and replaced by an improved recommendation, until a consensus has been found.

Domain experts and technical experts first have to have a rough idea about what the possible DescKits could be. That means the environment, in which one wants to create models, has to be clear. The group of domain experts then describes certain concepts that match the DescKitTypes and therefore are adapted to the problem to be solved. These concepts then have to be put into a hierarchy to derive DescKits. These DescKits then have to be created and added to the modeling layer M^{0*} .

If requirements models, e.g. conceptual data models, already exist, one can derive certain information from these to ease the above described process. Of course such an extraction cannot be done completely automatically. However, the results of the extraction represent a series of suggestions for the approval process. Figure 4 shows exemplarily the translation of a conceptual data model (here a UML class diagram) into suggestions for DescKitTypes and DescKits.

Note that suggestions for both DescKitTypes and DescKits that should be used for the DKL can be derived from a UML class diagram. Abstract classes e.g. are candidates for DescKitTypes. The same is true for “fundamental” classes, which however cannot be defined precisely. The number of parameters (class elements that represent primitive types) could give a hint for this decision. Other “not so fundamental” classes on the other hand could be candidates for DescKits. The description kit expert has to sort and rate all suggestions, but as a first suggestion this procedure could be very helpful.

Furthermore the group of service developers can input the existing SOA services into a service catalog. The WSDL descriptions include the ingoing and outgoing messages, which furthermore include a list of complex object types. These objects again represent candidates for concepts that may be represented by DescKits.

These two different ways to generate suggestions then have to be combined using best practices to come to common concepts. The result of the approval process should be then a consensus about which of these concepts should be represented by DescKitTypes and DescKits. With the help of the resulting DKL the description kit expert has then to design the DescKits on layer M^{0*} (see Figure 5). At this point these candidates for DescKits have to be put into a hierarchy that matches the hierarchy of DescKitTypes. This is done by a concretization on DescKit layer M^{0*} .

After the previous steps are done, the DescKits have to be implemented into the actual modeling project. The description kit expert again bears the responsibility for this. This is done iteratively accordingly to the procedure model: The first use of the DescKits to create Descriptions is done by persons, who also would perform the modeling within a

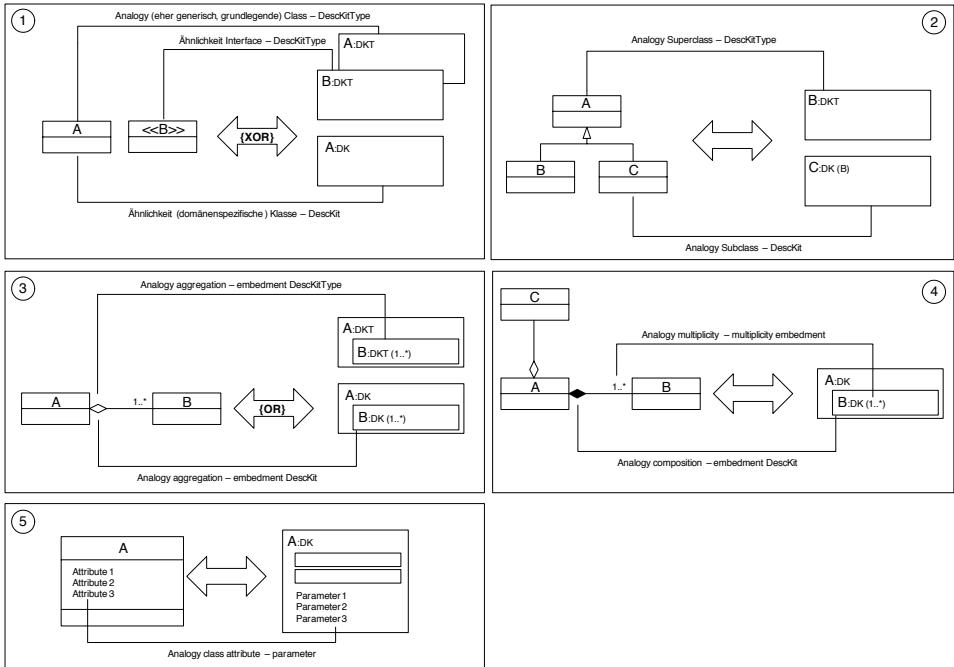


Figure 4: Transformation of a UML class diagram to DescKitTypes/DescKits

certain domain in classical modeling projects. These are the domain experts (perhaps with help of the modeling experts) and the service developers. A new aspect is, however, that the modeling process includes adaption processes: A DescKit can be modified during the modeling process.

On the meta-model layer M^1 no domain specific language constructs are modeled, but the language for the DescKits for the domain specific language constructs. Since the modeling layer M^{0*} is near to the layer M^0 , an iterative procedure allows to work on both layers alternately or even simultaneously. This means that the method supports an iterative adaptation process between layers M^{0*} and M^0 clearly and transparently and allows for example to “promote” modeling information from M^0 to M^{0*} .

At each time of the modeling on layer M^0 exactly on configuration of a DescKit of layer M^{0*} is active. Modifications of a DescKit, e.g. by defining new parameters, transfer a DescKit from one configuration to another. The adaptation operations should allow a modification that does not destroy the integrity of existing conceptual models. That means that a completely free modification of DescKits that have been used in models already is not possible without further ado. Not critical is always the addition of new elements, as far it is not obligatory. More difficult is the modification and deletion of elements. A change management is necessary for the DescKit, so that dependencies can be solved.

During the modeling process change requests may occur. These have to communicated

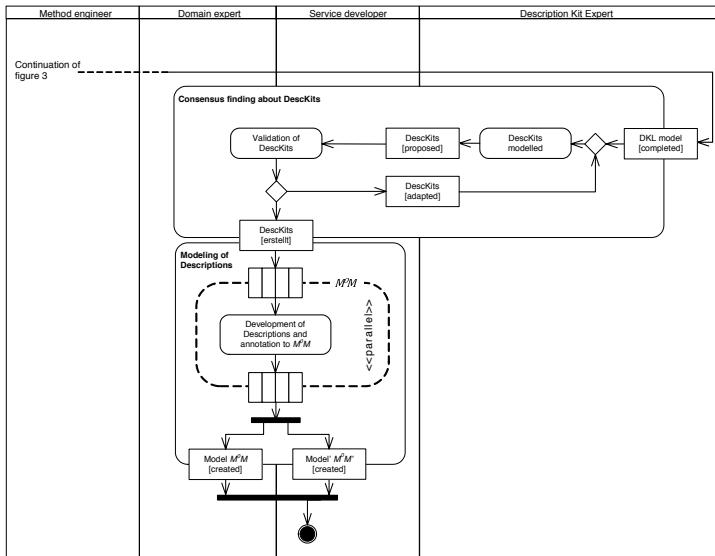


Figure 5: Procedure model to model creation 2

to the description kit expert. Using some feedback from the domain experts and technical experts, he plans modifications to the model on M^{0*} . If the modeling process is done in a team, clear responsibilities are important. It must be prevented that each method developer just creates DescKits as just needed by him. Certain persons perhaps should have no access to modifications on M^{0*} .

On layer M^0 DescKits now can or must — depending on how the DKL has been linked with the modeling language on M^1 — be used in concrete models. The modeling of Descriptions and within Descriptions is done hierarchically and therefore in another way as e.g. classical process modeling (see Figure 6). The hierachic top-down approach starts with a highly aggregated view on a certain fact and then step-by-step refines the description with more detailed information in a highly structured and well-defined way. A hierarchy of descriptions with increasing level of detail is the result.

This hierarchical way of modeling supports the modeler in distinguishing certain types of information about a real world phenomena from other types of information, to hide some part of the information or to handle different types of information differently. This is done without modifying language, but merely by an adaption of the descriptive structure of the DescKits. To use a DescKit, it has to be instantiated in a classical (linguistic) way. The description gets a unique name and is part of the ordinary modeling layer M^0 . At the same time a Description represents (in another sense of instantiation) a certain DescKit, which sets the framework (or guideline) for the further proceeding in the sense of “shaping” the description by the following tasks:

- Selection of a certain set of parameters and embedded DescKits (sub-DescKits) of

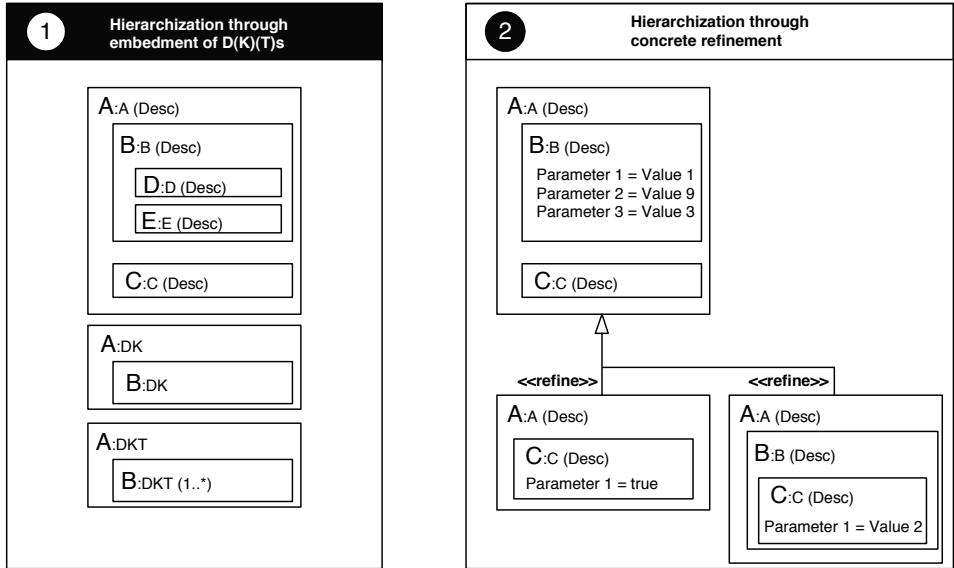


Figure 6: Way of hierarchization

the corresponding DescKit

- Setting values or constraints for the selected parameters
- Proceeding iteratively with the sub-DescKits

The set of Descriptions that are compliant with a certain DescKit is therefore determined by possible allocations of parameters, constraints and embedded Descriptions of a lower hierarchy. A certain fact can be modeled by the domain expert either completely from scratch, or by using pre-modeled Descriptions that can be refined. The DescKit also determines when the process of refining has to come to an end. This is dependent from the individual aim of the modeling project and can be fixed implicitly on layer M^0* . The hierarchic approach allows to describe real-world phenomena in models in a restricted way by obeying some guidelines, and furthermore offers the possibility to describe in a domain specific way certain phenomena that are hard to describe in an ordinary way.

4 Exemplary Application of the DKA

The aim of this section is to evaluate the applicability and usability of the DKA. This analysis is an important part of the present research methodology [20, p. 85], [44, p. 734]. To test the DKA in a realistic application one not only need a software implementation of the parts that can be formalized (language model, parts of the procedure model and the

algorithms), but also an organizational implementation of task that cannot be automated, like e.g. the creation of guidelines or the identification of a service.

A major role within the algorithms coming with the DKA plays the model comparison. This task is composed into two steps: A 1:1-mapping algorithm for simple descriptions and the so-called convolution algorithm, which is responsible for preparing complex models for an analysis by “folding” the different parts of a model along the relations into a single (in many cases artificial) description. The convolution results are then used to invoke the 1:1 comparison algorithm. This allows the comparison of arbitrarily complex models. We refer to [22] for an introduction into the algorithms and details on their implementation.

The exemplary scenario used for testing purposes (part of it presented here) contains — untypical for real-world applications — all possible process steps for a service, but also services that bundle the given process steps (by internally calling the finer granular services internally). See Figure 7 for part of the scenario.

It should be checked if the algorithm is able to find a certain service within a set of candidates, which were described using the DKA, and what probably suggested alternatives would be. The first step is a simple test that only uses the 1:1 -mapping algorithm. Then as a second step the convolution algorithm is used within a scenario based on whole process chains.

4.1 Exemplary Application of the Convolution Algorithm

Here we just show a very simplified example of the convolution algorithm. The test process consists of two process steps, described as interfaces using the DKA: “Create Document” (1) and “Send Document” (2), see the left side of Figure 7. The process is modeled in a way that step (1) generates a document for step (2).

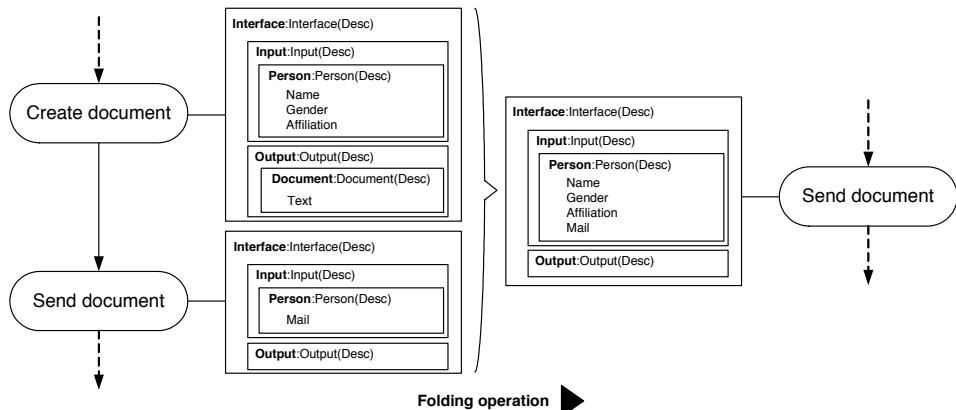


Figure 7: A simple convolution example

The folding algorithm now takes the two Descriptions (that use the DescKit “Interface”) and folds them along the relation (process flow) between them. The folding operation operates on the content of the Descriptions (especially embedded Descriptions, parameters, and values). It uses the 1:1-mapping algorithm for all embedded information and is doing the following:

- If two embedded contents match, combine them to one part of the result
- If two embedded contents do not match, embed them separately (as neighbors) into the result
- Intermediary content like here the document that step (1) produces, but step (2) then consumes, are removed
- These steps are done iteratively by proceeding into the depths of the embedding hierarchy

For the actual case the result is shown on the right hand side of Figure 7. All input and output objects are “collected”, while intermediary objects vanish from the convolution result. The result resembles now exactly the fact, that the complete process (step (1) and (2) together) “philosophically” simply represents a technical fulfillment of the business requirement of informing a person. It therefore should match similar processes that also fulfill similar requirements. The convolution result is the best foundation for that. It combines the “essence” of all process steps together and represents a single Description “Interface”, which can be used then in the 1:1-mapping algorithm to compare it with actual services.

Altogether, the prototypical implementation has shown in this and similar scenarios good results and therefore demonstrates the applicability and usability of the algorithm.

4.2 Exemplary Application of the 1:1-Mapping Algorithm

This section should demonstrate now the central part of the algorithms, namely the 1:1-mapping algorithm. It is used both within the convolution operation (for deciding how to “collect”) and after the convolution operation (to compare the convolution results), and therefore essential for realizing good results.

Using some parts of the previous example, Figure 8 shows how the 1:1-mapping algorithm works. The algorithm works in several steps:

1. Comparison of the exterior DescKitType: Matches (both are of DescKitType “Interface”)
2. Comparison of the exterior DescKit: matches
3. Comparison of the exterior Description: no match

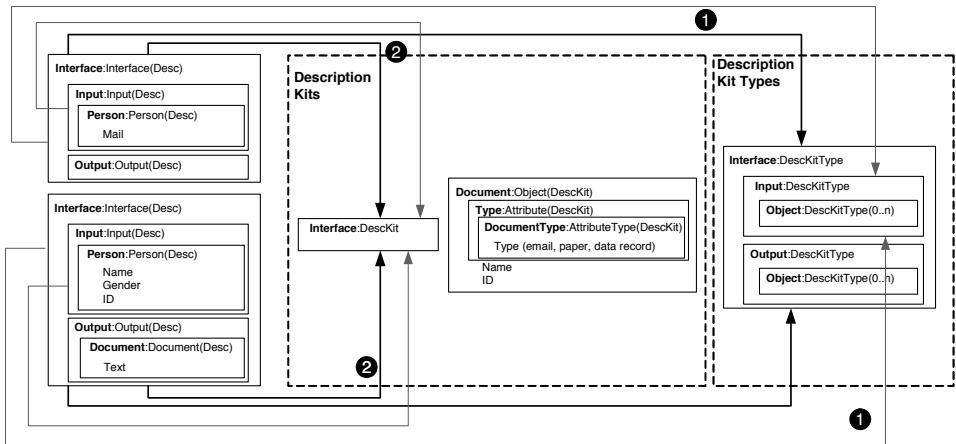


Figure 8: The 1:1-mapping algorithm in action

4. Proceed into the embedding hierarchy: Comparison of the embedded DescKitTypes, DescKits and Descriptions:
 - (a) First “Input” and “Output” match; proceed to the next hierarchy level
 - (b) “Person” matches with “Person” as DescKitType and as DescKit, while “Document” does not match another Description
 - (c) “Person” does not match as a Description, but now the algorithm can compare the set of parameters
 - (d) Each parameter again may contain constraints or values, which then can be compared

As can be seen, the algorithm begins with a rough comparison of the outer part of a description. This is of course only giving a slight idea of similarity, but resembles already some part that similar comparison approaches can do. The full strength the DKA unfolds by the hierarchical structure. The mapping algorithm can compare the whole internal structure of a description, which means that e.g. objects can be compared with all their states and dependencies, not only as objects itself.

The present examples show only part of the power of the algorithms, but the general idea should be transparent. The DKA combines a new approach on modeling with powerful comparison algorithms, and therefore can yield much better results than e.g. any attempt of simply standardizing business objects.

5 Evaluation and Summary

The addition (which can be enforced) of DescKits to process models allows on the one hand to use natural language for Descriptions in the sense of DescKits, on the other hand a restriction of the way how to model, and therefore allows to specify parts of the procedure model. The restrictions for using DescKits result in restrictions — in the sense of guidelines — for the ordinary modeling procedure. As a result an approach for the creation of procedure models has been created. For the special case of modeling in the sense of modeling by forming DescKits the creation of procedure models is automated by using the methodology.

5.1 Evaluation

Constraints in conceptual modeling The creation of the DKA has not the aim — in contrast to UML extensions — to include techniques into a modeling method, which allow arbitrary extensions to a modeling language. Quite in contrast, the aim of the DKA is a restriction of the freedom of modeling, without, however, restricting the modeler in his/her freedom to express certain (domain specific) ideas. The creation of DescKits can be seen as creating guidelines in the sense of a restriction of the use of natural languages when describing certain facts without destroying the link to the real world real-world phenomena.

Preventing language conflicts By introducing domain specific DescKits on layer M^{0*} it is possible to prevent homonym conflicts and synonym conflict. The solution of the abstraction conflict is also addressed by the DKA, since guidelines in the form of DescKits partly force the modeler to chose a certain abstraction level when creating Descriptions. In addition, the algorithms, which can compare the structure of Descriptions, may even identify similarities without having existing generalization or specialization relationships between DescKits on layer M^{0*} . By the embedding structure of the DescKits, a presetting is defined that partly can solve separation conflicts and annotation conflicts. Depending on the procedure model, also control flow conflicts may be able to be addressed by the embedding structure of DescKits.

Facilitating consensus finding The consensus finding between domain experts and technical experts is explicitly incorporated into the method. Beforehand, an extension and modification of the descriptive tools was available only by a modification of the meta-model. However, modifications of the meta-model can result in a corruption of existing object models. Therefore, a statical specification of a domain specific language can be problematic. By the introduction of the modeling layer M^{0*} an approach has been found, that makes the extension of a modeling language obsolete. The DKA implements a domain specific language now in an adaptable way. Even during the modeling on object layer the language is still flexible.

Generic algorithms When developing the DKA, a high focus was on the development of

generic algorithms that are controllable on the layer of the DescKitTypes. Therefore they can be used in very different scenarios.

A general problem is still to mark concrete objects within a model. Especially within the service identification scenario it is important for a specification for the propagation of business objects when orchestrating several service functionalities. If for example two processes consecutively process two person objects, which may be described by Descriptions. When composing these two functions, the question is, which person is which. Objects are not instantiations of a certain DescKit, but of a certain Description. However, a Description, even if completely specified, is only a description of an object state. This state can be matched always by a set of concrete objects. A function (or Interface) containing a description for an object only guarantees to work on objects that are in the specified state.

5.2 Conclusion

This article presents new aspects of the DKA that continue and go beyond what has been done in [22]. This includes the introduction of the role and procedure models and an more detailed description of how the different parts of the mapping algorithm apply in practice. It discusses the observed impacts, the difficulty in assessing the modeling language and the role of complementary and contextual factors. Furthermore, special intra-organizational collaboration during the consensus finding process allows to compare this method to classical modeling approaches. The domain expert creates a link from the modeled information to real-world phenomena already during the modeling process by following certain guidelines, represented by Description Kits. Hence, the information contained in these models is prepared for automatic processing. The semantic gap is significantly smaller than in a classical system development process.

Starting from this contribution, we see opportunities for further research in various directions. First, the DKA can only be successful if domain experts and technical experts can be convinced to accept the model-based methodology of the DKA. In the future, the prototypical implementation has to be extended to a full case study to thoroughly prove the applicability and usefulness of the DKA. Especially it has to be shown that the requirements on a model-based method can be met by the DKA better or more cost-efficient compared to ordinary methods. To possibility to combine the DKA with ordinary modeling methods will help to gain a higher acceptance in practice.

References

- [1] Aier, S. and Schönherr, M.: Flexibilisierung von Organisations- und IT- Architekturen durch EAI. In: Aier, S. and Schönherr (eds.): Enterprise Application Integration — Flexibilisierung komplexer Unternehmensarchitekturen. Berlin: GITO Verlag, 2004, p. 1–60.
- [2] Baker, S. and Dobson, S., Comparing service-oriented and distributed object architectures. In: Proceedings of the International Symposium on Distributed Objects and Applications 3760,

Springer, 2005 LNCS, pp. 631–645.

- [3] Balzert, H.: Die Entwicklung von Software-Systemen. Prinzipien, Methoden, Sprachen, Werkzeuge. Mannheim: Bibliographisches Institut, 1982.
- [4] Balzert, H.: Lehrbuch der Software-Technik: Software-Entwicklung. Heidelberg: Spektrum Akademischer Verlag, 2001, 2nd ed.
- [5] Becker, J., et. al.: Konstruktion von Methodiken: Vorschläge für eine begriffliche Grundlegung und domänen spezifische Anwendungsbeispiele. Westfälische Wilhelms-Universität Münster, Institut für Wirtschaftsinformatik, Arbeitsberichte des Instituts für Wirtschaftsinformatik 77, 2001.
- [6] Cervantes, H. and Haller, R. S., Technical Concepts of Service Orientation. In: Stojanovic, Z. and Dahanayake, A. (eds.): Service-Oriented Software System Engineering: Challenges and Practices. Hershey et al.: IDEA Group Publishing, 2005, p. 1–26.
- [7] Chmielewicz, K.: Forschungskonzeptionen der Wirtschaftswissenschaft. Schäffer-Poeschel Verlag, 1994.
- [8] Erl, T.: Service-oriented architecture: concepts, technology, and design, Prentice Hall PTR (2005)
- [9] Ferstl, O. K.; Sinz, E. J.: Grundlagen der Wirtschaftsinformatik. 1, 4, München, Wien: Oldenbourg, 2001.
- [10] Frank, U: Zur Verwendung formaler Sprachen in der Wirtschaftsinformatik: Notwendiges Merkmal eines wissenschaftlichen Anspruchs oder Ausdruck eines übertriebenen Szientismus? In: Becker, J., et. al. (eds.): Wirtschaftsinformatik und Wissenschaftstheorie: Bestandsaufnahme und Perspektiven. Wiesbaden: Gabler Verlag, pp. 127–160 (1999)
- [11] Frank, U.: Conceptual Modelling as the Core of the Information Systems Discipline — Perspectives and Epistemological Challenges. In: Haseman, D., et. al. (eds.): Proceedings of the Fifth Americaâs Conference on Information Systems (AMCIS 99). Milwaukee, pp. 695–697 (1999)
- [12] Gehlert, A.: Migration fachkonzeptueller Modelle. Berlin: Logos Berlin, 2007.
- [13] Greiffenberg, S.: Methodenentwicklung in Wirtschaft und Verwaltung. Hamburg: Dr. Kovac, 2004.
- [14] Griffel, F.: Componentware. Konzepte und Techniken eines Softwareparadigmas. dPunkt, Heidelberg (1998).
- [15] Gurr, C.; Tourlas, K.: Towards the principled design of software engineering diagrams. In: Proceedings of the 2000 International Conference on Software Engineering, pp. 509–518 (2000).
- [16] Gutzwiller, T.: Das CC RIM-Referenzmodell für den Entwurf von betrieblichen, transaktionorientierten Informationssystemen. Heidelberg: Physica, 1994.
- [17] Harel, D.; Rumpe, B.: Modeling languages: Syntax, semantics and all that stuff, part I: The basic stuff. Weizmann Institute Of Science, pp. 1–28 (2000).
- [18] Herrmann, H.-J.: Modellgestützte Planung im Unternehmen: Entwicklung eines Rahmenkonzepts. Wiesbaden: Gabler Verlag, 1991.
- [19] Heutschi, R.: Serviceorientierte Architektur: Architekturprinzipien und Umsetzung in der Praxis. Berlin, Heidelberg: Springer-Verlag (2007).

- [20] Hevner, A. R. et al.: Design Science in Information Systems Research. In: MIS Quarterly **28**(1), pp. 75–105 (2004)
- [21] Heym, M.: Methoden-Engineering: Spezifikation und Integration von Entwicklungsmethoden für Informationssysteme, Hochschule St. Gallen für Wirtschafts-, Rechts- und Sozialwissenschaften, Dissertation, 1993.
- [22] Juhrisch, M., Dietz, G., Esswein, W.: Perspectives on Semantic Business Process Modeling — A Generic Approach. In: Proceedings of the 13th Pacific Asia Conference on Information Systems (PACIS 2009) (2009).
- [23] Juhrisch, M., Dietz, G.: Constraints in Conceptual Modelling — Outlining an Approach to Business Driven Web Service Composition. Int. J. Internet and Enterprise Management **6**(3), 248–265 (2010).
- [24] Kraatzig, D. et al., Enterprise SOA: Service-Oriented Architecture Best Practices. Prentice Hall, 2004.
- [25] Kühne, T.: Matters of (Meta-) Modeling. In: Journal on Software and Systems Modeling, 5(4), pp. 369–385 (2006)
- [26] Linke, A.; Nussbaumer, M.; Portmann, P. R.: Studienbuch Linguistik. 5th ed., Tübingen: Max Niemeyer Verlag, 2004.
- [27] Lorenz, K.: Methode. In: Mittelstrass, J. (ed.): Enzyklopädie Philosophie und Wissenschaftstheorie Band 2. Stuttgart, 1995, pp. 876–879.
- [28] Lorenz, K.: Sprache, natürliche. In: Mittelstrass, J. (ed.): Enzyklopädie Philosophie und Wissenschaftstheorie: Sp-Z Band 4. Mannheim, Bibliographisches Institut (1996)
- [29] Matthes, F., The impact of SOA on the Enterprise Application Landscape. In: Proceedings of the SOA Days 2005 Business Conference. Bonn: Deutsche Post World Net, 2005.
- [30] McGovern, J. et al, Service-Oriented Architecture. In: McGovern, J. et al. (eds.): Java Web Services Architecture. San Francisco: Morgan Kaufmann, 2003, pp. 35–63.
- [31] Ortner, E.: Methodenneutraler Fachentwurf: zu den Grundlagen einer anwendungsorientierten Informatik. Stuttgart; Leipzig: Teubner Verlagsgesellschaft, 1997.
- [32] Pfeiffer, D.: Constructing comparable conceptual models with domain specific languages. In: Proceedings of the 15th European Conference on Information Systems (ECIS2007), pp. 876–888 (2007)
- [33] Pfeiffer, D., Gehlert, A.: A Framework for Comparing Conceptual Models. In: Desel, J., Frank, U (eds.): Enterprise Modelling and Information Systems Architectures: Proceedings of the Workshop in Klagenfurt. Bonn: Köllen Druck + Verlag GmbH, Lecture Notes in Informatics P-75, pp. 108–122 (2005).
- [34] Remme, M.: Konstruktion von Geschäftsprozessen: Ein modellgestützter Ansatz durch Montage generischer Prozeßpartikel. Wiesbaden: Gabler, 1997
- [35] Schaffner, J.; Meyer, B.: Mixed initiative use cases for semi-automated service composition: a survey. In: Proceedings of the 2006 international workshop on Service-oriented software engineering. Shanghai, China: ACM, pp. 6–12 (2006).
- [36] Schütte, R.: Grundsätze ordnungsgemäßer Referenzmodellierung: Konstruktion konfigurations- und anpassungsorientierter Modelle. Neue betriebswirtschaftliche Forschung 233, Wiesbaden: Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, 1998

- [37] Searle, J. R.: *Speech acts: an essay in the philosophy of language*. University Press, Cambridge, 1969.
- [38] Sessions, R., Fuzzy Boundaries. In: ACM Queue, 9 (2004), pp. 40–47.
- [39] Sinz, E. J.: Modellierung betrieblicher Informationssysteme: Gegenstand, Anforderungen und Lösungsansätze. In: Pohl, K., et. al. (eds.): Proceedings Modellierung '98, pp. 27–28 (1998).
- [40] Stachowiak, H.: Allgemeine Modelltheorie. Wien: Springer Verlag, 1973.
- [41] Stahlknecht, P.: Einführung in die Wirtschaftsinformatik. 7. Aufl., Berlin, 1995.
- [42] Stojanovic, Z., A Method for Component-Based and Service-Oriented Software Systems Engineering, Delft University of Technology, Dissertation, 2005
- [43] Tolvanen, J.-P. (1998) Incremental Method Engineering with Modeling Tools: Theoretical Principles and Empirical Evidence, University of Jyväskylä, Phd thesis.
- [44] Verschuren, P., Hartog, R.: Evaluation in Design-Oriented Research. In: Quality and Quantity, 39(6), pp. 733–762 (2005).
- [45] Veryard, R., Business Adaptability and Adaptation in SOA. In: CBDi Journal, (2003), November, pp. 15–23.
- [46] Vogels, W., Web Services Are Not Distributed Objects. In: IEEE Internet Computing, 7 (2003) 6.
- [47] Wand, Y.; Weber, R.: Research Commentary: Information Systems and Conceptual Modeling — A Research Agenda. In: Information Systems Research, 13(4), pp. 363–377 (2002).
- [48] Winter, A.: Referenz-Metaschema für visuelle Modellierungssprachen. Wiesbaden, Universität Koblenz-Landau, Dissertation, 2000.
- [49] WKWI: Rahmenempfehlung für die Universitätsausbildung in Wirtschaftsinformatik. Wissenschaftliche Kommission Wirtschaftsinformatik, Gesellschaft für Informatik e.V., 2007.
- [50] Zelewski, S.: Eignung von Petrinetzen für die Modellierung komplexer Realsysteme: Beurteilungskriterien. In: Wirtschaftsinformatik, 38(4), pp. 369–381 (1996).
- [51] Zelewski, S.: Grundlagen. In: Corsten, H., Reiss, M. (eds.): Betriebswirtschaftslehre. 3rd ed, München, pp. 1–125 (1999)
- [52] Zimmermann, O., Krogdahl, P., Gee, C.: Elements of Service-Oriented Analysis and Design. IBM developerWorks (2004)

Toward modeling constructs for audit risk assessment: Reflections on internal controls modeling

Stefan Strecker, David Heise, Ulrich Frank

Information Systems and Enterprise Modelling Research Group
Institute for Computer Science and Business Information Systems
University of Duisburg-Essen, Universitaetsstr. 9, 45141 Essen, Germany
{stefan.strecker | david.heise | ulrich.frank}@uni-due.de

Abstract: Auditors face a number of challenges when performing audit risk assessment. To cope with these challenges, methods are required that purposefully reduce the complexity inherent to internal control systems and that facilitate communication about internal control matters among groups of stakeholders with differing perspectives on the subject matter. In this paper, we investigate the potentials of an enterprise modeling approach to audit risk assessment and propose conceptualizations for modeling constructs as enhancements to enterprise modeling to support audit risk assessment.

Keywords: *Enterprise modeling; Language design; Internal controls; Audit risk assessment; Governance, Risk, Compliance*

1 The extended scope of audit risk assessment

Section 404 of the Sarbanes-Oxley-Act of 2002 (SOX) and both the Directives 2006/43/EC and 2008/30/EC of the European Parliament and of the European Council (“EuroSOX”) mandate the establishment, documentation and management of internal control systems and their subsequent auditing as part of audit risk assessment. Present auditing standards and guidelines commit auditors to gain an in-depth understanding of a firm’s business, its operations and processes, associated risks and internal controls when assessing the risk of material misstatement [SP02]. As relevant risks pervade the enterprise from operations to corporate strategizing, it needs to be questioned “[w]hy limit the analysis to the business process level?” [Dun06, 207]—when legal regulations and auditing standards prescribe assessing risks at all relevant levels of the organization [The92, Inf09]. Audit risk assessment therefore pertains to any risk of not achieving objectives not only risks related to financial reporting [CS02]. Hence, auditing standards “emphasize the importance of auditors gaining a broader understanding of an organization” [Car06, 171]. Put differently, auditing guidelines prescribe to account for the appropriate organizational context of internal controls: Auditors are required to understand a firm’s business, its risks and controls in place to treat risk exposure at all relevant organizational levels which implies an understanding of entity objectives, business processes, organizational structures, roles, responsibilities and resources.

Against this background, auditors face a number of challenges when performing audit risk assessment: They are confronted with the remarkable complexity of present day enterprises—especially considering the multifaceted interrelations between organizational structures, business processes, and IT assets. Auditors also have to deal with the complexity of internal control systems themselves: Controls occur for multiple organizational levels, refer to a multitude of different entities and address a variety of risks—apart from the sheer number of controls and their possible interactions [Mai00]. Moreover, auditing internal control systems requires the participation of stakeholders with different professional backgrounds and perspectives on internal control matters including executives, line managers, process owners, risk managers, internal and external auditors [SP02]. In this respect, the complexity challenge is intensified by different technical languages, differing mindsets, and resulting barriers to communicate and to participate—hampering in particular the collaboration between auditor and auditee.

To cope with these challenges, methods are required that purposefully reduce the complexity inherent to internal control systems and that facilitate communication about internal control matters among groups of stakeholders. The auditing and accounting literature recognizes the potentials of supporting audit risk assessment through conceptual models, in particular business process models in the context of process-level audit risk assessment (e.g. [Inf09, 132]). It is, however, acknowledged that present generic approaches to business process modeling do not provide adequate modeling constructs required for representing internal control systems with regard to effectively and efficiently supporting auditors when performing audit risk assessment [Car06]. In particular, it is criticized that present approaches focus on the business process level and do not provide support for adequately representing further relevant organizational context such as objectives, organizational roles, responsibilities and resources [Dun06]. Such modeling concepts are, however, common to enterprise modeling approaches such as ARIS [Sch00, Sch92], MEMO [Fra08, Fra02] and ArchiMate [Lan05] which supplement business process models with further abstractions of the enterprise (e.g. corporate goals and strategies, organizational structures, roles and resources). While current enterprise modeling approaches, thus, provide support for necessary organizational context, they—to our knowledge—do, however, not provide elaborate domain-specific modeling concepts for internal controls modeling.

In this respect, our motivation is twofold: We investigate the potentials of an enterprise modeling approach to audit risk assessment and propose conceptualizations for modeling constructs as enhancements to enterprise modeling approaches to support audit risk assessment. Next, we briefly review related literature (Section 2). Section 3 analyses and reconstructs technical terminology in the auditing domain. Section 4 introduces design goals and requirements a method for audit risk assessment should satisfy. The general prospects of an enterprise modeling approach to audit risk assessment are investigated in Section 5. In Section 6, we reflect on the design of domain-specific modeling constructs and discuss design issues. Section 7 presents concluding remarks.

2 Related work

Since McCarthy's work on the REA (resources, events, agents) model [McC79, McC82], auditing and accounting information systems literature recognizes the use of conceptual models of the enterprise for supporting accountants and auditors in understanding a firm's business [GSF04, DCH05, RBGR06]. Studies on the actual use of graphical representations support anecdotal evidence that system flowcharts and data flow diagrams are still the predominant means of graphical representation used in audit reviews [GSF04, 24]. A recent study shows, however, that business process modeling approaches such as the Business Process Modeling Notation (BPMN) or the Event-driven Process Chain (EPC) approach are gaining increasing acceptance in the auditing domain [ABC08]. At the same time, behaviorist research indicates that graphical representations of the enterprise advance the understanding of auditors over text-based documentation [ALM02, ABC04].

Carnaghan reviews different business process modeling approaches with regard to their support for audit risk assessment and concludes that present approaches do not allow for adequately expressing the semantics of internal controls and, consequently, further modeling constructs are required that explicitly capture the domain's technical language [Car06]. Dunn, in a review of the study, supports her conclusion by stating that "these tools were not designed with audit risk assessment suitability in mind but that is not to say that we couldn't develop one" [Dun06, 207]. This assessment, however, ignores contributions from the conceptual modeling community to the auditing domain.

For instance, Petri nets have for long been discussed as a means to document business processes and corresponding internal controls [PP97, CL03]. More recently, several approaches based on formal logic interpret internal controls in terms of formalized rules for a business process' control flow [GHSW08, LSG08, SGN07, GMS06, NS07]. These approaches primarily target compliance checking based on automated reasoning. Technology-focused research is aimed at computer-assisted auditing. Agrawal ea. [AJKL06], for instance, present a workflow engine that compares logs of process executions with predefined workflow schemas with regard to violations of audit constraints. Karagiannis ea. [KMS07] present modeling tool support for business process compliance. A further stream of literature deals with conceptual models of risks and chances [SHF10, SLKP07]. For example, zur Mühlen and Rosemann [zMR05] extend EPC with modeling concepts for risk types and related diagram types.

In summary, a rich body of literature contributes to support audit risk assessment through conceptual models. However, prior work has not discussed differentiated domain-specific modeling concepts for internal controls modeling. It also does not deal with the reuse of concepts in the context of enterprise modeling and of extending present enterprise modeling approaches with domain-specific concepts for internal controls modeling. Present modeling support for audit risk assessment focuses on business process models and on formal representations of internal controls. This brief analysis of the current state-of-the-art in literature motivates our research on domain-specific modeling constructs for audit risk assessment.

3 Domain analysis

Designing domain-specific modeling concepts presupposes reconstructing key terms and their semantics in the targeted domain. Reconstruction of domain-specific concepts is an iterative process involving not only the mere identification of candidate classes, their attributes and relations. Rather, it requires, for instance, identifying and resolving terminological ambiguity and truncation, e.g., by introducing additional abstractions and by shaping their conceptualization. This implies (re-)interpretation of observed terms and their semantics to design abstractions suitable for intended purposes, analyses and possibly for further future applications. One widespread approach to conceptual reconstruction—the one we follow here—is to review, analyze and interpret pertinent literature in the field under consideration. This section summarizes key findings from the conceptual reconstruction of the technical terminology in the auditing domain, specifically in the context of audit risk assessment.

In auditing literature and practice, “control”, “internal control”, and “internal control system” are commonly used terms [Moe08]. Despite their proliferation, a lack of precise definition and understanding of even these key domain concepts has repeatedly been criticized [Mai00]. The term “control” is in fact subject to a considerable diversity of disciplines, for example, “management control, organizational control, internal controls, operational control and financial control, which all seem to revolve around the same concept” [RBGR06]. The auditing perspective on *internal control* is decisively influenced by the Committee of Sponsoring Organizations of the Treadway Commission (COSO) [The92, The04] and subsequent auditing standards such as the Public Company Accounting Oversight Board (PCAOB) Auditing Standard No. 5 (for a discussion see [RBGR06] and [Mai00]):

“COSO defines internal control as a process, effected by an entity’s board of directors, management and other personnel. This process is designed to provide reasonable assurance regarding the achievement of objectives in effectiveness and efficiency of operations, reliability of financial reporting, and compliance with applicable laws and regulations. [...] Internal control is not merely documented by policy manuals and forms. Rather, it is put in by people at every level of an organization. [...]” ([The10]; adapted from [The92]).

While this very broad conceptualization provides insights into essential domain-specific concepts (e.g., objectives, policy, reasonable assurance), it also points to (necessary?) terminological ambiguity: Internal control obviously denotes *not only* a process but covers both procedural aspects (e.g., people, processes) and structural aspects (e.g., policy, organizational structures). Surprisingly, neither risk nor control objectives are mentioned—yet both constitute essential concepts in the frameworks provided by COSO and by the Information Systems Audit and Control Association (ISACA). In a later framework, COSO consequently adapts the internal control definition to the broader context of risk management: “Enterprise risk management is a process, [...] designed to identify potential events that may affect the entity, [...] to provide reasonable assurance regarding the achievement of entity objectives” [The04].



Figure 1: Initial reconstruction of domain terminology: Essential domain-specific concepts

Hence, in general, internal control can be said to refer to means to reduce risks of not achieving objectives [SP02]. Consequently, evaluation of internal controls by auditors is considered as a risk assessment process, often denoted as audit risk assessment. A first conclusion from this brief terminological analysis pertains to the very conception of “internal control”: Internal control cannot be conceived as a singular concept as such, but rather as an abstraction over various other concepts which in turn constitute an internal control. An initial reconstruction of the constituent concepts of “internal control” is shown in Fig. 1. It is mainly based on an analysis of the mentioned COSO documentation, two textbooks [DCH05, GSF04] as well as anonymized audit documents received from a Big Four auditor.

A key domain concept is *control objective*, sometimes also denoted as control goal [GSF04, 249]. It represents a desired state of an enterprise (“Prevent unauthorized refunds”) with respect to achieving an *entity objective* (“Minimize error rate of incorrect refunds”) that is threatened by a *risk* (“Internal fraud due to fraudulent behavior of employees”). A control objective is associated with a recommended course of action that should be taken to provide reasonable assurance that entity objectives will be met and, thus, corresponding risks of not achieving it are mitigated. The course of action can involve policies, procedures, practices, or organizational structures as concrete measures or means of control implemented to ensure effectiveness of a control. A possible means to prevent fraud is “segregation of duties”. Such a *control means* is aimed at achieving the control objective. It represents an abstraction over static means of control such as written policies or organizational structures and dynamic means of control such as activities and procedures. Alternative denominators to the control means concept could have been “control activity” [IT 07] and “control plan” [GSF04, 249]. Both, however, entail a significant risk of misinterpretation: The term “activity” raises associations with dynamic abstractions neglecting static aspects while the term “plan” emphasizes a perspective different from the intended means-end association. Examples for general control means given in COSO publications include the authorization of transactions as well as adequate safeguards of assets and records. The achievement of the desired outcome of a control objective is measured by an *indicator* (e.g., “Percentage of fraudulent refund transactions”) as is the severity of risk. Responsibilities (as in the RACI conceptualization: “Responsible”, “Accountable”, “Consulted”, “Informed”) are defined typically for more than one *organizational role* (“executive”, “business process owner”, etc.) with respect to a control objective. It is important to note that frameworks such as the ones provided by COSO and the ISACA (e.g., COBIT) assume relationships between internal controls that typically form a hierarchy, possibly a net of controls. Also note that monitoring and auditing an internal control system (e.g., performed as audit risk assessment by an external auditor) constitute processes detached from the actual internal control system in that the system itself becomes subject to the audit [The09].

4 Design goals and requirements

The general purpose of this design research project is to effectively and efficiently support auditors when performing audit risk assessment. More specifically, this work is aimed at supporting auditors in understanding a firm's business, its operations and processes, associated risks and internal controls when assessing the risk of material misstatement. Given that the auditor's understanding is educated in group processes [Dam05, 79], its purpose is to support group processes by reducing the complexity inherent in internal control systems and by providing abstractions tailored to the perspectives of stakeholders involved. Thus, this work is aimed at fostering and facilitating communication and collaboration among stakeholders involved in audit risk assessment—with a dedicated focus on auditor and auditee interaction. It also aims to increase transparency of internal control matters, specifically by visually representing internal controls as part of the organizational action systems, and by improving traceability of the controls in place to treat risk exposure. Ultimately, this paper is also intended to intensify the dialog between the accounting information systems and the conceptual modeling communities.

The overall design goal is to enhance present enterprise modeling approaches by constructs for internal controls modeling. In this paper, however, we do not present a complete language specification but discuss design decisions and alternatives based on a preliminary draft of a language specification as an initial step toward developing modeling constructs for audit risk assessment. Below, we refine the stated purpose and goal to establish five domain-specific requirements that a method aimed at supporting audit risk assessment should satisfy (for a rationale also see [SHF10]).

Requirement 1—Organizational Context: A method should link internal controls to the surrounding organizational action system composed of all organizational entities relevant to audit risk assessment. This organizational context is provided by (at least) entity objectives, business processes, (IT) resources, organizational structures, roles and involvement.

Requirement 2—System of internal controls: A method should account for relationships among internal controls on different organizational levels, from IT operations to business processes to value chains to the organization as whole.

Requirement 3—Justification and assumptions: A method should provide means for justifying the existence and importance of internal controls and for revealing assumptions underlying internal control justification.

Requirement 4—Diversity of implementation: A method should account for the diversity of actual means to achieve control objectives and of the resulting internal control implementation.

Requirement 5—Support for multiple perspectives: A method should provide perspectives specific to (groups of) stakeholders involved in the group process. A perspective should, as far as possible, correspond with the abstractions, concepts and (visual) representations known and meaningful to the targeted (group of) stakeholders. All perspectives should, on the other hand, be integrated with each other to foster cross-perspective communication and cooperation.

5 Illustration of an enterprise modeling approach

This section illustrates the prospects of supporting audit risk assessment with domain-specific modeling concepts integrated with an enterprise modeling approach. The analysis is based on two presuppositions: First, we presume that the enterprise modeling method is based on a language architecture that allows for reuse of existing modeling concepts (for an example see [Fra08]). Second, we assume that the enterprise modeling method provides language concepts for representing control flows, goals, roles, and organizational structures as is the case, for instance, with ARIS [Sch00, Sch92] and MEMO [Fra08, Fra02]—thereby providing concepts to represent the organizational context required for internal controls (cf. Req. 1). The MEMO approach was chosen to illustrate the application scenario in Fig. 2, because it fulfills both assumptions and integrates further concepts essential to audit risk assessment (e.g., risk [SHF10], indicator [FHK09] and IT resources [Kir05]). It is important to note that the shown diagram is not intended to predetermine a specific language design. Instead, it serves as an illustration of principle applications of enterprise models in the context of audit risk assessment.

The scenario is based on and inspired by a business process model of a refunding returned goods process drawn on by [Car06, 200] to compare different business process modeling approaches in the context of audit risk assessment. The scenario shows a goal model (top left) that represents (an excerpt of) a hierarchy of the enterprise’s strategic goals and subsequent business objectives; a business process model for “refunding returned goods” at three different levels of abstraction (i.e., an aggregated process and its decompositions; bottom left); a model of the corresponding organizational structure (including a model of organizational roles; top right) and a model of IT resources used in some of the processes (showing an information system abstraction of an ERP system; bottom right). Further models such as corresponding object models are not shown in the diagram for the sake of clarity. Relationships between concepts in different models are explicitly modeled by associations (e.g., the ERP system used in the business process “Authorize credit”) or by shared concepts (e.g., the organizational role “A/R manager” in both the process “Authorize credit” and in the organizational structure model).

Provided such an infrastructure exists, internal controls modeling can be supported by extensions to existing concepts and by introduction of additional abstractions. We will illustrate the latter first. The scenario shown in Figure 2 assumes that some constituent concepts of internal controls are represented by *additional* modeling concepts. In particular, the control objective concept represents such an addition. The control objective “Prevent unauthorized refunds” recommends a segregation of duties in the aggregated process “Refund returned goods”. The internal control semantics is further specified by the IT control “Prevent unauthorized transactions”, by the risk “Internal fraud” it aims to mitigate, by the audit activity “Audit refund transactions and detect irregularity”, and by the indicator “Percentage of fraudulent refund transactions” to measure achievement of the control objective. An *extension* to existing modeling concepts is shown as a visual overlay (a red triangle), which serves to highlight all those model elements that are related to an internal control. In Figure 2, for instance, the process “Authorize credit” is enriched with an overlay, as the process realizes a significant part of the segregation of duties. Similarly, overlays are



Figure 2: Illustration of an enterprise modeling approach to internal controls modeling

attached to the information system symbol “ERP system” in the IT landscape model and some of the organizational units in the structure model, since these elements are all related to the internal control(s).

Figure 2 also demonstrates how the control objectives can be associated with concepts that represent the organizational action system they are embedded in (cf. Req. 1). First, they can be associated to the entity objectives they are aimed at assuring the achievement of (i.e., the goal “Minimize error rate of incorrect refunds” in the goal model). Second, control objectives can be linked to static and dynamic abstractions representing means of control. For example, the above mentioned control objective refers to the business process “Refund returned goods”, whereas the actual realization of the recommended action “segregation of duties” is depicted at the most detailed level of the process model. Third, the IT control refers to an IT asset in the IT resource model (“ERP system”) that realizes the segregation at the information system level by authorization and system access policies. Linking the two control objectives also demonstrates how associations between controls aid in visualizing internal control systems. Finally, the integration with the model of the organizational structure emphasizes different types of involvement of organizational roles. For instance, the two roles participating in the “Credit customer account” process—“Sales Account Manager” and “A/R Manager”—are linked to the control objective specifying their type of involvement (i.e., ‘accountable’ respectively ‘responsible’), whereas a “Financial Officer” is regularly informed but not explicitly modeled as part of the business process.

With respect to the intended purpose of effectively and efficiently supporting auditors in understanding a firm’s business, its risks, and controls, such integrated models of the enterprise and its internal control system promise to provide an intuitive access and a comprehensible conceptual foundation for differentiated and structured analysis of the internal control system. By associating internal controls with further models (e.g., of business processes or IT landscapes) and by tagging affected reference objects with an overlay symbol, this approach facilitates internal control-related communication and collaboration between groups of stakeholders with different professional backgrounds (cf. Req. 5). By focusing on types (of controls, risks, processes etc.) rather than instances, such an approach purposefully reduces complexity and contributes to focusing on aspects relevant to the audit analysis.

Besides documentation—and thus queries on which controls exist in an enterprise—such integrated models support further analyses. On the one hand, they allow for analyzing controls in respect to the organizational context they affect. For instance, in Figure 2, the business control objective is associated to one of the firm’s business objectives, a business process, and an IT resource. For auditing purposes, this allows for comparing the current implementation of a control with, for instance, reference models of internal controls or check lists of prescribed control means. On the other hand, it allows for analyzing various organizational concepts with regard to whether they are affected by controls. Especially in organizational settings that experience rapid changes such an analysis can assist in preventing failure to comply with regulations. In Figure 2, for instance, an analysis of the committee “IT architecture board” reveals a relationship to a control objective, so that eliminating this organizational unit from the model (e.g., as a result of a reorganization project) raises an exception and notifies stakeholders of a likely compliance violation.

In summary, the illustrated enterprise modeling approach promises a number of advantages over textual representations, simple conceptual models or even present business process modeling approaches:

1. As a general prospect of enterprise modeling, the purposeful abstractions of the action system promise to reduce the complexity in analyzing a company's internal controls and, thus, per se announce support for internal and external auditors.
2. The proposed reuse of existing modeling concepts increases the productivity of both language design and language application. Language designers benefit from mature modeling concepts and notations and can focus on relevant additions and modifications. Modelers as language users benefit from the reuse of existing models (e.g., of business processes) and can focus on adding relevant contextual information (e.g., risks, indicators).
3. The partially formal specification of modeling concepts allows for model transformations into other representations (e.g., to some extent, into source code), which provides a foundation for developing corresponding information systems based on a model-driven development approach.
4. Reconstructing the technical terminology using such an enterprise model-based approach also carries the potential to contribute to a less ambiguous domain terminology (e.g., with respect to the term "internal control") in that it offers a conceptualization of key domain concepts with a partially formal semantics.

Based on these considerations, we envision that enterprise models enriched by dedicated internal control concepts can be used in audit reviews as audit evidence, i.e., as structured documentation of a firm's internal control system—to facilitate interpretation and assessment of controls by auditors.

6 Considerations on language design

Based on the corroborative assessment of the potentials of an enterprise modeling approach to audit risk assessment, this section outlines general considerations toward enhancing enterprise modeling approaches with domain-specific modeling concepts for audit risk assessment, and discusses essential decisions related to the design of these modeling constructs. In this section, we present preliminary specifications of modeling constructs as meta model excerpts specified using the MEMO Meta Modeling Language [Fra08]. These specifications are intended as a working draft for the following discourse and as a foundation for discussions with and discursive evaluation by peers and domain experts. In the following, we assume a modeling infrastructure as described in Sec. 5. The reuse of modeling concepts from existing modeling languages in the MEMO language family is visualized by a colored rectangle attached to the meta type header indicating the concept's origin (as suggested in [Fra08]).

6.1 Devising an infrastructure for internal controls modeling

Enhancing an enterprise modeling approach to support audit risk assessment requires conceptualizing modeling constructs based on domain-specific concepts. The initial design decision with respect to our targeted domain pertains to the conceptualization of an internal control. Based on the analysis in the previous sections, it appears justified to represent an internal control by its control objective and by the means of achieving the control objective. Hence, we decided to introduce two dedicated meta types, *ControlObjective* and *ControlMeans*. To represent the semantics of internal controls, further refinements are, however, necessary and detailed below. The rationale of introducing those two meta types is to enable dedicated audit analyses based on respective internal control models.

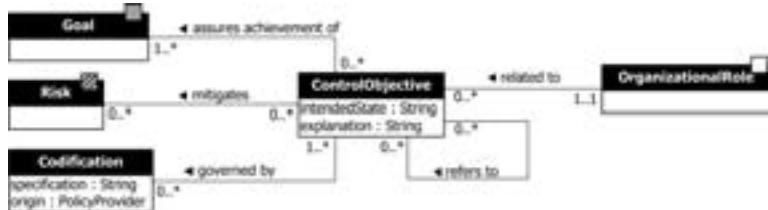


Figure 3: Conceptualization of “control objective”

ControlObjective. For conceptualizing the meta type *ControlObjective*, we propose to provide additional semantics by describing control objectives by a natural language specification of a desired state of the enterprise, for instance, by an attribute *intendedState* and an explanation of the intended state, *explanation*. Both attributes contribute to *Requirement 3—Justification and assumptions* in that they allow for providing a rationale for the existence and importance of an internal control. The importance of a control could have been further specified, for example, by an additional attribute, say *priority*. The present conceptualization subsumes this information under *explanation*. The justification is augmented by providing information on the *Codification* the control objective is governed by. In regard to the various legal regulations, auditing standards, and guidelines, feedback from practicing auditors revealed that it is recommended to keep track of these codifications (e.g. a certain clause in an auditing standard) and of the originating policy provider (e.g., COSO, PCAOB No. 5, COBIT). Note that in some cases it may be feasible to rephrase the natural language specification in a formal logic to support automated reasoning on control objectives (e.g., [SGN07]). In this respect, the proposed conceptualization simplifies future enhancements to cover such rule-based formalizations (e.g., as an additional attribute). The integration with an enterprise modeling approach allows to comfortably express further semantics of internal controls by associating the meta type *ControlObjective* to the meta types representing entity objectives (*Goal*), risks (*Risk*) and organizational structures (*OrganizationalRole*) and, thus, to promote reuse of modeling constructs and to address *Requirement 1—Organizational context*. For instance, associating a risk to a control objective implicitly provides a rationale for the existence of a control objective and is recommended to increase model comprehension. The explicit association of risks with

control objectives enables further analyses for auditing purposes (e.g., identification of risks without controls and vice versa [SP02]). Also note that *Requirement 2—System of internal controls* is addressed by the recursive association between control objective types to allow for links between internal controls. Those links enable further analyses such as which IT controls impact which business controls. Figure 3 illustrates a corresponding specification of *ControlObjective*.

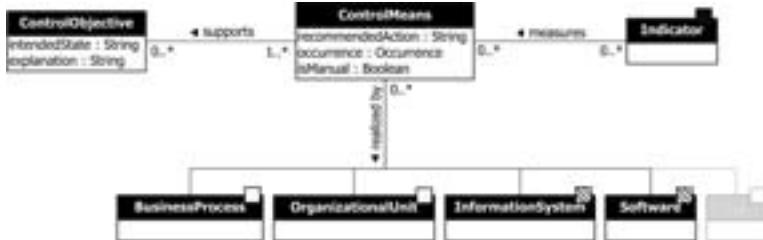


Figure 4: Conceptualization of “control means”

Control Means. Following our earlier analysis, we propose to conceptualize the meta type *ControlMeans* as an abstraction over both static and dynamic aspects of the means to achieve control. In the light of *Requirement 4—Diversity of implementation*, the conceptualization shown in Fig. 4 is aimed at providing flexibility while at the same time providing a structure for auditing purposes. We suggest to specify the recommended course of action by a natural language specification, *recommendedAction*. Such a specification could mention written policy and corresponding procedures. Further semantics is specified by *occurrence* to classify control means into preventive, detective or corrective controls according to their effect in time relative to the occurrence of a risk [GSF04, 253]. Another characteristic is captured by a functional differentiation between manual and automated control activities, *isManual*. Control means may be linked to indicators measuring the effectiveness and efficiency of a particular means by associating the meta type *ControlMeans* to the meta type *Indicator* which, for example, provides semantics to create comprehensive indicator systems [FHK09]. From the analysis in the previous sections we conclude that (1) multiple means exist that can be deployed to achieve a control objective; (2) the same means can be reused by several control objectives; (3) a certain means can pertain to several structural *and* procedural elements and thus exhibit a “multidimensional” characteristic (e.g., a written policy corresponding with a process “Authorize credit”). One approach would be to introduce dedicated concepts to represent the intricacies of control means (e.g., modeling concepts for policy and procedure). We have currently refrained from that option for two reasons: (1) The multitude of control means (e.g., [GSF04, 253ff.]) suggests the need for a high degree of flexibility with respect to representing the spectrum of relevant measures and the abstractions they refer to (cf. Req. 4); (2) by associating control means with existing concepts such as *BusinessProcess*, *OrganizationalUnit* or *InformationSystem* the semantics of many cases can be covered through additional organizational context (cf. Req. 1). For example, a control means “Segregation of duties” can be associated with a business process “Authorize credit” and with an information system “ERP system” to

implement a written policy. The current proposal, however, poses a number of notational problems, for example, how to visually identify all elements belonging to an internal control and its means. Introducing overlay symbols on notation elements is a response to this issue but relies on tool support and may, in practical applications, sacrifice clarity of the graphical notation. In the following, we will discuss further design issue with regard to our proposal.

6.2 Design issues

Involvement of organizational roles. A design issue relates to the different types of involvement that organizational roles—i.e., stakeholders—can have in relation to control objectives. Both, domain analysis and application scenario suggest differentiating the involvement of organizational roles into internal controls. For instance, the IT Governance Institute suggests four types of involvement in the well-known RACI charts [IT 07]: Responsible, Accountable, Consulted, and Informed. Thus, the conceptualization of the relation between roles and control objectives in Fig. 3 will probably not be sufficient for audit risk assessment purposes, as it lacks any elaborated semantics. Figure 5 illustrates two design alternatives that are feasible to represent these types of involvement: First, for each identified type of involvement a particular association between the meta-types representing the organizational role and the control objective is established (cf. Fig. 5(a)). Second, a meta-type serves as an “association class” between organizational role and control objective and allows for instantiating these four—and further—associations (cf. Fig. 5(b)). While the first alternative restricts modelers to predefined types of involvement and their min/max-cardinalities—and is thus likely to promote a more secure modeling; the second alternative provides more flexibility for modelers to add company-specific relations (e.g., “supports”) without adapting the meta-model.



Figure 5: Design alternatives for the specification of the involvement of organizational roles

Control Categories. Audit literature proposes several typifications of controls, among other, general controls, business process controls, financial controls (accounting perspective), and IT controls (IT operations perspective). In the field of IT risk management, further, possibly overlapping categories are general IT controls, pervasive IT controls, detailed IT controls, and application controls (e.g., [Inf09, 25]). When modeling a system of internal controls, it, thus, contributes to the reduction of complexity if control objectives are properly classified by their type of control. One solution to allow for representing such types would be to offer specific meta types that are specializations of the control objective, e.g., meta types *ApplicationControlObjective* and *GeneralControlObjective* (cf. Fig. 6(a)). Again, such a solution would restrict the users to predefined, disjunct types but at the same

time promote a secure modeling. On the other hand, the control category can be represented by a meta type that allows enterprises for instantiating their own categorization as well as having overlapping categories (cf. Fig. 6(b)). Since we are not aware of a widely accepted categorization of controls, we recommend to establish a flexible meta type *ControlCategory* which enables enterprises to build their own systematization.



Figure 6: Design alternatives for the specification of control categories

Monitoring and auditing processes So far, we have abstracted from the procedural, dynamic aspects of internal controls (such as the regular monitoring and auditing of refund transactions, cf. Fig. 2). In principle, those processes exhibit characteristics similar to business processes: They consist of a set of activities following a control flow (e.g., sequence, concurrency and alternative) and are performed by organizational units (usually internal or external auditors). However, an initial analysis of the peculiarities of audit processes reveals an interesting difference: While business processes are performed on a regular basis, usually in high frequency each day, audit processes, in contrast, may have very different frequencies that range from an event-driven instantiation to several instantiations per month or year to a continuous (since automated) execution. Also, audit processes differ from business processes in that they are specifically designed to run “outside” of a firm’s regular operations with the intention to control a particular “audit object” such as a business process, a record of transactions etc.

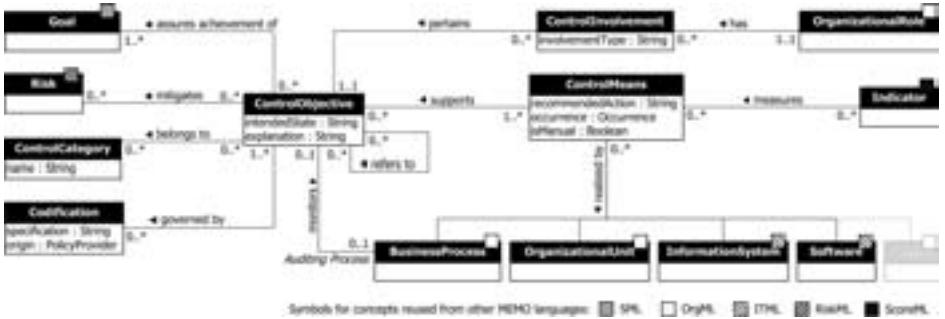


Figure 7: Specification of modeling concepts for internal controls modeling (meta model specified using the MEMO Meta Modeling Language [Fra08])

In a sense, audit processes are, therefore, associated with the specific audit object(s). However, present process modeling approaches—to our knowledge—do not provide modeling constructs to represent such qualified associations between processes (e.g., “controls”,

“audits”) or frequencies. Hence, we identify this as an open design issue for future research which may require further exchange with domain experts. As a workaround, we propose to utilize those business process modeling approaches for modeling auditing and monitoring processes that provide time-related events. As the MEMO business process modeling language includes differentiated concepts for temporal events, it seems feasible to reuse the *BusinessProcess* concept as is shown in Fig. 7 (see the role *Auditing Process*). The meta model consolidates the discussed design decisions and provides a foundation for future work on internal controls modeling.

7 Conclusions

This paper investigates the potentials of an enterprise modeling approach to audit risk assessment and proposes conceptualizations for modeling constructs as enhancements to enterprise modeling to support audit risk assessment. The approach is based on the assumption that enterprise models provide a substantial foundation for audit risk assessment in that they represent the organizational context (Req. 1) and support multiple perspectives (Req. 5).

Our contribution in this paper is threefold: First, we direct the discussion on supporting audit risk assessment through conceptual models to include further abstractions (i.e., goal models, role models and (IT) resource models) common to enterprise modeling—beyond business process modeling. Second, we refine and structure the technical terminology in the auditing domain by reconstructing key concepts. Third, we prepare for further research on a domain-specific modeling language for audit risk assessment by reflecting key considerations and decisions pertaining to internal controls modeling.

In this paper, we focus on language concepts—especially with regard to the internal control system, its justification, and implementation (cf. Req. 2–4)—and discuss design alternatives for corresponding modeling constructs as part of a design research project to develop a comprehensive enterprise modeling method for governance, risk, and compliance. However, developing a *method* requires further considerations besides language design. On the one hand, a method has to account for corresponding diagram types targeted at the perspectives of stakeholders involved in audit risk assessment (e.g., a dedicated internal control diagram as indicated in Fig. 2). On the other hand, a method demands for a process model that guides auditors and stakeholders in applying and interpreting the language concepts, for instance, for certain types of analyses. The effective and efficient use of such a method also presupposes the availability of a modeling tool that implements both the enterprise modeling method as well as the control-related enhancements. Such a methodical support remains on our research agenda and includes the MEMO modeling tool [GF10] (<http://www.wi-inf.uni-due.de/fgfrank/memocenter-en>).

References

- [ABC04] P. Alencar, J. E. Boritz, and Carla Carnaghan. The relative merits of diagrammatic versus textual representations: a literature review of theoretical and empirical perspectives. Working paper, University of Waterloo, 2004.
- [ABC08] P. Alencar, J. E. Boritz, and Carla Carnaghan. Business Modeling to Improve Auditor Risk Assessment: An Investigation of Alternative Representations. In *Proceedings of the 14th Annual International Symposium on Audit Research, ISAR 2008, Los Angeles, California, USA, May 30–31, 2008*, Los Angeles, CA, May 2008. American Accounting Association.
- [AJKL06] Rakesh Agrawal, Christopher M. Johnson, Jerry Kiernan, and Frank Leymann. Taming Compliance with Sarbanes-Oxley Internal Controls Using Database Technology. In Ling Liu, Andreas Reuter, Kyu-Young Whang, and Jianjun Zhang, editors, *Proceedings of the 22nd International Conference on Data Engineering, ICDE 2006, 3–8 April 2006, Atlanta, GA, USA*, page 92. IEEE Computer Society, 2006.
- [ALM02] T. S. Amer, R. F. Lucy, and J. Maris. The effects of system representation on the efficiency and effectiveness of control and maintenance reviews. Technical report, Northern Arizona University, Flagstaff, AZ, 2002.
- [Car06] Carla Carnaghan. Business process modeling approaches in the context of process level audit risk assessment: An analysis and comparison. *Int J of Account Inf Syst*, 7(2):170–204, 2006.
- [CL03] K. T. Chen and Ronald M. Lee. Knowledge-based evaluation of internal accounting control systems — a pattern recognition approach. In *Proceedings of the American Accounting Association Conference*, Honolulu, HI, 2003.
- [CS02] Margaret Crawford and William Stein. Auditing Risk Management: Fine in Theory but who can do it in Practice? *International Journal of Auditing*, 6(2):119–131, 2002.
- [Dam05] Marios Damianides. Sarbanes-Oxley and IT Governance: New Guidance on IT Control and Compliance. *Inf. Sys. Manage.*, 22(1):77–85, 2005.
- [DCH05] Cheryl Lynn Dunn, J. Owen Cherrington, and Anita Sawyer Hollander. *Enterprise Information Systems: A Pattern-Based Approach*. McGraw-Hill Irwin, Boston, MA, 3. ed., internat. ed. edition, 2005.
- [Dun06] Cheryl L. Dunn. Business Process Modeling Approaches in the Context of Process Level Audit Risk Assessment: An Analysis and Comparison : Discussion Comments. *International Journal of Accounting Information Systems*, 7(2):205–207, 2006.
- [FHK09] Ulrich Frank, David Heise, and Heiko Kattenstroth. Use of a Domain Specific Modeling Language for Realizing Versatile Dashboards. In Matti Rossi et al, editor, *Proceedings of the 9th OOPSLA workshop on domain-specific modeling (DSM)*, 2009.
- [Fra02] Ulrich Frank. Multi-Perspective Enterprise Modeling (MEMO): Conceptual framework and modeling languages. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS)*, pages 72–82, Honululu, 2002.
- [Fra08] Ulrich Frank. The MEMO Meta Modelling Language (MML) and Language Architecture. ICB Research Report 24, Institute for Computer Science and Business Information Systems (ICB), Duisburg-Essen University, Germany, 2008. http://www.icb.uni-due.de/fileadmin/ICB/research/research_reports/ICBReport24.pdf.

- [GF10] Jens Gulden and Ulrich Frank. MEMOCenterNG. A full-featured modeling environment for organisation modeling and model-driven software development. In Pnina Soffer and Erik Proper, editors, *Proceedings of the CAiSE Forum (Short Papers and Tool Demonstrations) of the 22nd International Conference on Advanced Information Systems Engineering (CAiSE'10), 7–11, June 2010*, Lecture Notes in Business Information Processing, Berlin, Heidelberg, 2010. Springer. To appear.
- [GHSW08] Guido Governatori, Jörg Hoffmann, Shazia Wasim Sadiq, and Ingo Weber. Detecting Regulatory Compliance for Business Process Models through Semantic Annotations. In Danilo Ardagna, Massimo Mecella, and Jian Yang, editors, *Business Process Management Workshops*, volume 17 of *Lecture Notes in Business Information Processing*, pages 5–17. Springer, 2008.
- [GMS06] Guido Governatori, Zoran Milosevic, and Shazia Sadiq. Compliance checking between business process and business contracts. In *Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference (EDOC'06), Hong Kong, China, Oct 16, 2006*, pages 16–20, Los Alamitos, CA, USA, October 2006. IEEE Computer Society.
- [GSF04] Ulric J. Gelinas, Steve G. Sutton, and Jane Fedorowicz. *Business processes and information technology*. South-Western Thomson Learning, Mason, Ohio, 2004.
- [Inf09] Information Systems Audit and Control Association. IS Standards, Guidelines and Procedures for Auditing and Control Professionals. Rolling Meadows, 2009.
- [IT 07] IT Governance Institute, editor. *CobiT 4.1: Framework, Control Objectives, Management Guidelines, Maturity Models*. IT Governance Institute of the Information Systems Audit and Control Association, Rolling Meadows, 2007.
- [Kir05] L. Kirchner. Cost Oriented Modelling of IT-Landscapes: Generic Language Concepts of a Domain Specific Language. In J. Desel and U. Frank, editors, *Proceedings of the Workshop on Enterprise Modelling and Information Systems Architectures (EMISA 2005)*, pages 166–179, 2005.
- [KMS07] Dimitris Karagiannis, John Mylopoulos, and Margit Schwab. Business Process-Based Regulation Compliance: The Case of the Sarbanes-Oxley Act. In *Requirements Engineering*, pages 315–321. IEEE, 2007.
- [Lan05] M. Lankhorst. *Enterprise Architecture at Work : Modelling, Communication and Analysis*. Springer, Berlin, 2005.
- [LSG08] Ruopeng Lu, Shazia Wasim Sadiq, and Guido Governatori. Measurement of Compliance Distance in Business Processes. *Inf. Sys. Manag.*, 25(4):344–355, 2008.
- [Mai00] Steven Maijor. The Internal Control Explosion. *International Journal of Auditing*, (4):101–109, 2000.
- [McC79] W. E McCarthy. An entity-relationship view of accounting models. *The Accounting Review*, 54(4):667–686, 1979.
- [McC82] W. E McCarthy. The REA accounting model: A generalized framework for accounting systems in a shared data environment. *The Accounting Review*, 57(3):554–578, 1982.
- [Moe08] Robert R. Moeller. *Sarbanes-Oxley Internal Controls : Effective Auditing with AS5, CobiT and ITIL*. John Wiley & Sons, Hoboken, NJ, 2008.

- [NS07] Kioumars Namiri and Nenad Stojanovic. Applying Semantics to Sarbanes Oxley Internal Controls Compliance. In Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, and Marc Ronthaler, editors, *GI Jahrestagung*, volume 109 of *Lecture Notes in Informatics*, pages 222–226. GI, 2007.
- [PP97] Jens Pitthan and Mathias Philipp. Einsatz von Petri-Netzen für die Aufnahme, Dokumentation und Analyse Interner Kontrollsysteme im Rahmen der Jahresabschlußprüfung. In W. Stucky and U. Winand, editors, *Petri-Netze zur Modellierung verteilter DV-Systeme – Erfahrungen im Rahmen des DFG-Schwerpunktprogramms “Verteilte DV-Systeme in der Betriebswirtschaft”*, number 350, pages 87–104. Institut für Angewandte Informatik und Formale Beschreibungssprachen (AIFB), Universität Karlsruhe (TH), Karlsruhe, Germany, March 1997.
- [RBGR06] P. Rikhärdsson, P. Best, P. Green, and M. Rosemann. Business Process Risk Management, Compliance, and Internal Control: A Research Agenda. Technical report, Department of Business Studies, Management Accounting Research Group, Aarhus School of Business, Aarhus, Denmark, 2006.
- [Sch92] A.-W. Scheer. *Architecture of Integrated Information Systems : Foundations of Enterprise Modelling*. Springer, Berlin, 1992.
- [Sch00] A.-W. Scheer. *ARIS : Business Process Modeling*. Springer, Berlin, Heidelberg, 3 edition, 2000.
- [SGN07] Shazia Wasim Sadiq, Guido Governatori, and Kioumars Namiri. Modeling Control Objectives for Business Process Compliance. In Gustavo Alonso, Peter Dadam, and Michael Rosemann, editors, *BPM 2007*, volume 4714 of *Lecture Notes in Computer Science*, pages 149–164, Berlin, 2007. Springer.
- [SHF10] Stefan Strecker, David Heise, and Ulrich Frank. RiskM: A multi-perspective modeling method for IT risk assessment. *Information Systems Frontiers. Accepted for publication in the Special Issue on Governance, Risk and Compliance Applications in Information Systems*, 2010.
- [SLKP07] Amadou Sienou, Elyes Lamine, Achim Karduck, and Hervé Pingaud. Conceptual Model of Risk: Towards a Risk Modelling Language. In Mathias Weske, Mohand-Said Hadid, and Claude Godart, editors, *WISE Workshops*, volume 4832 of *Lecture Notes in Computer Science*, pages 118–129. Springer, 2007.
- [SP02] L. F. Spira and M. Page. Risk Management — The reinvention of internal control and the changing role of internal audit. *Accounting, Auditing & Accountability Journal*, 16(4):640–661, 2002.
- [The92] The Committee of Sponsoring Organizations of the Treadway Commission. Internal Control — Integrated Framework, September 1992.
- [The04] The Committee of Sponsoring Organizations of the Treadway Commission. Enterprise Risk Management – Integrated Framework (Executive Summary), September 2004.
- [The09] The Committee of Sponsoring Organizations of the Treadway Commission. Guidance on Monitoring Internal Control Systems: Introduction. <http://www.coso.org>, 2009.
- [The10] The Committee of Sponsoring Organizations of the Treadway Commission. What is internal control? <http://www.coso.org/resources.htm>, September 2010.
- [zMR05] M. zur Muehlen and M. Rosemann. Integrating Risks in Business Process Models. In *Proceedings of the 16th Australasian Conference on Information Systems (ACIS 2005)*, pages 62–72, 2005.

Konstruktion eines adaptiven Referenzmodells für den ambulanten Sektor

Hannes Schlieter, Maik Bürger, Werner Esswein

Lehrstuhl für Wirtschaftsinformatik, insb. Systementwicklung

TU Dresden

Münchner Platz 3

01062 Dresden

Hannes.Schlieter@tu-dresden.de

Maik.Buerger@mailbox.tu-dresden.de

Werner.Esswein@tu-dresden.de

Abstract: Die Qualität von Produkten und Dienstleistungen gewinnt in der Verbraucherwahrnehmung immer mehr an Bedeutung. Aufgrund von politisch-rechtlichen Änderungen, der zunehmenden Emanzipation der Patienten und nicht zuletzt dem Streben nach evidenzbasierter Medizin hält das Thema Qualitätsmanagement in den ambulanten Sektor Einzug. Im vorliegenden Aufsatz wird die Erarbeitung eines Referenzmodells für den ambulanten Sektor präsentiert. Dadurch soll die Einführung eines zertifizierungsfähigen Qualitätsmanagementsystems in der Praxislandschaft zum einen beschleunigt, zum anderen durch die Bereitstellung von Best-Practice-Lösungen qualitativ verbessert werden. Die Untersuchung basiert auf einer empirischen Prozessanalyse in Form einer qualitativen Befragung von Ärzten und der Auswertung bestehender Dokumentationen in der Praxis. Dieser Arbeit liegen die Theorien der Referenzmodell- sowie Qualitätsforschung zugrunde. Der vorliegende Aufsatz bietet somit einen Ansatz für die referenzmodellgestützte Umsetzung eines Qualitätsmanagementsystems (QMS) in die medizinische Praxis des ambulanten Sektors.

1 Einleitung

Durch die 72. Gesundheitsministerkonferenz, welche 1999 stattfand, wurden Ziele für eine einheitliche Qualitätsstrategie im Gesundheitswesen festgelegt. Speziell für den ambulanten Sektor wurde als Ergebnis ein systematisches Qualitätsmanagement festgeschrieben [Ku05].

Der Gesetzgeber verankerte im Gesetz zur Modernisierung der gesetzlichen Krankenversicherung (GMG) "strukturelle Reformen, um die Qualität der medizinischen Versorgung zu verbessern" [GM03]. Das Gesetz greift die Erfahrungen, die im Bereich des stationären Sektors gemacht wurden, auf und erweitert die Pflicht der niedergelassenen Ärzte "...sich nun nicht mehr nur an sektorübergreifenden Aktivitäten (zu) beteiligen, sondern praxisintern Qualitätsmanagement (QM) ein(zu)föhren und weiter(zu)entwickeln" [DG04].

Ein Qualitätsmanagementsystem (QMS) ist der Bestandteil des Managementsystems zum Leiten und Lenken einer Organisation bezüglich der Qualität [DI05]. Ziel eines QMS ist die Erlangung der Fähigkeit eines Unternehmens, die Qualität ihres Produktes oder ihrer Dienstleistung entsprechend ihrer gesetzten Ziele beeinflussen zu können. Das GMG hat der Gemeinsame Bundesausschuss¹ durch eine Richtlinie konkretisiert. Die Richtlinie definiert die Grundanforderungen des QMS zum einen hinsichtlich aufbauorganisatorischer Elemente, zum anderen bezüglich ablauforganisatorischer Elemente der Praxisorganisation.

Des Weiteren werden die Instrumente für die Umsetzung und ein Zeitplan für die Einführung festgelegt. Dieser gibt einen Pfad vor, nach dem die Praxen innerhalb von vier Jahren die Grundanforderungen des QMS zu erfüllen haben [BG05]. Mit Beschluss 2005 soll dieser Prozess bis zum 31.12.2009 abgeschlossen sein. Aufgrund der bisher fehlenden Sanktionierungsmaßnahmen und dem Fakt, dass gute Qualität von den Leistungsträgern nur in geringem Maße belohnt wird, haben bisher ca. 20 % der Arztpraxen noch kein QMS implementiert [SG09], [Wa09].

Die Aufnahme in der Ärzteschaft ist dabei durchaus gespalten, dies konnte eine Studie der Stiftung Gesundheit und die geführten Interviews zeigen. Immer noch sehen viele Ärzte in der Implementierung eines QMS eher einen zusätzlichen, unvergüteten Mehraufwand als eine Chance, die Transparenz, Qualität und Kontrolle über die Praxisprozesse zu erhöhen [SG08]. Die bisher am meisten eingesetzten QMS in der Praxislandschaft sind Systeme, die den Normstandard der DIN EN ISO 9000-Familie (Deutsches Institut für Normung e.V.; Europäische Normen; International Organization for Standardization) zur Grundlage haben (24,1 %), und das QMS "Qualität und Entwicklung in der Praxis" (QEP, 26,4 %). Das QEP ist ein von den Kassenärztlichen Vereinigungen speziell auf Praxisanforderungen ausgerichtetes QMS, das zunehmend an Bedeutung gewinnt. Neben den fünf am weitesten verbreiteten QMS existieren darüber hinaus noch circa vierzig weitere QMS, die im ambulanten Sektor Anwendung finden. Sie sind zum größten Teil Derivate oder Eigenentwicklungen [SG09]. Ein Kernbestandteil eines QMS ist typischerweise die Dokumentation der wichtigsten Prozesse des Unternehmens respektive der Praxen, um über das Bewusstsein über die Prozesse eigene Zielstellungen zu definieren, Maßnahmen zu initiieren und deren Erreichung messen zu können. Die Anforderungen an die Dokumentation der Prozesse variieren dabei je nach QMS und Zielstellung, die mit dem System verbunden sind.

¹ Der Gemeinsame Bundesausschuss (G-BA) ist das oberste Beschlussgremium der gemeinsamen Selbstverwaltung der Ärzte, Zahnärzte, Psychotherapeuten, Krankenhäuser und Krankenkassen in Deutschland

Die Norm Familie der DIN EN ISO 9000ff. – folgend als DIN9000 bezeichnet – lässt bspw. die notwendige Zahl bzw. Form der Prozessdokumentation offen. In der DIN9000 wird beschrieben, dass die "Dokumentation der Wechselwirkung der Prozesse des Qualitätsmanagementsystems" Bestandteil eines QMS ist [DI08]. Andere Modelle, die den Fokus eher auf die Konformität zum GMG haben, geben eine konkrete Zahl an Prozessen vor, die für das QMS dokumentiert sein müssen. Eine Übersicht zu den Vorgaben zur Prozessdokumentation für die QMS ist in nachfolgender Tabelle dargestellt (siehe Tab. 1). Der tatsächliche Nutzen, im Sinne eines echten QMS, ist bei der Vorgabe konkreter Prozesse, einer fixen Anzahl zu erhebender Prozesse oder fertig ausgefüllter QM-Handbücher anzuzweifeln. Daraus wird das Bestreben sichtbar, Ressourcen, die für die Einführung eines QMS aufgewendet werden, gering zu halten.

Tabelle 1: Übersicht verwendeter QMS im niedergelassenen Sektor [KB08], [KI05]

Name des QMS	QEP®	KPQM 2006 KVWL Praxis Qualitätsmanagement	KTQ® Kooperation und Transparenz im Gesundheitswesen	EPA European Praxis Assessment	DIN9000
Entwickler	KBV/KV	Prinarum GmbH im Auftrag der KV WL	KTQ® gGmbH	TOPAS Europe, Bertelsmann Stiftung	Normreihe der ISO/DIN/EN
Bestandteile des QM-Verfahrens	Abläufe, Tätigkeiten sowie die Rahmenbedingungen aller Praxisbereiche	Vorliegen von Ablaufschemata, Verfahrensanweisungen von mindestens 10 Praxisprozessen mit Bezug zu: <ul style="list-style-type: none">• Patienten (Diagnostik / Therapie)• Mitarbeitern u. Personalführung• Administration	Tätigkeiten und Rahmenbedingungen der Praxis	Betrachtung von Abläufen in der Praxis	Rahmen der DIN EN ISO 9000 ff. <ul style="list-style-type: none">• Qualitätsmanagementsystem• Verantwortung der Leistung• Management der Ressourcen• Produktrealisierung• Messung, Analyse und Verbesserung
Bewertungsinstrumente	Qualitätszielkatalog		Bewertungskatalog	Frage- und Assessmentbögen, Benchmarking Datenbank	max. durch externe Beratung
Anforderung an Prozessdokumentation	Prozessorientierter Ansatz QEP Manual mit Musterdokumenten z. B. Checklisten, Ablauf / Prozessbeschreibungen, interne Regellungen	Prozessorientierter Ansatz Klare Forderung zur Dokumentation von Pflichtprozessen mit Hilfe von Flow-Charts	Reines Bewertungsverfahren auf Basis einer Gesamtpunktzahl implizite Forderung nach dokumentierten Prozessablaufbeschreibungen	Prozessorientiertes Verfahren inkl. Checklisten und Ablaufbeschreibungen	Prozessorientierung im Hauptkapitel "Produktrealisierung", störanfällige Prozesse in Form von Ablaufbeschreibungen zu hinterlegen

Die Zielstellung ist somit, ein QMS kostengünstig, zeitlich effizient aber auch bedarfssadäquat zu implementieren. Der Arzt bzw. der beauftragte Dienstleister sollen nicht durch eine zu strikte Lösungsvorgabe bei der Umsetzung behindert aber dennoch unterstützt werden.

Die fachkonzeptuelle Modellierung bietet einen Ansatz, um komplexe Informationssysteme durch diagrammatische Beschreibungen in verständlicher Form abzubilden. Speziell die Konstruktion komplexer Prozessmodelle ist ein aufwändiger Prozess. Angesichts dessen wird seit längerem der Ansatz der Wiederverwendung verfolgt [Ha99]. Bei der fachkonzeptuellen Modellierung erfolgt die Wiederverwendung mithilfe von Referenzmodellen [Sc98]. Durch ihre Ausrichtung an Common-Practice- bzw. Best-Practice-Lösungen geht zudem die Erwartung einher, das Gestaltungsrisiko zu minimieren und ihre Akzeptanz zu erhöhen [ELS10].

Auf Basis von Referenzmodellen können Lösungen für den Informationssystem-Gestalter vorgedacht und mit Hilfe von unterstützenden Konstruktionstechniken (Referenzmodelltechnik) die Erstellung von abgeleiteten Modellen – im weiteren Verlauf als Anwendungsmodelle bezeichnet – erleichtert werden. Konstruktionstechniken (Adaptionsmechanismen) werden in Referenzmodellen implementiert, um die gezielte und methodische Wiederverwendung, d. h. die Adaption des Referenzmodells, zu unterstützen. Konstruktionstechniken sind Regeln, welche die Art und Weise der Wiederverwendung von Modellinhalten und deren Anpassung und Erweiterung definieren [Br03], [BB04]. Sind in einem Referenzmodell diese Konstruktionstechniken implementiert, wird auch von einem adaptiven Referenzmodell gesprochen [De06].

Das Ziel der vorliegenden Forschungsarbeit liegt in der Schaffung eines Instruments zur Unterstützung des Informationssystemgestalters bei der Konstruktion eines Anwendungsmodells durch ein adaptives Referenzmodell. Grundlage für die Arbeit ist eine empirische Prozessanalyse, bei der sechs niedergelassene Ärzte, je zweimal befragt wurden. Weiterhin wurden die bestehenden Dokumentationen der Ärzte mit herangezogen. Die Prozessanalyse bildet die Grundlage für das zu entwickelnde Referenzmodell. Dieses soll die Einführung bzw. Weiterentwicklung eines (zertifizierungsfähigen) Qualitätsmanagementsystems in der Praxislandschaft zum einen beschleunigen, zum anderen qualitativ verbessern. Die Gestaltungsfreiheit soll jedoch nicht zu sehr eingeschränkt werden.

Nachdem in der Einleitung die Motivation sowie eine Einordnung der Arbeit vorgenommen wurden, folgen im Kapitel zwei die Grundlagen zur Referenzmodellierung und Referenzmodellkonstruktion. Danach werden im Kapitel drei der Rahmen für die Prozessanalyse gelegt und die Ergebnisse der Analyse vorgestellt. Sie sind die Voraussetzung für das im Kapitel vier zu erarbeitende Referenzmodell für den ambulanten Sektor. Der Beitrag schließt mit einer kritischen Würdigung und einem Ausblick auf weiteren Forschungsbedarf.

2 Grundlagen

In diesem Kapitel werden die Grundlagen zur Referenzmodellierung gelegt. Ein eindeutiges Verständnis des Begriffes Referenzmodell existiert in der Wirtschaftsinformatik bis zum aktuellen Zeitpunkt nicht. Je nach unterstelltem Verständnis des Modellbegriffs (abbildungs- oder konstruktionsorientiert) existieren in der Literatur Definitionen, welche häufig wiederverwendet werden. FETTKE und LOOS stellen Definitionen nach HARS, SCHÜTTE sowie VOM BROCKE in das Zentrum der Begriffsdiskussion [FL04], [Ha94], [Sc98], [Br03].

HARS unterstellt seinem Modellbegriff eine ausdrückliche Abbildungsbeziehung. Er spricht den Modellen, die für den Entwurf anderer Modelle herangezogen werden, den Begriff des Referenzmodells zu. Dem Referenzmodell werden die Eigenschaften: Allgemeingültigkeit, Anpassbarkeit und die Anwendbarkeit zugesprochen. Unter Anwendbarkeit wird dabei die Möglichkeit der Verwendung des Referenzmodells ohne spezielle Anpassungen verstanden [Ha94].

SCHÜTTE definiert das Referenzmodell als "das Ergebnis einer Konstruktion eines Modellierers, der für Anwendungssystem- und Organisationsgestalter Informationen über allgemeingültig zu modellierende Elemente eines Systems zu einer Zeit als Empfehlungen mit einer Sprache deklariert, sodass ein Bezugspunkt für ein Informationssystem geschaffen wird" [Sc98]. SCHÜTTE sieht ein Merkmal von Referenzmodellen im Gegensatz zu HARS im Empfehlungscharakter. Referenzmodelle dienen demnach als Bezugs punkt für Gestaltung von spezifischen Anwendungsmodellen [Sc98], [FL04].

VOM BROCKE greift den Referenzmodellbegriff neu auf und kritisiert die bisher typischen Eigenschaften der Allgemeingültigkeit und des Empfehlungscharakters von Referenzmodellen. VOM BROCKE identifiziert ein Spannungsfeld zwischen der Deklaration und der Adaption von Referenzmodellen [Br03]. Der Empfehlungscharakter des Modells sollte nach VOM BROCKE nicht vor seiner eigentlichen Verwendung deklariert werden [BB04]. Infolgedessen definiert VOM BROCKE ein Referenzmodell als "ein Informationsmodell, das Menschen zur Unterstützung der Konstruktion von Anwendungsmodellen entwickeln oder nutzen, wobei die Beziehung zwischen Referenz- und Anwendungsmodell dadurch gekennzeichnet ist, dass Gegenstand oder Inhalt des Referenzmodells bei der Konstruktion des Gegenstands oder Inhalts des Anwendungsmodells wieder verwendet werden" [Br03]. Als wichtigstes Kennzeichen von Referenzmodellen hebt VOM BROCKE die inhaltsbezogene Unterstützung von Konstruktionsprozessen durch Referenzmodelle hervor.

Thomas greift das von VOM BROCKE identifizierte Spannungsfeld auf und weiß auf folgende Szenarien hin:

1. Ein Referenzmodell wird als solches erstellt und deklariert, jedoch nicht als solches genutzt.
2. Ein Referenzmodell wird als solches erstellt und deklariert und auch zur Erstellung von Anwendungsmodellen genutzt.

3. Ein Modell wird zur Erstellung eines Anwendungsmodells genutzt. Das Modell wurde jedoch nicht ausdrücklich als Referenzmodell erstellt oder deklariert [Th06].

Die von THOMAS vorgestellten Szenarien der Beziehungsarten zwischen "Referenzmodell" und daraus abgeleiteten Modell werfen das Problem auf, dass ein Referenzmodell, welches als solches erstellt und deklariert, jedoch nicht als solches genutzt wird, kein Referenzmodell ist [Th06]. Demzufolge charakterisiert sich das Referenzmodell erst über die Wiederverwendung. Diese strikte Positionierung führt zu dem Problem, dass Modelle, die für die Wiederverwendung konstruiert werden, auf ihre Wiederverwendung warten müssen. Zwei Argumente sollen angeführt werden, um diese strikte Position zu entkräften. Zum einen bedingt wissenschaftliches Vorgehen auch Lösungen vorzudenken, deren Evaluierung in der Praxis noch aussteht bzw. deren Anerkennung in der Forschungsgemeinschaft erlangt werden muss [Fe06], [FB08]. Dieses Vordenken bedingt, dass auch Referenzmodelle erarbeitet, diskutiert und eventuell abgeändert werden müssen, bevor sie in der Praxis Anwendung finden können. Zum anderen fordert die gezielte und methodische Unterstützung der Wiederverwendung, dass Konstruktionstechniken in das Referenzmodell integriert werden müssen, welche diese – ausgenommen einer reinen Analogiekonstruktion – vom Anwendungsmodell unterscheiden. Somit existiert im Rahmen der Konstruktion ein Unterschied zwischen Konstruktion eines spezifischen Modells und Konstruktion eines Referenzmodells [BB04], [Br03]. Nach der Meinung der Autoren ist ein Referenzmodell auch das speziell zur Wiederverwendung konstruierte Modell, das noch Anwendung finden muss. Insbesondere wenn es auf Basis empirischer erarbeiteter Detailmodelle hervorgeht.

In das Referenzmodell müssen spezielle Sprachkonstrukte in die eigentliche Modellierungssprache einfließen, um die adaptiven Elemente beschreiben zu können. Dies führt zu einer geeigneten Referenzmodellierungsgrammatik [Br09]. In der Forschungsgemeinschaft haben sich fünf reine Konstruktionstechniken herauskristallisiert. Dies sind die Konfiguration (nicht generierende Konstruktionstechniken), Instanziierung, Freie Modifikation (bzw. Spezialisierung), Aggregation und die Analogiekonstruktion (generierende Konstruktionstechniken) [Br03], [BDK2004], [Br09], [De06]. In der Praxis kommen häufig hybride Konstruktionstechniken zum Einsatz, die ein besseres Kosten-Nutzen-Verhältnis versprechen als die Reinformen. So ist es beispielsweise möglich, nur ausgewählte Modellbereiche mit Konfigurationsmechanismen für die Wiederverwendung aufzubereiten und andere Bereiche für die Analogiekonstruktion zu belassen. In der Praxis ist dies auch vom Modellierungswerkzeug abhängig.

Aktuell hat sich noch keine standardisierte Referenzmodellierungssprache durchgesetzt [Br03], [Th06]. Im Rahmen von Arbeiten zur ARIS-Methode (Architektur Integrierter Informationssysteme) werden oft die Ereignigesteuerte Prozesskette (EPK) sowie das Entity Relationship Model (ERM) verwendet. Weiterhin kommt bei objektorientierten Referenzmodellen häufig die Unified Modeling Language (UML) zum Einsatz [Th06].

Die wesentlichen Veröffentlichungen von Vorgehensmodellen zur Referenzmodellkonstruktion gehen auf die Arbeiten von SCHÜTTE, VOM BROCKE, BECKER, SCHWEGMANN, SCHLAGHECK und DELFMANN zurück [Sc98], [BKK01], [Sc99], [Sc00], [De06]). Den Grundstein legte Schütte mit der Veröffentlichung seiner Arbeit "Vorgehensmodell zur Erstellung und Anwendung von Referenzmodellen unter Nutzung der Grundsätze ordnungsgemäßer Modellierung" [Sc98]. Das Vorgehensmodell beschreibt die Konstruktion in den Phasen Problemdefinition, Konstruktion des Referenzmodellrahmens, Konstruktion der Referenzmodellstruktur, Komplettierung und Anwendung [Sc98]. Die Phasen sind relativ allgemein gefasst, dienen aber als Grundlage für verschiedenste Arbeiten der Referenzmodellierung, so auch zur verteilten Referenzmodellierung [Br03], zum Referenzmodellmanagement [Th06] und zur adaptiven Referenzmodellierung [De06].

Es bildet weiterhin Grundlage für das Vorgehen zur objektorientierten Referenzmodellierung nach SCHWEGMANN und SCHLAGHECK [Br03], [Th06], [Sc99], [Sc00]. Im Rahmen einer nicht-objektorientierten Referenzmodellierung greifen BECKER ET AL. in ihrer "Methodik zur konfigurativen Referenzmodellierung" die Untersuchungen von SCHÜTTE auf und organisieren die Ergebnisse in einen Projektrahmen mit Empfehlungen zur Referenzmodellkonstruktion [BDK02]. Der aktuelle Stand dieses Vorgehensmodells kondensiert DELFMANN [De06]. Es beschreibt die Phasen:

1. Projektziele definieren
2. Referenzmodellierungstechnik definieren
3. Referenzmodellierungstechnik implementieren
4. Referenzmodell erstellen
5. Referenzmodellevaluierung
6. Referenzmodell vermarkten

Ziel des Ansatzes von DELFMANN ist die Beschreibung eines Vorgehens für die Konstruktion und Anwendung eines adaptiven Referenzmodells, d. h., eines Referenzmodells, das um Konstruktionstechniken erweitert wurde [De06]. Dieses Vorgehen bildet die Grundlage für das im Kapitel 4 zu konstruierende Referenzmodell.

3 Prozessanalyse

Nachdem die Grundlagen zur Referenzmodellierung gelegt wurden, wird im Folgenden die Prozessanalyse und deren Ergebnisse vorgestellt. Die Ergebnisse dieser Analyse bilden die Grundlage für das zur Wiederverwendung bestimmte Referenzmodell. Die Pfeiler der Prozessanalyse bilden zum einen die qualitative Befragung [BD09] von Hausärzten, Fachärzten und Zahnärzten, zum anderen die Aufarbeitung bestehender Dokumentationen in den untersuchten Praxen.

3.1 Vorgehen

In der Phase der Entwicklung des Interviewleitfadens haben sich zwei prinzipielle Vorgehen der Analyse herauskristallisiert. In Abstimmung mit dem Arzt und dem Charakter der Praxis wurde die Wahl getroffen, nach welcher Methode die Prozesse erarbeitet werden. Zu Beginn eines jeden Interviews wurden vorgehensunabhängig die allgemeinen Daten der Praxen erfasst (siehe Tab. 2).

Tabelle 2: Niederlassungsprofil der befragten Ärzte

Niederlassung	Facharzt für Neurologie & Psychiatrie	Facharzt für Allgemeinmedizin (Hausarzt)	Facharzt für Kardiologie; Facharzt für Neurologie & Psychiatrie	Zahnarzt
Anzahl Ärzte/Mitarbeiter	1/2	2/4	2/8	1/4
Anzahl Interviews	3	2	2	2
Bestellpraxis ²	Ja	Nein	Ja	Ja
Anteil Spontanbesuche	< 5%	> 95%	< 5%	< 3%
Leistungen (Auszug)	Elektromyographie Elektroneurographie Hirnstrommessung	Quantitative Laborschnelltests EKG Lungenfunktionstest 24h Blutdruckmessung Wundversorgung	Quantitative Laborschnelltests Ruhe-EKG Lungenfunktionstest Fahrradergonometrie Langzeit-Blutdruck-Messung Sonographie Herz Sonographie (Organ/Gefäße) Psychiatrische, neurologische und psychotherapeutische Behandlung Hirnstrommessung	Prophylaxe Computergestützte Keramik-Restauration Bleaching professionelle Zahnreinigung Röntgen

Die erste Methode basiert auf der Erfassung der Diagnosen, die einen Großteil der Prozesse in der Praxis umfassen. Dazu wurden annotierte Flussdiagramme aufgenommen, die um Beschreibungen im Fließtext ergänzt wurden. Die Wahl des Flussdiagramms für die Analyse erfolgte aufgrund der z. T. komplexen Problemsituationen, die im Gespräch relativ schnell erfasst und in verständlicher Form modelliert werden mussten. Dieses Vorgehen wurde hauptsächlich bei der Befragung der Bestellpraxen angewandt, um so Prozesse zu identifizieren, welche oft wiederholt und somit im weiteren Verlauf der Analyse als Standard-Prozesse bewertet werden können.

² Eine Bestellpraxis charakterisiert einen Typ von Arztpraxis, in welcher Patienten zu einem vorher vereinbarten Termin für eine bestimmte Behandlung erscheinen.

Die zweite Methode basiert auf der direkten Aufnahme des Leistungskataloges der Praxis. Dieses Vorgehen wurde hauptsächlich bei der Befragung der Hausarztpraxen angewandt, da hier viele Standardprozesse im Praxisalltag vorherrschen, hingegen der Leistungskatalog sehr heterogen ist.

Nachdem die Prozesse identifiziert werden konnten, wurden diese Prozesse sukzessive detailliert und ergänzt. Als Befragungsrahmen dienten dazu typische Prozessmerkmale [BK08], die im Interview abgefragt wurden:

- Prozessziel (Wirkung)
- Zeitliche, sachlogische Abfolge von Aktivitäten (Synonym: Funktionen)
- Prozessinput (Menschen, Räume, Geräte, Informationsobjekte)
- Prozessoutput (Ergebnis, Informationsobjekte)
- Prozessoutcome (insb. bei medizinischer Maßnahme bzw. Therapie)

Am Beispiel des Prozesses der Elektrokardiographie (EKG) soll dies im Folgenden verdeutlicht werden. Zunächst wurde der für den Prozess notwendige Prozessinput erfragt und erfasst. Dies sind beim EKG die Arzthelferin und der Arzt sowie die Räumlichkeit mit entsprechenden Sitz- und Liegemöglichkeiten, die notwendigen Apparate, der Patient und nicht zuletzt Informationsobjekte (bspw. analoge oder digitale Patientenakte). Als Prozessziel wurde die leitliniengerechte Versorgung und Zufriedenheit der Patienten benannt. Die zeitlich-sachlogische Abfolge von Aktivitäten, als Beispiel die Ableitung der Herzspannungskurve, wurde anhand einer bestehenden Dokumentation in Form eines annotierten Flussdiagramms aufgenommen und bei fachlichen Unklarheiten jeweils mit dem Interviewpartner diskutiert (vgl. Abb.1). Die Prozesserfassung schließt mit der Benennung des Prozessoutputs ab. Dies ist der EKG des Patienten in analoger und bzw. oder in digitaler Form. Da keine direkte therapeutische Wirkung mit dieser Untersuchung verbunden ist, wurde kein Outcome beschrieben. Eine stichprobenhafte Befragung des Patienten könnte zumindest über die Dimension Patientenzufriedenheit Aufschluss geben.

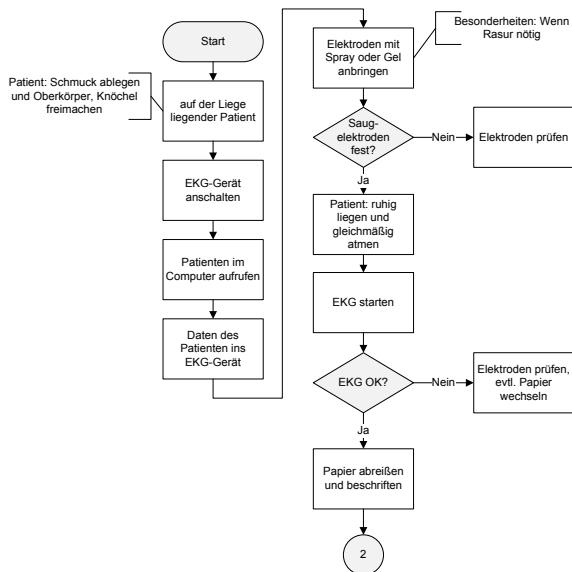


Abbildung 1 : Flussdiagramm zum EKG-Prozess (Ausschnitt)

Der direkten Prozessaufnahme beim Interviewpartner folgte, als Grundlage für die anschließende Referenzmodellkonstruktion, eine Modellierung der erhobenen Datenbasis in der Zielsprache des Referenzmodells in Form von annotierten UML-Aktivitätsdiagrammen (vgl. Abb. 2). Die Annotationen sind eine Voraussetzung für die spätere Evaluation von Konfigurationsparametern und generischer Modellbestandteile des Referenzmodells.

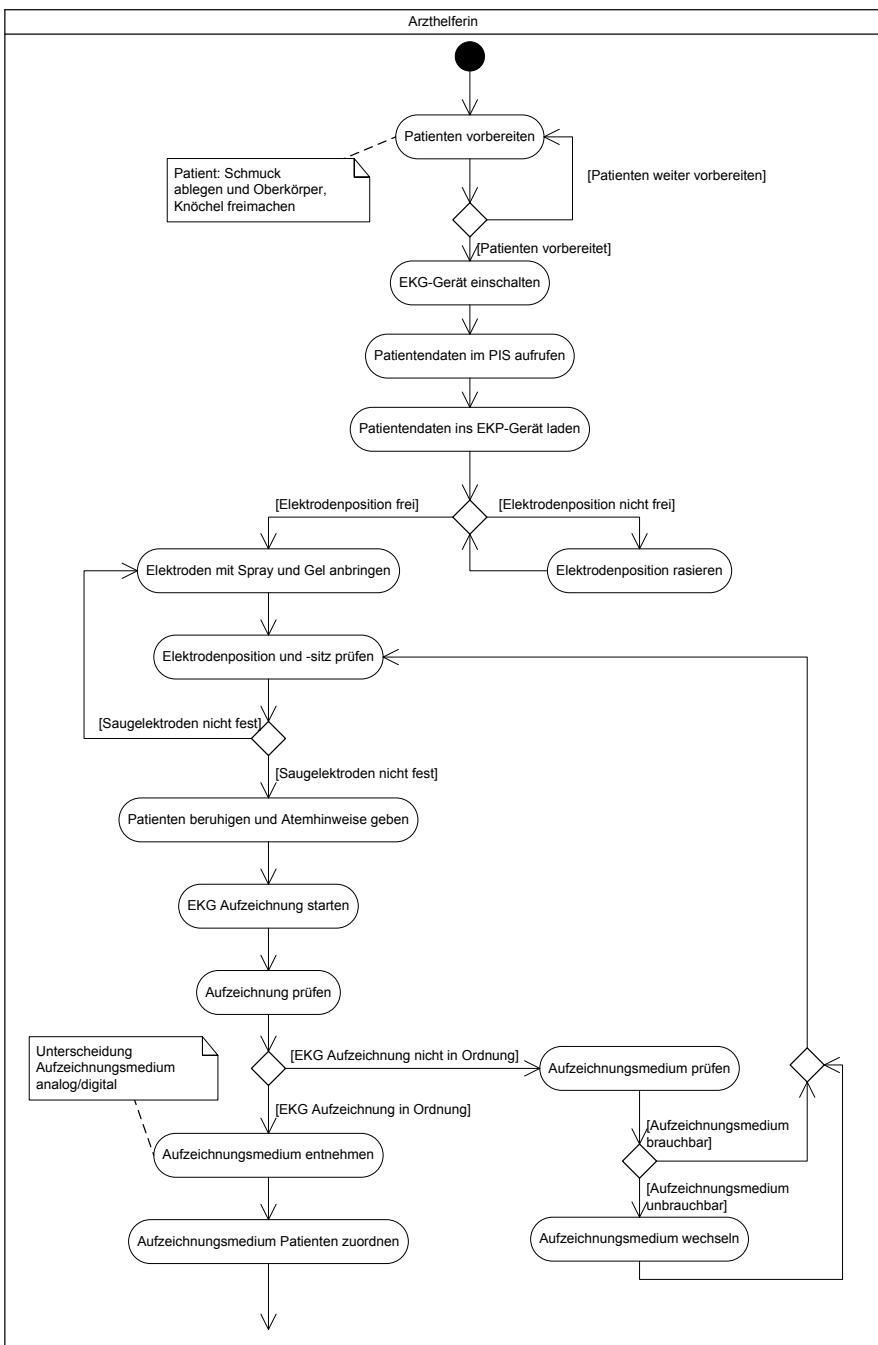


Abbildung 2: Modell des EKG-Prozesses in Form eines annotierten UML-Aktivitätsdiagramms (Ausschnitt)

3.2 Ergebnisse der Prozessanalyse

Zwei Ergebnisse sind nach der Prozessanalyse hervorzuheben. Aus der Fülle der aufgenommenen Prozesse wurde, voreiligend auf den zu entwickelnden Referenzmodellrahmen, ein Ordnungsrahmen für Prozesse des ambulanten Sektors erarbeitet (vgl. Abb. 3). Die unterschiedlichen Wertschöpfungsbereiche (Kernprozesse, Unterstützungsprozesse und Managementprozesse) werden von den in der Prozessanalyse erarbeiteten Prozessen ausgefüllt. Die darauf basierenden Detailmodelle dienen als Grundlage für die Konstruktion der Referenzprozesse.

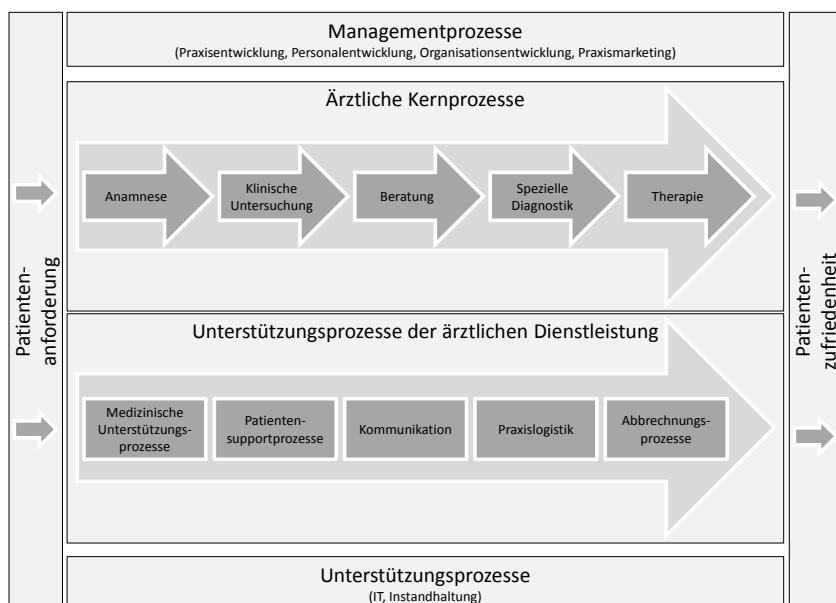


Abbildung 3: Ordnungsrahmen der ärztlichen Dienstleistung

Ein weiteres Ergebnis der Prozessanalyse sind die Erkenntnisse über die von den Autoren als Standardprozesse bezeichneten Abläufe. Diese sind zum Beispiel: Patientenaufnahme, Kommunikation im Tagesgeschäft (u. a. Krankenkassen, Kassenärztliche Vereinigungen), Notfallpatientenbehandlung, Stammdatenverarbeitung usw. Die genannten Prozesse finden in allen untersuchten Praxen Verwendung und bilden, wie auch der Ordnungsrahmen, die Grundlage für die Referenzmodellkonstruktion. Das Ermitteln der Standardprozesse stellte im Verlauf der Prozessanalyse eine besondere Schwierigkeit dar, da zum Teil relativ verschiedene Fachgebiete in der Analyse betrachtet wurden. Die hier angegebenen Standardprozesse sind somit als kleinste gemeinsame Menge ähnlicher Prozesse zu verstehen.

4 Vorgehen zur Konstruktion des adaptiven Referenzmodells

Als roter Faden für die Entwicklung des Referenzmodells wurde das "Vorgehensmodell zur Konstruktion adaptiver Referenzmodelle" herangezogen [De06]. Die Anforderungen, die zu der Entscheidung führten, sind einerseits der hohe Detaillierungsgrad der Methode, andererseits die Unabhängigkeit von der Modellierungssprache. Zudem skizziert Delfmann in seiner Arbeit, wie Konstruktionstechniken sowohl in Modelle, die Verhaltensaspekte als auch Modelle, die strukturelle Aspekte beschreiben implementiert werden können. Projektspezifische Anpassungen wurden nur vorgenommen, wenn die Zielrichtung des Projekts den Vorgaben des Vorgehensmodells entgegenstand.

4.1 Projektziele definieren

Im ersten Schritt des Vorgehensmodells von DELFMANN sind die Verwertungsstrategie der Ergebnisse, die Darlegung der Entwicklungsperspektive und die Adoptionsparameterstruktur sowie ein Anforderungsprofil darzulegen [De06].

Die Projektziele wurden bereits in der Einleitung des Aufsatzes aufgezeigt (vgl. Kap. 1). Zielstellung ist die Unterstützung der Einführung eines QMS im ambulanten Sektor durch Referenzmodelle. Die Frage nach dem Marktmodell erschließt sich über die Nutzung des Referenzmodells. Im aktuellen Zustand handelt es sich noch um ein wissenschaftliches Referenzmodell [FL04].

Durch die Befragung konnten der Typ der Praxis (Fach- oder Hausarztpraxis), Art der Datenverarbeitung (durchgängig elektronisch oder nicht), der Fachbereich und Umfang des Bestellsystems als Adoptionsparameter identifiziert werden.

Die Anforderungen an das Referenzmodell leiten sich aus dem Forschungsziel, die Einführung eines zertifizierungsfähigen QMS im ambulanten Sektor zu unterstützen, und den intrinsischen Referenzmodelleigenschaften ab:

- Wiederverwendbarkeit für ähnliche Problemstellungen
- bedarfsgerechte Adoptionsunterstützung
- Beschleunigung der Erarbeitung eines Anwendungsmodells
- Empfehlungscharakter der Modelle

4.2 Referenzmodellierungstechnik definieren

Der folgende Schritt nutzt die Ergebnisse der Projektzieldefinition, welche in Form eines Anforderungsprofils vorliegen. Das Ziel, den Adoptionsprozess bedarfsgerecht zu unterstützen macht die Spezifikation von Konstruktionstechniken notwendig. Sie ermöglichen zum einen die Konstruktion, zum anderen die spätere Adaption des Referenzmodells.

Im Gegensatz zu der Arbeit von DELFMANN wurde die UML als Modellierungstechnik für das zu konstruierende Referenzmodell verwendet. Diese Entscheidung wurde aufgrund der hohen Verbreitung und Akzeptanz der UML in der Forschungsgemeinschaft gefällt. Zum aktuellen Zeitpunkt ist dies eine durch die Autoren gefällte Entscheidung. Eine Evaluierung von Referenzmodellierungssprachen für diese Domäne ist für die weitere Forschung angedacht.

Für die Konstruktion war es notwendig, die Spezifikation der UML um die Sprachkonstrukte einer Referenzmodellierungsgrammatik zu erweitern. Das Konzept der Region wird in der UML durch ein gestricheltes Rechteck repräsentiert. Durch die Region können Modellelemente innerhalb eines Diagramms gruppiert werden.

Zusätzlich zur Region wurde eine Spezialisierung der Region – im weiteren Verlauf RM-Region genannt – als Sprachkonstrukt eingeführt. Als graues Rechteck dient dieses Konzept dazu Gruppierung von Elementen des Referenzmodells vorzunehmen, die durch eine Konstruktionstechnik unterstützt werden. Eine spezielle Form der Annotation – im weiteren Verlauf RM-Annotation genannt – der RM-Region gibt dabei die Art der Konstruktionsunterstützung an. Die Kennzeichnung der RM-Region erfolgt durch die Spezialisierung der Region mit dem Stereotyp "<<RM-Region>> Konstruktionstechnik:". Auf die RM-Region kann verzichtet werden, wenn die Konstruktionstechnik nur für ein Element (Instanz) definiert wird. Dann kann die RM-Annotation direkt dem Element hinzugefügt werden. Die Kennzeichnung der RM-Annotation erfolgt durch die Spezialisierung der Annotation mit dem Stereotyp "<<RM-Annotation>> Konstruktions-technik". Die gleichzeitige Verwendung der RM-Region und der RM-Annotation ist möglich, sobald mehrere Konstruktionstechniken sowohl für Gruppen von Elementen als auch für Einzelemente verwendet werden.

Die Angabe der Konstruktionstechnik erfolgt entsprechend den fünf Konstruktionstechniken: Konfiguration, Instanziierung, Spezialisierung, Aggregation und die Analogiekonstruktion [Br03]. Die Formulierung der Regeln erfolgt nach einem Doppelpunkt hinter der Konstruktionstechnik in natürlicher Sprache. Zum Beispiel würde die Annotation einer Konfiguration einer Elementselektion wie folgt lauten: "<<RM-Annotation>> Konfiguration: Element nur zu übernehmen bei durchgängiger elektronischer Datenverarbeitung".

Die adaptative Referenzmodellierungstechnikdefinition ist das Ergebnis dieser Phase der Referenzmodellkonstruktion. Sie besteht aus den Modellelementen des Aktivitätsdiagramms der UML, ergänzt um die Konzepte der RM-Region und RM-Annotation.

4.3 Referenzmodell Implementierung

Gegenstand der Referenzmodellimplementierung ist die Auswahl eines geeigneten Modellierungswerkzeuges und – wenn nötig – dessen Konfiguration. DELFMANN und KNACKSTEDT haben eine Evaluierung von Werkzeugunterstützung hinsichtlich der Unterstützung von Referenzmodellierungsgrammatiken durchgeführt. Von den sieben untersuchten Tools konnte keines eine volle Unterstützung der generierenden Konstruktionsmechanismen unterstützen [DK07]. Doch es wurde festgestellt, dass in Tools, die über eigene Skriptsprachen bzw. Makrodeklaration Erweiterungen zulassen, die Konstruktionstechniken umgesetzt werden können. Diese Untersuchung bestätigte die Entscheidung, das meta-CASE tool (Computer-aided software engineering) Cubetto Toolset® für die Implementierung zu nutzen [Se10]. Als Meta-CASE-Tool werden Modellierungswerkzeuge bezeichnet, in die sich beliebige Modellierungssprachen zur Nutzung implementieren lassen [St96]. Im Cubetto Toolset® können auf Basis der Meta-Modellierungssprache E3 spezifische Modelle in das Werkzeug implementiert und genutzt werden. Dies sowie eine breite Skriptunterstützung und der bereitgestellte Grafikeditor ermöglichen, Elemente der Referenzmodellierungsgrammatik in die Sprache einzuführen. Im Rahmen des vorliegenden Beitrags wurde entsprechend das Meta-Modell des Aktivitätsdiagramms erweitert. Dafür wurde das UML-Metamodell im Werkzeug um den Objekttyp "RM-Region", der eine Spezialisierung des Objekttyps "Region" darstellt, ergänzt und eine neue Präsentation (Rechteck mit grauem Hintergrund und Textfeld) hinzugefügt.

4.4 Referenzmodellerstellung

Nach der Definition der adaptiven Referenzmodellierungstechnikdefinition und des adaptiven Referenzmodellierungstools als Ergebnis der Abschnitte 4.2 und 4.3 wird in diesem Abschnitt das adaptive Referenzmodell entwickelt. Die eigentliche Referenzmodellerstellung beginnt mit der Konstruktion des Referenzmodellrahmens. Der Referenzmodellrahmen wurde auf Grundlage des aus der Prozessanalyse resultierenden Ordnungsrahmens (vgl. Abs. 3.2) entwickelt. Er dient der Strukturierung und Systematisierung der Vielzahl von Einzelprozessen und bietet somit einen übersichtlichen Einstiegspunkt in das Modell [De06].

Die Konstruktion der Detailmodelle erfolgte auf Basis der in der Prozessanalyse erhobenen Daten. Dabei wurde mit den Basiselementen gestartet. Dabei stellen die Basiselemente Modellelemente dar, die in den Anwendungsmodellen jeweils ähnlich waren. Darauffolgend wurden die Detailmodelle der Prozessanalyse um die entsprechenden Konstruktionstechnik und bzw. oder um die Konfigurationsparameter ergänzt. Dazu wurden die im Abschnitt 4.2 definierten Konzepte genutzt.

Ein hybrides adaptives Referenzmodell ist das Ergebnis der Konstruktion. Das heißt, in dem Referenzmodell werden generierende und nicht generierende Konstruktionstechniken kombiniert. Die Entscheidung für die jeweilige Konstruktionstechnik erfolgt anhand der Verbindlichkeit, die durch das Modell für den Adoptionsprozess kommuniziert werden soll, und die Eigenschaften der Konstruktionstechnik in Bezug zu den Anforderungen an das Referenzmodell (vgl. Abs. 4.1).

Somit kann weder von einem monolithischen Modell noch von der Konstruktion einer Komponentenbibliothek gesprochen werden [De06]. Das resultierende Referenzmodell umfasst somit die Definition der Adoptionsregeln sowie die vorgedachten Modellbestandteile.

Abbildung 4 zeigt beispielhaft das Ergebnis der Referenzmodellerstellung am Ausschnitt der EKG-Untersuchung. Sie gliedert sich in den Kernprozessbereich "Spezielle Diagnostik" des Referenzmodellrahmens ein. Der dargestellte Ausschnitt zeigt, wie anhand der Konfigurationsparameter: elektronische oder nicht elektronische Datenverarbeitung (vgl. Abs. 4.2), eine Adoptionsunterstützung (Konfiguration) realisiert wird. Die Möglichkeit der Spezialisierung wird anhand der Verwendung einer RM-Annotation an der Aktivität "EKG-Datensatz ins Praxis Informationssystem (PIS) übertragen" aufgezeigt. Das finale Referenzmodell umfasst 15 Detailmodelle die sich in den Ordnungsrahmen (vgl. Abb. 3) einordnen.

Abbildung 5 zeigt eine durchgeführte Adaption im Werkzeug Cubetto Toolset®. Das Anwendungsmodell wurde für eine Hausarztpraxis mit durchgängigen EDV-gestützten Prozessen adaptiert. Zum Einsatz kamen dabei die Adoptionsmechanismen Spezialisierung und Konfiguration (vgl. Abs. 4.2).

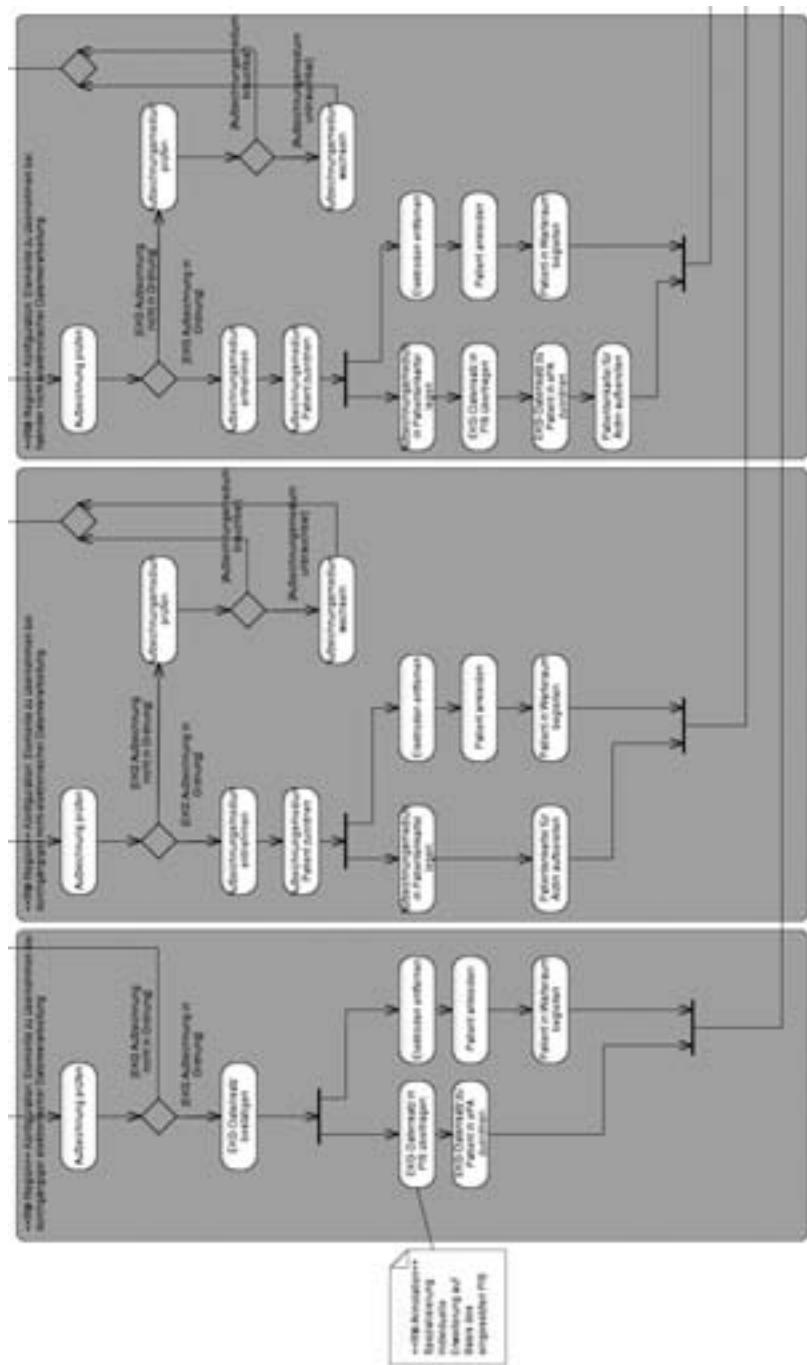


Abbildung 4 - Ausschnitt des adaptiven Referenzmodells – EKG-Prozess

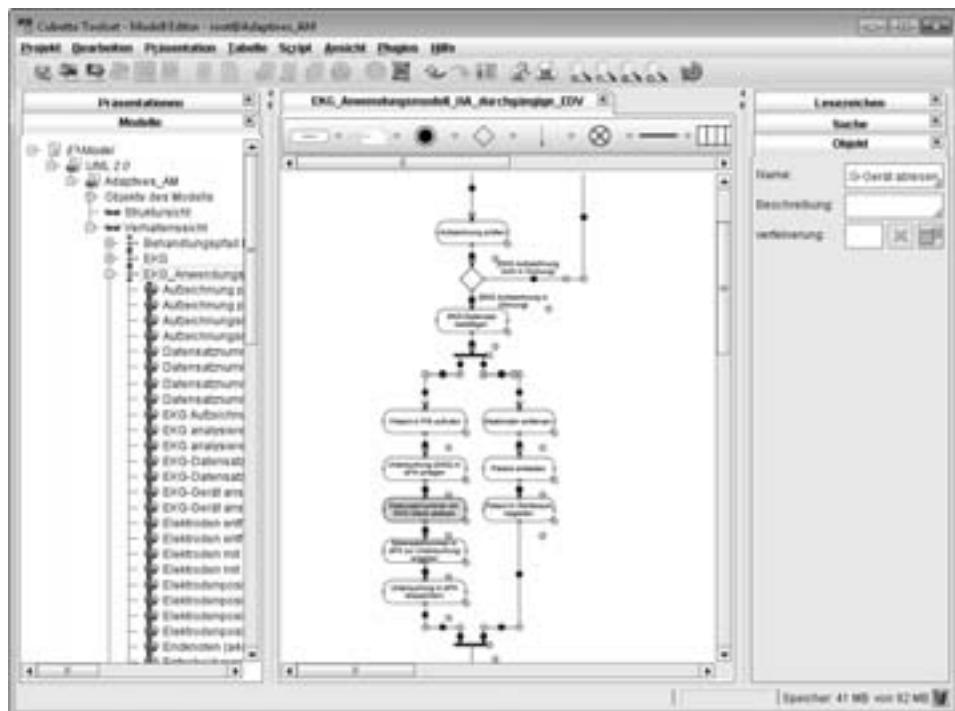


Abbildung 5 – Ausschnitt des adaptierten Anwendungsmodells (EDV durchgängig, Hausarzt) im Cubetto Toolset®

4.5 Referenzmodellbewertung

Teile des erarbeiteten Referenzmodells wurden abschließend den befragten Ärzten vorgestellt. Diese stimmten zu, dass ihnen auf Basis dieser Modelle die Einführung ihres QMS leichter gefallen wäre. Dabei hat einer, der befragten Fachärzte geäußert, dass im Rahmen einer geplanten Zertifizierung das entwickelte Referenzmodell bei der Umsetzung des QMS verwendet werden soll.

5 Zusammenfassung und Ausblick

In dem Artikel wurden die Ergebnisse der aktuellen Forschungsarbeit zur Schaffung eines Referenzmodells zur Unterstützung der Einführung eines QMS für den ambulanten Sektor vorgestellt. Dazu wurden die Grundlagen der Referenzmodellierung erörtert und die Ergebnisse der Prozessanalyse vorgestellt (vgl. Kap. 2 u. 3). Darauf aufbauend wurde der Prozess der Referenzmodellkonstruktion dargelegt und unter Einbeziehung der Methode von DELFMANN ein Referenzmodell erstellt (vgl. Kap. 4).

Es konnte gezeigt werden, dass eine referenzmodellgestützte Einführung eines QMS im ambulanten Sektor möglich ist. Darüber hinaus konnte verdeutlicht werden, wie auf Basis von empirisch erarbeiteten Detailmodellen, Referenzmodelle für eine bestimmte Domäne erstellt werden können. Der Prozess der Referenzmodellkonstruktion zeigt auch die hohe Komplexität der Referenzmodellbildung. Dies unterstreicht die Argumentation von VOM BROCKE, der den Erstellungsaufwand eines Referenzmodells für sehr hoch einschätzt und einen Trade-Off erst über die Einsparung, die mit dem Referenzmodell realisiert werden kann, sieht. Dies setzt unweigerlich eine hohe Zahl der Anwendungen voraus [Br07].

In der durchgeführten Befragung wurde zudem das Problem der verschiedenen Fachbereiche und geringen Standardisierung in den Prozessen erkannt, sodass das Referenzmodell auf einem relativ abstrakten Niveau bleibt. Auch die Heterogenität der Fachbereiche lässt das Referenzmodell in Richtung eines Ordnungsrahmens mit teilweise vorgedachten Lösungen tendieren. Um dieser Tendenz zu entgegnen, wird in der zukünftigen Forschung Abstand genommen, ein solch breit angelegtes Referenzmodell zu schaffen. Vielmehr sollten fachbereichsspezifische Referenzmodelle gefertigt werden, auch wenn damit ein Teil der Generalität des Referenzmodells verworfen wird. Daraus entsteht weiterhin der Vorteil, dass medizinische Prozesse leichter in das Referenzmodell zu integrieren sind. Beispielsweise können dies Inhalte, die durch die medizinischen Fachgesellschaften bereitgestellten Medizinischen Leitlinien, sein.

Des Weiteren ist die Anzahl der Befragung zu erhöhen, um allgemeingültige Aussagen besser untermauern zu können. Für die nötige Validität der Ergebnisse müssen in der weiteren Forschung die Ergebnisse der Arbeit noch eine empirische Evaluation durchlaufen. Bevor im Rahmen einer Fallstudie die Ergebnisse verifiziert werden, wird zunächst die Evaluierung des Referenzmodells und dessen Prozesse durch die Befragung möglicher Anwender erfolgen.

Weiterhin muss die Eignung der Modellierungssprache im Kontext dieser Arbeit evaluiert werden. Hier sollten insbesondere die Themen Darstellung und Toolunterstützung der Konstruktionstechniken, sowie die Sprachadäquanz von Modellen für die medizinische Domäne untersucht werden. Auch die Explikation der Adoptionsmethode wird in einem weiteren Arbeitsschritt durchgeführt werden.

Des Weiteren sollte ein Konzept zur Verbreitung und Bereitstellung der Referenzmodelle erarbeitet werden. In diesem Zusammenhang sei die Arbeit der Open Model Initiative (OMI) genannt [FSK07]. Sie wurde gegründet, um zu untersuchen welche Hemmnisse es bei der Anwendung von Referenzmodellen gibt. Dazu zählen z. B. der Einfluss der Modellierungssprache und Probleme der Modellmigration. Weiterhin sind die Aufgaben und Anforderungen von Metainformationen für das Auffinden eines geeigneten Referenzmodells und dessen Bewertung zu klären. Von der Beantwortung dieser und ähnlicher Fragen hängt die Entwicklung geeigneter Werkzeuge für die Erstellung, den Austausch, die Verbreitung und die Anwendung von Referenzmodellen ab.

Literaturverzeichnis

- [BB04] vom Brocke, J.; Buddendick, C.: Konstruktionstechniken für die Referenzmodellierung - Systematisierung, Sprachgestaltung und Werkzeug-unterstützung. In: Becker, Jörg (Hrsg.): Referenzmodellierung. Grundlagen, Techniken und domänenbezogene Anwendung ; mit 6 Tabellen. Heidelberg: Physica-Verl., 2004; S. 19–49.
- [BD09] Bortz, J.; Döring, N.: Forschungsmethoden und Evaluation. Für Human- und Sozialwissenschaftler. 4., überarb. Aufl., Nachdr. Heidelberg: Springer-Medizin-Verl. (Springer-Lehrbuch Bachelor, Master), 2009.
- [BDK02] Becker, J.; Delfmann, P.; Knackstedt, R.; Kuropka, D.: Konfigurative Referenzmodellierung. In: Becker, Jörg (Hrsg.): Wissensmanagement mit Referenzmodellen. Konzepte für die Anwendungssystem- und Organisationsgestaltung. Heidelberg: Physica-Verl., 2002; S. 25–144.
- [BDK04] Becker, J.; Delfmann, P.; Knackstedt, R.: Konstruktion von Referenzmodellierungssprachen. Ein Ordnungsrahmen zur Spezifikation von Adoptionsmechanismen für Informationsmodelle. In: Wirtschaftsinformatik, Jg. 46, H. 4, 2004; S. 251–264.
- [BG05] Bundesministerium für Gesundheit: Richtlinie des Gemeinsamen Bundesausschusses über grundsätzliche Anforderungen an ein einrichtungsinternes Qualitätsmanagement für die an der vertragsärztlichen Versorgung teilnehmenden Ärzte, Psychotherapeuten und medizinischen Versorgungszentren (Qualitätsmanagement-Richtlinie vertragsärztliche Versorgung) in der Fassung vom 18. Oktober 2005 - Beschluss veröffentlicht: BAuz. Nr. 248 (S. 17329) vom 31.12.2005, 2005.
- [BK08] Becker, J.; Kahn, D.: Der Prozess im Fokus. In: Becker, Jörg; Kugeler, Martin; Rosemann, Michael (Hg.): Prozessmanagement. Ein Leitfaden zur prozessorientierten Organisationsgestaltung. 6., überarb. und erw. Aufl. Berlin: Springer, 2008; S. 3–15.
- [BKK01] Becker, J.; Knackstedt, R.; Kuropka, D.; Delfmann, P.: Subjektivitätsmanagement für die Referenzmodellierung: Vorgehensmodell und Werkzeugkonzept. In: Proceedings of the KnowTech 2001. Dresden, 2001.
- [Br03] vom Brocke, Jan: Referenzmodellierung. Gestaltung und Verteilung von Konstruktionsprozessen. Berlin: Logos (Advances in information systems and management science, 4), 2003.
- [Br07] vom Brocke, J.: Design Principles for Reference Modeling: Reusing Information Models by Means of Aggregation, Specialisation, Instantiation, and Analogy. In: Fettke, P.; Loos, P. (eds.): Reference Modeling for Business Systems Analysis. Idea Group Publishing, Hershey PA, 2007; S. 47–75.
- [Br09] Braun, R.: Referenzmodellierung. Grundlegung und Evaluation der Technik des Modell-Konfigurationsmanagements. Berlin: Logos. (Advances in information systems and management science, 44), 2009.
- [De06] Delfmann, P.: Adaptive Referenzmodellierung. Methodische Konzepte zur Konstruktion und Anwendung wiederverwendungsorientierter Informationsmodelle. Berlin: Logos-Verl. (Advances in information systems and management science, 25), 2006.
- [DI05] Qualitätsmanagementsysteme. Grundlagen und Begriffe; (ISO 9000:2005); Dreisprachige Fassung, Berlin: Beuth (DIN EN ISO), 2005.
- [DI08] Qualitätsmanagementsysteme. Anforderungen (ISO 9001:2008); Dreisprachige Fassung. Berlin: Beuth (DIN EN ISO), 2009.
- [DK07] Delfmann, P.; Knackstedt, R.: Konfiguration von Informationsmodellen: Untersuchungen zu Bedarf und Werkzeugunterstützung. In: Oberweis, A.; Weinhardt, C.; Gimpel, H.; Koschmider, A.; Pankratius, V.; Schnizler, B. (Hg.): eOrganisation. Service-, Prozess-, Market-Engineering. Bd. 2. 8. Internationale Tagung Wirtschaftsinformatik (WI 2007). Karlsruhe: Univ.-Verl., 2007; S. 127–144.
- [ELS10] Esswein, W.; Lehrmann, S.; Schlieter, H.: Referenzmodelle. In: WISU - Das Wirtschaftsstudium, Jg. 39, H. 3, 2010; S. 371–375.

- [FB08] Fettke, P.; vom Brocke, J.: Referenzmodell. Online verfügbar unter <http://www.oldenbourg.de:8080/wi-encyklopaedie/lexikon/is-management/Systementwicklung/Softwarearchitektur/Wiederverwendung-von-Softwarebausteinen/Referenzmodell>, zuletzt aktualisiert am 19.09.2008, zuletzt geprüft am 15.05.2010, 2008.
- [Fe06] Fettke, P.: Referenzmodellevaluation. Konzeption der strukturalistischen Referenzmodellierung und Entfaltung ontologischer Gütekriterien. Berlin: Logos-Verl. (Wirtschaftsinformatik - Theorie und Anwendung, 5), 2006.
- [FL04a] Fettke, P.; Loos, P.: Referenzmodellierungsorschung - Langfassung eines Aufsatzes. In: Working Papers of the Research Group Information Systems & Management, H. 16, 2004; S. 7–52.
- [FL04b] Fettke, P.; Loos, P.: Referenzmodellierungsorschung. In: Wirtschaftsinformatik, Jg. 46, H. 5, 2004; S. 311–340.
- [FSK07] Frank, U.; Strecker, S.; Koch, S.: Open Model - ein Vorschlag für ein Forschungsprogramm der Wirtschaftsinformatik - Langfassung. ICB-Research Report, Institut für Informatik und Wirtschaftsinformatik (ICB), Universität Duisburg-Essen, Campus Essen, Nr. 8, Februar 2007. Online verfügbar unter <http://www.openmodels.org/system/files/ICBReport08.pdf>, 2007; zuletzt geprüft am 15.05.2010.
- [GD04] Diel, F.; Gibis, B.: Qualitätsmanagement. Mehrwert für die Arztpraxis. In: Deutsches Ärzteblatt, Jg. 101, H. 10, 2004; S. A620-A622.
- [GM03] Deutscher Bundestag: Gesetz zur Modernisierung der gesetzlichen Krankenversicherung (GKV-Modernisierungsgesetz - GMG). In: Bundesgesetzblatt, Nr. 55, 14.11.2003; S. 2190–2258.
- [Ha94] Hars, A.: Referenzdatenmodelle. Grundlagen effizienter Datenmodellierung. Wiesbaden: Gabler (Schriften zur EDV-orientierten Betriebswirtschaft), 1994.
- [Ha99] Hammel, C.: Generische Spezifikation betrieblicher Anwendungssysteme. Ein Ansatz zur generischen Spezifikation betrieblicher Anwendungssysteme, Bamberg, Aachen: Shaker (Berichte aus der Wirtschaftsinformatik), 1999.
- [KB08] Kassenärztliche Bundesvereinigung (KBV): QM-Systeme im Vergleich. Spezifische QM-Systeme bzw. -Verfahren für Praxen. Online verfügbar unter <http://www.kbv.de/qep/11622.html>, zuletzt aktualisiert am 17.07.2008, zuletzt geprüft am 08.05.2010.
- [KI05] Knon, D.; Ibel, H.: Qualitätsmanagement in der Arztpraxis. München: Hanser (Pocket-Power, 44), 2005.
- [Ku05] Kunhardt, H.; Dannert, E.; Porzsolt, F.; Sigle, J.: Medizinisches Qualitätsmanagement. In: Lehmann, T. M. (Hrsg.): Handbuch der medizinischen Informatik. 2., vollst. neu bearb. Aufl. München: Hanser, 2005; S. 774–813.
- [Sc00] Schlagheck, B.: Objektorientierte Referenzmodelle für das Prozeß- und Projektcontrolling. Grundlagen, Konstruktion, Anwendungsmöglichkeiten. Wiesbaden: Dt. Univ.-Verl. [u.a.] (Gabler Edition Wissenschaft Informationsmanagement und Controlling), 2000.
- [Sc98] Schütte, R.: Grundsätze ordnungsmäßiger Referenzmodellierung. Konstruktion konfigurations- und anpassungsorientierter Modelle. Wiesbaden: Gabler (Neue betriebswirtschaftliche Forschung, 233), 1998.
- [Sc99] Schwegmann, A.: Objektorientierte Referenzmodellierung. Theoretische Grundlagen und praktische Anwendung. Wiesbaden: Dt. Univ.-Verl. [u.a.] (Gabler Edition Wissenschaft Informationsmanagement und Controlling), 1999.
- [Se10] semture.de: Cubetto Toolset. Online verfügbar unter <http://www.semture.de/Cubetto-Toolset/Cubetto-Toolset/>, 2010; zuletzt geprüft am 15.05.2010.

- [SG08] Stiftung Gesundheit: Qualitätsmanagement in der ärztlichen Praxis 2008. Eine deutschlandweite Befragung niedergelassener Ärztinnen und Ärzte. Online verfügbar unter http://www.stiftung-gesundheit.de/PDF/studien/Studie_QM_2008.pdf, 2008; zuletzt geprüft am 05.05.2010.
- [SG09] Stiftung Gesundheit: Qualitätsmanagement in der ärztlichen Praxis 2009. Eine deutschlandweite Befragung niedergelassener Ärztinnen und Ärzte. Online verfügbar unter http://www.stiftung-gesundheit.de/PDF/studien/Studie_QM_2009.pdf, 2009; zuletzt geprüft am 13.05.2010.
- [St96] Strahringer, S.: Metamodellierung als Instrument des Methodenvergleichs. Eine Evaluierung am Beispiel objektorientierter Analysemethoden. Aachen: Shaker (Berichte aus der Betriebswirtschaft), 1996.
- [Th06] Thomas, O.: Management von Referenzmodellen. Entwurf und Realisierung eines Informationssystems zur Entwicklung und Anwendung von Referenzmodellen. Berlin: Logos-Verl. (Wirtschaftsinformatik - Theorie und Anwendung, 1), 2006.
- [Wa09] Wasem, J.: Pay for Performance (P4P) in der ambulanten Gesundheitsversorgung - Qualität bestimmt die Vergütung. Workshop "focus – Zukunftsperspektiven im Gesundheitswesen" 09. Oktober 2009 Online verfügbar unter http://www.kvno.de/importiert/termine/focus/p4p09102009/focus_P4P_091009_Wasem.pdf, 2009; zuletzt geprüft am 20.06.2010

Anwendung der enhanced Telecom Operations Map auf Dienstleistungsunternehmen in der Stromwirtschaft am Beispiel eines Prozesses im Bereich Messwesen

M. Deindl, E. Naß, T. Beckhoff

Bereich Informationsmanagement
FIR an der RWTH Aachen
Pontdriesch 14-16
52062 Aachen
matthias.deindl@fir.rwth-aachen.de
eric.nass@fir.rwth-aachen.de
tim.beckhoff@fir.rwth-aachen.de

Abstract: Seit der Liberalisierung der Energiewirtschaft Ende der 90er Jahre des letzten Jahrhunderts spielt die Industrialisierung dieses Sektors eine immer wichtigere Rolle. Um im Wettbewerb bestehen zu können, müssen Unternehmen der Stromwirtschaft sich stärker prozessorientiert aufstellen und ihre Organisation dementsprechend anpassen. Ein entsprechendes Referenzmodell fehlt zurzeit, könnte jedoch die Modellierung und das Management von Geschäftsprozessen unterstützen. Da zwischen Strom- und Telekommunikationswirtschaft viele Parallelen bestehen, wird im folgenden Beitrag die Übertragbarkeit der in der Telekommunikationswirtschaft etablierten enhanced Telecom Operations Map (eTOM) auf die Stromwirtschaft überprüft. Hierzu wird die Methode der Analogiebetrachtung verwendet. Vorläufiges Ergebnis der vorgestellten Untersuchung ist die grundsätzliche Eignung des eTOM-Frameworks für eine Übertragung auf die Stromwirtschaft.

1 Motivation und Zielsetzung

Während besonders kleine und mittlere Energieversorgungsunternehmen (EVU) noch dabei sind, ihre Geschäftsprozesse an Unbundling¹ und Öffnung des Marktes anzupassen, steht mit der Liberalisierung des Messwesens schon die nächste große Herausforderung bevor. Dabei gewinnt eine sowohl kostengünstige als auch effiziente Informationssystemarchitektur nicht zuletzt durch den Einsatz von sogenannten Smart Meters (elektronische Stromzähler) enorm an Bedeutung [Co10]. Um mit der Geschwindigkeit des Markts mithalten zu können sind Referenzprozessmodelle hilfreich, die Energieversorgungsunternehmen dabei unterstützten neue Prozesse wertschöpfend auszurichten und flexibel an sich ändernde Rahmenbedingungen anzupassen [Da04].

Durch das 2008 in Kraft getretene „Gesetz zur Öffnung des Messwesens bei Strom und Gas für Wettbewerb“ hat jeder Anschlussnutzer das Recht seinen Messdienstleister und seinen Messstellenbetreiber frei zu wählen. Die hierdurch neu entstehenden Marktrollen waren bis dato in den Strukturen des EVU integriert und stellten eher einen unterstützenden Prozess beim Verkauf von Strom dar, als ein separates Glied der Wertschöpfungskette. Der Charakter eines wertschöpfenden Prozesses wird durch die vom Gesetzgeber auferlegte Pflicht zum Einbau von Smart Meters bei Neuinstallationen verstärkt. Hier bieten unter anderen Fernauslesung oder eine kundengerechte Aufbereitung der Daten Möglichkeiten, neue Stufen in die Wertschöpfung zu implementieren [Ha08].

Bei der Klassifizierung des Wertschöpfungstyps wird ersichtlich, dass sich die Prozesse im Messwesen erheblich von denen im Netzsegment unterscheiden. Während Netzbetreiber aus Prozesssicht produzierenden Unternehmen ähneln [St10], gibt es beim Messdienstleister kaum Beschaffungsaktivitäten und die physische Leistungserbringung beschränkt sich auf die Einrichtung und Instandhaltung der Messpunkte. Dafür müssen große Datenmengen verarbeitet und die Qualität der Messdaten überwacht werden. Zusätzlich ist auch die angebotene Leistung weniger homogen als beim Produkt Strom, da neben dem Basiservice der klassischen Messung in den nächsten Jahren mit der Einführung intelligenter Stromnetze weitere Value Added Services im Bereich des Messwesens entstehen werden, z.B. Energieberatung oder Hausgerätesteuerung. Bei allen jetzigen und zukünftigen Prozessen stellt der Datenaustausch zwischen verschiedenen Unternehmen gleichzeitig zentrales Element als auch größte Herausforderung in der Prozesskette dar.

Um beim Kunden eine qualitativ hochwertige Leistung zu erbringen, müssen Prozesse über Unternehmensgrenzen hinweg eindeutig beschrieben und mit dem gleichen Verständnis für Qualität bearbeitet werden. Dies kann durch die Einführung von branchenweiten Standards und die Nutzung eines einheitlichen Prozessverständnisses erreicht werden [Us09].

¹ Unter Unbundling versteht man die Entflechtung von Erzeugung, Übertragung und Verteilung des Stroms auf rechtlicher, informationeller, operationeller und buchhalterischer Ebene (vgl. [Bo07a]).

Um die oben genannten Herausforderungen zu bewältigen, können Referenzmodelle verwendet werden, die die Gestaltung und Umsetzung von Geschäftsprozessen unterstützen [Be98]. Ein Referenzmodell zur Organisationsgestaltung mit dem Schwerpunkt auf kundennahe Geschäftsprozesse der Stromwirtschaft existiert momentan nicht, ist aber Gegenstand aktueller Arbeiten [De10].

Die Energiewirtschaft weist in vielerlei Hinsicht Ähnlichkeiten zur Telekommunikationswirtschaft auf [Wi09], da zum einem beide Branchen zu den Netzindustrien zählen, zum anderem die Geschäftsmodelle von Telekommunikations-Service Providern und Dienstleistungsunternehmen in der Stromwirtschaft vergleichbar sind. Um Leistungen zum Kunden bringen zu können, müssen diese in beiden Fällen an ein physikalisches Netz angeschlossen sein. Zudem spielen langfristige vertragliche Bindungen und differenzierte Tarifgestaltungen in beiden Branchen eine zentrale Rolle. Auch die Fakturierung und Abrechnung ist in ihren Grundzügen ähnlich ausgeprägt. Aus diesem Grund wird die Hypothese aufgestellt, dass sich die „enhanced Telecom Operations Map“ (eTOM), ein Rahmenwerk für Geschäftsprozesse von Unternehmen im Bereich der Telekommunikation und IT-Dienstleistung [Tm09b], für einen Analogieschluss eignet [Ke04]. Zu prüfen ist, ob die stärkere Trennung marktlicher Rollen in der Energiewirtschaft, z.B. zwischen Netz und Vertrieb, gegen die Verwendung eines einheitlichen Referenzmodells spricht.

Zielsetzung dieses Artikels ist somit die Evaluation der Übertragbarkeit der eTOM auf den vorher skizzierten Anwendungsfall.

Im folgenden Abschnitt wird auf die Entwicklung eines auf die Stromwirtschaft bezogenen Referenzmodells eingegangen. Neben der Zielsetzung wird auch das methodische Vorgehen erläutert und verwandte Literatur diskutiert, um die Untersuchungen im Rahmen dieses Artikels einzurunden. In Abschnitt 3 wird am Beispiel eines Messdienstleisterprozesses die Übertragung der enhanced Telecom Operations Map auf die Stromwirtschaft überprüft und bewertet. In Abschnitt 4 werden die Untersuchungen schließlich zusammengefasst und ein Ausblick auf weiteren Forschungsbedarf gegeben.

Referenzprozessmodellierung in der Energiewirtschaft

2.1 Zielsetzung und Merkmale der Referenzmodellierung

Zielsetzung der in [De10] beschriebenen Arbeiten ist die Entwicklung eines Organisations-Referenzmodells für Prozesse der stromwirtschaftlichen Funktionen Lieferung und Messung. Zur Einordnung und Abgrenzung der Arbeiten werden modellbezogene Unterscheidungsmerkmale verwendet. Abbildung 1 veranschaulicht die für das Referenzmodell zutreffenden Ausprägungen der Unterscheidungsmerkmale durch farbliche Hinterlegung.

- Adressatenkreis: Man kann beim Adressatenkreis und dem damit verbundenen Zweck zwischen Organisations- und Anwendungssystemgestaltung [RoSc99], [Be08] differenzieren. Das hier betrachtete Referenzprozessmodell dient der Organisationsgestaltung mit dem Ziel, insbesondere die Geschäftsprozessmodellierung bzw. das -management zu unterstützen.
- Fachbezug: Die Beschreibungsebenen können nach Nähe zur informationstechnischen Umsetzung unterschieden werden. Scheer teilt die Ebenen in das Fachkonzept, DV-Konzept (Datenverarbeitungskonzept) und die Implementierung ein [Sc01]. Relevant für die Untersuchung hier ist das Fachkonzept, da es den engsten Bezug zu betriebswirtschaftlichen Fragestellungen aufweist.
- Aspekt und Sicht: Im Sinne des Komplexitätsmanagements hat es sich bewährt, die Modellierung in verschiedenen Beschreibungssichten vorzunehmen [Sc01]. Mit der Berücksichtigung von Funktions-, Daten- und Prozesssichten deckt es dabei sowohl Struktur- als auch Verhaltenssichten ab [RoSc99], [Be08]. Im Rahmen des Anwendungszwecks des Geschäftsprozessmanagements ist dabei vor allem die Prozesssicht relevant [ScNü00].
- Evaluation: Laut Fettke und Loos ist die Evaluierung von Modellen ein wichtiger Aspekt, der vielfach vernachlässigt wird, da er nicht zwingend Bestandteil des Konstruktionsprozesses war [FeLo04a]. Referenzprozessmodelle sollen sowohl formal nach den Grundsätzen ordnungsmäßiger Modellierung evaluiert werden [Be95], als auch durch Fallstudien bzw. durch exemplarische Anwendung praktisch überprüft werden.
- Anwendungsdomäne: Ein Modell ist nur dann nützlich, wenn es sich auf die jeweilige Modellierungssituation anwenden lässt. Fettke und Loos, die systematisch Referenzmodelle erhoben haben und eine Übersicht über den aktuellen Stand der Referenzmodellierung geben, erwähnen zwar zahlreiche Anwendungsdomänen, nicht aber die Stromwirtschaft [FeLo04a], [FeLo04b]. Wie oben bereits erwähnt, sind gerade die Funktionen der Messung und Lieferung für die zukünftige Entwicklung in der Stromwirtschaft relevant und repräsentativ in Bezug auf die aktuellen Herausforderungen (Smart Metering, Smart Grids, neue Dienstleistungen, Kundenorientierung, etc.). Aus diesem Grund beschränkt sich die Untersuchung auf diese beiden Funktionen. Schwerpunktmaßig soll dabei der wirtschaftliche und weniger der technische Aufgabenbereich [Be09] betrachtet werden. Der Fokus liegt auf jenen Prozessen, bei denen eine direkte Kundeninteraktion stattfindet, wie beispielsweise der Lieferantenwechsel- oder Abrechnungsprozess.

	Merkmale		Ausprägung				
Modellierung	Adressatenkreis (vgl. [ROSEMANN u. SCHUETTE 1999])		Anwendungssystemgestaltung			Organisationsgestaltung	
	Fachbezug (vgl. [SCHEER 2001])		Fachkonzept		DV-Konzept		Implementierung
	Aspekt (vgl. [ROSEMANN u. SCHUETTE 1999])		Verhaltensmodell				Strukturmodell
	Sicht (vgl. [SCHEER 2001])		Funktion	Prozesssicht		Datensicht	Organisationssicht
	Evaluation (vgl. [FETTKE u. LOOS 2004a])		Formale Evaluation		Fallstudien		Reale Nutzung
Domäne	Branche (vgl. [FETTKE u. LOOS 2004])		Industrie	Handel	Öffentliche Verwaltung	Stromwirtschaft	...
	Aufgabenbereich (vgl. [BEENKEN et al. 2009])		Wirtschaftlich			Technisch	
	Stromwirtschaftliche Funktionen (vgl. [DEINDL et al. 2010])		Erzeugung	Verteilung	Handel	Lieferung	Messung

Abbildung 1: Merkmale des zu entwickelnden Referenzmodells

2.2 Methodisches Vorgehen

Die Modellierung von Referenzmodellen ist eine konstruktive, gestalterische Aufgabe, die durch eine geeignete Methodik unterstützt werden muss. Die Gestaltung des Referenzmodells lehnt sich dabei an die von Schütte entwickelte Vorgehensweise an (vgl. z.B. [RoSc99], [Be03], [Br03]), wie in Abbildung 2 veranschaulicht.

Zur Problemdefinition im Schritt 1 des Vorgehens gehört nicht nur die Beschreibung der Modellierungsziele und des sich daraus ergebenden Nutzens, sondern auch die Abgrenzung des im Modell abzubildenden Untersuchungsbereichs. Im zweiten Schritt wird auf hohem Abstraktionsniveau eine einheitliche Modellstruktur geschaffen und eine Klassifikation von Modellbestandteilen vorgenommen. Die Gestaltung kann durch einen Ordnungsrahmen unterstützt werden, der der Modellierung eine einheitliche Struktur verleiht und einen anschaulichen Zugang zu weiteren Modellinhalten ermöglicht [RoSc99]. Auf Grundlage des Referenzmodellrahmens wird im dritten Schritt das „wie“ spezifiziert. Bei der Entwicklung von Teilmodellen werden hierbei der Rahmen inhaltlich gefüllt und die Modellsichten detailliert. Ergebnis des Modellierungsschrittes sind Referenzprozess- und Referenzdatenmodelle sowie Verbindungen zwischen den Modelltypen. Es wird eine Validierung nach Modellierung des Referenzmodellrahmens und der -struktur durch Anwender zur Qualitätskontrolle empfohlen. So kann das Referenzmodell anhand der Grundsätze ordnungsgemäßer Modellierung überprüft werden [Be03]. Anschließend wird das Referenzmodell im Schritt 4 des Vorgehens vervollständigt, indem Querverbindungen sowohl zur Inter- als auch zur Intramodellkonsistenz hinzugefügt werden. Weiterhin kann in diesem Schritt das Modell um quantitative Aussagen und Best Practices ergänzt werden. Im fünften Schritt kann das Referenzmodell zur Erstellung unternehmens- und anwendungsspezifischer Modelle verwendet werden.

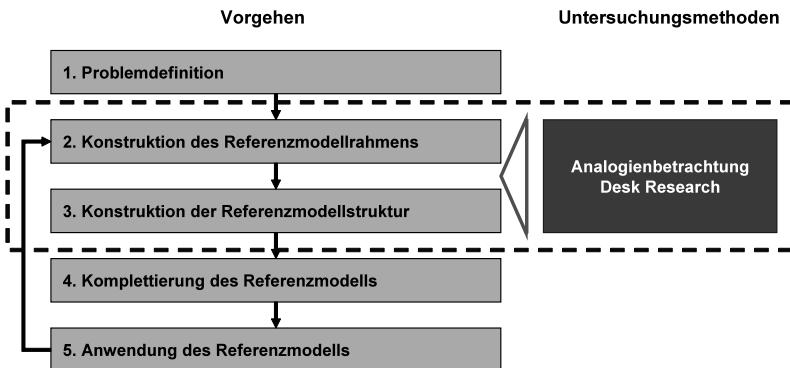


Abbildung 2: Vorgehen bei der Referenzmodellentwicklung nach [RoSc99]

Im Rahmen dieses Beitrags wird untersucht, ob sich das Prozessmodell der enhanced Telecom Operations Map (eTOM) als Grundlage für eine Analogiebetrachtung bei der Entwicklung eines Referenzmodells für Dienstleistungsunternehmen der Stromwirtschaft (wie in Abschnitt 2.1 beschrieben) eignet. Konkret wird überprüft, ob auf Grundlage von eTOM die Phasen der Entwicklung des Referenzmodellrahmens und der Konstruktion der Referenzmodellstruktur unterstützt werden können. Mit Hilfe des Analogieschlusses kann ein solcher erster Modellentwurf entwickelt werden. Auf Grundlage von schriftlichen Befragungen, Experteninterviews und Workshops mit Dienstleistungsunternehmen der Stromwirtschaft können anschließend Anpassungsbedarf des ersten Entwurfs identifiziert werden und Schritte 2-4 durchgeführt werden.

Nach [Ha02] wird unter Analogie die „erkennbare Ähnlichkeit in Form, Eigenschaft oder Funktion zweier Phänomene (Gegenstände oder Abläufe) verstanden“. Die Analogie wird im Rahmen der Referenzmodellierung als geeignete Konstruktionstechnik für Modelle angesehen, bei der vorhandene ähnliche Lösungen auf neue Problemstellungen übertragen werden [Be04]. Die Analogiebetrachtung lässt sich grundsätzlich in vier Schritte gliedern, wie in Abbildung 3 dargestellt.

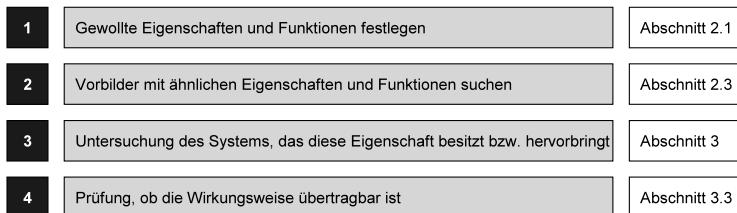


Abbildung 3: Schritte der Analogiebetrachtung nach [Ha2002]

Abschnitt 2.1 beschreibt in groben Zügen die Zielsetzung des zu entwickelnden Referenzmodells und gibt somit Aufschluss über die im ersten Schritt der Analogiebetrachtung zu definierenden Eigenschaften und Funktionen. Vorbilder mit ähnlichen Eigenschaften und Funktionen (Schritt 2) werden in Abschnitt 2.3 diskutiert. Schwerpunkt des Beitrages sind die Schritte 3 und 4 der Analogiebetrachtung. In Abschnitt 3 des Artikels wird entsprechend die eTOM als ein System mit ähnlichen Eigenschaften und Funktionen auf ihre Anwendbarkeit hin untersucht und bewertet.

2.3 Verwandte Arbeiten

Um eine Vorauswahl zu treffen, welche Referenzmodelle hinsichtlich eines möglichen Analogieschlusses untersucht werden sollen, betrachtet die vorliegende Arbeit lediglich Referenzmodelle, für die bereits aufgezeigt wurde, dass Parallelen zwischen ihrer eigenen Anwendungsdomäne und der Stromwirtschaft bestehen. In [De10] wurden einige Referenzmodelle auf ihre Übertragbarkeit auf die Stromwirtschaft hin untersucht. Nach jetzigem Stand erscheinen unter anderem das Supply Chain Operations Reference Model SCOR und die eTOM für eine Analogiebetrachtung in der Stromwirtschaft geeignet. Eine umfassendere Betrachtung möglichst aller in Frage kommender Referenzmodelle soll Aufgabe zukünftiger Arbeiten sein.

Übertragung von SCOR auf die Energiewirtschaft – Das SCOR-Modell ist ein Ansatz zur Beschreibung von Lieferketten für produzierende Unternehmen und Dienstleister und stellt ein normatives Beschreibungsmodell dar [Po05], [Bo07b]. SCOR verknüpft dabei Prozess-Elemente mit Leistungskennzahlen, optimalen Geschäftspraktiken und Besonderheiten bei der Ausführung von Lieferkettenaktivitäten. Mit seinem Domänenbezug eignet sich SCOR besonders für die Übertragung auf technische Funktionen der Energiewirtschaft, wie beispielsweise das Asset Management (Netzausbau, -Instandhaltung etc.) [St10]. Wie oben beschrieben liegt der Schwerpunkt des zu entwerfenden Referenzmodells auf den Funktionen der Lieferung und Messung, die sich von ihrer Beschaffenheit von netztechnischen Funktionen wie in [St10] untersucht, wesentlich unterscheiden. Im Vordergrund steht hier vielmehr der Dienstleistungscharakter und nicht der logistische Aspekt der Lieferkette. Aus diesem Grund scheint SCOR für eine Analogiebetrachtung im Rahmen der hier vorgestellten Referenzmodellentwicklung eher ungeeignet.

eTOM für Versorgungsunternehmen – Die enhanced Telecom Operations Map (eTOM) ist ein Rahmenwerk für Geschäftsprozesse von Unternehmen im Bereich der Telekommunikation und IT-Dienstleistungen, welche vom TeleManagement Forum herausgegeben wird. Im Mittelpunkt dieses generischen Referenzmodells stehen Informations- und Kommunikationsdienste, die in einer sich ständig ändernden Unternehmensumgebung ein stabiles System gewährleisten sollen [Tm09b], [Fü08]. eTOM wurde unabhängig von bestimmten Technologien entwickelt, sodass ihre Flexibilität als hoch einzustufen ist. Weiterhin kann sie gut an verschiedene Ausgangslagen angepasst werden [Ha04]. eTOM repräsentiert einen top-down Ansatz von Geschäftsprozessen, lässt aber größtenteils offen, mit welchen Mitteln diese Prozesse umgesetzt werden können. In der Telekommunikationsindustrie ist sie bereits weit verbreitet und auch für Service Anbieter anderer Branchen wird sie zunehmend attraktiver, da eTOM im Vergleich zu ihrer Vorgängerin TOM nicht mehr nur auf klassische Telekommunikationsdienste ausgerichtet ist [Br07]. Die Anwendbarkeit auf Geschäftsprozesse in der Energiewirtschaft wurde bisher jedoch noch nicht umfassend untersucht. Laut KEBER lässt sich eTOM jedoch aufgrund des generischen Ansatzes auf Unternehmen der Versorgungswirtschaft anwenden [Ke04].

Zusammenfassend lässt sich feststellen, dass die eTOM sich eher eignet, die Prozesse der Messung und Lieferung abzubilden. Im Folgenden wird daher untersucht, wie eTOM für die Modellierung eines Beispielprozesses eines Messdienstleisters verwendet werden kann.

3 Anwendung der eTOM auf die Energiewirtschaft

3.1 Die Enhanced Telecom Operations Map

Die eTOM teilt sich grundsätzlich in drei Bereiche. In der Hauptprozessgruppe „Strategy, Infrastructure and Product“ werden alle Geschäftsprozesse abgebildet, die mit der Planung und Entwicklung von Produkten und Services verknüpft sind. Die klassischen indirekten Unternehmensfunktionen wie beispielsweise Risikomanagement oder Personalwesen werden durch den Bereich „Enterprise Management“ abgedeckt. Für das Tagesgeschäft wesentlich relevanter ist die dritte Hauptprozessgruppe „Operations“, die alle operativen Aktivitäten des Unternehmens umfasst und die im Folgenden näher betrachtet wird.

Je nach gewünschtem Detaillierungsgrad lassen sich Geschäftsprozesse in verschiedenen Ebenen der eTOM verorten. Dabei sind die obersten beiden Ebenen, in der eTOM „Level 0“ und „Level 1“ oder konzeptuelle bzw. CEO-Sicht benannt, eher für die Abbildung des gesamten Unternehmens und seines Umfeldes geeignet und dienen als unterstützendes Werkzeug für das strategische Management (siehe Abbildung 4). Sie stellen neben den Hauptprozessgruppierungen auch die mit dem Unternehmen in Verbindung stehende Marktteilnehmer dar und benennen diese explizit, im Gegensatz zum modellierten Unternehmen selbst, das implizit durch seine Prozesse definiert ist. In den darunter liegenden „Level 2“ und „Level 3“ Ebenen lassen sich dann die gewünschten Geschäftprozesse als Kombination von vordefinierten Teilprozessen zum Beispiel als Flussdiagramm modellieren. Die eTOM enthält hierfür über 400 ausführlich beschriebene Prozesse und stellt es dem Anwender darüber hinaus frei, zusätzliche Prozesse zu definieren oder auf einer selbst zu erstellenden fünften Ebene die „Level 3 Prozesse“ weiter in Teilprozesse zu zerlegen.

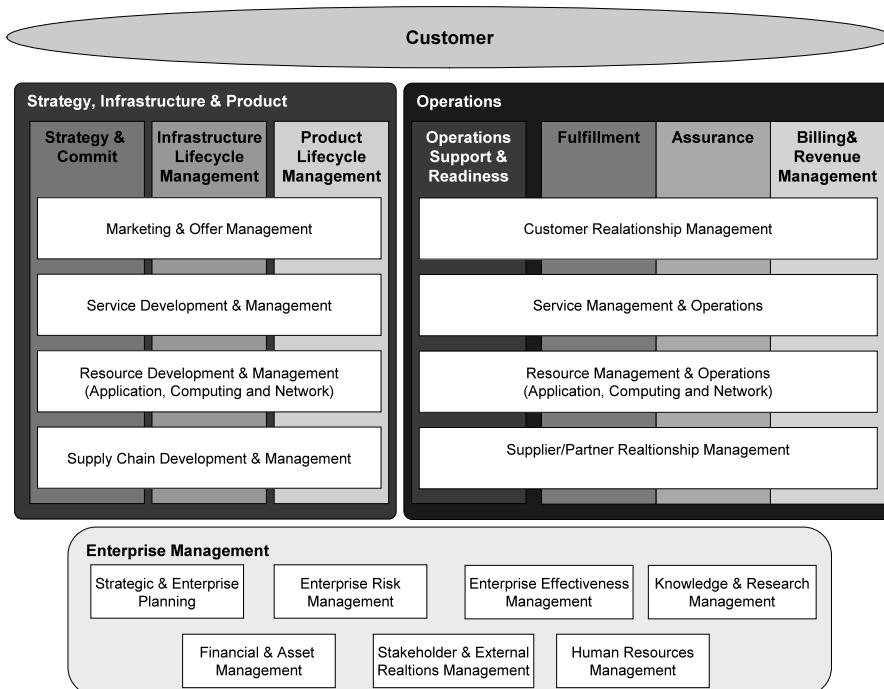


Abbildung 4: Enhanced Telecom Operations Map Level 1 nach [Tm09b]

Zur Einordnung der Prozesse sieht das Referenzprozessmodell eine Matrix aus horizontalen und vertikalen Prozesskategorien vor. Die horizontalen Prozesskategorien stellen dabei einzelne Stufen im Prozessfluss innerhalb des Unternehmens vom Lieferanten zum Kunden dar. Sie sind hierarchisch geordnet und bauen aufeinander auf. Die unterste Stufe „Supplier / Partner Relationship Management“ enthält alle Prozesse, die für die externe Beschaffung von Ressourcen oder die Beziehungen zu in der Wertschöpfungskette vorgelagerten Unternehmen verantwortlich sind. Die darüber liegende Stufe „Resource Management & Operations“ bündelt Prozesse, die intern für die Verfügbarkeit von Ressourcen sorgen. Diese werden benötigt, um in der Kategorie „Service Management & Operations“ in konkrete Leistungen transformiert und letztendlich über die oberste Stufe „Customer Relationship Management“ an den Kunden gebracht zu werden.

Gekreuzt werden diese funktionalen Stufen von vertikalen Ende-zu-Ende-Prozesskategorien, die sich in die nicht direkt mit der Leistungserbringung beim Kunden verbundenen Support-Prozesse und den FAB-Block (Fulfillment, Assurance, Billing) teilen. Die FAB-Prozesse beschreiben gewissermaßen den typischen Lebenszyklus einer einzelnen Servicedienstleistung, die zuerst erbracht, dann hinsichtlich ihrer Qualität überwacht und zuletzt dem Kunden in Rechnung gestellt wird [Tm09b].

In Anlehnung an das Vorgehen der Referenzmodellierung kann man die eTOM-Ebenen 0 und 1 als Ordnungsrahmen auffassen, der die Phase der Entwicklung des Referenzmodellrahmens unterstützen kann. Der Ordnungsrahmen wird durch die beiden weiteren eTOM - Ebenen 2 und 3 ausdetailliert, die somit zur Konstruktion der Referenzmodellstruktur herangezogen werden können. In den beiden folgenden Abschnitten wird zunächst ein Prozess eines Messdienstleisters beschrieben, der daraufhin beispielhaft auf die Ebenen 0 bis 3 der eTOM angewendet wird.

3.2 Beispielprozess eines Messdienstleisters

Prinzipiell lassen sich die in der eTOM beschriebenen Prozessflüsse, Strukturen und Kategorien in ähnlicher Weise auch bei Dienstleistungsunternehmen der Stromwirtschaft finden. Im Detail weisen sie jedoch Unterschiede auf, die eine Anpassung des Rahmenwerks erfordern.

Als Beispiel soll hier die Neubeauftragung eines Messdienstleisters durch einen Kunden dienen, der bisher alle mit der Stromversorgung verbundenen Dienstleistungen von einem Anbieter, z.B. dem regionalen Stadtwerk, bezogen hat und nun die Vorteile eines Smart Meters mit Fernauslese nutzen möchte. Der Prozess ist repräsentativ, da er mit der Einführung der „Wechselprozesse im Messwesen“ (WiM) durch die Bundesnetzagentur in 2011 in dieser oder ähnlicher Form verbindlich für den deutschen Energiemarkt wird. Der Prozess besitzt also Soll-Charakter und kann als Referenz angesehen werden.

Da der Messdienstleister in diesem Fall auch für die Installation und Wartung des intelligenten Zählers verantwortlich ist, ist er gleichzeitig auch Messstellenbetreiber. Der betrachte Prozess wird durch den Auftrag des Kunden angestoßen. Die darauf folgenden Prozessschritte sind größtenteils durch die Geschäftsprozessordnung „Messstellenbetreiber- und Messdienstleisterprozesse bei Strom und Gas“ festgelegt [Bd09]. Der Beispielprozess entspricht dabei einem Ausschnitt des Prozesses „Messstellenumbau bei Beginn oder Ende Messstellenbetrieb“, wie in Abbildung 5 veranschaulicht. Dieser konkrete Prozess wurde gewählt, da hier nicht nur die Kommunikation mit einem Kunden sondern auch die Interaktion mit anderen Anbietern eine Rolle spielt. Somit kann hierüber die interorganisationale Eignung des Prozessmodells geprüft werden. Darüber hinaus muss ein Referenzprozessmodell für Dienstleister der Energiewirtschaft, die von der Bundesnetzagentur definierten und zwingend vorgeschriebenen Prozesse abdecken können, da dies sonst als Ausschlusskriterium zu werten wäre.

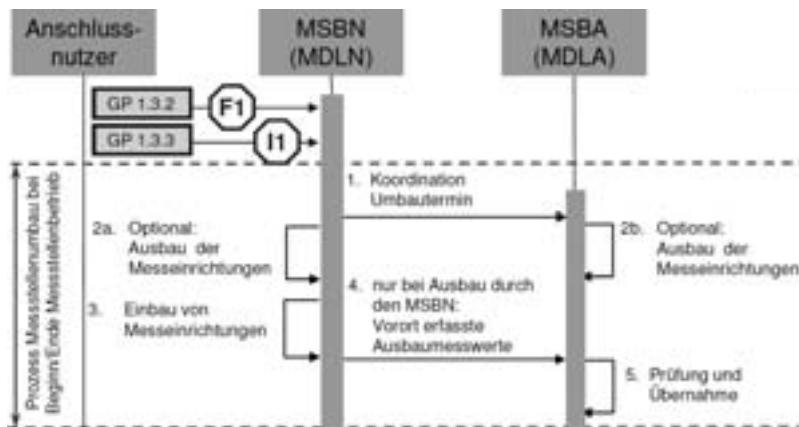


Abbildung 5: Ausschnitt des Sequenzdiagramms „Messstellenumbau bei Beginn oder Ende Messstellenbetrieb“ [Bd09]

Demnach vereinbaren zuerst neuer (MSBN) und alter (MSBA) Messstellenbetreiber einen Umbautermin und klären, wer den alten Zähler zu diesem Termin ausbaut. Im Falle des Ausbaus durch den neuen Messdienstleister wird zusätzlich vereinbart, ob das Altgerät von ihm entsorgt wird oder eine Rückgabe an den Eigentümer stattfindet. Falls der neue Zähler über Fernauslesedienste verfügt beginnt die Messdienstleistung unmittelbar mit Installation der Messstelle. Der Prozess endet mit der Meldung des letzten Zählerstandes durch den neuen an den alten Messdienstleister und dessen Prüfung und Übernahme der Daten. Durch die Spezifikation der Messstellenbetreiber- und Messdienstleisterprozesse bei Strom wird nur der Nachrichtenaustausch zwischen den verschiedenen Marktteilnehmern geregelt. Zusätzlich dazu werden im Folgenden auch die internen Prozesse des neuen Messdienstleisters mit Hilfe der eTOM modelliert.

3.3 Anwendungen der eTOM auf den Beispielprozess

In diesem Abschnitt wird die eTOM auf einen Beispielprozess angewendet. Folgende Kriterien werden hierbei überprüft:

- **Grad der Anpassung:** Es wird bewertet, wie hoch der Grad der notwendigen Anpassung des Originals (eTOM) ist, um den stromwirtschaftlichen Beispielprozess bis auf Level 3 zu modellieren.
- **Klarheit der Modellierung:** Es wird überprüft, ob die verwendeten Bezeichnungen für die Funktionen auf Level 0 – 3 verständlich und eindeutig sind.
- **Perspektivität:** Es wird darüber hinaus evaluiert, ob sich die eTOM für die spezielle Perspektive eignet, die bei Beispielprozess eingenommen wird, und ob auch weitere stromwirtschaftliche Rollen mit eTOM abgedeckt werden können.
- **Sichten:** Mit der Berücksichtigung von Funktions-, Daten- und Prozesssichten muss das Referenzmodell für die Lieferung und Messung sowohl Struktur- als auch Verhaltenssichten abbilden können.

Im ersten Schritt wird anhand des Beispielprozesses untersucht, ob sich die eTOM Level 0 und 1 für die Konstruktion des Modellrahmens eignen. Zunächst sei deshalb der Messdienstleister auf der obersten eTOM-Ebene betrachtet, um ihn hinsichtlich seiner Kernprozesse zu charakterisieren und im Marktumfeld einzurordnen.



Abbildung 6: eTOM Level 0 eines Messdienstleisters in Anlehnung an [Tm09b]

Neben den drei Hauptprozessgruppierungen zeigt die eTOM die mit dem Unternehmen in Geschäftsbeziehungen stehenden Marktteilnehmer und definiert die elementarsten Querschnittsprozesse. An den Beginn der innerbetrieblichen Wertschöpfung setzt die eTOM die Kommunikation und Interaktion mit Zulieferern und Partnern. Wie in Abbildung 6 dargestellt, lassen sich diese Querschnittsprozesse auf den in Abschnitt 3.2 beschriebenen Anwendungsfall anwenden. Absatzmarktseitig sind das die Entwicklung und Bereitstellung der angebotenen Produkte und Dienstleistungen sowie die Betreuung der Schnittstelle zum Kunden. Darunter liegend werden die für die Wertschöpfung essentiellen Services benannt, die auf die firmeninternen Ressourcen zurückgreifen. In diesem Fall der Betrieb der Server, an die die Zähler ihre Daten senden und die gesamte Softwarelandschaft des Unternehmens, die zur Abwicklung des Prozesses benötigt wird.

Nachdem das Unternehmen jetzt aus konzeptueller Sicht betrachtet wurde, kann eine detailliertere Beschreibung auf „Level 1“ erfolgen. Die Querschnittsprozesse werden aufgeschlüsselt und je nach Beschaffenheit den Hauptprozessgruppierungen „Strategy, Infrastructure & Product“ oder „Operations“ zugeordnet und entsprechend ihrer Funktion benannt. Zusätzlich kommen die vertikalen Ende-zu-Ende-Prozesse hinzu und vervollständigen das Ordnungsschema des Referenzprozessmodells.

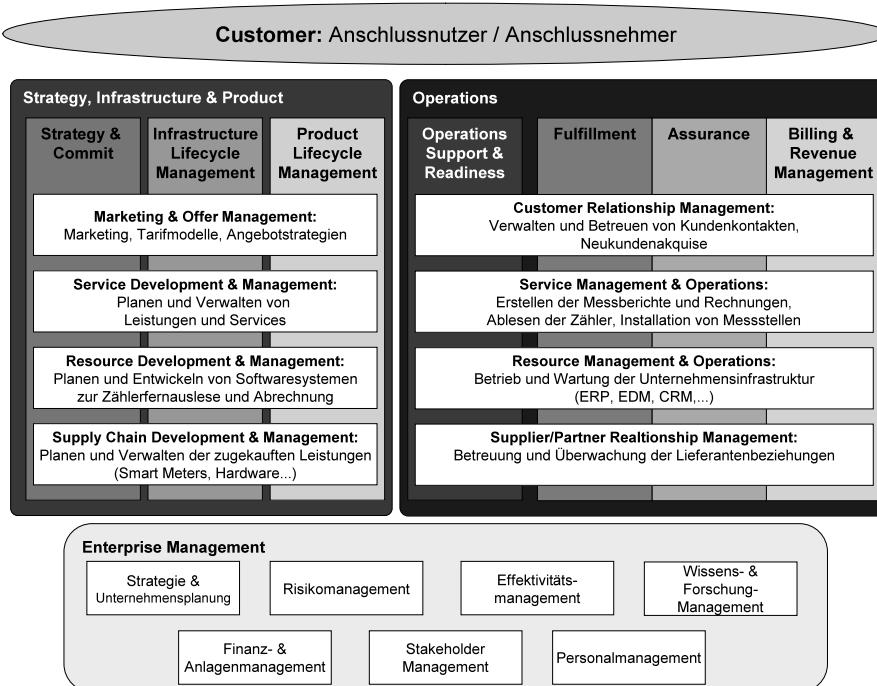


Abbildung 7: eTOM Level 1 eines Messdienstleisters in Anlehnung an [Tm09b]

Der betrachtete Messdienstleister kann aus „Level 1 Sicht“ modelliert werden, indem die Querschnittsprozesse spezifiziert werden, wie in Abbildung 7 illustriert. Hier lässt sich bereits der oben beschriebene Beispielprozess verorten, der sich gänzlich in der Hauptprozessgruppe „Operations“ abspielt. Zuerst beauftragt der Anschlussnutzer den Messdienstleister mit der Modernisierung des Zählpunktes und der darauf folgenden Messung. Alle direkt mit dem Kunden in Verbindung stehenden Prozesse, wie z.B. eine vorhergehende Beratung, der Vertragsabschluss oder die Festsetzung eines Termins zum Zählerwechsel finden sich im Funktionsblock Customer Relationship Management im Bereich Fulfillment. Auch bei der Erstellung und Erbringung der eigentlichen Leistung und der Bereitstellung der dafür benötigten Ressourcen sind ausschließlich Fulfillment-Prozesse beteiligt. Auf der untersten Ebene regelt der Messdienstleister die Beziehungen zu Netzbetreiber und Energielieferant und legt vertraglich fest, welche Leistungen bei welchen Qualitätsstandards von ihm zu erfüllen sind und in welcher Form die zukünftige Kommunikation unter den Marktteilnehmern stattfindet. Eine solche Vereinbarung kann das Datenformat der übermittelten Messdaten fest schreiben, die Abrechnungsmodalitäten regeln oder entsprechende Entgelte beziffern.

Es kann zusammenfassend festgestellt werden, dass sich auf Level 0 und 1 der eTOM der Anwendungsfall des Messstellenumbaus ohne größere Anpassungen darstellen lässt. Die Ebenen können so als Grundlage für einen Ordnungsrahmen des zu entwickelnden Referenzmodells dienen.

Anschließend wird der Beispielprozess auf Ebene 2 und 3 ausdetailliert. Wenn man den Geschäftsprozess mit Installation des Zählers als beendet betrachtet, ergeben sich auf Level 2 folgende in Abbildung 8 hervorgehobene Funktionen, mit denen der gesamte Geschäftsablauf beschrieben werden kann. Diese Funktionen auf Ebene 2 fassen die verwendeten Funktionen der Ebene 3 zusammen, die unten beschrieben werden. Das Ende des Geschäftsprozesses kann innerhalb gewisser Grenzen frei gewählt und die unmittelbar nachfolgenden Unternehmensaktivitäten wie die Überwachung der Qualität der Messdaten oder die Abrechnung als separate Prozesse modelliert werden. Bei der Wahl der Modellierungsbreite ist jedoch besonders bei der Definition von Referenzprozessen auf eine ausgewogene Balance zwischen einem ausreichend hohem Detaillierungsgrad und, besonders im Sinne der Standardisierung, auf Übersichtlichkeit und klare Prozessgrenzen zu achten. Als mögliches Kriterium bietet sich an, einen Prozess als beendet zu betrachten, wenn eine Leistungserbringung beim Kunden erfolgt ist.

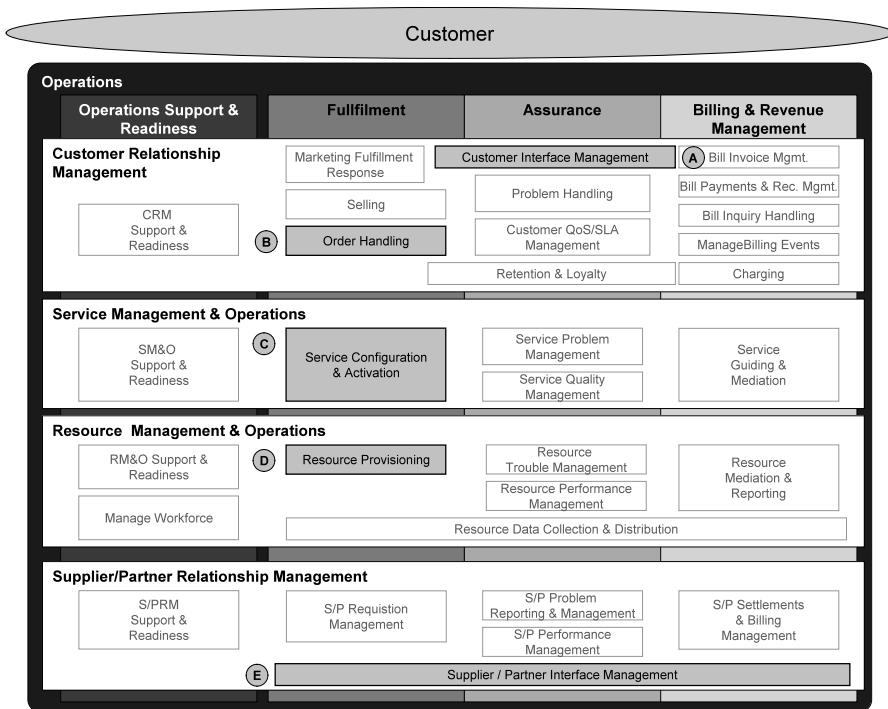


Abbildung 8: eTOM Level 2 für den Beispielprozess in Anlehnung an [Tm09b]

Da eTOM weder Modellierungssprache noch Modellierungstool vorschreibt, gewisse Standards für eine unternehmensübergreifende Kommunikation aber unabdingbar sind, wird der Beispielprozess mithilfe der Modellierungssprache „Business Process Modeling Notation“ (BPMN) auf Level 3 dargestellt. BPMN eignet sich zur verständlichen grafischen Notation von Geschäftsprozessen für alle Beteiligten des Prozessmanagements wie Mitarbeiter der Fachabteilung, Modellierer und Entwickler von IT-Systemen (vgl. z.B [Wh04], [OM06]). Der Prozess wurde auf Level 3 vollständig neu modelliert, allerdings wurde hierbei nach Möglichkeit auf vordefinierte Elemente des Funktionsbaums von eTOM zurückgegriffen. Deren Zusammenwirken und zeitliche Abfolge in einem Diagramm dargestellt werden (vgl. Abbildung 9). Die Buchstaben A-E ordnen dabei die Prozessschritte den in Abbildung 8 hervorgehobenen Prozessen auf Level 2 zu.

Der Kundenauftrag wird durch den Prozess „Manage Request“ bearbeitet. Hier wird geprüft und entschieden, ob der Auftrag angenommen wird. Wenn diese Frage positiv beantwortet werden kann, schließt sich der Teilprozess „Track & Manage Customer Order an“, der zum einen den Prozess „Mediate & Orchestrate Supplier/Partner Interactions“ initiiert, zum anderen aber bis zur Erfüllung des Auftrages den Gesamtprozess überwacht und auf Anfrage jeder Zeit über den Prozessstatus informieren kann. Im Supplier Relationship Management findet derweil ein Informationsaustausch - im Diagramm durch gestrichelte Pfeile dargestellt - zwischen neuem und altem Messstellenbetreiber statt, an dessen Ende eine Vereinbarung über die Modalitäten des Zählerumbaus steht. Diese Ergebnisse werden vom Prozess „Analyze & Report S/P Interactions“ wieder zurück an das Customer Relationship Management gegeben, in welchem dann der Prozess „Mediate & Orchestrate Customer Interactions“ einen Umbautermin mit dem Kunden vereinbart. Sind diese Grundvoraussetzungen erfüllt, kann der Prozess „Service Configuration & Activation“ spezifizieren, welche weiteren Prozessschritte zur Erfüllung des Kundenauftrages benötigt werden und diese anstoßen. Er legt darüber hinaus fest, welche Ressourcen eingesetzt werden sollen. Für die Bereitstellung dieser Ressourcen übernimmt dann der Prozess „Configure & Activate Resource“ die Verantwortung.

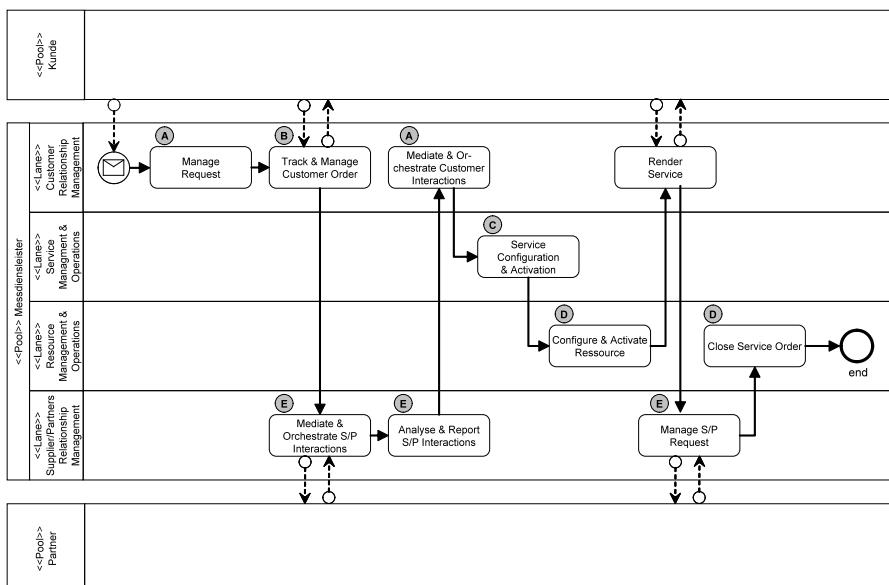


Abbildung 9: eTOM Level 3 für den Beispielprozess in Anlehnung an [Tm09b]

Bis hierhin lässt sich der Gesamtprozess mit den in der eTOM beschriebenen Level 3 Prozessen modellieren, für die konkrete Leistungserstellung beim Kunden ist allerdings kein Prozess vorhanden, so dass der Teilprozess „Render Service“, also der Umbau der Zähler beim Kunden für eine vollständige Modellierung hinzugefügt werden muss [Do07]. Wenn der Umbau erfolgt ist, erstattet der neue Messstellenbetreiber dem alten darüber Meldung und übermittelt ihm gegebenenfalls die Ablesewerte der ausgebauten Messeinrichtung. Mit dem Prozessschritte „Close Service Order“ wird der Prozess beendet. Mögliche Folgeprozesse wie zum Beispiel das Aufnehmen der Fernauslesung, die Überwachung der Servicequalität oder das Einpflegen der Kundenstammdaten können in separaten Prozessen modelliert werden.

3.4 Bewertung der Anwendbarkeit von eTOM auf den Anwendungsfall

Grad der Anpassung: Der Prozess des Messstellenumbaus konnte mit den Elementen der eTOM auf den Ebenen 0 bis 3 modelliert werden. Dabei ließen sich die Funktionen bis auf Ebene 2 in dem vorhandenen Rahmen verorten. Die Level 0 bis 2 sind sehr allgemein gehalten, weshalb hier die stromwirtschaftlichen Funktionen auch gut verortet werden konnten. Der Prozess auf Level 3 wurde neu definiert, jedoch basierend auf Funktionen, die durch die eTOM beschrieben werden [Tm09a].

Klarheit der Modellierung: Die sehr abstrakten Funktionen auf Level 0 bis 2 charakterisieren den Beispielprozess nicht eindeutig, jedoch findet eine Präzisierung auf Ebene 3 statt. Eine ausführliche Beschreibung der einzelnen Funktionen auf den unterschiedlichen Ebenen (wie bei eTOM vorhanden) würde weitere Klarheit schaffen.

Perspektivität: Der Beispielprozess deckt jedoch nur einen kleinen Ausschnitt der Anwendungsdomäne ab, die durch das zu entwickelnde Referenzmodell erfasst werden soll. Für verschiedene stromwirtschaftliche Rollen bestehen Anforderungen an ein Referenzprozessmodell. Es ist also zu prüfen, ob ein Referenzmodell basierend auf eTOM für die Funktionen der Lieferung und Messung möglich ist oder ob es eine spezifische Ausprägung je nach energiewirtschaftlicher Rolle geben muss. Denkbar wäre eine für alle Marktteilnehmer zu Verfügung stehende Basis der elementarsten und von allen benötigten Prozessen. Darauf aufbauend könnten die rollenspezifischen Prozesse in einzelnen Modulen zusammengefasst werden, sodass sich jedes Unternehmen, je nachdem welche Markttrollen es besetzt, ein individuelles Referenzprozessmodell zusammenstellen könnte, die Interoperabilität aber trotzdem gewahrt bleiben würde.

Sichten: Die eTOM deckt vor allem funktionale und prozessuale Aspekte ab (als Verhaltenssichten). Diese stehen zwar für das Anwendungsgebiet der Organisationsgestaltung bzw. des Prozessmanagements im Vordergrund, jedoch muss eine zusätzliche Datensicht berücksichtigt werden.

Nach jetziger Einschätzung kann eTOM Level 0 und Level 1 als Grundlage herangezogen werden, um darauf aufbauend einen Ordnungsrahmen für das zu entwerfende Referenzmodell zu entwickeln, das auf die Besonderheiten der Branche zugeschnitten ist.

4 Zusammenfassung und Ausblick

EVU und Dienstleistungsunternehmen in der Stromwirtschaft müssen mit den gestiegenen Wettbewerbsanforderungen aus der Liberalisierung umgehen. Ein wichtiger Aspekt hierbei ist eine angemessene Organisationsgestaltung und eine stärkere Prozessorientierung. Hierfür fehlen zurzeit jedoch geeignete Ordnungsrahmen in Sinne von Referenzmodellen. Im Beitrag wurde daher die Anwendbarkeit der eTOM auf Dienstleistungsunternehmen der Stromwirtschaft überprüft. Anhand eines konkreten Beispiels ließ sich zeigen, dass grundsätzliche Anforderungen abgedeckt werden konnten. Einige Aspekte lassen eine Anpassung der eTOM auf die Besonderheiten der Stromwirtschaft jedoch als notwendig erscheinen. So sind die Rollen in der Stromwirtschaft stärker getrennt als in der Telekommunikationsindustrie. Neben den Netzbetreibern und Lieferanten bestehen weitere Rollen wie Erzeuger, Zwischenhändler, Messdienstleister und Messstellenbetreiber. Es muss noch gezeigt werden, für welche dieser Rollen die eTOM einen angemessenen Bezugspunkt darstellen kann. Darüber hinaus sind die Detailprozesse der Ebene 3 nach wie vor stark auf die Telekommunikationsbranche ausgerichtet. So müssen vor allem Begrifflichkeiten aus dem Stromsektor ergänzt werden, um die Akzeptanz in diesem Umfeld zu erhöhen.

Im Artikel wurde die Analogiebetrachtung anhand eines einzelnen Prozesses durchgeführt. Um eine belastbare Aussage treffen zu können, ob sich eTOM für eine Adaption an die stromwirtschaftlichen Funktionen eignet, müssen jedoch deutlich mehr Prozesse untersucht werden. Relevant sind hierbei weitere Wechselprozesse im Messwesen und die Geschäftsprozesse für die Belieferung der Kunden mit Elektrizität [BU07]. Im Rahmen der in Abschnitt 2 vorgestellten Arbeiten wurden bereits Lieferantenprozesse wie der Lieferbeginn und der Lieferantenwechsel betrachtet. Auch hier wurde eine grundsätzlich Eignung von eTOM für den Analogieschluss festgestellt.

Die Eignung von eTOM für stromwirtschaftliche Prozesse im Bereich der Lieferung und Messung sollte darüber hinaus sowohl formal zum Beispiel nach den Grundsätzen ordnungsmäßiger Modellierung evaluiert werden [Be95], als auch durch Fallstudien bzw. durch exemplarische Anwendung praktisch überprüft werden.

Auf das Detaillierungsniveau bezogen bleibt eTOM sehr abstrakt, d.h. sie stellt nur einen groben Bezugsrahmen dar, der noch umfangreich auf das jeweils betrachtete Unternehmen angepasst werden muss. Um eine stärkere normative Komponente bei Business Management Projekten bereitstellen zu können, ist eine weitere Ausdetaillierung zu prüfen. Hierbei müsste eine weitere Ebene unter der vorhandenen Level 3 Ebene definiert werden. Spätestens dann ist auch die Verwendung einer standardisierten Modellierungsnotation sinnvoll, die mehr oder weniger implementierungsfähig ist. Neben der besonders in Deutschland weit verbreiteten eEPK käme hier vor allem die BPMN in Betracht. Eine Bewertung der Eignung der verschiedenen Modellierungsnotationen steht jedoch noch aus.

Arbeiten in der Zukunft werden sich zudem mit der Erweiterung des hier vorgestellten Referenzmodells um korrespondierende Datenmodelle beschäftigen. Dies ist zwingend notwendig, wenn das Modell nicht nur zur Prozess(re)organisation sondern auch als Grundlage für die Implementierung von IT-Systemen dienen soll. Auch hierbei gibt es Vorbilder durch die Arbeiten des TM Forums. Das zu eTOM zugehörige Datenmodell SID (Shared Information and Data Model) [Tm09c] kann als Ausgangspunkt genommen werden, um ein Datenmodell für den Stromsektor zu entwickeln. Auch hier können vermutlich einige generische strukturbestimmende Merkmale übernommen und durch spezifisch branchenbezogene ergänzt werden.

Literaturverzeichnis

- [Bd09] Bdew; Vku, BEN, AfM+E: Messstellenbetreiber- und Messdienstleisterprozesse bei Strom und Gas, Stand: 23.02.2009, 2009.
- [Be95] Becker, J.; Rosemann, M.; Schütte, R.: Grundsätze ordnungsmäßiger Modellierung. In: Wirtschaftsinformatik 37 (1995), S. 435-445.
- [Be98] Becker, J.; Rosemann, M.; Schütte, R.: Referenzmodellierung: State-of-the-art und Entwicklungsperspektiven, 1998.
- [Be03] Becker, J.; Rosemann, M.; Schütte, R.: Grundsätze ordnungsmässiger Modellierung. In: Wirtschaftsinformatik 37 (2003), S. 435-445.
- [Be04] Becker, J.: Referenzmodellierung. X, 180. Physica-Verl: Heidelberg, 2004.
- [Be08] Becker, J.: Prozessmanagement : ein Leitfaden zur prozessorientierten Organisationsgestaltung. Springer: Berlin [u.a.], 2008.
- [Be09] Beenken, P. et al.: Sicherheitsorientierte Gestaltung von Anwendungslandschaften in der Energiewirtschaft. VDE VERLAG GmbH, 2009.
- [Bo07a] Boltkart, A.: Entflechtung und Regulierung in der deutschen Energiewirtschaft, Rudolf Haufe GmbH & Co. KG: München, 2007.
- [Bo07b] Bolstorff, P.A.; Poluha, R.G.; Rosenbaum, R.G.: Spitzenleistungen im Supply Chain Management. Online-Ressource. Springer-Verlag: Berlin, Heidelberg, 2007.
- [Br07] Brenner, M.: Werkzeugunterstützung für ITIL-orientiertes Dienstmanagement. 244. Books on Demand: Norderstedt, 2007.
- [Bu07] Bundesnetzagentur: Umsetzung des Beschlusses BK6-06-009, 2007.
- [Co10] Logica Management Consulting: Die Energiewirtschaft wird smart, Auswirkungen von Smart Grids auf die IT deutscher Energieversorger, 2010.
- [Da04] Dalal, N.P. et al.: Toward an integrated framework for modeling enterprise processes. In: Communications of the ACM 47 (2004) 3, S. 83-87.
- [De10] Deindl, M. et al.: Reference Process Modelling for Utility. In: Referenzprozessmodellierung für Unternehmen der Energiewirtschaft 52 (2010), S. 65 - 72.
- [Do07] Dous, M.: Kundenbeziehungsmanagement für interne IT-Dienstleister. XIX, 237. Dt. Univ.-Verl: Wiesbaden, 2007.
- [FeLo04a] Fettke, P.; Loos, P.: Referenzmodellierungsforschung. In: Wirtschaftsinformatik 46 (2004), S. 331-340.
- [FeLo04b] Fettke, P.; Loos, P.: Systematische Erhebung von Referenzmodellen – Ergebnisse der Voruntersuchung, 2004.
- [Fü08] Fürlinger, B.: eTOM - Enhanced Telecom Operations Map: Design und Erstellung von Telekom-Referenzprozessen, 2008
- [Ha02] Haberfellner, R. et al.: Systems Engineering-Methoden und Praxis. In: Auflage, Verlag Industrielle Organisation, Zürich, 2002.
- [Ha04] Haeusler, O.; Schwicker, A.; Ebersberger, S.: Arbeitspapiere Wirtschaftsinformatik. Online-Ressource. Univ: Gießen, 2004.
- [Ha08] Haller, T.; Hoffmann, S.; Rentschler, M.-D.: Smart Metering - neue Möglichkeiten für den Energievertrieb im Massenkundenbereich. In: et - Energiewirtschaftliche Tagesfragen 6 (2008), S. 54-59.
- [Ke04] Keber, B.: Applying eTOM to Public Utilities Industry, 2004.
- [OM06] Object Management Group (OMG): Business Process Modeling Notation Specification, 2006.
- [Po05] Poluha, R.G.: Anwendung des SCOR-Modells zur Analyse der Supply Chain, 2005.

- [RoSc99] Rosemann, M.; Schuette, R.: Multiperspektivische Referenzmodellierung. In: Referenzmodellierung. State-of-the-Art und Entwicklungsperspektiven. Heidelberg (1999), S. 22-44.
- [Sc01] Scheer, A.W.: ARIS-Modellierungsmethoden, Metamodelle, Anwendungen. Springer, 2001.
- [ScNü00] Scheer, A.W.; Nüttgens, M.: ARIS architecture and reference models for business process management. In: Business Process Management: Models, Techniques, and Empirical Studies (2000), S. 376-389.
- [Sc00] Schlagheck, B.: Objektorientierte Referenzmodelle für das Prozess-und Projektcontrolling. Dt. Univ.-Verl., 2000.
- [St10] Steiner, I.: Supply Chain Management im Netzsgment der Energiewirtschaft. In: et - energiewirtschaftliche Tagesfragen 3 (2010).
- [Tm09a] TM Forum (TMF): Business Process Framework - Addendum D: Process Decompositions and Descriptions. Release 8.0, 2009.
- [Tm09b] TM Forum (TMF): eTOM The Business Process Framework Release 8.0, 2009.
- [Tm09c] TM Forum (TMF): Shared Information/Data (SID) Model - Business View Concepts, Principles, and Domains, 2009.
- [Us09] Uslar, M.; Schwarz, K.; Hein, F.: Untersuchung des Normungsumfeldes des BMWi-Förderschwerpunktes "e-Energy - IKT-basiertes Energiesystem der Zukunft", Studie für das BMWi, 2009.
- [Br03] Vom Brocke, J.: Referenzmodellierung. Logos, 2003.
- [Wh04] White, S.A.: Introduction to BPMN. In: IBM Cooperation (2004), S. 2008-2029.
- [Wi09] Wildemann, H.: Stadtwerke, Erfolgsfaktoren europäischer Infrastruktur- und Versorgungsdienstleister, 2009.

Komponentenorientierte betriebliche Anwendungssysteme

Service- und komponentenorientierte Informationssystemarchitekturen für die strategische Beschaffung von hybriden Produkten – ein Vergleichsrahmen

Holger Schrödl

Lehrstuhl für Wirtschaftsinformatik und Systems Engineering
Universität Augsburg
Universitätsstr. 16
86179 Augsburg
holger.schroedl@wiwi.uni-augsburg.de

Abstract: Unternehmen suchen zur Vermeidung von Niedrigpreisstrategien intensiv nach neuen Wegen, sich in einem globalen Markt vom Mitbewerb zu unterscheiden. Eine vielversprechende Antwort bieten hybride Produkte als Möglichkeit, bestehende Angebote eigenständiger Produkte und Services zu einer integrierten Problemlösung für spezifische Kundenanforderungen auszuweiten. Diese Differenzierungsstrategie führt zu einer steigenden Abhängigkeit zwischen dem anbietendem Unternehmen und dessen Lieferanten. Liefernetze fungieren im Gegensatz zu einer lieferantenzentrierten Beschaffungsstrategie als wesentlicher Wegbereiter in der Gestaltung integrierter Lösungsangebote. Es stellt sich dabei die Frage, wie solche Liefernetzwerke optimal mittels Informationssystemen modelliert werden können. Von besonderem Interesse ist dabei die Frage, welche Informationssystemarchitekturen geeignet sind, die Anforderungen für hybride Leistungsangebote in strategischen Liefernetzen in durchgängige Prozessunterstützung umzusetzen. Im vorliegenden Beitrag wird dazu ein Vergleichsrahmen für Informationssystemarchitekturen erstellt, der vier unterschiedliche Informationssystemarchitekturen auf die Eignung zur Modellierung strategischer Liefernetze für hybride Wertschöpfung untersucht. Im Rahmen einer Literaturstudie werden wesentliche Merkmale und Merkmalsausprägungen für einen Vergleichsrahmen identifiziert. Das Ergebnis sind Handlungsempfehlungen für den Einsatz geeigneter Informationssystemarchitekturen für die Identifikation, Bewertung und Auswahl von Lieferanten im Kontakt hybrider Wertschöpfung in Liefernetzen.

1 Hybride Wertschöpfung und strategische Liefernetze

Während hybride Leistungsbündel in der Praxis bereits Einzug gehalten haben, zeigt das wissenschaftliche Fundament erhebliche Defizite. Das Management von hybriden Leistungsbündel ist mit den etablierten Methoden der unternehmerischen Leistungsprozesse zur unzureichend realisierbar. Grundlage für ein erfolgreiches Management hybrider Wertschöpfung ist eine der Wertschöpfung angepasste IT-Architektur zur Realisierung unternehmensübergreifender Leistungs- und Informationsflüsse [KOP00].

1.1 Hybride Wertschöpfung

Globalisierung und steigende Vergleichbarkeit von Produkten fordert Unternehmen immer mehr dazu, neue Strategien zu entwickeln, sich gegenüber dem Mitbewerb zu differenzieren. Viele Produkthersteller reagieren darauf mit einem Wandel des Geschäftsmodells vom Produkthersteller hin zum Lösungsanbieter [JOH03]. Ein zentraler Aspekt für einen Lösungsanbieter sind dabei hybride Leistungsbündel. Hybride Leistungsbündel sind Produktformen, die sowohl Sachleistungen wie auch Dienstleistungen in integrierter Form anbieten. Durch diese Integration entstehen Leistungsangebote, deren Wertigkeit für den Kunden meist die Wertigkeit der einzelnen Teilleistungen übersteigt [JOH03],[KER06],[REI01].

Die Integration von Produkten und Dienstleistungen zu einem kombinierten Konstrukt wird in der Literatur nicht einheitlich bezeichnet. Neben dem Begriff des „hybriden Leistungsbündels“ werden auch Bezeichnungen wie „Produktbegleitende Dienstleistungen“, „Performance Contracting“, „Lösung“ und „hybrides Produkt“ verwendet [LEI08]. Diese unterschiedlichen Bezeichnungen resultieren aus der Art und Weise, wie die Bündelung von Sach- und Dienstleistungen zu einem integrierten Angebot vorgenommen wird. Produktbegleitende Dienstleistungen sind immaterielle Leistungen, die den Kundennutzen eines Angebots erhöhen, indem beispielweise zu einer Produktionsmaschine eine entsprechende Dienstleistung angeboten wird [VDM03]. Mit dem Begriff „Performance Contracting“ werden Leistungen bezeichnet, die keinen Produktverkauf mehr vorsehen, sondern der Kunde erwirbt eine Leistung zur Nutzung der Produkte mit entsprechenden Dienstleistungskomponenten [BAC01]. Die Bezeichnung „Lösung“ hat in der Literatur unterschiedliche Aspekte. So wird unter einer Lösung eine Kombination aus Produkten und Dienstleistungen verstanden, deren Integration einen höheren Kundenwert darstellt als die jeweiligen Einzelteile [JOH03]. Darüber hinaus wird der Aspekt der Problemlösung für einen Kunden als primärer Zweck einer Lösung betont [SAW06]. Die Begriffe „hybrides Produkt“ und „hybrides Leistungsbündel“ werden häufig synonym verwendet. Ein hybrides Produkt wird charakterisiert als Leistungsbündel aus einer speziell abgestimmten Kombination aus Sach- und Dienstleistungen mit dem Ziel einer auf den Kunden ausgerichteten Problemlösung [KER06].

Hybride Leistungsbündel sind signifikant in das Wertschöpfungssystem des Kunden eingebettet [BÖH06]. Die Integration von Produkten und Dienstleistungen im Sinne eines hybriden Leistungsbündels erfolgt zum einen durch eine technische Integration, zum Zweiten durch eine prozessurale Integration. Durch die Gestaltung entsprechender Schnittstellen auf technischer und prozessurale Ebene werden hybride Leistungsbündel in die Systemlandschaft des Kunden integriert. Zusätzlich zu dieser Systemintegration findet auch eine ökonomische Integration statt [BAC01], [BUR02]. Betriebliche Leistungsprozesse des Kunden, die die Wertschöpfung des Kunden ausmachen, werden damit ergänzt und erweitert durch in die Leistungsprozesse integrierte hybride Leistungsbündel.

1.2 Strategische Liefernetzwerke

Die Relevanz einer Beschaffungsfunktion im Unternehmen ist über die letzten drei Jahrzehnte konstant gewachsen. Von einer eher operationalen Sicht ohne Einfluss auf Langfrist-Strategien in den 70er Jahren [MCI98] über die Erkennung der Wichtigkeit eines Supplier Relationship Managements in den 80ern bekommt die Beschaffungsfunktion mittlerweile strategische Relevanz [KAU02] und wird damit zu einer Kernfunktion moderner Unternehmen. Der signifikante Zusammenhang zwischen der Einrichtung einer strategischen Beschaffungsfunktion und dem finanziellen Erfolg eines Unternehmens ist empirisch nachgewiesen [CAR99a].

Eine enge Beschaffer-Lieferant-Beziehung ist notwendig, um Prozessverbesserung in der Beschaffung sowie Kostenreduktionen zu erreichen, die dann zwischen Beschaffer und Lieferant aufgeteilt werden können [CAR99b]. Daher sind Liefernetze geeignet, neben operationalen Aspekten in der Beschaffung auch strategische Aspekte zu realisieren. Im Folgenden wird als strategisches Liefernetz ein Netzwerk von Lieferanten bezeichnet, die sich über mehrere Lieferebenen erstrecken und Langzeitbeziehungen fokussiert. Wie in Abbildung 1 dargestellt hat das strategische Liefernetz einen festen Teil, den fokalen Lieferanten, und einen dynamischen Teil, die Lieferanten der Ebenen tier-1 bis tier-n [ALB07]. Dieser flexible Teil ermöglicht eine bedarfsgerechte Erweiterung und Modifikation des Liefernetzes.

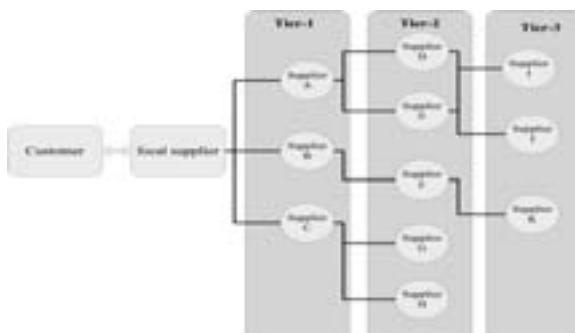


Abbildung 1: Strategisches Liefernetz

Um einen Wettbewerbsvorteil aus solchen Liefernetzen zu erzielen, erhält die Auswahl, die Entwicklung, das Management und die Integration von potenziellen und bestehenden Lieferanten zentrale Bedeutung. Moderne Informationssysteme werden benötigt, um nachhaltiges Erstellen und Management solcher Liefernetze zu erreichen [KOP00]. Schwierigkeiten bereiten dabei die Unterstützung von großen Netzwerken mit mehreren Lieferantenebenen. Der hohe Grad an Komplexität, der aus den dynamischen Änderungen im Liefernetz resultiert, ist der zentrale Grund für das Fehlen praktischer Implementierung. Weitere Gründe liegen in den Schwierigkeiten, Einheiten in Liefernetzen zu identifizieren, Netzwerkstrukturen zu modellieren sowie in dem hohen koordinativem Aufwand [LAM00].

1.3 Informationssystemarchitekturen

Informationssysteme verbinden die Elemente Mensch, Aufgabe und Technik und werden daher auch als soziotechnische Systeme bezeichnet [HEI90]. Informationssysteme werden in Organisationen eingesetzt, die dadurch betriebliche Funktionen realisieren können. Diese betrieblichen Funktionen basieren auf den Informationen, die im Informationssystem enthalten sind. Die Informationen werden den einzelnen Beteiligten der Organisation durch das Informationssystem zur Verfügung gestellt [DAV85].

Die Informationssystem-Architektur beschreibt die Anordnung der grundlegenden Systemelemente eines Informationssystems, ihre Verknüpfung miteinander sowie Prinzipien für ihre Konstruktion und Weiterentwicklung. Um dem ganzheitlichen Charakter einer Informationssystem-Architektur gerecht zu werden, soll durch die Informationssystem-Architektur die Geschäftsstrategie, die Prozess- und Aufbauorganisations-Architektur, die Anwendungs-, Daten- und Kommunikations-Architektur sowie die Infrastruktur abgebildet werden [KRC90]. Die Informationssystem-Architektur beschreibt somit neben technischen Elementen auch fachliche Elemente. Der primäre Zweck der Informationssystem-Architektur liegt jedoch nicht in der Formulierung der Geschäftsstrategie, sondern in der Konstruktion von technischen Systemen, welche die fachlichen Anforderungen umsetzen.

Zur Verringerung der hohen Komplexität, die bei einer Gesamtsicht auf ein Informationssystem entsteht, wird meist eine ebenenbezogene Betrachtungsweise durchgeführt. Da Informationssysteme sowohl fachliche wie auch technische Elemente verbinden, differenzieren die einzelnen Betrachtungsebenen entweder auf strategisch-fachliche oder auf operativ-technische Aspekte. Winter und Fischer [WIN07] identifizieren fünf wesentliche Ebenen in einer Analyse verschiedener Ansätze zur IS-Architektur – die Strategieebene, die Organisationsebene, die Software- und Datenebene, die Integrationsebene sowie die Infrastrukturebene.

1.4 Aufbau des Beitrags

Der vorliegende Beitrag ist wie folgt aufgebaut: Nach der Darstellung des aktuellen Forschungsstandes zu hybrider Wertschöpfung und strategischen Liefernetzen in Kapitel 1 werden in Kapitel 2 vier verschiedene, für die Realisierung flexibler Informationssysteme geeignete Informationssystemarchitekturen analysiert. In Kapitel 3 werden auf Basis einer Literaturstudie Merkmale und Merkmalsausprägungen hybrider Wertschöpfung und strategischer Liefernetze identifiziert und darauf aufbauen eine Klassifikation erstellt. In Kapitel 4 wird die Klassifikation auf die ausgewählten Informationssystemarchitekturen angewendet und daraus Gestaltungsempfehlungen für die Modellierung von Informationssystemarchitekturen für hybride Wertschöpfung abgeleitet. In Kapitel 5 erfolgt eine Zusammenfassung sowie ein Ausblick auf weiteren Forschungsbedarf.

2 Informationssystemarchitekturen für die Modellierung strategischer Liefernetze zur hybriden Wertschöpfung

Im Rahmen einer Literaturrecherche wurden unterschiedliche Informationssystemarchitekturen untersucht, inwieweit diese geeignet sind, eine Modellierung strategischer Liefernetze zur hybriden Wertschöpfung durchführen zu können. Hierzu wurden Gestaltungsprinzipien identifiziert, die im Kontext service-orientierter Architekturen zur Anwendung kommen. Es sind diese die Gestaltungsprinzipien Modularisierung, Abstraktion, Granularität und Prozessorientierung [THO09]. Ergänzt werden diese Gestaltungsprinzipien noch durch das Prinzip der Fachorientierung. Fachkomponenten bieten die Möglichkeit, Funktionalitäten einer Fachdomäne zu kapseln und zentral als Service zur Verfügung zu stellen. Diese Fachkomponenten können dann mit geringem Aufwand in individuelle Anwendungssysteme integriert werden [TUR02a]. Auf Basis dieser Gestaltungsprinzipien werden im Folgenden vier unterschiedliche Informationssystemarchitekturen betrachtet.

2.1 Service- und komponentenorientierte Architekturen

Betriebliche Anwendungssysteme nutzen immer mehr das service-orientierte Paradigma, um flexibel und erweiterungsfähig zu sein. Service-orientierte Architekturen dienen hierbei als Möglichkeit, flexible Anwendungssysteme zu realisieren, die aus einer Vielzahl von autonomen Subsystemen zusammengestellt werden [KRA08]. Es werden hierbei nicht nur Anwendungssystemkomponenten betrachtet, die in einem Unternehmen liegen. Vielmehr werden Komponenten auch als Dienste über das Internet bezogen, um bedarfsgerecht betriebliche Anwendungssysteme zu realisieren [FRE02]. Ein wesentlicher Baustein service-orientierter Architekturen im betrieblichen Umfeld sind Fachkomponenten. Fachkomponenten bündeln eine spezifische Menge von Diensten und Subkomponenten, um diese dann in einer speziellen Anwendungsdomäne anbieten zu können [TUR02a].

Akzeptierte Implementierungsformen service-orientierter Architekturen sind dabei beispielsweise Web-Services [CHE04], Enterprise Service Bus [SCH05] oder Cloud-Architekturen [THO08].

2.2 Elektronische Marktplätze

Ein elektronischer Markt „basiert auf einem oder mehreren zusammenarbeitenden Informationssystemen, die einzelne oder alle Phasen einer Markttransaktion elektronisch unterstützen, integrieren oder gar automatisieren“ [MÜL97]. Die Marktplätze selbst können dabei auf unterschiedliche Produkte spezialisiert sein, z.B. Rohstoffe, Wirtschaftsgüter, Informationsgüter oder Dienstleistungen, was sich letztendlich aus der Zusammensetzung der Teilnehmer ergibt. Durch die Entwicklung automatisierter Verhandlungskonzepte könnten sowohl Käufer als auch Verkäufer automatisiert und autonom auf elektronischen Marktplätzen handeln. Weitere wichtige Charakteristika von elektronischen Marktplätzen sind die Vereinfachung der Informationsbeschaffung und Informationsauswertung sowie die Verringerung einer zeitlichen und räumlichen Distanz zwischen den Marktteilnehmern [SCH93].

2.3 Agentenbasierte Systeme

Der Einsatz von Software-Agenten, die im Namen menschlicher Anwender autonom wirtschaftliche Transaktionen durchführen, z.B. im Rahmen elektronischer Märkte und des elektronischen Geschäftsverkehrs, stellt eine Synthese von Konzepten der Künstlichen Intelligenz, der Wirtschaftsinformatik und der Wirtschaftswissenschaften dar. Der Begriff des Software-Agenten wird im Folgenden so verstanden:

„An agent is an encapsulated computer system that is situated in some environment, and that is capable of flexible autonomous action in that environment in order to meet its design objectives“ [WOO99]. Agenten unterscheiden sich von traditionellen Software-Anwendungen vor allem durch ihre Autonomie, die als zielgerichtetes und proaktives, d.h. selbst-startendes Verhalten begriffen werden kann. Diese Eigenschaft erlaubt es ihnen, in einer definierten Umgebung zusammen mit anderen Agenten kontinuierlich und selbstständig Aufgaben zielgerichtet auszuführen. Diese Aufgaben lassen sich dabei in unterschiedliche Aufgabengebiete einteilen (vgl. [ROS00]: Präsentationsagenten, Aufgabenagenten und Informationsagenten. Eine weitere Differenzierung führt auf die Einsatzmöglichkeiten von Filteragenten [GRO99] zur automatisierten Informationsfilter, Notifikationsagenten [HAW95] und Reminder-Agenten zur automatisierten Bearbeitung von wiederkehrenden Standardaufgaben [KNU01]. Durch diese Vielfalt sind sie daher gerade für den Einsatz in dynamischen Umgebungen mit einer Vielzahl sich ständig ändernder Informationen und parallel durchgeführten Prozesse geeignet. Der elektronische Geschäftsverkehr in offenen Netzen und dort insbesondere elektronische Märkte, stellen eine solche Umgebung dar.

2.4 Elektronische soziale Netzwerke

Elektronische soziale Netzwerke, die in der Literatur auch als „social software“ bezeichnet werden, sind Anwendungssysteme, die unter Ausnutzung von Techniken des Internets Aspekte der menschlichen Kommunikation, Interaktion und Zusammenarbeit unterstützen [SIX05]. Sie können daher in den Kontext von Groupware-Anwendungen eingeordnet werden. Im engeren Sinne kann social software als internetbasierte Anwendung gesehen werden, die Informations-, Identität- und Beziehungsmanagement in öffentlich zugänglichen sozialen Netzwerken unterstützen [SCH06]. Social software bietet den Anwendern eine Vielzahl von Funktionen zur Unterstützung von Zusammenarbeit. So können beispielsweise Freundschaften über Kontaktnetzwerke gepflegt werden sowie Informationen geordnet werden und diese Ordnung anderen Anwendern zugänglich gemacht werden. Social software kann unterschiedlich realisiert werden. Eine mögliche Klassifikation ist beispielsweise die Unterteilung in Weblogs, Wikis, Social Tagging Anwendungen und Social Network Anwendungen [RIC07].

3 Ein Vergleichsrahmen für Informationssystemarchitekturen zur Modellierung strategischer Liefernetze für hybride Wertschöpfung

3.1 Methodische Vorüberlegungen

Zur Entwicklung eines Vergleichsrahmens für Informationssystemarchitekturen zur Modellierung strategischer Liefernetze für hybride Wertschöpfung wird eine Klassifikation entworfen. Generell wird von einer Klassifikation gesprochen, wenn ein Betrachtungsobjekt zunächst nach bestimmten Merkmalen gegliedert wird und zu diesen Merkmalen die relevanten Ausprägungen ermittelt werden [ENG71]. Es findet dabei keine Verknüpfung der unterschiedlichen Kriterien statt [KNO72]. Für die hier entwickelte Klassifikation wird als Basis der Referenzrahmen für die Entwicklung strategischer Liefernetze verwendet [ALB06]. Aus diesem Referenzrahmen werden die relevanten Merkmale für einen Vergleichsrahmen abgeleitet. Die aus dem Referenzrahmen abgeleitete Klassifikation wird um spezifische, für hybride Wertschöpfung relevante Merkmale erweitert [SCH10]. Hierbei werden die Merkmale, die notwendig sind zur Realisierung hybrider Wertschöpfung, identifiziert und für den Vergleichsrahmen herangezogen. Insgesamt umfasst die Klassifikation acht Merkmale, die in zwei Merkmalsklassen zusammengefasst wurden.

3.2 Merkmale mit Bezug auf die Modellierung von strategischen Liefernetzen

Im Rahmen einer Modellierung strategischer Liefernetze werden verschiedene Funktionen angewendet, um ein auf einen Bedarf angepasstes Liefernetz zu etablieren. Diese Funktionen können in drei verschiedene Bereiche gegliedert werden: strategische Bedarfsplanung, strategische Netzwerkmodellierung und die strategische Netzwerkqualifikation [ALB06],[SCH10].

Die strategische Bedarfsplanung stellt einen Unternehmensrahmen für alle Beschaffungsvorgänge dar. Dieser Unternehmensrahmen besteht aus einer konsistenten und unternehmensweit gültigen Beschaffungsstrategie, einer Ermittlung des strategischen Bedarfs (Bedarfsplanung) sowie die Definition von Methoden und Werkzeugen zur Steuerung des Beschaffungsvorgangs. Die strategische Netzwerkmodellierung umfasst Methoden zur Identifikation, Bewertung und Auswahl von strategischen Liefernetzen. Die Methoden finden dabei nicht nur Anwendung auf die direkten potenziellen Lieferanten (Tier-1), sondern umfassen auch Lieferanten in darauffolgenden Lieferketten. Bei einer positiven Bewertung eines strategischen Liefernetzes wird das Liefernetz ausgewählt und über einen Verhandlungsprozess vertraglich an das beschaffende Unternehmen gebunden. Die Qualifikation von strategischen Liefernetzen dient der Qualitätsüberwachung und Qualitätssteigerung bestehender Liefernetze. In Abhängigkeit von Bewertungsergebnissen werden Maßnahmen initiiert, um die Qualität des strategischen Liefernetzwerkes zu verbessern.

3.3 Merkmale mit Bezug auf hybride Wertschöpfung

Hybride Wertschöpfung kann auf die Beteiligten in einem Liefernetzwerk unterschiedlich aufgeteilt werden. So kann die hybride Wertschöpfung rein auf Kundenseite, rein auf Anbieterseite oder in einer Mischform auftreten, bei der sowohl Anteile beim Kunden wie auch Anteile beim Anbieter liegen. Daher ist eine serviceorientierte Architektur für strategische Liefernetze nach dem Interaktionsgrad, also dem Grad des wechselseitigen aufeinander Einwirkens der beteiligten Akteure und/oder Systeme zu bewerten.

Ein weiteres Merkmal ist die Kategorisierung des hybriden Leistungsbündels. Die Komponenten eines hybriden Leistungsbündels können in vier Kategorien eingeteilt werden: standardisierte Kaufprodukte, standardisierte Dienstleistungen, kundenindividualisierte Kaufprodukte und kundenindividualisierte Dienstleistungen [BUR07]. Das hybride Leistungsbündel kombiniert Bestandteile aus diesen Kategorien zu einem Gesamtangebot an den Kunden.

Als drittes Merkmal wird der Grad der Integration der Leistungserbringung des hybriden Leistungsbündels betrachtet. Der Integrationsgrad der Leistungserbringung anhand von sieben Kriterien beschrieben werden, die Werte jeweils innerhalb eines Kontinuum zwischen niedrig und hoch annehmen können [BUR07]. Die Kriterien reichen dabei vom Grad der technischen Integration bis zur Variabilität der Leistungserbringung. Da in Sinne eines Kontinuums keine klare Kategorisierung möglich ist, wird zum Zwecke der Beurteilung das Kontinuum in die drei Abschnitte klein, mittel und groß vorgenommen. Klein und groß folgt den Definitionen in [BUR07]. Für die Ausprägung „mittel“ wird definiert, dass hierfür drei der sieben Kriterien bei „klein“ anzusiedeln sind und vier der Kriterien bei „groß“ und analog in der Umkehrung.

Das vierte Merkmal beschreibt das Ergebnis des hybriden Leistungsbündels. Ergebnisse hybrider Leistungsbündel können in drei Kategorien eingeteilt werden. Die erste Möglichkeit ist die Produktorientierung. Ein Beispiel hierfür ist eine Produktionsmaschine, die mit einem Finanzierungsangebot verkauft wird. Die zweite Möglichkeit ist die Verfügbarkeitsorientierung. Ein Beispiel hierfür ist der Verkauf einer Produktionsmaschine mit einer Verfügbarkeitsgarantie für die Laufzeit der Maschine. Die dritte Möglichkeit ist die ergebnisorientierung. Ein Beispiel hierfür ist der Verkauf einer Produktionsleistung, die durch eine Produktionsmaschine erbracht wird [MEI05].

Als abschließendes Merkmal in der Merkmalsklasse der hybriden Wertschöpfung wird der Produktlebenszyklus betrachtet. Der Lebenszyklus eines hybriden Leistungsbündels kann mit den Ausprägungen Vornutzung, Nutzung und Nachnutzung beschrieben werden.

4 Anwendung des Vergleichsrahmens

Die in den bisherigen Abschnitten hergeleiteten Merkmale und deren Ausprägungen werden in der Tabelle 1 dargestellt. Hierzu wird zu jedem Merkmal und dessen Merkmalsausprägungen die Eignung der jeweiligen Architektur bewertet.

	Merkmal	Merkmalsausprägung	Service- und komponentenorientierte Architekturen	Marktplätze	Agenterbasierte Systeme	Soziale Netzwerke
Strategische Bedarfsplanung	Definition von Beschaffungsstrategien	✓	✗	✗	✗	✗
	Kalkulation von Beschaffungsbedarf	✓	✗	✗	✗	✗
	Definition von Managementfunktion	✓	✗	✗	✗	✗
Strategische Netzwerkmodellierung	Identifikation strategischer Netzwerke	✓	✓	✓	✓	✓
	Evaluation strategischer Netzwerke	✓	✓	✓	✓	(✓)
	Selektion strategischer Netzwerke	(✓)	✗	✓	✓	✗
Qualifikation strategischer Netzwerke	Qualitätsbewertung von Liefernetzen	✓	✗	(✓)	✓	✓
	Qualitätssteigerung von Liefernetzen	✓	(✓)	(✓)	✓	✓
Wechselseitige Interaktion	Kundenzentriert	(✓)	✓	(✓)	✓	✓
	Anbieterzentriert	✓	✓	✓	✓	(✓)
	Wechselseitig	(✓)	✓	(✓)	(✓)	(✓)
Leistungsbündelkategorisierung	standardisierte Kaufprodukte	✓	✓	✓	✓	✓
	standardisierte Dienstleistungen	✓	✓	✓	✓	✓
	kundenindividualisierte Kaufprodukte	(✓)	(✓)	(✓)	✓	✗
	kundenindividualisierte Dienstleistungen	(✓)	✗	(✓)	✓	✗
Integrationsgrad der Leistungserbringung	Klein	✓	✓	✓	✓	✓
	Mittel	✓	(✓)	✓	✓	✓
	Groß	(✓)	✗	(✓)	✓	✓

	Merkmal	Merkmalsausprägung	Service- und komponenten-orientierte Architekturen	Marktplätze	Agentenbasierte Systeme	Soziale Netzwerke
	Ergebnistyp	Produkt-orientierung	✓	✓	✓	✓
		Verfügbarkeits-orientierung	✓	✓	✓	✓
		Ergebnis-orientierung	(✓)	✗	✗	✗
Produktlebenszyklus	Vornutzung	✓	✓	✓	✓	✓
	Nutzung	✓	✓	✓	✓	✓
	Nachnutzung	✓	(✓)	✗	✗	✗

Legende: ✓ Merkmal erfüllt, (✓) Merkmal teilweise erfüllt, ✗ Merkmal nicht erfüllt

Tabelle 1: Klassifikation von Informationssystemarchitekturen

Der Gegenüberstellung ist zu entnehmen, dass es beim aktuellen Entwicklungsstand der unterschiedlichen Informationssystemarchitekturen keine durchgängige Lösung gibt, strategische Liefernetze für hybride Wertschöpfung zu realisieren. Service- und komponentenbasierte Architekturen zeichnen sich zwar durch eine größtmögliche Übereinstimmung mit den Merkmalen aus, weisen aber in bestimmten Bereichen dennoch Defizite auf. So ist beispielsweise die Selektion strategischer Liefernetze nur eingeschränkt unterstützt. Die Vertragsverhandlungen, die in diesem Vorgang stattfinden, sind durch service-orientierte Architekturen aufgrund der fehlenden Eigendynamik nur in einfachen Konstellationen nutzbar [ALB06]. Durch die Anforderung, Service-Komponenten bei allen Beteiligten zur Verfügung zu stellen, um eine durchgängige Unterstützung zu erreichen, ist eine hybride Wertschöpfung, die eher auf Kundenseite stattfindet, nur eingeschränkt abbildbar. Weiterhin sind service-orientierte Architekturen limitiert, wenn es um die Unterstützung stark individualisierter hybrider Produkte sowie einen hohen Komplexitätsgrad hybrider Wertschöpfung geht. Als letztes Defizit kann die mangelnde Unterstützung von hybrider Wertschöpfung aufgeführt werden, die ergebnisorientiert ist.

Elektronische Marktplätze bieten keine durchgängige Möglichkeit, strategische Liefernetze für hybride Produkte zu modellieren. Zentraler Kritikpunkt dabei ist, dass elektronische Marktplätze als offene Systeme fungieren und daher keine zentrale Kontrollinstanz aufweisen [STE06]. Allerdings lassen sich Teilaspekte der Modellierung durch den Einsatz elektronischer Marktplätze wesentlich verbessern. So ist beispielsweise bei einer hybriden Wertschöpfung, die eher kundenseitig stattfindet, ein elektronischer Markt eine Möglichkeit, eine service-orientierte Architektur zu ergänzen. Die Strukturen von elektronischen Märkten ermöglichen eine flexible Herstellung hoch individueller Produkte mit einem hohen Anteil an Kundenintegration [SCH04]. Unter dem Schlagwort des „Reverse Marketing“ lassen sich Kundenwünsche und Kundenbedürfnisse frühzeitig in den Erstellungsprozess hybrider Produkte integrieren und führen somit zu einem signifikanten Wettbewerbsvorteil. Ein Beispiel hierfür ist die Ideenplattform www.ideastorm.com von DELL Computers. Auf diesem Markt können bestehende und potenzielle Kunden Ideen für neue Produkte eintragen. Um aus allen Ideen die besten herauszukristallisieren, können andere Kunden die Vorschläge für neue Produkte bewerten. Automatisch kommen die bestbewerteten Ideen aus der gesamten Ideenmenge zum Vorschein.

Agentenbasierte Architekturen bieten insgesamt eine gute Möglichkeit, strategische Liefernetze für hybride Wertschöpfung zu modellieren. Gegenüber service-orientierten Architekturen sind allerdings Defizite vor allem im Bereich der strategischen Bedarfsplanung vorhanden. Allerdings stellen agentenbasierte Systeme aufgrund ihrer Eigendynamik eine gute Möglichkeit dar, die strategische Netzwerkmodellierung, insbesondere die Selektion von strategischen Liefernetzen zu verbessern. Ein Anwendungsbeispiel hierzu lässt sich aus der Branche der Energieversorger beobachten. Energieversorger verstehen sich vermehrt als Anbieter hybrider Produkte bestehend aus den eigentlichen Leistungen, den Serviceangeboten sowie Dienstleistungen wie optimale beispielsweise optimale Preisbestimmung. Hierzu wurde bereits ein Smart Energy Grid vorgestellt [LAM10]. Es basiert auf einer Agententechnologie und ermöglicht intelligente Koordinationsmechanismen, um eine ausgewogene Verteilung von Energiebedarf und Versorgungskapazitäten bezogen auf entsprechende Leistungskapazitäten und Qualitätsansprüchen. Dies ermöglicht eine bedarfsorientierte und kundenspezifische Konfiguration der Energieversorgung. Es ist zu erwarten, dass agentenbasierte Systeme in den stattfinden Verhandlungsszenarien eine wesentliche Verbesserung der IT-Unterstützung des strategischen Beschaffungsprozesses liefern. Dabei kann eine schrittweise Implementierung durchgeführt werden, die von einfachen Kommunikationsagenten über ein Mehrfachagenten-System für Verhandlungen hin zu einem Mehrfachagenten-System für die Koordination von unternehmensübergreifenden Beschaffungsprozessen reicht [TUR02b].

Die Integration von sozialen Netzwerken in die Modellierung strategischer Liefernetze für hybride Wertschöpfung scheint eine vielversprechende Möglichkeit zu sein, die Defizite der etablierten Informationssystemarchitekturen in Bezug auf hybride Wertschöpfung auszugleichen. Zum einen bieten soziale Netzwerke eine erweiterte Möglichkeit in der Identifikation und Evaluation von strategischen Liefernetzen. Exemplarisch kann hier das Angebot der IBM Blue Pages genannt werden [KOC07]. Dieser Dienst ermöglicht eine Expertensuche zu spezifischen Themen und umfasst eine Basis von mehr als 450.000 Expertenprofilen. Innerhalb dieses Dienstes können Funktionen wie Tagging, Instant Messaging oder auch eine grafische Netzwerkdarstellung genutzt werden. Eine Expertensuche ermöglicht das Identifizieren von potenziellen Dienstleistungsanbietern. Zum zweiten bieten soziale Netzwerke durch ihre Anwenderzentrierung die Möglichkeit, hybride Wertschöpfung auf Kundenseite weitaus besser zu unterstützen wie etablierte Architekturen. So kann bereits festgestellt werden, dass soziale Netzwerke eine bessere Integration von Kunden in den Erstellungsprozess der Produkte zulässt und somit neue, kundenseitige Kreativitäts- und Innovationspotenziale ausgeschöpft werden können [HEI10]. Abschließend kann festgehalten werden, dass gerade bei steigender Individualisierung und Komplexität der hybriden Wertschöpfung soziale Netzwerke eine beträchtliche Erweiterung der Möglichkeiten darstellt, eine IT-Unterstützung der Beschaffungsprozesse zu unterstützen.

4 Zusammenfassung und Ausblick

Ziel des vorliegenden Beitrags war die Klassifikation von Informationssystemarchitekturen im Kontext der Modellierung strategischer Liefernetze für hybride Wertschöpfung. Auf Basis einer Literaturstudie wurden hierzu vier verschiedene Informationssystemarchitekturen untersucht, inwieweit diese geeignet sind, strategische Liefernetze für hybride Wertschöpfung zu modellieren. Hierzu wurde ein Vergleichsrahmen entwickelt, der acht Merkmale mit insgesamt 24 Merkmalsausprägungen umfasst. Auf Basis dieses Vergleichsrahmens wurden Gestaltungsempfehlungen entwickelt, durch welche Informationssystemarchitekturen eine geeignete Unterstützung strategischer Netzwerkmodellierung für hybride Wertschöpfung erfolgen kann.

Trotz der Breite der untersuchten Architekturen kann festgehalten werden, dass die Unterstützung strategischer Netzwerkmodellierung für hybride Wertschöpfung durch Informationssystemarchitekturen unzureichend ist. Dementsprechend liefert dieser Beitrag erstmalig eine umfassende Bewertung aktueller Informationssystemarchitekturen und zeigt Potenziale auf, wie durch Integration unterschiedlicher Informationssystemarchitekturen die Unterstützung hybrider Wertschöpfung in einem strategischen Liefernetz verbessert werden kann. Der entwickelte Vergleichsrahmen bietet sowohl spezifische Merkmale für strategische Liefernetze wie auch für hybride Wertschöpfung und lässt sich leicht durch unternehmensspezifische Merkmale erweitern. Der Vergleichsrahmen kann daher für Unternehmen, die eine bessere IT-Unterstützung für hybride Wertschöpfung realisieren wollen, eine Entscheidungsunterstützung für die Auswahl der passenden Informationssystemarchitektur darstellen.

Dieser Beitrag stellt einen ersten Schritt in Richtung einer Typologisierung von strategischen Liefernetzen für hybride Wertschöpfung dar. Aufbauend auf den Ergebnissen dieses Beitrags liegt weiterer Forschungsbedarf in der Identifikation architekturübergreifender Aspekte in der Modellierung strategischer Liefernetze für hybride Wertschöpfung, vor allem im Kontext der Integration sozialer Netzwerke. Durch die Integration solcher Technologien scheint sich ein weites Spektrum an Optimierungspotenzial für die Gestaltung von strategischen Liefernetzen zu öffnen. Weiterer Forschungsbedarf besteht in der praktischen Validierung des Vergleichsrahmens. Hier bietet sich eine Anwenderbefragung an, um den bestehenden Vergleichsrahmen zu validieren sowie mögliche Erweiterungen zu erkennen.

Literaturverzeichnis

- [ALB06] Albani, A. & Zaha, J. M. A Reference Model for Strategic Supply Network Development. in Reference Modeling for Business Systems Analysis, edited by P. Fettke & P. Loos (Idea Group Pub. Hershey PA, 2006), pp. 217-240.
- [ALB07] Albani, A. Müssigmann, N. & Zaha, J. M. Reference model for the domain of strategic supply network development. in Reference modeling for business systems analysis, edited by P. Fettke & P. Loos (Idea Group Publ. Hershey PA, 2007), pp. 217-240.

- [BAC01] Backhaus, K. & Kleinkamp, C. Marketing von investiven Dienstleistungen, edited by M. Bruhn & H. Meffert (Gabler, Wiesbaden, 2001), Vol. 2, pp. 73–101.
- [BÖH06] Böhmann, T. & Krcmar, H. Komplexitätsmanagement als Herausforderung hybrider Wertschöpfung im Netzwerk, edited by F. Wojda & A. Barth (DUV, Wiesbaden, 2006), pp. 81–106.
- [BUR02] Burr, W. Service Engineering bei technischen Dienstleistungen: eine ökonomische Analyse der Modularisierung (DUV, Wiesbaden, 2002).
- [BUR07] Burianek, F., Ihl, C., Bonnemeier, S. & Reichwald, R. Typologisierung hybrider Produkte. Ein Ansatz basierend auf der Komplexität der Leistungserbringung (TUM Lehrstuhl für Betriebswirtschaftslehre - Information Organisation u. Management, München, 2007).
- [CAR99a] Carr, A. S. & Pearson, J. N. Strategically managed buyer-supplier relationships and performance outcomes, *Journal of Operations Management* 17, 497–519 (1999).
- [CAR99b] Carr, A. S. & Smeltzer, L. R. The relationship of strategic purchasing to supply chain management, *European Journal of Purchasing and Supply Management* 5, 43–51 (1999).
- [CHE04] Chen, I. J. & Paulraj, A. Towards a theory of supply chain management: the constructs and measurements, *Journal of Operations Management* 22, 119–150 (2004).
- [DAV85] Davis, G. & Olson, M. H. Management information systems: conceptual foundations, structure, and development (McGraw-Hill, New York, 1985).
- [ENG71] Engelen, G. Der Begriff der Klassifikation (Buske, Hamburg, 1971).
- [FRE02] Fremantle, P., Weerawarana, S. & Khalaf, R. Enterprise Services, *Communications of the ACM* 45, 77–82 (2002).
- [GRO99] Grote, B., Rose, T. & Peter, G. Filter and broker. An integrated architecture for information mediation of dynamic sources (FAW, Ulm, 1999).
- [HAW95] Hawryszkiewycz, I. & Rose, T. Notification Agents for Maintaining Awareness. in Proceedings of the Second Int. Conference on Concurrent Engineering: Research and Applications (McLean, Va. 1995), pp. 305–314.
- [HEI10] Heidemann, J. Online Social Networks: Ein sozialer und technischer Überblick, Informatik-Spektrum (2010).
- [HEI90] Heinrich, L. J. Der Prozess der Systemplanung und -entwicklung. in Handbuch Wirtschaftsinformatik, edited by K. Kurbel & H. Strunz (Schäffer-Poeschel; Poeschel, Stuttgart, 1990), pp. 199–214.
- [JOH03] Johansson, J. E., Krishnamurthy, C. & Schlissberg, H. E. Solving the solutions problem, *McKinsey Quarterly*, 116–125 (2003).
- [KAU02] Kaufmann, L. Purchasing and Supply Management: A Conceptual Framework, edited by L. Kaufmann (Hahn, Wiesbaden, 2002), Vol. 2, pp. 3–33.
- [KER06] Kersten, W., Zink, T. & Kern, E.-M. Wertschöpfungsnetzwerke zur Entwicklung und Produktion hybrider Produkte: Ansatzpunkte und Forschungsbedarf, edited by T. Blecker & H. G. Gemünden (Erich Schmidt Verlag, Berlin, 2006), pp. 189–202.
- [KNO72] Knoblich, H. Die typologische Methode in der Betriebswirtschaftslehre, *Wirtschaftswissenschaftliches Studium* 1, 141–147 (1972).
- [KNU01] Knublauch, H. & Rose, T. Werkzeugunterstützte Prozessanalyse zur Identifikation von Anwendungsszenarien für Agenten. in Verteilte Informationssysteme auf der Grundlage von Objekten, Komponenten und Agenten, edited by S. Jablonski, et al. (2001), pp. 37–53.
- [KOC07] Koch, M., Richter, A. & Schlosser, A. Produkte zum IT-gestützten Social Networking in Unternehmen, *WIRTSCHAFTSINFORMATIK* 49, 448–455 (2007).
- [KOP00] Kopanaki, E., Smithson, S., Kanellis, P. & Martakos, D. The Impact of Interorganizational Information Systems on the Flexibility of Organizations. in Proceedings of the Sixth Americas Conference on Information Systems (AMCIS), edited by M. H. Chung (Long Beach, CA, 2000), p. 434.

- [KRA08] Krafzig, D. Banke, K. & Slama, D. Enterprise SOA. Service-oriented architecture best practices. 7th ed. (Prentice-Hall, Upper Saddle River, NJ, 2008).
- [KRC90] Krcmar, H. Bedeutung und Ziele von Informationssystem-Architekturen, WIRTSCHAFTSINFORMATIK 32, 395–402 (1990).
- [LAM00] Lambert, D. M. & Cooper, M. C. Issues in supply chain management, Industrial Marketing Management 29, 65–83 (2000).
- [LAM10] Lamparter, S. Becher, S. & Pirker, M. A Generic Strategy Framework for Policy-directed Autonomous Trading Agents. in Multikonferenz Wirtschaftsinformatik 2010, edited by M. Schumann, L. M. Kolbe, M. H. Breitner & A. Frerichs (Universitätsverlag Göttingen, Göttingen, 2010), pp. 783–795.
- [LEI08] Leimeister, J. M. & Glauner, C. Hybride Produkte - Einordnung und Herausforderung für die Wirtschaftsinformatik, WIRTSCHAFTSINFORMATIK, 248–251 (2008).
- [MCI98] McIvor, R. Humphreys, P. & McAleer, E. The evolution of the purchasing function, Journal of Strategic Change 6, 165–179 (1998).
- [MEI05] Meier, H. Uhlmann, E. & Kortmann, D. Hybride Leistungsbündel, Werkstatttechnik online 95, 528–532 (2005).
- [MÜL97] Müller, G. Kohl, U. & Schoder, D. Unternehmenskommunikation: Telematiksysteme für vernetzte Unternehmen (Addison-Wesley-Longman, Bonn, 1997).
- [REI01] Reiss, M. & Präuer, A. Solutions Providing: Was ist Vision-was Wirklichkeit, Absatzwirtschaft 44, 48–53 (2001).
- [RIC07] Richter, A. & Koch, M. Social Software - Status quo und Zukunft, Bericht der Fakultät für Informatik (2007).
- [ROS00] Rose, T. Knublauch, H. & Peinl, G. Agenten in der pro-aktiven Prozessunterstützung, Industrie Management, 50–53 (2000).
- [SAW06] Sawhney, M. Wolcott, R. & I. Arroniz, I. The 12 Different Ways For Companies To Innovate, MIT Sloan Management Review 47, 75–81 (2006).
- [SCH04] Schmid, B. F. Elektronische Märkte. in Business Engineering - Die ersten 15 Jahre, edited by H. Österle, A. Back, R. Winter & W. Brenner (Springer Berlin2004), pp. 171–202.
- [SCH05] Schild, U. Lebenszyklusrechnung und lebenszyklusbezogenes Zielkostenmanagement (Gabler, Wiesbaden, 2005).
- [SCH06] Schmidt, J. Social Software: Onlinegestütztes Informations-, Identitäts- und Beziehungsmanagement, Forschungsjournal Neue Soziale Bewegungen, 37–46 (2006).
- [SCH10] Schrödl, H. Gugel, P. & Turowski, K. Modellierung strategischer Liefernetze für hybride Leistungsbündel. in Diskussionsbeiträge des 2. Workshops Dienstleistungsmodellierung, edited by O. Thomas & M. Nüttgens (Klagenfurt, 2010), pp. 1–18.
- [SCH93] Schmid, B. Elektronische Märkte, WIRTSCHAFTSINFORMATIK 35, 465–480 (1993).
- [SIX05], Sixtus, M. W wie Wiki (2005), 2009.
- [STE06] Stemmann, B. Dissertation. Universität Marburg, 2006.
- [THO08] Thomas, O. Walter, P. & Loos, P. Product-Service Systems: Konstruktion und Anwendung einer Entwicklungsmethodik, WIRTSCHAFTSINFORMATIK, 208–219 (2008).
- [THO09] Thomas, O. Leyking, K. & Scheid, M. Vorgehensmodelle zur Entwicklung serviceorientierter Softwaresysteme, 9. Internationale Tagung Wirtschaftsinformatik, 181–189 (2009).
- [TUR02a] Turowski, K. Vereinheitlichte Spezifikation von Fachkomponenten (Gesellschaft für Informatik, Arbeitskreis 5.10.3, Augsburg, 2002).
- [TUR02b] Turowski, K. Agent-based e-commerce in case of mass customization, International Journal of Production Economics 75, 69–81 (2002).
- [VDM03], VDMA-Einheitsblatt. Produktbegleitende Dienstleistungen für Kunststoff- und Gummimaschinen (Beuth, 2003).

- [WIN07] Winter, R. & Fischer, R. Essential Layers, Artifacts and Dependendies Of Enterprise Architecture, *Journal of Enterprise Architecture*, 7–18 (2007).
- [WOO99] M. J. Wooldridge. Intelligent Agents, edited by G. Weiss (MIT Press1999).

Potentiale serviceorientierter Architekturen für Software-Werkzeuge des Geschäftsprozessmanagements

Constantin Houy, Markus Reiter, Peter Fettke, Peter Loos

Institut für Wirtschaftsinformatik (IWi)
im Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI) GmbH
Stuhlsatzenhausweg 3, Geb. D3₂
66123 Saarbrücken
{Constantin.Houy | Markus.Reiter | Peter.Fettke | Peter.Loos}@iwi.dfki.de

Abstract: Serviceorientierte Architekturen (SOA) bieten durch lose Kopplung und flexible Wiederverwendbarkeit sowie die Integration von Services erhebliche Potentiale für eine effektive und effiziente Gestaltung betrieblicher Informationssysteme. Diese Grundprinzipien von SOA ermöglichen innovative Software-distributionsmodelle, z. B. das Software-as-a-Service-Konzept (SaaS). Werden das SaaS-Konzept und SOA zwar im Kontext gängiger Unternehmenssoftware für das Enterprise Resource Planning (ERP), das Customer Relationship Management (CRM) oder das Supply Chain Management (SCM) bereits intensiver diskutiert, so sind beide Ansätze im Bereich aktueller Software-Werkzeuge für das Geschäftsprozessmanagement jedoch bisher von geringer Bedeutung. Im vorliegenden Beitrag werden die Potentiale der Serviceorientierung für GPM-Werkzeuge beschrieben und ein erster Vorschlag für ein Rahmenkonzept einer serviceorientierten Dienstesplattform konzipiert, die Software-Services für das GPM zusammenfasst. Darüber hinaus werden erste Überlegungen und Vorschläge zur Implementierung einer solchen Dienstesplattform vorgestellt, die momentan im Kontext des Konsortialforschungsprojektes PROWIT von mehreren Forschungs- und Industriepartnern vorangetrieben wird (Work-in-Progress). Es zeigt sich, dass SOA erhebliche Nutzenpotentiale für GPM-Software-Werkzeuge und innovative Handlungsoptionen für das Geschäftsprozessmanagement der Zukunft bieten können.

1 Einleitung

1.1 Ausgangssituation

Serviceorientierte Architekturen (SOA) stellen einen innovativen Ansatz zur effektiven und effizienten Gestaltung von Informationssystemen dar. Durch lose Kopplung und flexible Integration von Services bieten SOA erhebliche Potentiale für einen verteilten Betrieb und eine flexible Vernetzung verteilter Systemkomponenten [Si07]. Insbesondere ermöglichen solche Architekturen die Vernetzung verteilter, autonomer Dienste zur Gestaltung plattformunabhängiger betrieblicher und überbetrieblich einsetzbarer Anwendungssysteme [EW08]. Obwohl komponentenorientierte Architekturen seit mehr als vier Jahrzehnten im Bereich des Software Engineering bekannt und erforscht werden [Si08], stellt der serviceorientierte Ansatz nach wie vor einen aktuellen Trend der technischen Gestaltung von Informationssystemen dar [Of08].

Das Geschäftsprozessmanagement (GPM) hat in den letzten Jahren erheblich an Bedeutung gewonnen und stellt mit seinen Methoden und Konzepten für den Entwurf, die Konfiguration, Verwaltung und Analyse von Geschäftsprozessen [We07] einen etablierten Ansatz für eine informationstechnische Realisierung fachlicher Anforderungen in der Unternehmenspraxis dar [Ko09]. Dementsprechend kommen in Unternehmen zunehmend Werkzeuge und Systeme für die Modellierung und das Management von Geschäftsprozessen zum Einsatz [Fe09]. Aufgrund der ausgeprägten Dynamik heutiger betriebswirtschaftlicher Handlungssysteme, in denen zunehmend mobile Akteure in unternehmensübergreifenden Geschäftsprozessen agieren, sind allerdings häufig Anpassungen von Prozessen und Organisationsstrukturen an neu entstehende Rahmenbedingungen notwendig. Diese Anpassungen sind sowohl im Rahmen der kontinuierlichen Prozessverbesserung als auch im Kontext von Ausnahmebehandlungen in Geschäftsprozessen erforderlich. Letzteres kann sich ergeben, da in der Entwurfsphase von Prozessmodellen nicht sämtliche möglichen Ereignisse vorhergesehen werden können, die während ihrer Ausführung eintreten und diese Ausführung beeinflussen können.

Um den Anforderungen an eine flexible Gestaltung von Geschäftsprozessen gerecht zu werden, sind insbesondere Software-Werkzeuge für das GPM notwendig, die die in der jeweiligen Situation erforderlichen Funktionalitäten zur Verfügung stellen können. In diesem Zusammenhang bieten SOA einen innovativen Architekturansatz für GPM-Software-Werkzeuge. Ad hoc benötigte Services, wie z. B ein Service für die Transformation eines Geschäftsprozessmodells in eine andere Modellierungssprache, können bei einem solchen Architekturansatz nach Bedarf in das GPM-Werkzeug integriert werden. Trotz der erheblichen Potentiale von SOA für eine flexible Gestaltung von GPM-Werkzeugen zeigt eine aktuelle Marktstudie, dass diese nur bedingt SOA mit lose gekoppelten und dynamisch zur Laufzeit gebundenen GPM-Diensten unterstützen [KD08]. Darüber hinaus wird deutlich, dass GPM-Werkzeuge für den Entwurf, die Ausführung und das Controlling von Geschäftsprozessen nach wie vor hauptsächlich in Form von Client-Server-basierten Software-Produkten und kaum als Dienstleistungen im Sinne des internetbasierten Distributionskonzeptes Software-as-a-Service (SaaS) angeboten werden.

1.2 Ziele des Beitrags

Vor diesem Hintergrund zielt der vorliegende Beitrag darauf ab, die Potentiale der Serviceorientierung für GPM-Werkzeuge konzeptionell aufzuarbeiten und ein Rahmenkonzept für eine serviceorientierte Dienstesplattform zu entwickeln. Im Einzelnen lassen sich folgende Teilziele des Beitrags formulieren:

1. Darstellung von Potentialen der Serviceorientierung für GPM-Werkzeuge,
2. ein erster Vorschlag für ein Rahmenkonzept einer serviceorientierten Dienstesplattform für das GPM und
3. Illustration einer möglichen Umsetzung des Rahmenkonzeptes am Beispiel eines potentiellen Anwendungsszenarios aus dem aktuell vom BMBF geförderten Konsortialforschungsprojekt „Prozessorientierter Web-2.0-basierter integrierter Telekommunikationsservice (PROWIT)“, an dem sowohl Forschungs- als auch Industriepartner beteiligt sind.

1.3 Aufbau des Beitrags

Nach dieser Einleitung beschreibt Kapitel 2 die Serviceorientierung als zukunftsweisen-den Gestaltungsansatz sowohl aus softwaretechnischer als auch aus betriebswirtschaftlicher Perspektive. Kapitel 3 zeigt die bisherige Entwicklung von Software-Werkzeugen für das Geschäftsprozessmanagement auf, bevor in Kapitel 4 Nutzenpotentiale von SOA für GPM-Software-Werkzeuge herausgestellt werden. Kapitel 5 erarbeitet den Vorschlag für ein Rahmenkonzept der GPM-Dienstesplattform, während in Kapitel 6 erste Ideen für die prototypische Implementierung des Rahmenkonzeptes und seine zukünftige Anwendung anhand eines überbetrieblichen Anwendungsszenarios behandelt werden, das sich mit dem Geschäftsprozessmanagement im Rahmen des technischen Kundendienstes für komplexe Investitionsgüter auseinandersetzt. Kapitel 7 resümiert die Inhalte und bietet einen Ausblick.

2 Serviceorientierung als innovativer Gestaltungsansatz

2.1 Serviceorientierung aus softwaretechnischer Perspektive

Die Serviceorientierung stellt ein Paradigma der Softwareentwicklung dar, das die Wiederverwendbarkeit und eine flexible Konfigurierbarkeit lose gekoppelter und zur Laufzeit gebundener Softwarekomponenten fördert [Ac02]. Dadurch kann sowohl die Entwicklung als auch der Einsatz von Informationssystemen effektiver und effizienter gestaltet werden [Or00].

Entwickelt hat sich die Serviceorientierung vor dem Hintergrund der ständig an Komplexität zunehmenden Anforderungen an Entwicklungstechniken für Informationssysteme [Do05]. Zu Beginn der Entwicklung von Computerprogrammen wurden diese noch in maschinennahen Assemblersprachen verfasst. Mit zunehmender Leistungsfähigkeit der Rechner haben sich darauffolgend die ersten Hochsprachen entwickelt, die eine Codierung von Algorithmen in einer der menschlichen Sprache näheren Form ermöglichen. Die Softwareentwicklung mit imperativen Hochsprachen führte aufgrund der sequenziellen Codierung bei zunehmend komplexen Softwareprojekten schnell zu unübersichtlichen Programmstrukturen, denen man mit der Kapselung einzelner Programmfunctionen begegnete. Mit zunehmender Komplexität wurde die Kapselung immer größerer Einheiten von Daten und Funktionen erforderlich, was zunächst zur objektorientierten Programmierung und auf einem höheren Abstraktionsniveau zur Komponentenorientierung führte. Unter Berücksichtigung technologieübergreifender Interoperabilität sowie durch die Kapselung einer Vielzahl von Klassen und Daten in Softwarekomponenten können sinnvoll verteilbare sowie effektiv wieder verwendbare Komponenten zur Verfügung gestellt werden, die sich auch als einzelne Softwarebausteine bzw. Services vermarkten lassen [Ac02]. Komponentenorientierte Architekturen, bei denen lose gekoppelte und zur Laufzeit gebundene Software-Services im Vordergrund stehen, werden häufig als SOA bezeichnet [He07]. Im Folgenden wird der Service-Begriff aus Sicht der Betriebswirtschaft geschildert.

2.2 Serviceorientierung aus betriebswirtschaftlicher Perspektive

Services oder Dienstleistungen im betriebswirtschaftlichen Sinne stellen vorwiegend immaterielle, selbstständige Leistungen dar, die nutzenstiftende Wirkungen an Menschen oder an Objekten erzielen sollen [MB09, S. 19]. Diese Services können sowohl Leistungen für Endkunden als auch einen Input im Rahmen eines Leistungserstellungsprozesses eines weiteren Wirtschaftsakteurs darstellen. Dienstleistungen spielen in sämtlichen Industrienationen eine immer größere Rolle. Für Unternehmen haben sie insbesondere deshalb an Bedeutung gewonnen, weil sie ein bedeutendes Merkmal zur Abgrenzung von der Konkurrenz darstellen [RM06]. Mit der ingenieurmäßigen Entwicklung innovativer Services setzen sich das Service Engineering bzw. die Service Science auseinander [CS06].

Im Bereich integrierter Informationssysteme sind innovative Services von besonderer Bedeutung, da in diesem Kontext eine Vereinigung der betriebswirtschaftlichen Anforderungen sowie der softwaretechnischen Möglichkeiten zur Entwicklung innovativer Handlungsoptionen geführt hat. Eine Entwicklung, die daraus resultierte, ist das internetbasierte Softwaredistributionsmodell, das aktuell unter dem Stichwort Software-as-a-Service (SaaS) diskutiert und durch das serviceorientierte Programmierparadigma ermöglicht wird.

2.3 Software-as-a-Service

Software-as-a-Service gilt als ein bedeutender Trend im Bereich der integrierten Informationsverarbeitung. SaaS basiert auf der Idee der On-Demand-Distribution von Software, die seit den späten 1990er-Jahren diskutiert und in Konzepten wie Application Service Providing (ASP), Business Service Providing (BSP) eine praktische Ausprägung gefunden hat [BHB10]. Die Verantwortung für Betrieb und Wartung der Software, die als Dienstleistung über das Internet angeboten wird, übernimmt der Softwareanbieter. Die Nutzung der Software wird dabei nicht über Lizenzgebühren und zusätzlich anfallende Kosten für die Wartung, Updates etc. abgegolten, sondern häufig über monatlich anfallende Gebühren für die Nutzung von Softwarekomponenten und die dazugehörigen Serviceleistungen vergütet [Ma09], so z. B. bei der Customer-Relationship-Management-Lösung (CRM) des Anbieters Salesforce.com oder der Enterprise-Resource-Planning-Lösung (ERP) Business ByDesign der SAP. Verglichen mit klassischer Standardsoftware resultieren aus dieser Eigenschaft aus Anwendersicht Kostenvorteile und eine schnellere Verfügbarkeit der SaaS-Lösung, allerdings auch eine geringere Anpassungsfähigkeit der Software an die eigenen organisatorischen Anforderungen [BLH08]. Der Nutzer einer SaaS-Lösung greift üblicherweise über einen Webbrowser auf die Unternehmenssoftware zu und ist somit in der Lage, die jeweiligen Anwendungen ortsunabhängig und flexibel zu nutzen [Ma09]. Aus Anbietersicht ergeben sich durch Skalen- und Standardisierungseffekte unter anderem günstigere Bereitstellungs-kosten für die Software-Services. Das SaaS-Konzept kann somit sowohl für die Anbieterseite als auch für die Kundenseite erhebliche Potentiale bieten. SaaS ist vor allem im Bereich von Office-Anwendungen, CRM, ERP, SCM (Supply Chain Management) und im Bereich von Anwendungen für Kommunikation, Kooperation und Content Management im Einsatz [BHB10].

3 Software-Werkzeuge für das Geschäftsprozessmanagement

3.1 Terminologische Abgrenzung

Das Geschäftsprozessmanagement (GPM) fasst Methoden und Konzepte zusammen, die dem Entwurf, der Konfiguration, der Verwaltung und Analyse von Geschäftsprozessen dienen [We07]. Geschäftsprozesse stellen „zusammengehörende Abfolge von Unternehmensverrichtungen zum Zweck einer Leistungserstellung“ dar [Sc02, S. 3]. GPM ermöglicht eine zielgerichtete Steuerung von Geschäftsprozessen und unterstützt das Erreichen der strategischen und operativen Ziele eines Unternehmens [SS08]. Es lässt sich als Lebenszyklus mit mehreren Phasen beschreiben. Der in Abbildung 1 visualisierte Lebenszyklus des Geschäftsprozessmanagements fasst essentielle Bestandteile gängiger Vorgehensmodelle in der Geschäftsprozessmanagementliteratur zusammen [HFL10].



Abbildung 1: Lebenszyklus des Geschäftsprozessmanagements

Für das GPM haben sich im Laufe der Zeit unterschiedliche Werkzeugkategorien entwickelt. Zu ihnen zählen unter anderem Werkzeuge für die Prozessmodellierung, die Ausführung von Geschäftsprozessen (Workflow-Management-Systeme) oder das Monitoring von Geschäftsprozessen. Diese Werkzeuge haben sich hinsichtlich mehrerer unterschiedlicher Merkmale weiterentwickelt. Einen bedeutenden Aspekt stellt der Funktionsumfang dieser Werkzeuge dar, der parallel zu den neuen Entwicklungen im Bereich der GPM-Forschung stets gewachsen ist. So unterstützen aktuelle Modellierungswerkzeuge beispielsweise eine Vielzahl unterschiedlicher Modellierungsmethoden, die im Kontext der GPM-Forschung in den letzten Jahren entwickelt wurden, wie z. B. Ereignisgesteuerte Prozessketten (EPK), UML-Aktivitätsdiagramme oder die Business Process Modeling Notation (BPMN). Darüber hinaus bieten die meisten etablierten GPM-Werkzeuge beispielsweise auch eine Reihe geeigneter Schnittstellen für den Datenaustausch mit anderen Werkzeugen an. Diese Schnittstellen ermöglichen den Import und Export relevanter Daten, unter anderem etwa Prozesskostendaten, Objektbibliotheken im XML-Format, ausführbare Prozessmodelle im XML Process Definition Language (XPDL)-Format bzw. der Business Process Execution Language (BPEL) sowie den Export von HTML- oder Grafikformaten [DF08].

3.2 Entwicklungslinien und Trends der Architektur von GPM-Werkzeugen

GPM-Werkzeuge haben sich in den letzten Jahren auch hinsichtlich ihrer Systemarchitektur weiterentwickelt. Zu Beginn der Entwicklung von Werkzeugen für das Geschäftsprozessmanagement – meistens reine Zeichenwerkzeuge – verfügten diese in der Regel über monolithische Systemarchitekturen, bei denen ein einziger Rechner die Rechenleistung für die Präsentation, die Verarbeitung und die Datenhaltung zur Verfügung stellt. Diese monolithischen Systemarchitekturen führten häufig dazu, dass verschiedene Werkzeuge nicht kompatibel zueinander waren. In Ermangelung geeigneter Schnittstellen konnten meistens auch keine Prozessmodelle zwischen unterschiedlichen Systemen ausgetauscht werden.

Mit zunehmender Verbreitung des Client-Server-Modells, das durch die Verteilung der notwendigen Ressourcen in einem Netzwerk einige Potentiale für die Gestaltung von Informationssystemen bietet, etablierte sich dieser Architekturansatz auch bei GPM-Werkzeugen. Gängige GPM-Werkzeuge wie z. B. das ARIS-Toolset, BONAPART etc. verfügen über eine Client-Server-Architektur und ermöglichen die zentrale Pflege der Daten in einem Repository. Abbildung 2 visualisiert in Anlehnung an [RS03, S. 257] den Entwicklungsschritt von einer monolithischen Systemarchitektur hin zu einer Client-Server-Architektur bei GPM-Werkzeugen.

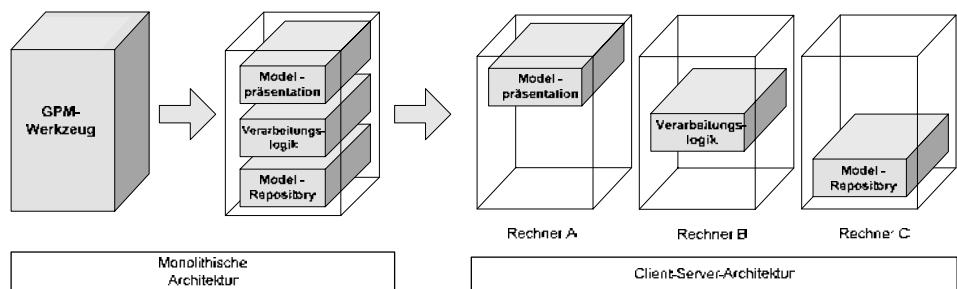


Abbildung 2: Entwicklung von monolithischen zu Client-Server-Architekturen bei GPM-Werkzeugen

Die Fraunhofer Marktstudie [SW08] dokumentiert zwar, dass aktuelle GPM-Werkzeuge das Management von SOA mithilfe aktueller XML-Standards unterstützen. Dabei steht allerdings die Komposition verschiedener betriebswirtschaftlicher Services, z. B. von ERP- oder CRM-Services im Vordergrund und nicht von Services, die das GPM, z. B. die Modellierung oder das Monitoring von Prozessen, unterstützen. Ersteres wird häufig in der Literatur unter dem Leitgedanke „Kombination von SOA und GPM“ diskutiert, z. B. bei [Ma07; Lu08; OHB09], und ist nicht mit dem hier behandelten Thema „SOA für GPM-Werkzeuge“ zu verwechseln.

Die Architektur von GPM-Werkzeugen selbst entwickelt sich bisher nur langsam in Richtung Serviceorientierung [KD08]. Einige der GPM-Werkzeuge, die in der Marktstudie berücksichtigt wurden, verfügen zwar über geeignete Exportformate für den Austausch von Daten, sehen aber bisher nicht den Einbezug anderer Services vor [DH08, S. 164]. Einen ersten Ansatz, der in diese Richtung geht, bietet das Open-Source-Werkzeug YAWL (Yet Another Workflow Language) an [AH05], das ein Einbeziehen externer Services in die Workflow-Ausführung ermöglicht (<http://www.yawlfoundation.org>).

Der aktuelle Trend in Richtung Software/Platform-as-a-Service lässt sich auch zunehmend bei kommerziellen Anbietern von GPM-Werkzeugen feststellen. So bieten beispielsweise die IDS Scheer AG oder Signavio (<http://www.signavio.com>) einen web-basierten Modellierungsdienst („Modelling-as-a-Service“) an, bei dem über einen Webbrowser Geschäftsprozesse modelliert werden können. Es existieren auch erste web-basierte Werkzeuge mit unterschiedlichen Funktionalitäten für ein „GPM-as-a-Service“. So ermöglicht die Service-Plattform „RunMyProcess“ (<http://www.runmyprocess.com>) die Modellierung, Ausführung und das Monitoring von Geschäftsprozessen, und die Social-BPM-Plattform „ARISalign“ (<http://www.arisalign.com>) stellt unterschiedliche GPM-Funktionalitäten unter Einbezug von Anwendungen aus dem Bereich Soziale Netzwerke und Web 2.0 zur Verfügung. Allerdings handelt es sich bei diesen Ansätzen um Implementierungen klassischer Client-Server-Architekturen, bei denen die Präsentationsschicht im Browser läuft, während die Verarbeitungslogik und die Datenhaltung in einem zentralen Rechenzentrum angesiedelt sind.

Es zeigt sich, dass bei den Anbietern die Interoperabilität von Werkzeugen für das Geschäftsprozessmanagement zwar immer stärker in den Vordergrund rückt. Allerdings werden die zahlreichen Potentiale, die SOA für GPM-Werkzeuge anbieten können, bisher nicht ausgeschöpft. Es ist davon auszugehen, dass die Entwicklung der Systemarchitekturen von GPM-Werkzeugen hin zur Serviceorientierung in Zukunft weiterhin an Bedeutung gewinnen wird [Lö07]. In diesem Kontext wird aus Sicht der Systemarchitektur die Verarbeitungslogik in einzelne Software-Services aufgeteilt. Diese können nicht nur auf unterschiedlichen Rechnern in einem Unternehmen, sondern auch unternehmensübergreifend verteilt werden. Abbildung 3 visualisiert im Kontext von GPM-Werkzeugen die Entwicklung von der Client-Server-Architektur zu einer SOA.

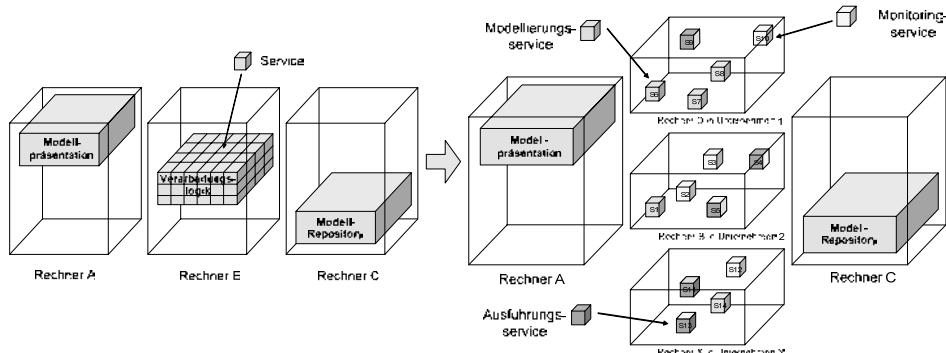


Abbildung 3: Entwicklung von Client-Server-Architekturen zu SOA bei GPM-Werkzeugen

Software- und Serviceanbieter könnten im Zusammenhang mit solchen Systemarchitekturen einzelne spezialisierte GPM-Services für die unterschiedlichen Phasen des GPM-Lebenszyklus anbieten, bzw. Lösungsanbieter spezialisierte Services anderer Anbieter in ihre GPM-Lösung integrieren [BDH08; BLH08]. Die Vision des „GPM-as-a-Service“ bzw. „GPM-in-the-Cloud“ [SK09], bei dem individuelle GPM-Lösungen aus verfügbaren GPM-Services kombiniert werden können, könnte dadurch in den nächsten Jahren umfassender realisiert werden. Im folgenden Kapitel werden die Nutzenpotentiale von SOA und potentielle Services für GPM-Werkzeuge näher untersucht.

4 Nutzenpotentiale von SOA für GPM-Werkzeuge

Die Literatur stellt unterschiedliche Nutzenpotentiale von SOA dar. Es existieren einige Arbeiten, die basierend auf Literaturreviews diese Potentiale zusammenfassen. Häufig genannt werden eine höhere Flexibilität und Agilität von IT-Infrastrukturen und eine verbesserte Unterstützung der Prozessorientierung in Unternehmen [AHB09], sowie eine höhere Produktivität durch verkürzte Entwicklungszeiten [BY10]. Weitere empirische Befunde, z. B. die Fallstudie im GPM-Kontext von Offermann et al. [OHB09], zeigen, dass durch die grundlegenden Prinzipien der Interoperabilität, Integration, Komponierbarkeit und Wiederverwendbarkeit von Services die grundlegenden Ziele des Einsatzes von SOA, nämlich eine höhere Effizienz, Effektivität und Anpassbarkeit von IT-Architekturen im Vergleich zu alternativen Architekturen besser erreicht werden können.

Auf Basis einer SOA können nutzbringende internetbasierte Dienste für das GPM flexibel kombiniert und in einer zentralen Dienstplattform integriert werden. Dadurch wird es möglich, individuelle und erweiterbare GPM-Lösungen für Unternehmen gemäß deren konkreten Anforderungen zu komponieren. Im Vergleich zur Anschaffung standardisierter Lösungen etablierter Anbieter, kann die notwendige Infrastruktur effizienter und effektiver gestaltet werden. Darüber hinaus bieten SOA die Möglichkeit, spezialisierte, branchenspezifische Services unterschiedlicher Anbieter zu kombinieren. Im Folgenden wird eine exemplarische Auswahl potentieller Diensttypen dargestellt, die in kombinierter Form das GPM effektiv unterstützen können:

1. *Transformationsdienste für Geschäftsprozessmodelle.* Solche Dienste ermöglichen es, Prozessmodelle in unterschiedliche Modellierungssprachen zu transformieren, z. B. eine EPK in ein Petrinetz zu übersetzen. Dies ist unter anderem notwendig, um die automatisierte Ausführbarkeit von Prozessmodellen vorzubereiten [VZS05].
2. *Dienste für Modellvergleich.* Ein Modellvergleichsdienst birgt erhebliche Potentiale für die Komplexitätsreduktion von Prozessstrukturen. Darüber hinaus ermöglicht ein solcher Dienst die Identifikation von Strukturanalogien in Prozessmodellen [FL05; Ho10]. Werden beispielsweise analoge Prozesse mit ähnlichem Output identifiziert, die von unterschiedlichen Organisationseinheiten verrichtet werden, so können diese Prozesse effizienter organisiert werden.
3. *Dienste für Process Mining.* Solche Dienste unterstützen die Generierung von Prozessmodellen aus Prozesslogdateien [TTM08]. Die daraus entstandenen Modelle bilden Prozesstypen auf Basis von tatsächlich abgelaufenen Prozessinstanzen ab. Die dadurch entwickelten Prozessmodelle sind insbesondere deshalb von Bedeutung, da sie nicht selten von den idealtypischen, von Methodenexperten konstruierten Prozessmodellen abweichen.
4. *Referenzmodelle und Referenzmodellkataloge.* Der Zugriff auf Referenzprozessmodelle über in die Plattform integrierte Referenzmodellkataloge oder Prozessbibliotheken bietet die Möglichkeit, unkompliziert auf das Wissen aus Best-Practice-Modellen zuzugreifen und dies bei der Modellierung und Anpassung der eigenen Geschäftsprozesse zu berücksichtigen [FL02; FL04].
5. *Dienste für verteilte Modellierung.* Eine verteilte Modellierung erhöht den Konsens darüber, ob ein Prozessmodell die tatsächlich ablaufenden Prozessinstanzen reprä-

sentiert. Dienste für eine verteilte Modellierung müssen dabei das Anlegen, das gleichzeitige Bearbeiten sowie die Freigabe und Publikation kooperativ bearbeiteter Modelle unterstützen. Ferner regeln sie den Zugriff auf die Prozessmodelle [Do09].

6. *Dienste für Reifegradmodelle.* Im Rahmen der Prozessstrategieplanung bieten Reifegradmodelle ein vielseitig einsetzbares Mittel für die strategische Steuerung des Geschäftsprozessmanagements. In diesem Kontext existieren unterschiedliche Reifegradmodellkonzepte, deren Umsetzung in einer GPM-Diensteplattform die Steuerung des gesamten GPM-Lebenszyklus unterstützen kann.
7. *Dienste für die Ausführung von Geschäftsprozessen.* Sogenannte Workflow-Engines stellen Dienste für eine Ausführung definierter Geschäftsprozessmodelle unter Berücksichtigung der festgelegten Organisationseinheiten zur Verfügung. Eine wichtige Bedingung ist, dass die Geschäftsprozessmodelle in der unterstützten Modellierungssprache vorliegen müssen.
8. *Dienste für Modellevaluation.* Evaluationsdienste dienen der Prüfung der syntaktischen Fehlerfreiheit von Prozessmodellen. Die Vermeidung von Verklemmungen („deadlocks“) unterstützt eine automatisierte Ausführbarkeit von Prozessen [DJ05].
9. *Dienste für semantische Annotation von Prozessmodellen.* Die semantische Annotation von einzelnen Elementen in Prozessmodellen auf Basis von Ontologien bietet erhebliche Möglichkeiten, die maschinelle Ausführbarkeit und Analyse sowie die Kommunikation Beteiligter über Prozessmodelle zu unterstützen [TF09]. Die dazu notwendigen Dienste, wie Basis- oder Domänenontologien, z. B. die lexikalische Datenbank WordNet (<http://wordnet.princeton.edu>), werden im Netz angeboten und können ebenso in eine Diensteplattform für das GPM integriert werden.
10. *Dienste für das Monitoring von Geschäftsprozessen.* Solche Dienste unterstützen eine Prüfung der Performance von Prozessen zur Echtzeit auf Basis der Daten, die von den Diensten zur Ausführung von einzelnen Prozessinstanzen geliefert werden. Die Einschätzung der jeweiligen Performance basiert auf einem Vergleich der vor der Ausführung definierten Kennzahlen mit den tatsächlich entstandenen Daten.
11. *Dienste für grafische Aufbereitung von Prozesskennzahlen.* Für das Controlling des Geschäftsprozessmanagements sind die bei der Ausführung von Prozessinstanzen erreichten Kennzahlen ein bedeutendes Hilfsmittel. Werkzeuge, die eine grafische Darstellung des aktuellen Standes und der Entwicklung dieser Kennzahlen unterstützen, ermöglichen anschauliche Übersichten für das Controlling des Geschäftsprozessmanagements. Einen solchen Dienst stellt z. B. die ARIS Mashzone (<http://www.mashzone.com>) zur Verfügung.
12. *Dienste für prozessorientierte Kommunikation, Koordination und Kooperation auf Basis von Web 2.0.* Web-2.0-basierte Kommunikationsdienste, wie z. B. Wikis, Blogs, Soziale Netzwerke, Social Tagging etc. können eingesetzt werden, um prozessrelevantes Wissen effektiv und effizient innerhalb einer Community von Personen, die an einem Prozess beteiligt sind (Prozess-Community), zu verteilen [VFL10]. Durch Web-2.0-Anwendungen kann Prozesswissen sowohl für die Ad-hoc-Behandlung von Ausnahmen in Geschäftsprozessen als auch im Kontext der kontinuierlichen Prozessverbesserung effektiv kommuniziert und dokumentiert werden. Beispielsweise ermöglichen Soziale Netzwerke die Bildung von Prozess-

Communities, in denen auf effektive und effiziente Art und Weise prozessbezogene sowie personenbezogene Informationen (z. B. Ausnahmenzustände in einem Prozess, Wechsel von Verantwortlichkeiten, Wissensprofile von Mitarbeitern) unter Berücksichtigung unterschiedlicher Rollen kommuniziert werden können.

13. *Sensordienste für das GPM.* Die Berücksichtigung von Sensorinformationen bietet besondere Potentiale für das GPM. Sensordienste können sowohl den aktuellen Status eines Geschäftsprozesses als auch den aktuellen Kontext von Prozessbeteiligten erheben. Dadurch entstehen verschiedene Möglichkeiten der Automatisierung von Kommunikations- und Steuerungsprozessen im GPM. In diesem Zusammenhang sind sowohl Hardware-Sensoren (z. B. GPS zur Standortermittlung von Personen oder Betriebsmitteln) als auch Software-Sensoren (z. B. Terminkalender oder der Status in einem Sozialen Netzwerk) von Bedeutung, da diese bedeutende Informationen für die Entscheidungsfundierung im GPM liefern können.
14. *Telekommunikationsdienste für das GPM.* Internetbasierte Kommunikationsdienste wie E-Mail, Video-Conferencing, Instant Messaging, Internet-Telefonie per Skype oder andere Telekommunikationsdienste wie Mobiltelefonie oder SMS werden verstärkt in die Unternehmenskommunikation eingebunden und haben sich teilweise bereits etabliert. Die Integration von Telekommunikationsdiensten in eine GPM-Dienstplattform bietet vor allem in Kombination mit den im vorangegangenen Abschnitt beschriebenen Sensordiensten erhebliche Potentiale für das GPM. Erhobene Kontextinformation wie z. B. der Standort einer Person ermöglicht die automatisierte Auswahl des optimalen Kommunikationsmediums in Abhängigkeit der Dringlichkeit und der zu kommunizierenden Inhalte. Wird beispielsweise ad hoc eine Vertriebsprozessinstanz vom Leitstand angepasst, so kann ein Außendienstmitarbeiter, der zur Prozessbearbeitung gerade unterwegs ist, darüber am besten über einen Mobiltelefondienst informiert werden. Sind hingegen umfassende Prozessänderungen für zukünftige Prozessinstanzen zu kommunizieren und befindet sich der Mitarbeiter an seinem Arbeitsplatz, so werden die Änderungen effektiver und effizienter über einen asynchronen Dienst, z. B. per E-Mail, mitgeteilt.

Diese Darstellung zeigt, dass eine Reihe bedeutender Dienste existieren, die in aktuellen GPM-Werkzeugen nur teilweise in integrierter Form vorliegen. Darüber hinaus wird deutlich, dass SOA erhebliche Potentiale für die Integration dieser Dienste, die bisher als spezialisierte Einzellösungen zur Verfügung stehen, in eine Plattform für das Geschäftsprozessmanagement bieten. Im Folgenden wird ein Vorschlag für ein Rahmenkonzept einer SOA für ein integriertes GPM-Werkzeug dargestellt.

5 Rahmenkonzept einer SOA für eine GPM-Dienstplattform

Vor dem Hintergrund der präsentierten Potentialen unterteilt sich das Rahmenkonzept für serviceorientierte GPM-Werkzeuge in mehrere Funktionsblöcke, die jeweils Services für das Geschäftsprozessmanagement, für die Kommunikation der Prozessbeteiligten, für die Erhebung von Kontextinformationen über Sensoren und für die Kommunikation der einzelnen Komponenten („Middleware“) zusammenfassen. Der Architekturvorschlag und seine lose gekoppelten Funktionsblöcke werden in Abbildung 4 visualisiert.

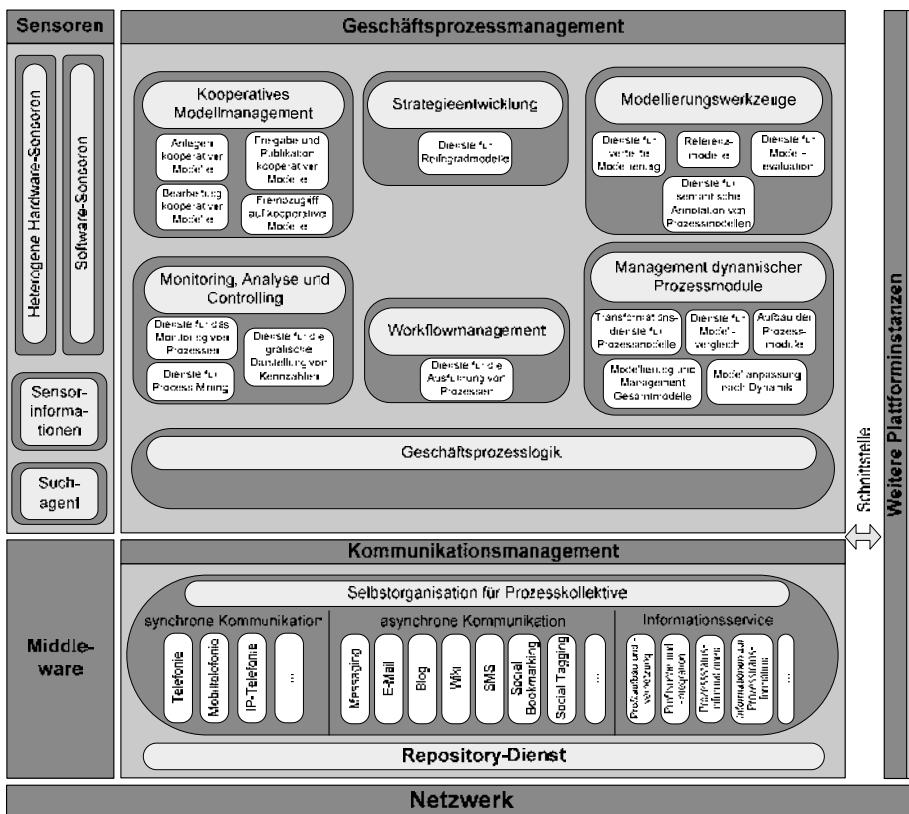


Abbildung 4: Serviceorientierte Architektur der GPM-Diensteplattform

Das Rahmenkonzept berücksichtigt unterschiedliche Typen von Sensordiensten für das Geschäftsprozessmanagement, die auf Hard- und Software-Sensoren aufsetzen. Eine Vielfalt physikalischer Ereignisse und Umgebungsparameter, die im GPM potentiell von Bedeutung sind, können durch Hardware-Sensoren erfasst werden. Die Beschaffenheit und Komplexität solcher Hardware-Sensoren und der möglichen Auswertungslogik kann stark variieren. Daraus resultieren heterogene Sensornetzwerke, die Informationen unterschiedlicher Qualität und Aussagekraft ermöglichen. Diese heterogene Sensorlandschaft umfasst sowohl Kleinsensoren mit begrenzter Rechen- und Kommunikationsleistung zur Erfüllung einfacherer Aufgaben als auch komplexere Sensoren.

Aufgrund der typischerweise starken Verteilung der Hardware-Sensoren und deren heterogener Beschaffenheit ist eine Middleware notwendig, die die Sensorinformationen vereinheitlicht und dann an die sogenannte Geschäftsprozesslogik weitergeben kann. Die Middleware steuert im vorgeschlagenen Rahmenkonzept die Registrierung und Verwaltung der Sensoren, und bietet die Möglichkeit zur gezielten Suche nach Informationen. Sensoren können sich dabei direkt beim Suchagenten registrieren. Von der Middleware ausgehende Suchanfragen lösen die Abfrage der Teilmenge von Sensoren aus, welche die benötigten Daten liefern kann. Diese Architektur ermöglicht dadurch eine nahtlose Integration von Sensoren und GPM-Diensten.

Software-Sensoren, die in die Geschäftsprozesslogik integriert sind (z. B Terminkalender, Statusinformationen aus einem Sozialen Netzwerk etc.), können weitere bedeutende Kontextinformationen für das GPM liefern und über die Middleware adressiert werden. Die Kontextinformationen, die über die Geschäftsprozesslogik ermittelt werden, ermöglichen eine der Prozesssituation angemessene und automatisierbare Entscheidung über ein geeignetes Kommunikations- und Kooperationsmedium für die Behandlung von Ausnahmenfällen. Darüber hinaus umfasst die Geschäftsprozesslogik Komponenten zur Überwachung von definierten Geschäftsprozessen. Ein Informationsservice kann so zu jeder Zeit den aktuellen Status eines Prozesses für den Nutzer aufbereiten. Zudem kann in Ausnahmezuständen eine Anfrage zur Kommunikation an die entsprechenden Teilsysteme, die der Selbstorganisation des Prozesskollektivs dienen (synchrone und asynchrone Telekommunikations- bzw. Web-2.0-Dienste), gestellt werden.

6 Zur prototypischen Implementierung und Anwendung

Im Rahmen des BMBF geförderten Konsortialforschungsprojektes „Prozessorientierter Web-2.0-basierter integrierter Telekommunikationsservice (PROWIT)“, das mehrere Forschungs- und Industriepartner vorantreiben, wird das vorgeschlagene Rahmenkonzept für die GPM-Diensteplattform momentan prototypisch auf Basis der Open-Source-Portal-Lösung Liferay (<http://www.liferay.com>) implementiert. Liferay ermöglicht die individuelle und flexible Konfiguration der Benutzeroberfläche auf Basis sogenannter Portlets, die Container für einen oder mehrere GPM-Services darstellen [Ma07]. Dies fördert die Anpassbarkeit und Erweiterbarkeit der GPM-Diensteplattform.

Im Zentrum des Projektes „PROWIT“ steht die flexible Integration von Telekommunikations- und Web-2.0-Diensten in das Geschäftsprozessmanagement unter besonderer Berücksichtigung innovativer Sensornetzwerke. Vor diesem Hintergrund werden im Folgenden bedeutende Potentiale der zukünftigen Anwendung der angestrebten GPM-Diensteplattform im Kontext eines potentiellen Einsatzszenarios erläutert, das Prozesse des technischen Kundendienstes für komplexe Investitionsgüter fokussiert.

Während komplexe Investitionsgüter wie z. B. Stapler-, Regalsysteme oder Kompressoranlagen vor einigen Jahren in der Regel noch komplett an einen unternehmerisch tätigen Kunden verkauft wurden, setzen sich heutzutage immer stärker dienstleistungsorientierte Betreibermodelle durch, bei denen die Leistung, die ein komplexes Investitionsgut erzeugt, als Dienstleistung verkauft wird (hybrides Leistungsbündel). In diesem Fall kümmert sich der Leistungsanbieter darum, dass das Investitionsgut eine vereinbarte Leistung erbringt. Das Investitionsgut stellt in einem solchen Szenario den Schnittpunkt zweier verschiedener Geschäftsprozessstränge dar. Einerseits ist es in den Produktionsprozess des Kunden eingebettet, in dem es Dienstleistungen erbringt. Andererseits stellt es das Kernobjekt des Betriebs- und Wartungsprozesses des Betreibers dar. Im Wartungsfall ist deshalb ein hohes Maß an geschäftsprozessbezogener Kommunikation notwendig, die durch die skizzierte GPM-Diensteplattform effektiv unterstützt werden kann. Im Rahmen der Wartung liegen häufig Ausnahmesituationen vor, deren Behebung fallabhängig zu kommunizieren ist. Abbildung 5 visualisiert exemplarisch einen solchen Prozess, der im Folgenden näher beschrieben wird.

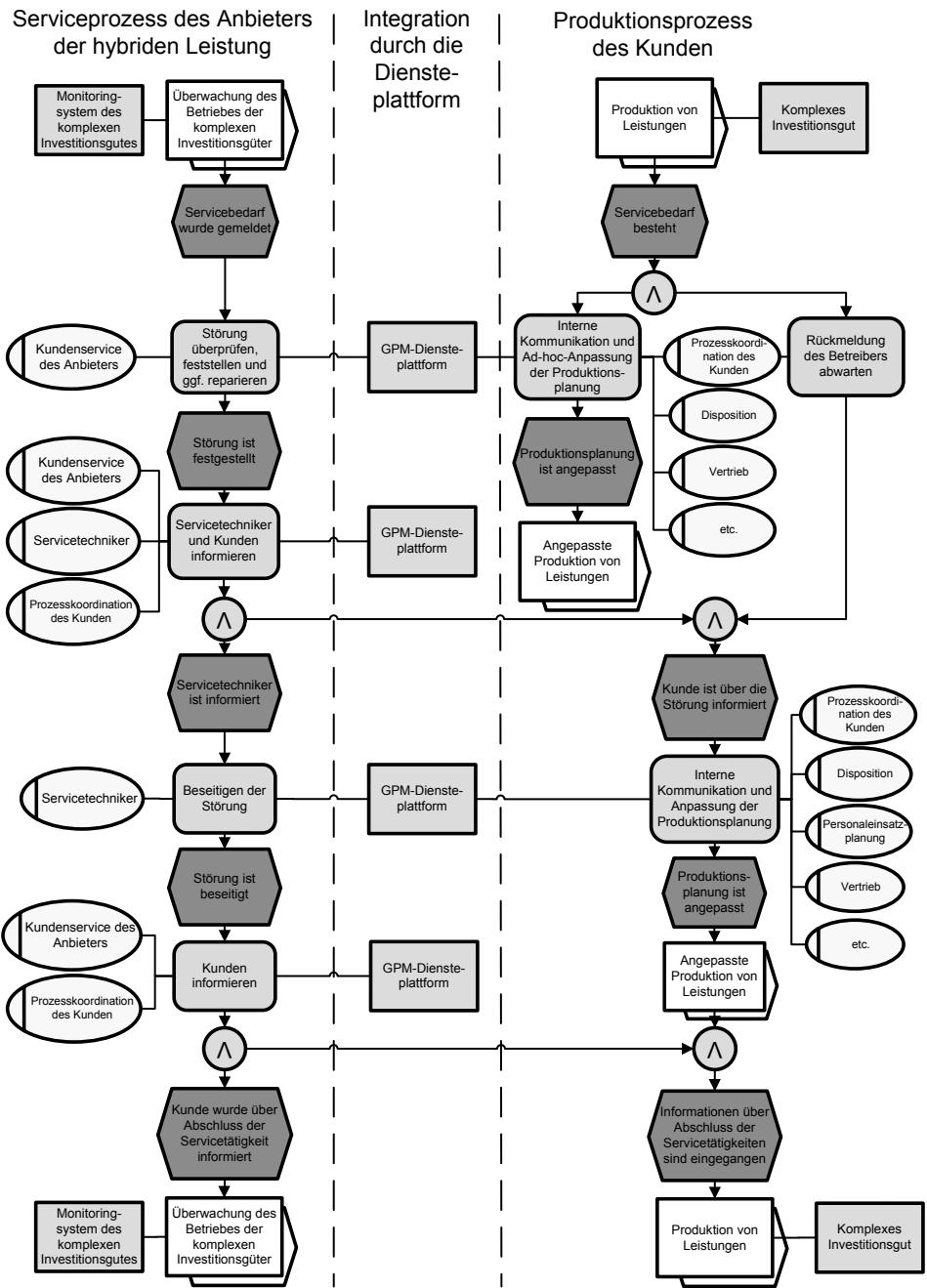


Abbildung 5: Prozessintegration des Anbieters der hybriden Leistung und des Kunden sowie prozessbezogener Kommunikation und Koordination anhand der GPM-Dienstesplattform

Im hier dargestellten Anwendungsszenario verfügen die komplexen Investitionsgüter über automatische Monitoring-Systeme, die basierend auf Sensorinformationen ihren aktuellen Status kommunizieren, d. h. gegebenenfalls Ausfälle bzw. Probleme melden. Wird der Betreiber über einen Ausfall einer Anlage per SMS automatisch informiert, so wird ein definierter Wartungs-Workflow angestoßen. Der Betreiber kann per Internet auf die Anlage zugreifen und den Mangel eventuell selbst beheben. Ist dies nicht der Fall, so stößt er über die Dienstplattform den Wartungsprozess des Vor-Ort-Servicetechnikers an und informiert darüber auch den Kunden. Die Integration unterschiedlicher Kommunikationsdienste ermöglicht eine adäquate Kommunikation zwischen den Beteiligten. Dem Servicetechniker können weitere Informationen zu dem Wartungsobjekt über synchrone Kommunikationsdienste mitgeteilt werden. Über ein mobiles Endgerät kann der Techniker auch auf Problemlösungsstrategien zugreifen, die im Kontext von Präzedenzfällen in einem Wiki dokumentiert wurden. Ebenso kann der Kunde über einen geeigneten, bzw. seinen Präferenzen entsprechenden Kommunikationsdienst über den aktuellen Stand des Wartungsprozesses bzw. über dessen erfolgreichen Abschluss informiert werden. Abbildung 6 visualisiert überblickartig die Kommunikationsbeziehungen.

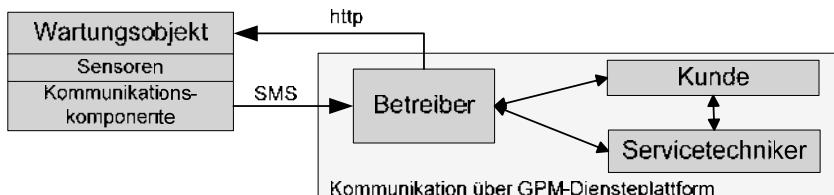


Abbildung 6: Kommunikationsbeziehungen im Anwendungsszenario

So erreicht die GPM-Dienstplattform eine stärkere unternehmensübergreifende Verzahnung, Synchronisation und Integration der Ausführungsebene des Geschäftsprozessmanagements des Betreibers und des Kunden. Die besonderen Vorteile der SOA der Dienstplattform liegen vor allem in der Möglichkeit der flexiblen Integration sowohl der Sensoren der unterschiedlichen Anlagen als auch der Sensoren zur Ermittlung des Kontextes der am Prozess beteiligten Personen. Auch die Vielzahl unterschiedlicher Kommunikationsdienste ermöglicht die Auswahl desjenigen Dienstes, der im jeweiligen Prozesskontext am sinnvollsten erscheint. Die Integration von Web-2.0-Diensten ermöglicht darüber hinaus eine langfristige Dokumentation von Erfahrungen, die in Prozessen entstanden sind, und fördert somit die ständige Weiterentwicklung der Prozesse.

7 Resümee und Ausblick

Der vorliegende Beitrag zeigte nach der Einführung der grundlegenden Begriffe und Konzepte im Bereich der Serviceorientierung und des Geschäftsprozessmanagements Potentiale von SOA für Werkzeuge des Geschäftsprozessmanagements auf. Darauf basierend wurde ein Rahmenkonzept für eine serviceorientierte GPM-Dienstplattform entwickelt und erste Überlegungen und Vorschläge für eine Implementierung dargelegt. Darauffolgend wurden die besonderen Anwendungspotentiale einer solchen Dienstplattform in einem Anwendungsszenario dargestellt.

Die skizzierten Überlegungen zeigen, dass die Serviceorientierung für Werkzeuge des GPM erhebliche Potentiale und neue Handlungsoptionen bieten kann. Insbesondere die flexible Integration von Services kann einen innovativen Beitrag zur Gestaltung des GPM leisten. So haben Unternehmen die Möglichkeit, individuell konfigurierte GPM-Werkzeuge einzusetzen, die auf ihre besonderen Anforderungen eingehen. Die weiteren Arbeiten im Projekt PROWIT zielen darauf ab, die Realisierbarkeit der Lösung sowie die aus ihr entstehenden Handlungsoptionen zu erforschen und auch ihre Grenzen aufzuspüren. Im Kontext der Realisierung sind vor allem standardisierte Datenaustauschformate notwendig, um eine problemlose unternehmensübergreifende Zusammenarbeit zu unterstützen [MNN07]. Darüber hinaus sind bei der Systemimplementierung vor allem auch Datenschutzaspekte zu berücksichtigen, um einen ungewollten Zugriff auf sensible Prozessdaten zu vermeiden, die im Kontext serviceorientierter Architekturen die Unternehmensgrenzen verlassen. Ausgehend von den zahlreichen Potentialen der Serviceorientierung sowie des Software-as-a-Service-Konzeptes bleibt zu beobachten, wie sich diese Ansätze im Bereich von GPM-Werkzeugen weiterhin entwickeln werden.

Danksagung

Die hier dargestellten Forschungsergebnisse wurden im Rahmen des Projektes „Prozessorientierter Web-2.0-basierter integrierter Telekommunikationsservice (PROWIT)“ erarbeitet, das durch das Bundesministerium für Bildung und Forschung (BMBF) unter dem Förderkennzeichen FKZ 01BS0833 gefördert wird. Die Autoren bedanken sich darüber hinaus bei den anonymen Gutachtern für die konstruktiven Anmerkungen, die zur Verbesserung dieses Artikels beigetragen haben.

Literaturverzeichnis

- [AHB09] Abelein, U., Habryn, F., Becker, A.: Towards a Holistic Framework for Describing and Evaluating Business Benefits of a Service Oriented Architecture. In: (Tosic, V., Hrsg.): Proceedings of the IEEE 13th Enterprise Distributed Object Computing Conference Workshops (EDOCW), Auckland, New Zealand 2009, S. 282-289.
- [Ac02] Ackermann, J., Brinkop, F., Conrad, S., Fettke, P., Frick, A., Glistau, E., Jaekel, H., Kotlar, O., Loos, P., Mrech, H., Raape, U., Ortner, E., Overhage, S., Sahm, S., Schmietendorf, A., Teschke, T., Turowski, K.: Vereinheitlichte Spezifikation von Fachkomponenten - Memorandum des Arbeitskreises 5.10.3 Komponentenorientierte betriebliche Anwendungssysteme, 2002. <http://www.wi2.info/fachkomponenten/docs/MEMO/Memorandum-final-2-44-mit-literatur-Web.pdf>
- [BHB10] Benlian, A., Hess, T., Buxmann, P.: Chancen und Risiken des Einsatzes von SaaS - die Sicht der Anwender. In: Wirtschaftsinformatik & Management 2 (2010) 2, S. 23-32.
- [BY10] Boh, W. F., Yellin, D. M.: Enablers and benefits of implementing Service-Oriented Architecture: an empirical investigation. In: International Journal of Information Technology and Management 9 (2010) 1, S. 3-29.
- [BDH08] Buxmann, P., Diefenbach, H., Hess, T.: Die Softwareindustrie. Ökonomische Prinzipien, Strategien, Perspektiven. Springer, Berlin et al. 2008.
- [BLH08] Buxmann, P., Lehmann, S., Hess, T.: Software as a Service. In: Wirtschaftsinformatik 50 (2008) 6, S. 500-503.
- [CS06] Chesbrough, H., Spohrer, J.: A Research Manifesto for Service Science. In: Communication of the ACM 49 (2006) 7, S. 35-40.

- [Do09] Dollmann, T., Fettke, P., Loos, P., Vanderhaeghen, D.: Web 2.0 Enhanced Automation of Collaborative Business Process Model Management in Cooperation Environments. In: Proceedings of the 20th Australasian Conference on Information Systems (ACIS), Melbourne, Australia 2009, S. 653-663.
- [Do05] Dostal, W., Jeckle, M., Melzer, I., Zengler, B.: Service-orientierte Architekturen mit Web Services. Konzepte - Standards - Praxis. Spektrum-Akademischer Verlag, Heidelberg 2005.
- [DF08] Drawehn, J., Frings, S.: Schnittstellen der Werkzeuge. In: (Spath, D., Weisbecker, A., Hrsg.): Business Process Management Tools 2008. Eine evaluierende Marktstudie zu aktuellen Werkzeugen. Fraunhofer IAO, Stuttgart 2008, S. 127-147.
- [DH08] Drawehn, J., Höß, O.: Ausführung von Geschäftsprozessen. In: (Spath, D., Weisbecker, A., Hrsg.): Business Process Management Tools 2008. Eine evaluierende Marktstudie zu aktuellen Werkzeugen. Fraunhofer IAO, Stuttgart 2008, S. 148-169.
- [EW08] Eymann, T., Winter, R.: SOA - Ein neues Paradigma der Gestaltung verteilter Informationssysteme? In: Wirtschaftsinformatik 50 (2008) 1, S. 70.
- [Fe09] Fettke, P.: Ansätze der Informationsmodellierung und ihre betriebswirtschaftliche Bedeutung: Eine Untersuchung der Modellierungspraxis in Deutschland. In: Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung 61 (2009) 8, S. 550-580.
- [FL02] Fettke, P., Loos, P.: Der Referenzmodellkatalog als Instrument des Wissensmanagements - Methodik und Anwendung. In: (Becker, J., Knackstedt, R., Hrsg.): Wissensmanagement mit Referenzmodellen. Konzepte für die Anwendungssystem- und Organisationsgestaltung. Springer, Berlin et al. 2002, S. 3-24.
- [FL04] Fettke, P., Loos, P.: Referenzmodellierungsforschung. In: Wirtschaftsinformatik 46 (2004) 5, S. 331-340.
- [FL05] Fettke, P., Loos, P.: Zur Identifikation von Strukturanalogien in Datenmodellen – Ein Verfahren und seine Anwendung am Beispiel des Y-CIM-Referenzmodells von Scheer. In: Wirtschaftsinformatik 47 (2005) 2, S. 89-100.
- [Ga07] Gaitanides, M.: Prozessorganisation – Entwicklung, Ansätze und Programme des Managements von Geschäftsprozessen. 2. Aufl., Vahlen, München 2007.
- [Gr98] Griffel, F.: Componentware: Konzepte und Techniken eines Softwareparadigmas. dpunkt, Heidelberg 1998.
- [He07] Heutschi, R.: Serviceorientierte Architektur. Architekturprinzipien und Umsetzung in die Praxis. Springer, Berlin 2007.
- [HFL10] Houy, C., Fettke, P., Loos, P.: Empirical Research in Business Process Management – Analysis of an emerging field of research. In: Business Process Management Journal 16 (2010) 4, accepted.
- [Ho10] Houy, C., Fettke, P., Loos, P., Van der Aalst, W. M. P., Krogstie, J.: BPM-in-the-Large – Towards a higher level of abstraction in Business Process Management. Global Information Systems Processes. World Computer Congress (WCC 2010). Springer, Brisbane, Australia 2010, accepted.
- [KLL09] Ko, R. K. L., Lee, S. S. G., Lee, E. W.: Business process management (BPM) standards: A survey. In: Business Process Management Journal 15 (2009) 5, S. 744-791.
- [KD08] Kopperger, D., Drawehn, J.: Marktüberblick Business Process Management. In: (Spath, D., Weisbecker, A., Hrsg.): Business Process Management Tools 2008. Eine evaluierende Marktstudie zu aktuellen Werkzeugen. Fraunhofer IAO, Stuttgart 2008, S. 24-38.
- [Lö07] Lörincz, P.: Evolution of Enterprise Systems. In: Proceedings of the International Symposium on Logistics and Industrial Informatics (LINDI), Wildau, Germany 2007, S. 64-69.
- [Lu08] Luo, D., Jiang, J., Sheng, B., Yang, M.: Research on Business Process Management System Based on SOA. In: (Pan, Y., Hrsg.): Proceedings of the IEEE 9th International Conference on Computer-Aided Industrial Design and Conceptual Design (CAID/CD), Kunming, China 2008, S. 1100-1103.
- [Ma07] Ma, C., Bacon, L., Petridis, M., Windall, G.: A Design Pattern for Integration of Business Process Management Systems. In: Proceedings of the IEEE International Conference on Information Reuse and Integration, Las Vegas, USA 2007, S. 239-244.
- [Ma09] Matt, C.: Software as a Service. In: Zeitschrift für Controlling & Management 53 (2009) 3, S. 151-152.

- [MB09] Meffert, H., Bruhn, M.: Dienstleistungsmarketing. Grundlagen - Konzepte - Methoden. 6. Aufl., Gabler, Wiesbaden 2009.
- [MNN07] Mendling, J., Neumann, G., Nüttgens, M.: Interchange Formats for Reference Models. In: (Fettke, P., Loos, P., Hrsg.): Reference Modeling for Business Systems Analysis. Idea, Hershey, PA, USA, et al. 2007, S. 337-354.
- [Of08] Offermann, P.: SOAM - Eine Methode zur Konzeption betrieblicher Software mit einer Serviceorientierten Architektur. In: Wirtschaftsinformatik 50 (2008) 6, S. 461-471.
- [OHB09] Offermann, P., Hoffmann, M., Bub, U.: Benefits of SOA: Evaluation of an Implemented Scenario against Alternative Architectures. In: (Tosic, V., Hrsg.): Proceedings of the IEEE 13th Enterprise Distributed Object Computing Conference Workshops (EDOCW), Auckland, New Zealand 2009, S. 352-359.
- [Or00] Ortner, E.: Komponentenorientierte Anwendungsentwicklung. In: Information Management & Consulting 15 (2000) 4, S. 62-72.
- [RS03] Rautenstrauch, C., Schulze, T.: Informatik für Wirtschaftswissenschaftler und Wirtschaftsinformatiker. Springer, Berlin et al. 2003.
- [RM06] Rust, R. T., Miu, C.: What Academic Research Tells Us About Service. In: Communications of the ACM 49 (2006) 7, S. 49-54.
- [Sc02] Scheer, A.-W.: ARIS – Vom Geschäftsprozeß zum Anwendungssystem. 4. Aufl., Springer, Berlin et al. 2002.
- [SK09] Scheer, A.-W., Klueckmann, J.: BPM 3.0. In: (Dayal, U., Eder, J., Koehler, J., Reijers, H.A., Hrsg.): Business Process Management. Lecture Notes in Computer Science, Vol. 5701. Springer Verlag, Berlin et al. 2009, S. 15-27.
- [SS08] Schmelzer, H. J., Sesselmann, W.: Geschäftsprozessmanagement in der Praxis. 6. Aufl., Hanser Verlag, München 2008.
- [Si07] Siedersleben, J.: SOA revisited: Komponentenorientierung bei Systemlandschaften. In: Wirtschaftsinformatik 49 (2007) Sonderheft, S. S110-S117.
- [Si08] Sinz, E. J.: SOA und die bewährten methodischen Grundlagen der Entwicklung betrieblicher IT-Systeme. In: Wirtschaftsinformatik 50 (2008) 1, S. 70-72.
- [SW08] Spath, D., Weisbecker, A. (Hrsg.): Business Process Management Tools 2008 – Eine evaluierende Marktstudie zu aktuellen Werkzeugen. Fraunhofer IAO, Stuttgart 2008.
- [TF09] Thomas, O., Fellmann, M.: Semantische Prozessmodellierung - Konzeption und informations-technische Unterstützung einer ontologiebasierten Repräsentation von Geschäftsprozessen. In: Wirtschaftsinformatik 51 (2009) 6, S. 506-518.
- [TTM08] Tiwari, A., Turner, C. J., Majeed, B.: A review of business process mining: state-of-the-art and future trends. In: Business Process Management Journal 14 (2008) 1, S. 5-22.
- [AH05] van der Aalst, W. M. P., ter Hofstede, A. H. M.: YAWL: yet another workflow language. In: Information Systems 30 (2005) 4, S. 245-275.
- [DJ05] van Dongen, B. F., Jansen-Vullers, M. H.: Verification of SAP reference models. In: (van der Aalst, W.M.P., Benatallah, B., Casati, F., Curbera, F., Hrsg.): Business Process Management. Lecture Notes in Computer Science, Vol. 3649. Springer, Berlin et al. 2005, S. 464-469.
- [VFL10] Vanderhaeghen, D., Fettke, P., Loos, P.: Organisations- und Technologieoptionen des Geschäftsprozessmanagements aus der Perspektive des Web 2.0. In: Wirtschaftsinformatik 52 (2010) 1, S. 17-32.
- [VZS05] Vanderhaeghen, D., Zang, S., Scheer, A.-W.: Interorganisationales Geschäftsprozessmanagement durch Modelltransformation. In: (Scheer, A.-W., Hrsg.): Veröffentlichungen des Instituts für Wirtschaftsinformatik (IWi), Universität des Saarlandes, Vol. 182, Saarbrücken 2005,
- [We07] Weske, M.: Business Process Management – Concepts, Languages, Architectures. Springer, Berlin et al. 2007.

Datentechnische Integration und Visualisierung von Anforderungen innerhalb von CAD-Systemen

Tim Sadek, Daniel Meuris

Lehrstuhl für Maschinenelemente und Konstruktionslehre
Ruhr-Universität Bochum
Universitätsstr. 150
44801 Bochum
sadek@lmk.rub.de
meuris@lmk.rub.de

Abstract: Ein wesentliches Ziel der Produktentwicklungsprozesse ist die fortlaufende Reduzierung der Komplexität ohne wichtige Merkmale aus den Augen zu verlieren. Weiterhin zeichnet sich die virtuelle Produktentwicklung durch eine durchgängige Integration verschiedener Produktinformationen und Produktmodelle aus, wobei dem Gestaltmodell in Form von CAD-Daten eine zentrale Bedeutung zukommt. Dabei basieren diese CAD-Systeme jedoch nicht oder nur zum Teil auf den Grundprinzipien der Konstruktionsmethodik, so dass die vor dem Entwurf entstandenen Informationen nur schwer mit dem Gestaltmodell verknüpft werden können. Am Beispiel der Anforderungsentwicklung wird in dieser Arbeit ein CAD-integriertes Werkzeug vorgestellt, dass es dem Konstrukteur ermöglicht, Anforderungen direkt im CAD-System zu erfassen und mit dem Geometrie-Modell zu verknüpfen. Neben der datentechnischen Integration wird auch eine Visualisierung von Anforderungen im CAD-System erstellt. Diese unterstützt den Konstrukteur, um die Vielzahl von abstrakten Informationen zu Baugruppen, Bauteilen und Features zuordnen zu können, was die Qualität und Konsistenz der erzeugten Modelle erhöht.

1 Einleitung

Die Art und Weise, wie eine technische Problemstellung in eine Lösung überführt werden kann, wird mit den Hilfsmitteln der Konstruktionsmethodik beschrieben. Kerngedanke ist dabei die Gliederung des gesamten Entwicklungsprozesses in mehrere Teilphasen, deren Arbeitsergebnisse zum Ziel haben, das zukünftige Produkt bei steigendem Konkretisierungsgrad zu beschreiben. Die *Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte VDI 2221* definiert diesen Vorgang wie folgt:

„Die Produktentwicklung (Entwickeln und Konstruieren) umfasst die Gesamtheit aller Tätigkeiten mit denen – ausgehend von einer Aufgabenstellung – eine Lösung und die zur Herstellung und Nutzung benötigten Informationen erarbeitet werden. [...] Sie

schließen die gedankliche Zusammensetzung der einzelnen Funktionen und Teile eines Produktes, den Aufbau zu einem Ganzen und das Festlegen aller Eigenschaften ein.“ [VDI93]

Am Ende dieser Prozesse steht ein realer oder virtueller Prototyp¹, der verschiedene Rückschlüsse auf das Verhalten des realen Produktes zulässt. Insbesondere die virtuelle Entwicklung von Produktmodellen wurde bedingt die großen Fortschritte der Informations- und Kommunikationstechnik in den letzten Jahrzehnten massiv weiterentwickelt. Heute existiert unter dem Sammelbegriff der CAx-Technologie² eine Vielzahl von problem- und lösungsspezifischer Entwurfswerkzeuge, die den Entwickler bei Analyse- und Syntheseaufgaben unterstützen. Problematisch ist aber, dass sich der Einsatz dieser Werkzeuge bislang auf die späten Entwicklungsphasen, die durch die Gestalt eines Produktes in ihren verschiedenen Ausprägungen dominiert werden, erstreckt [Con08]. WEBER formuliert diesen Umstand wie folgt: „*Existing TTS concepts have not shown any impact on CAx-Development*“[Web09] Demzufolge ist es bis heute ein Problem abstrakte Informationen aus den frühen Phasen sowie die Entwicklungsprozesse in einem einheitlichen, rechnerbasierten Systemmodell zu integrieren, weil die verfügbaren Softwarewerkzeuge bislang nur auf bauteil- und geometrieorientierten Modellen und nicht explizit auf den Grundlagen einer Konstruktionsmethodik basieren. Dabei sollte gerade die Gestaltung eines Produktes permanent an Informationen aus den frühen Phasen gespiegelt werden können, um einerseits grundlegende Fehler zu vermeiden und andererseits den Kommunikationsaufwand, der durch ein fortwährendes, manuelles Abgleichen von Geometriemeerkmalen und technischen Anforderungen entsteht, zu verringern. Die mitunter wichtigsten Informationen, die in frühen Entwicklungsphasen gesammelt werden, sind in den Produktanforderungen zu sehen, da sie alle Rahmenbedingungen, Restriktionen und geforderten Eigenschaften des zukünftigen Produktes beschreiben [Ehr06]. Dieser enormen Bedeutung werden jedoch die momentan verfügbaren kommerziellen CAD-Systeme nicht gerecht, da sie keine Werkzeuge bereitstellen, die eine zentrale Verwaltung von Produktanforderungen erlauben. Deshalb soll in dieser Arbeit eine prototypische, werkzeugbasierte Integration der Abbildung von technischen Anforderungen in CAD-Systemen vorgestellt werden. Dies soll zum Einen als Kommunikationsbasis für den Konstrukteur dienen und zum Anderen helfen, die Schnittstellenprobleme zwischen den einzelnen Entwicklungsphasen zu beherrschen.

Dabei sollen folgende Fragestellungen beantwortet werden:

- Wie können die Produktweiten Abhängigkeiten von Anforderungen in einem CAD-System verwaltet werden?
- Wie kann das produktsspezifische Wissen aus den frühen Entwicklungsphasen im CAD-System modelliert und visualisiert werden?

¹ Diese Arbeit befasst sich vordergründig mit virtuellen Prototypen, die im Folgenden mit dem Begriff „*Produktmodell*“ bezeichnet werden.

² Beispiele solcher Systeme: CAM – Computer Aided Manufacturing, CAE – Computer Aided Engineering, CASE – Computer Aided Software Engineering und CAD – Computer Aided Design

Nach den einleitenden Bemerkungen in diesem Kapitel werden zunächst die Begriffe der virtuellen Produktentwicklung und der technischen Anforderungsentwicklung geklärt. Aufbauend auf diesen Erläuterungen wird dann der Integrationsansatz vorgestellt, der neben der informationstechnischen Grundlage auch das methodische Fundament für eine Integration von Anforderungen in ein CAD-System legt. Schließlich wird der „*EPI-Editor*“³ als prototypisches Werkzeug vorgestellt und am Beispiel der Konstruktion eines Planetengetriebes evaluiert.

2 Virtuelle Produktentwicklung

Die klassische Produktentwicklung wird durch die zunehmende Unterstützung von Software-Systemen im Produktentwicklungsprozess immer mehr durch die virtuelle Produktentwicklung ersetzt [Gho08]. Virtuelle Produktentwicklung umfasst somit „*im weitesten Sinne alle rechnerunterstützten Entwicklungsprozesse, einschließlich der Konzipierung, Konstruktion, sowie Simulation der Entstehung und des Verhaltens zukünftiger Produkte*“ [Abr07]. Ein wichtiger Begriff im Zusammenhang der virtuellen Produktentwicklung ist der Modellbegriff. Ein Modell ist ein Abbild realer oder künstlicher Objekte, das die für einen Anwendungsfall relevanten Informationen beschreibt. Diese Zuordnung ist disjunkt und „*wird nur für die modellbenutzenden Subjekte beschränkt auf festgelegte Zeitintervalle im Kontext gedanklicher oder tatsächlicher Operationen hergestellt*“ [Sta73]. Demzufolge ist das Produktmodell eine schematische Sammlung von Informationen, die das reale Produkt so beschreiben, dass Rückschlüsse auf das spätere Verhalten möglich werden [Suh93].

2.1 Partialmodelle der virtuellen Produktentwicklung

Bedingt durch die Komplexität heutiger Produkte und ihrer Entwicklungsprozesse kann ein Produktmodell nicht als ganzheitliche Einheit betrachtet werden. Vielmehr zerfällt es in Partialmodelle, die sich an den einzelnen Entwicklungsphasen orientieren [Lin08]. Abbildung 1 stellt diese Partialmodelle und die Tätigkeiten, die zu ihrer Erstellung führen, da.

³ Early-Phase-Integration Editor

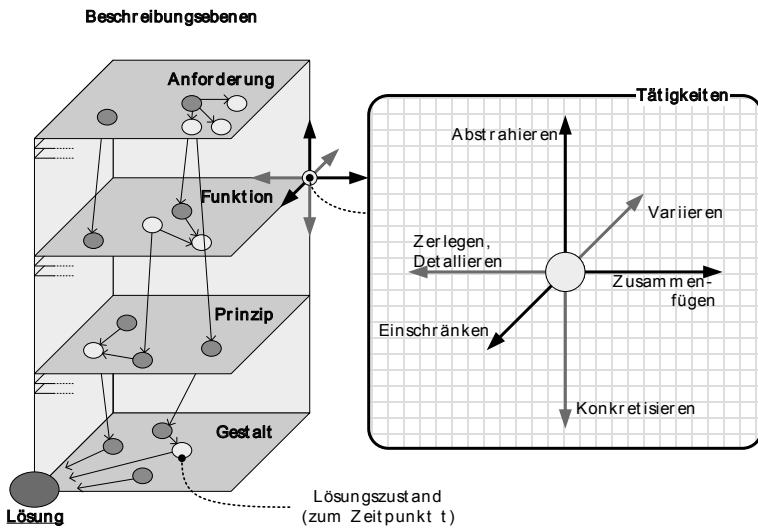


Abbildung 1: Die vier wesentlichen Partialmodelle der Produktentwicklung [Lin08]

Das **Anforderungsmodell** enthält die Anforderungen an das zukünftige Produkt, sowie alle Rahmenbedingungen der Entwicklung. Die Modellelemente des Anforderungsmodells sind Texte, Skizzen und technische Daten, sogenannte Spezifikationen. Aufgrund der Heterogenität der Informationen besitzt das Anforderungsmodell meistens einen informalen Charakter, was dessen Integration in die übrigen Partialmodelle erschwert. Das **Funktionsmodell** beschreibt die Art und Weise des funktionalen Systemzusammenhangs in Form von funktionalen Strukturen. Die Modellelemente des Funktionsmodells sind Funktionen, als Subjekt-Verb-Kombinationen und Relationen, die in Form von zwei-dimensionalen Graphen dargestellt werden. Das **Prinzipmodell** bildet die physikalische Wirkstruktur eines Produktes ab. Dieses Modell wird durch Texte, Skizzen und technischen Daten beschrieben, wobei die Skizzen oftmals einer formalen Notation unterliegen. Beispielsweise gibt es allgemein akzeptierte Notationen für die Skizzierung der Wirkungsweise von getriebetechnischen Systemen oder elektrotechnischen Wirkzusammenhängen. Das **Gestaltmodell** enthält die Geometrie und Struktur des Produktes. Hier werden Bauteile und Baugruppen bzw. deren Beziehungen und Bestandteile durch geometrische Körper beschrieben. Weiterhin bildet das Gestaltmodell die Informationsbasis für nachfolgende Fertigungsprozesse, da aus der Gestalt eines Produktes die notwendigen Fertigungsmittel und -prozesse abgeleitet werden können.

2.2 CAX-Technologie

Aufgabe der virtuellen Produktentwicklung ist nun die Bereitstellung von rechnerinternen Beschreibungen und methodischer Unterstützung bei der Entwicklung dieser Partialmodelle. Insbesondere die zunehmende Komplexität von Produkten erfordert den intensiven Einsatz von CAX-Technologien, da sich dadurch „*signifikant erweiterte Möglichkeiten zur Produktmodellierung, die damit auch zu neuen Formen*

methodischer Vorgehensmodelle in der Produktentwicklung führen“[Vaj08], ergeben. Dabei ist gerade im Maschinenbau die CAD-Methodik ein zentraler Baustein und Treiber für neue Innovationen [Szi01].

Heutige Systeme bilden die Geometrie parametrisch ab. Unter Verwendung von parametrischen Beschreibungen der Elemente wird es möglich, im Nachhinein das Modell durch Variation der Parameter und der damit verknüpften mathematischen Zusammenhänge zu verändern bzw. zu steuern. Die Verwendung von sogenannten Features erlaubt neben der Integration von Modellparametern auch die Einbringung von semantischen Informationen im Geometriemodell, da sie im Zusammenhang mit der CAD-Technologie als Aggregation von Geometrie und Semantik gesehen werden kann [Spu97]. WEBER definiert Features als „*informationstechnische Elemente, die Bereiche von besonderem (technischem) Interesse von einzelnen oder mehreren Produkten darstellen*“[Web96]. Abbildung 2 zeigt den Aufbau von CAD-Modellen am Beispiel eines handelsüblichen PCs. Neben der Struktur, die sich in Gesamtsystem, Modul und Bauteil untergliedert; sind auch verschiedene Features aufgeführt. So können auf der Gesamtsystemebene verschiedene Konfigurationsmöglichkeiten des Produkts durch Features ausgedrückt werden. Auf der Modulebene werden spezifische elektronische Schnittstellen durch Features repräsentiert. Weiterhin werden geometrische Operationen durch Features abgebildet. So entspricht das „*Entfernen*“ eines zylinderartigen Volumens einer metrischen Bohrung. Diese Information kann dann von Benutzern während der Produktion genutzt werden, um die Teile zu fertigen.

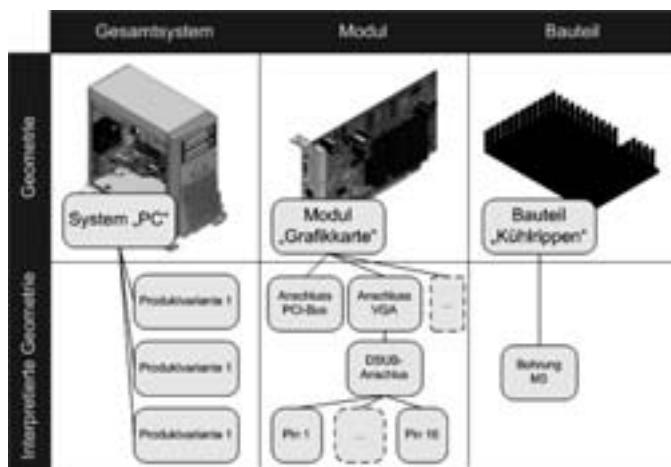


Abbildung 2: Features als Aggregation von Geometrie und Semantik auf verschiedenen Detailierungsebenen

Zusammenfassend lässt sich sagen, dass sich die Funktionalität von CAD-Systemen nicht nur über die reine Erstellung von Geometrie erstreckt. Vielmehr ist unter diesem Begriff die „*umfassende Unterstützung aller Konstruktionstätigkeiten durch Rechnerhilfsmittel zu verstehen, d.h. es gehören außer der Geometrieerzeugung auch alle anderen Tätigkeiten des Berechnens des Simulierens und der*

Informationsgewinnung zum Zwecke der Erstellung eines Produktmodells dazu“ [Vaj08].

3 Technische Anforderungsentwicklung

Die Überführung der technischen Problemstellung in ein Problemmodell ist Aufgabe der Planungsphase bzw. der Anforderungsentwicklung⁴. Dabei kann zwischen Kunden- und Systemanforderungen [Ehr06] unterschieden werden. Die Kundenanforderungen werden in einem Lastenheft aufgeführt und beinhalten die Beschreibung des technischen Problems. Das Lastenheft ist die Vertragsgrundlage zwischen dem Kunden und dem Lieferanten und das entwickelte Produkt muss diese Anforderungen erfüllen. Mit Hilfe der Kundenanforderungen werden dann die Systemanforderungen entwickelt. In einem sogenannten Pflichtenheft zusammengestellt, beantworten sie die Frage, *wie* und *womit* ein Element des Lastenhefts zu lösen ist. Sie stellen somit eine Konkretisierung der Kundenanforderungen dar. Diese Transformation muss immer eine Prüfung auf Realisierbarkeit und Widerspruchsfreiheit beinhalten, denn Mängel in den Anforderungen sind hauptverantwortlich für das Scheitern von Entwicklungsprojekten [Ehr07].

Weiterhin haben die Anforderungen nicht nur eine technische Dimension, sondern sie legen vielmehr auch die Rahmenbedingungen fest, in denen die Entwicklung ablaufen soll. Dazu gehören Kosten- und Zeitvereinbarungen, juristische Aspekte in Form von zu beachtenden Normen und Gesetzen, oder auch die verfügbaren Fertigungs- und Produktionsverfahren. Somit bestimmen die Anforderungen die Vorgehensweise in allen weiteren Entwicklungsphasen und sind erheblich für die entstehenden Kosten verantwortlich [Pah07, Ehr06].

3.1 Klassifizierung von Anforderungen

Eine wichtige Aufgabe während der Entwicklung von Anforderungen ist die Klassifizierung bzw. Strukturierung und ihre Spezifizierung. Es gibt zwei Typen von Anforderungen. Die qualitativen Anforderungen beschreiben Produkteigenschaften, die oftmals aus Kundenwünschen abgeleitet werden, in rein textueller Form. Diese Formulierungen können implizit („Die Pumpe ist geräuscharm auszuführen.“) oder explizit („Das Pumpe darf nicht lauter als der Motor sein.“) sein. Daneben sind es die quantitativen Anforderungen, die als Punkt-, Minimal-/Maximal- oder Bereichsforderungen unter allen Umständen erfüllt werden müssen. Dabei haben Punktfordernisse keinen Toleranzbereich („Übersetzung Getriebe i=10“). Bereichsforderungen geben ein zulässiges Intervall an („Bauhöhe 300mm ± 2mm“), innerhalb dessen die Forderung als erfüllt anzusehen ist. Minimal-/Maximalfordernisse geben einen festgelegten Wert an, ab dem die Forderung erfüllt ist („Wirkungsgrad ≥ 95%“).

⁴ Die Planungsphase umfasst noch weitere Tätigkeiten, die jedoch eine übergeordnete Rolle spielen. Dazu gehört z.B. die Ideenfindung.

Diese Forderungen können mehr oder weniger gut erfüllt werden. Ein Lösungsvorschlag, der diesen Forderungen nicht genügt, ist inakzeptabel.

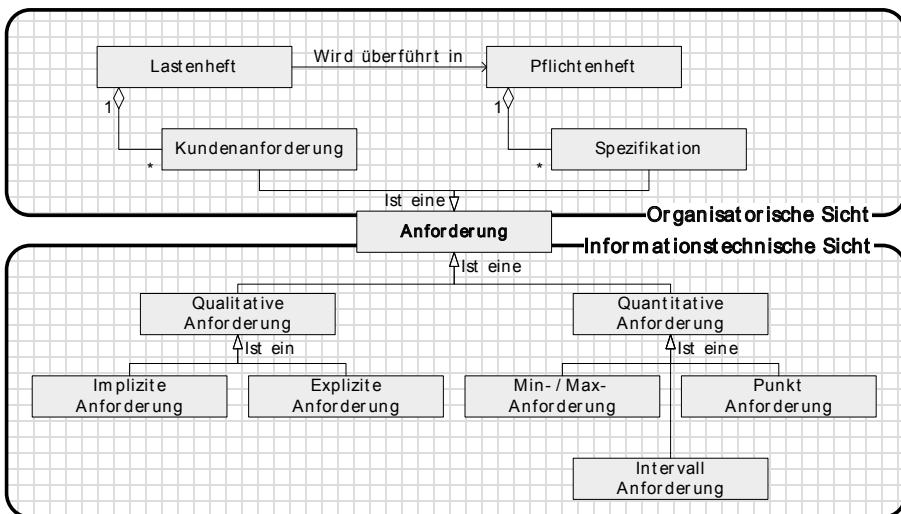


Abbildung 3: Organisatorische und datentechnische Sicht auf Anforderungen (in Anlehnung an [Lin08])

Abbildung 3 bringt die erwähnten Begriffe in einer objektorientierten Struktur zusammen. Gezeigt werden die organisatorische Sicht auf die Anforderungen und die damit verbundene Unterteilung in Lasten- und Pflichtenhefte sowie die informationstechnische Sicht, die die Anforderungen nach ihrem Informationsgehalt unterteilt.

3.2 Beziehungen zwischen Anforderungen

Die Klassifizierung von Anforderungen reicht aber nicht aus, um schon in dieser frühen Entwicklungsphase mögliche Fehlerquellen zu entdecken. Vielmehr müssen die einzelnen Anforderungen in ihrer Gesamtheit betrachtet und ihre Beziehungen untereinander ermittelt werden. Dabei gibt es verschiedene Typen von Beziehungen, die Anforderungen miteinander verknüpfen. Abbildung 4 zeigt diese Relationen und belegt sie mit Beispielen.

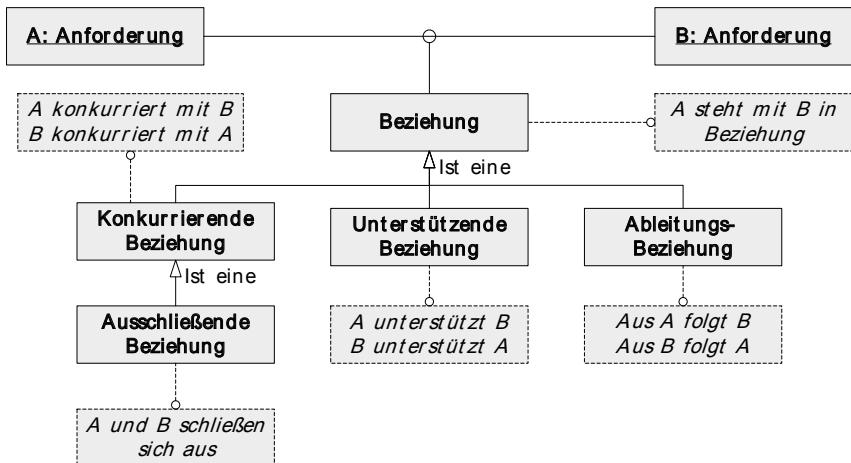


Abbildung 4: Mögliche Beziehungen von Anforderungen (in Anlehnung an [Lin08])

Wie die Abbildung zeigt können zwei⁵ Anforderungen miteinander konkurrieren und sich im Extremfall sogar ausschließen. Dies ist z.B. bei Automobilen in Bezug auf den Kraftstoffverbrauch und die Stabilität der Fall. Um den Verbrauch zu minimieren, muss eine Leichtbauweise angestrebt werden. Eine stabile Karrossiere sollte allerdings eine hohe Steifigkeit besitzen, die wiederum bei Leichtbauweise nur sehr schwer realisiert werden kann. Im Gegensatz dazu können Anforderungen sich gegenseitig auch positiv beeinflussen. Um beim Beispiel des Automobils zu bleiben, könnten das z.B. Anforderungen bezüglich der Gewichtsreduktion und der Cabriolet-Bauweise sein.

Zuletzt gibt es noch eine weitere Anforderungsbeziehung, die die Nachverfolgbarkeit von Änderungen abbilden kann. Die Ableitungsbeziehung zeigt an, dass eine Anforderung notwendig wurde, weil sie aus einer ursprünglichen Menge von Anforderungen resultiert. Ein Beispiel für solch eine Beziehung sind beispielsweise Anforderungen, die nötig werden, wenn der Kunde ein Produkt in ein bestimmtes Land liefern möchte.

4 Integrationsansatz

Da der zentralen Bedeutung der Anforderungen (s. Kap. 3) auch in der rechnergestützten Konstruktion Rechnung getragen werden muss, stellt sich die Frage, wie die objektorientierte Beschreibungsmöglichkeiten von Anforderungen und deren Beziehungen (s. Kap. 3) in CAD-Werkzeuge integriert werden können. Abbildung 6 zeigt die relevanten Partialmodelle für diese Integration.

⁵ Es handelt sich eigentlich um n:m Beziehungen. Der Übersichtlichkeit halber sollen hier jeweils nur zwei Anforderungen miteinander verknüpft werden.

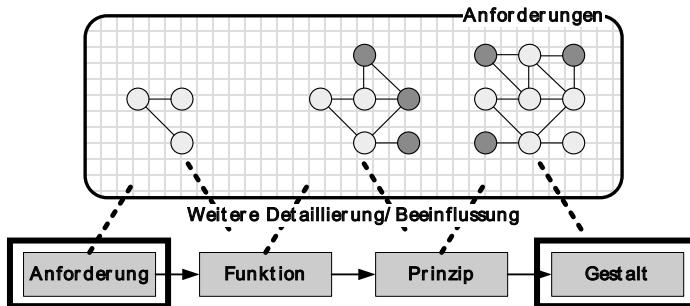


Abbildung 5: Bedeutung der Anforderungen (in Anlehnung an [Lin08])

Demzufolge muss ein CAD-integriertes Werkzeug implementiert werden, das in der Lage ist, dem Konstrukteur das Produktwissen, das in den frühen Entwicklungsphasen aufgebaut wird, verfügbar zu machen bzw. es zu visualisieren. Dabei sollen insbesondere Anforderungen bzw. deren Strukturen fokussiert werden. Der Benutzer soll erkennen, welche Features und Parameter des CAD-Modells in welcher Art und Weise durch eine Anforderung hervorgerufen wurden. Außerdem sollen die externen Einflussfaktoren, die in Form von Normen, Richtlinien, Gesetzen und anderen Dokumenten abgelegt sind, in der CAD-Umgebung im besten Falle kontextsensitiv bereitgestellt werden. Methodische Grundlage für die Integration bildet die Produktmodellierung auf Basis von Produktmerkmalen und Eigenschaften nach WEBER [Web03]. Dieser integriert die Nutzung von rechnerbasierten Hilfsmitteln in die Produktentwicklungsprozesse und ist zudem gut auf die CAD-Technologie anwendbar [Vaj08]. Als Bezeichnung für diesen Ansatz haben sich die englischen Bezeichnungen *Characteristic-Property-Modelling (CPM)* für den Produktmodell-Teil und *Process-Driven-Development (PDD)* für den prozessmodellierenden Anteil durchgesetzt. Zentrale Annahme dieses Ansatzes ist die Unterteilung eines Produktmodells in Produktmerkmale und Produkteigenschaften. Produktmerkmale sind alle Elemente des Produktmodells, die vom Entwickler direkt beeinflussbar sind, wie z.B. die Geometrie und Parameter innerhalb der CAD-Modelle. Produkteigenschaften sind die Daten, die das zukünftige Verhalten des Produkts beschreiben. Dazu können Eigenschaften, wie Festigkeit, Kinematik usw. gehören. Sie können nicht direkt manipuliert werden, sondern ergeben sich aus den verschiedenen Produktmerkmalen. Zwar werden diese Attribute des Produktmodells auch schon in den klassischen Vorgehensmodellen verwendet, aber bei CPM/PDD rücken diese in den Mittelpunkt der Betrachtung. Abbildung 6 zeigt den für die Integration von Anforderungen in ein CAD-System relevanten Regelkreis, der aus Anwendung des CPM/PDD Ansatzes für das hier vorgestellte Problem hervorgeht.

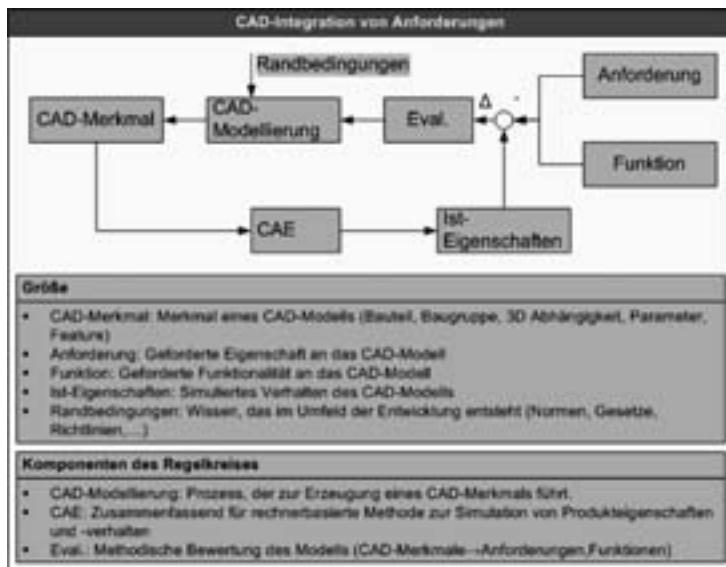


Abbildung 6: Grundlage der Anforderungsintegration (in Anlehnung an [Web03])

Ausgehend von Produktanforderungen und -funktionen erfolgt nach einer methodischen Bewertung die CAD-Modellierung. Ziel ist es CAD-Merkmale abzuleiten, die aus den Anforderungen entnommen werden können. Zwar können nie die Anforderungen an ein Produkt in ihrer Gesamtheit direkt auf CAD-Merkmale abgebildet werden, aber die Komplexität der Konstruktion und des Konstruktionsprozesses kann allein durch die Bereitstellung der Anforderungen im CAD-Modell reduziert werden und somit eine effizientere Arbeitsweise unterstützen. Die Gesamtheit des Produktwissens, das sich aus den Randbedingungen der Entwicklung ergibt, wird während dieses Vorgangs ebenfalls zur Verfügung gestellt, um dem Konstrukteur unnötige Recherchearbeiten zu ersparen. Die Rückführung erfolgt über computerbasierte Analyseverfahren (z.B. FEM – Finite Elemente Methode), die die verschiedenen CAD-Merkmale wieder auf Anforderungen abbilden können.

5 Entwurf und Implementierung des CAD-Anforderungswerkzeugs

Für die Implementierung der Schnittstelle wurde das CAD-System „Autodesk Inventor 2009“ (AI) des Anbieters „Autodesk“⁶ gewählt. Es bietet dem Software-Entwickler vielfältige Möglichkeiten zur Erweiterung der Grundfunktionalität durch Anbindung eigenen Codes. Neben der Makrosprache *Visual Basic for Applications (VBA)* die von Microsoft vertrieben wird, können über das *Component Object Model (COM)* beliebige

⁶ <http://www.autodesk.de>

COM-fähige⁷ Codebibliotheken entweder integriert als *AddIn*- oder unabhängig als *Standalone*-Systeme, die verfügbaren Objekte von AI nutzen.

Das CAD-System bietet die Möglichkeit der Modellierung von einzelnen Bauteilen und von ganzen Baugruppen. Die Bereitstellung der verfügbaren Befehle zur Manipulation des CAD-Modells wird dabei über die Verwendung von sog. Umgebungen gesteuert. So gibt es eine Umgebung für die Bauteilbearbeitung, den Zusammenbau oder für die Zeichnungsableitung. Des Weiteren besteht die Möglichkeit der Definition eigener Umgebungen, die die vorhandenen erweitern. So gibt es beispielsweise eine Umgebung für die Festigkeitsanalyse von Bauteilen, die die Befehle der Bauteilumgebung um die Aufprägung von Lasten oder Lagerungen, Analyseverfahren oder Berichterstellung erweitert. Für die CAD-Integration werden zwei neue Umgebungen definiert, die jeweils für Bauteile und Baugruppen die neue Funktionalität der Anforderungs- und Funktionsbearbeitung bereitstellen.

AI bildet die CAD-Modelle in der API als featurebasierte Modelle ab. Über die entsprechenden Features können dann die Geometrieelemente ermittelt werden, die durch bestimmte Features erzeugt, gelöscht oder manipuliert wurden. Eine Stärke von AI ist Verwendung des Attribut-Konzepts. Dies erlaubt eine interne⁸ Attributierung beliebiger Datenobjekte des CAD-Modells. Die Organisation dieser Attribute erfolgt dabei über Attributsätze, die über einen modellweiten, eindeutigen Bezeichner angelegt werden können. Jeder dieser Attributsätze kann wiederum beliebig viele Attribute aufnehmen. Zulässige Datenwerte für die Attribute sind Zahlen (Ganzzahl, Fließkomma), Wahrheitswerte (Boolean), Zeichenketten (String) oder beliebige Binärdaten (Binary). Mit Hilfe der Attribute können somit in AI beliebige Objekte des CAD-Modells mit externen Informationen angereichert werden.

Die Implementierung dieser CAD-Integration wird als Add-In ausgeführt. Gegenüber VBA bringt das den Vorteil der Möglichkeit der Verwendung einer objektorientierten Programmiersprache. Die Wahl der Programmiersprache fällt auf C#. Diese Sprache ist Bestandteil der .NET Framework Plattform von *Microsoft* und weist in der grammatischen Definition große Ähnlichkeit zu *JAVA* auf.

Zusammenfassend kann gesagt werden, dass die Softwarelösung als COM-AddIn für die CAD-Software Autodesk Inventor 2009 implementiert wird. Dabei kommt die Programmiersprache C# zum Einsatz, die auf dem .NET Framework 2.0 basiert. Es werden zwei Benutzerumgebungen erstellt, die jeweils für Bauteile und Baugruppen eine Betrachtung von Anforderungen erlauben.

Abbildung 9 zeigt die Architektur des *Early Phase Integration Editors* (EPI). Dargestellt werden die Befehlsumgebungen für Bauteile und Baugruppen, die über ein COM-Addin mit dem EPI Editor kommunizieren. Zusätzlich erfolgt die Bearbeitung der Datendefinitionen in einem Verwaltungsmodul. Über ein Datenzugriffsmodul kann der

⁷ Z.B. Clients, die mit C++, .Net Sprachen oder mit Visual Basic implementiert wurden.

⁸ Attribute können nur über die API und nicht über die herkömmliche Oberfläche gesteuert werden.

EPI-Editor mit drei verschiedenen Datenbasen interagieren. Die Datenbank *Partialmodell-Definitionen* beherbergt die datenbezogenen Definitionen der Modellelemente und deren Verknüpfungen. Hier werden die Konzepte für Anforderungen(vgl. Kap. 3 → Abbildung 3, Abbildung 5) in einer relationalen Struktur auf Basis von SQL abgelegt. Hier kommt der SQL Server Express zum Einsatz. Die instanzierten Modellelemente werden in der Datenbank für Partialmodell-Instanzen verwaltet. Dabei entsprechen jeweils ein Modellelement und eine Relation einem Datensatz. Im dritten Datenteilbereich wird das verknüpfte Wissen in Form von Dokumenten oder Internet-Verknüpfungen abgelegt.

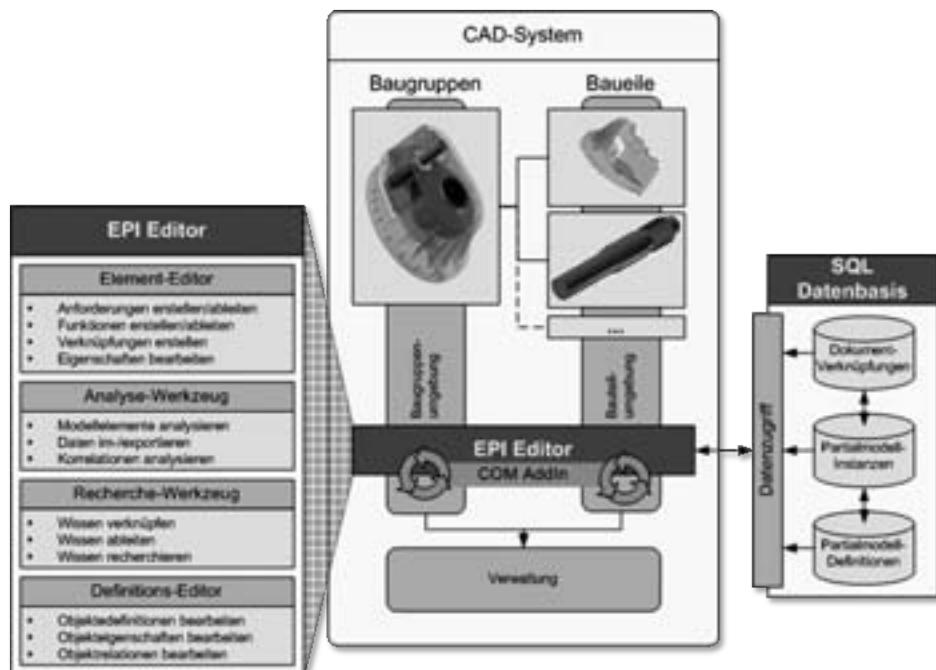


Abbildung 7: Architektur des EPI-Editors

Der EPI-Editor selber erlaubt das Editieren und Analysieren der Partialmodelle, die Recherche und Ableitung von Wissen. Weiterhin wird das gesamte Partialmodell mit allen verfügbaren Daten angezeigt. Dadurch kann der Entwickler jederzeit einen Überblick über das Gesamtsystem erhalten, um eventuelle Problemstellen ausfindig zu machen. Der EPI Editor wird als Zusatzpaket zu der normalen Funktionalität von AI konzipiert. Um die zusätzlichen Befehle nutzen zu können, muss die spezielle Umgebung aktiviert werden. In AI gibt es ein Menü, das die verfügbaren Umgebungen auflistet. In dem Hauptformular der Anwendung werden die Bestandteile der Partialmodelle aufgelistet. Zudem erfolgt hier die Neuanlage und Bearbeitung der Modellelemente. Abbildung 9 zeigt das Formular, das sich in zwei Bereiche untergliedert.

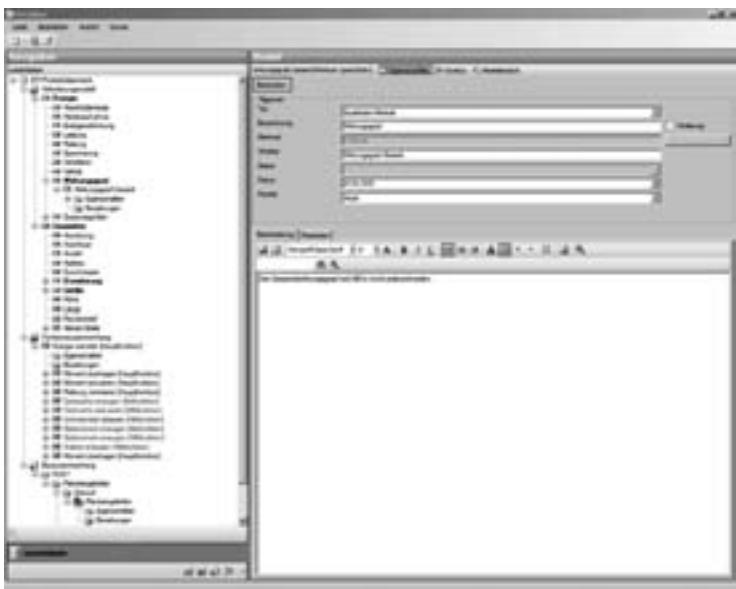


Abbildung 8: Hauptformular des EPI-Editors

Im rechten Teil werden die Bestandteile der Partialmodelle für Anforderungen und Bauzusammenhänge als Baumstruktur abgebildet. Die Verwendung hierarchischer Modellstrukturen bietet für die einzelnen Modelle verschiedene Vorteile. Die Anforderungen können in dieser Darstellung zu ihren Merkmalen zugeordnet werden. Baustrukturen, die sich wiederum in Bauteile und Baugruppe untergliedern, bilden die Grundlage für die Verbindung zu konkreten CAD-Daten. Die fortlaufende Konkretisierung wird durch die Darstellung als Strukturbau und durch grafische Notationen unterstützt. Der Baum zeigt alle verfügbaren Daten zu einem Modellelement und parallel dazu werden wichtige Informationen, wie verknüpfte Anforderungen oder Parameter in einem Grafikfenster angezeigt (vgl. Abbildung 10).

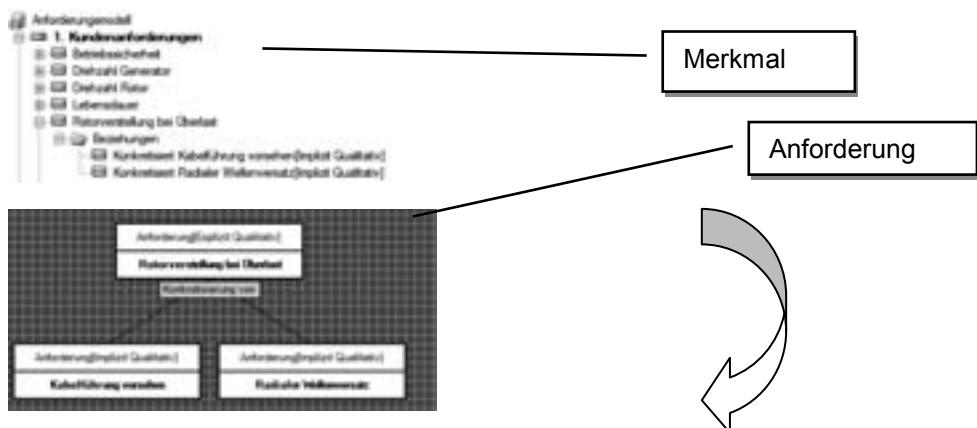


Abbildung 9: Baum- und Grafikdarstellung der Modell-Elemente

Die hier gezeigten Formulare und Oberflächenelemente werden in einer Bibliothek gekapselt. So wird es möglich, sie aus der CAD-Umgebung heraus aufzurufen, um so den Entwickler gezielt auf wesentliche Aspekte des technischen Systems und der Randbedingungen aufmerksam zu machen. Zusätzlich kann durch die Verknüpfung von Parametern und Anforderungen gezeigt werden, welche Parameter durch Anforderungen determiniert sind.

6 Evaluierung des Integrationswerkzeugs am Beispiel eines Planetengetriebes

Der entwickelte Editor soll am Beispiel der Entwicklung eines Planetengetriebes für eine Windkraftanlage evaluiert werden. Dieses technische System bietet sich an, weil die grundlegenden Wirkprinzipien des Antriebsstranges bekannt sind und somit eine Anpassungs- oder Variantenkonstruktion vorliegt. Demzufolge entfällt die Erarbeitung von funktionalen Zusammenhängen oder Wirkprinzipien [Pah07].

6.1 Klärung und Präzisierung der Aufgabenstellung

Die Entwicklung des Getriebes erfordert zunächst die inhaltliche Klärung der Aufgabenstellung. Die Anforderungen des Kunden sind in diesem Fall nur sehr rudimentär und bedürfen der weiteren Konkretisierung. Im günstigsten Fall führt diese Konkretisierung so weit, dass die quantitativen Merkmale in Form von Parametern schon direkt auf bestimmte CAD-Objekte abgebildet werden können. Um die Unterstützung des EPI Editors während der Aufgabenklärung aufzuzeigen, sind die Anforderungen an so ein Getriebe sind nur sehr grob gehalten. Tabelle 1 zeigt die geforderten Eigenschaften.

Bezeichnung	Qualitativ	Quantitativ
Eingangsleistung	Die Gesamtleistung des Getriebes	$\geq 1,5\text{MW}$
Antriebsdrehzahl	Die Drehzahl des Rotors	$= 15 \text{ min}^{-1}$
Abtriebsdrehzahl	Die Drehzahl des Generators	$= 1500 \text{ min}^{-1}$
Wirkungsgrad	Gesamtwirkungsgrad des Getriebes	$\geq 95\%$
Lebensdauer	Minimale Lebensdauer in Jahren	$\geq 20 \text{ Jahre}$

Tabelle 1: Kunden-Anforderungen an das Planetengetriebe

Die Aufgabenklärung wird vom EPI-Editor in zweierlei Hinsicht unterstützt. Zunächst können Checklisten verwaltet werden. Diese Fragenkataloge sollen helfen, auf Probleme bei der Entwicklung explizit hinzuweisen. Dabei unterteilen sich die Fragen in zwei

Kategorien. Fragen an den Kunden sind im Gespräch mit dem Kunden zu klären und interne Frage müssen im Entwicklungsteam geklärt werden.

Die zweite methodische Unterstützung besteht in der Verwaltung von einer hierarchischen Hauptmerkmalsliste. Die Bezeichnung und die Struktur sind dabei frei wählbar. So sind Einteilungen nach technischen Hauptmerkmalen oder nach organisatorischen Prozessmerkmalen denkbar. Abbildung 11 zeigt die Darstellung im Baum des EPI-Editors.

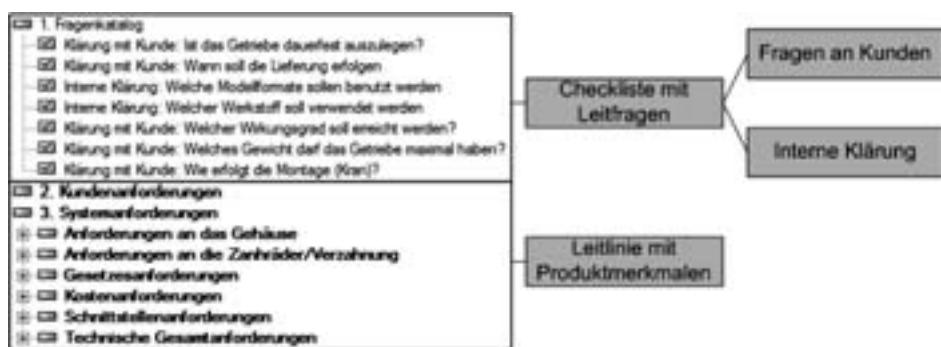


Abbildung 10: Methodische Unterstützung der Aufgabenklärung

Im Folgenden wird nun der Einsatz des EPI-Editors zur weiteren Konkretisierung dieser Entwicklungsaufgabe aufgezeigt

6.2 Definition und Konkretisierung von Anforderungen

Zunächst helfen die Leitfragen weitere Anforderungen zu bestimmen. Die Frage nach der Dauerfestigkeit des Getriebes führt beispielsweise zur Anforderung *Lebensdauer*. Diese Anforderung ist quantifizierbar und erhält somit einen Parameter Lebensdauer der zahlenmäßig mit 20 Jahren belegt wird. Daraus abgeleitet ergibt sich die Anforderung an die Dichtheit des Getriebes und einer hohen Stabilität des Gehäuses. Zur visuellen Unterstützung sind diese Konkretisierungsbeziehungen in dem Grafikfenster des EPI-Editors abgebildet. In Abbildung 14 wird diese Abfolge im rechten Teil dargestellt.

Die Frage nach einem zu verwendenden Werkstoff führt zu der Anforderung *Gussgerechte Gestaltung*. Dieser Anforderung sind Dokumente zur Recherche hinterlegt worden, um bei der Konstruktion das gussgerechte Gestalten zu ermöglichen. Der EPI-Editor hat für diese Dokumente einen Viewer, der auf Internet-Browser Technologie basiert, für alle gängigen Dokumentformate integriert. Abbildung 14 stellt diese Anforderung im linken Teil dar. Zusätzlich ist hier auch die Eingabemaske für Anforderungen abgebildet.

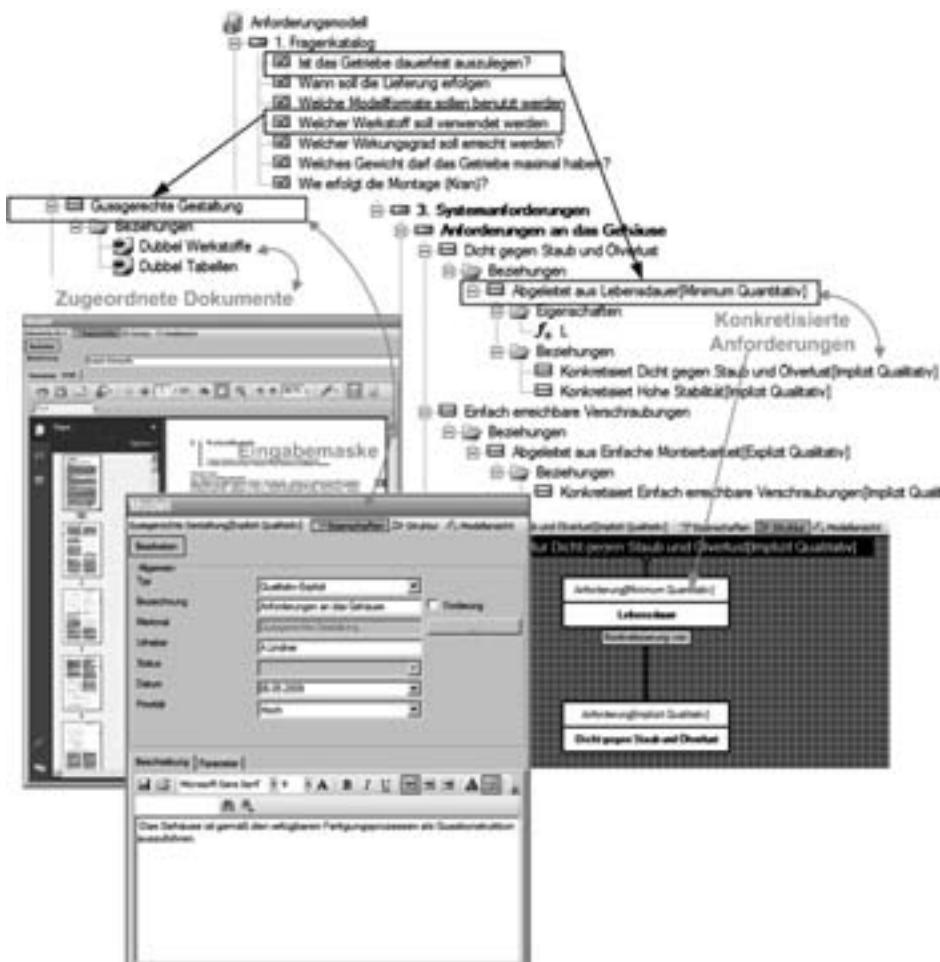


Abbildung 11: Anforderungen im EPI Editor

6.3 Gliederung in realisierbare Module

Die Anlage der Baustuktur erfolgt analog zur Anforderungsdefinition. Es können Baugruppen und Bauteile verwendet werden und zu einer hierarchischen Gesamtstruktur verbunden werden. Wenn dieser Vorgang abgeschlossen ist, erfolgt die Zuordnung von Anforderungen. Beispielsweise kann die Anforderung „*Gussgerechtes Gestalten*“ direkt mit den Gehäusekomponenten des Getriebes verbunden werden. Durch die applikationsübergreifende Vernetzung der jeweiligen Datensätze werden Anforderungen, die bereits mit anderen Anforderungen korrelieren, auch mit dem Bauelement verknüpft. So entsteht ein durchgängiges Modell. Abbildung 13 zeigt die Baustuktur des gesamten Getriebes und im rechten Teil werden die Komponenten der

Baugruppe „*Stirnradstufen*“ detailliert aufgelistet. Hier lässt sich erkennen, wie die Anforderungen mit den Komponenten verbunden sind.

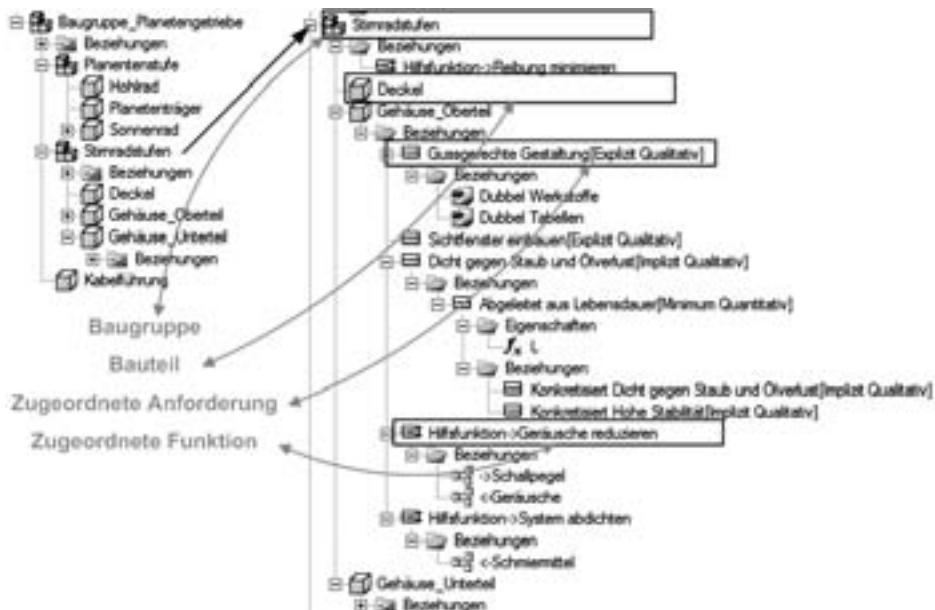


Abbildung 12: Baustrukturen im EPI-Editor

6.4 Verknüpfung mit CAD

Wenn die Baustuktur im EPI-Editor konzipiert und mit verschiedenen Modellelementen verknüpft wurde, kann die gesamte Baugruppe über einen Befehl erzeugt werden. Daneben gibt es auch die Möglichkeit der gezielten Erzeugung einzelner Bauteile oder Baugruppen. Somit kann der Konstrukteur gezielt bestimmte Modellteile konkretisieren. Bauteile, die bereits über ein zugeordnetes Modell verfügen, werden im Strukturbauum unterstrichen dargestellt und zusätzlich kann das Modell über den integrierten Viewer betrachtet werden. Der Befehl zur Modellerzeugung erstellt neben den Bauteildateien selbst, grafische Repräsentationen im Modelfenster des CAD-Systems. Hier werden dem Konstrukteur dann gezielt Informationen zu Anforderungen und verknüpften Dokumenten angezeigt. Abbildung 14 zeigt diese Visualisierung im Strukturbauum und im Grafikfenster. Hier sind drei verschiedene Anforderungen mit dem CAD-Modell verknüpft worden.

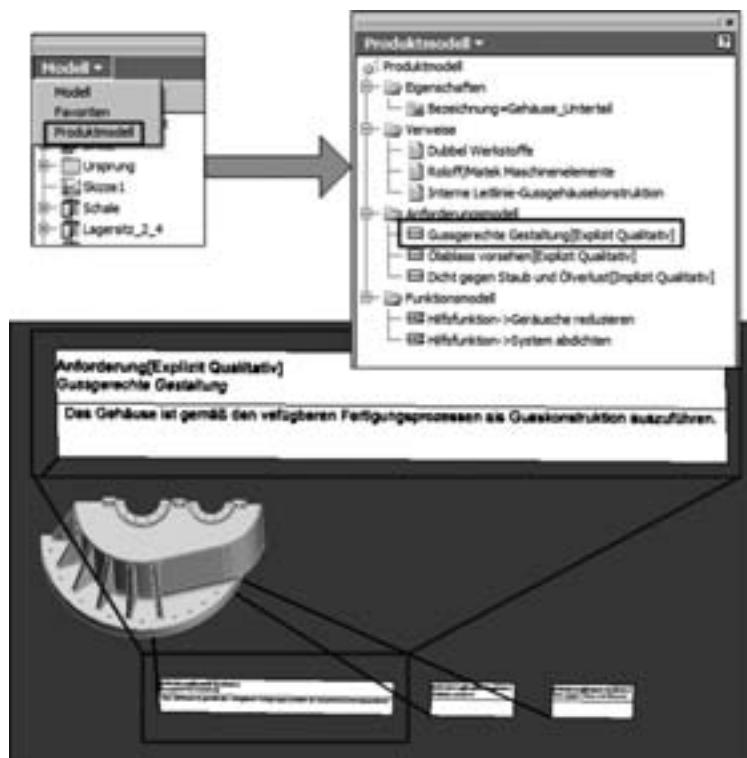


Abbildung 13: Anforderungsvisualisierung im CAD-System

Die Verbindung zwischen CAD-Dokument und EPI-Datenbank ist bidirektional, so dass der Konstrukteur direkt aus dem CAD-System einzelne Datensätze aufrufen kann und somit Anforderungen und Dokumente recherchieren kann. Weiterhin besteht die Möglichkeit der Ableitung von Parametern, die sich aus EPI-Objekten ergeben. Demzufolge können automatisiert Parametertabellen angelegt werden. Abbildung 15 zeigt die grundsätzliche Abfolge des Verknüpfens von CAD- und EPI-Daten. Zunächst wird über den Befehl „*CAD-Verknüpfung ableiten*“ ein Dialog gestartet, der die Parameter der EPI-Datenbank denen des CAD-Modells gegenüberstellt. Nach der Auswahl einer EPI-CAD Kombination kann dieses Paar dann verbunden werden. Diese neu angelegte Beziehung wird dann sowohl im CAD-Dokument als auch in der EPI Datenbank abgelegt. In dem hier gezeigten Beispiel wird eine Bereichsanforderung bezüglich der Wanddicke des Getriebegehäuses definiert. Daraufhin führt die EPI-Datenbank zwei Parameter, die mit dieser Anforderung verknüpft sind. Über den Auswahldialog kann dann einer dieser Parameter mit einem Modellparameter verknüpft werden.

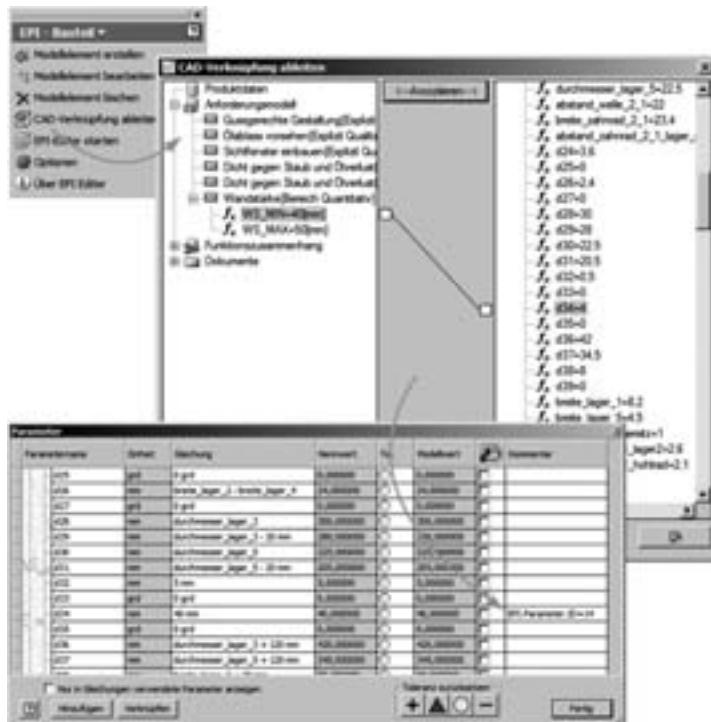


Abbildung 14: Verknüpfen von CAD und EPI

Der Benutzer kann dann später in der Parametertabelle von AI erkennen, dass ein Parameter mit Anforderungen verknüpft wurde. Über den EPI-Hauptdialog können dann wiederum die hinterlegten Informationen abgefragt werden.

7 Zusammenfassung

Die virtuelle Produktentwicklung zeichnet sich durch eine durchgehende Integration verschiedener Produktinformationen und -modelle aus, wobei dem Gestaltmodell in Form von CAD-Daten eine zentrale Bedeutung zukommt. Ein wesentliches Ziel der Produktentwicklungsprozesse ist die fortlaufende Reduzierung der Komplexität ohne wichtige Merkmale aus den Augen zu verlieren. Um dieses Ziel zu erreichen wird das Produktmodell in viele Partialmodelle zerlegt, die getrennt voneinander betrachtet werden. Diese Arbeit beschäftigte sich mit der Fragestellung in welcher Art und Weise das Anforderungsmodell in ein CAD-System implementiert werden kann, um so das Produktwissen, das in diesen Modellen abgebildet wird, den Konstrukteuren vollständig zugänglich zu machen.

Zu diesem Zweck wurde ein Werkzeug entwickelt, das die Funktionalität eines herkömmlichen CAD-Systems in der Art erweitert, dass eine Modellierung und Visualisierung von Anforderungen und deren Beziehungen möglich wird. Weiterhin

können Parameter des CAD-Systems mit technischen Werten, die aus Anforderungen entnommen werden können, verknüpft werden. Durch diese neuen Funktionen kann der Konstrukteur jederzeit aus dem CAD-System heraus, wesentliche Anforderungen überprüfen, was zu einem qualitativ höherwertigen Modell und somit zu einem effizienterem Konstruktionsprozess führt.

Literaturverzeichnis

- [Abr07] Abramovici, M.; Meimann, V.: *Ein Ansatz zur Erhöhung der Stabilität der CAD-Modelle für die Verbesserung der Modell-Wiederverwendung in der Virtuellen Produktentwicklung*, In: Tagungsband zum 5. Kolloquium Konstruktionstechnik, Eigendruck, Dresden, 2007
- [Cha06] Chasiotis, C.: *Prozessbegleitende Wissensdokumentation und integrierte Wissensvisualisierung in der Digitalen Produktentwicklung*, Dissertation Shaker, Aachen, 2006
- [Con08] Conrad, K.-J.: *Taschenbuch der Konstruktionstechnik*, Carl Hanser Verlag, München Wien, 2008
- [Ehr06] Ehrlenspiel, K.: *Integrierte Produktentwicklung*, Carl Hanser Verlag, München Wien, 2006
- [Ehr07] Ehrlenspiel, K.; Kiewert, A.; Lindemann, U.: *Kostengünstig Entwickeln und Konstruieren*, Axel Springer Verlag, Berlin Heidelberg, 2006
- [Gho08] Ghoffrani, M.: *Entwicklung und Einführung eines flexiblen Softwaresystems zur Konfigurierung virtueller Produkte*, Dissertation, Shaker, Aachen, 2008
- [Lin08] Lindemann, U.: *Konzeptentwicklung und Gestaltung technischer Produkte*, Axel Springer Verlag, Berlin, 2008
- [Pah07] Pahl, G.; Beitz, W.; Feldhusen, J.; Grote, K.-H.: *Konstruktionslehre*, Axel Springer Verlag, Berlin, 2007
- [Spu97] Spur, G.; Krause, F.-L.: *Das virtuelle Produkt*, Carl Hansen Verlag, München, 1997
- [Sta73] Stachowiak, F.: *Allgemeine Modelltheorie*, Axel Springer Verlag, Berlin, 1973
- [Suh93] Suhm, A.: *Produktmodellierung in wissensbasierten Konstruktionssystemen auf der Basis von Lösungsmustern*, Dissertation, Shaker, Karlsruhe, 1993
- [Szy01] Szykman, S.; Sriram, R.D.; Regli, W.C.: *The Role of Knowledge in Next-Generation Product-Development Systems*, In: Journal of Computing and Information Science in Engineering, 1/2001
- [Vaj08] Vajna, S.; Weber, C.; Bley, H.; Zeman, K.: *CAx für Ingenieure*, 2. Aufl., Axel Springer Verlag, Berlin Heidelberg, 2008
- [VDI93] VDI-Richtlinie 2221: *Methodik für das Entwickeln und Konstruieren technischer Systeme und Produkte*, Beuth Verlag, Düsseldorf, 1993
- [Web96] Weber, C.: *What is a Feature and What is ist Use?*, In: Results of FEMEX Working Group I. 29th International Symposium on Automotive Technology and Automation, Florenz, 1996
- [Web05] Weber, C.: *CPM/PDD – An Extended Theoretical Approach to Modelling Products and Product Development Processes*, In: Bley, H.; Jansen, H.; Krause, F.-L.; Shpitalni, M. (Hrsg.): *Proceedings Of The 2nd German-Israeli Symposium*, Stuttgart, 2005
- [Web09] Weber, C.: *THEORY OF TECHNICAL SYSTEMS (TTS) – EXISTING APPROACHES AND CHALLENGES*, In: Bergendahl, M.N.; Grimheden, M.; Leifer, L.: *Proceedings of ICED'09*, Stanford, 2009

Internetquellen

[ADS09] Offizielle Webseite von *Autodesk Inc.*, URL:<http://www.autodesk.de>, Letzter Zugriff: Dezember 2009

GI-Edition Lecture Notes in Informatics

- | | | | |
|------|---|------|---|
| P-1 | Gregor Engels, Andreas Oberweis, Albert Zündorf (Hrsg.): Modellierung 2001. | P-18 | Elmar J. Sinz, Markus Plaha (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2002. |
| P-2 | Mikhail Godlevsky, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications, ISTA'2001. | P-19 | Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund. |
| P-3 | Ana M. Moreno, Reind P. van de Riet (Hrsg.): Applications of Natural Language to Information Systems, NLDB'2001. | P-20 | Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund (Ergänzungsband). |
| P-4 | H. Wörn, J. Mühlung, C. Vahl, H.-P. Meinzer (Hrsg.): Rechner- und sensor-gestützte Chirurgie; Workshop des SFB 414. | P-21 | Jörg Desel, Mathias Weske (Hrsg.): Promise 2002: Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen. |
| P-5 | Andy Schürr (Hg.): OMER – Object-Oriented Modeling of Embedded Real-Time Systems. | P-22 | Sigrid Schubert, Johannes Magenheim, Peter Hubwieser, Torsten Brinda (Hrsg.): Forschungsbeiträge zur "Didaktik der Informatik" – Theorie, Praxis, Evaluation. |
| P-6 | Hans-Jürgen Appelrath, Rolf Beyer, Uwe Marquardt, Heinrich C. Mayr, Claudia Steinberger (Hrsg.): Unternehmen Hochschule, UH'2001. | P-23 | Thorsten Spitta, Jens Borchers, Harry M. Sneed (Hrsg.): Software Management 2002 – Fortschritt durch Beständigkeit |
| P-7 | Andy Evans, Robert France, Ana Moreira, Bernhard Rumpe (Hrsg.): Practical UML-Based Rigorous Development Methods – Countering or Integrating the extremists, pUML'2001. | P-24 | Rainer Eckstein, Robert Tolksdorf (Hrsg.): XMIDX 2003 – XML-Technologien für Middleware – Middleware für XML-Anwendungen |
| P-8 | Reinhard Keil-Slawik, Johannes Magenheim (Hrsg.): Informatikunterricht und Medienbildung, INFOS'2001. | P-25 | Key Poussotchi, Klaus Turowski (Hrsg.): Mobile Commerce – Anwendungen und Perspektiven – 3. Workshop Mobile Commerce, Universität Augsburg, 04.02.2003 |
| P-9 | Jan von Knop, Wilhelm Haverkamp (Hrsg.): Innovative Anwendungen in Kommunikationsnetzen, 15. DFN Arbeits-tagung. | P-26 | Gerhard Weikum, Harald Schöning, Erhard Rahm (Hrsg.): BTW 2003: Datenbanksysteme für Business, Technologie und Web |
| P-10 | Mirjam Minor, Steffen Staab (Hrsg.): 1st German Workshop on Experience Management: Sharing Experiences about the Sharing Experience. | P-27 | Michael Kroll, Hans-Gerd Lipinski, Kay Melzer (Hrsg.): Mobiles Computing in der Medizin |
| P-11 | Michael Weber, Frank Kargl (Hrsg.): Mobile Ad-Hoc Netzwerke, WMAN 2002. | P-28 | Ulrich Reimer, Andreas Abecker, Steffen Staab, Gerd Stumme (Hrsg.): WM 2003: Professionelles Wissensmanagement – Erfahrungen und Visionen |
| P-12 | Martin Glinz, Günther Müller-Luschnat (Hrsg.): Modellierung 2002. | P-29 | Antje Düsterhöft, Bernhard Thalheim (Eds.): NLDB'2003: Natural Language Processing and Information Systems |
| P-13 | Jan von Knop, Peter Schirmacher and Viljan Mahni (Hrsg.): The Changing Universities – The Role of Technology. | P-30 | Mikhail Godlevsky, Stephen Liddle, Heinrich C. Mayr (Eds.): Information Systems Technology and its Applications |
| P-14 | Robert Tolksdorf, Rainer Eckstein (Hrsg.): XML-Technologien für das Semantic Web – XSW 2002. | P-31 | Arslan Brömmе, Christoph Busch (Eds.): BIOSIG 2003: Biometrics and Electronic Signatures |
| P-15 | Hans-Bernd Bludau, Andreas Koop (Hrsg.): Mobile Computing in Medicine. | | |
| P-16 | J. Felix Hampe, Gerhard Schwabe (Hrsg.): Mobile and Collaborative Business 2002. | | |
| P-17 | Jan von Knop, Wilhelm Haverkamp (Hrsg.): Zukunft der Netze – Die Verletzbarkeit meistern, 16. DFN Arbeitstagung. | | |

- | | | | |
|------|--|------|---|
| P-32 | Peter Hubwieser (Hrsg.): Informatische Fachkonzepte im Unterricht – INFOS 2003 | P-48 | Anatoly Doroshenko, Terry Halpin, Stephen W. Liddle, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications |
| P-33 | Andreas Geyer-Schulz, Alfred Taudes (Hrsg.): Informationswirtschaft: Ein Sektor mit Zukunft | P-49 | G. Schiefer, P. Wagner, M. Morgenstern, U. Ricket (Hrsg.): Integration und Datensicherheit – Anforderungen, Konflikte und Perspektiven |
| P-34 | Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 1) | P-50 | Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 1) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm |
| P-35 | Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 2) | P-51 | Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 2) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm |
| P-36 | Rüdiger Grimm, Hubert B. Keller, Kai Rannenberg (Hrsg.): Informatik 2003 – Mit Sicherheit Informatik | P-52 | Gregor Engels, Silke Seehusen (Hrsg.): DELFI 2004 – Tagungsband der 2. e-Learning Fachtagung Informatik |
| P-37 | Arndt Bode, Jörg Desel, Sabine Rathmayer, Martin Wessner (Hrsg.): DeLFI 2003: e-Learning Fachtagung Informatik | P-53 | Robert Giegerich, Jens Stoye (Hrsg.): German Conference on Bioinformatics – GCB 2004 |
| P-38 | E.J. Sinz, M. Plaha, P. Neckel (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2003 | P-54 | Jens Borchers, Ralf Kneuper (Hrsg.): Softwaremanagement 2004 – Outsourcing und Integration |
| P-39 | Jens Nedon, Sandra Frings, Oliver Göbel (Hrsg.): IT-Incident Management & IT-Forensics – IMF 2003 | P-55 | Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): E-Science und Grid Ad-hoc-Netze Medienintegration |
| P-40 | Michael Rebstock (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2004 | P-56 | Fernand Feltz, Andreas Oberweis, Benoit Otjacques (Hrsg.): EMISA 2004 – Informationssysteme im E-Business und E-Government |
| P-41 | Uwe Brinkschulte, Jürgen Becker, Dietmar Fey, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle, Thomas Runkler (Edts.): ARCS 2004 – Organic and Pervasive Computing | P-57 | Klaus Turowski (Hrsg.): Architekturen, Komponenten, Anwendungen |
| P-42 | Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Economy – Transaktionen und Prozesse, Anwendungen und Dienste | P-58 | Sami Beydeda, Volker Gruhn, Johannes Mayer, Ralf Reussner, Franz Schwiegert (Hrsg.): Testing of Component-Based Systems and Software Quality |
| P-43 | Birgitta König-Ries, Michael Klein, Philipp Obreiter (Hrsg.): Persistence, Scalability, Transactions – Database Mechanisms for Mobile Applications | P-59 | J. Felix Hampe, Franz Lehner, Key Pousttchi, Kai Ranneberg, Klaus Turowski (Hrsg.): Mobile Business – Processes, Platforms, Payments |
| P-44 | Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): Security, E-Learning, E-Services | P-60 | Steffen Friedrich (Hrsg.): Unterrichtskonzepte für inforrmatische Bildung |
| P-45 | Bernhard Rumpe, Wolfgang Hesse (Hrsg.): Modellierung 2004 | P-61 | Paul Müller, Reinhard Gotzhein, Jens B. Schmitt (Hrsg.): Kommunikation in verteilten Systemen |
| P-46 | Ulrich Flegel, Michael Meier (Hrsg.): Detection of Intrusions of Malware & Vulnerability Assessment | P-62 | Federrath, Hannes (Hrsg.): „Sicherheit 2005“ – Sicherheit – Schutz und Zuverlässigkeit |
| P-47 | Alexander Prosser, Robert Krimmer (Hrsg.): Electronic Voting in Europe – Technology, Law, Politics and Society | P-63 | Roland Kaschek, Heinrich C. Mayr, Stephen Liddle (Hrsg.): Information Systems – Technology and ist Applications |

- | | | | |
|------|--|------|---|
| P-64 | Peter Liggesmeyer, Klaus Pohl, Michael Goedicke (Hrsg.): Software Engineering 2005 | P-80 | Mareike Schoop, Christian Huemer, Michael Rebstock, Martin Bichler (Hrsg.): Service-Oriented Electronic Commerce |
| P-65 | Gottfried Vossen, Frank Leymann, Peter Lockemann, Wolffried Stucky (Hrsg.): Datenbanksysteme in Business, Technologie und Web | P-81 | Wolfgang Karl, Jürgen Becker, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle (Hrsg.): ARCS'06 |
| P-66 | Jörg M. Haake, Ulrike Lucke, Djamshid Tavangarian (Hrsg.): DeLFI 2005: 3. deutsche e-Learning Fachtagung Informatik | P-82 | Heinrich C. Mayr, Ruth Breu (Hrsg.): Modellierung 2006 |
| P-67 | Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 1) | P-83 | Daniel Huson, Oliver Kohlbacher, Andrei Lupas, Kay Nieselt and Andreas Zell (eds.): German Conference on Bioinformatics |
| P-68 | Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 2) | P-84 | Dimitris Karagiannis, Heinrich C. Mayr, (Hrsg.): Information Systems Technology and its Applications |
| P-69 | Robert Hirschfeld, Ryszard Kowalczyk, Andreas Polze, Matthias Weske (Hrsg.): NODE 2005, GSEM 2005 | P-85 | Witold Abramowicz, Heinrich C. Mayr, (Hrsg.): Business Information Systems |
| P-70 | Klaus Turowski, Johannes-Maria Zaha (Hrsg.): Component-oriented Enterprise Application (COAE 2005) | P-86 | Robert Krimmer (Ed.): Electronic Voting 2006 |
| P-71 | Andrew Torda, Stefan Kurz, Matthias Rarey (Hrsg.): German Conference on Bioinformatics 2005 | P-87 | Max Mühlhäuser, Guido Rößling, Ralf Steinmetz (Hrsg.): DELFI 2006: 4. e-Learning Fachtagung Informatik |
| P-72 | Klaus P. Jantke, Klaus-Peter Fähnrich, Wolfgang S. Wittig (Hrsg.): Marktplatz Internet: Von e-Learning bis e-Payment | P-88 | Robert Hirschfeld, Andreas Polze, Ryszard Kowalczyk (Hrsg.): NODE 2006, GSEM 2006 |
| P-73 | Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): "Heute schon das Morgen sehen" | P-90 | Joachim Schelp, Robert Winter, Ulrich Frank, Bodo Rieger, Klaus Turowski (Hrsg.): Integration, Informationslogistik und Architektur |
| P-74 | Christopher Wolf, Stefan Lucks, Po-Wah Yau (Hrsg.): WEWoRC 2005 – Western European Workshop on Research in Cryptology | P-91 | Henrik Stormer, Andreas Meier, Michael Schumacher (Eds.): European Conference on eHealth 2006 |
| P-75 | Jörg Desel, Ulrich Frank (Hrsg.): Enterprise Modelling and Information Systems Architecture | P-92 | Fernand Feltz, Benoît Otjacques, Andreas Oberweis, Nicolas Poussing (Eds.): AIM 2006 |
| P-76 | Thomas Kirste, Birgitta König-Ries, Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Informationssysteme – Potentiale, Hindernisse, Einsatz | P-93 | Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 1 |
| P-77 | Jana Dittmann (Hrsg.): SICHERHEIT 2006 | P-94 | Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 2 |
| P-78 | K.-O. Wenkel, P. Wagner, M. Morgenthern, K. Luzi, P. Eisermann (Hrsg.): Land- und Ernährungswirtschaft im Wandel | P-95 | Matthias Weske, Markus Nütgens (Eds.): EMISA 2005: Methoden, Konzepte und Technologien für die Entwicklung von dienstbasierten Informationssystemen |
| P-79 | Bettina Biel, Matthias Book, Volker Gruhn (Hrsg.): Softwareengineering 2006 | P-96 | Saartje Brockmans, Jürgen Jung, York Sure (Eds.): Meta-Modelling and Ontologies |
| | | P-97 | Oliver Göbel, Dirk Schadt, Sandra Frings, Hardo Hase, Detlef Günther, Jens Nedon (Eds.): IT-Incident Management & IT-Forensics – IMF 2006 |

- P-98 Hans Brandt-Pook, Werner Simonsmeier und Thorsten Spitta (Hrsg.): Beratung in der Softwareentwicklung – Modelle, Methoden, Best Practices
- P-99 Andreas Schwill, Carsten Schulte, Marco Thomas (Hrsg.): Didaktik der Informatik
- P-100 Peter Forbrig, Günter Siegel, Markus Schneider (Hrsg.): HDI 2006: Hochschuldidaktik der Informatik
- P-101 Stefan Böttiger, Ludwig Theuvsen, Susanne Rank, Marlies Morgenstern (Hrsg.): Agrarinformatik im Spannungsfeld zwischen Regionalisierung und globalen Wertschöpfungsketten
- P-102 Otto Spaniol (Eds.): Mobile Services and Personalized Environments
- P-103 Alfons Kemper, Harald Schöning, Thomas Rose, Matthias Jarke, Thomas Seidl, Christoph Quix, Christoph Brochhaus (Hrsg.): Datenbanksysteme in Business, Technologie und Web (BTW 2007)
- P-104 Birgitta König-Ries, Franz Lehner, Rainer Malaka, Can Türker (Hrsg.) MMS 2007: Mobilität und mobile Informationssysteme
- P-105 Wolf-Gideon Bleek, Jörg Raasch, Heinz Züllighoven (Hrsg.) Software Engineering 2007
- P-106 Wolf-Gideon Bleek, Henning Schwentner, Heinz Züllighoven (Hrsg.) Software Engineering 2007 – Beiträge zu den Workshops
- P-107 Heinrich C. Mayr, Dimitris Karagiannis (eds.) Information Systems Technology and its Applications
- P-108 Arslan Brömmel, Christoph Busch, Detlef Hühnlein (eds.) BIOSIG 2007: Biometrics and Electronic Signatures
- P-109 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 1
- P-110 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 2
- P-111 Christian Eibl, Johannes Magenheim, Sigrid Schubert, Martin Wessner (Hrsg.) DeLF 2007: 5. e-Learning Fachtagung Informatik
- P-112 Sigrid Schubert (Hrsg.) Didaktik der Informatik in Theorie und Praxis
- P-113 Sören Auer, Christian Bizer, Claudia Müller, Anna V. Zhdanova (Eds.) The Social Semantic Web 2007 Proceedings of the 1st Conference on Social Semantic Web (CSSW)
- P-114 Sandra Frings, Oliver Göbel, Detlef Günther, Hardo G. Hase, Jens Nedon, Dirk Schadt, Arslan Brömmel (Eds.) IMF2007 IT-incident management & IT-forensics Proceedings of the 3rd International Conference on IT-Incident Management & IT-Forensics
- P-115 Claudia Falter, Alexander Schliep, Joachim Selbig, Martin Vingron and Dirk Walther (Eds.) German conference on bioinformatics GCB 2007
- P-116 Witold Abramowicz, Leszek Maciszek (Eds.) Business Process and Services Computing 1st International Working Conference on Business Process and Services Computing BPSC 2007
- P-117 Ryszard Kowalczyk (Ed.) Grid service engineering and management The 4th International Conference on Grid Service Engineering and Management GSEM 2007
- P-118 Andreas Hein, Wilfried Thoben, Hans-Jürgen Appelrath, Peter Jensch (Eds.) European Conference on ehealth 2007
- P-119 Manfred Reichert, Stefan Strecker, Klaus Turowski (Eds.) Enterprise Modelling and Information Systems Architectures Concepts and Applications
- P-120 Adam Pawlak, Kurt Sandkuhl, Wojciech Cholewa, Leandro Soares Indrusiak (Eds.) Coordination of Collaborative Engineering - State of the Art and Future Challenges
- P-121 Korbinian Herrmann, Bernd Bruegge (Hrsg.) Software Engineering 2008 Fachtagung des GI-Fachbereichs Softwaretechnik
- P-122 Walid Maalej, Bernd Bruegge (Hrsg.) Software Engineering 2008 - Workshopband Fachtagung des GI-Fachbereichs Softwaretechnik

- P-123 Michael H. Breitner, Martin Breunig, Elgar Fleisch, Ley Poustchi, Klaus Turowski (Hrsg.) Mobile und Ubiquitäre Informationssysteme – Technologien, Prozesse, Marktfähigkeit Proceedings zur 3. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2008)
- P-124 Wolfgang E. Nagel, Rolf Hoffmann, Andreas Koch (Eds.) 9th Workshop on Parallel Systems and Algorithms (PASA) Workshop of the GI/ITG Speciel Interest Groups PARS and PARVA
- P-125 Rolf A.E. Müller, Hans-H. Sundermeier, Ludwig Theuvßen, Stephanie Schütze, Marlies Morgenstern (Hrsg.) Unternehmens-IT: Führungsinstrument oder Verwaltungsbürde Referate der 28. GIL Jahrestagung
- P-126 Rainer Gimlich, Uwe Kaiser, Jochen Quante, Andreas Winter (Hrsg.) 10th Workshop Software Reengineering (WSR 2008)
- P-127 Thomas Kühne, Wolfgang Reisig, Friedrich Steimann (Hrsg.) Modellierung 2008
- P-128 Ammar Alkassar, Jörg Siekmann (Hrsg.) Sicherheit 2008 Sicherheit, Schutz und Zuverlässigkeit Beiträge der 4. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI) 2.-4. April 2008 Saarbrücken, Germany
- P-129 Wolfgang Hesse, Andreas Oberweis (Eds.) Sigsand-Europe 2008 Proceedings of the Third AIS SIGSAND European Symposium on Analysis, Design, Use and Societal Impact of Information Systems
- P-130 Paul Müller, Bernhard Neumair, Gabi Dreßl Rodosek (Hrsg.) 1. DFN-Forum Kommunikations-technologien Beiträge der Fachtagung
- P-131 Robert Krimmer, Rüdiger Grimm (Eds.) 3rd International Conference on Electronic Voting 2008 Co-organized by Council of Europe, Gesellschaft für Informatik and E-Voting.CC
- P-132 Silke Seehusen, Ulrike Lucke, Stefan Fischer (Hrsg.) DeLF1 2008: Die 6. e-Learning Fachtagung Informatik
- P-133 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.) INFORMATIK 2008 Beherrschbare Systeme – dank Informatik Band 1
- P-134 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.) INFORMATIK 2008 Beherrschbare Systeme – dank Informatik Band 2
- P-135 Torsten Brinda, Michael Fothe, Peter Hubwieser, Kirsten Schlüter (Hrsg.) Didaktik der Informatik – Aktuelle Forschungsergebnisse
- P-136 Andreas Beyer, Michael Schroeder (Eds.) German Conference on Bioinformatics GCB 2008
- P-137 Arslan Brömme, Christoph Busch, Detlef Hühnlein (Eds.) BIOSIG 2008: Biometrics and Electronic Signatures
- P-138 Barbara Dinter, Robert Winter, Peter Chamoni, Norbert Gronau, Klaus Turowski (Hrsg.) Synergien durch Integration und Informationslogistik Proceedings zur DW2008
- P-139 Georg Herzwurm, Martin Mikusz (Hrsg.) Industrialisierung des Software-Managements Fachtagung des GI-Fachausschusses Management der Anwendungsentwicklung und -wartung im Fachbereich Wirtschaftsinformatik
- P-140 Oliver Göbel, Sandra Frings, Detlef Günther, Jens Nedon, Dirk Schadt (Eds.) IMF 2008 - IT Incident Management & IT Forensics
- P-141 Peter Loos, Markus Nüttgens, Klaus Turowski, Dirk Werth (Hrsg.) Modellierung betrieblicher Informationssysteme (MobiIS 2008) Modellierung zwischen SOA und Compliance Management
- P-142 R. Bill, P. Korduan, L. Theuvßen, M. Morgenstern (Hrsg.) Anforderungen an die Agrarinformatik durch Globalisierung und Klimaveränderung
- P-143 Peter Liggesmeyer, Gregor Engels, Jürgen Münch, Jörg Dörr, Norman Riegel (Hrsg.) Software Engineering 2009 Fachtagung des GI-Fachbereichs Softwaretechnik

- P-144 Johann-Christoph Freytag, Thomas Ruf, Wolfgang Lehner, Gottfried Vossen (Hrsg.) Datenbanksysteme in Business, Technologie und Web (BTW)
- P-145 Knut Hinkelmann, Holger Wache (Eds.) WM2009: 5th Conference on Professional Knowledge Management
- P-146 Markus Bick, Martin Breunig, Hagen Höpfner (Hrsg.) Mobile und Ubiquitäre Informationssysteme – Entwicklung, Implementierung und Anwendung 4. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2009)
- P-147 Witold Abramowicz, Leszek Maciaszek, Ryszard Kowalczyk, Andreas Speck (Eds.) Business Process, Services Computing and Intelligent Service Management BPSC 2009 · ISM 2009 · YRW-MBP 2009
- P-148 Christian Erfurth, Gerald Eichler, Volkmar Schau (Eds.) 9th International Conference on Innovative Internet Community Systems I²CS 2009
- P-149 Paul Müller, Bernhard Neumair, Gabi Dreßel Rodosek (Hrsg.) 2. DFN-Forum Kommunikationstechnologien Beiträge der Fachtagung
- P-150 Jürgen Münch, Peter Liggesmeyer (Hrsg.) Software Engineering 2009 - Workshopband
- P-151 Armin Heinzl, Peter Dadam, Stefan Kirn, Peter Lockemann (Eds.) PRIMIUM Process Innovation for Enterprise Software
- P-152 Jan Mendling, Stefanie Rinderle-Ma, Werner Esswein (Eds.) Enterprise Modelling and Information Systems Architectures Proceedings of the 3rd Int'l Workshop EMISA 2009
- P-153 Andreas Schwill, Nicolas Apostolopoulos (Hrsg.) Lernen im Digitalen Zeitalter DELFI 2009 – Die 7. E-Learning Fachtagung Informatik
- P-154 Stefan Fischer, Erik Maehle Rüdiger Reischuk (Hrsg.) INFORMATIK 2009 Im Focus das Leben
- P-155 Arslan Brömmel, Christoph Busch, Detlef Hühlein (Eds.) BIOSIG 2009: Biometrics and Electronic Signatures Proceedings of the Special Interest Group on Biometrics and Electronic Signatures
- P-156 Bernhard Koerber (Hrsg.) Zukunft braucht Herkunft 25 Jahre »INFOS – Informatik und Schule«
- P-157 Ivo Grosse, Steffen Neumann, Stefan Posch, Falk Schreiber, Peter Stadler (Eds.) German Conference on Bioinformatics 2009
- P-158 W. Claupein, L. Theuvsen, A. Kämpf, M. Morgenstern (Hrsg.) Precision Agriculture Reloaded – Informationsgestützte Landwirtschaft
- P-159 Gregor Engels, Markus Luckey, Wilhelm Schäfer (Hrsg.) Software Engineering 2010
- P-160 Gregor Engels, Markus Luckey, Alexander Pretschner, Ralf Reussner (Hrsg.) Software Engineering 2010 – Workshopband (inkl. Doktorandensymposium)
- P-161 Gregor Engels, Dimitris Karagiannis Heinrich C. Mayr (Hrsg.) Modellierung 2010
- P-162 Maria A. Wimmer, Uwe Brinkhoff, Siegfried Kaiser, Dagmar Lück-Schneider, Erich Schweighofer, Andreas Wiebe (Hrsg.) Vernetzte IT für einen effektiven Staat Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI) 2010
- P-163 Markus Bick, Stefan Eulgem, Elgar Fleisch, J. Felix Hampe, Birgitta König-Ries, Franz Lehner, Key Pousttchi, Kai Rannenberg (Hrsg.) Mobile und Ubiquitäre Informationssysteme Technologien, Anwendungen und Dienste zur Unterstützung von mobiler Kollaboration
- P-165 Gerald Eichler, Peter Kropf, Ulrike Lechner, Phayung Meesad, Herwig Unger (Eds.) 10th International Conference on Innovative Internet Community Systems (I²CS) – Jubilee Edition 2010 –

- P-166 Paul Müller, Bernhard Neumair,
Gabi Dreßl Rodosek (Hrsg.)
3. DFN-Forum Kommunikationstechnologien
Beiträge der Fachtagung
- P-167 Robert Krimmer, Rüdiger Grimm (Eds.)
4th International Conference on
Electronic Voting 2010
co-organized by the Council of Europe,
Gesellschaft für Informatik and
E-Voting.CC
- P-168 Ira Diethelm, Christina Dörge,
Claudia Hildebrandt,
Carsten Schulte (Hrsg.)
Didaktik der Informatik
Möglichkeiten empirischer
Forschungsmethoden und Perspektiven
der Fachdidaktik
- P-169 Michael Kerres, Nadine Ojstersek
Ulrik Schroeder, Ulrich Hoppe (Hrsg.)
DeLF1 2010 - 8. Tagung
der Fachgruppe E-Learning
der Gesellschaft für Informatik e.V.
- P-170 Felix C. Freiling (Hrsg.)
Sicherheit 2010
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 5. Jahrestagung des
Fachbereichs Sicherheit der Gesellschaft
für Informatik e.V. (GI)
5.–7. Oktober 2010 Berlin
- P-171 Werner Esswein, Klaus Turowski,
Martin Juhrisch (Hrsg.)
Modellierung betrieblicher
Informationssysteme (MobiIS 2010)
Modellgestütztes Management

The titles can be purchased at:

Köllen Druck + Verlag GmbH

Ernst-Robert-Curtius-Str. 14 · D-53117 Bonn

Fax: +49 (0)228/9898222

E-Mail: druckverlag@koellen.de

