

Forschungsberichte



Björn Ebel

**Modellierung von Zielsystemen
in der interdisziplinären Produktentstehung**

Modeling of System of Objectives
in Interdisciplinary Product Engineering

Band 85

Herausgeber: o. Prof. Dr.-Ing. Dr. h.c. A. Albers

Copyright: IPEK ■ Institut für Produktentwicklung, 2015
Karlsruher Institut für Technologie (KIT)
Universität des Landes Baden-Württemberg und
nationales Forschungszentrum in der Helmholtz-Gemeinschaft

Alle Rechte vorbehalten

Druck: Stolzenberger Druck und Werbung GmbH & Co. KG, Leimen
06224-7697915

ISSN 1615-8113

Modellierung von Zielsystemen in der interdisziplinären Produktentstehung

Zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
der Fakultät für Maschinenbau
Karlsruher Institut für Technologie (KIT)

genehmigte
Dissertation

von

Dipl.-Ing. Björn Ebel
aus Freiburg im Breisgau

Tag der mündlichen Prüfung: 30. April 2015
Hauptreferent: Prof. Dr.-Ing. Dr. h.c. A. Albers
Korreferent: Prof. Dr.-Ing. J. Gausemeier

Vorwort des Herausgebers

Wissen ist einer der entscheidenden Faktoren in den Volkswirtschaften unserer Zeit. Der Unternehmenserfolg wird in der Zukunft mehr denn je davon abhängen, wie schnell ein Unternehmen neues Wissen aufnehmen, zugänglich machen und verwerten kann. Die Aufgabe eines Universitätsinstitutes ist es, hier einen wesentlichen Beitrag zu leisten. In den Forschungsarbeiten wird ständig Wissen generiert. Dieses kann aber nur wirksam und für die Gemeinschaft nutzbar werden, wenn es in geeigneter Form kommuniziert wird. Diese Schriftenreihe dient als eine Plattform zum Transfer und macht damit das Wissenspotenzial aus aktuellen Forschungsarbeiten am IPEK – Institut für Produktentwicklung Karlsruhe (ehemals: Institut für Maschinenkonstruktionslehre und Krafftfahrzeugbau) verfügbar.

Die Forschungsfelder des Institutes sind die methodische Entwicklung und das Entwicklungsmanagement, die rechnergestützte Optimierung von hochbelasteten Strukturen und Systemen, die Antriebstechnik mit einem Schwerpunkt auf den Gebieten Antriebsstrang-Engineering und Tribologie von Lager- und Funktionsreibsystemen, die Mikrosystemtechnik mit dem Focus auf die zugehörigen Entwicklungsprozesse sowie die Mechatronik. Die Forschungsberichte werden aus allen diesen Gebieten Beiträge zur wissenschaftlichen Fortentwicklung des Wissens und der zugehörigen Anwendung – sowohl den auf diesen Gebieten tätigen Forschern als auch ganz besonders der anwendenden Industrie – zur Verfügung stellen. Ziel ist es, qualifizierte Beiträge zum Produktentwicklungsprozess zu leisten.

Albert Albers

Vorwort zu Band 85

Der Prozess der Produktentstehung, also der Weg von der Erfassung von Kundenbedürfnissen, ihrer Interpretation, der Schaffung von Produktlösungen bis hin zur Umsetzung dieser Lösungen in die Produktion ist der zentrale Prozess im Maschinen- und Fahrzeugbau. Eine zukunftsorientierte Gestaltung dieser Prozesse ist zwingend notwendig, da die heutigen Produkte zunehmend komplexer werden und interdisziplinäre Lösungen als mechatronische Konzepte darstellen. Eine zentrale Frage ist, wie in den heutigen Käufermärkten, d. h. Märkten, bei denen Produktangebote im Überfluss vorhanden sind, eine Produktdifferenzierung und eine erfolgreiche Innovation durch Unternehmen geleistet werden kann. Kern des Erfolges ist dabei die Beherrschung des komplexen Produktentstehungsprozesses moderner Produkte und die Unterstützung dieses Prozesses durch geeignete Methoden.

Mit der Systemtechnik als Ergebnis der Systemwissenschaften, liegt ein grundlegender Ansatz zur Beschreibung komplexer Systeme vor. Dieser Ansatz, der vorwiegend von Systemwissenschaftlern vor dem Hintergrund der Beschreibung komplexer gesellschaftlicher Zusammenhänge oder auch Prozesse entwickelt worden ist, kann einen ganz entscheidenden Beitrag leisten, um mit der Komplexität der modernen Produktentstehungsprozesse umzugehen. ROPOHL hat hier mit seinen Arbeiten zur Systemtheorie einen wichtigen Beitrag geleistet und mit der Formulierung des ZHO-Systems aus Zielsystem, Objektsystem und Handlungssystem eine grundlegende Basis für die Beschreibung industrieller Prozesse gelegt. ALBERS hat dies aufgenommen und auf der Basis des ZHO-Modells einen systemtheoretischen Ansatz zur Beschreibung der allgemeinen Produktentstehung erarbeitet. Ein Kern dieses Modells ist das sog. Zielsystem. Hier werden initial die Kundenbedürfnisse und die Bedürfnisse des eigenen Unternehmens an ein zukünftiges Produkt beschrieben. Diese Informationen dienen dann dazu, über das Handlungssystem entsprechende Produktlösungen zu erarbeiten. Dabei stellt das Handlungssystem im Allgemeinen das Unternehmen mit seinen Ressourcen und den verbundenen Partnern im Netzwerk dar. Das Zielsystem beschreibt die zukünftigen geplanten Fähigkeiten und Bedarfe des Produktes und vernetzt diese mit den entsprechenden Begründungen aus dem Markt, der Gesellschaft, den rechtlichen Rahmenbedingungen und den Bedürfnissen des Unternehmens. Aus diesen Zielen des Zielsystems werden dann die Anforderungen abgeleitet und beschrieben. Das Zielsystem ist also das zentrale Element für die Produktdefinition und das zentrale System, in dem alle relevanten Informationen zur Produktbeschreibung und zur Produktgenerierung zusammengeführt werden. Der ZHO-Ansatz stellt eine ganz grundlegende Lösung für Produktentstehungsprozesse dar. Allerdings ist die Modellierung von Zielsystemen eine große Herausforderung, da vielfältigste Informationen strukturiert in ein Gesamtsystem überführt und vernetzt dargestellt werden müssen. Hierbei ist

zu bedenken, dass das Zielsystem während des Produktentstehungsprozesses kontinuierlich weiter mit Informationen gefüllt und damit konkretisiert und fokussiert wird. Um diese Informationsvielfalt zu beherrschen, ist ein grundlegender neuer Ansatz zur Modellierung notwendig.

An dieser Stelle setzt die Forschungsarbeit von Herrn Dr.-Ing. Björn Ebel an. Er hat eine Methodik zur Modellierung von Zielsystemen erforscht, umgesetzt und diese in der praktischen Anwendung validiert. Die Arbeit leistet einen hervorragenden Beitrag zur Forschung an Produktentstehungsprozessen und besticht durch ihre gleichzeitige hohe Relevanz für die praktische Produktentwicklung in den Unternehmen.

April, 2015

Albert Albers

Kurzfassung

Heutige Produktentstehung ist geprägt von einer Vielzahl zu erfüllender Ziele und Anforderungen. Diesen Zielen und Anforderungen steht eine ebenso große Vielzahl an potenziellen Lösungsmöglichkeiten gegenüber, die sich aus unterschiedlichen Lösungsanteilen verschiedener Disziplinen wie Maschinenbau, Elektrotechnik und Informatik zusammensetzen. Da die Ziele und Anforderungen sowie die Lösungsmöglichkeiten stark verflochten sind, ist heutige interdisziplinäre Produktentstehung zwangsläufig komplex und unsicherheitsbehaftet. Die Bedeutung einer ganzheitlichen und durchgängigen Modellierung des Zielsystems hat damit in den vergangenen Jahren erheblich zugenommen.

In dieser Arbeit wird eine Methode zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung entwickelt. Die Methode baut auf der Systemtheorie auf, integriert wesentliche Aspekte der Modelltheorie und berücksichtigt, dass der Mensch im Mittelpunkt der Produktentstehung steht. Hierdurch wird der Komplexität heutiger Produkte und Produktentstehungsprozesse Rechnung getragen und eine Anwendbarkeit der Methode in der interdisziplinären Produktentstehung sichergestellt. Die Methode folgt dabei der Prämisse, dass Zielsysteme vernetzt, nachvollziehbar und kontinuierlich im Produktentstehungsprozess modelliert werden müssen, um der Komplexität heutiger interdisziplinärer Produktentstehung Rechnung tragen zu können. Auf dieser Basis können Zielsystemmodelle zur Extraktion entscheidungsrelevanter Informationen genutzt werden, wodurch die individuelle und organisatorische Akzeptanz der Methode in der Ingenieurspraxis sichergestellt wird.

Die Methode zur Modellierung von Zielsystemen baut zum einen auf relevanten Grundlagen und aktuellen Forschungsergebnissen auf, insbesondere aus den Bereichen der Modellierung von Produktentstehungsprozessen und Zielsystemen. Zum anderen werden praxisrelevante Erkenntnisse für die Modellierung von Zielsystemen anhand zweier Untersuchungen im Kontext realer Produktentwicklungsprojekte gewonnen. Hierdurch kann die Methode an den tatsächlichen Bedarfen von Produktentwicklern in der Ingenieurspraxis ausgerichtet werden. Auf dieser Basis wird eine praxisrelevante Strukturierung von Zielsystemen entwickelt, die erforderliche Partialmodelle, Elementtypen, Relationen und Attribute enthält. Zusätzlich werden Modellierungsregeln und ein Werkzeug zur Modellierung von Zielsystemen entwickelt, die eine Operationalisierung und Validierung der Methode in der Produktentwicklungspraxis ermöglichen.

Die durchgeführte Validierung bestätigt die Tauglichkeit der Methode zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung und zeigt gleichzeitig Verbesserungspotenziale und zukünftige Forschungsbedarfe auf.

abstract

Today's product engineering is characterized by a variety of objectives and requirements that are to be fulfilled. These objectives and requirements are confronted with an even equal variety of potential solutions that are composed of variable solution patterns of several disciplines, especially mechanical and electrical engineering as well as informatics. Since objectives, requirements and the potential solutions are highly interdependent, product engineering is inevitably complex and uncertainty is ubiquitous. Therefore, a holistic and traceable modeling of systems of objectives becomes ever more acute.

Within this thesis a method is developed that enables to model systems of objectives in interdisciplinary product engineering. The method bases on system theory, integrates essential aspects of model theory and takes into account that human are in the center of product engineering. Through this, the complexity of today's products and product engineering processes is taken account and the applicability of the method within interdisciplinary product engineering is ensured. The method adopts the system of objectives as to be modeled networked, traceable and continuous within the whole product engineering process in order to consider the complexity of today's product engineering. On this basis, system of objectives allow the extraction of crucial information, by what an individual and organizational acceptance of the method in engineering design practice is achieved.

The method for modelling system of objectives builds on relevant basics and current research results with focus on modelling of product engineering processes and system of objectives. Additionally, two investigations in the context of real product engineering projects reveal practice-oriented findings for the modelling of system of objectives. This allows the alignment of the method with the actual needs of product engineers in today's engineering practice. On this basis, a practice-oriented structuring of system of objectives is developed, containing necessary partial models, elements, relations and attributes. In addition, a set of modelling rules and a tool for modelling system of objectives are developed, which allow an operationalization and validation in engineering design practice.

The validation carried out confirms the suitability of the method for modelling system of objectives within interdisciplinary product engineering and reveals room for improvement as well as future needs for research.

Danksagung

Die vorliegende Arbeit entstand im Rahmen meiner Tätigkeiten als akademischer Mitarbeiter am IPEK – Institut für Produktentwicklung am Karlsruher Institut für Technologie (KIT).

Mein besonderer Dank gilt meinem Doktorvater Herrn Prof. Dr.-Ing. Dr. h.c. Albert Albers, der mich durch die übertragene Verantwortung, das mir entgegengebrachte Vertrauen und die damit einhergehenden Gestaltungsfreiräume in besonderer Weise gefördert hat. In unzähligen, teils sehr lebhaften Diskussionen nahm er positiven Einfluss auf mein Denken und Handeln und trug somit ganz erheblich zum Gelingen dieser Arbeit bei.

Für die Übernahme des Korreferats bedanke ich mich bei Herrn Prof. Dr.-Ing. Jürgen Gausemeier, der durch seine ganzheitliche Sicht auf die Produktentstehung wichtige Impulse für die Arbeit gegeben hat.

Dem gesamten IPEK-Team möchte ich für die kollegiale und konstruktive Atmosphäre danken, die mich stets motiviert und inspiriert hat. Insbesondere den aktuellen und ehemaligen Kolleg(inn)en des Maschinenbauhochhauses möchte ich meinen Dank aussprechen für die wissenschaftlichen und nicht-wissenschaftlichen Gespräche am IPEK und anderswo, für die Unterstützung in allen fachlichen wie auch organisatorischen Angelegenheiten.

Danken möchte ich auch meinen ehemaligen Kollegen von Heidelberger Druck, die wesentlich dazu beitrugen, meine Forschung an den Bedarfen und Randbedingungen heutiger Ingenieurspraxis auszurichten.

Mein größter Dank gilt meiner Familie. Allen voran meinen Eltern möchte ich danken für das, was sie mir mitgegeben und ermöglicht haben. Meiner Frau Christina danke ich für die immerwährende Unterstützung, insbesondere in der Schlussphase meiner Dissertation, die ich ohne dich nicht gemeistert hätte. Und dir Milena danke ich, dass du immer wieder ein Lächeln auf mein Gesicht zauberst.

Björn Ebel

*Der Langsamste, der sein Ziel nicht aus den Augen verliert,
geht immer noch geschwinder, als der ohne Ziel umherirrt.*

Gotthold Ephraim Lessing

*Kaum verloren wir das Ziel aus den Augen,
verdoppelten wir unsere Anstrengungen.*

Mark Twain

Inhalt

| | | |
|----------|---|-----------|
| 1 | Einleitung | 1 |
| 2 | Grundlagen und Stand der Forschung | 5 |
| 2.1 | System- und Modelltheorie | 5 |
| 2.1.1 | Allgemeine Systemtheorie..... | 5 |
| 2.1.2 | Allgemeine Modelltheorie..... | 11 |
| 2.1.3 | Zwischenfazit | 16 |
| 2.2 | Systemmodell der Produktentstehung | 16 |
| 2.2.1 | ZHO-Modell | 16 |
| 2.2.2 | Prinzipien des ZHO-Modells | 20 |
| 2.2.3 | Erweitertes ZHO-Modell..... | 21 |
| 2.2.4 | Zwischenfazit | 23 |
| 2.3 | Komplexität, Unsicherheit und Iterationen | 24 |
| 2.3.1 | Komplexität in der Produktentstehung | 24 |
| 2.3.2 | Unsicherheit in der Produktentstehung | 27 |
| 2.3.3 | Iterationen in der Produktentstehung | 30 |
| 2.3.4 | Zwischenfazit | 34 |
| 2.4 | Produktentstehungsprozesse | 34 |
| 2.4.1 | Mentale Modelle und Modelle elementarer Problemlösung | 36 |
| 2.4.2 | Modelle operativer Problemlösung..... | 37 |
| 2.4.3 | Vorgehens- und Phasenmodelle der Produktentstehung | 41 |
| 2.4.4 | Übergreifende und ganzheitliche Modelle der Produktentstehung | 50 |
| 2.4.5 | Zwischenfazit | 55 |
| 2.5 | Zielsysteme in der Produktentstehung..... | 55 |
| 2.5.1 | Ziele und Anforderungen | 65 |
| 2.5.2 | Abbildung von Zielsystemen | 72 |
| 2.5.3 | Zwischenfazit | 76 |
| 3 | Motivation und Zielsetzung der Arbeit | 79 |
| 3.1 | Motivation | 79 |
| 3.2 | Zielsetzung | 80 |
| 4 | Forschungsdesign | 83 |
| 4.1 | Forschungsfragen | 83 |
| 4.2 | Forschungsvorgehen..... | 84 |
| 5 | Verständnis von Zielsystemen | 87 |
| 5.1 | Erfolgsfaktoren der Interdisziplinarität..... | 87 |
| 5.1.1 | Konzeption des Expertenworkshops | 88 |
| 5.1.2 | Ergebnisse des Expertenworkshops | 90 |

| | | |
|-----------|---|------------|
| 5.1.3 | Zusammenfassung und Interpretation der Ergebnisse | 92 |
| 5.2 | Zwecke von Zielen | 95 |
| 5.3 | Beurteilung von Zielen..... | 98 |
| 5.3.1 | Beurteilungsdimensionen von Zielen | 99 |
| 5.3.2 | Matrizen von Beurteilungsdimensionen | 105 |
| 6 | Untersuchungen zur Modellierung von Zielsystemen | 115 |
| 6.1 | Modellierung von Zielsystemen mittels Semantic MediaWikis | 115 |
| 6.1.1 | Zielsetzung der Untersuchung | 115 |
| 6.1.2 | Rahmenbedingungen der Untersuchung..... | 117 |
| 6.1.3 | Ergebnisse und Interpretation der Untersuchung | 118 |
| 6.2 | Modellierung von Zielsystemen mittels Mind-Maps..... | 130 |
| 6.2.1 | Zielsetzung der Untersuchung | 130 |
| 6.2.2 | Rahmenbedingungen der Untersuchung..... | 132 |
| 6.2.3 | Ergebnisse und Interpretation der Untersuchung | 133 |
| 6.3 | Erkenntnisse aus den Untersuchungen | 147 |
| 7 | Modellierung von Zielsystemen | 151 |
| 7.1 | Strukturierung von Zielsystemen | 151 |
| 7.1.1 | Partialmodelle im Zielsystem | 151 |
| 7.1.2 | Elementtypen im Zielsystem | 156 |
| 7.1.3 | Relationen im Zielsystem..... | 160 |
| 7.1.4 | Attribute im Zielsystem..... | 163 |
| 7.2 | Regeln zur Modellierung von Zielsystemen | 166 |
| 7.2.1 | Allgemeine Modellierungsregeln | 166 |
| 7.2.2 | Spezifische Modellierungsregeln | 171 |
| 7.2.3 | Extraktion entscheidungsrelevanter Informationen..... | 173 |
| 7.3 | Werkzeug zur Modellierung von Zielsystemen..... | 175 |
| 7.3.1 | Ziele der Werkzeugentwicklung | 175 |
| 7.3.2 | Ergebnis der Werkzeugentwicklung | 183 |
| 7.4 | Validierung der Methode zur Modellierung von Zielsystemen | 184 |
| 7.4.1 | Retrospektive Modellierung..... | 184 |
| 7.4.2 | Einsatz in der Entwicklungspraxis | 186 |
| 7.4.3 | Abschließende Bewertung | 191 |
| 8 | Zusammenfassung und Ausblick..... | 193 |
| 8.1 | Zusammenfassung..... | 193 |
| 8.2 | Ausblick..... | 194 |
| 9 | Literaturverzeichnis | 197 |
| 10 | Anhang – Anleitung des Werkzeugs SystemModeler | 211 |

1 Einleitung

Die Herausforderungen heutiger Produktentstehung sind vielfältig und stehen oftmals in unmittelbarer Beziehung zum Erfolg eines Produkts. Dabei war die richtige Entwicklung des richtigen Produkts noch nie trivial. Jedoch waren die Ziele einer Produktentstehung und deren potenzielle Lösungsmöglichkeiten noch nie so vielfältig und untereinander verflochten wie heute.

Die Entwicklung des richtigen Produkts setzt die Definition der richtigen Ziele voraus. Hierbei müssen neben den eigentlichen Produktzielen auch weitere Ziele aus den Bereichen Finanzen, Marketing/Vertrieb, Produktion und Projektmanagement definiert werden¹. Stellt bereits die Definition der einzelnen Ziele eine erhebliche Herausforderung dar, führt die Abstimmung der einzelnen Ziele untereinander oftmals zu Problemen. Nicht selten werden Unverträglichkeiten oder Widersprüche zwischen einzelnen Zielen erst im Verlauf eines Produktentstehungsprozesses identifiziert, wodurch zumeist erhebliche Änderungsaufwände in Kosten und Zeit entstehen. Werden wesentliche Unstimmigkeiten in den Zielen eines Produkts bis zur Markteinführung nicht identifiziert, ist das Scheitern des Produkts oftmals bereits vorgezeichnet. Jedoch stellt die Definition der richtigen Ziele noch keine Garantie für die Entwicklung eines erfolgreichen Produkts dar. Zum einen existieren stets nicht-vorhersehbare bzw. nicht-beeinflussbare Faktoren mit Einfluss auf den Erfolg eines Produkts, zum anderen müssen die definierten Ziele im Prozess der Produktentstehung richtig umgesetzt werden. Hierzu dürfen die Ziele nicht losgelöst voneinander betrachtet werden und abgeleitete Anforderungen an das Produkt müssen nachvollzogen werden können. Werden Produktentwickler mit isolierten Anforderungen konfrontiert, sind sie gezwungen die zugrundeliegenden Ziele zu antizipieren und auf dieser Basis das Produkt zu entwickeln. Dies führt nicht selten dazu, dass die explizit definierten Anforderungen zwar erfüllt sind, die zugrundeliegenden Ziele jedoch nicht erreicht werden.

Um den Herausforderungen heutiger Produktentstehung begegnen zu können, müssen die vielfältigen Ziele und abgeleiteten Anforderungen sowie deren Wechselwirkungen als Zielsystem aufgefasst und modelliert werden². Hierbei muss auf die Komplexität heutiger Zielsysteme sowohl mit einer angemessenen Zielsystemmodellierung als auch mit einer adäquaten Gestaltung des Produktentste-

¹ vgl. Bader 2007, S. 20

² vgl. Albers, Klinger & Ebel 2013

hungsprozesses reagiert werden. Eine Abbildung der Ziele und Anforderungen in Form linearer Lasten- und Pflichtenhefte³, die in einem sequentiell verlaufenden Produktentstehungsprozess im Rahmen einer Aufgabenklärung erstellt werden⁴, ist hierbei nicht mehr in der Lage, mit der Komplexität heutiger Zielsysteme erfolgreich umzugehen. Die Modellierung ganzer Zielsysteme erfordert heutzutage die vernetzte und nachvollziehbare Abbildung aller relevanten Ziele und abgeleiteten Anforderungen. Da der Komplexität heutiger Produktentstehung nur auf Basis einer kontinuierlichen Validierung begegnet werden kann, muss die Modellierung des Zielsystems darüber hinaus auf eine kontinuierliche Verfeinerung der Zielsysteminhalte im Rahmen eines iterativ verlaufenden Produktentstehungsprozesses abgestimmt sein⁵. Die Modellierung von Zielsystemen kann folglich nicht mehr auf Basis des Verständnisses einmal zu fixierender Lasten- und Pflichtenhefte erfolgen⁶ – Zielsysteme müssen vernetzt, nachvollziehbar und kontinuierlich im Produktentstehungsprozess modelliert werden.

Das Ziel der vorliegenden Arbeit ist die Entwicklung einer Methode zur Modellierung von Zielsystemen, die den Herausforderungen heutiger interdisziplinärer Produktentstehungsprozesse gewachsen ist. Die Methode muss daher eine vernetzte, nachvollziehbare und kontinuierliche Modellierung aller relevanten Ziele und Anforderungen erlauben. Gleichzeitig muss die Methode die Nutzung des Zielsystemmodells unterstützen und hierdurch einen dem Modellierungsaufwand angemessenen Mehrwert generieren. Andernfalls kann keine individuelle und organisatorische Akzeptanz der Methode in der Ingenieurspraxis erreicht werden⁷. Die Methode zur Modellierung von Zielsystemen muss den einzelnen Produktentwickler folglich dabei unterstützen, relevante Ziele und Anforderungen zu identifizieren, effizient zu modellieren und aus dem resultierenden Zielsystemmodell Schlüsse ziehen zu können, die für den Entwickler nicht auf der Hand liegen.

Um diese Ziele zu erreichen, baut die Arbeit auf erforderlichen Grundlagen auf und bezieht dabei den aktuellen Stand der Forschung mit ein (vgl. Kapitel 2 in Abbildung 1-1). Hierbei werden Schwerpunkte in den Bereichen der Modellierung von Produktentstehungsprozessen und Zielsystemen gelegt. Auf dieser Basis wird die Arbeit fundiert motiviert und eine detaillierte Zielsetzung abgeleitet (Kapitel 3). Hieraus leiten sich drei zentrale Forschungsfragen sowie ein hierauf abgestimmtes

³ vgl. VDI 2519 2001

⁴ vgl. VDI 2221 1993

⁵ vgl. Albers, Ebel & Lohmeyer 2012

⁶ vgl. Darlington & Culley 2002

⁷ vgl. Albers, Lohmeyer & Radimersky 2012

Forschungsvorgehen ab, das die Beantwortung der Forschungsfragen und damit die Erfüllung der Zielsetzung ermöglicht (Kapitel 4). Hierbei wird zuerst ein Verständnis von Zielen und Zielsystemen aufgebaut, das als wesentliches Fundament für die Modellierung von Zielsystemen in der interdisziplinären Produktentstehung erforderlich ist (Kapitel 5). Anschließend wird die Modellierung von Zielsystemen in der industriellen Praxis untersucht, um im Kontext interdisziplinärer Produktentstehungsprozesse relevante Erkenntnisse für die Strukturierung und Modellierung von Zielsystemen zu gewinnen (Kapitel 6). Auf dieser Basis wird eine Strukturierung von Zielsystemen entwickelt, die den formulierten Zielen der Arbeit gerecht wird. Zur Sicherstellung der Operationalisierbarkeit dieser Strukturierung werden darüber hinaus Regel sowie ein Werkzeug zur Zielsystemmodellierung entwickelt. Abschließend wird die Methode im Rahmen zweier interdisziplinärer Produktentstehungsprojekte einer Validierung unterzogen (Kapitel 7).

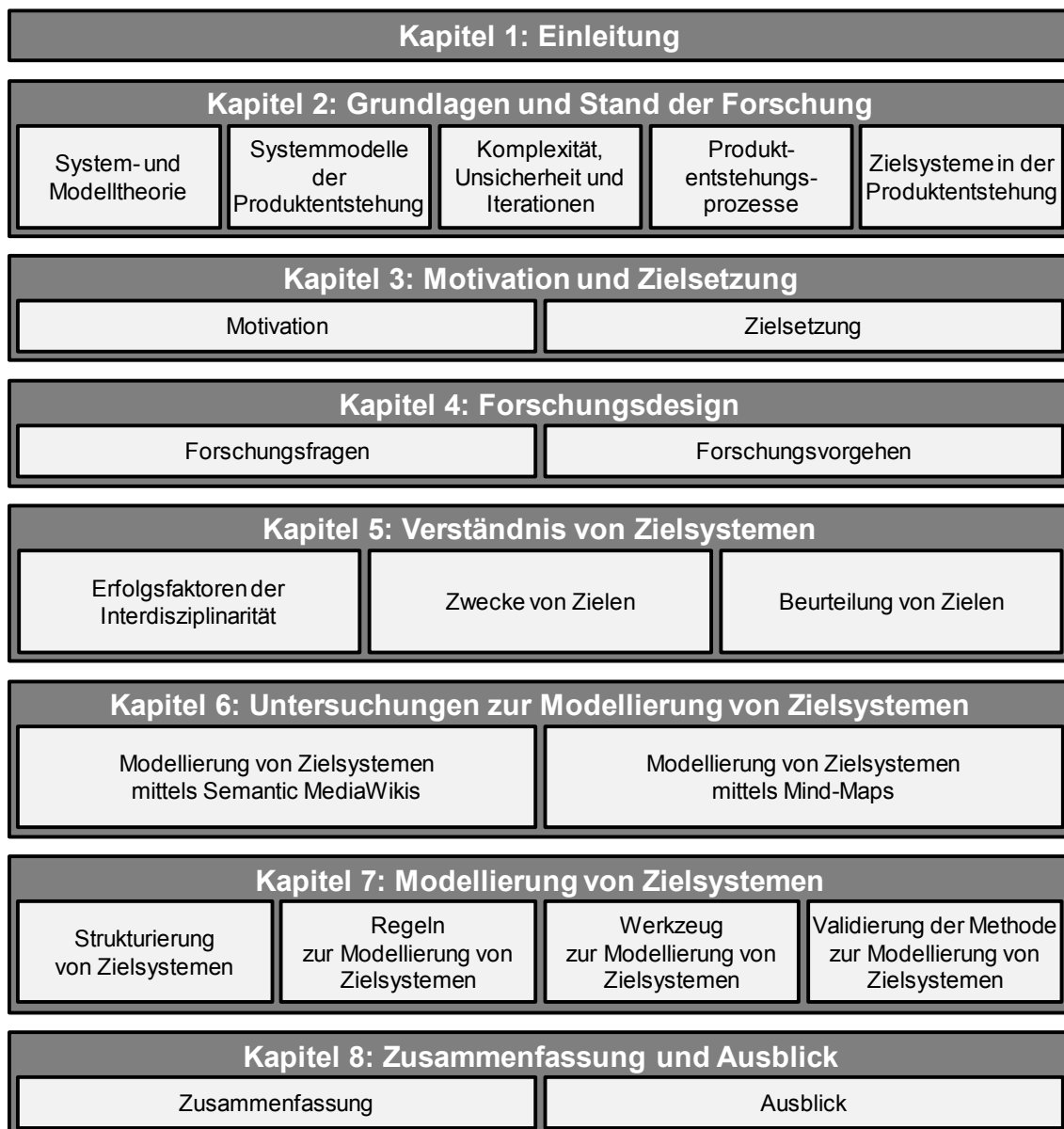


Abbildung 1-1: Aufbau und Struktur der Arbeit

2 Grundlagen und Stand der Forschung

In diesem Kapitel werden die Grundlagen für die Arbeit und der relevante Stand der Forschung dargestellt.

2.1 System- und Modelltheorie

2.1.1 Allgemeine Systemtheorie

Die Allgemeine Systemtheorie ist eine umfassende Theorie zur disziplinenübergreifenden Systembeschreibung. Sie bildet damit eine gemeinsame Basis für eine notwendige disziplinspezifische Ausprägungen bzw. disziplinäre Anwendung der Systemtheorie und damit eine gemeinsame Basis interdisziplinärer Forschung. BERTALANFFY⁸ formuliert folgende Ziele der Allgemeinen Systemtheorie:

- Unterstützung der zunehmenden Integration in den Disziplinen,
- Schaffung exakter Theorien und Wissenschaften außerhalb der Physik,
- Entwicklung einer Einheit der Wissenschaften und eines einheitlichen, systemtheoretischen Weltbilds,
- Vereinfachung und Abstraktion wissenschaftlicher Erklärungsmodelle,
- Unterstützung der wissenschaftlichen Ausbildung und Schaffung wissenschaftlicher Generalisten

Systembegriff

Die Ursprünge der Systemtheorie lassen sich bis in die griechische Philosophie zurückverfolgen. Bereits Aristoteles unterscheidet zwischen der bloßen Menge (pan) und der Ganzheit (holon) von Elementen. Ganzheit liegt demnach vor, wenn die Anordnung der Elemente von Bedeutung ist. Ist die Anordnung hingegen nicht von Bedeutung, handelt es sich lediglich um eine Vielzahl⁹. In den 1930er Jahren präzisiert BERTALANFFY¹⁰ in seiner Allgemeinen Systemlehre den Begriff der **Ganzheitlichkeit**: „Die Eigenschaften und Verhaltensweisen höherer Ebenen sind nicht durch die Summation der Eigenschaften und Verhaltensweisen ihrer Bestandteile erklärbar, solange man diese isoliert betrachtet. Wenn wir jedoch das Ensemble der Bestandteile und Relationen kennen, die zwischen ihnen bestehen, dann sind die höheren Ebenen von den Bestandteilen ableitbar“. Das Ganze ist demnach die Summe der einzelnen Elemente und die Summe der Relationen

⁸ vgl. Bertalanffy 1969, S. 38

⁹ vgl. Rophol 2009, S. 71

¹⁰ vgl. Bertalanffy 1949, S. 25

zwischen ihnen. BERTALANFFY beschreibt damit **emergente** Systemeigenschaften, die PULM¹¹ als solche Eigenschaften definiert, die „nicht mehr aus den Eigenschaften der Elemente des Systems, sondern nur auf der betrachteten Systemebene erklärbar sind. Emergenz stellt das Prinzip der Entstehung dieser Eigenschaften dar“. In den 1950er Jahren wurde durch WIENER die **Kybernetik** begründet, die einen weiteren Bestandteil der heutigen Allgemeinen Systemtheorie bildet. Wiener versteht unter der Kybernetik „the entire field of control and communication theory, whether in the machine or in the animal“¹². STEMPFLE¹³ sieht in der Kybernetik die Ursprünge der Regelungssysteme, die auf einem Informationstransfer zwischen dem System und dessen Umwelt sowie einer Rückkopplung zwischen der Umwelt und dem System basieren. In den späten 1970er Jahren verfasst BERTALANFFY¹⁴ die *General System Theory*, die nach LOHMEYER¹⁵ das Schlüsselwerk des heutigen Systemverständnisses darstellt. Dennoch stellt ROPOHL¹⁶ heraus, dass die Allgemeine Systemtheorie bis heute daran leidet, dass drei unterschiedliche Systemdeutungen vertreten werden, die jeweils einen Systemaspekt in den Vordergrund stellen, während der Systembegriff in Wirklichkeit drei Aspekte umfasst: das **hierarchische**, das **strukturelle** und das **funktionale Systemkonzept** (vgl. Abbildung 2-1).

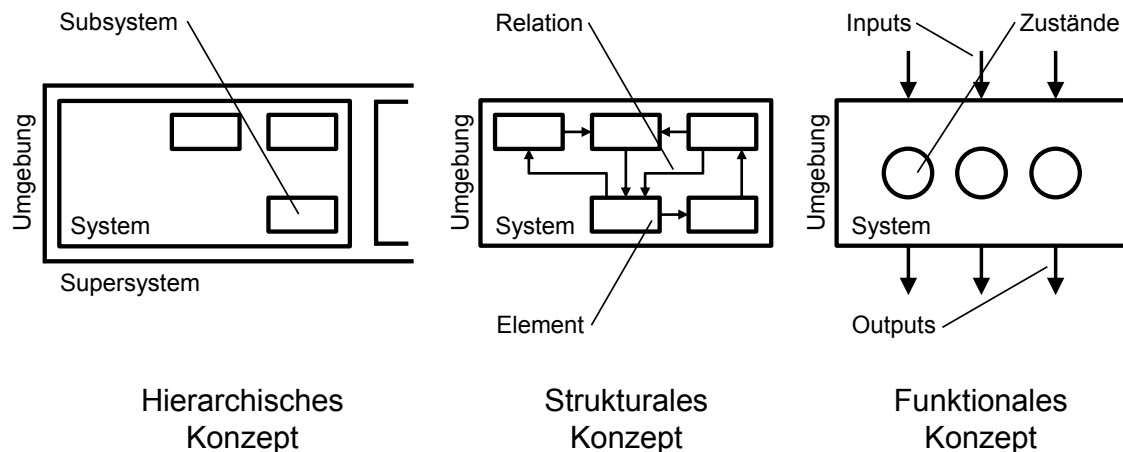


Abbildung 2-1: Konzepte der Systemtheorie¹⁷

Im **hierarchischen Systemkonzept** kommt der Grundsatz des Systemdenkens zum Ausdruck¹⁸. Es umfasst zum einen die Festlegung einer Systemgrenze und damit die

¹¹ vgl. Pulm 2004, S. 19

¹² vgl. Wiener 1949, S.19

¹³ vgl. Stempfle 2007, S. 7

¹⁴ vgl. Bertalanffy 1969

¹⁵ vgl. Lohmeyer 2012, S. 14

¹⁶ vgl. Ropohl 2009, S. 75

¹⁷ vgl. Ropohl 2009, S. 76

¹⁸ vgl. Oerding 2009, S. 53

Abgrenzung eines Systems zu seiner Umgebung. Zum anderen betont das hierarchische Systemkonzept den Umstand, dass Systeme selbst aus Systemen bestehen – Systeme demnach gleichzeitig Sub- und Supersystem sein können. Eine umfassende Systembetrachtung kann folglich mehrere Stufen der Systemhierarchie umfassen¹⁹. „Bewegt man sich in der Hierarchie abwärts, so erhält man eine detailliertere Erklärung des Systems, während man, wenn man sich in der Hierarchie aufwärts bewegt, ein tieferes Verständnis seiner Bedeutung gewinnt“²⁰.

Das **strukturele Systemkonzept** besteht darin, ein System als Ganzheit miteinander verknüpfter Elemente zu betrachten. Ein wesentlicher Bestandteil des Konzepts ist die Vielfalt der möglichen Beziehungsgeflechte und die daraus resultierenden unterschiedlichen Systemeigenschaften. Ein weiterer Bestandteil des Konzepts befasst sich mit der Beschaffenheit der Elemente, von der es letztendlich abhängt, wie gut oder schlecht sie sich in ein System integrieren lässt (integrale Qualität der Elemente²¹).

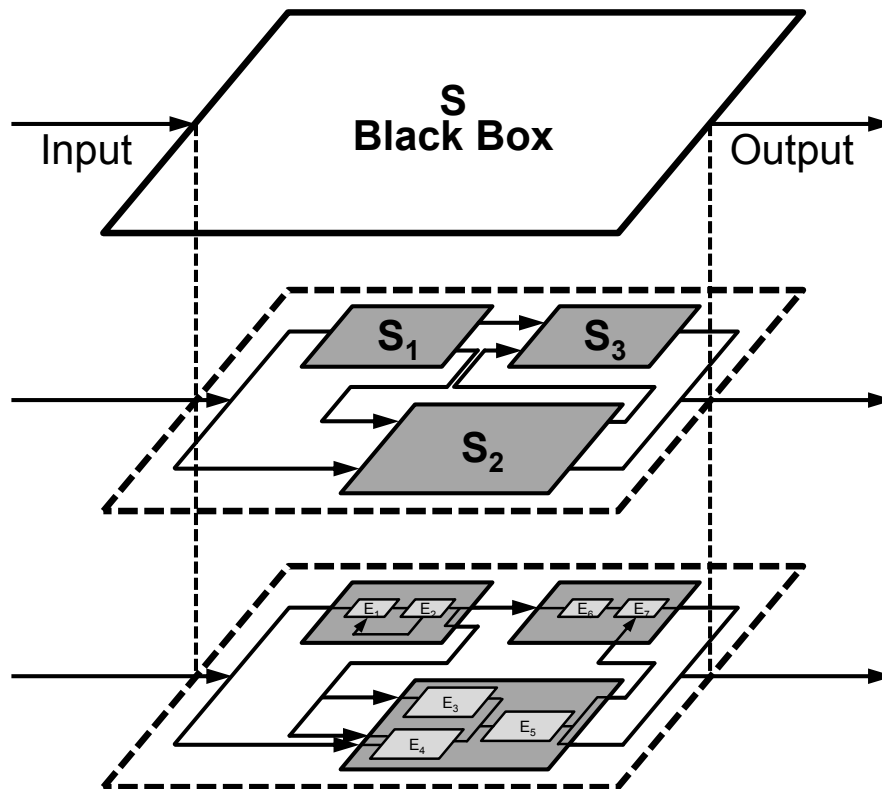


Abbildung 2-2: Kombination der Systemkonzepte²²

¹⁹ vgl. Ropohl 2009, S. 77

²⁰ vgl. Mesarovic & Macko 1969, S. 35

²¹ vgl. Pfeiffer 1965, S. 43

²² vgl. Patzak 1982, S. 51 und Ehrlenspiel 2009, S. 19

Das **funktionale Systemkonzept** rückt das beobachtbare Systemverhalten in den Mittelpunkt der Betrachtung und lässt den tatsächlichen inneren Aufbau außer Acht. Es „behandelt nicht Dinge sondern Verhaltensweisen“ und „fragt nicht: ‚Was ist dieses Ding?‘, sondern: ‚Was tut es?‘“²³. Das funktionale Systemkonzept fokussiert damit auf die Eingangsgrößen (Inputs), Ausgangsgrößen (Outputs) und Zustände eines Systems.

Diese drei Systemkonzepte schließen einander nicht aus – Sie stellen vielmehr verschiedene Aspekte eines Systemmodells dar und können in Kombination verwendet werden (vgl. Abbildung 2-2). Folgerichtig integriert ROPHOL²⁴ verschiedene Systemdefinitionen²⁵, die jeweils nur einzelne Aspekte abdecken, zu einer neuen Definition:

Definition 2-1: System

Ein System ist das Modell einer Ganzheit, die Beziehungen zwischen Attributen (Inputs, Outputs, Zustände) aufweist, die aus miteinander verknüpften Teilen bzw. Subsystemen besteht, und die von ihrer Umgebung bzw. von einem Supersystem [durch eine Systemgrenze] abgegrenzt wird.²⁶

Funktionsbegriff

Auf der Grundlage des mathematischen Systembegriffs²⁷ beschreibt ROPHOL²⁸ ein System als Quadrupel aus den Mengen α (Attribute), φ (Funktionen) σ (Subsystemen) und π (Relationen). **Attribute** und **Funktionen** repräsentieren dabei die äußeren Eigenschaften eines Systems – Funktionen stellen die Beziehung zwischen den Attributen dar (vgl. funktionales Systemkonzept). **Subsysteme** und **Relationen** repräsentieren die inneren Bestandteile eines Systems – Relationen setzen Subsystemen in Beziehung (vgl. struktureles Systemkonzept)²⁹. ROPHOL führt an, dass mit diesem Systemverständnis ein deskriptiver Funktionsbegriff festgelegt wird, der Zusammenhänge zwischen Attributen (Inputs, Outputs und Zustände) eines Systems beschreibt³⁰. Gleichzeitig betont ROPHOL, dass neben dem deskriptiven Funktionsbegriff ein teleologisch-normativer Funktionsbegriff existiert, der den

²³ vgl. Ashby 1974, S. 15

²⁴ vgl. Ropohl 2009, S. 77

²⁵ vgl. Hall & Fagen 1956, Greniewski & Kempisty 1963, Lange 1965, Wintgen 1968, Klir 1972

²⁶ vgl. Ropohl 2009, S. 77

²⁷ vgl. Stachowiak 1973, S. 244ff

²⁸ vgl. Ropohl 2009, S. 312

²⁹ vgl. Stempfle 2007, S. 10

³⁰ vgl. Rophol 2009, S. 79

intendierten Zweck eines Systems beschreibt. Um der so entstehenden Doppeldeutigkeit vorzubeugen, schlägt ROPOHL vor, den Funktionsbegriff nur im deskriptiven Sinn zu benutzen, da der intendierte Zweck eines Systems außerhalb der Systemgrenze liegt und mit anderen systemtheoretischen Mitteln modelliert werden kann. Studien zur Anwendung des Funktionsbegriffs im Ingenieursumfeld haben jedoch gezeigt, dass in der Praxis der Funktionsbegriff sowohl mit deskriptiver als auch mit teleologisch-normativer Konnotation Verwendung findet³¹. Im Rahmen einer umfassenden Analyse bestehender Literatur identifiziert ERDEN 18 verschiedene Notationen des Funktionsbegriffs³², auf deren Basis VERMAAS das 5-Key-Concepts zur Beschreibung der unterschiedlichen Bedeutungen des Funktionsbegriffs herleitet. Im Kern unterscheidet VERMAAS zwischen drei archetypischen Bedeutungen von Funktionen:

- Funktion als vorgesehenes **Verhalten** (intended behavior)
- Funktion als gewünschte **Wirkung** (desired effects)
- Funktion als **Zweck** der Gestaltung (purposes for which the device is designed)

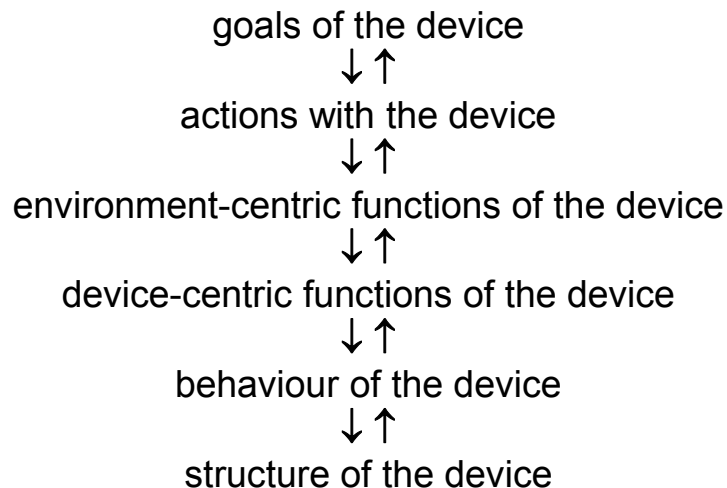
Auf Basis dieser Archetypen und Arbeiten von BROWN und BLESSING³³ ordnet VERMAAS die in der Ingenieurspraxis Verwendung findenden Bedeutungen des Funktionsbegriffs zusammen mit angrenzenden Konzepten durchgängig an (vgl. Abbildung 2-3). Er betont, dass die in der Ingenieurspraxis vorhandene Mehrdeutigkeit des Funktionsbegriffs auch Vorteile mit sich bringt, da so eine Brücke zwischen den unterschiedlichen Konzepten geschlagen werden kann. Darüber hinaus ist es möglich, pragmatische und situationsgerechte Simplifizierungen vorzunehmen, beispielsweise das Verbergen (cloaking) oder das Umgehen (bypassing) einzelner Konzepte. Hierbei schlägt ALINK³⁴ eine getrennte Betrachtung von Bedeutung, Darstellung und Formulierung einzelner Funktionen vor, um die Verwendung des Funktionsbegriffs in der Ingenieurspraxis zu verbessern.

³¹ vgl. Eckert et al. 2011, S. 833

³² vgl. Erden et al. 2008

³³ vgl. Brown & Blessing 2005

³⁴ vgl. Alink 2010, S. 160

Abbildung 2-3: 5-Key-Concepts nach VERMAAS³⁵

Insgesamt geht VERMAAS von einem teleologisch-normativen Funktionsbegriff aus, d.h. dass eine Funktion immer in Zusammenhang mit Wünschen, Absichten oder Zielen steht. Dieses Funktionsverständnis ist weit verbreitet und deckt sich mit Ansichten von PAHL ET AL³⁶., EHRENSPIEL³⁷, PONN UND LINDEMANN³⁸ und anderen³⁹. Dieses Verständnis des Funktionsbegriffs kann jedoch bei der Anwendung auf ein bestehendes System zu Problemen führen, da eine Abgrenzung zwischen der gewünschten und der tatsächlichen Funktion nicht sinnvoll möglich ist. Dies wird insbesondere dann problematisch, wenn die tatsächliche Funktion nicht den Wünschen, Absichten oder Zielen entspricht. ZINGEL⁴⁰ schlägt daher eine Differenzierung des Funktionsbegriffs entsprechend der von ROPOHL dargelegten Unterscheidung zwischen deskriptivem und teleologisch-normativem Funktionsverständnis vor, da beide Aspekte benötigt werden⁴¹. Ausgehend von der Definition von ROPOHL⁴² ergibt sich damit folgende Definition für Funktion:

Definition 2-2: Funktion

Eine Funktion beschreibt die gewünschte oder tatsächliche Beziehung zwischen Attributen (Inputs, Outputs, Zustände) eines Systems.

³⁵ vgl. Vermaas 2010

³⁶ vgl. Pahl et al. 2007, S. 44

³⁷ vgl. Ehrlenspiel 2009, S. 19

³⁸ vgl. Ponn & Lindemann 2011, S. 61

³⁹ vgl. z.B. VDI 2221 1993

⁴⁰ vgl. Zingel 2013, S. 12

⁴¹ vgl. Alink 2010, S. 25

⁴² vgl. Ropohl 2009, S. 79

2.1.2 Allgemeine Modelltheorie

Der abbildtheoretische Modellbegriff besitzt eine lange Historie und seine erkenntnistheoretischen Grundlagen können bis in die Antike verfolgt werden⁴³. Nach THOMAS⁴⁴ besitzt der Terminus Modell vom Ursprung des Wortes her eine zweifache Doppeldeutung im Sinne von Abbild von etwas bzw. Vorbild für etwas sowie als reale, stoffliche Konstruktion bzw. als Gedankenkonstruktion. Hierauf kann zurück geführt werden, dass die alltagssprachliche als auch die wissenschaftliche Verwendung des Modellbegriffs nicht einheitlich sind – „würde man [...] verschiedenen Wissenschaftlern die Frage stellen, was eigentlich ein Modell ist, so würde man kaum übereinstimmende Antworten erhalten“⁴⁵. Weitestgehend Konsens herrscht in der Auffassung, dass ein Modell ein „**adäquates Abbild der betrachteten Wirklichkeit**“⁴⁶ und ein „**Mittel der menschlichen Erkenntnisgewinnung**“⁴⁷ ist. Hieraus wird deutlich, dass ein Modell stets mit einem menschlichen Subjekt in Verbindung steht, das die Adäquatheit bestimmt, das den Wirklichkeitsausschnitt festlegt und das Erkenntnisse zu einem bestimmten Zweck erlangen möchte. Hieraus folgt, „dass ein Modell nicht nur eine Zusammenfassung der bekannten Fakten sein darf; das Modell hat vielmehr auch die Aufgabe, das aus den Fakten zu gewinnende mehr oder weniger unvollständige Bild zu einem Gesamtbild zu ergänzen“⁴⁸. STACHOWIAK⁴⁹ untersucht in seiner Allgemeinen Modelltheorie den Modellbegriff in seinen Hauptcharakteristika auf intuitiv-umgangssprachliche Weise und liefert mit dem Abbildungsmerkmal, dem Verkürzungsmerkmal und dem pragmatischen Merkmal drei Hauptmerkmale des allgemeinen Modellbegriffs. STACHOWIAK präzisiert damit den Modellbegriff und klärt insbesondere die Wechselwirkung zwischen Modell und Modellerschaffer bzw. Modellbenutzer.

Abbildungsmerkmal

„Modelle sind stets Modelle von etwas, nämlich Abbildungen, Repräsentationen natürlicher oder künstlicher Originale, die selbst wieder Modelle sein können.“⁵⁰ Das Abbildungsmerkmal beschreibt damit die einfache, jedoch essenzielle Tatsache, dass ein Modell ein **Abbild eines Originals** ist. Als Original kann jede Entität aufgefasst werden, die von einem auf Wissensvermehrung intendierten Subjekt

⁴³ vgl. Keuth 1978, S. 28 und Müller 1983, S. 62

⁴⁴ vgl. Thomas 2005, S. 6

⁴⁵ vgl. Stoff 1969, S. 17

⁴⁶ vgl. Kosiol 1961, S. 321

⁴⁷ vgl. Klaus 1967, S. 412

⁴⁸ vgl. Klaus 1962, S. 247

⁴⁹ vgl. Stachowiak 1973

⁵⁰ vgl. Stachowiak 1973, S. 131

erfahrbar ist. Unter Abbildung wird im mathematischen Sinn eine Zuordnung von Originalattributen zu Modellattributen verstanden⁵¹.

Verkürzungsmerkmal

„Modelle erfassen im Allgemeinen nicht alle Attribute des durch sie repräsentierten Originals, sondern nur solche, die den jeweiligen Modellerschaffern und/oder Modellbenutzern relevant erscheinen.“⁵² Das Verkürzungsmerkmal beschreibt die Tatsache, dass ein Modell im Allgemeinen nicht vollumgänglich seinem Original entspricht. Bewusst und unbewusst **verzichtet der Modellerschaffer auf Attribute des Originals**, deutet Attribute um und/oder fügt neue Attribute ein. Eine bewusste Verkürzung des Modells nimmt der Modellerschaffer vor dem Hintergrund der wahrgenommenen Attribute des Originals sowie der für die Modellbenutzung als relevant eingestuften Attribute vor. Eine unbewusste Verkürzung nimmt der Modellerschaffer vor, da er zum einen nicht alle Attribute des Originals wahrnimmt und zum anderen nicht alle tatsächlich relevanten Attribute zur Nutzung des Modells durch einen (anderen) Modellbenutzer als relevant einstuft.

Pragmatisches Merkmal

„Modelle sind ihren Originalen nicht per se eindeutig zugeordnet. Sie erfüllen ihre Ersetzungsfunktion a) für bestimmte – erkennende und/oder handelnde, modellbenutzende – Subjekte, b) innerhalb bestimmter Zeitintervalle und c) unter Einschränkung auf bestimmte gedankliche oder tatsächliche Operationen.“⁵³ Ein Modell eines Originals existiert nicht zum Selbstzweck – „eine pragmatisch vollständige Bestimmung des Modellbegriffs hat nicht nur die Frage zu berücksichtigen, wovon etwas Modell ist, sondern auch, für wen, wann und wozu“⁵³. Der kybernetische Modellbegriff nach KLAUS⁵⁴, der Original, Modell und Subjekt in Verbindung setzt, wird damit um die Komponenten des Zeitbezugs und der **Zweckorientierung** ergänzt. Diese Ergänzung ist insbesondere dann relevant, wenn der Zeitpunkt der Modellbildung und der Modellnutzung wesentlich differieren. Dies ist oft dann der Fall, wenn Modellerschaffer und Modellbenutzer nicht identisch sind⁵⁵. Nach SCHLITT⁵⁶ wird hierdurch eine zusätzliche Verallgemeinerung des kybernetischen Modellbegriffs vorgenommen, da das Modellsubjekt in Modellerschaffer und Modellbenutzer differenziert wird.

⁵¹ vgl. Thomas 2005, S. 9

⁵² vgl. Stachowiak 1973, S. 132

⁵³ vgl. Stachowiak 1973, S. 132

⁵⁴ vgl. Klaus 1967, S. 413

⁵⁵ vgl. Stachowiak 1983, S. 118

⁵⁶ vgl. Schlitt 2004, S. 29

Definition 2-3: Modell⁵⁷

A Model is a purpose-dependent, finite, simplified, but still adequate representation of whatever is modelled, allowing us to abstract from its unimportant properties and details and to concentrate only on the most specific and most important traits.

Modellbildung und Modellnutzung

Durch die Differenzierung des Modellsubjekts in Modellerschaffer und Modellbenutzer wird nach SCHLITT gleichzeitig eine **Kommunikationsbeziehung** zwischen den beteiligten Aufgabenträgern postuliert⁵⁸. Diese Kommunikationsbeziehung hat einen wesentlichen Einfluss auf die Effizienz der Modellbildung einerseits und auf die Effektivität der Modellnutzung andererseits (vgl. Abbildung 2-4).

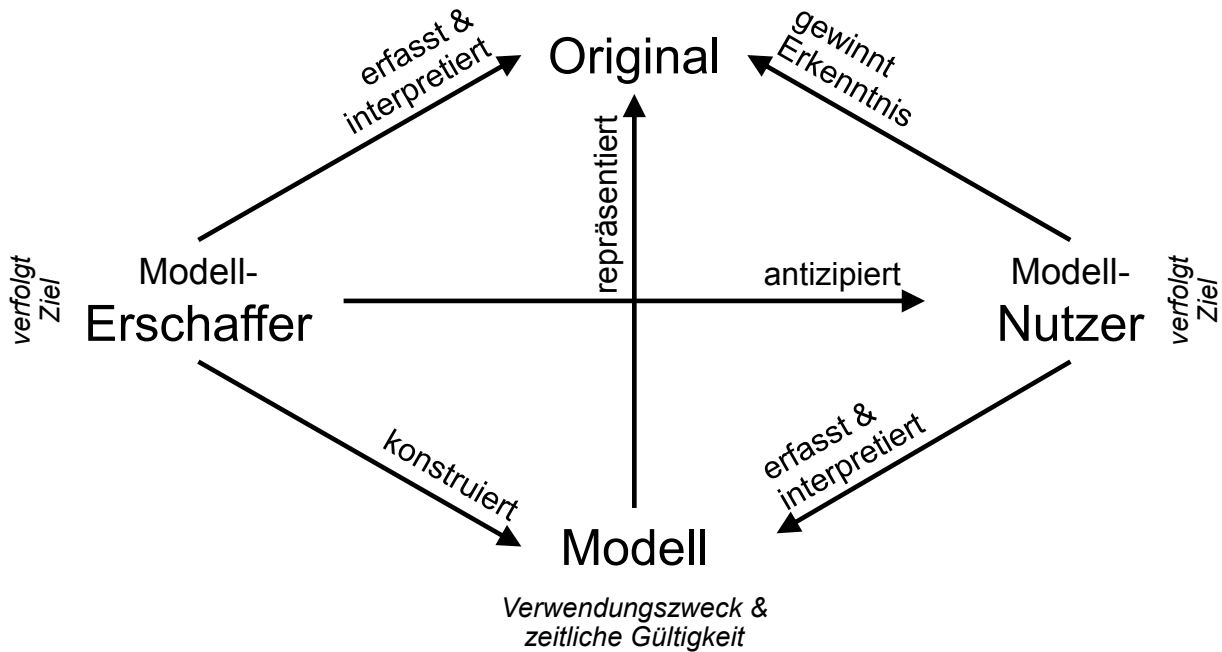


Abbildung 2-4: Modellbildung und Modellnutzung⁵⁹

In der Modellbildung erfasst und interpretiert der Modellerschaffer das abzubildende Objekt unter Nutzung interner Modelle⁶⁰ (vgl. Abbildung 2-5). Bereits bei der Erfassung des Objekt werden unvermeidliche Verkürzungen vorgenommen, da die „Aufnahmefähigkeit der menschlichen Sinne gegenüber einer komplexen Realität stark eingeschränkt sind.“⁶¹ Die Interpretation der erfassten Originalattribute erfolgt

⁵⁷ vgl. Avgoustinov 2007, S. 9

⁵⁸ vgl. Schlitt 2004, S. 29

⁵⁹ eigene Darstellung in Anlehnung an Klaus 1967 und Stachowiak 1973

⁶⁰ vgl. Klaus 1967, S. 287

⁶¹ vgl. Schlitt 2004, S. 31

vor dem Hintergrund des Modellierungszwecks und ist eng verbunden mit der Erfassung des Originals – „alle Perzeption ist denkbestimmt, umgekehrt ist alles Denken ursprünglich an perzeptuelle Anschauung gebunden“⁶². Der nachfolgende Übergang eines internen Modells in ein externes Modell ist nicht trivial und mit Übersetzungsproblemen verbunden⁶³ – das externe Modell muss aktiv und gestalterisch konstruiert werden. Der Schritt der Konstruktion erzeugt das ziel- und zweckkonforme, kommunikationsfähige Modell und ist maßgeblich am Modellierungsziel und -zweck ausgerichtet.

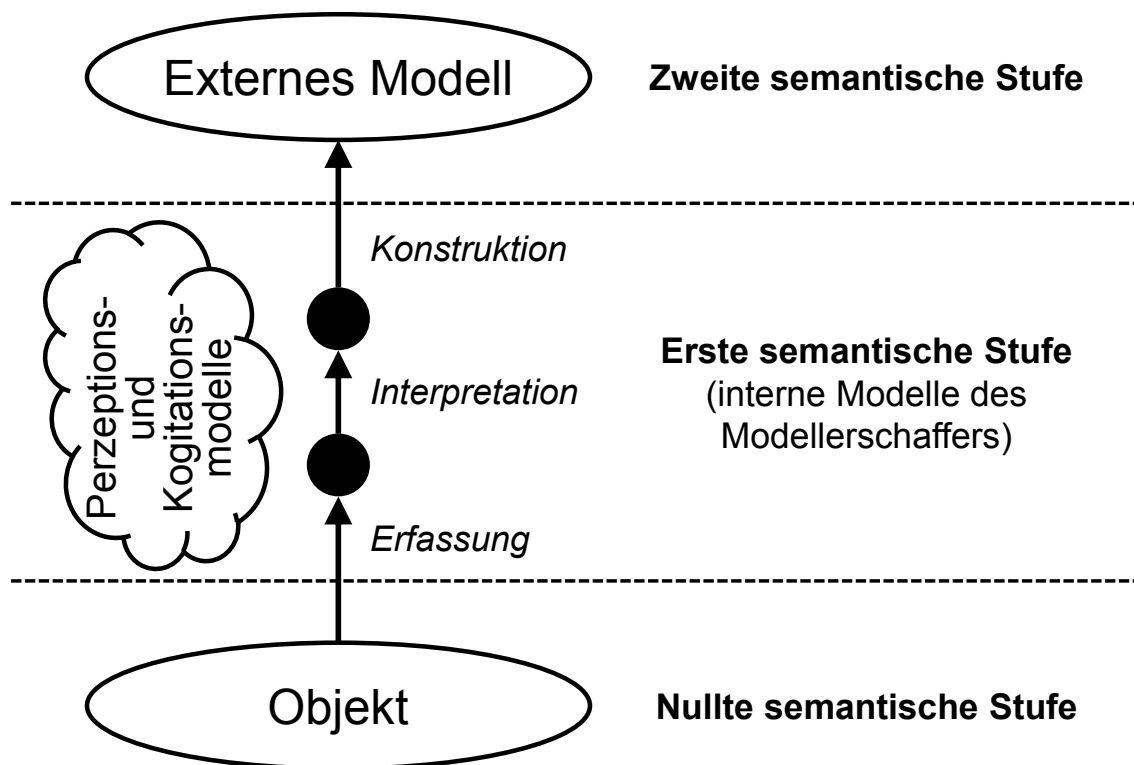


Abbildung 2-5: Semantische Stufen der Modellbildung⁶⁴

Die Effizienz der Modellbildung und die Effektivität der Modellnutzung sind stark davon abhängig, in welchem Maße der Modellerschaffer die spätere Modellnutzung bzw. den späteren Modellbenutzer antizipiert, d.h. in welchem Maße die **mentalen Modelle**⁶⁵ der Subjekte übereinstimmen. Sind Modellerschaffer und Modellbenutzer nicht identisch, müssen neben dem eigentlichen Abbild des Originals auch der Zweck und die zeitliche Gültigkeit des Modells externalisiert werden. Die Umfänglichkeit und Qualität dieser zusätzlichen Modellmerkmale sollten dabei zum einen auf die Ausprägung dieser Merkmale selbst und zum anderen auf die internen

⁶² vgl. Stachowiak 1973, S. 210

⁶³ vgl. Zschocke 1995, S. 102ff

⁶⁴ vgl. Stachowiak 1973

⁶⁵ vgl. Dutke 1994, S. 76ff

Modelle des Modellbenutzers abgestimmt sein. Gelingt es dem Modellerschaffer die Perzeptions- und Kogitationsmodelle des Modellbenutzers zu antizipieren, kann eine starke Verkürzung des externen Modells vorgenommen werden, da die fehlenden Attribute durch den Modellbenutzer korrekt ergänzt werden können. Diesem Potenzial steht das Risiko entgegen, dass der Modellbenutzer das Modell falsch oder unvollständig erfasst, unzulässig interpretiert und folglich zu falschen Schlüssen bzw. Erkenntnissen über das Original gelangt. Sind Modellerschaffer und Modellbenutzer identisch, kann davon ausgegangen werden, dass die internen Modelle ebenfalls nahezu identisch sind, so dass eine starke Verkürzung des externen Modells gewöhnlich keine negativen Auswirkungen auf die Modellnutzung haben. Ist die Zeitspanne zwischen Modellbildung und Modellnutzung hingegen groß, kann eine starke Modellverkürzung auch bei identischen Subjekten zu Problemen führen.

Durchgängigkeit und Konsistenz von Modellen

In der Anwendung von Modellen müssen oftmals mehrere Modelle emergent zusammenwirken, um den gewünschten Erkenntnisgewinn zu realisieren. Beispiele hierfür sind die Kopplung von domänenspezifischen Partialmodellen⁶⁶ oder die situationsspezifische Anpassung der Modellgranularität, um eine „zoomende Arbeitsweise“⁶⁷ zu ermöglichen. Diese situationsspezifische Änderung der Modellverkürzung ist der Tatsache geschuldet, dass ein menschlicher Modellbenutzer nur eine begrenzte Anzahl an Informationseinheiten (Chunks⁶⁸) gleichzeitig speichern kann. Grundvoraussetzung für ein solches Zusammenwirken mehrerer (Sub-)Modelle ist die integrale Qualität⁶⁹ der einzelnen Modelle, d.h. deren Kompatibilität untereinander⁷⁰. Bilden mehrere Modelle unterschiedliche Aspekte eines Originals ab, muss darüber hinaus sichergestellt werden, dass die Ausprägungen der einzelnen Modellattribute stets genau ein Original repräsentieren – die Modelle sich also nicht widersprechen. ALBERS und LOHMEYER⁷¹ sprechen in diesem Zusammenhang von **Durchgängigkeit** und **Konsistenz** von Modellen (vgl. Abbildung 2-6).

⁶⁶ vgl. Sander, Kellner & Albers 2013

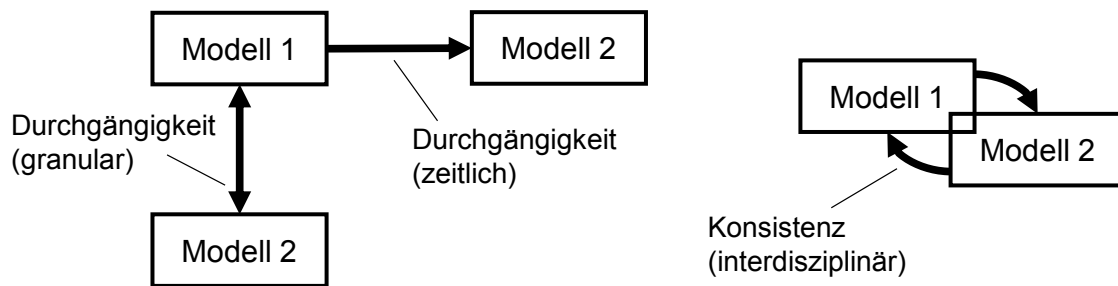
⁶⁷ vgl. Ehrlenspiel 2009, S. 79

⁶⁸ vgl. Miller 1956, S.81

⁶⁹ vgl. Pfeiffer 1965, S. 43

⁷⁰ vgl. auch Abschnitt 2.1.1

⁷¹ vgl. Albers & Lohmeyer 2012

Abbildung 2-6: Durchgängigkeit und Konsistenz von Modellen⁷²

2.1.3 Zwischenfazit

Mit der Allgemeinen Systemtheorie steht eine umfassende Theorie zur interdisziplinären Systembeschreibung zur Verfügung. Im Kern der Theorie stehen die drei Systemkonzepte nach ROPOHL die jeweils einen Aspekt des Systembegriffs aufgreifen und bei integrativer Anwendung ein ganzheitliches Denken in Systemen ermöglichen. Damit kann die allgemeine Systemtheorie insbesondere auch zur Beschreibung des komplexen, soziotechnischen Systems der Produktentstehung genutzt werden. Basis jeder Beschreibung ist dabei die Bildung und Nutzung von Modellen. Die Allgemeine Modelltheorie nach STACHOWIAK beschreibt drei Hauptmerkmale des allgemeinen Modellbegriffs und liefert damit eine Grundlage zur Untersuchung der Modellbildung und Modellnutzung im Prozess der Produktentstehung. Hierbei sind das pragmatische und das verkürzende Merkmal von besonderer Bedeutung, da hierdurch die Effizienz und Effektivität der Modellbildung und -nutzung maßgeblich beeinflusst wird.

2.2 Systemmodell der Produktentstehung

2.2.1 ZHO-Modell

Auf Basis der Allgemeinen Systemtheorie leitet ROPOHL⁷³ einen systematischen Ansatz zur Beschreibung der Produktentstehung ab. Hierzu unterteilt ROPOHL das sozio-technische System der Produktentstehung in drei miteinander wechselwirkende Systeme mit dem Anspruch, dass das resultierende Modell der Systemtechnik "an alle Erscheinungen und Probleme herangetragen werden [kann], mit denen der Ingenieur konfrontiert wird"⁷⁴ (vgl. Abbildung 2-7). Nach ROPOHL⁷⁵ wird im **Handlungssystem** ein **Sachsystem** (Objektsystem) gemäß einem **Zielsystem** verwirklicht. Dabei beeinflusst das Handlungssystem das Zielsystem, während das entstandene Sachsystem (Objektsystem) auf das Handlungssystem zurückwirkt oder

⁷² vgl. Lohmeyer 2013, S. 19

⁷³ vgl. Ropohl 2009

⁷⁴ vgl. Ropohl 1975, S. 32

gar dessen Bestandteil wird. Die natürliche, technische und gesellschaftliche Umgebung dieser Systeme spielt dabei eine nicht zu unterschätzende Rolle.

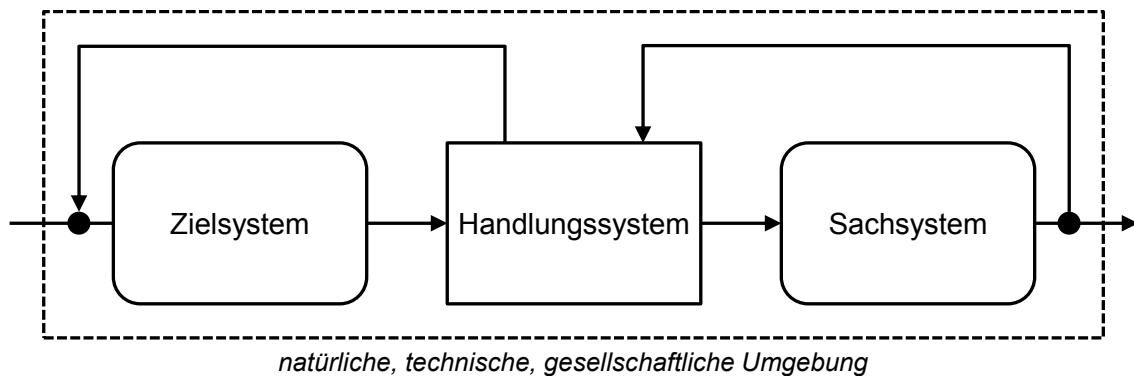


Abbildung 2-7: Systeme der Systemtechnik⁷⁵

MEBOLDT⁷⁶ nutzt in seiner Arbeit die Theorie der Systemtechnik als Grundlage für das **integrierte Produktentstehungsmodell (iPeM)**⁷⁷ und nennt folgende Stärken des ZHO-Modells in der Anwendung auf die Produktentstehung:

- Durch die systemtechnische Logik können die Zusammenhänge zwischen Ziel, Handlungen und Ergebnissen auf allen Ebenen der Produktentstehung beschrieben und modelliert werden.
- Es werden soziale Elemente gleichwertig zu technischen Elementen abgebildet.
- Es wird eine einheitliche Sprache geschaffen.
- Die ZHO-Systeme lassen sich in allen Disziplinen anwenden.
- Es wird ein formaler Modellstandard geschaffen.
- Das Modell erfüllt die Anforderung an Einfachheit und Komplexität.

Systeme des ZHO-Modells

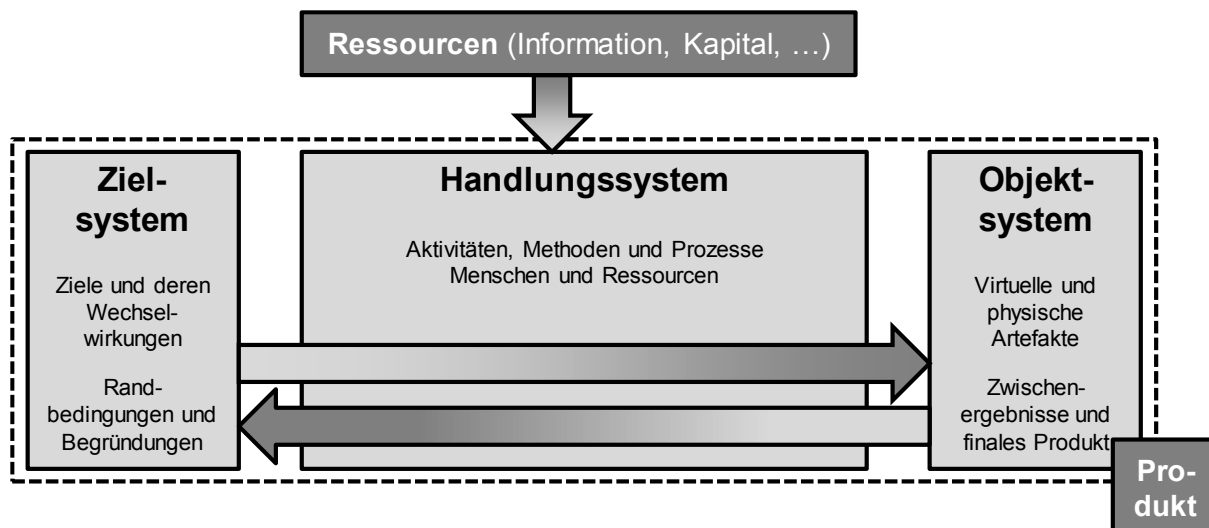
ALBERS⁷⁸ formuliert in seinen zentralen Hypothesen der Produktentstehung, dass auf Basis der Systemtheorie die Produktentstehung als Transformation eines anfangs vagen Zielsystems in ein Objektsystem durch ein Handlungssystem beschrieben werden kann. Diese drei Systeme stehen dabei in enger Wechselwirkung miteinander und bestimmen den Erfolg eines Produktentstehungsprozesses (vgl. Abbildung 2-8).

⁷⁵ vgl. Ropohl 1975, S. 33

⁷⁶ vgl. Meboldt 2008, S. 157

⁷⁷ vgl. Albers & Meboldt 2007

⁷⁸ vgl. Albers 2010

Abbildung 2-8: ZHO-Modell nach ALBERS⁷⁹

Das **Handlungssystem**⁸⁰ ist ein sozio-technisches System und besteht aus strukturierten Aktivitäten, Methoden und Prozessen. Weiterhin enthält es alle für die Realisierung der Produktentstehung erforderlichen Ressourcen wie z.B. (Entwicklungs-)Personal, Budget, Material, Maschinen und weitere. Das Handlungssystem erstellt das Zielsystem und das Objektsystem, die beide jeweils nur über das Handlungssystem miteinander verbunden sind.

Das **Zielsystem**⁸⁰ beinhaltet die mentale Vorstellung der geplanten Eigenschaften eines Produkts. Es umfasst alle relevanten Ziele, deren Wechselwirkungen und Randbedingungen, die für die Entwicklung des richtigen Produkts erforderlich sind. Darüber hinaus beinhaltet das Zielsystem auch die Begründungen für die enthaltenen Elemente, Attribute und Relationen. Die Ziele beschreiben dabei den gewünschten zukünftigen Zustand des Produkts (und seiner Komponenten) und dessen Kontext, nicht jedoch die Lösung als solche. Das Zielsystem wird im Verlauf eines Produktlebenszyklus⁸¹ und hierbei insbesondere in der Produktentstehung fortwährend erweitert und konkretisiert.

Das **Objektsystem**⁸⁰ enthält alle Dokumente und Artefakte, die als Teillösungen während des Entstehungsprozesses anfallen. Es ist vollständig, sobald der geplante Zielzustand erreicht ist. Das eigentliche Produkt ist neben Zwischenprodukten (z.B. Zeichnungen oder Prototypen) eines der Elemente des Objektsystems. Das Handlungssystem nimmt während des Produktentstehungsprozesses ständig Teile des Objektsystems auf und leitet daraus durch Schritte der Analyse und Synthese neue Ziele ab. Endergebnis eines Produktentstehungsprozesses ist neben dem

⁷⁹ vgl. Albers & Braun 2011a

⁸⁰ vgl. Albers & Braun 2011a

⁸¹ vgl. Gausemeier et al. 2000

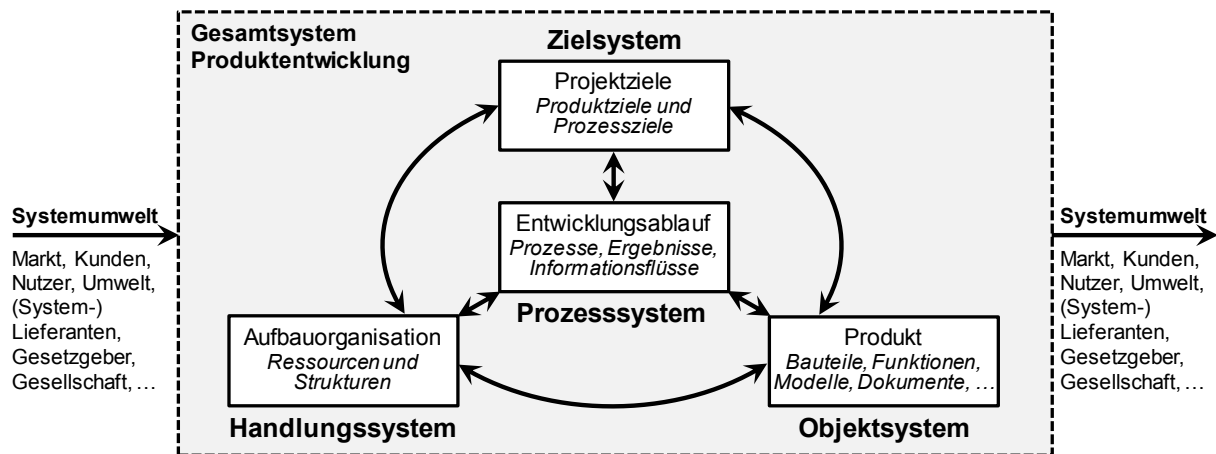
eigentlichen Produkt auch die Summe aller Zwischenergebnisse im Objektsystem, das vollständige Zielsystem und nicht zuletzt das gesammelte Erfahrungswissen im Handlungssystem aus der Durchführung und Reflexion des Prozesses.

ZHOP-Modell

PATZAK⁸² beschreibt das System der Produktentstehung durch ein System-Quadrupel aus **Zielsystem**, **Programmsystem**, **Wirksystem** und **Objektsystem**. Sein Verständnis von Ziel- und Objektsystem deckt sich dabei weitestgehend mit dem von ROPOHL bzw. ALBERS. Für PATZAK stellen Programmsysteme „die zu setzenden Aktivitäten als Mittel zur Zielerreichung dar. Sie liefern in Form von Plänen [...] Aufschluss über die Art und Weise der durchzuführenden Aktivitäten. [...] Wirksysteme sind [...] die Träger der Aktivitäten, haben daher für das Programmsystem unmittelbaren Mittelcharakter. Sie setzen sich in ihrer allgemeinen Form – als soziotechnische Systeme – aus belebten Komponenten (Mensch, sonstige Lebewesen) und unbelebten Komponenten (Sachmittel) zusammen, deren Zusammenwirken durch formelle, aber auch informelle Beziehungsmuster (Organisation) ermöglicht wird.“⁸² NEGELE⁸³ greift die Arbeiten von PATZAK auf und entwickelt auf Basis des System-Quadrupels den **ZOPH-Ansatz** zur ganzheitlichen Modellierung von Produktentwicklungssystemen. NEGELE nimmt dabei eine Umbenennung und weitere Präzisierung der einzelnen Systeme vor – das Programmsystem wird in Prozesssystem, das Wirksystem in Handlungssystem umbenannt, so dass der ZHOP-Ansatz die vier Systeme **Ziel-, Handlungs-, Objekt- und Prozesssystem** umfasst (vgl. Abbildung 2-9). PATZAK und NEGELE unterscheiden damit explizit zwischen Planung, Vernetzung und Steuerung von Aktivitäten (Programm- bzw. Prozesssystem) und der ausführenden Instanz von Aktivitäten (Wirk- bzw. Handlungssystem). Aufgrund der engen Vernetzung von Planung, Steuerung und Ausführung von Aktivitäten betrachtet ALBERS beide Aspekte integriert in einem Handlungssystem.

⁸² vgl. Patzak 1982, S. 30

⁸³ vgl. Negele 1998, S. 53

Abbildung 2-9: ZHOP-Modell der Produktentwicklung⁸⁴

2.2.2 Prinzipien des ZHO-Modells

LOHMEYER⁸⁵ analysiert auf Basis der Arbeiten von MEBOLDT⁸⁶ die Wechselwirkungen zwischen Ziel-, Handlungs- und Objektsystem und leitet hieraus drei Prinzipien ab, die ein vertieftes Verständnis von Produktentstehung auf Basis des ZHO-Modells ermöglichen.

Prinzip der indirekten Wechselwirkung

Das Prinzip der indirekten Wechselwirkungen besagt, dass jede „Handlung nur vollständig durch das Systemtripler Ziel-, Handlungs- und Objektsystem beschrieben werden kann. Dabei besteht zwischen Ziel- und Objektsystem nur eine indirekte gegenseitige Wechselwirkung, die über das Handlungssystem bestimmt wird“⁸⁷. LOHMEYER⁸⁵ betont, dass Ziel- und Objektsystem passiv-reaktive Systeme sind, die sich zwar nicht selbständig oder gegenseitig entwickeln, jedoch ein Systemverhalten aufweisen, das sich als Reaktion gegenüber den Aktivitäten des Handlungssystems als aktives System manifestiert. Indirekt existierende Wechselwirkungen zwischen Ziel- und Objektsystem müssen folglich aktiv durch das Handlungssystem aufgezeigt und gestaltet werden.

Prinzip der doppelten Kontingenz

Das Prinzip der doppelten Kontingenz besagt, dass sowohl das Ziel- als auch das Objektsystem anfangs kontingent d.h. unbestimmt und offen sind. Erst durch die gleichwertige Kopplung der beiden Systeme kann ein Produkt zielgerichtet entwickelt

⁸⁴ vgl. Negele 1998, S. 139

⁸⁵ vgl. Lohmeyer 2013, S. 25

⁸⁶ vgl. Meboldt 2009

⁸⁷ vgl. Meboldt 2009, S. 194

werden.⁸⁸ Dies ist auf die indirekte Wechselwirkung zwischen Ziel- und Objektsystem im zeitlichen Kontext, d.h. im Verlauf der Produktentstehung zurückzuführen. Anfangs vage definierte Ziele bestimmen die Entwicklung erster Objekte. Die folgende Konkretisierung oder Änderung des Zielsystems wird maßgeblich durch Erkenntnisse aus der Systemumwelt oder durch Erkenntnisse aus den zuvor definierten Objekten bestimmt. Folglich führt die zielgerichtet Wahl zweckmäßiger Objekte durch das Handlungssystem zu einer Konkretisierung oder Änderung des Zielsystems. Durch diese Kopplung bestimmen sich das Ziel- und Objektsystem mit zunehmender Entwicklungsreife gegenseitig. Hierdurch reduziert sich die Kontingenz und es entsteht Ordnung, aus der letztendlich das Produkt hervorgeht.⁸⁹

Prinzip der zweckmäßigen Zuordnung

Das Prinzip der zweckmäßigen Zuordnung besagt, dass ein Element nicht absolut dem Ziel-, Handlungs- oder Objektsystem zugeordnet ist. Die Zuordnung ist relativ und hängt immer vom Zeitpunkt, Zweck und Standpunkt ab.⁹⁰ Dieses Prinzip betont den Modellcharakter des ZHO-Modells – insbesondere das pragmatische Merkmal, d.h., dass der Modellbenutzer zweckorientiert Erkenntnisse über das abgebildete Original gewinnen können muss. Folglich ist weniger die zweifelsfrei richtige Zuordnung eines Elements von Bedeutung, sondern vielmehr die Tatsache, dass der bzw. die Modellbenutzer diese Zuordnung zum Erkenntnisgewinn nutzen können. Beispiele sind Prüfstände, die als Entwicklungsergebnis dem Objektsystem zuzuordnen sind, im Falle ihrer Anwendung zum Zwecke der Verifizierung jedoch als Teil des Handlungssystems zu sehen sind. Ein weiteres Beispiel sind Anforderungslisten, die als physisches Produkt dem Objektsystem zuzuordnen sind, deren Inhalte jedoch Teil des Zielsystems sind.

2.2.3 Erweitertes ZHO-Modell

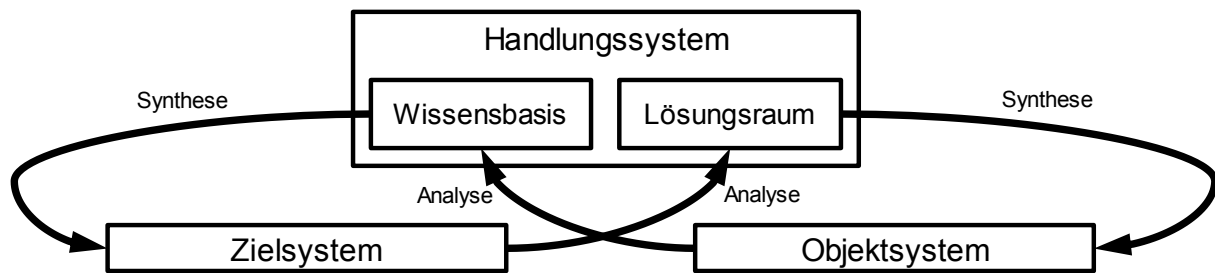
ALBERS und LOHMEYER⁹¹ liefern mit dem erweiterten ZHO-Modell ein menschenzentriertes Erklärungsmodell, das den Entwickler als denkenden und handelnden Menschen im Mittelpunkt eines unsicherheitsbehafteten und iterativ verlaufenden Produktentstehungsprozess beschreibt. Hierzu fokussieren sie in der Betrachtung des Handlungssystems auf das Individuum Entwickler und verkürzen diesen auf seine **Wissensbasis** und seinen **Lösungsraum** (vgl. Abbildung 2-10).

⁸⁸ vgl. Meboldt 2008, S. 195

⁸⁹ vgl. Lohmeyer 2013, S. 26

⁹⁰ vgl. Meboldt 2008, S. 156

⁹¹ vgl. Albers, Lohmeyer & Ebel 2011

Abbildung 2-10: Erweitertes ZHO-Modell⁹¹

LOHMEYER definiert Wissensbasis als „die Menge an Domänenwissen und fallspezifischem Wissen, welches innerhalb des Handlungssystems eines bestimmten Produktentstehungsprozesses bereitsteht.“⁹² „Der Lösungsraum beschreibt die Menge aller zulässigen Lösungen zu einem Problem und richtet so die mentale, virtuelle und physische Modellierung von Objekten aus.“⁹³ Um die Wechselwirkungen zwischen den Subsystemen Wissensbasis und Lösungsraum mit dem Ziel- und Objektsystem zu modellieren, greift LOHMEYER auf die Konzepte der Analyse und Synthese zurück, die er wie folgt definiert:

Definition 2-4: Analyse

Die Analyse beschreibt eine Handlung, die das Verstehen eines existierenden Systems bezweckt. Das Resultat einer Analyse ist Erkenntnis.⁹⁴

Definition 2-5: Synthese

Die Synthese beschreibt eine Handlung, die das Erschaffen eines bis dato noch nicht existierenden Systems bezweckt. Das Resultat einer Synthese ist ein Ziel oder ein Objekt.⁹⁴

Auf dieser Basis beschreibt Lohmeyer die Produktentstehung als **iterativen Prozess der co-evolutionären Entwicklung** von Ziel- und Objektsystem durch kontinuierliche Analyse- und Syntheseaktivitäten des Handlungssystems⁹⁵. Zu Beginn eines Produktentstehungsprozesses gilt es, das fallspezifische Wissen der Wissensbasis derart zu vermehren, dass Folgeaktivitäten unter akzeptabler Unsicherheit abgeleitet werden können. Hierzu werden bestehende Objekte, d.h. Informationsträger, wie z.B. bestehende Produkte oder abgeschlossene Projektdokumentationen, analysiert. Im Verlauf des Produktentstehungsprozesses entstehen zusätzliche Objekte, wie z.B.

⁹² vgl. Lohmeyer 2013, S. 91

⁹³ vgl. Lohmeyer 2013, S. 99

⁹⁴ vgl. Lohmeyer 2013, S. 108

⁹⁵ vgl. Lohmeyer 2013, S. 115

virtuelle oder physische Prototypen, durch deren Analyse weiteres fallspezifisches Wissen aufgebaut werden kann. Durch das vertiefte fallspezifische Wissen ist es dem Handlungssystem/Entwickler im Folgenden möglich, Elemente des Zielsystems festzulegen, zu verfeinern oder zu ändern und damit Definitionslücken zu schließen. Hierbei entscheidet das Handlungssystem bzw. der Entwickler basierend auf seiner Wissensbasis, welche Elemente aus welchem Grund Teil des Zielsystems werden. Die explizierten Elemente des Zielsystems werden im Anschluss durch das Handlungssystem bzw. dem (anderen) Entwickler erfasst und interpretiert bzw. mental redefiniert. Hieraus bildet der Entwickler die Begrenzung seines mentalen Lösungsraums, d.h. die Menge der zulässigen Lösungen. Innerhalb des eingegrenzten Lösungsraums entwickelt das Handlungssystem schließlich Lösungen, die zunächst mental vorgedacht und dann in Form expliziter Modelle, z.B. Skizzen, Diagrammen oder Prototypen, synthetisiert werden. Mit diesem Syntheseschritt bildet sich ein geschlossener Kreislauf, der Produktentstehung als iterativen und co-evolutionären Prozess beschreibt⁹⁶.

2.2.4 Zwischenfazit

Produktentstehung lässt sich auf Basis der Systemtheorie als System-Tripel beschreiben, das aus drei miteinander wechselwirkenden Systemen besteht. Dabei steht nur das Handlungssystem in direkter Wechselwirkung mit dem Ziel- und Objektsystem. Indirekte Wechselwirkungen zwischen den beiden Systemen müssen für eine erfolgreiche Produktentstehung durch das Handlungssystem kontinuierlich aufgezeigt und gestaltet werden, da sich Ziel- und Objektsystem mit zunehmender Entwicklungsreife gegenseitig bestimmen. Die hierfür notwendigen kontinuierlichen Analyse- und Synthese-Aktivitäten werden durch das erweiterte ZHO-Modell begründet, das damit in der Lage ist, den iterativen und co-evolutionären Produktentstehungsprozess zu beschreiben. Das Modell rückt den Entwickler in den Mittelpunkt der Betrachtung und ermöglicht es damit, die Analyse und Synthese des Zielsystems in einem komplexen und unsicherheitsbehafteten Prozess genauer zu beschreiben und hierauf aufbauend neue Methoden zur Zielsystemmodellierung abzuleiten.

⁹⁶ vgl. Albers, Lohmeyer & Ebel 2011

2.3 Komplexität, Unsicherheit und Iterationen

2.3.1 Komplexität in der Produktentstehung

Der Begriff Komplexität leitet sich vom lateinischen Wort „complexum“ ab, das „umschlingen“, „umfassen“ aber auch „ineinander flechten“ bedeutet⁹⁷ und wird oftmals intuitiv und undefiniert verwendet. Komplexität beschreibt im Allgemeinen eine Situation, die so viele Aspekte beinhaltet, dass sie unverständlich, unbegreiflich oder verwirrend erscheint⁹⁸. Aufbauend auf diesem Verständnis beschreibt DÖRNER⁹⁹ **Komplexität als subjektive Größe** – „to a certain extent, complexity is in the eye of the beholder: what is complex for one observer, may be simple for another one“¹⁰⁰. Nach MEBOLDT¹⁰¹ basiert diese Ansicht auf dem konstruktivistischen Verständnis, dass Komplexität erst durch den Betrachter entsteht und ein Maß für die Unbestimmbarkeit, einen Möglichkeitsüberschuss bzw. einen Mangel an Informationen darstellt. Nach LUHMANN entsteht ein komplexes System, wenn auf Grund der beschränkten Relationierbarkeit der Elemente nicht mehr jedes Element jederzeit mit jedem anderen verknüpft sein kann.¹⁰²

Auf Basis der Systemtheorie legt SIMON ein weitaus weniger subjektives Verständnis des Komplexitätsbegriffs vor. Er versteht unter einem komplexen System „one made up of large number of parts that have many interactions“.¹⁰³ Dieses Verständnis findet sich ebenfalls in den Arbeiten von PATZAK¹⁰⁴ wieder, der Komplexität als Zusammenwirken von **Varietät** (Elementvielfalt) und **Konnektivität** (Beziehungsvielfalt) versteht. Die Varietät setzt sich dabei aus den unterschiedlichen Arten und der Anzahl von Elementen zusammen, die Konnektivität aus den unterschiedlichen Arten und der Anzahl von Beziehungen zwischen den Elementen. Das Zusammenwirken von Varietät und Konnektivität macht dabei die **strukturelle Komplexität**¹⁰⁵ eines Systems aus (vgl. Abbildung 2-11).

⁹⁷ vgl. Kluge 2002

⁹⁸ vgl. La Porte 1975, S. 5

⁹⁹ vgl. Dörner 1998, S. 61

¹⁰⁰ vgl. Heylighen 1999

¹⁰¹ vgl. Meboldt 2008, S. 79

¹⁰² vgl. Luhmann 1994, S. 66

¹⁰³ vgl. Simon 1994, S. 183

¹⁰⁴ vgl. Patzak 1982, S. 22

¹⁰⁵ vgl. Biedermann, Diepold & Lindemann 2010

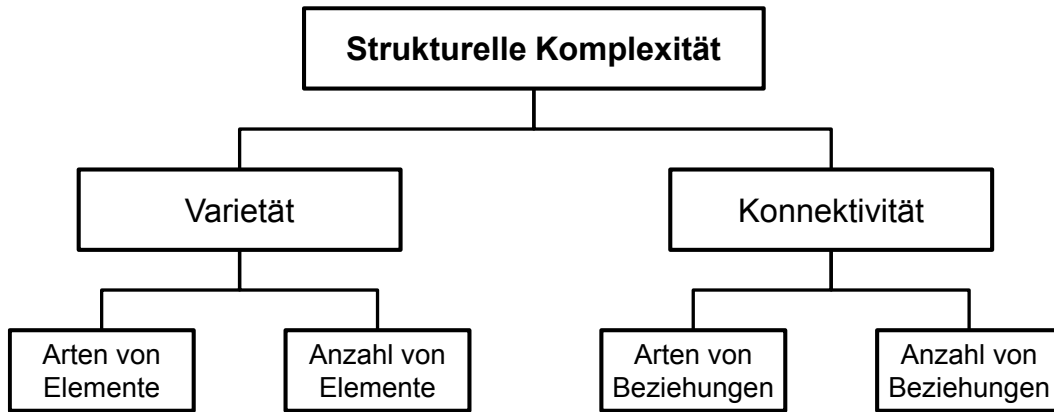


Abbildung 2-11: Strukturelle Komplexität¹⁰⁴

Die Komplexität eines Systems wird neben der Varietät und Konnektivität der enthaltenen Systemelemente noch durch deren Veränderlichkeit und Dynamik geprägt, die SCHUH¹⁰⁶ als **dynamische Komplexität** bezeichnet. Dabei hat neben der Anzahl und Art der Zustände innerhalb eines Systems auch die Variabilität der Struktur einen wesentlichen Einfluss auf die dynamische Komplexität¹⁰⁷. Die Dynamik und Veränderung eines Systems wird dabei stark von der Offenheit der Systemgrenze und damit der Interaktion zu anderen Systemen beeinflusst¹⁰⁸.

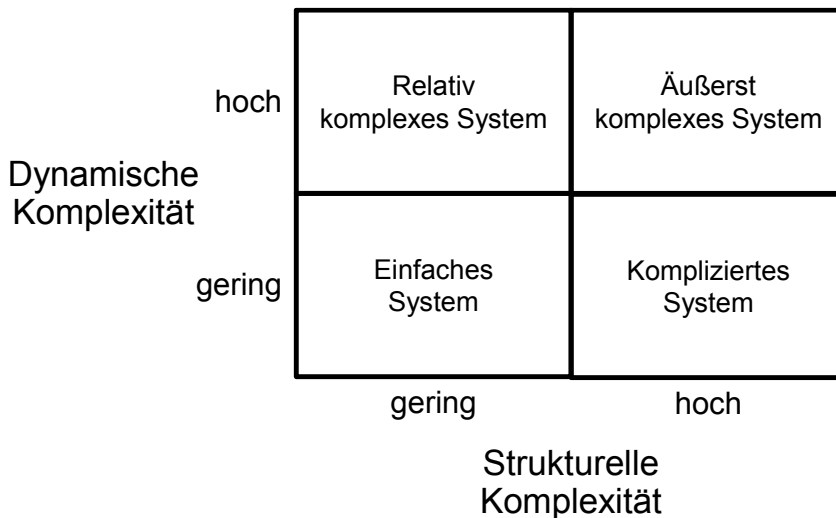


Abbildung 2-12: Strukturelle und dynamische Komplexität¹⁰⁹

SCHUH¹⁰⁹ unterscheidet auf Basis der Ausprägung struktureller und dynamischer Komplexität zwischen vier Typen von Systemen (vgl. Abbildung 2-12):

¹⁰⁶ vgl. Schuh 2005, S. 6

¹⁰⁷ vgl. Dörner 2003, S. 62

¹⁰⁸ vgl. Felgen 2007, S. 15

¹⁰⁹ vgl. Schuh 2005, S. 6

- **einfaches System** (wenig Elemente, Beziehungen und Verhaltensmöglichkeiten)
- **kompliziertes System** (viele Elemente und Beziehungen; Verhalten ist deterministisch)
- **relativ komplexes System** (wenig Elemente und Beziehungen; hohe Vielfalt an Verhaltensmöglichkeiten; keine vollständige Beherrschbarkeit möglich)
- **äußerst komplexes System** (Vielzahl von Elementen mit vielfältigen Beziehungen; große Vielfalt an Verhaltensmöglichkeiten mit veränderlichen Wirkungsverläufen zwischen den Elementen)

WEBER¹¹⁰ greift dieses allgemeingültige Verständnis von Komplexität auf und schlägt für die Produktentstehung eine Unterteilung in **Produkt- und Prozesskomplexität** vor. Produktkomplexität resultiert dabei aus der Anzahl und Vernetzung von Komponenten sowie der Variantenzahl eines Produkts. Als Komplexitätstreiber im Prozess nennt WEBER die Organisation und die an der Produktentstehung beteiligten Disziplinen. Nach LINDEMANN ET AL.¹¹¹ sind in der Produktentstehung Markt-, Produkt-, Prozess- und Organisationskomplexität ganzheitlich zu betrachten. Dabei werden insbesondere die starken Wechselwirkungen zwischen Produkt- und Prozesskomplexität und die Notwendigkeit einer gemeinsamen Betrachtung häufig hervorgehoben¹¹². Bei der Betrachtung von Komplexität muss dabei zwischen **interner** (beeinflussbarer) und **externer** (nicht beeinflussbarer) Komplexität unterschieden werden¹¹³, da dies starken Einfluss auf den Umgang mit Komplexität hat.

Im Umgang mit Komplexität kann prinzipiell zwischen zwei Strategien unterschieden werden – Komplexitätsreduktion und Komplexitätsmanagement¹¹⁴. **Komplexitätsreduktion** hat zum Ziel, Komplexität zu vermeiden oder zu reduzieren. Diese Strategie ist jedoch nur bei einfachen und komplizierten Systemen sinnvoll anwendbar, da für eine Komplexitätsreduktion das System vollständig beherrscht werden muss. Dies ist oftmals nicht der Fall, da nicht alle systemcharakterisierende und systembeeinflussende Attribute gleichzeitig erfasst werden können, die Attribute nicht zugänglich sind oder nicht eindeutig erfasst werden können¹¹⁵. Darüber hinaus besteht bei einer Komplexitätsreduktion stets die Gefahr, dass emergente Systemeigenschaften verloren gehen. Insofern ist Komplexität nicht grundsätzlich negativ zu sehen, sondern oftmals erforderlich, um gewünschte Systemeigenschaften zu realisieren.

¹¹⁰ vgl. Weber 2005, S. 4

¹¹¹ vgl. Lindemann, Maurer & Braun 2009, S. 27

¹¹² vgl. z.B. Göpfert 1998, Baldwin & Clark 2000, Eppinger & Salminen 2001

¹¹³ vgl. Schuh & Schwenk 2001, S. 10

¹¹⁴ vgl. Lohmeyer 2013, S. 9

¹¹⁵ vgl. Wegehaupt 2004, S.39

„**Komplexitätsmanagement** ist die angemessene Handhabung der Komplexität um Ziele zu erreichen und das Potenzial komplexer Kompositionen zu nutzen“¹¹⁶. Komplexitätsmanagement hat folglich zum Ziel, mit der erforderlichen Komplexität bestmöglich umzugehen und komplexe Systeme, auch solche, die sich einer vollständigen Kontrolle entziehen, handhabbar zu machen¹¹⁷. Ein wesentlicher Schlüssel für den erfolgreichen Umgang mit Komplexität ist die Bildung und Nutzung geeigneter Modelle – „every good regulator of a system must be a model of that system.“¹¹⁸ Dabei darf die Komplexität erst durch die Fremdkomplexität des abzubildenden Realitätsbereiches entstehen, nicht durch die Eigenkomplexität des (Meta-)Modells selbst, mit dem die Realität abgebildet wird¹¹⁹. Komplexität kann nur auf Basis der einfachen dynamischen Modellbildung beherrscht werden¹²⁰. Für die Modellbildung komplexer Systeme bedeutet dies, dass das Meta-Modell auf einfachen und eindeutigen Regeln beruhen muss, die in der Modellbildung dynamisch angewendet werden können¹²¹.

2.3.2 Unsicherheit in der Produktentstehung

Komplexität in der Produktentstehung steht in starker Wechselwirkung mit der vorherrschenden Unsicherheit. So geht MALIK¹²² davon aus, dass Unsicherheit ein Merkmal von Komplexität ist. Hieraus folgert MAURER, dass eine verringerte Komplexität zu geringeren Unsicherheiten führt¹²³. SUH¹²⁴ hingegen nimmt an, dass Komplexität erst durch Unsicherheit entsteht. Komplexität und Unsicherheit verbindet, dass sie in der Produktentstehung allgegenwärtig sind und nicht ignoriert werden dürfen¹²⁵.

DE WECK¹²⁵ versteht unter Unsicherheit ein „amorphous concept that is used to express both the probability that certain assumptions made during design are incorrect as well as the presence of entirely unknown facts that might have a bearing on the future state of a product or system and its success in the marketplace“. EARL ET AL.¹²⁶ teilen Unsicherheit in vier Kategorien ein, bekannte Unsicherheiten (known

¹¹⁶ vgl. Puhl 1999, S. 23

¹¹⁷ vgl. Albers, Deigendesch & Meboldt 2008

¹¹⁸ vgl. Conant & Ashby 1970, S. 89.

¹¹⁹ vgl. Meboldt 2008, S. 151

¹²⁰ vgl. Bolz 2007, S. 27

¹²¹ vgl. Meboldt S. 2008, S. 153

¹²² vgl. Malik 2003, S. 292

¹²³ vgl. Maurer 2007, S. 33

¹²⁴ vgl. Suh 1999

¹²⁵ vgl. de Weck, Eckert & Clarkson 2007

¹²⁶ vgl. Earl et al. 2005

uncertainties), unbekannte Unsicherheiten (unknown uncertainties), Unsicherheit der Beschreibung (uncertainties in the description) und Unsicherheit der Daten (uncertainties in the data). Bekannte Unsicherheiten sind vorhersehbar, können beschrieben und in letzter Konsequenz auch aufgelöst werden. Unbekannte Unsicherheiten sind per Definition unvorhersehbar und können daher nicht unmittelbar aufgelöst werden. Unsicherheiten der Beschreibung entstehen bei Verwendung eines Modells aufgrund dessen Verkürzung oder aufgrund eines unklaren Verwendungszwecks. (Modell-Unsicherheit¹²⁷). Unsicherheit der Daten beschreibt die Unvollständigkeit und Ungenauigkeit vorhandener bzw. genutzter Daten (Daten-Unsicherheit¹²⁷). Dieses Verständnis findet sich im Wesentlichen auch bei HASTINGS und MCMANUS¹²⁸ wieder, die jedoch zwischen lediglich drei Kategorien von Unsicherheit unterscheiden und diese als Kontinuum verstehen (vgl. Abbildung 2-13). Auf der einen Seite des Kontinuums steht dabei die **statistische Unsicherheit** (statistically characterized variables/phenomena), die dadurch gekennzeichnet ist, dass sie sich statistisch erfassen und klar eingrenzen lässt. In der Mitte des Kontinuums befindet sich die **bekannte Unsicherheit** (known unknowns), die sich zwar prinzipiell eingrenzen lässt, jedoch teilweise aus vollständig unbekanntem Werten besteht. Mit entsprechendem (Zeit-)Aufwand lässt sich die bekannte Unsicherheit in statistische Unsicherheit überführen – meist wird bekannte Unsicherheit jedoch qualitativ oder semi-analytisch gehandhabt. Auf der anderen Seite des Kontinuums befinden sich die **unbekannten Unsicherheiten**, die auch HASTINGS und MCMANUS als per Definition unbekannt und unvorhersehbar charakterisieren. Darüber hinaus betrachten HASTINGS und MCMANUS¹²⁸ Unsicherheit aus einer Entwickler-Perspektive und leiten hieraus ab, dass Unsicherheit entweder in Form von **Wissenslücken** (lack of knowledge) oder in Form von **Definitionslücken** (lack of definition) vorliegt. Wissenslücken liegen vor, wenn Informationen, die für eine vernünftige und begründete Systementwicklung erforderlich sind bzw. wären, nicht oder nur unpräzise vorliegen. Definitionslücken liegen vor, wenn relevante Entscheidungen (noch) nicht getroffen sind und/oder das System (noch) unzureichend spezifiziert ist.

Im Umgang mit Unsicherheit kann prinzipiell zwischen zwei Strategien unterschieden werden¹²⁹:

- Verringerung der Unsicherheit
- Schutz des Systems

¹²⁷ vgl. Kreye, Goh & Newnes 2011

¹²⁸ vgl. Hastings & McManus 2004

¹²⁹ vgl. de Neufville 2004

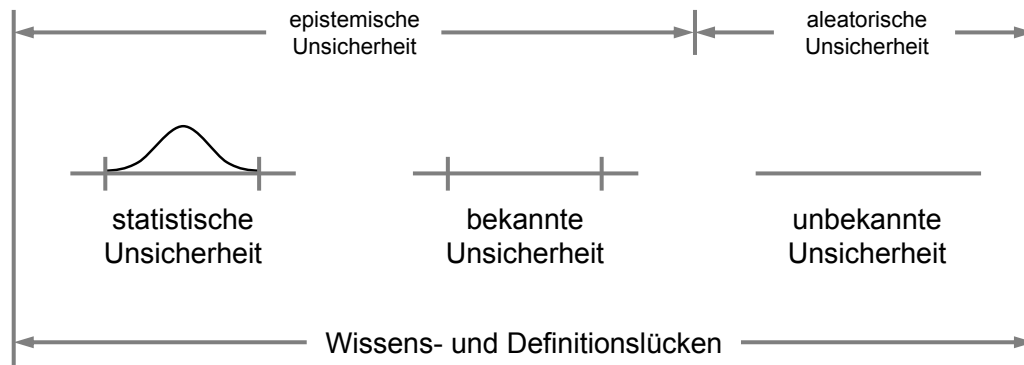


Abbildung 2-13: Unsicherheit in der Produktentstehung

Verringerung der Unsicherheit hat zum Ziel, Wissenslücken zu schließen und damit die Auftrittswahrscheinlichkeit eines unerwünschten Ereignisses zu minimieren, beispielsweise durch eine weiterführende Analyse des Systems, der Akquise weiterer Daten oder der Anwendung von Risiko-Management-Methoden¹³⁰. ALBERS¹³¹ betont in diesem Zusammenhang die Bedeutung der Validierung als zentrale Aktivität der Produktentstehung zur Schließung von Wissenslücken. **Schutz des Systems** hat zum Ziel, die Auswirkungen von Unsicherheit zu reduzieren, nicht die Unsicherheit selbst. Ein aktiv geschütztes System ist in der Lage, sich selbstständig an geänderte Randbedingungen anzupassen und somit Unsicherheiten abzufangen. LINDEMANN¹³² sieht hierbei in flexiblen Produktentstehungsprozessen und interdisziplinären Entwicklerteams zwei wesentliche Möglichkeiten, um sozio-technische System aktiv gegen Unsicherheit zu schützen. ALBERS¹³³ sieht in einer **kontinuierlichen Validierung**, die im Prozess der Produktentstehung verankert ist, einen Schlüssel, um mit Unsicherheit erfolgreich umzugehen. Passiv geschützte Systeme sind in der Lage, auch ohne Anpassung ihrer inneren Struktur, beispielsweise durch Wahl entsprechender Sicherheitsfaktoren, den Auswirkungen von Unsicherheit zu widerstehen.

Die Wahl der Strategie wird dabei insbesondere von vier Faktoren beeinflusst:

- Höhe der Unsicherheit
- Grad der Beeinflussbarkeit der Unsicherheit
- Aufwand der Beeinflussung der Unsicherheit
- Höhe des Risikos als direkte Auswirkung der Unsicherheit

¹³⁰ vgl. Chalupnik, Wynn, Clarkson 2013

¹³¹ vgl. Albers 2010, Albers & Düser 2010

¹³² vgl. Lindemann & Lorenz 2008

¹³³ vgl. Albers, Ebel & Lohmeyer 2012

Die Einschätzung der Höhe der vorliegenden Unsicherheit gestaltet sich oftmals schwierig, da sich nur im Falle statistischer Unsicherheit objektive Aussagen ableiten lassen. In anderen Fällen muss häufig auf subjektive Bewertungen z.B. Experteneinschätzungen zurückgegriffen werden¹³⁴. Der Grad der Beeinflussbarkeit beschreibt, inwiefern sich die Unsicherheit reduzieren lässt und bewegt sich zwischen den Polen reduzierbar (epistemische Unsicherheit¹³⁵) und nicht-reduzierbar (aleatorische Unsicherheit¹³⁵). Dabei ist insbesondere die Quelle der Unsicherheit bezüglich des betrachteten Systems relevant, da Unsicherheiten diesseits der Systemgrenze (endogene Unsicherheit) zumeist einfacher zu beeinflussen sind als Unsicherheiten jenseits der Systemgrenze (exogene Unsicherheit)¹³⁶. Prinzipiell beeinflussbare Unsicherheit lässt sich mit entsprechendem Zeit- und Kostenaufwand reduzieren¹³⁷ – Beeinflussbarkeit und Aufwand der Beeinflussung stehen jedoch nicht in einem konstanten Abhängigkeitsverhältnis und müssen daher separat bewertet werden. Risiko wird häufig als direkte Auswirkung von Unsicherheit benannt¹³⁸ und muss daher bei der Wahl einer Strategie zum Umgang mit Unsicherheit ebenfalls berücksichtigt werden. Risiko wird dabei als Funktion¹³⁹ bzw. Produkt¹⁴⁰ der Eintrittswahrscheinlichkeit und der Konsequenzen des Eintritts definiert. HASTINGS und McMANUS¹⁴⁰ betonen, dass sowohl **Risiken** (negative Konsequenzen) als auch unerwarteten **Chancen** (positive Konsequenzen) vorliegen können.

2.3.3 Iterationen in der Produktentstehung

Unsicherheiten im Produktentstehungsprozess führen zu Fehleinschätzungen, Fehlentscheidungen und schlussendlich zu Änderungen und Iterationen. ALMEFELT ET AL.¹⁴¹ nennen mögliche Ursachen für **Änderungen im Produktentstehungsprozess**:

- Im Projekt erzeugtes Wissen (beispielsweise durch Testen)
- Anforderungen, die sich als widersprüchlich herausstellen
- Technische Probleme
- Synergiepotenziale
- Unerwartete Vorgaben zur Kosteneinsparung
- Neue Gesetzeslage

¹³⁴ vgl. Wiebel et al. 2013

¹³⁵ vgl. Muschik 2011, S. 29

¹³⁶ vgl. Chalupnik, Wynn & Clarkson 2009

¹³⁷ vgl. Hastings & McManus 2004, S. 4

¹³⁸ vgl. Hubbard 2007, S. 46, Thunnissen 2005, S. 28, Hastings & McManus 2004, S. 2

¹³⁹ vgl. Thunnissen 2005, S. 28

¹⁴⁰ vgl. Hastings & McManus 2004, S. 5

¹⁴¹ vgl. Almfefelt et al. 2006

- Unerwartete Wettbewerbssituation
- Geänderte Marktbedürfnisse bzw. Kundenanforderungen

Diese Ursachen verdeutlichen, dass sich ein erheblicher Anteil der Unsicherheit erst im Prozess der Produktentstehung beseitigen lässt, indem Entscheidungen getroffen werden¹⁴², das System zunehmend detailliert und konkretisiert und somit Wissen erzeugt wird¹⁴³. ALBERS¹⁴⁴ geht davon aus, dass aus diesem Grund das Zielsystem im Prozess der Produktentstehung kontinuierlich erweitert wird und nennt mit Festlegen, Verfeinern und Ändern drei **Erweiterungsmechanismen des Zielsystems**. Hierbei werden Erweiterungen im Zielsystem insbesondere im Zuge der Validierung vorgenommen, die je nach Ergebnis der Validierung als Festlegung, Verfeinerung und/oder Änderung ausgeprägt sind. Daher bestimmen die erzeugten (Zwischen-) Ergebnisse sowie die (hieraus) gewonnenen Erkenntnisse das weitere Vorgehen im Produktentstehungsprozess¹⁴⁵.

Aufgrund von Unsicherheiten ist weder das Produkt noch der Produktentstehungsprozess vollständig planbar – Unsicherheiten und Änderungen sind Charakteristika des Prozesses¹⁴⁶. Die Konkretisierung und Detaillierung des zu entwickelnden Systems verläuft daher selten linear ab, sondern in Form von Iterationen¹⁴⁷. Auf Basis des ZHO-Modells versteht LOHMEYER¹⁴⁷ unter Iteration das vollständige Durchlaufen der vier Schritte Analyse des Ziels, Synthese des Objekts, Analyse des Objekts und Synthese des Ziels, wobei der Ausgangspunkt je nach Kontext variieren kann. WYNN¹⁴⁸ unterscheidet zwischen sechs nicht-orthogonalen **Perspektiven von Iterationen**:

- Exploration: Iteration zum Erkunden von Problem- und Lösungsraum
- Konvergenz: Iteration zur Annäherung an einen definierten Zielzustand
- Verfeinerung: Iteration zur Optimierung im Detail
- Nachbesserung: Iteration zur Anpassung an geänderte Randbedingungen
- Verhandlung: Iteration zur Auflösung von Zielkonflikten
- Wiederholung: Iteration im Sinne von wiederholtem ähnlichen Vorgehen

WYNN, ECKERT und CLARKSON¹⁴⁹ beschreiben, dass einzelne Iterationstypen charakteristisch für gewisse Abschnitte im Produktentstehungsprozess sind. Während zu Beginn des Prozesses noch hauptsächlich Iterationen zur Exploration

¹⁴² vgl. Lévárdy & Browning 2009

¹⁴³ vgl. Kreimeyer 2010, S. 64

¹⁴⁴ vgl. Albers, Ebel & Lohmeyer 2012

¹⁴⁵ vgl. Vajna 2005, S. 371

¹⁴⁶ vgl. Darlington & Culley 2002

¹⁴⁷ vgl. Lohmeyer 2013, S. 122

¹⁴⁸ vgl. Wynn 2007, S. 75

¹⁴⁹ vgl. Wynn, Eckert & Clarkson 2007

durchlaufen werden, verlagert sich der Schwerpunkt mit zunehmender Produktreife in Richtung Konvergenz und Verfeinerung.

MEBOLDT und MATTHIESEN¹⁵⁰ heben hervor, dass das Verständnis von Prozess und Iterationen stark zwischen Entwicklung und Management differiert und es daher schwierig ist, im Rahmen der Projektierung geplante Iterationen zu verankern. (vgl. Abbildung 2-14). MATTHIESEN und MEBOLDT liefern indirekt eine Erklärung für das unterschiedliche Verständnis, indem sie zwischen In-Stage- und Cross-Stage-Iteration unterscheiden. Eine **In-Stage-Iteration** betrifft nur Entscheidungen, die im Rahmen eines bzw. des aktuellen Stages¹⁵¹ liegen. Sie führt zu Erkenntnisgewinn ohne die Projektkosten bzw. die Projektdurchlaufzeit negativ zu beeinflussen und ist daher aktiv zu provozieren. Eine **Cross-Stage-Iteration** betrifft hingegen Entscheidungen, die in früheren Stages getroffen wurden. Auch sie führt zu Erkenntnisgewinn, ist jedoch mit erheblichem Zeit- und Kostenaufwänden verbunden, da das Zielsystem in der Regel nicht verfeinert sondern in weiten Teilen geändert wird, wodurch ein erheblicher Änderungsbedarf im Ziel- und insbesondere Objektsystem entsteht. Cross-Stage-Iterationen sind demnach durch gezielte In-Stage-Iterationen zu vermeiden.

| | Prozess | Iteration |
|------------------------------------|--|--|
| Verständnis der Entwicklung | Jeder Entwicklungsprozess ist einzigartig und geprägt von den in ihm auftretenden Problemen, die immer anders sind. | Iterationen sind alltäglich. Sie sind essentielle Lernzyklen, die einen kontinuierlichen Wissensgewinn und so eine Verringerung der Unsicherheit erlauben. |
| Verständnis des Managements | Der Entwicklungsprozess ist ein verbindlicher Standard. Er ist essentiell, um die Kommunikation, die Ressourcen und Investitionen im Unternehmen zu steuern. | Iterationen sind Ausnahmen. Sie kosten Zeit und Geld. Iterationen müssen unterbunden werden. Das Ziel ist ein iterationsfreier Prozess: „Mach's gleich richtig!“ |

Abbildung 2-14: Unterschiedliches Verständnis von Prozess und Iteration¹⁵⁰

Der (zeitliche und monetäre) Aufwand einer Änderung wird dabei wesentlich von den geänderten Zielsystem-Elementen und deren Relationen bestimmt. ECKERT, CLARKSON und ZANKER¹⁵² liefern ein systemisches Rahmenwerk zur Beschreibung von Änderungen in Produktentstehungsprozessen. Im Mittelpunkt der Beschreibung

¹⁵⁰ vgl. Meboldt, Matthiesen & Lohmeyer 2012

¹⁵¹ vgl. Cooper 1990

¹⁵² vgl. Eckert, Clarkson & Zanker 2003

stehen die Änderungssensitivität einzelner Elemente des Zielsystems und die systemweite Fortpflanzung einzelner Änderungen (vgl. Abbildung 2-15).

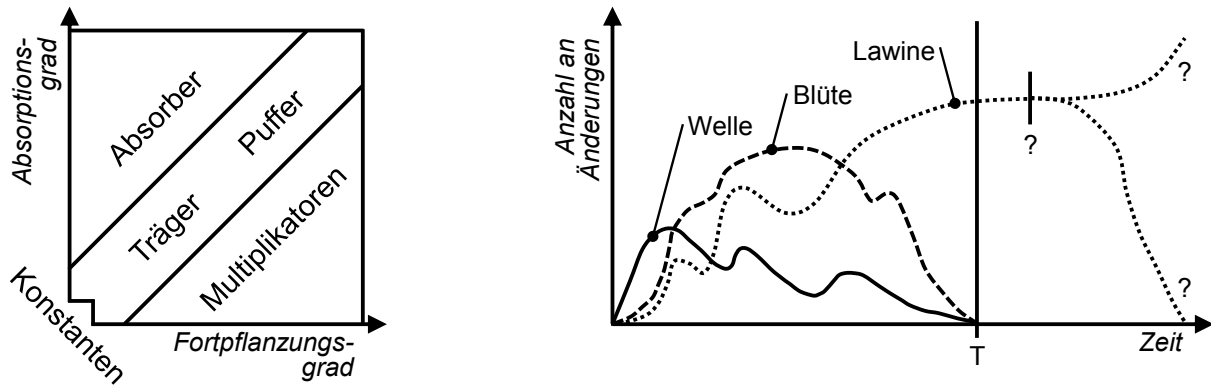


Abbildung 2-15: Änderungssensitivität und Änderungsverlauf¹⁵²

Um die Änderungssensitivität zu beurteilen, schlagen sie vor, die Menge an Änderungen, die ein Zielsystem-Element aufnehmen kann (Absorptionsgrad), über der Menge an Änderungen, die ein Zielsystem-Element weitergibt (Fortpflanzungsgrad), aufzutragen. Hieraus lassen sich vier Änderungssensitivitäts-Typen ableiten. Konstanten sind von Änderungen nicht betroffen – sie nehmen weder Änderungen auf, noch gehen Änderungen von ihnen aus. **Absorber** können mehr Änderungen aufnehmen, als sie selbst weitergeben – sie reduzieren die Komplexität eines Änderungsprozesses und sind daher entsprechend wertvoll. **Träger** absorbieren eine ähnliche Menge an Änderungen wie sie selbst verursachen – beides auf niedrigem bis mittlerem Niveau. **Multiplikatoren** erzeugen mehr Änderungen als sie selbst aufnehmen können – sie erhöhen die Komplexität eines Änderungsprozesses und sollten daher aufmerksam beobachtet und mit Bedacht geändert werden. Um den Gesamtaufwand einer Änderung beurteilen zu können, ist es erforderlich, neben der auslösenden Änderung auch alle möglicherweise betroffenen Systeme zu berücksichtigen. Eine optimale Kombination zu ändernder Systeme auszuwählen ist ein äußerst komplexer Vorgang – prinzipiell kann zwischen drei Verläufen von Änderungsprozessen unterschieden werden (vgl. Abbildung 2-15). **Änderungs-Wellen** beginnen mit einer signifikanten Anzahl an Änderungen, die jedoch mit fortschreitender Zeit stark abnehmen und schließlich in Absorbieren enden. Änderungs-Wellen sind überschaubar und in der Regel vollständig vorhersehbar und planbar. **Änderungs-Blüten** sind umfangreiche Änderungsprozesse, die jedoch noch vorherseh- und damit planbar sind. Trotz des erheblichen Aufwands führen alle Änderungen schließlich zu Absorbieren, der Änderungsprozess gerät nicht außer Kontrolle und kann in der geplanten Zeit abgeschlossen werden. Im Gegensatz dazu sind **Änderungs-Lawinen** zum geplanten Zeitpunkt nicht abgeschlossen. Aufgrund unvorhergesehener Multiplikatoren werden immer mehr Änderungen in Gang gesetzt – im schlimmsten Fall gerät der Änderungsprozess vollständig außer Kontrolle.

2.3.4 Zwischenfazit

Komplexität und Unsicherheit sowie die hieraus resultierenden Iterationen sind immanenter Bestandteil der Produktentstehung. Komplexität setzt sich aus einem strukturellen und einem dynamischen Anteil zusammen, kann sinnvoll nur in Grenzen reduziert und muss für eine erfolgreiche Produktentstehung durch geeignete Modelle beherrscht werden. Hierzu muss die Modellbildung auf Basis möglichst einfacher Meta-Modelle erfolgen, die sich dynamisch im Kontext des jeweiligen Modellierungszwecks anwenden lassen. Ein wesentlicher Komplexitätstreiber in der Produktentstehung ist die vorherrschende Unsicherheit, die sich in Form von Wissens- und Definitionslücken manifestiert. Große Teile dieser Unsicherheit können und müssen durch entsprechende Entwicklungsaktivitäten im Verlaufe der Produktentstehung beseitigt werden, wodurch das anfangs vage Zielsystem kontinuierlich erweitert wird. Unbekannte Unsicherheiten müssen durch einen robusten und iterativen Produktentstehungsprozess mittels kontinuierlicher Validierungsaktivitäten abgefangen werden. Hierdurch wird es möglich, das Zielsystem im Wesentlichen kontinuierlich zu verfeinern und umfangreiche Änderungen sowie damit verbundene zeit- und kostspielige Änderungsprozesse zu vermeiden. Aufgrund unbekannter Unsicherheiten lassen sich Änderungen im Zielsystem jedoch nicht gänzlich vermeiden. Umso wichtiger ist es daher, die Relationen im Zielsystem zu überblicken und auf dieser Basis die Auswirkungen einer (möglichen) Zielsystem-Änderung vorhersehen und planen zu können.

2.4 Produktentstehungsprozesse

Ziel der Produktentstehung ist es, funktionsfähige und produzierbare Produkte zu generieren¹⁵³, die am Markt erfolgreich sind. Um mit der Komplexität und Unsicherheit in den Entstehungsprozessen technischer Produkte umgehen zu können, bedarf es entsprechender Modelle der Prozesse, um die realen Abläufe verstehen, planen, steuern und verbessern zu können¹⁵⁴. Dabei besteht die wesentliche Herausforderung darin, die Vielzahl an Verknüpfungen zwischen und innerhalb der Domänen Produkt und Prozess zu überblicken, durch welche die erforderlichen Informationsflüsse fließen¹⁵⁵. Diese Vielzahl an Verknüpfungen entsteht nicht nur durch das zu entwickelnde Produkt selbst, sondern insbesondere auch durch die beteiligten Akteure und deren vielfältigen und hochgradig vernetzten

¹⁵³ vgl. Ponn & Lindemann 2011

¹⁵⁴ vgl. Negele et al. 1999

¹⁵⁵ vgl. Earl, Johnson & Eckert 2005, S. 187 und Browning, Fricke & Negele 2006

Aktivitäten. Eine geeignete Modellierung der Produktentstehung muss demnach auch Akteure, deren Aktivitäten und Vernetzungen untereinander abbilden können.¹⁵⁶

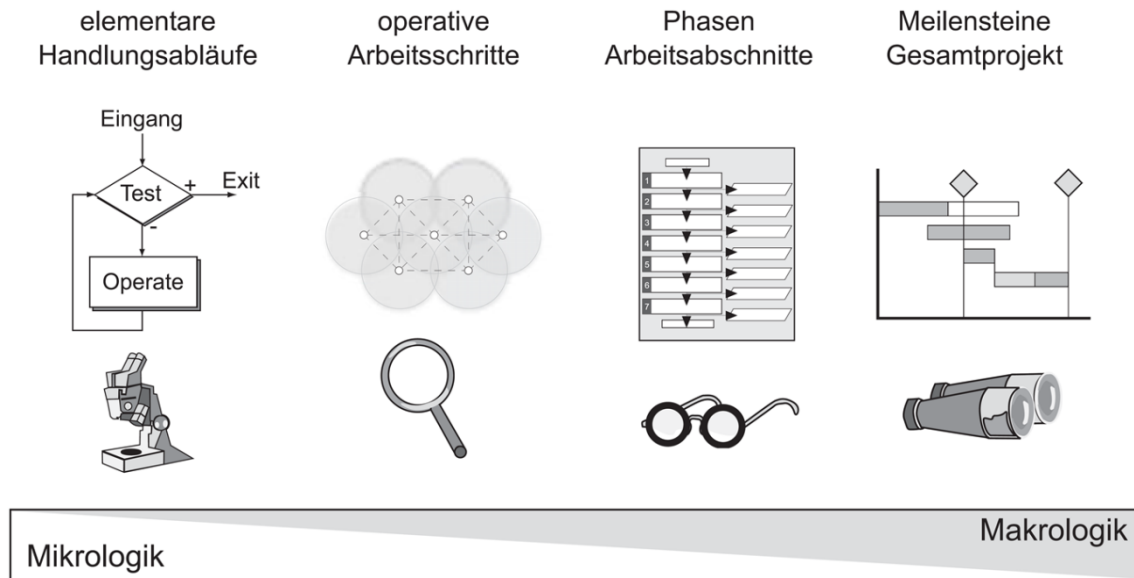


Abbildung 2-16: Auflösungsgrade des Produktentstehungsprozesses¹⁵⁷

LINDEMANN¹⁵⁷ unterscheidet grundsätzlich in vier Auflösungsgrade des Produktentstehungsprozesses und ordnen diese in einem Kontinuum zwischen den Polen Mikrologik und Makrologik an (vgl. Abbildung 2-16). Prozessmodelle der **Mikrologik** beschreiben elementare Denk- und Handlungsabläufe einzelner Individuen. Hierbei handelt es sich um Zyklen der Analyse, Synthese und Bewertung, die sich im Rahmen von Sekundenbruchteilen im Gehirn der Individuen abspielen. Prozessmodelle der **Makrologik** beschreiben hingegen ganze Produktentstehungsprojekte – dies jedoch nur auf einem relativ niedrigen Auflösungsgrad, da andernfalls die Komplexität der Abbildung zu hoch wird und der Modellnutzen verloren geht. Zwischen diesen Polen sind Prozessmodelle angesiedelt, die einzelne operative Arbeitsschritte abbilden (Problemlösungsprozesse) und zu Phasen bzw. Arbeitsabschnitten zusammenfassen. Die Grenzen zwischen diesen vier Auflösungsgraden sind nicht scharf definiert und einzelne Prozessmodelle sind teilweise in der Lage, mehrere Auflösungsgrade abzudecken¹⁵⁸. ALBERS¹⁵⁹ betont, dass zur ganzheitlichen Modellierung von Produktentstehungsprozessen die einzelnen Modelle verschiedener Auflösungsgrade erforderlich sind und auf Basis der Systemtheorie kontinuierlich und konsistent ineinander überführt werden müssen.

¹⁵⁶ vgl. Braun, Ebel & Albers 2013

¹⁵⁷ vgl. Lindemann 2009, S. 38

¹⁵⁸ vgl. Lindemann 2009, S. 39

¹⁵⁹ vgl. Albers & Lohmeyer 2012

2.4.1 Mentale Modelle und Modelle elementarer Problemlösung

Modelle elementarer Problemlösung fokussieren auf einzelne Individuen und haben zum Ziel, die individuellen und impliziten Problemlösungsprozesse zugänglich zu machen. Hierdurch kann ein intersubjektives Verständnis von Problemlösung geschaffen werden, das einen wesentlichen Baustein für **geteilte mentale Modelle** darstellt¹⁶⁰ (vgl. Abbildung 2-17).

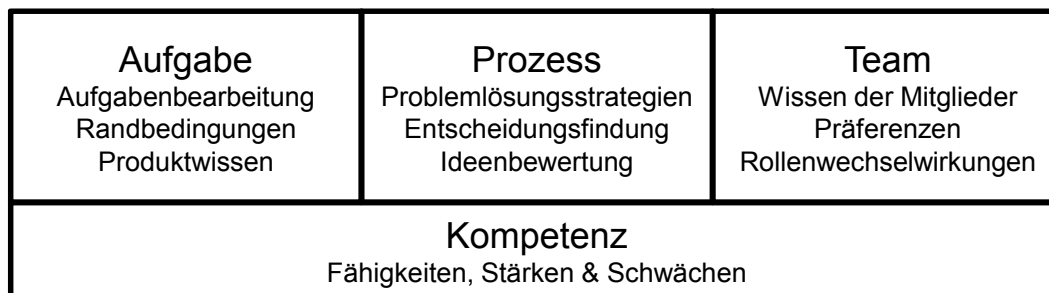


Abbildung 2-17: Typen von mentalen Modellen¹⁶⁰

Mentale Modelle sind individuell konstruierte, interne Arbeitsmodelle der Welt, die es einem Menschen effizient erlauben, neue Informationen aufzunehmen, Vorhersagen zu treffen und Schlüsse zu ziehen¹⁶¹. Gelingt es, die mentalen Modelle der einzelnen Individuen in Einklang zu bringen (geteilte mentale Modelle), ist es den einzelnen Individuen möglich, Erklärungen und Erwartungen richtig zu formulieren und ihre Handlungen, einschließlich der Anpassung ihres Verhaltens, entsprechend ihrer Aufgabe und der anderen Individuen zu koordinieren¹⁶². Hierbei sind ein gemeinsames Begriffs- und Verständnismodell von entscheidender Bedeutung, um gemeinsame mentale Modelle auszubilden¹⁶³ und damit die synergetische, funktionale Bündelung der mentalen Leistungsfähigkeit eines Teams zu realisieren¹⁶⁴. BIRKHOFFER UND JAENSCH¹⁶⁵ heben hervor, dass erst durch geteilte mentale Modelle eine zielführende Interaktion zwischen Individuen auf Basis externer Repräsentationen möglich wird. Geteilte mentale Modelle sind daher eine wesentliche Voraussetzung für kollektive Problemlösung in Produktentstehungsprozessen.

Das TOTE-Modell¹⁶⁶ und der PDCA-Zyklus¹⁶⁷ sind einfache Modelle zur Beschreibung von Problemlösungsprozessen auf elementarer Ebene (vgl. Abbildung 2-18).

¹⁶⁰ vgl. Neumann, Badke-Schaub & Lauche 2006

¹⁶¹ vgl. Badke-Schaub, Neumann, Lauche & Mohammed 2007

¹⁶² vgl. Lohmeyer 2013, S. 81

¹⁶³ vgl. Albers, Turki & Lohmeyer 2012

¹⁶⁴ vgl. Langan-Fox, Anglim & Wilson 2004

¹⁶⁵ vgl. Birkhofer & Jänsch 2003

¹⁶⁶ vgl. Miller, Galanter & Pribram 1960, S. 26

¹⁶⁷ vgl. Deming 1982, S. 88

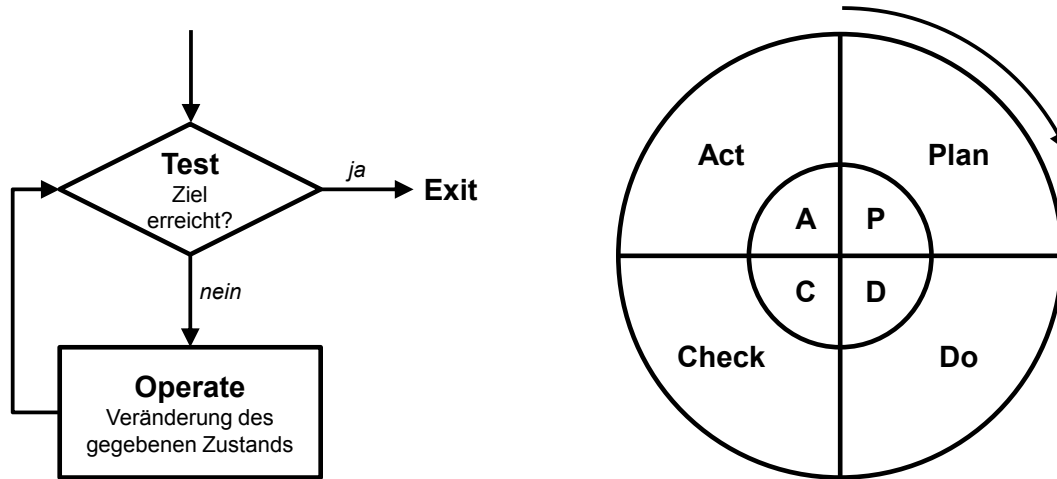


Abbildung 2-18: TOTE-Modell und PDCA-Zyklus¹⁶⁸

Das **TOTE-Modell** entstammt der Kybernetik und beschreibt einen Problemlösungsprozess als Regelkreis aus Testen (Test) und Handeln (Operate), der solange durchlaufen wird, bis ein definierter Sollzustand oder ein alternatives Abbruchkriterium, beispielsweise Zeit, erreicht ist (Exit). Der Regelkreis entspricht einem Versuch- und Irrtum-Verhalten (Trial and Error) und stellt damit einen Problemlösungsprozess auf elementarster Ebene dar, weswegen das TOTE-Modell in vielen operativen Problemlösungsmethoden berücksichtigt wird¹⁶⁹. Der **PDCA-Zyklus** hat seine Ursprünge in der Qualitätssicherung¹⁷⁰ und besteht aus den vier Schritten Planen (plan), Umsetzen (do), Überprüfen (check) und Handeln (act)¹⁷¹. Planen umfasst die Analyse des aktuellen Zustands, das Erkennen von Verbesserungspotenzialen und das Entwickeln neuer Ansätze. Umsetzen bedeutet das pragmatische, ressourceneffiziente Testen und Optimieren des entwickelten Ansatzes in begrenztem Umfang. Auf Basis dieser ersten Implementierung und der resultierenden Testergebnisse wird der entwickelte Ansatz sorgfältig überprüft (Überprüfen) und gegebenenfalls zur Einführung in großem Umfang (Handeln) freigegeben.

2.4.2 Modelle operativer Problemlösung

Modelle operativer Problemlösung beschreiben ein systematisches Vorgehen zur Lösung von Problemen auf Basis einzelner Arbeitsschritte. Sie beschreiben das Vorgehen nicht auf elementarer Ebene wie beispielsweise das TOTE-Modell,

¹⁶⁸ vgl. Ehrlenspiel 2009, S. 83 und Unger 2003, S.59

¹⁶⁹ vgl. Lindemann 2009, S. 40

¹⁷⁰ vgl. Lohmeyer 2012, S. 110

¹⁷¹ vgl. Unger 2003, S.58

sondern bilden komplexe Handlungsabläufe ab. Prinzipiell lassen sich alle Modelle operativer Problemlösung auf drei Hauptschritte abstrahieren¹⁷²

- Ziel beziehungsweise Problem klären
- Lösungsalternativen generieren
- Entscheidung herbeiführen

Problemlösungszyklus der Systemtechnik

Der Problemlösungszyklus der Systemtechnik nach DAENZER UND HUBER¹⁷³ besteht aus sechs Schritten, die sich in die Abschnitte Zielsuche, Lösungssuche und Lösungsauswahl gliedern lassen (vgl. Abbildung 2-19).

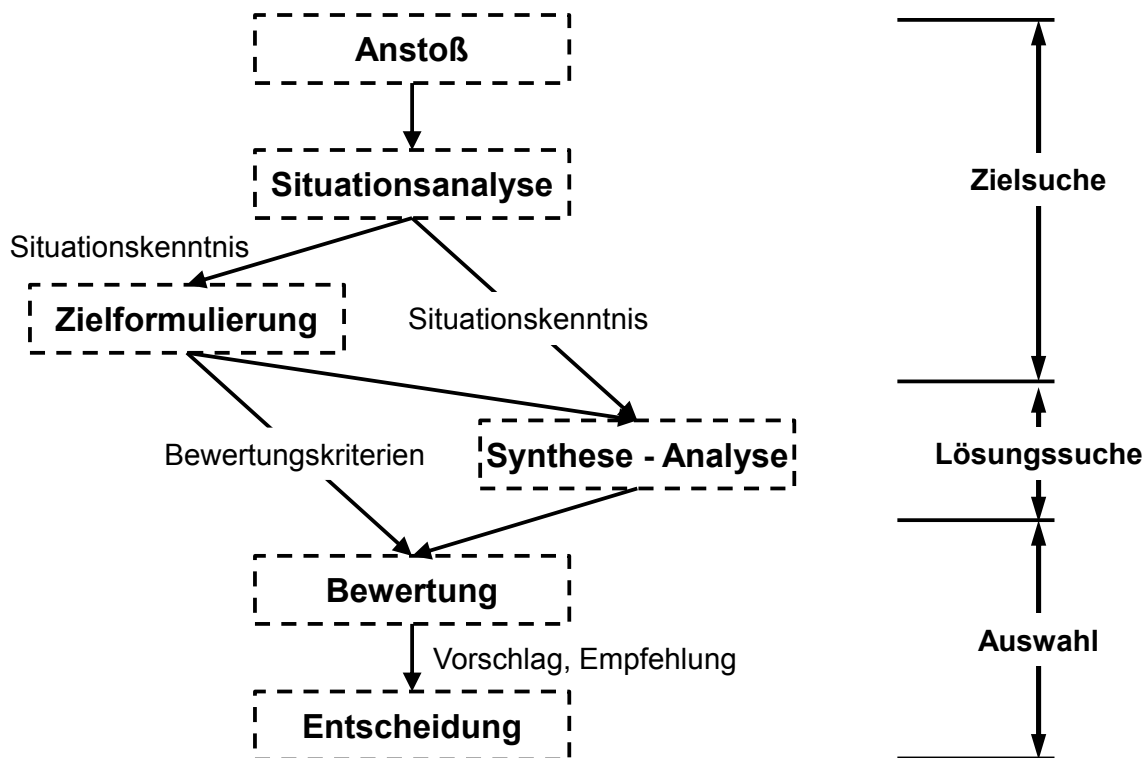


Abbildung 2-19: Problemlösungszyklus der Systemtechnik¹⁷³

In der Situationsanalyse wird durch das Erkennen von Stärken und Schwächen des Systems sowie der Gefahren und Chancen für das System in der Zukunft die Situation begreifbar. Das Untersuchungsfeld und der Gestaltungsbereich werden strukturiert bzw. abgegrenzt und eine Informationsbasis für die folgenden Schritte geschaffen. In der Zielformulierung werden verschiedene Interessen formuliert und ausgeglichen. Wichtige Impulse auf die Fragen was erreicht, bzw. was vermieden werden soll, ergeben sich bereits aus der Situationsanalyse und müssen in Form von

¹⁷² vgl. Lindemann 2009, S. 41

¹⁷³ vgl. Daenzer & Huber 1996, S. 96

Zielen expliziert werden. Diese Ziele sind die Basis sowohl für die Lösungssuche, als auch für die Bewertung. In wechselseitigen Schritten erfolgt in der Lösungssuche die konstruktive und kreative Synthese, um Lösungen zu finden und zu gestalten sowie die kritische Analyse, um Lösungen zu prüfen. Hierdurch werden mehrere Lösungen erarbeitet, die prinzipiell die formulierten Ziele erfüllen. In der Bewertung werden die unterscheidbaren Handlungs- bzw. Lösungsalternativen anhand der Bewertungskriterien aus der Zielformulierung bewertet. Basis ist die Informationsbeschaffung über Lösungseigenschaften bzw. Wirkungen möglicher Handlungsalternativen. Der Bewertung folgt eine formale Entscheidung.¹⁷⁴

Münchener Vorgehensmodell (MVM)

Das Münchener Vorgehensmodell zur Problemlösung¹⁷⁵ umfasst sieben Schritte, die sowohl sequentiell als auch iterativ durchlaufen werden und bei Bedarf in kleine Einzelschritte aufgeteilt werden können. Um den linearen Verlauf vieler Vorgehensmodelle zur Problemlösung aufzubrechen, wird das Münchener Vorgehensmodell in Form eines Netzwerks dargestellt. Hierdurch sollen reale Prozesse mit ihrem sprunghaften Verlauf besser abgebildet werden können, als lineare Darstellungen mit lediglich einem Verweis auf erlaubte bzw. notwendige Rücksprünge.

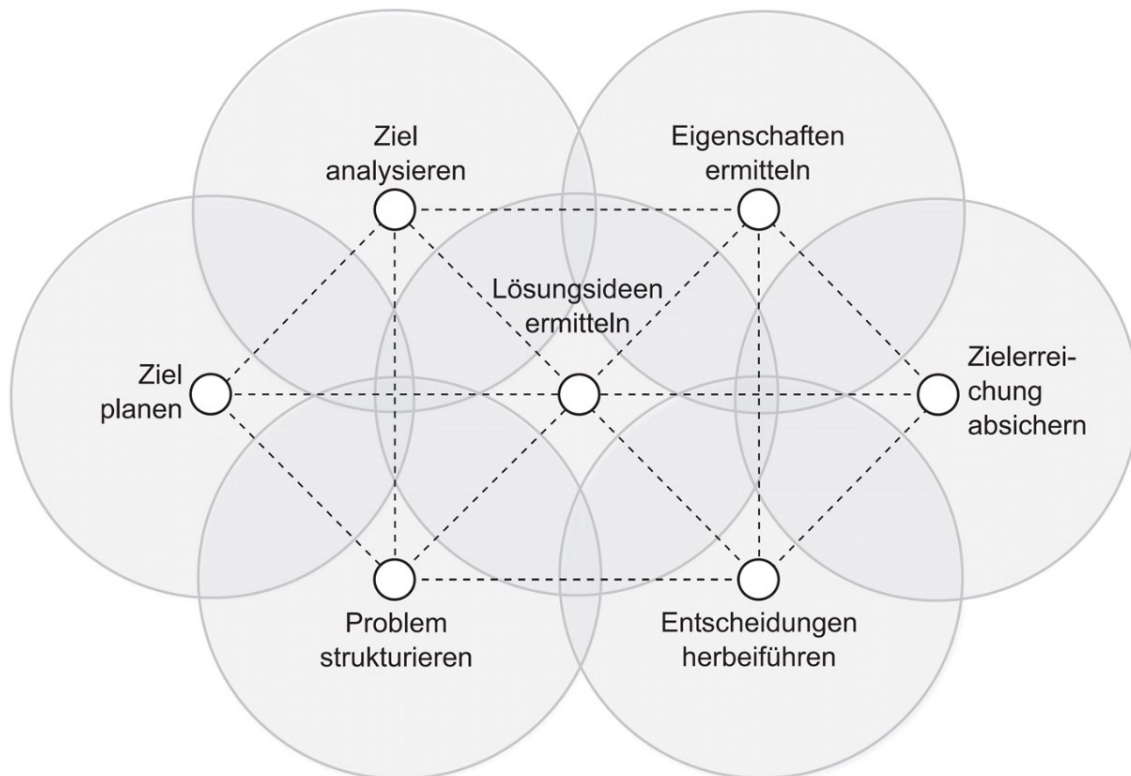


Abbildung 2-20: Das Münchener Vorgehensmodell zur Problemlösung¹⁷⁵

¹⁷⁴ vgl. Gerst 2002, S. 11

¹⁷⁵ vgl. Lindemann 2009, S. 47

Das Element Ziel planen enthält eine Analyse der aktuellen und ggf. der zukünftigen Situation sowie die Ableitung konkreter Maßnahmen. Das Element Ziel analysieren umfasst die Klärung und Beschreibung des gewünschten Zielzustandes. Dies umfasst in der Regel die Formulierung konkreter und detaillierter Anforderungen sowie das Aufdecken des Beziehungsgeflechts zwischen diesen Anforderungen. Dabei werden Anforderungen nicht nur an das Produkt als angestrebtes Endergebnis gestellt sondern insbesondere auch an Zwischenziele im Prozess. Im Schritt Problem strukturieren werden Handlungsschwerpunkte ermittelt, die eine Fokussierung bei der anschließenden Lösungssuche ermöglichen. Hierzu erfolgt eine Neu- oder Umformulierung des Problems bzw. eine Umstrukturierung oder Abstraktion des betrachteten Systems, wodurch insbesondere Fixierungen und Denkbarrieren aufgehoben werden können. Darüber hinaus werden zur Verfügung stehende Handlungs- und Gestaltungsfreiräume herausgearbeitet. Das Element Lösungsideen ermitteln beschreibt die Suche nach vorhandenen Lösungen und das Generieren neuer Lösungen. Hierbei wird betont, dass stets mehrere Alternativen zu erarbeiten sind und ein Schwerpunkt in der Ordnung und Kombination von Teillösungen zu optimalen Gesamtlösungen liegt. Unter Eigenschaften ermitteln ist die Ermittlung und Konkretisierung der Ausprägungen relevanter Merkmale der erarbeiteten Lösungsideen oder Lösungsalternativen zu verstehen. Die Eigenschaftsanalyse ist ein wichtiger Schritt, um Aussagen über eine Lösung bzw. ein System hinsichtlich der Zielerreichung treffen zu können. Das Element Entscheidung herbeiführen repräsentiert die Bewertung von Lösungsideen und -alternativen sowie das eigentliche Treffen einer Auswahl. Das Element Zielerreichung absichern trägt zur Verminderung von Risiken bei der Umsetzung der Entscheidung bei. Hierzu müssen mögliche Risiken erkannt und bewertet werden und gegebenenfalls Maßnahmen zur Risikominimierung abgeleitet werden.

SPALTEN

Die SPALTEN-Problemlösungsmethode nach ALBERS¹⁷⁶ ist eine universelle Vorgehensweise zur Behandlung von Problemen unterschiedlicher Randbedingungen und Komplexitätsgrade¹⁷⁷. SPALTEN besteht aus sieben Schritten, die im Problemlösungsprozess sequenziell oder fallspezifisch angewandt werden können (vgl. Abbildung 2-21). Jeder einzelne SPALTEN-Schritt kann durch einen separaten SPALTEN-Prozess bearbeitet werden, d.h. die SPALTEN-Problemlösungsmethode

¹⁷⁶ vgl. Albers, Saak & Burkardt 2002

¹⁷⁷ vgl. Saak 2007, S. 35

ist fraktal¹⁷⁸. Vor bzw. zwischen jedem SPALTEN-Schritt fordert die Methode eine Überprüfung der vorliegenden Informationen und des Problemlösungsteams.

Die sieben Schritte sind¹⁷⁹:

- **Situationsanalyse** – Vorbereitende Informationserfassung über Soll- und Ist-Zustand sowie verknüpfte Randbedingungen.
- **Problemeingrenzung** – Untersuchung der gesammelten Information zur Eingrenzung des Kerns der weiteren Betrachtung; Definition von Entscheidungskriterien.
- **Alternative Lösungssuche** – Kreative, diskursive und/oder recherchierende Lösungsfindung für die Überwindung des Delta zwischen Soll- und Ist-Zustand.
- **Lösungsauswahl** – Auswahl einer umzusetzenden Lösung nach zuvor definierten Kriterien.
- **Tragweitenanalyse** – Systematische Untersuchung von Chancen und Risiken, die mit der getroffenen Auswahl verbunden sind und gegebenenfalls Definition von Maßnahmenplänen.
- **Entscheiden und Umsetzen** – Verantwortlicher Beschluss zur Lösungsumsetzung und Implementierung der Lösung (gegebenenfalls durch weitere zu bestimmende Aktivitäten).
- **Nachbereiten und Lernen** – Reflektion des Problemlösungsprozesses und gegebenenfalls Festhalten von Erkenntnissen für zukünftige Prozesse.

Kennzeichnend für die SPALTEN-Methode ist die zyklische Ausweitung und Verdichtung der Informationsmenge im Problemlösungsprozess (atmender Prozess)¹⁸⁰.

2.4.3 Vorgehens- und Phasenmodelle der Produktentstehung

Vorgehens- und Phasenmodelle der Produktentstehung strukturieren den Produktentstehungsprozess in inhaltlich und zeitlich abtrennbare Arbeitsabschnitte bzw. Phasen. Im Gegensatz zu den Modellen der elementaren und operativen Problemlösung, die die Mikrologik des Produktentstehungsprozesses darstellen, bilden Vorgehens- und Phasenmodelle die Makrologik aus. Hierbei finden innerhalb der einzelnen Arbeitsabschnitte bzw. Phasen die Modelle der elementaren und operativen Problemlösung Anwendung¹⁸¹.

¹⁷⁸ vgl. Albers, Braun & Muschik 2010

¹⁷⁹ vgl. Braun 2013, S. 90

¹⁸⁰ vgl. Albers & Braun 2011a, S. 21

¹⁸¹ vgl. Lindemann 2009, S. 43

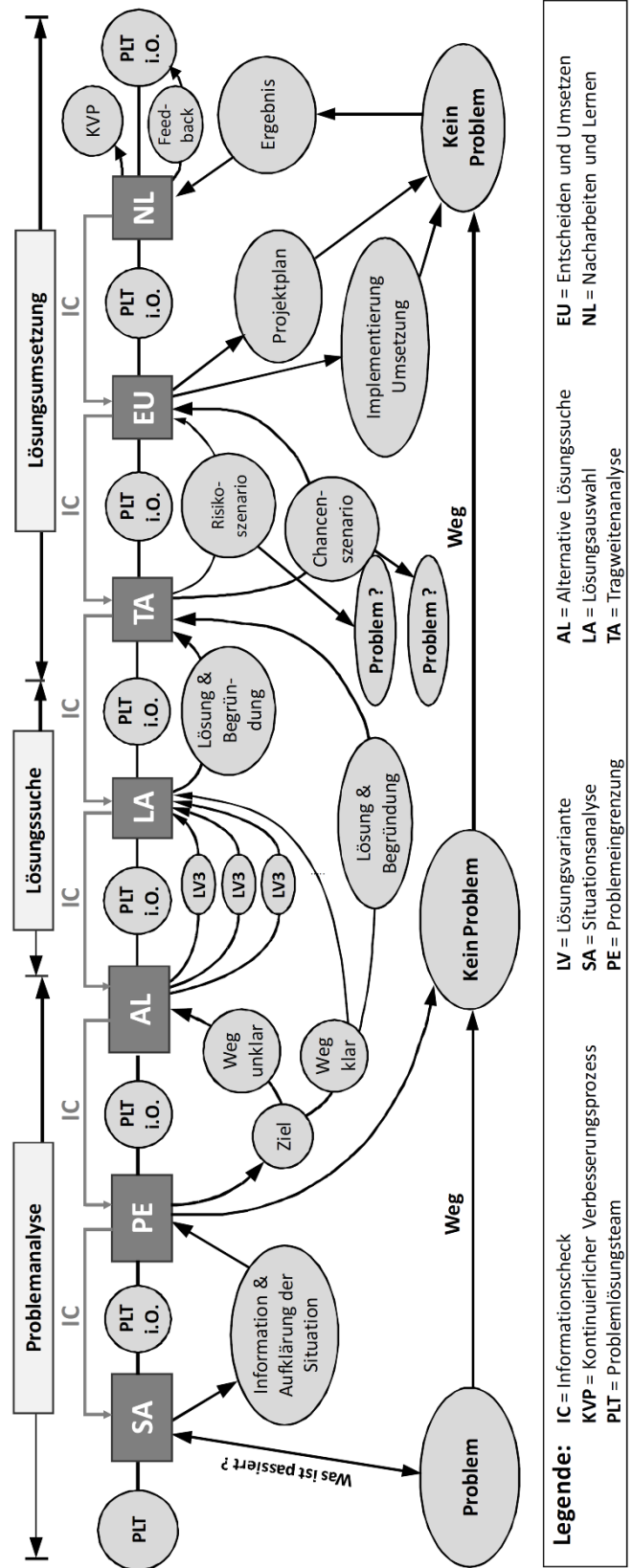


Abbildung 2-21: SPALTEN-Problemlösungsmethode¹⁸²

¹⁸² vgl. Albers et al. 2005

VDI 2221

Die VDI-Richtlinie 2221¹⁸³ behandelt allgemeingültige, branchenunabhängige Grundlagen methodischen Entwickelns und Konstruierens und definiert Arbeitsabschnitte und Arbeitsergebnisse, die als Leitlinie für ein Vorgehen in der Praxis dienen können (vgl. Abbildung 2-22).

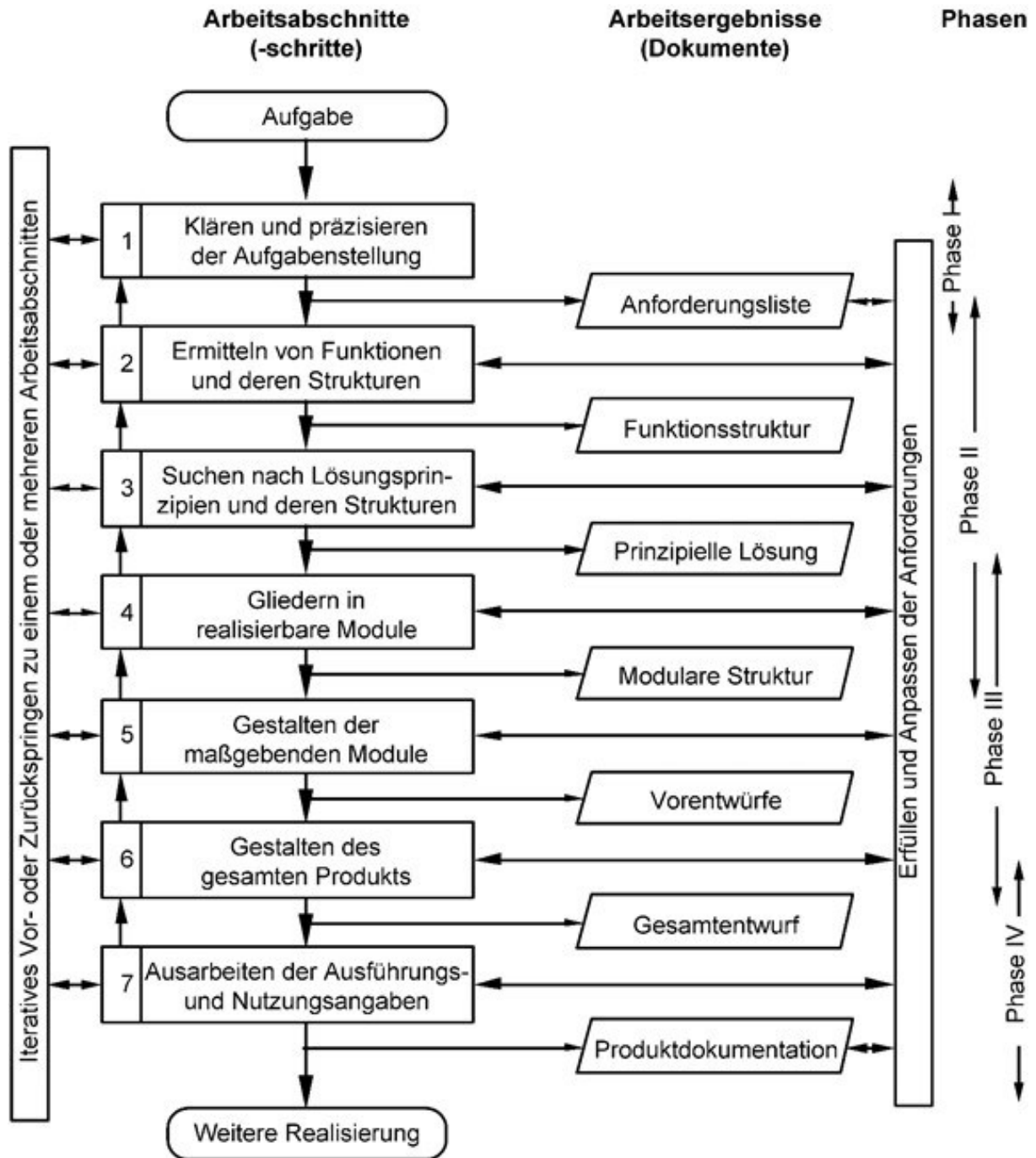


Abbildung 2-22: Generelles Vorgehen beim Entwickeln und Konstruieren¹⁸³

¹⁸³ vgl. VDI 2221 1993

Das Vorgehensmodell besteht aus insgesamt sieben Arbeitsabschnitten mit jeweils einem Arbeitsergebnis die durch einen sequenziellen Problemlösungsprozess untergliedert werden. Um dem iterativen Verlauf von Produktentstehungsprozessen gerecht zu werden, sieht das Modell das Vor- und Zurückspringen zu einem oder mehreren Arbeitsschritten vor. Um der kontinuierliche Erweiterung der Wissensbasis im Produktentstehungsprozess¹⁸⁴ Rechnung zu tragen, sieht das Modell einen ständigen Abgleich bzw. Anpassung der definierten Anforderungen vor. Obwohl die Richtlinie damit den Anspruch erhebt, einen iterativen Produktentstehungsprozess wie er in der Praxis vorherrscht, abbilden zu können, wirkt die Richtlinie in ihrer Struktur und Anwendung sehr starr¹⁸⁵. ALBERS kritisiert im Besonderen den streng sequenziellen Problemlösungsprozess, der über die Lebensphasen gelegt ist¹⁸⁶, wohingegen BIRKHOFER insbesondere den sequenziellen Ablauf der Arbeitsabschnitte kritisiert¹⁸⁷, der verhindert, dass Iterationen, wie sie in der Realität auftreten, abgebildet werden können. In der VDI-Richtlinie 2221 findet demnach keine ausreichende Trennung zwischen den Inhalten der einzelnen Arbeitsabschnitte und ihrem zeitlichen Auftreten im Produktentstehungsprozess statt, wodurch weder mit Iterationen noch mit der kontinuierlichen Erweiterung der Wissensbasis sinnvoll umgegangen werden kann.

VDI 2206

Die VDI-Richtlinie 2206¹⁸⁸ ist ein praxisorientierter Leitfaden zur systematischen und domänenübergreifenden Unterstützung der Entwicklung mechatronischer Systeme. Sie ersetzt dabei nicht die etablierten disziplinspezifischen Leitfäden und Vorgehensmodelle, sondern führt diese zusammen¹⁸⁹. Hierbei betont die VDI-Richtlinie 2206 insbesondere, dass es keine „kanonisierbare Optimalform des Konstruktionsprozesses gibt, der der Konstrukteur in einem festen Ablaufplan folgen kann“ und leitet hieraus die Zielsetzung eines flexiblen Vorgehensmodells ab, das sich im Wesentlichen auf drei Elemente stützt¹⁹⁰:

- Allgemeiner Problemlösungszyklus auf der Mikroebene
- V-Modell auf der Makroebene
- Vordefinierte Prozessbausteine zur Bearbeitung wiederkehrender Arbeitsschritte bei der Entwicklung mechatronischer Systeme

¹⁸⁴ vgl. Albers, Lohmeyer & Ebel 2011

¹⁸⁵ vgl. Meboldt 2009, S. 37

¹⁸⁶ vgl. Albers 2006, S. 10

¹⁸⁷ vgl. Weiß 2006, S. 11

¹⁸⁸ vgl. VDI 2206

¹⁸⁹ vgl. VDI 2206, S. 3

¹⁹⁰ vgl. VDI 2206, S. 26

Der Problemlösungszyklus der Mikroebene basiert auf dem Problemlösungszyklus der Systemtechnik¹⁹¹ und kann daher flexibel an bestehende Randbedingungen angepasst werden¹⁹². Das V-Modell auf der Makroebene stellt die logische Abfolge der wesentlichen Teilschritte zur Entwicklung mechatronischer Produkte dar¹⁹³ und ist problem- bzw. aufgabenspezifisch anzupassen, so dass die zeitliche Abfolge der Teilschritte in der tatsächlichen Anwendung abweichen kann. Das V-Modell beschreibt damit ein generisches und systematisches Vorgehen zum Entwurf mechatronischer Systeme und untergliedert den Entwicklungsprozess in drei übergeordnete Phasen (vgl. Abbildung 2-23):

- Domänenübergreifender Systementwurf
- Domänenspezifischer Entwurf
- Systemintegration

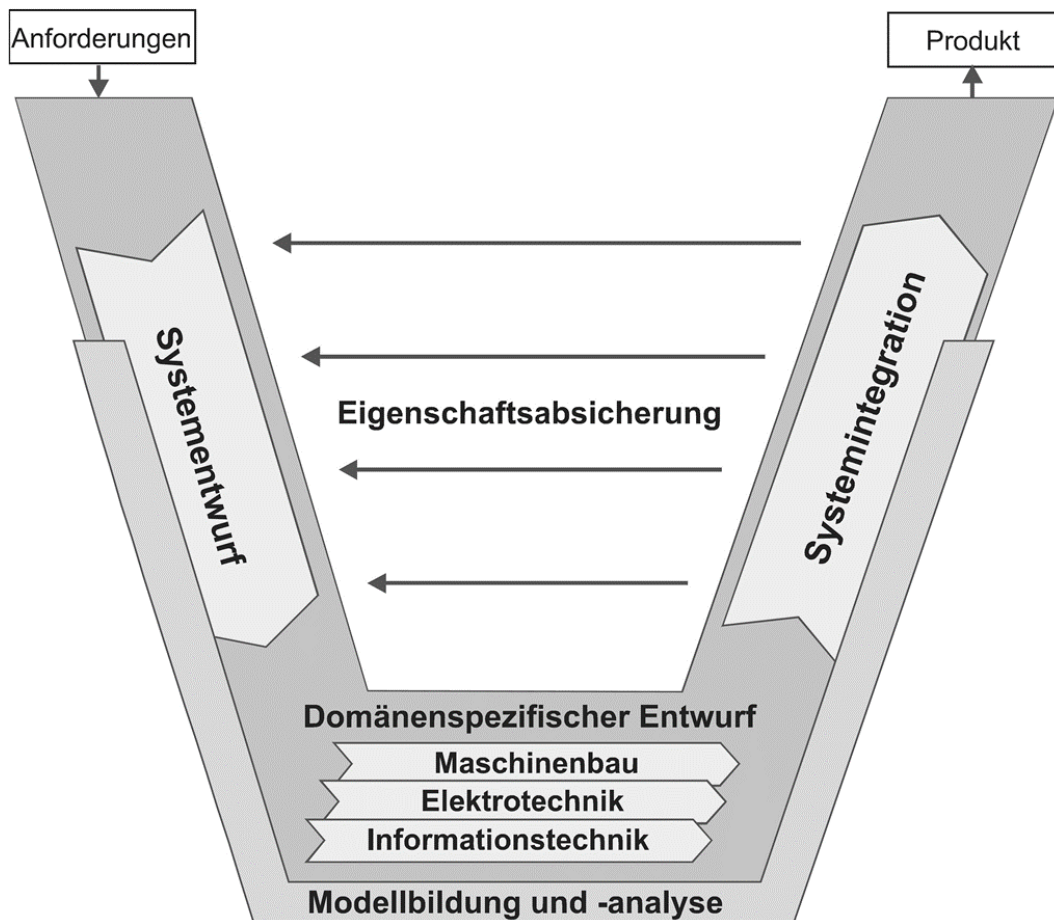


Abbildung 2-23: V-Modell der Entwicklung mechatronischer Systeme¹⁹⁴

¹⁹¹ vgl. Daenzer & Huber 1996, S. 96 und Haberfellner et al. 2012

¹⁹² vgl. Hellenbrand 2013, S. 39

¹⁹³ vgl. Bröhl 1995 und Flath et al. 2000

¹⁹⁴ vgl. VDI 2221 2004, S. 40

Im **domänenübergreifenden Systementwurf** werden auf Basis der formulierten Anforderungen die wesentlichen physikalischen und logischen Wirkungsweisen des Produktes festgelegt und spezifiziert. Hierzu wird die Gesamtfunktion des Produkts in wesentliche Teilfunktionen zerlegt (partitioniert¹⁹⁵), für diese Teilfunktionen geeignete Wirkprinzipien identifiziert und die Funktionserfüllung im Systemzusammenhang überprüft. Hierbei werden bereits zu diesem frühen Zeitpunkt **Testfälle** – also die Randbedingungen für eine spätere Validierung – definiert¹⁹⁶. In der Phase des **domänenspezifischen Entwurfs** erfolgt eine weitere Konkretisierung der Teilsysteme, die zumeist getrennt in den einzelnen Disziplinen abläuft. In der **Systemintegration** werden die Teilsysteme der verschiedenen Disziplinen zu Gesamtsystemen zusammengeführt und deren korrektes Zusammenwirken untersucht.¹⁹⁷ Die drei übergeordneten Phasen werden von Modellbildungs- und Modellanalysetätigkeiten begleitet, die zur Ermittlung und Absicherung der Systemeigenschaften während des Entwicklungsprozesses genutzt werden. Im Rahmen der Eigenschaftsabsicherung wird überprüft, ob das (Teil-)System die spezifizierten Anforderungen erfüllt. Dabei entsteht ein komplexes mechatronisches Erzeugnis in der Regel nicht innerhalb eines Makrozyklus. Vielmehr sind mehrere Durchläufe erforderlich. So entsteht beispielsweise in einem ersten Schritt ein **Labormuster**, dann ein **Funktionsmuster** und erst in einem dritten Makrozyklus das finale Produkt¹⁹⁸. ZINGEL sieht in dieser Form bzw. diesem Verständnis der Eigenschaftsabsicherung eines der größten Probleme in der praktischen Anwendung der VDI-Richtlinie 2206, da eine kontinuierliche Eigenschaftsabsicherung im stark sequenziell verlaufenden domänenspezifischen Entwurf nur schwer umzusetzen ist. „Beispielsweise ist nicht vorgesehen, die als Ausgangspunkt der Entwicklung verwendeten Anforderungen in irgendeiner Form zu ergänzen oder anzupassen. Ebenso wird keine Rückführung der Ergebnisse des domänenspezifischen Entwurfs in den Systementwurf durchgeführt.“¹⁹⁹ Auch HELLENBRAND sieht in der Phase des domänenspezifischen Entwurfs die größten Defizite der VDI-Richtlinie 2206, da die Wichtigkeit der Synchronisation der Disziplinen zwar genannt wird, jedoch keine durchgehende methodische Unterstützung vorliegt²⁰⁰.

¹⁹⁵ vgl. Jansen 2007

¹⁹⁶ vgl. Braun 2013, S. 49

¹⁹⁷ vgl. Hellenbrand 2013, S. 41

¹⁹⁸ vgl. VDI 2206 2004, S. 31

¹⁹⁹ vgl. Zingel 2013, S. 85

²⁰⁰ vgl. Hellenbrand 2013, S. 42

Stage-Gate-Prozess

Der Stage-Gate-Prozess (auch Wasserfall-Prozess genannt) ist das am weitesten verbreitete Modell zur Planung und Steuerung ganzer Produktentstehungsprozesse²⁰¹. Der ideale Stage-Gate-Prozess besteht aus einzelnen Phasen (Phases), die durch Meilensteine (Stage-Gates oder Reviews) voneinander getrennt sind (vgl. Abbildung 2-24).

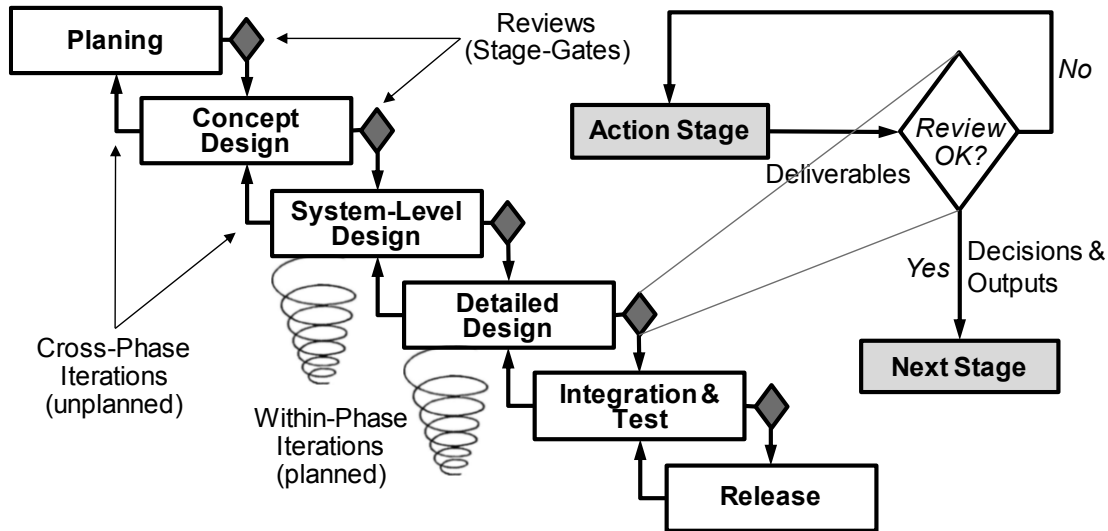


Abbildung 2-24: Der ideale Stage-Gate-Prozess²⁰²

COOPER unterscheidet drei Generationen von Stage-Gate-Prozessen²⁰³. Die erste Generation von Stage-Gate-Prozessen ist streng in Phasen gegliedert und legt einen starken Fokus auf die Planung und Steuerung der technischen Aspekte der Entwicklungstätigkeit²⁰⁴. Einzelne **Phasen** werden strikt durch **Phasenreviews** voneinander abgetrennt, d.h. erst wenn alle geforderten **Arbeitsergebnisse** (Deliverables) vorliegen, kann in die nächste Phase übergegangen werden (Entscheidungen). Diese Form des Stage-Gate-Prozesses eignet sich sehr gut, wenn zeitlich stabile Anforderungen vorliegen und die im Produkt eingesetzten Technologien verstanden sind – ist dies nicht der Fall, liefern Stage-Gate-Prozesse der ersten Generation nicht befriedigende Ergebnisse, da Iterationen mit erheblichen Zeit- und Kostenaufwänden verbunden sind und eine Anpassung von Anforderungen nicht vorgesehen ist.²⁰⁵ Die zweite Generation von Stage-Gate-Prozessen baut weiterhin auf Phasen auf, denen jeweils Reviews folgen, jedoch wird die Flexibilität des Prozesses durch überlappende Phasen bzw. überlappende Abschnitte innerhalb

²⁰¹ vgl. Cooper 2001, McConnel 1996, Smith & Reinertsen 1992, Ulrich & Eppinger 2000

²⁰² vgl. Unger 2003, S. 46 & S. 62

²⁰³ vgl. Cooper 1994

²⁰⁴ vgl. Braun 2013, S. 54

²⁰⁵ vgl. Unger 2003, S. 47

einer Phase verbessert. Hierdurch können Cross-Phase-Iterationen besser abgefangen werden, da eine Interaktion zwischen unterschiedlichen Akteuren und Aktivitäten innerhalb einer Phase möglich wird und Konflikte bzw. Inkonsistenzen nicht erst während bzw. nach einem Meilenstein aufgedeckt werden. Die dritte Generation von Stage-Gate-Prozessen beinhaltet flexible und überlappende Fuzzy Gates anstatt der vergleichsweise starren und fixen Meilensteine. Erst hierdurch lassen sich Prozesse mit überlappenden Phasen bzw. überlappenden Abschnitten innerhalb der Phasen effektiv steuern. Hierdurch steigt jedoch der Anspruch an das Management solcher Prozesse deutlich an, da sich beispielsweise die Nachverfolgung und Überprüfung von Meilenstein-Arbeitsergebnissen (Deliverables) erheblich erschwert²⁰⁶. Jedoch werden auch in der dritten Generation von Stage-Gate-Prozessen weiterhin einzelne Phasen sequentiell durchlaufen, wodurch eine phasenübergreifende, kontinuierliche Validierung kaum zu realisieren ist und das Risiko ungeplanter Cross-Phase-Iterationen nur schwer auf ein akzeptables Maß gesenkt werden kann.

Spiral-Prozess

Der Spiral-Prozess²⁰⁷ entstammt der Software-Entwicklung und legt im Gegensatz zum Stage-Gate-Prozess einen besonderen Fokus auf geplante Iterationen, die bewusst mehrere klassische Phasen des Stage-Gate-Prozesses umfassen (vgl. Abbildung 2-25)²⁰⁸.

Während in Stage-Gate-Prozessen die Wechselwirkungen zwischen den einzelnen Phasen dadurch berücksichtigt werden sollen, dass die Phasen mehr oder weniger stark überlappen, verfolgt der Spiral-Prozess den Ansatz, dass alle Phasen mehrmals durchlaufen werden und hierdurch die **Produktreife** kontinuierlich zunimmt. Hierbei wird jeder Umlauf mit einem Meilenstein beendet, der nicht mehr nur die vorangegangene Phase bewertet, sondern den Projektfortschritt insgesamt. Dieses Vorgehen ist in besonderem Maße für die Entwicklung von Systemen geeignet, deren Anforderungen zu Beginn der Entwicklung nicht vollständig bekannt sind²⁰⁹. Um das Potenzial des Spiral-Prozesses zur Risiko-Minimierung in Produktentstehungsprozessen zu heben, müssen die Meilensteine insbesondere den Produktreifegrad überprüfen, da andernfalls die Meilensteine nur als einfache Phasen-Gates eines Stage-Gate-Prozesses wirken²¹⁰.

²⁰⁶ vgl. McConnel 1996

²⁰⁷ vgl. Boehm 1988

²⁰⁸ vgl. Unger 2003, S. 50

²⁰⁹ vgl. Balzert 2000, S. 133 und Kallmeyer 1998, S. 51

²¹⁰ vgl. Unger 2003, S. 52

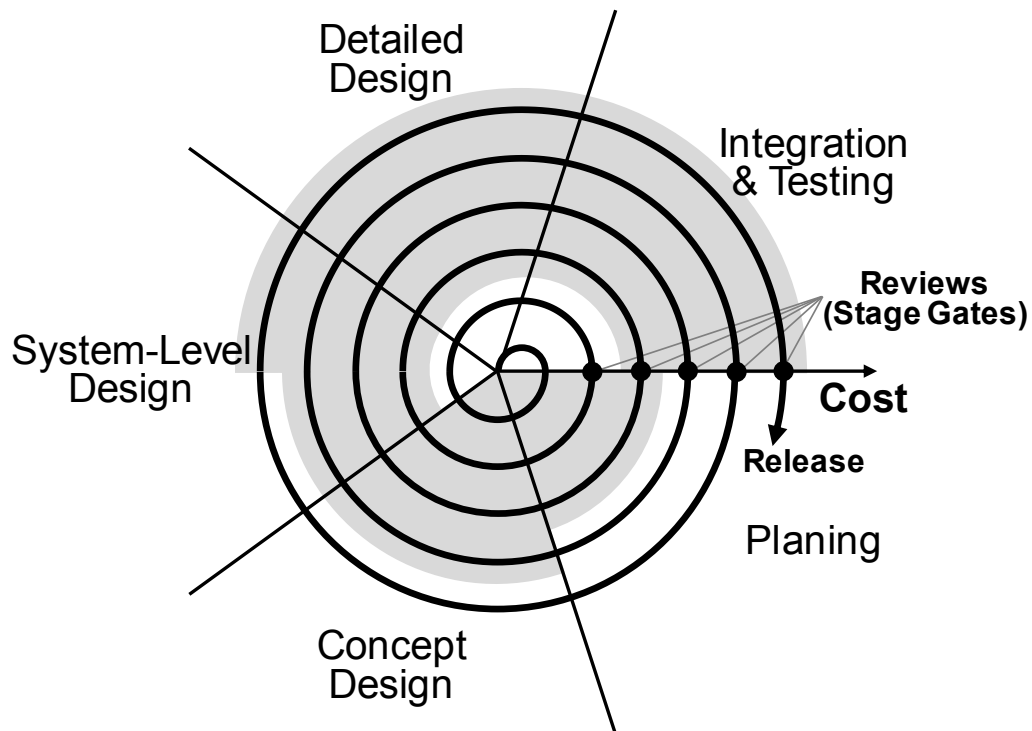


Abbildung 2-25: Der Spiral-Prozess

Der **Produktreifegrad** kann als „Zustand eines Produkts hinsichtlich definierter Indikatoren zu einem beliebigen Zeitpunkt²¹¹ oder als „Grad der Erfüllung der Forderungen an ein Produkt“²¹² definiert werden. Seine Bestimmung ist nicht trivial und muss von individuellen Projektanforderungen abhängig gemacht werden²¹³. BRAUN²¹³ listet eine Reihe von Ansätzen zur Ermittlung des Produktreifegrades auf. So lässt sich der Produktreifegrad auf Basis erfüllter Projektanforderungen (Produktfortschritt²¹⁴) bzw. Produkteigenschaften²¹⁵ oder auf Basis abgeschlossener Arbeitspakete (Projektfortschritt²¹⁶) ermitteln. Eine pragmatische Möglichkeit zur Ermittlung des Produktreifegrades sieht PFEIFER-SILBERBACH²¹⁷ in der Erfassung erfüllter Meilensteine/Deliverables oder freigegebener Dokumente. MÜLLER weist darauf hin, dass Reifegrade nur in Kombination mit Absicherungsaktivitäten sinnvoll betrachtet werden können.²¹⁸

²¹¹ vgl. Pfeifer-Silberbach 2005, S. 2

²¹² vgl. Weinzierl 2006, S. 26

²¹³ vgl. Braun 2013, S. 56

²¹⁴ vgl. Burghardt 2008, S. 395

²¹⁵ vgl. Müller 2007, S. 89

²¹⁶ vgl. Burghardt 2008, S. 397

²¹⁷ vgl. Pfeifer-Silberbach 2005, S. 69

²¹⁸ vgl. Müller 2007, S. 84

2.4.4 Übergreifende und ganzheitliche Modelle der Produktentstehung

Übergreifende und ganzheitliche Modelle der Produktentstehung verfolgen den Anspruch,

- mehrere Auflösungsgrade des Produktentstehungsprozesses,
- verschiedene Aktivitäten und Phasen der Produktentstehung und
- die unterschiedlichen Stakeholder im Prozess wie z.B. Kunde, Management und Entwicklung

übergreifend und ganzheitlich abbilden und adressieren zu können. Erst hierdurch lässt sich die Komplexität realer Produktentstehungsprozesse abbilden und die Nachteile und Limitationen stark zweckfokussierter Ansätze, die oftmals relevante Wechselwirkungen unberücksichtigt lassen, überwinden.

Münchener Produktkonkretisierungsmodell (MKM)

Das Münchener Produktkonkretisierungsmodell (MKM) beschreibt ein generelles Vorgehen beim Entwickeln und Konstruieren und stellt damit eine Weiterentwicklung der VDI-Richtlinie 2221 dar²¹⁹. Dabei überwindet das Münchener Produktkonkretisierungsmodell das stark phasenorientierte Vorgehen der VDI-Richtlinie 2221, indem es die starken Wechselwirkungen zwischen Anforderungsraum und Lösungsraum im gesamten Entwicklungsprozess hervorhebt (vgl. Abbildung 2-26).

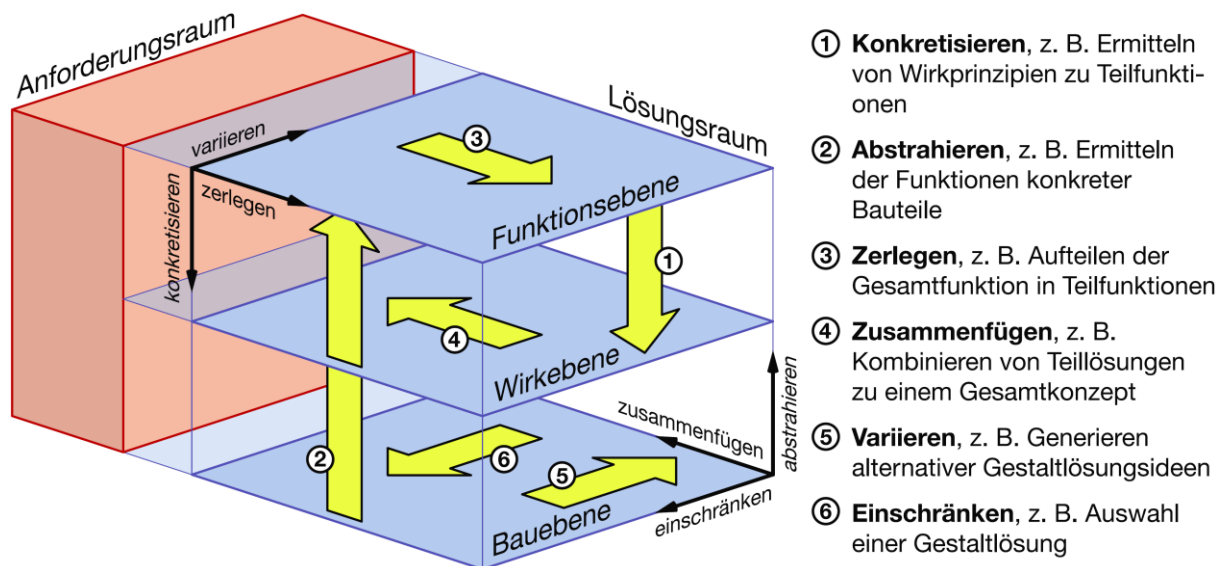


Abbildung 2-26: Münchener Produktkonkretisierungsmodell (MKM)²²⁰

Der **Anforderungsraum** repräsentiert die technischen Entwicklungsziele bzw. geforderten Produkteigenschaften. Der **Lösungsraum** spannt die Menge aller

²¹⁹ vgl. Zingel 2013, S. 27

²²⁰ vgl. Ponn & Lindemann 2011, S. 28

Lösungsmöglichkeiten zur Umsetzung der Anforderungen in einem Produkt auf. PONN UND LINDEMANN²²¹ weisen darauf hin, dass parallel zur Konkretisierung der Lösungen im Lösungsraum, die Anforderungen ergänzt, detailliert und konkretisiert werden müssen, weshalb sich die Anforderungen über die gesamte Achse der Produktkonkretisierung ziehen. Eine Anforderungsklä rung zu Beginn des Entwicklungsprojektes stellt die Weichen für alle folgenden Entwicklungsaktivitäten. Im Verlauf des weiteren Entwicklungsprozesses wird der Anforderungsraum entsprechend der drei Konkretisierungsebenen **Funktions-, Wirk- und Bauebene** konkretisiert. Die Funktionsebene beinhaltet Funktionen, die das Produkt bzw. Bestandteile des Produkts auf abstrahierter Ebene zweckorientiert beschreiben. Im Funktionsmodell sind die Funktionen des Produktes sowie deren Zusammenhänge (Funktionsstruktur) dargestellt. Auf der Wirkebene lassen sich die prinzipiellen Lösungsideen und Lösungskonzepte einer technischen Problemstellung darstellen. Wirkprinzipien zur Realisierung einzelner Teilfunktionen werden dabei zur Wirkstruktur und hierauf aufbauend zu Gesamtkonzepten verknüpft. Die Bauebene beinhaltet die konkrete Gestalt eines Produkts, die erforderlich ist, um das Produkt herstellen zu können. Ergebnis ist das Baumodell, das alle Bauteile und Baugruppen sowie deren Verknüpfung in der Baustruktur enthält.

PONN UND LINDEMANN²²² betonen, dass das Modell einer flexiblen Anwendung bedarf. Die Übergänge zwischen den Ebenen des Lösungsraums sind fließend und die Konkretisierung erfolgt durch zahlreiche Iterationen und Änderungen bzw. Verfeinerungen. Der Entwicklungsprozess stellt tendenziell einen kontinuierlichen Übergang vom Abstrakten zum Konkreten dar – je nach Entwicklungssituation sind jedoch auch Schritte der Abstraktion oder ein mehrfacher Wechsel zwischen den Konkretisierungsebenen des Lösungsraums vonnöten.

3-Zyklen-Modell der Produktentstehung

Das 3-Zyklen-Modell der Produktentstehung beschreibt den Produktentstehungsprozess als iterative Interaktion der drei Zyklen Strategische Produktplanung, Produktentwicklung und Produktionssystementwicklung²²³. Damit wird die Produktentwicklung nicht mehr isoliert betrachtet, sondern als eine von drei Zyklen in den Gesamtprozess von der Produkt- bzw. Geschäftsidee bis zum Serienanlauf integriert (vgl. Abbildung 2-27). Die iterative Natur dieses Prozesses wird dadurch in

²²¹ vgl. Ponn & Lindemann 2011, S. 27

²²² vgl. Ponn & Lindemann 2011, S. 28

²²³ vgl. Gausemeier, Plass & Wenzelmann 2009

den Mittelpunkt gerückt, dass die drei Bereiche als Zyklen und nicht als stringente Folge von Phasen und Meilensteinen aufgefasst werden²²⁴.

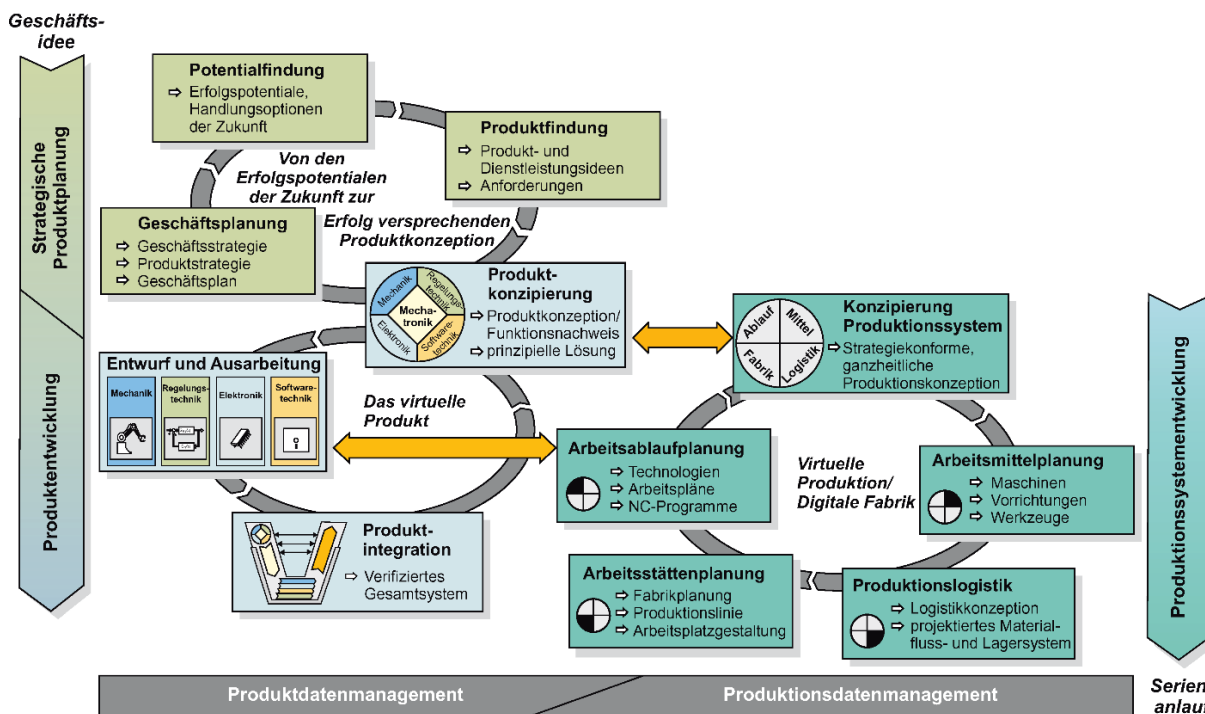


Abbildung 2-27: 3-Zyklen-Modell der Produktentstehung²²⁵

Der erste Zyklus²²⁶ (**Strategischen Produktplanung**) ist charakterisiert durch das Finden von zukünftigen Erfolgspotenzialen. Der Zyklus umfasst die Aufgabenbereiche Potenzialfindung, Produktfindung, Geschäftsplanung und Produktkonzipierung. Das Ziel der Potenzialfindung ist das Erkennen der Erfolgspotenziale der Zukunft sowie die Ermittlung entsprechender Handlungsoptionen. Basierend auf den erkannten Erfolgspotenzialen befasst sich die Produktfindung mit der Suche und der Auswahl neuer Produkt- und Dienstleistungsideen. In der Geschäftsplanung wird die Geschäfts- und Produktstrategie zur Umsetzung der neuen Idee erstellt und in Form eines Geschäftsplans fixiert. Der Geschäftsplan erbringt den Nachweis, ob mit der neuen Idee ein attraktiver Return on Investment zu erzielen ist. Der zweite Zyklus²²⁶ (**Produktentwicklung**) umfasst die fachgebietsübergreifende Produktkonzipierung, den fachgebietspezifischen Entwurf und die entsprechende Ausarbeitung sowie die Integration der Ergebnisse der einzelnen Fachgebiete zu einer Gesamtlösung. Der dritte Zyklus²²⁶ (**Produktionssystementwicklung**) umfasst die Konzipierung des Produktionssystems, wobei die vier Aspekte Arbeitsablaufplanung, Arbeitsmittelpassung, Arbeitsstättenplanung und Produktionslogistik integrativ betrachtet werden

²²⁴ vgl. Braun 2013, S. 63

²²⁵ vgl. Gausemeier, Lanza & Lindemann 2012, S. 16

²²⁶ vgl. Gausemeier, Tschirner & Dumitrescu 2013

müssen. Die Produkt- und Produktionssystementwicklung müssen parallel und eng aufeinander abgestimmt vorangetrieben werden. Nur so kann sichergestellt werden, dass alle Möglichkeiten der Gestaltung eines leistungsfähigen und kostengünstigen Produktes ausgeschöpft werden.

Integriertes Produktentstehungsmodell (iPeM)

Das integrierte Produktentstehungsmodell (iPeM) nach ALBERS basiert auf dem ZHO-Modell der Systemtechnik und ist ein ganzheitliches Rahmenwerk zur Erfassung und Unterstützung der Produktentstehung, das verschiedene Ansätze und Methoden der Produktentstehung integriert²²⁷. Das iPeM beschreibt die Produktentstehung auf Basis eines generischen Metamodells, als Überführung eines (anfangs vagen) Zielsystems in ein Objektsystem mit Hilfe von Aktivitäten²²⁸. Die Aktivitäten bilden zusammen mit dem Ressourcensystem ein Handlungssystem und lassen sich in Aktivitäten der Produktentstehung und Aktivitäten der Problemlösung untergliedern²²⁹ (vgl. Abbildung 2-28). Aktivität wird hierbei als Oberbegriff für alle Aggregationsebenen von Arbeitseinheiten verstanden²³⁰. ALBERS UND BRAUN konnten zeigen, dass aufgrund der generischen Natur des iPeM diese wenigen Elemente ausreichend sind, um die Produktentstehung ganzheitlich zu umfassen²³¹.

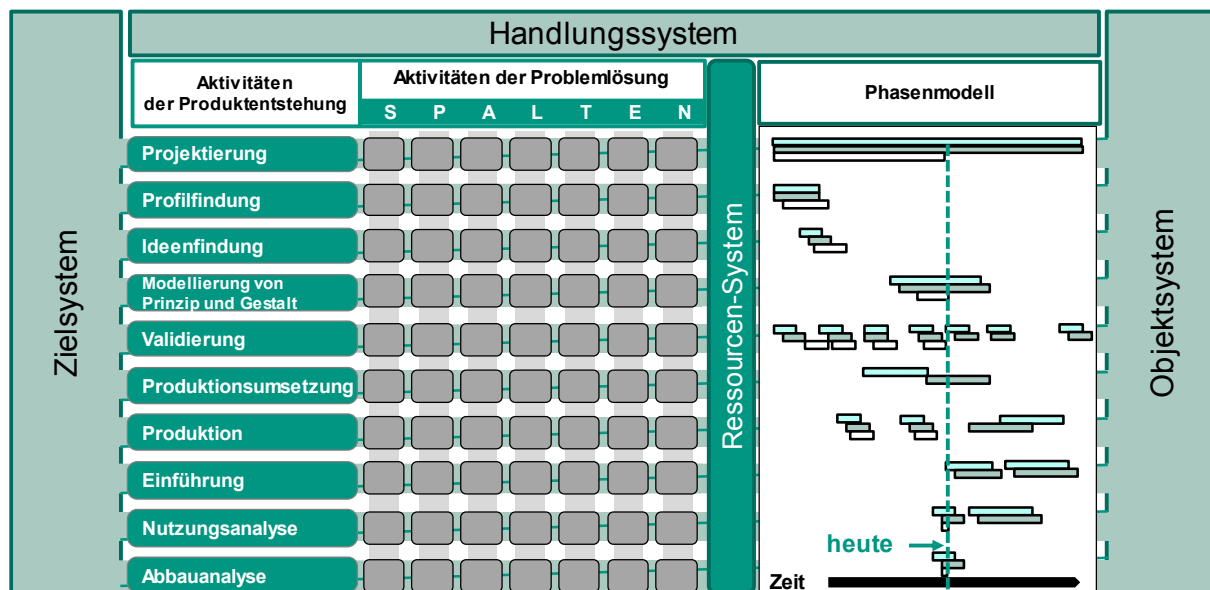


Abbildung 2-28: Das integrierte Produktentstehungs-Modell (iPeM)²²⁷

²²⁷ vgl. Albers & Braun 2011b

²²⁸ vgl. Albers 2010

²²⁹ vgl. Braun 2013, S. 86

²³⁰ vgl. Angermeier 2015

²³¹ vgl. Albers & Braun 2012

Die **Aktivitäten der Produktentstehung**²³² (Produktentstehungsaktivitäten – PE-Aktivitäten) orientieren sich am Produktlebenszyklus, sind in ihrer zeitlichen Abfolge jedoch nicht daran gebunden. Die **Aktivitäten der Problemlösung** (Problemlösungsaktivitäten – PL-Aktivitäten) entsprechen den sieben Schritten des Problemlösungsprozesses SPALTEN. Durch orthogonale Anordnung der Aktivitäten der Produktentstehung über den Aktivitäten der Problemlösung entsteht eine Matrix aus 70 Aktivitäten (**Aktivitätenmatrix**), die eine generische und projektunabhängige Modellierung von Produktentstehungsprozessen ermöglicht. Der Projektbezug entsteht erst, wenn im Phasenmodell einzelne Produktentstehungsaktivitäten in einen zeitlichen Bezug gesetzt werden. Hierdurch lassen sich parallele und/oder iterative Aktivitäten projektbezogen in Phasen anordnen, wodurch nach ALBERS der sequenzielle Ablauf klassischer Phasenmodelle überwunden²³³ und der Individualität jedes einzelnen Produktentstehungsprozesses²³⁴ Rechnung getragen werden kann. Auf oberster Aggregationsebenen decken die definierten Produktentstehungsaktivitäten hierbei sämtliche Aktivitäten ab, die im Rahmen der Produktentstehung durchgeführt werden. Zur Operationalisierung müssen diese Produktentstehungsaktivitäten im Regelfall branchen-, firmen- bzw. produktspezifisch durch weitere Produktentstehungsaktivitäten verfeinert werden. Dabei bleiben die Problemlösungsaktivitäten auf jeder Aggregationsebene erhalten, so dass jede Produktentstehungsaktivität durch den stets gleichen Problemlösungsprozess bearbeitet werden kann. Die entstehende Untergliederung definierter Projektphasen in projektspezifische Produktentstehungsaktivitäten ermöglicht es, eine Durchgängigkeit zwischen makroskopischer und mikroskopischer Prozessmodellierung zu erreichen. Hierdurch lassen sich die Bedarfe des Managements und der Entwicklung an Prozessmodellierung gleichermaßen bedienen, wodurch beispielsweise ein Produktentstehungsprozess aus Sicht des Managements und der Entwicklung integrativ modelliert und verbessert werden kann²³⁵. Weiterhin können die übertragbaren Inhalte projektspezifischer Produktentstehungsaktivitäten in der Aktivitätenmatrix hinterlegt und somit das generelle Domänenwissen²³⁶ eines Unternehmens gesichert bzw. ausgebaut werden²³⁷. Hierdurch wird ein entscheidender Mehrwert im Umgang mit Wissen

²³² vgl. Braun 2013, S. 88

²³³ vgl. Albers, Sadowski & Marxen 2011

²³⁴ vgl. Albers 2010

²³⁵ vgl. Albers, Braun & Ebel 2013

²³⁶ vgl. Klein 1998

²³⁷ vgl. Meboldt 2008, S. 199

geschaffen, der insbesondere im Rahmen der **Produktgenerationsentwicklung** nach ALBERS²³⁸ von besonderer Bedeutung ist.

2.4.5 Zwischenfazit

Produktentstehungsprozesse müssen zum Ziel haben, mit der immanenten Komplexität und Unsicherheit heutiger Produktentstehung erfolgreich umzugehen und damit Produkte hervorzubringen, die im Markt erfolgreich sind. Hierzu müssen individuelle Problemlösungsprozesse und Aktivitäten auf Entwicklungsebene ebenso berücksichtigt werden, wie die übergeordnete Planung und Steuerung ganzer Entwicklungsprojekte auf Managementebene mittels Phasen und Meilensteinen. Nur durch eine ganzheitliche Betrachtung von Aktivitäten und Phasen, den zugrundeliegenden Zielen und resultierenden Objekten sowie der Vernetzung all dieser Elemente, können heutige, stark iterativ verlaufende Produktentstehungsprozesse abgebildet und damit geplant und gesteuert werden. Ein zentraler Fokus muss dabei auf den kontinuierlichen Abgleich zwischen Zielen und Objekten (Verifizierung und Validierung) gelegt werden, da nur so zeit- und kostspielige Cross-Phase-Iterationen zu Gunsten erwünschter In-Phase-Iterationen vermieden werden können. Die Verifizierung und Validierung darf dabei nicht als nachgelagerter Prozessschritt in einem sequenziell verlaufenden Produktentstehungsprozess implementiert sein – Validierung muss als zentrale Aktivität der Produktentstehung verstanden werden²³⁹ und muss kontinuierlich im Produktentstehungsprozess durchgeführt werden. Der hierdurch realisierbare Erkenntnisgewinn im gesamten Verlauf der Produktentstehung erfordert eine kontinuierliche Erweiterung und Konkretisierung des Zielsystems entsprechend der Entwicklung des Objektsystems. Zur Steuerung solcher validierungsbasierter Produktentstehungsprozesse bedarf es zum einen geeigneter Prozessziele, die über die punktuelle Vorgabe von Arbeitsergebnissen, die zu einzelnen Meilensteinen vorliegen müssen, hinausgehen. Zum anderen muss der aktuelle Reifegrad des Produktes und der zugrundeliegenden Ziele erfasst und bewertet werden können.

2.5 Zielsysteme in der Produktentstehung

Das Zielsystem in der Produktentstehung beinhaltet nach ALBERS alle relevanten Ziele, deren Wechselwirkungen, Randbedingungen und Begründungen, die für die Entwicklung des richtigen Produktes erforderlich sind²⁴⁰. BADER stellt heraus, dass

²³⁸ vgl. Albers, Bursac & Wintergerst 2015

²³⁹ vgl. Albers 2010

²⁴⁰ vgl. Albers 2010

hierbei die Produktziele zwar eine zentrale Rolle einnehmen, diese jedoch mit einer Vielzahl weiterer Ziele interagieren (vgl. Abbildung 2-29)²⁴¹.

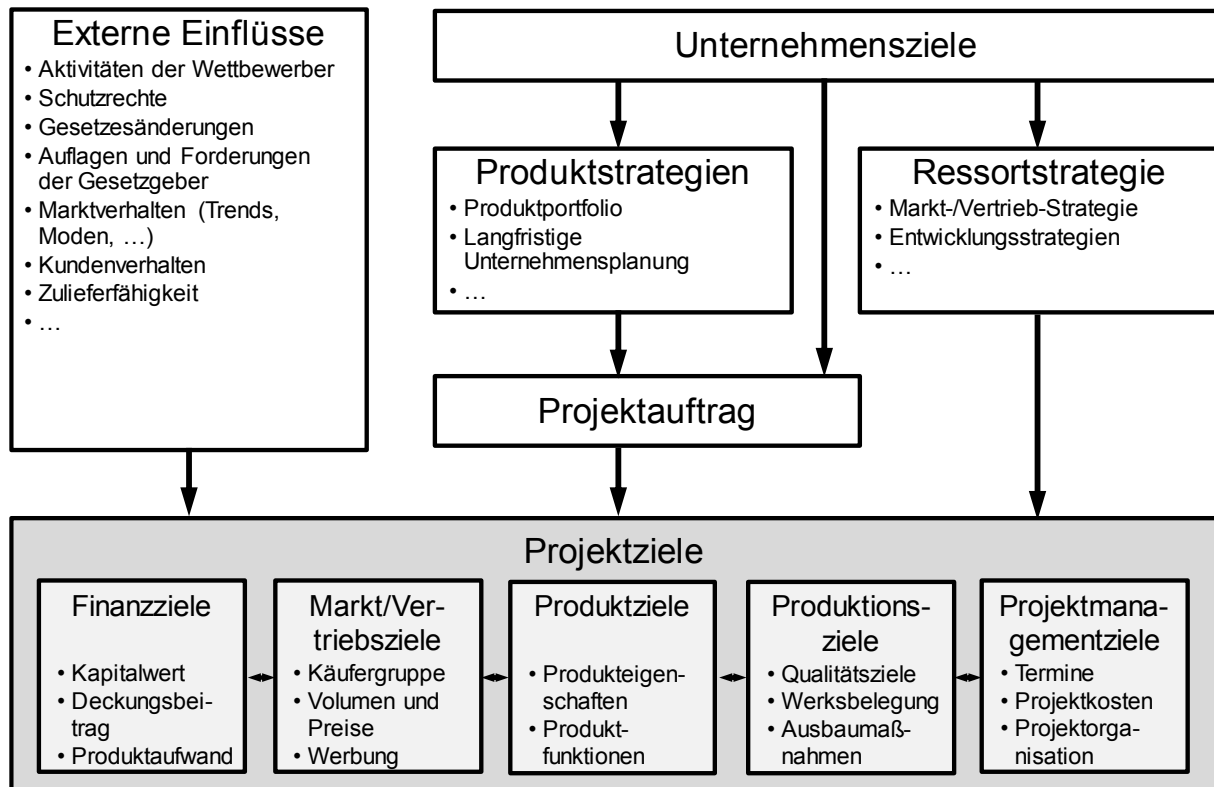


Abbildung 2-29: Zielsystem eines Entwicklungsprojekts²⁴²

Unternehmensziele umfassen die Zielsetzung auf höchster Unternehmensebene, beispielsweise die mittel- oder langfristige Absicherung des Unternehmenserfolgs durch Maximierung des Gewinns²⁴³. Auf Basis hieraus abgeleiteter Produktstrategien werden einzelne Projektaufträge erstellt, mit denen die Entwicklung eines Produkts beauftragt wird. Aus den Zielvorgaben im Projektauftrag sowie unter Berücksichtigung der Ressortstrategien werden die Projektziele abgeleitet, die sich nach BADER in fünf miteinander wechselwirkende Sub-Zielsysteme untergliedern lassen²⁴⁴. Die Finanzziele legen z.B. das einzusetzende Kapital und das zu erwirtschaftende Ergebnis fest. Die Markt/Vertriebs-Ziele umfassen beispielsweise die geplanten Absatzzahlen der verschiedenen Märkte, die anvisierten Verkaufspreise oder die beabsichtigten Werbemaßnahmen. Die Produktziele beschreiben das zu entwickelnde Produkt. Die Produktionsziele bestimmen u.a. die Belegung der Produktionsstätten oder für das Projekt erforderliche Erweiterungs- und Baumaß-

²⁴¹ vgl. Bader 2007, S. 19

²⁴² vgl. Bader 2007, S. 20

²⁴³ vgl. Wöhe & Döring 2013

²⁴⁴ vgl. Bader 2007, S. 21

nahmen der Produktionsanlagen. Die Projektmanagementziele umfassen Forderungen an Termine, an die Organisation und an die Projektkosten. Neben diesen unternehmensinternen Einflüssen auf das Zielsystem eines Entwicklungsprojekts müssen insbesondere Einflüsse des Unternehmensumfelds auf das Projekt bei der Definition stimmiger Projekt- bzw. Produktziele berücksichtigt werden.

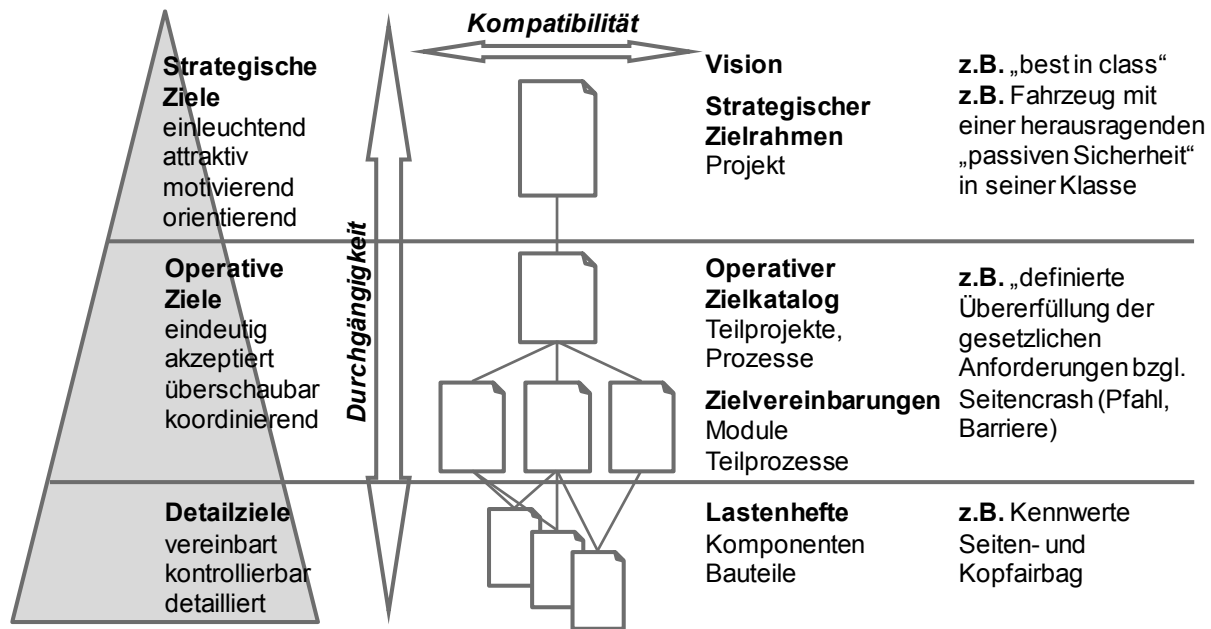


Abbildung 2-30: Durchgängigkeit und Kompatibilität im Zielsystem²⁴⁵

Eiletz²⁴⁶ fordert zur Operationalisierung des Zielsystems eine **Durchgängigkeit der Ziele** von abstrakten strategischen Zielen bis hin zu konkreten Detailzielen, um so die Ausrichtung der Ziele an den jeweils übergeordneten Zielen sicherzustellen (vgl. Abbildung 2-30). Darüber hinaus müssen die Elemente des Zielsystems der Organisationsstruktur des Entwicklungsprojekts bzw. den Entwicklungsumfängen zugeordnet werden, da nur so die Verbindlichkeit der Ziele gewährleistet werden kann. Die hierzu erforderliche Fragmentierung des Zielsystems hat zur Folge, dass in besonderem Maße auf die **Kompatibilität der Ziele** untereinander zu achten ist²⁴⁷.

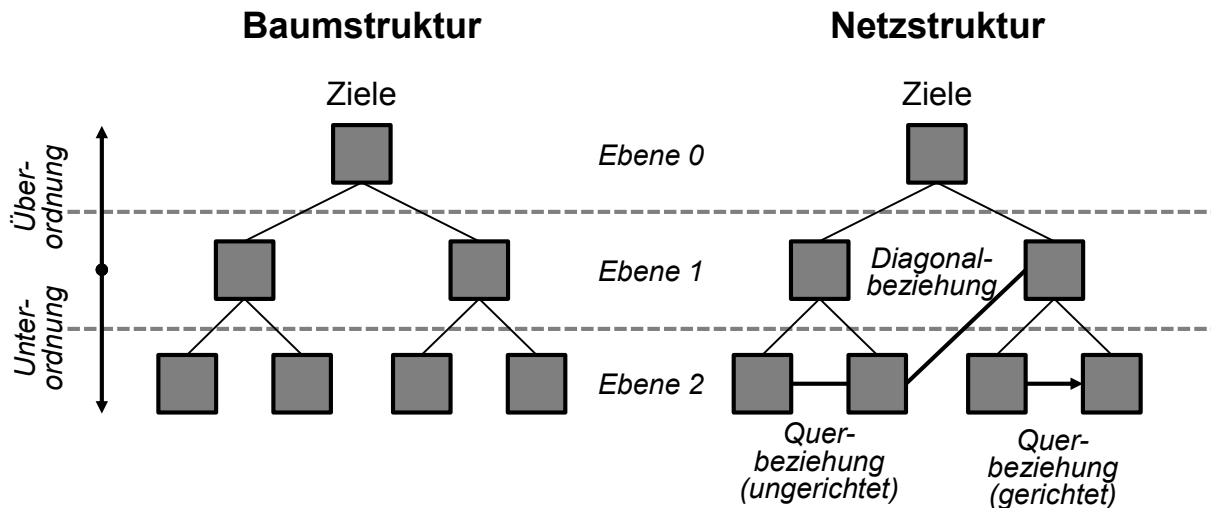
Zur Sicherstellung der Durchgängigkeit und Kompatibilität der Ziele schlägt GEBAUER²⁴⁸ eine semantische Vernetzung der Ziele im Zielsystem vor. GEBAUER unterscheidet hierzu grundsätzlich zwischen hierarchischen und nicht-hierarchischen Beziehungen zwischen den einzelnen Elementen des Zielsystems (vgl. Abbildung 2-31).

²⁴⁵ vgl. Eiletz 1999, S. 61

²⁴⁶ vgl. Eiletz 1999, S. 61

²⁴⁷ vgl. Eiletz 1999, S. 62

²⁴⁸ vgl. Gebauer 2001, S. 45

Abbildung 2-31: Grundsätzliche Beziehungsarten im Zielsystem²⁴⁹

Die **hierarchischen Beziehungen** bilden dabei eine Baumstruktur aus, innerhalb derer zwischen Abstraktions- und Bestandsbeziehungen unterschieden werden kann. Die **nicht-hierarchischen Beziehungen** bilden eine Netzstruktur aus, die sich mit der hierarchischen Baumstruktur überlagert. Hierbei kann zum einen zwischen ungerichteten und gerichteten Beziehungen unterschieden werden. Eine **ungerichtete Beziehung** besteht, wenn zwei Elemente des Zielsystems sich gegenseitig beeinflussen (bidirektional), wohingegen eine **gerichtete Beziehung** die Beeinflussung eines Elements durch ein anderes zum Ausdruck bringt (unidirektional). Zum anderen können nicht-hierarchische Beziehungen in Neben- und Diagonalbeziehungen unterschieden werden. Zur Bildung der semantischen Netzstruktur innerhalb des Zielsystems schlägt GEBAUER²⁵⁰ sechs Relationen zwischen Elementen des Zielsystems vor:

1. Zerlegt sich in – Ein Element ist in zwei oder mehr Elemente zerlegt
2. Setzt sich zusammen aus – Ein Element setzt sich aus zwei oder mehr Elementen zusammen
3. Erzeugt aus – Ein Element wird aus einem anderen Element erzeugt
4. Unterstützend – Ein Element wirkt unterstützend auf ein anderes Element
5. Konkurrierend – Ein Element konkurriert mit einem anderen Element
6. Gegensätzlich – Ein Element ist zu einem anderen Element gegensätzlich

Auch KRUSE²⁵¹ unterscheidet zwischen Struktur- und Semantikbeziehungen im Zielsystem. **Strukturbeziehungen** bilden die Hierarchie im Zielsystem aus und führen zu einer Konkretisierung, Spezialisierung, Dekomposition oder Variation der

²⁴⁹ vgl. Gebauer 2001, S. 48

²⁵⁰ vgl. Gebauer 2001, S. 50

²⁵¹ vgl. Kruse 1996, S. 78

einzelnen Elemente. **Semantikbeziehungen** beschreiben die Wirkungen der einzelnen Elemente des Zielsystems aufeinander und können zwischen ausschließend, konkurrierend und unterstützend unterschieden werden.

Um auf Basis dieser Beziehungen ein Produkt übergreifend auf Systemebene beschreiben zu können, schlägt STECHERT²⁵² eine Untergliederung in elf **Partialmodelle** vor, die im Produktentstehungsprozess durchgängig eingesetzt werden sollen. Hierdurch können alle Aspekte, die für die Entwicklung des richtigen Produkts erforderlich sind, strukturiert und grundsätzlich in hierarchischer Form modelliert werden.

- **Stakeholdermodell** – Im Stakeholdermodell werden sämtliche über den Produktlebenslauf involvierte Rollen erfasst und strukturiert abgebildet.
- **Produktlebenslaufmodell** – Das Produktlebenslaufmodell umfasst sämtliche Aktivitäten, die während des Produktlebens durchgeführt werden.
- **Produktumgebungsmodell** – Das Modell der Produktumgebung beschreibt außerhalb der Systemgrenzen liegende Gegebenheiten, die Einfluss auf das Produkt haben.
- **Systemideemodell** – Das Systemideemodell stellt die verschiedenen Aspekte der Systemidee dar.
- **Zielmodell** – Das Zielmodell beinhaltet alle Ziele, die das zukünftige Produkt erfüllen soll.
- **Anforderungsmodell** – Das Anforderungsmodell bildet alle notwendigen Anforderungen ab und stellt diese z.B. als Vertragsgrundlage, Entwicklungslaufplan, Statusüberprüfung oder zum Testen bereit.
- **Funktionsmodell** – Im Funktionsmodell wird das zu entwickelnde Produkt in die zu erfüllenden Funktionen zerlegt und z.B. in Form einer Funktionsstruktur dargestellt.
- **Strukturmodell** – Das Strukturmodell stellt die strukturellen Komponenten dar. Es werden die einzelnen Module (z.B. Gelenk) und deren Anordnung modelliert.
- **Verhaltensmodell** – Dieses Modell beschreibt das Verhalten des Produkts. Es werden Eigenschaften beschrieben, die über die reine Funktionserfüllung hinausgehen.
- **Kostenmodell** – Im Kostenmodell werden alle interessierenden Kosten erfasst und bestimmten Kostenträgern zugewiesen.

²⁵² vgl. Stechert 2010, S. 41

Testmodell – Das Testmodell beinhaltet die Testkriterien, die für eine Bewertung und Abnahme des Produkts bzw. von Konzepten notwendig sind.

FRANK²⁵³ schlägt zur Beschreibung der Prinziplösung selbstoptimierender Systeme eine Spezifikationstechnik vor, die aus neun **kohärenten Partialmodellen** besteht (vgl. Abbildung 2-32).

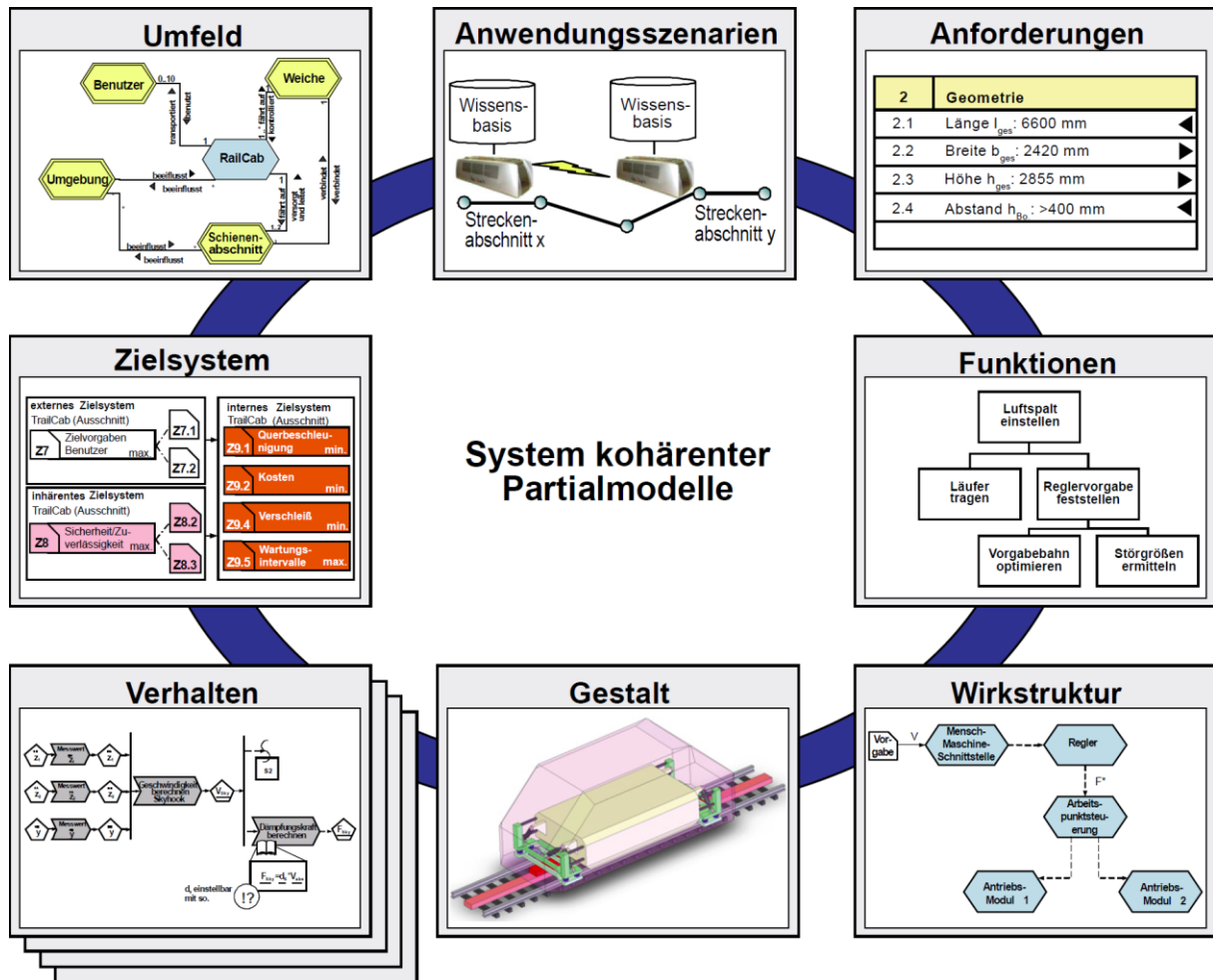


Abbildung 2-32: Spezifikationstechnik zur Beschreibung der Prinziplösung²⁵³

FRANK²⁵⁴ betont, dass es während des Entwurfs eines Systems nicht möglich und auch nicht gewollt ist, die einzelnen Partialmodelle detailliert und konkret zu beschreiben. Vielmehr muss der grundlegende Aufbau und die Wirkungsweise des Systems identifiziert und darüber im Entwicklungsteam Einigkeit erzielt werden. Die einzelnen Partialmodelle enthalten demnach keine vollständigen Beschreibungen des Systems, sondern bilden lediglich einzelne, jedoch relevante Aspekte des Systems ab.

²⁵³ vgl. Frank 2006, S. 79

²⁵⁴ vgl. Frank 2006, S. 103

- **Zielsystem** – Das Zielsystem repräsentiert Ziele und ihre Verknüpfungen, wobei Ziele das geforderte, gewünschte oder zu vermeidende Verhalten eines Systems formulieren.
- **Umfeld** – Das Umfeldmodell beschreibt das Umfeld des zu entwickelnden Systems und seine Einbettung in das Umfeld sowie relevante Einflussbereiche und Einflüsse auf das System.
- **Anwendungsszenarien** – Anwendungsszenarien spezifizieren, in welcher Art und Weise sich das System in einem Zustand und einer bestimmten Situation verhalten soll bzw. welche Zustandsübergänge aufgrund welcher Ereignisse stattfinden sollen.
- **Anforderungen** – Dieses Partialmodell stellt die rechnerinterne Repräsentation der Anforderungen dar, die in Form von Anforderungslisten vorliegen.
- **Funktionen** – Das Funktionsmodell behandelt eine hierarchische Gliederung der Gebrauchsfunktionen zur Definition der grundsätzlichen Funktionalität des Systems.
- **Wirkstruktur** – In der Wirkstruktur werden die Lösungsmuster (Wirkprinzipien, Softwaremuster und Wirkmuster) repräsentierenden Systemelemente, deren Merkmale sowie die Beziehung der Systemelemente zueinander abgebildet, wodurch die grundsätzliche Struktur des Systems abgebildet wird.
- **Gestalt** – Dieses Modell beinhaltet Angaben über Anzahl, Form, Lage, Anordnung sowie Art der Wirkflächen und Wirkflächenpaare. Das Gestaltmodell beinhaltet insbesondere die Baustruktur des Systems.
- **Verhalten** – Das Verhaltensmodell bildet das logische und physikalische Verhalten des Systems ab. Fokus des logischen Verhaltens bildet die Reihenfolge von möglichen Zuständen und Aktivitäten des Systems und damit das Kommunikations- und Kooperationsverhalten zwischen einzelnen Systemen. Beispiele für physikalisches Verhalten eines Systems sind die Festigkeit, Temperaturbeständigkeit und elektromagnetische Verträglichkeit eines Systems oder dessen kinematisches bzw. dynamische Verhalten.

Die einzelnen Partialmodelle werden im Produktentstehungsprozess sukzessive im Wechselspiel entwickelt, wobei FRANK²⁵⁵ diese Entwicklung hauptsächlich den Phasen Planen und Klären der Aufgabenstellung und Konzipierung²⁵⁶ zuordnen. ALBERS²⁵⁷ hingegen sieht die Entwicklung der Partialmodelle als eine Aufgabe an, die kontinuierlich im gesamten Prozess der Produktentstehung zu gestalten ist. In den meisten Fällen schließt dies sogar die Zeitspanne ab Produktionsstart bzw.

²⁵⁵ vgl. Frank 2006, S. 130

²⁵⁶ vgl. Pahl et al. 2007, S. 194

²⁵⁷ vgl. Albers, Lohmeyer & Ebel 2011

Produkteinführung im Markt mit ein, da durch die Produktion und Nutzung des entwickelten Produktes oftmals neue Erkenntnisse einhergehen, die eine Anpassung des Zielsystems erforderlich machen und die im Rahmen der Produktgenerationsentwicklung²⁵⁸ in das Zielsystem des Nachfolgeprodukts einfließen.

Diese verschiedenen Paritalmodelle sind erforderlich, um die von EILETZ²⁵⁹ geforderte Durchgängigkeit im Zielsystem zu realisieren, da nur so die Nachvollziehbarkeit²⁶⁰ (traceability) der Ziele und Anforderungen erreicht werden kann. EDWARDS UND HOWELL verstehen unter **Nachvollziehbarkeit** „provide a relationship between the requirements, the design and the final implementation of the system“²⁶¹. POHL²⁶² listet verschiedene Aspekte und Techniken der Systementwicklung auf, die durch Nachvollziehbarkeitsinformationen unterstützt bzw. ermöglicht werden:

- **Nachweisbarkeit** – Nachweis, dass eine Anforderung im entwickelten System (korrekt und vollständig) berücksichtigt wurde.
- **Gold Plating** – Aufdecken von Funktionen oder Qualitäten, die nicht auf Anforderungen zurückgeführt werden können und daher keine Rechtfertigung besitzen.
- **Änderungsmanagement** – Ermittlung von Änderungsauswirkungen, d.h. welche Anforderungen, Komponenten oder Testfälle sind von einer Änderung betroffen und wie hoch ist der Änderungsaufwand.
- **Qualitätssicherung, Wartung und Pflege** – Identifikation der Ursachen und Auswirkungen von Fehlern sowie die Bestimmung der von einem Fehler betroffenen Systemteile und die Prognose des Aufwands zu Beseitigung eines Fehlers.
- **Reengineering** – Nachvollziehbarkeit, welche Funktionen eines Vorgängersystems bei Änderungen der Anforderungen betroffen sind.
- **Wiederverwendbarkeit** – Wiederverwendbarkeit von Entwicklungsartefakten auf Basis formulierter Anforderungen.
- **Projektverfolgbarkeit** – Bestimmung des Projektfortschritts auf Basis in der Systemarchitektur berücksichtigter/implementierter/getesteter Anforderungen.
- **Risikomanagement** – Identifikation risikobehafteter Artefakte (z.B. Komponenten).
- **Zurechenbarkeit** – Bestimmung des Entwicklungsaufwands einzelner (Kunden-)Anforderungen.

²⁵⁸ vgl. Albers, Bursac & Wintergerst 2015

²⁵⁹ vgl. Eiletz 1999, S. 61

²⁶⁰ vgl. Ramesh & Jarke 2001

²⁶¹ vgl. Edwards & Howell 1991 nach Ramesh & Jarke 2001

²⁶² vgl. Pohl 2007, S. 504

- **Prozessverbesserung** – Probleme im Entwicklungsprozess können besser auf ihre Ursachen zurückgeführt werden, was eine gezielte Beseitigung der Probleme bzw. Verbesserung des Prozesses ermöglicht.

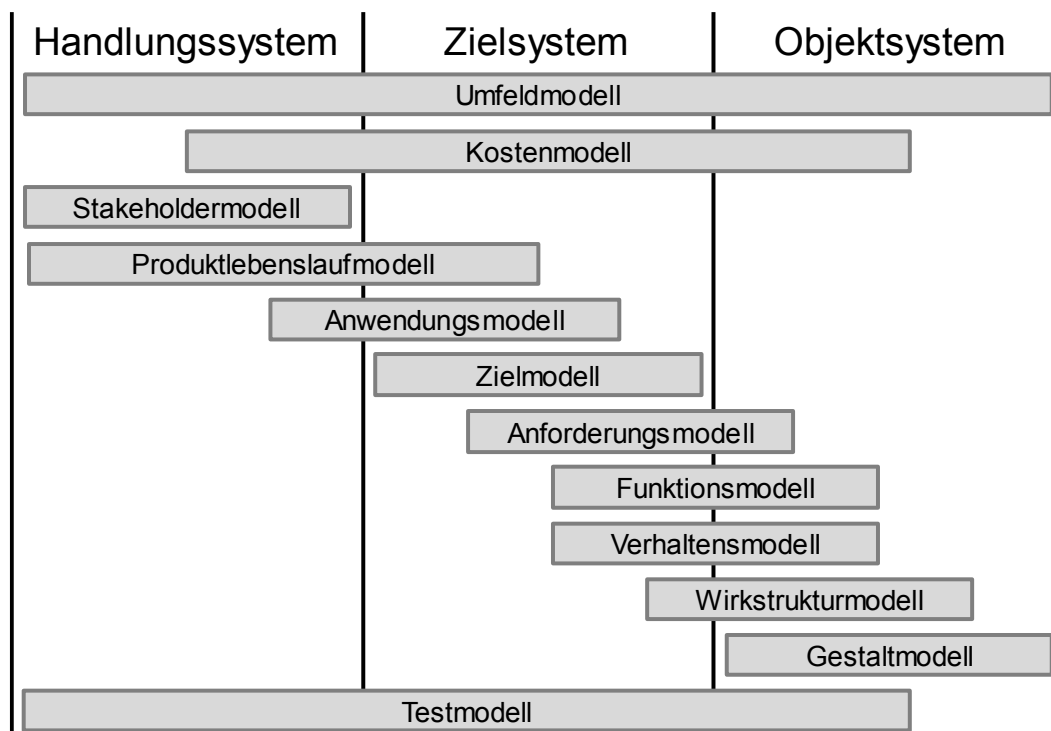
Weiterhin führt POHL²⁶³ 20 Relationen aus fünf Klassen an, mittels derer die die Nachvollziehbarkeit im Zielsystem hergestellt werden kann:

- **Bedingungen (conditions)**
Einschränkung (constraint), Vorbedingung (precondition).
- **Inhalt (content)**
Gleichheit (similar), Vergleich (compares), Widerspruch (contradicts), Konflikt (conflicts).
- **Dokumentation (documents)**
Beispiel für (example for), Testfall für (test-case for), Begründung (purpose), Verantwortlich für (responsible for), Hintergrund (background), Kommentar (comments)
- **Abstraktion (abstraction)**
Klassifikation (classifies), Aggregation (aggregates), Generalisierung (generalizes).
- **Evolution (evolutionary)**
Ersetzung (replaces), Erfüllung (satisfies), Grundlage (based on), Formalisiert (formalizes), Verfeinerung (refines).

Zur Abbildung aller relevanten Ziele, deren Wechselwirkungen, Randbedingungen und Begründungen²⁴⁰ müssen folglich auch solche Partialmodelle abgebildet werden, die nicht primär dem Zielsystem, sondern eher dem Handlungs- bzw. Objektsystem zugeordnet werden können. ALBERS, EBEL UND SAUTER²⁶⁴ schlagen zur Modellierung von projektspezifischen Informationen daher eine Untergliederung der Inhalte entsprechend dem ZHO-Ansatz vor. Hierdurch können auch solche Elemente des Zielsystems nachvollzogen werden, die in Wechselwirkung mit dem Handlungs- bzw. Objektsystem stehen (vgl. Abbildung 2-33). Hierbei müssen die Aspekte, die eher dem Handlungs- und Objektsystem zugeordnet werden, lediglich soweit abgebildet werden, wie es zum Zweck der Nachvollziehbarkeit aller Zielsystem-Elemente erforderlich ist.

²⁶³ vgl. Pohl 2007, S. 508

²⁶⁴ vgl. Albers, Ebel & Sauter 2010

Abbildung 2-33: Partialmodelle der Produktentstehung²⁶⁵**Definition 2-6: Zielsystem²⁶⁶**

Ein vollständiges Zielsystem beinhaltet alle relevanten Ziele, deren Wechselwirkungen, Randbedingungen und Begründungen, die für die Entwicklung des richtigen Produkts erforderlich sind.

Da das Zielsystem im Verlaufe des Produktlebenszyklus²⁶⁷ und hierbei insbesondere im Produktentstehungsprozess fortwährend erweitert und konkretisiert wird²⁶⁸, kann insbesondere zu Beginn eines Produktentstehungsprozesses kein vollständiges Zielsystem vorliegen. Jedoch muss jeder Produktentstehungsprozess mit einem initialen Zielsystem beginnen, das den Impuls für die Entwicklung liefert. Durch dieses initiale Zielsystem wird idealerweise sichergestellt, dass das richtige Produkt entwickelt werden kann, ohne dabei umfangreiche Abänderungen im Zielsystem und damit verbundene überflüssige Iterationen im Produktentstehungsprozess in Kauf nehmen zu müssen. Zur Entwicklung des richtigen Produkts erfordert jedoch auch ein vollständiges initiales Zielsystem weitere Ergänzungen und Verfeinerungen des Zielsystems sowie damit verbundene, nutzenstiftende Iterationen.

²⁶⁵ vgl. Stechert 2010 und Frank 2006

²⁶⁶ vgl. Albers 2010

²⁶⁷ vgl. Gausemeier et al. 2000

²⁶⁸ vgl. Abschnitt 2.2.1

Im Rahmen der Produktgenerationsentwicklung werden daher oftmals wesentliche Anteile des Zielsystems eines Vorgängerprodukts übernommen und lediglich ausgewählte Inhalte abgeändert bzw. ergänzt²⁶⁹. Werden zusätzlich relevante Erkenntnisse, die im Rahmen des Produktlebenszyklus des Vorgängerprodukts gewonnen wurden, in das initiale Zielsystem des Nachfolgeprodukts übernommen, wird ein wesentlicher Beitrag zu einer erfolgreichen Produktentstehung geleistet.

Definition 2-7: Initiales Zielsystem²⁷⁰

Ein vollständiges initiales Zielsystem beinhaltet alle relevanten Ziele, deren Wechselwirkungen, Randbedingungen und Begründungen, die eine Entwicklung des richtigen Produkts ohne überflüssige Iterationen erlauben.

2.5.1 Ziele und Anforderungen

Ziele und Anforderungen sind wesentliche Elemente des Zielsystems und bestimmen maßgeblich die Ausrichtung des zukünftigen Produktes. Dabei wird häufig eine klare Trennung zwischen Zielen und Anforderungen gefordert, die oftmals jedoch nicht vorgenommen wird²⁷¹. Nach OERDING²⁷² lässt sich dies im Kern darauf zurückführen, dass eine klare Trennung zwischen Zielen und Anforderungen nur bedingt möglich ist. LINDEMANN²⁷³ geht davon aus, dass bei der Entwicklung eines Produktes eine Vielzahl von Entwicklungszielen erreicht werden müssen, beispielsweise Termine, Ressourcenplanung oder teamspezifische Ziele. Beziehen sich diese Entwicklungsziele auf das zu entwickelnde Produkt und den dafür notwendigen Entwicklungsprozess, spricht LINDEMANN von Anforderungen. Nach EHRENSPIEL²⁷⁴ sind Ziele eine Soll-Vorstellungen vom Auftraggeber. Sie haben einen Aufforderungscharakter und können unscharf sein. Hieraus werden Anforderungen abgeleitet, die die Solleigenschaften des zukünftigen Produktes beschreiben. Auch CROSS²⁷⁵ unterscheidet zwischen Zielen und Anforderungen. Ziele artikulieren nach CROSS einen Bedarf (need) und beschreiben damit, was durch das zu entwickelnde Produkt erreicht werden soll wohingegen Anforderungen beschreiben, was das Produkt tun oder können soll (nicht was es sein soll). Diese Unterscheidung wird ebenfalls von

²⁶⁹ vgl. Albers, Bursac & Wintergerst 2015

²⁷⁰ vgl. Lohmeyer 2012, S. 66

²⁷¹ vgl. Stechert 2010, S. 11

²⁷² vgl. Oerding 2009, S. 85

²⁷³ vgl. Lindemann 2009, S. 95

²⁷⁴ vgl. Ehrlenspiel 2009, S. 90

²⁷⁵ vgl. Cross 2008, S. 77, S. 105 und S. 106

ALBERS²⁷⁶ geteilt, der eine möglichst transparente Durchgängigkeit zwischen Zielen und Anforderungen fordert. EILETZ²⁷⁷ versteht Ziele als gemeinsam vereinbarte zukünftige Sollzustände, die durch aktives Handeln erstrebt oder vermieden werden. **Ziele** stellen nach EILETZ somit Aussagen darüber dar, was mit einer zu gestaltenden Lösung erreicht bzw. vermieden werden soll, wohingegen **Anforderungen** gewünschte Eigenschaften der Lösung bzw. gewünschte Sachverhalte formulieren. ALBERS²⁷⁶ und BADER²⁷⁸ betonen, dass sich Anforderungen aus Zielen und weiteren (Rand-)Bedingungen ableiten sollen (vgl. Abbildung 2-34). LOHMEYER²⁷⁹ versteht unter einer Randbedingung eine aus dem Umfeld der Entwicklung resultierende Einschränkung, die von anderen verantwortet wird und daher zwar ermittelt, aber nicht eigenständig definiert oder geändert werden kann. MUSCHIK²⁸⁰ unterscheidet hierbei zwischen exogenen Randbedingungen, die außerhalb eines Unternehmens ihren Ursprung haben und endogenen Randbedingungen, die innerhalb eines Unternehmens entstammen.

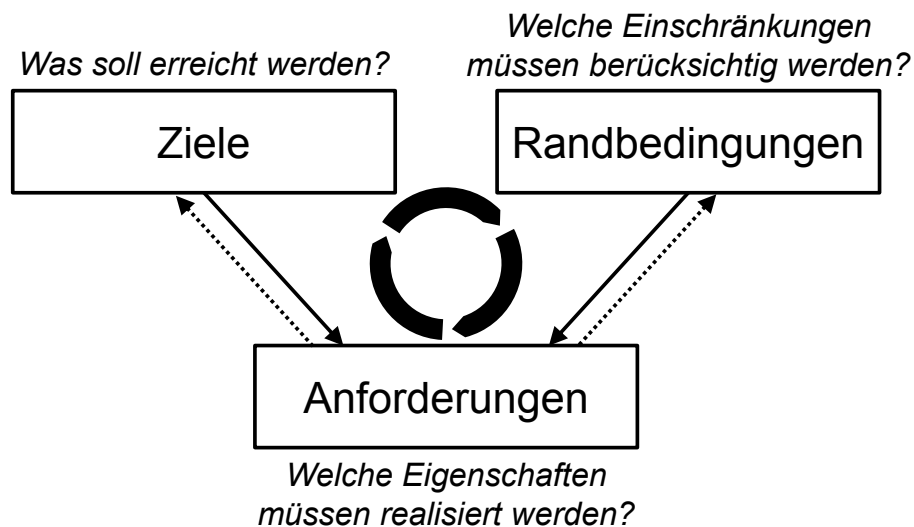


Abbildung 2-34: Ziele, Randbedingungen und Anforderungen

POHL²⁸¹ greift in seinem Werk Requirements Engineering ebenfalls die Begriffe Ziel, Anforderung und Randbedingung auf und setzt diese in Bezug zueinander. POHL versteht unter einem **Ziel**, die intentionale Beschreibung eines charakteristischen Merkmals des zu entwickelnden Systems bzw. des zugehörigen Entwicklungsprozesses. Eine **Anforderung** ist eine Bedingung oder Eigenschaft, die ein System oder

²⁷⁶ vgl. Albers, Klingler & Ebel 2013

²⁷⁷ vgl. Eiletz 1999, S. 8 und S. 10

²⁷⁸ vgl. Bader 2007, S. 12

²⁷⁹ vgl. Lohmeyer 2012, S. 61

²⁸⁰ vgl. Muschik 2010, S. 17

²⁸¹ vgl. Pohl 2007, S. 13, S. 18 und S. 91

eine Person benötigt, um ein Problem zu lösen oder ein Ziel zu erreichen bzw. eine Bedingung oder Eigenschaft, die ein System oder eine Systemkomponente aufweisen muss, um einen Vertrag zu erfüllen oder einem Standard, einer Spezifikation oder einem anderen formell auferlegten Dokument zu genügen. Weiterhin kann unter dem Begriff der Anforderung auch die dokumentierte Repräsentation einer solchen Bedingung oder Eigenschaft verstanden werden. **Randbedingungen** sind nach POHL nur schwer oder gar nicht veränderbar und schränken die Art und Weise ein, wie ein System entwickelt wird (vgl. Abbildung 2-35).

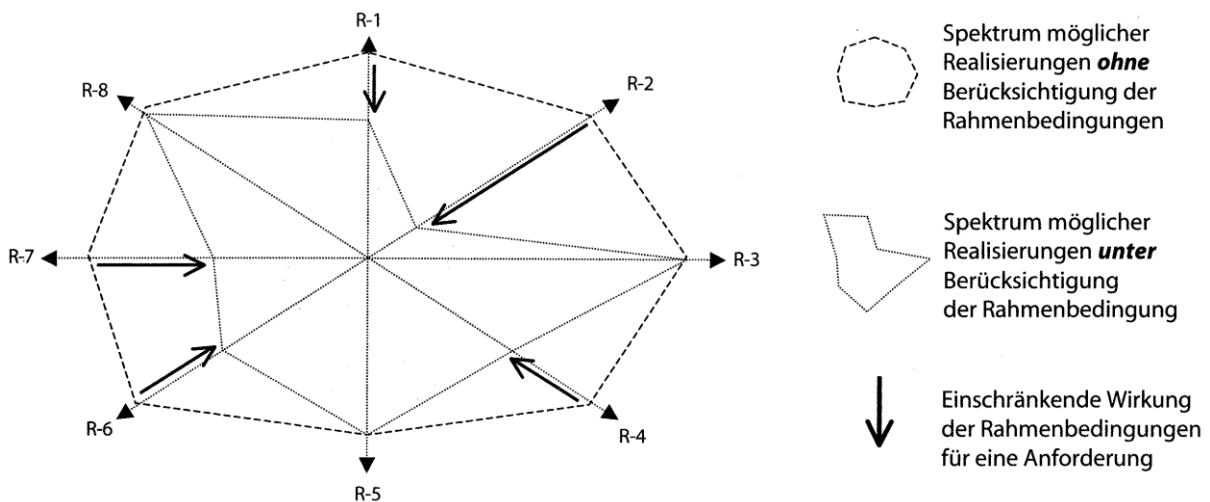


Abbildung 2-35: Wirkung von Randbedingungen²⁸²

Anforderungen werden in der Regel in funktionale und nichtfunktionale Anforderungen differenziert²⁸³. GLINZ gibt eine breite Übersicht über verschiedene Definitionen für funktionale und nichtfunktionale Anforderungen²⁸³. Eine **funktionale Anforderung** beschreibt eine vom System bzw. einer Systemkomponente bereitzustellende Funktion oder einen bereitzustellenden Service²⁸⁴. **Nichtfunktionale Anforderungen** definieren (qualitative) Eigenschaften des gesamten Systems, einer Systemkomponente oder einer Funktion²⁸⁵. POHL²⁸⁶ vermeidet dabei bewusst das Konzept der nichtfunktionalen Anforderung, da diese in realen Spezifikationen oftmals unterspezifizierte Anforderungen darstellen, die bei einer entsprechenden Detaillierung in funktionale Anforderungen und **Qualitätsanforderungen** überführt werden können. Unterspezifizierte Anforderungen bergen dabei die Gefahr, dass sie

²⁸² vgl. Pohl 2007, S. 19

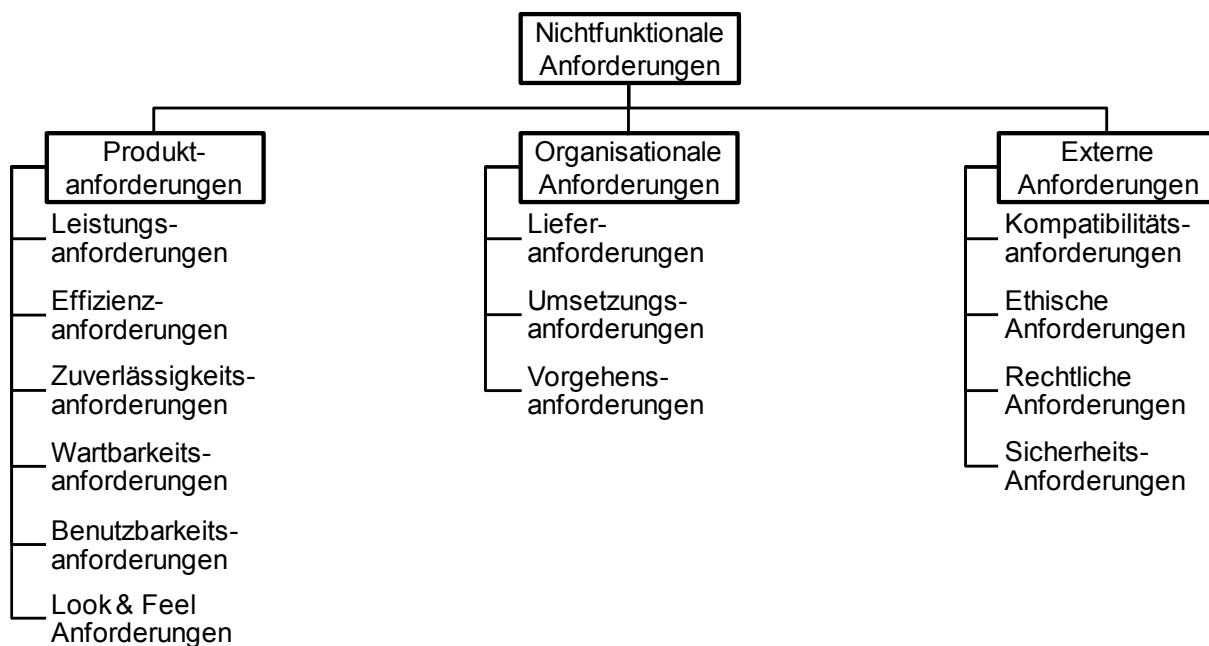
²⁸³ vgl. Glinz 2007

²⁸⁴ vgl. Pohl 2007, S. 15

²⁸⁵ vgl. Robertson & Robertson 2006, S. 175

²⁸⁶ vgl. Pohl 2007, S. 16

ein weites Spektrum von Interpretationen zulassen und damit auch nur schwer überprüfbar sind. Funktionale Anforderungen unterscheiden sich in der Regel von Produkt zu Produkt wohingegen nichtfunktionale Anforderungen oftmals übertragbar sind (vgl. Abbildung 2-36). Eine Untergliederung produktarchitekturrelevanter Anforderungen wird durch das Modell FURPS+ zur Verfügung gestellt²⁸⁷. Es untergliedert Anforderungen in Funktionalität (functionality), Benutzbarkeit (useability), Zuverlässigkeit (reliability), Leistung (Performance) und Unterstützbarkeit (supportability). Unterstützbarkeit subsumiert wiederum Testbarkeit (testability), Anpassbarkeit (adptability), Wartbarkeit (maintainability), Kompatibilität (compatibility), Konfigurierbarkeit (configurability), Installierbarkeit (installability), Skalierbarkeit (scalability) und Ortbarkeit (localizability). Das Plus weist auf zusätzliche Anforderungen hin, die in der Regel Randbedingungen repräsentieren. Hierzu zählen Realisierungs-Anforderungen (design and implementation requirements), Schnittstellen-Anforderungen (interface requirements) und physikalische Anforderungen (physical requirements).

Abbildung 2-36: Nichtfunktionale Anforderungen²⁸⁸²⁸⁷ vgl. Grady 1992²⁸⁸ vgl. Robertson & Robertson 2006 S. 175 und Sommerville 2004

Definition 2-8: Ziel²⁸⁹

Ein Ziel ist die bewusste gedankliche Vorwegnahme eines zukünftigen Soll-Zustands, der gemeinsam vereinbart und durch aktives Handeln erreicht oder vermieden wird.

Definition 2-9: Anforderung²⁹⁰

Eine Anforderung ist

- (1) Eine Eigenschaft oder Bedingung, die ein System oder eine Person benötigt, um ein Problem zu lösen oder ein Ziel zu erreichen.
- (2) Eine Eigenschaft oder Bedingung, die ein System oder eine Systemkomponente aufweisen muss, um einen Vertrag, eine Norm, eine Spezifikation oder ein anderes formell vorgegebenes Dokument zu erfüllen.
- (3) Eine dokumentierte Repräsentation einer Eigenschaft oder Bedingung wie in den ersten beiden Punkten beschrieben.

Definition 2-10: Randbedingung²⁹¹

Eine Randbedingung ist eine aus dem Umfeld der Entwicklung resultierende Einschränkung, die von anderen verantwortet wird und daher zwar ermittelt, aber nicht eigenständig definiert oder geändert werden kann.

POHL²⁹² betont den positiven Effekt, den eine explizite Berücksichtigung von Zielen auf den Umgang mit Anforderungen im Produktentstehungsprozess hat und zeigt folgende Auswirkungen auf:

- **Besseres Systemverständnis** – Ziele fördern ein besseres Systemverständnis und damit die Akzeptanz für das zu entwickelnde System.
- **Gewinnung von Anforderungen** – Ziele leiten die Gewinnung von Anforderungen an.
- **Identifikation und Bewertung von Lösungsalternativen** – Ziele fördern die systematische Identifikation und Bewertung von alternativen Lösungs- bzw. Realisierungsmöglichkeiten.

²⁸⁹ vgl. Eiletz 1999, S. 11

²⁹⁰ vgl. IEEE 610.12 1990

²⁹¹ vgl. Lohmeyer 2012, S. 61

²⁹² vgl. Pohl 2007, S. 89

- **Aufdecken irrelevanter Anforderungen** – Ziele unterstützen die Identifikation irrelevanter Anforderungen.
- **Rechtfertigung von Anforderungen** – Ziele begründen Anforderungen, wodurch sich Anforderungen rechtfertigen lassen.
- **Vollständigkeit der Anforderungen** – Ziele ermöglichen eine Überprüfung der Vollständigkeit der Anforderungen.
- **Identifikation und Auflösung von Konflikten** – Ziele unterstützen die Auflösung von Anforderungskonflikten durch Aufzeigen der zugrundeliegenden Intentionen.
- **Stabilität von Zielen** – Ziele bilden eine stabile Basis für ein Entwicklungsprojekt, da sie sich deutlich seltener ändern als (lösungsorientierte) Anforderungen.

HABERFELLNER ET AL.²⁹³ stellen fest, dass Ziele nicht auf der Hand liegen. Entsprechend betonen sie, dass dem Prozess der **Zielbildung** eine große Bedeutung zukommt, da Ziele nur dann die Lösungssuche steuern können, wenn sie formuliert, allen relevanten Akteuren bekannt und von allen akzeptiert sind. EILETZ²⁹⁴ führt an, dass im Rahmen der Zielbildung eine bewusste Auswahl von Anforderungen aus der Summe der Anforderungen sämtlicher Anforderungssteller getroffen werden muss. Darüber hinaus muss auch eine Antizipation zukünftiger Anforderungen stattfinden, die sinnvoll nur auf Basis vorhandener Ziele erfolgen kann.

Obwohl Anforderungen aus Zielen abgeleitet werden können, kann der Zielbildungsprozess nicht sequentiell verlaufen, beginnend bei Zielen, über Randbedingungen bis hin zu Anforderungen. DARLINGTON UND CULLEY²⁹⁵ führen dies auf die starken Wechselwirkungen zwischen Zielen, Anforderungen und Randbedingungen zurück. So führen Sie an, dass es in der Regel nicht möglich, aus der Beschreibung der Ziele mittels einer direkten logischen Beziehung auf die Lösung zu schließen. Daher tendieren Kunden bei der Formulierung der Bedarfe dazu, sowohl abstrakte als auch sehr konkrete Ziele bzw. Merkmale zu nennen. LINDEMANN²⁹⁶ führt dies auch darauf zurück, dass viele für den Zielbildungsprozess relevante Informationen nur implizit vorliegen und es daher oftmals zu einer vermischten Darlegung der Ziele und möglicher Umsetzungen kommt. LINDEMANN schlägt daher eine systematische Befragung von Kunden und Mitarbeitern mittels Fragetechniken vor, um das implizite

²⁹³ vgl. Haberfellner et al. 2012, S. 219

²⁹⁴ vgl. Eiletz 1999, S. 12

²⁹⁵ vgl. Darlington & Culley 2002

²⁹⁶ vgl. Lindemann 2009, S. 96

Wissen herauszuarbeiten. EHRENSPIEL²⁹⁷ betont in diesem Zusammenhang die generelle Wichtigkeit einer intensiven Kommunikation mit dem Endkunden bzw. Auftraggeber. Zur Dokumentation (erster) Ziele und Anforderungen schlägt ALBERS²⁹⁸ eine grafische Aufbereitung in Form von Mind-Maps vor, um so die vorhandenen Zusammenhänge, Querverweise und Abhängigkeiten einfach abbilden zu können. Neben Kunden und Mitarbeitern stellen explizit vorliegende Informationen eine weitere wesentliche Quelle zur Identifikation von Zielen und Anforderungen dar (vgl. Abbildung 2-37).



Abbildung 2-37: Quellen für Ziele und Anforderungen²⁹⁹

In der industriellen Praxis, insbesondere im deutschsprachigen Raum, werden Ziele und Anforderungen zumeist in Form von Lasten- und Pflichtenheften dokumentiert³⁰⁰. Wesentliches Element zur Strukturierung der Ziele und Anforderungen sowohl im Lasten- als auch im Pflichtenheft stellen die **Anforderungslisten** dar³⁰¹. Das **Lastenheft**³⁰² stellt dabei eine Zusammenstellung aller Anforderungen des Auftraggebers hinsichtlich Liefer- und Leistungsumfang dar und definiert damit das *Was* und *Wofür* der Entwicklung. Das **Pflichtenheft**³⁰² enthält das Lastenheft und erweitert dieses um Realisierungsanforderungen. Das Pflichtenheft definiert damit das *Wie* und *Womit* die Anforderungen des Auftraggebers zu realisieren sind. Das

²⁹⁷ vgl. Ehrlenspiel 2009, S. 375

²⁹⁸ vgl. Albers, Klingler & Ebel 2013

²⁹⁹ vgl. Lindemann 2009, S. 97

³⁰⁰ vgl. Pohl 2007, S. 231

³⁰¹ vgl. Ponn & Lindemann 2011, S. 40

³⁰² vgl. VDI 2519 2001

Lastenheft wird vom Auftraggeber erstellt und dient als Ausschreibungs-, Angebots- und Vertragsgrundlage. Das Pflichtenheft wird in der Regel nach Auftragserteilung vom Auftragnehmer erstellt, falls erforderlich unter Mitwirkung des Auftraggebers. Das Pflichtenheft bedarf der Genehmigung durch den Auftraggeber, das nach erfolgter Genehmigung die verbindliche Vereinbarung für die Realisierung und Abwicklung des Projektes für Auftraggeber und Auftragnehmer wird. Die Aufteilung der Ziele und Anforderungen in ein Lasten- und Pflichtenheft birgt jedoch die Gefahr, dass zum einen die Verbindung zwischen Zielen, und Anforderungen verloren geht, da diese auf unterschiedliche Dokumente verteilt sind. Zum anderen birgt es die Gefahr, dass das Lasten- und Pflichtenheft von unterschiedlichen Personen, aus unterschiedlichen Abteilungen erstellt wird. Hieraus können erhebliche (Cross-Stage) Iterationen entstehen, da sich als realisierbar eingeschätzte Kundenwünsche als nicht (voll) umsetzbar herausstellen können oder wichtige (personengebundene) Informationen verloren gegangen sind³⁰³.

2.5.2 Abbildung von Zielsystemen

Seit geraumer Zeit werden Ansätze zur Abbildung von Zielsystemen erforscht, die den Anspruch haben, deutlich über die statische und wenig vernetzte Dokumentation von Anforderungen in rein textueller Form hinaus zu gehen. DARLINGTON UND CULLEY³⁰⁴ bringen den Bedarf auf den Punkt: „Accordingly, it is no longer possible to treat the specification as a fixed prescriptive document, it must become an open medium, capable of transmitting the functional and performance requirements and necessary technical adjustments.“ EBERT³⁰⁵ gibt einen Überblick über in der Praxis verbreitete Ansätze und Werkzeuge zur Abbildung von Anforderungen. (vgl. Abbildung 2-38). Hierbei sieht er DOORS von IBM als Markt- und Technologieführer im Bereich des Anforderungsmanagements an³⁰⁶, das sich insbesondere für große Projekte eignet.

Viele aktuelle Ansätze aus der Forschung zur Abbildung von Zielsystemen nutzen als Basis die Systems Modeling Language (SysML)³⁰⁷. Im Gegensatz zu Ansätzen, die dem Stand der Technik zuzuordnen sind und die entsprechend bereits eine Verbreitung in der Praxis gefunden haben, verfolgen diese Ansätze die modellbasierte Abbildung des zu entwickelnden Systems und der hierfür erforderlichen Partialmodelle. Mit Hilfe von SysML lassen sich die Struktur, das Verhalten und die

³⁰³ vgl. Braun, Ebel & Albers 2013

³⁰⁴ vgl. Darlington & Culley 2002

³⁰⁵ vgl. Ebert 2012, S. 334

³⁰⁶ vgl. Ebert 2012, S. 336

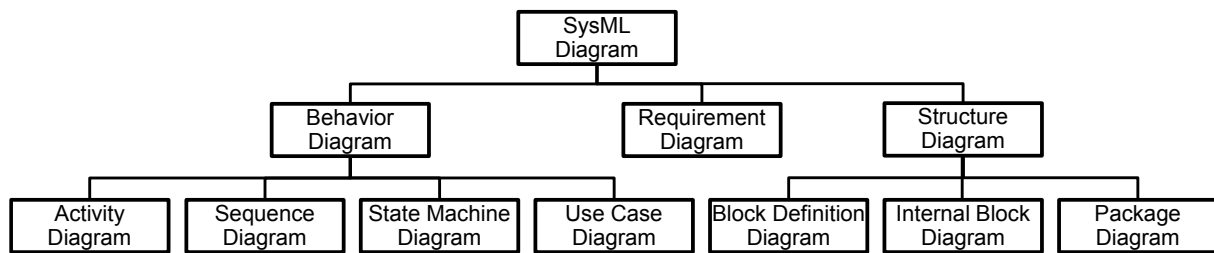
³⁰⁷ vgl. z.B. Frank 2006, Stechert 2010, Zingel 2013

Anforderungen eines Systems auf formale Art und Weise beschreiben und miteinander in Beziehung setzen³⁰⁸. Die SysML bietet hierfür verschiedene Diagrammtypen, auf deren Basis eine durchgängige Modellierung des Systems mittels grafischer Elemente erfolgen kann (vgl. Abbildung 2-39).

| Werkzeug | Hersteller | Schlüsselfunktionen | Kosten |
|-----------------------|---------------|--|---------|
| DOORS | IBM | Dokumentation, große Projekte, Produktlinien, Nachverfolgbarkeit, XML-Unterstützung, ReqIF, Anbindung an UML-Tools, Testmanagement-Tools, PLM-Tools, MS Projekts | Hoch |
| Enterprise Architect | Sparx Systems | Dokumentation, UML-basierte Analyse und Modellierung, Nachverfolgbarkeit, Teststatus, XML-Unterstützung | Mittel |
| IRqA | Visure | Dokumentation, OO-Analyse und ERM, Nachverfolgbarkeit, Teststatus, XML-Unterstützung, ReqIF, MS-Office-Anbindung | Mittel |
| MKS Integrity | PTC | ALM-Umgebung mit Komponenten für RE, Dokumentation, Analyse, Nachverfolgbarkeit, Konsistenzsicherung, XML-Unterstützung, ReqIF, MS-Office-Anbindung | Hoch |
| OSRMT | Open-Source | Einstiegswerkzeug, Dokumentation, Projektmanagement, Konfigurationsmanagement, Modellierung | Niedrig |
| Polarion Requirements | Polarion | Web-2.0-Umgebung, internetbasiertes RE, Anbindung an ALM-Umgebung, Workflow-Management | Mittel |
| PREEvision | Vector | PLM-/ALM-Umgebung mit Data-Backbone zur werkzeugübergreifenden Datenkonsistenz, ReqIF-Import/Export, Change-, Projekt, Testdatenmanagement | Hoch |
| Reqtify | Geensys | Dokumentation, Nachverfolgbarkeit, MS-Office-Anbindung, Simulink-Anbindung | Mittel |
| Requirements Composer | IBM | Zentralisierte Plattform für die Erhebung, Analyse, Validierung und Verwaltung von Anforderungen, gemeinsame Arbeit an den Anforderungen im Team, Einbeziehung von Kunden und Fachabteilungen, nahtlose Integration in Lifecycle | Hoch |
| RMF | various | Eclipse-basiert, Einstiegswerkzeug, ReqIF-Datenaustausch | Niedrig |
| RMTrak | RBC | Einstiegswerkzeug, Dokumentation, MS-Office, SQL Interface | Niedrig |
| Truereq PLM | Truereq | Einstiegswerkzeug, Dokumentation, XML-Unterstützung, kostenlose Einzelplatzlizenz | Niedrig |

Abbildung 2-38: Werkzeuge des Requirements Engineering³⁰⁵

³⁰⁸ vgl. Alt 2012, S. 30

Abbildung 2-39: Diagrammtypen der SysML³⁰⁹

FRANK³¹⁰ und STECHERT³¹¹ nutzen die SysML, um die von ihnen definierten Partialmodelle zu modellieren. ALBERS UND ZINGEL³¹² nutzen und erweitern die SysML, um eine funktionsbasierte Modellierung mechatronischer Systeme realisieren zu können, die auf dem Contact & Channel Ansatz aufbaut³¹³. GILZ UND EIGNER³¹⁴ zeigen auf, wie SysML-Modelle in PLM integriert werden können, um deren Aktualität und Verfügbarkeit gewährleisten zu können. IWANEK ET AL.³¹⁵ stellen fest, dass SysML mittlerweile in der Industrie als Modellierungssprache weit verbreitet ist. Gleichzeitig stellen sie fest, dass die Akzeptanz der Modellierungssprache bei Softwareentwicklern nach wie vor höher zu sein scheint, als bei Ingenieuren. Sie führen dies zum einen darauf zurück, dass die SysML auf der UML (Unified Modeling Language) basiert und daher Konzepte aus der Software-Entwicklung in den Kontext der Systemmodellierung überträgt. Dabei erfordert insbesondere die Objektorientierung grundlegende Softwarekenntnisse und setzt ein Umdenken bei Entwicklungsingenieuren voraus. Zum anderen fokussieren unterstützende Methoden vor allem auf die Entwicklung komplexer und softwareintensiver Systeme. IWANEK ET AL. schlagen daher eine fachdisziplinübergreifende Spezifikationstechnik (conceptual design specification technique for the engineering of mechatronic systems – CONSENS) vor, deren Ursprung in der Konstruktionslehre liegt und somit Vorgehensweisen und Begrifflichkeiten von Ingenieuren enthält. Die definierten Sprachelemente und Modellierungsmethoden sind hierbei eng aufeinander abgestimmt und können mittels des intrinsischen Erweiterungsmechanismus der SysML in ein entsprechendes Werkzeug (beispielsweise Sparx Enterprise Architect) integriert werden. Weiterhin ungelöst bleibt jedoch, dass eine intuitive Systembeschreibung aufgrund der Konzepte und Bezeichnungen der SysML selbst, nicht ohne weiteres möglich ist.

³⁰⁹ vgl. Zingel 2013, S. 56

³¹⁰ vgl. Frank 2006

³¹¹ vgl. Stechert 2012

³¹² vgl. Albers & Zingel 2013

³¹³ vgl. Albers & Sadowski 2014

³¹⁴ vgl. Gilz & Eigner 2013

³¹⁵ vgl. Iwanek et al. 2013

Trotz dieser Ansätze bleibt Microsoft Office weiterhin eines der meist genutzten Werkzeuge zur Abbildung von Zielsystemen³¹⁶. Dies kann auf die hohe subjektive Effizienz dieses Werkzeugs und der damit verbundenen bzw. umsetzbaren Methoden zurückgeführt werden (vgl. Abbildung 2-40).

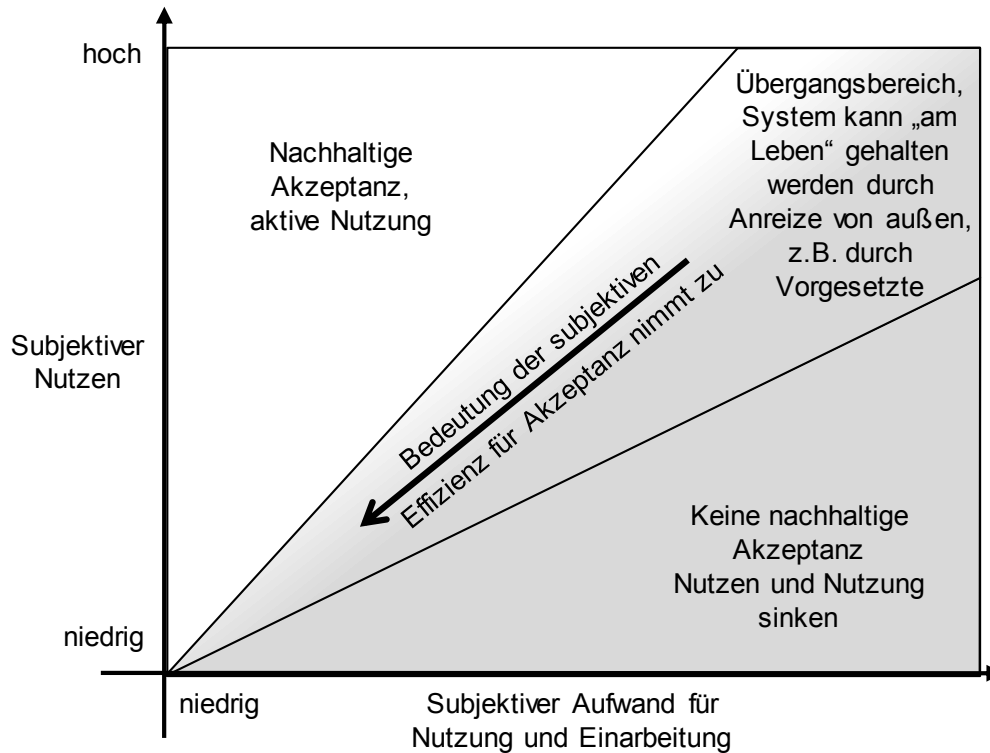


Abbildung 2-40: Subjektive Effizienz von Methoden³¹⁷

SAUTER³¹⁷ definiert die subjektive Effizienz als das Verhältnis aus subjektivem Nutzen zu subjektivem (Mehr-)Aufwand, die ein Anwender bei der Verwendung eines Systems empfindet. SAUTER kommt zum Schluss, dass eine nachhaltige Akzeptanz und aktive Nutzung von Methoden und Werkzeugen des Wissensmanagements nur erreicht werden kann, wenn der individuelle Anwender einen Mehrwert bei der Nutzung empfindet. Hierbei ist der Aufwand zur Einarbeitung und Nutzung von Microsoft Office – insbesondere Outlook, Word, Excel und Powerpoint – in der Regel sehr gering und der realisierbare Nutzen in der Regel direkt sichtbar.

BADKE-SCHAUB³¹⁸ untersucht die Defizite von Entwicklungsmethoden, die einen effektiven Transfer der Methoden in die Ingenieurspraxis verhindern und identifiziert drei Hauptkategorien von Defiziten. „*Performance*“ bezieht sich auf die Leistungsfähigkeit einer Methode, die entweder nicht ausreichend ist oder aber vom Nutzer nur

³¹⁶ vgl. Hood et al. 2007

³¹⁷ vgl. Sauter 2012, S. 197

³¹⁸ vgl. Badke-Schaub, Daalhuizen & Roozenburg 2011

schwer eingeschätzt werden kann. *Presentation* umfasst die Darstellung und Handhabbarkeit einer Methode, die z.B. spezifisches Vorwissen oder auch bestimmte Kompetenzen erfordern, über die der Entwickler (noch) nicht verfügt. *Process* beschreibt die Einbindung der Methodenanwendung in das reale Entwicklungsumfeld. Dies kann beispielsweise deren flexible Anpassung an die vorliegende Situation beinhalten.³¹⁹ ALBERS³¹⁹ erweitert diese Kategorien um die Dimension der individuellen und organisatorischen Akzeptanz und definiert damit wichtige Anforderungen an Methoden und Werkzeuge zur Abbildung von Zielsystemen (vgl. Abbildung 2-41).

| | Performance | Presentation | Process |
|------------------------|----------------------------------|---------------------|--------------------------------|
| individuell | wahrgenommene Leistungsfähigkeit | intuitiv handhabbar | flexibel & anpassbar |
| organisatorisch | Aufwand-Nutzen-Verhältnis | lehr- & lernbar | wiederverwendbar & erweiterbar |

Abbildung 2-41: Beurteilung der Akzeptanz von Methoden und Werkzeugen³¹⁹

2.5.3 Zwischenfazit

Das Zielsystem der Produktentstehung enthält relevante Ziele, deren Wechselwirkungen, Randbedingungen und Begründungen. Ein vollständiges Zielsystem enthält alle Ziele, die für die richtige Entwicklung des richtigen Produkts erforderlich sind. Kernbestandteil der Entwicklungstätigkeit ist daher die Vervollständigung eines anfangs vagen Zielsystems bis zu einem Grad, der eine erfolgreiche Produktentstehung ermöglicht. Neben der Entwicklung der einzelnen Elemente des Zielsystems ist hierbei insbesondere die Entwicklung der Relationen zwischen den Elementen von Bedeutung. Erst hierdurch kann die erforderliche Durchgängigkeit im Zielsystem realisiert und ein umfassendes Systemverständnis aufgebaut werden. Zur Gewährleistung der Nachvollziehbarkeit aller Elemente des Zielsystems müssen darüber hinaus aus solche Aspekte berücksichtigt werden, die zwar mit dem Zielsystem in Beziehung stehen, diesem aber nicht primär zugeordnet sind. Neben Zielen und Anforderungen als wesentliche Elemente des Zielsystems müssen daher weitere Partialmodelle abgebildet und mit dem Zielsystem vernetzt werden. Hierbei

³¹⁹ vgl. Albers, Lohmeyer & Radimersky 2012

besteht die zentrale Herausforderung zum einen darin, geeignete Partialmodelle und Relationen zu identifizieren, die eine durchgängige und nachvollziehbare Modellierung eines sich kontinuierlich verändernden Zielsystems ermöglichen. Zum anderen werden geeignete Modellierungsmethoden und Modellierungswerkzeuge benötigt, die eine individuelle und organisatorische Akzeptanz genießen und damit eine nachhaltige und aktive Nutzung erfahren können.

3 Motivation und Zielsetzung der Arbeit

3.1 Motivation

Heutige Produkte müssen einer Vielzahl an Zielen, Anforderungen und Randbedingungen genügen, um am Markt erfolgreich sein zu können. Die zunehmende Komplexität der Produkte führt hierbei zu dem, dass die Anzahl und Arten von Zielen, Anforderungen und Randbedingungen und deren Beziehungen untereinander stetig zunimmt. Zum anderen führt eine Steigerung der Produktkomplexität zwangsläufig zu einer gesteigerten Prozesskomplexität, da eine Vielzahl von Akteuren unterschiedlicher Disziplinen und eine Vielzahl vernetzter Aktivitäten erforderlich sind, um in der heutigen Wettbewerbssituation erfolgreiche Produkte entwickeln zu können. Hierdurch ist die Planung und Steuerung heutiger Produktentstehungsprozesse erheblich anspruchsvoller geworden – die vorherrschende Unsicherheit im Prozess der Produktentstehung muss durch eine entsprechende Prozessgestaltung kompensiert werden. Hierbei kommt der kontinuierlichen Verifikation und Validierung des Produkts und seiner Teilsysteme sowie dem erfolgreichen Umgang mit Iterationen im Produktentstehungsprozess eine besondere Bedeutung zu.

Eine der wesentlichen Herausforderungen besteht hierbei darin, die richtigen Ziele, Anforderungen und Randbedingungen zu erfassen, deren Wechselwirkungen untereinander zu erkennen und die richtigen Schlüsse für die Entwicklung des Produkts zu ziehen – und dies im gesamten Prozess der Produktentstehung. Hierfür ist es erforderlich, dass die Summe der Ziele, Anforderungen und Randbedingungen von allen Akteuren als Zielsystem aufgefasst wird, das gemeinsam und kontinuierlich entwickelt werden muss. Die richtigen Elemente des Zielsystems zu erfassen, erfordert heutzutage mehr denn je das synergetische Zusammenwirken verschiedener Akteure unterschiedlichster Disziplinen, da nur so der Komplexität heutiger Produkte und entsprechender Zielsysteme Rechnung getragen werden kann. Hierbei ist jedoch oftmals unklar, welche Faktoren ein erfolgreiches Zusammenwirken der Akteure unterschiedlicher Disziplinen begünstigen bzw. erschweren. Weiterhin werden die Akteure bei der eigentlichen Modellbildung des Zielsystems nur unzureichend unterstützt. Häufig werden die verschiedenen Zwecke, die mit einer Zielsystemmodellierung verfolgt werden, nicht vollständig überblickt. Darüber hinaus wird die Einschätzung vorhandener bzw. potenzieller Elemente des Zielsystems hinsichtlich deren Relevanz für einzelne Akteure bzw. den Fortgang des Produktentstehungsprozesses nur unzureichend unterstützt. Hierdurch wird es den Akteuren erheblich erschwert, einen geeigneten Verkürzungsgrad bei der Explikation und

Dokumentation zu wählen. Dies kann sich negativ auf die Effizienz der Explikation und Dokumentation sowie auf die Effektivität bzw. Qualität des resultierenden Zielsystems auswirken. Um ein umfassendes Zielsystemverständnis realisieren zu können, müssen neben den eigentlichen Elementen des Zielsystems auch deren Relationen untereinander und insbesondere zu angrenzenden Partialmodellen berücksichtigt und abgebildet werden, da andernfalls die erforderliche Durchgängigkeit im Zielsystem nicht realisiert werden kann. Hierbei sind die erforderlichen Relationen und Partialmodelle nur teilweise bekannt und eine pragmatische Anwendung im Prozess der Produktentstehung nicht ausreichend geklärt, so dass es den betroffenen Akteuren bzw. Entwicklern kaum möglich ist, mit hoher Effizienz ein nutzenstiftendes Zielsystem, das der Komplexität heutiger Produkte und Produktentstehungsprozesse gerecht wird, zu modellieren. Einfache Ansätze, die beispielsweise auf Microsoft Office aufsetzen, sind kaum mehr in der Lage, die erforderliche Komplexität heutiger Zielsysteme sinnvoll abzubilden. Komplexe Ansätze wie beispielsweise SysML fügen (noch) eine erhebliche Eigenkomplexität zur eigentlichen Komplexität des abzubildenden Zielsystems hinzu, so dass deren individuelle und organisatorische Akzeptanz nur schwer realisiert werden kann.

3.2 Zielsetzung

Ziel dieser Arbeit ist die Entwicklung einer schlanken Methode, die es dem Entwickler mechatronischer Produkte erlaubt, effizient, selbstständig und im gesamten Verlauf des Produktentstehungsprozesses ein nutzenstiftendes Zielsystem zu modellieren.

Um der Komplexität heutiger Produkte und entsprechender Prozesse gerecht zu werden, soll die Methode auf der Systemtheorie basieren und Aspekte der Modelltheorie berücksichtigen. Hierbei soll der einzelne Entwickler insbesondere bei der Frage einer adäquaten Verkürzung unterstützt werden, wodurch die Effizienz der Modellbildung wesentlich erhöht werden soll. Hierzu sollen auch übergeordnete Zwecke der Zielsystemmodellierung identifiziert werden, da der Modellierungszweck einen erheblichen Einfluss auf die mögliche Modellverkürzung hat. Weiterhin soll die Methode selbst auf möglichst wenigen und einfachen Regeln basieren, die in der Modellbildung dynamisch angewendet werden können. Hierdurch soll die Eigenkomplexität der Methode minimiert und die individuelle Akzeptanz in der Anwendung der Methode erhöht werden.

Um in einem Unternehmen Anwendung finden zu können, muss die Methode anschlussfähig an bestehende Formen der Zielsystemmodellierung sein, da in der Regel ein nicht unerheblicher Anteil des Unternehmenswissens in entsprechenden Referenz-Zielsystemen abgebildet ist. Hierbei müssen sich insbesondere Strukturen und Inhalte bestehender Lasten- und Pflichtenhefte integrieren lassen, da diese

textuelle Repräsentation bestehender Zielsysteme sehr weit verbreitet ist. Die Methode muss jedoch in der Lage sein, die Elemente des Zielsystems nicht nur isoliert und statisch abzubilden. Vielmehr muss die Methode die Vernetzung der Elemente untereinander sowie deren Entwicklung im Prozess der Produktentstehung abbilden können. Hierbei muss die Methode der Dynamik heutiger Produktentstehungsprozesse folgen können und es erlauben, auftretende Iterationen sowie resultierende Änderungen effizient abzubilden. Weiterhin darf die Methode ein aktivitätenbasiertes Vorgehen der Entwickler innerhalb einzelner Phasen des Produktentstehungsprozesses nicht behindern – idealerweise begünstigt die Methode ein solches Vorgehen sogar. Dies bedeutet, dass dem Entwickler eine Struktur innerhalb des Zielsystems zur Verfügung gestellt werden muss, die ihm einerseits erlaubt, alle Zielsystem-relevanten Informationen abzulegen bzw. wiederzufinden und die ihm andererseits potenzielle Lücken im Zielsystem aufzeigt, wodurch entsprechende Aktivitäten ausgelöst und durchgeführt werden können. Durch die Struktur des Zielsystems muss darüber hinaus sichergestellt werden, dass eine Nachvollziehbarkeit zwischen allen abgelegten Informationen möglich ist, da andernfalls die erforderliche Durchgängigkeit im Zielsystem nicht erreicht werden kann. Hierzu müssen geeignete Partialmodelle, Elemente innerhalb der Partialmodelle und entsprechende Relationen zwischen den Elementen identifiziert und durch die Methode bereitgestellt werden. Hierdurch soll der Entwickler in die Lage versetzt werden, relevante Ziele, deren Wechselwirkungen, Randbedingungen und Begründungen in einem Modell abbilden zu können. Der hierfür erforderliche Mehraufwand in der Modellierung im Vergleich zu konventionellen Ansätzen, die auf Lasten- und Pflichtenheften aufbauen, muss hierbei durch einen entsprechenden Mehrwert, der über die reine Dokumentation des Zielsystems hinausgeht, kompensiert werden. Daher muss es durch das explizit vorliegende Modell des Zielsystems möglich sein, Schlüsse zu ziehen, die für einen Entwickler nicht auf der Hand liegen – andernfalls kann die individuelle und organisatorische Akzeptanz der Methode nur schwer erreicht werden.

Um die entwickelte Methode umsetzen zu können, muss dem Entwickler mechatronischer Produkte ein entsprechendes Werkzeug zur Verfügung gestellt werden, das ebenfalls den Anspruch der individuellen Akzeptanz erfüllt. Daher soll die Methode mittels eines möglichst intuitiv handhabbaren, flexiblen und anpassbaren Werkzeugs umgesetzt werden. Die Wechselwirkungen, die zwischen Methode und umsetzendem Werkzeug zu erwarten sind, sollen hierbei erfasst und qualitativ bewertet werden. Hieraus sollen relevante Werkzeugeigenschaften identifiziert werden, die einen positiven Einfluss auf die Modellierung von Zielsystemen haben. Auf dieser Basis soll ein Werkzeug abgeleitet werden, das möglichst optimal auf die zu entwickelnde Methode abgestimmt ist.

4 Forschungsdesign

4.1 Forschungsfragen

Auf Basis des aufgezeigten Standes der Forschung, der hieraus abgeleiteten Motivation und der formulierten Zielsetzung dieser Arbeit, werden drei zentrale Forschungsfragen aufgestellt, die zur Erreichung der Zielsetzung beantwortet werden müssen. Die Forschungsfragen bauen prinzipiell aufeinander auf und ihre gemeinsame Beantwortung stellt den wesentlichen wissenschaftlichen Beitrag dieser Arbeit dar. Zum Zwecke der Detaillierung und Operationalisierung sind die drei zentralen Forschungsfragen nochmals in Teilfragen untergliedert. Die Beantwortung der Teilfragen ermöglicht dabei die Beantwortung der jeweils übergeordneten zentralen Forschungsfrage.

Forschungsfrage 1: Verständnis von Zielsystemen

Wie kann das Verständnis für Ziele und Zielsysteme in der interdisziplinären Produktentstehung gesteigert werden?

- Was sind Erfolgsfaktoren der interdisziplinären Zusammenarbeit?
- Welche übergeordneten Zwecke von Zielen existieren?
- Welche generischen Kriterien erleichtern die Beurteilung von Zielen?

Die erste Forschungsfrage zielt darauf ab, den einzelnen Entwickler mechatronischer Produkte beim Umgang mit Zielen und Zielsystemen im Kontext interdisziplinärer Produktentstehungsprozessen zu unterstützen. Durch ein gesteigertes Verständnis von Zielen und Zielsystemen soll eine selbstständige, effiziente und nutzenstiftende Modellierung von Zielen und Zielsystemen ermöglicht bzw. erleichtert werden.

Forschungsfrage 2: Struktur von Zielsystemen

Welche Struktur eines Zielsystemmodells ist in der Lage, alle Zielsystem-relevanten Informationen abzubilden?

- Welche Partialmodelle müssen gebildet werden?
- Welche Elemente müssen vorhanden sein?
- Welche Relationen zwischen den Elementen müssen vorhanden sein?
- Welche Attribute sind zur Charakterisierung der Elemente nützlich?

Die zweite Forschungsfrage hat zum Ziel, eine geeignete Zielsystem-Referenzstruktur zu identifizieren, die es dem Entwickler mechatronischer Produkte prinzipiell erlaubt, sämtliche Zielsystem-relevanten Informationen zu modellieren und

das resultierende Zielsystemmodell zur Extraktion entscheidungsrelevanter Informationen zu nutzen.

Forschungsfrage 3: Modellierung von Zielsystemen

Wie können Zielsysteme im Prozess einer interdisziplinären Produktentstehung effizient modelliert und genutzt werden?

- Wie kann der Entwickler bei der Modellierung des Zielsystems unterstützt werden?
- Wie können entscheidungsrelevante Informationen aus dem modellierten Zielsystem extrahiert werden?
- Welche Werkzeugeigenschaften haben einen positiven Einfluss auf die Modellierung von Zielsystemen?

Die dritte Forschungsfrage zielt auf die Anwendung der entwickelten Strukturierung von Zielsystemen im Prozess der interdisziplinären Produktentstehung ab. Hierbei gilt es zu klären, wie die Strukturierung durch die Entwickler mechatronischer Produkte angewandt werden soll und wie der einzelne Entwickler hierdurch bei der Entwicklung mechatronischer Produkte entscheidungsrelevante Informationen aus dem Zielsystemmodell extrahieren kann.

4.2 Forschungsvorgehen

Zur Beantwortung der Forschungsfragen und damit zur Erreichung der Zielsetzung dieser Arbeit wird ein Forschungsvorgehen bestehend aus drei Phasen umgesetzt, das in seinen Grundzügen dem Vorgehen der *Design Research Methodology* nach BLESSING UND CHAKRABARTI³²⁰ entspricht (vgl. Abbildung 4-1).

In der ersten Phase wird ein Verständnis von Zielen und Zielsystemen in der interdisziplinären Produktentstehung aufgebaut, das als grundlegende Basis für die zu entwickelnde Methode dient (vgl. Kapitel 5). Im Rahmen eines Expertenworkshops mit Teilnehmern aus Wissenschaft und Industrie werden hierzu praxisrelevante Erfolgsfaktoren der interdisziplinären Zusammenarbeit ermittelt und in Bezug zur Modellierung von Zielsystemen gesetzt (vgl. Abschnitt 5.1). Hierauf aufbauend wird die eigentliche Modellbildung von Zielsystemen näher untersucht. Auf Basis des erweiterten ZHO-Modells und der hierin beschriebenen Analyse- und Syntheseaktivitäten werden die übergeordneten Zwecke von Zielen identifiziert (vgl. Abschnitt 5.2). Hieraus und durch eine Verdichtung des relevanten Standes der

³²⁰ vgl. Blessing & Chakrabarti 2009

Forschung werden zum Abschluss der ersten Phase generische Kriterien zur Beurteilung von Zielen gebildet (vgl. Abschnitt 5.3).

In der zweiten Phase wird die Modellierung von Zielsystemen in der industriellen Praxis untersucht (Kapitel 6). Anhand zweier Fallstudien, die im Kontext realer Produktentwicklungsprojekte durchgeführt werden, sollen wesentliche Erkenntnisse für die Strukturierung und Modellierung von Zielsystemen gewonnen werden. Hierbei soll ein wesentliches Augenmerk auf die Anwendbarkeit der Modellierungsmethode und des jeweils gewählten Werkzeuges durch die involvierten Entwickler gelegt werden. Daher erfolgt die Zielsystemmodellierung in beiden Fallstudien mit intuitiv handhabbaren Werkzeugen, die sich entsprechend den Erfordernissen der Fallstudien flexibel anpassen lassen. In der ersten Fallstudie (Abschnitt 6.1) wird eine mögliche Vorfizierung auf bestehende Partialmodelle sowie deren Elemente, Relationen und Attribute bewusst vermieden, um die tatsächlichen Bedarfe hinsichtlich Struktur und Modellierung von Zielsystemen identifizieren zu können. Es wird lediglich zwischen Elementen des Zielsystems und korrespondierenden Objekten (Elementen des Objektsystems) unterschieden. Als Werkzeug zur Modellierung wird ein Semantic MediaWiki eingesetzt. In die zweite Fallstudie (Abschnitt 6.2) fließen Erkenntnisse der ersten Fallstudie sowie der ersten Phase (Verständnis von Zielsystemen) der Arbeit ein. Zur Modellierung des Zielsystems werden Mind-Maps eingesetzt.



Abbildung 4-1: Forschungsvorgehen

In der dritten Phase werden die Erkenntnisse der beiden vorangegangenen Phasen genutzt, um eine Methode zur Modellierung von Zielsystemen in der interdisziplinären Produktentwicklung abzuleiten (Kapitel 7). Hierbei wird zuerst eine Möglichkeit der Strukturierung von Zielsystemen aufgezeigt, die neben erforderlichen Partialmodellen und deren Elementen auch nutzenstiftende Relationen und Attribute beinhaltet (Abschnitt 7.1). Anschließend wird die Nutzung bzw. Nutzbarkeit der Zielsystemstruktur mittels darauf abgestimmter Modellierungsregeln, die auf den wesentlichen Erkenntnissen der beiden Fallstudien basieren, abgesichert (Abschnitt 7.2). Auf Basis der gesammelten Erfahrungen wird ein intuitiv anwendbares und flexibel einsetzbares Werkzeug entwickelt (Abschnitt 7.3), das die Validierung der entwickelten Methode ermöglicht. Die abschließende Validierung der Methode zur Zielsystemmodellierung erfolgt anhand einer retrospektiven Betrachtung eines IPEK-internen Entwicklungsprojekts sowie der produktiven Anwendung der Methode in einem Entwicklungsprojekt eines Industrieunternehmens (Abschnitt 7.4).

5 Verständnis von Zielsystemen

In diesem Kapitel wird untersucht, wie das Verständnis von Zielen und Zielsystemen in der interdisziplinären Produktentstehung gesteigert werden kann. Hierdurch soll dem Entwickler mechatronischer Produkte der Umgang mit Zielen und Zielsystemen im soziotechnischen System der Produktentstehung erleichtert und damit einen Beitrag zu erfolgreichen Produktentstehungsprozessen geleistet werden.

5.1 Erfolgsfaktoren der Interdisziplinarität³²¹

Die erfolgreiche Entwicklung heutiger Produkte erfordert mehr denn je die erfolgreiche Zusammenarbeit diverser Akteure **unterschiedlicher Disziplinen**. Dies ist zum einen darin begründet, dass die Komplexität heutiger Produkte von einzelnen Akteuren nicht mehr alleine beherrscht werden kann, da dem einzelnen Menschen weder die hierfür erforderliche Vielzahl an Kompetenzen noch die kognitive Leistungsfähigkeit zur Verfügung steht. Zum anderen hängt der Erfolg heutiger Produkte nicht mehr nur davon ab, ob das Produkt den Markt- und Kundenbedürfnissen gerecht wird, sondern auch, ob das Produkt zur richtigen Zeit auf dem Markt verfügbar ist. Zur Sicherstellung einer ausreichend kurzen Produkteinführungszeit (time-to-market) sind Unternehmen zur Parallelisierung von Produktentstehungsaktivitäten gezwungen, wodurch sich zwangsläufig auch die erforderliche Kapazität und damit die Vielzahl der am Produktentstehungsprozess beteiligten Akteure erhöht.

In der heutigen Produktentstehung treffen folglich notwendigerweise eine Vielzahl unterschiedlicher Menschen mit unterschiedlichen Kompetenzen und unterschiedlichsten Denk- und Verhaltensmustern aufeinander. Trotz der teils offensichtlichen, teils jedoch auch verborgenen Unterschiede in diesen Denk- und Verhaltensmustern, ist die erfolgreiche Zusammenarbeit der Menschen keine Seltenheit. Die unterschiedlichen disziplinspezifischen Erfahrungen und die hieraus resultierenden Denk- und Verhaltensmuster sind oftmals sogar der notwendige Ursprung einer letztendlich erfolgreichen interdisziplinären Zusammenarbeit. Häufig führen jedoch gerade die verborgenen Diskrepanzen in diesen Mustern zu Problemen, die nicht selten auch den Ursprung für das Scheitern eines Produktes bilden. Da der **Mensch im Mittelpunkt der Produktentstehung** steht und von ihm jede Handlung innerhalb des Produktentstehungsprozesses ausgeht, ist es für ein gesteigertes Verständnis detaillierter Aspekte der Produktentstehung erforderlich, zunächst die Erfolgsfaktoren

³²¹ vgl. Albers, Ebel & Alink 2011

ren, aber auch die Barrieren, die eine interdisziplinäre Zusammenarbeit zwischen einzelnen Menschen begünstigen bzw. behindern, zu identifizieren.

5.1.1 Konzeption des Expertenworkshops

Um der Fragestellung nach **Erfolgsfaktoren und Barrieren interdisziplinärer Zusammenarbeit** in der Produktentstehung nachzugehen, werden die Meinungen und Sichten ausgewählter Vertreter aus Industrie und Wissenschaft im Rahmen eines **Expertenworkshops** zusammen gebracht. Die Gültigkeit und Qualität der aus dem Workshop resultierenden Erkenntnisse soll im Wesentlichen über die Zusammenstellung der Teilnehmer und deren ausgewiesene Erfahrung im Bereich der interdisziplinären Zusammenarbeit abgesichert werden. Um der Vielschichtigkeit der Fragestellung bestmöglich gerecht zu werden, wird auf eine angemessene Heterogenität der Teilnehmer geachtet. Seitens der Industrie nehmen Vertreter aus verschiedenen Branchen (Anlagenbau, Fahrzeugbau und Gerätebau) teil, die in ihren Unternehmen jeweils unterschiedliche Positionen einnehmen (Entwicklungsingenieur, Gruppenleiter, Abteilungsleiter). Seitens der Wissenschaft sind Vertreter der Ingenieurwissenschaften (Maschinenbau, Elektrotechnik, Bauingenieurwesen) sowie der Technikfolgenabschätzung und der Technikphilosophie geladen. Insgesamt nehmen 14 Wissenschaftler (hiervon acht Professoren, darunter Prof. Albert Albers, Prof. Udo Lindemann und Prof. Günter Ropohl) sowie sechs Vertreter aus der Industrie (hiervon drei in leitender Position) am Expertenworkshop teil.

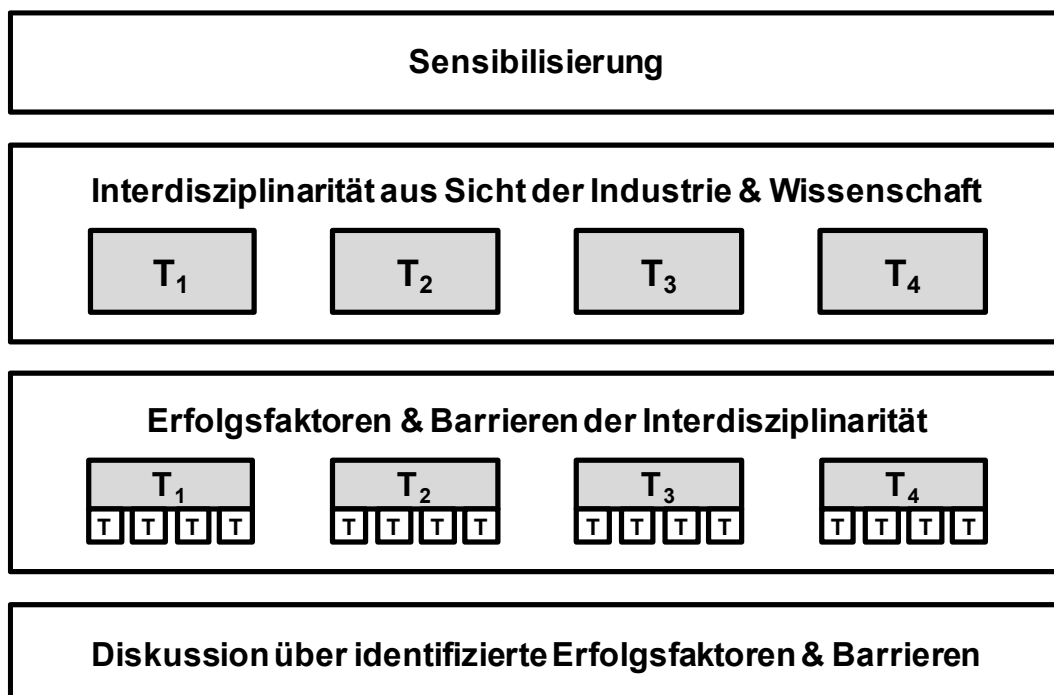


Abbildung 5-1: Konzeption des Expertenworkshops

Der eintägige Expertenworkshop zur Identifikation und Benennung von Erfolgsfaktoren und Barrieren ist in die vier Phasen *Sensibilisierung*, *Sichten*, *Erfolgsfaktoren &*

Barrieren und *Diskussion* untergliedert (vgl. Abbildung 5-1). In der ersten Phase (*Sensibilisierung*) sollen alle Teilnehmer des Expertenworkshops für die Probleme und Herausforderungen interdisziplinärer Zusammenarbeit in der Produktentstehung sensibilisiert werden. Hierzu werden im Rahmen eines **Impulsvortrags** die heutigen und zukünftigen Probleme und Herausforderungen aufgezeigt, die bei der Entwicklung humanoider Roboter und (teil-)autonomer Fahrzeuge zu bewältigen sind bzw. sein werden. Im Anschluss berichten vier Teilnehmer (jeweils zwei Vertreter aus Industrie und Wissenschaft) über Probleme und Herausforderungen interdisziplinärer Zusammenarbeit aus ihre jeweiligen Sicht (Phase *Sichten*). Diese Berichte und sechs Leitfragen dienen als wesentliche Ausgangsbasis zur Identifikation und Benennung von Erfolgsfaktoren und Barrieren interdisziplinärer Zusammenarbeit (Phase *Erfolgsfaktoren & Barrieren*). Die sechs **Leitfragen** sind:

- Welche Disziplinen sind bei der Entstehung zukünftiger Produkte erforderlich?
- Welche sind die typischen Stärken und Schwächen der beteiligten Disziplinen?
- Welche Probleme treten im Verlaufe eines Produktentstehungsprojekts vermehrt auf?
- Wie kann erreicht werden, dass alle beteiligten Disziplinen an einem Strang ziehen?
- Wie können die unterschiedlichen Modelle der beteiligten Disziplinen zusammen geführt werden?
- Wie sollte ein zukünftiger Produktentstehungsprozess organisiert sein?

Um ein möglichst breites Spektrum an Erfolgsfaktoren und Barrieren interdisziplinärer Zusammenarbeit zu erhalten, werden die 20 Teilnehmer in vier Gruppen zu je fünf Teilnehmern aufgeteilt. Hierbei ist jeder Gruppe genau ein Teilnehmer zugeordnet, der im Rahmen der Phase *Sichten* berichtet hat (vgl. Abbildung 5-1). Hierdurch soll es den Teilnehmern ermöglicht werden, die berichteten Probleme und Herausforderungen als Ausgangsbasis der Diskussion zu nutzen und gleichzeitig abstrakte Aspekte der Diskussion am berichteten Beispiel diskutieren zu können. Um die Teilnehmer durch die formulierten Leitfragen nicht unnötig einzuschränken und damit die Diskussion zu hemmen, ist es den Gruppen freigestellt, einzelne Fragen zu streichen, umzuformulieren oder neue Fragen hinzu zu fügen. Zur Strukturierung, Visualisierung und Dokumentation der Diskussion und Diskussionsergebnisse stehen den Gruppen Metaplan-Stellwände und -Karten zur Verfügung. Die letzte Phase (*Diskussion*) wird eingeleitet durch die Vorstellung der Gruppenarbeitsergebnisse, die ein jeweils zu Beginn der Gruppenarbeit definierter Teilnehmer präsentiert. Nach der Vorstellung aller Gruppenarbeitsergebnisse wird die **Diskussion** durch die Fragestellung nach relevanten Erfolgsfaktoren und Barrieren interdisziplinärer Zusammenarbeit eröffnet. Die Diskussion und damit der Expertenworkshop schließt

ab mit einer **Blitzlichtrunde**, in der jeder Teilnehmer nochmals die Möglichkeit hat, die aus seiner Sicht wichtigsten Erfolgsfaktoren und Barrieren zu benennen.

5.1.2 Ergebnisse des Expertenworkshops

Im Folgenden werden die Ergebnisse des Expertenworkshops aufgezeigt und interpretiert. Hierzu wird jeder einzelne Vortrag der Phase *Sichten* gebündelt mit den Ergebnissen der entsprechenden Gruppe vorgestellt, um Zusammenhänge mit der Phase *Erfolgsfaktoren & Barrieren* deutlicher erkennen zu können. Im Anschluss werden die Ergebnisse der einzelnen Gruppen zusammengefasst und im Kontext der Zielsetzung dieser Arbeit interpretiert.

Ergebnisse der Gruppe 1

Der Vortrag beleuchtet die Kooperation im Bauwesen als ein traditionell sehr interdisziplinäres Fachgebiet. Als Beispiel dient die Stadtplanung, da hierbei in jedem Fall Vertreter aus den Disziplinen der Ingenieurwissenschaften, der Kunst und der Soziologie zusammen arbeiten. Um die Zusammenarbeit der Disziplinen in diesem Kontext zu verbessern, wird empfohlen, die **Schnittstellen** zwischen den beteiligten Disziplinen bzw. Akteuren mittels einer durchgängigen Planung und Überwachung des Bauprozesses zu verringern. Darüber hinaus wird vorgeschlagen, das als relevant für eine erfolgreiche Zusammenarbeit eingeschätzte **Vertrauen** zwischen den einzelnen Akteuren zu stärken, was insbesondere durch eine erhöhte **Zuverlässigkeit** der Akteure und deren Aussagen erreicht werden kann.

Die Gruppe benennt ebenfalls das **Vertrauen** zwischen den Akteuren als eine wesentliche Grundvoraussetzung für erfolgreiche (interdisziplinäre) Zusammenarbeit. Nur auf Basis von Vertrauen ist es möglich, die zahlreichen Barrieren der Interdisziplinarität zu überwinden. Im Mittelpunkt der Betrachtung steht dabei der Mensch, dessen Vertrauen durch **Zuverlässigkeit** verdient werden muss. Damit das Vertrauen zwischen den Akteuren nicht aufgrund von Missverständnissen oder Fehlinterpretationen gefährdet wird, sollte eine klare **Transparenz der Verantwortlichkeiten** vorhanden sein. Um auf Basis von Vertrauen ein effektives und effizientes Arbeiten im interdisziplinären Kontext zu ermöglichen, muss ein **gemeinsamen Ziel- und Problemverständnisses** geschaffen werden, das es auch zu explizieren gilt. Hierfür ist ein **gemeinsames Verständnis** sowohl auf sprachlicher wie auch auf fachlicher Ebene erforderlich. Die hierfür erforderliche **Kommunikation** in einem Unternehmen – insbesondere die verbale und informelle – kann durch eine räumliche Nähe der Akteure sowie eine flache hierarchische Ordnung zwischen den Akteuren positiv beeinflusst werden. Auch die professionelle Mediation zwischen den Akteuren kann die Kommunikation zwischen den Akteuren verbessern.

Ergebnisse der Gruppe 2

Der zweite Vortrag fokussiert auf die Probleme und Herausforderungen, denen ein Entwicklungsingenieur im Alltag seiner Arbeit bei einem großen deutschen Maschinenbauunternehmen begegnet. Hierbei wird hervorgehoben, dass die Nutzbarmachung von personengebundenem Wissen eine wesentliche Herausforderung darstellt, deren Bewältigung eine zumindest teilweise **gemeinsame Sprache**, insbesondere jedoch das **Können und Wollen der beteiligten Akteure** zur Gestaltung einer kooperativen Zusammenarbeit, erfordert. Das Können scheint eng verknüpft zu sein mit der **Fähigkeit des Reflektierens**, beispielsweise der eigenen Disziplin, des eigenen Standpunkts und des eigenen Handelns. Das Wollen spiegelt sich in einem **übergreifenden Ziel- und Problemverständnis** wider.

Die Gruppe benennt als elementare Voraussetzung für eine erfolgreiche interdisziplinäre Kooperation eine **Unternehmenskultur**, die **Offenheit, Transparenz** und **Vertrauen** repräsentiert sowie beteiligte Akteure, die den Erfolg wollen. Darüber hinaus stuft die Gruppe ein **gemeinsames Ziel- und Problemverständnis** als äußerst relevant ein und hebt zwei Fähigkeiten hervor, die zur Identifikation von Problemen und der Definition von Zielen erforderlich sind. Zum einen müssen Akteure in der Lage sein, einen **Perspektivenwechsel** vollziehen zu können d.h. zwischen einer konkreten und einer abstrakten Perspektive bzw. zwischen Detail und Ganzem frei skalieren zu können. Zum anderen müssen Akteure fähig sein, auch komplexe Sachverhalte anderen Akteuren **verständlich machen** zu können.

Ergebnisse der Gruppe 3

Der dritte Vortrag beleuchtet die kooperative Zusammenarbeit anhand eines Beispiels aus den frühen 1980ern im Bereich der Militärforschung. Hierbei wird die Bedeutung einer klaren, realistischen und **abgestimmten Zielsetzung** betont, wodurch ein übergeordneter Handlungsrahmen für alle beteiligten Disziplinen entsteht. Darüber hinaus betont der Vortrag die Relevanz definierter Ansprechpartner und im erweiterten Sinne die **Transparenz der Verantwortlichkeiten**.

Die Gruppe benennt die **disziplinspezifischen Denkstrukturen** als eine der entscheidenden Barrieren einer interdisziplinären Kooperation. Im Rahmen der Produktentstehung kann jedoch insbesondere der Disziplin Maschinenbau die Rolle eines Vermittlers zwischen den Disziplinen zukommen, da Akteure der Disziplin Maschinenbau bereits durch ihre Ausbildung aber auch durch das übliche Umfeld der Produktentstehung zu einer disziplinenübergreifenden Denkweise gezwungen werden. So wird neben dem obligatorischen Denken in Gestalt auch das Denken in Funktionen und damit die Sicht auf ein **System als Ganzes** gefordert.

Ergebnisse der Gruppe 4

Der vierte Vortrag erläutert anhand der Entwicklung eines komplexen mechatronischen Produkts die Herausforderung der **Zielkonfliktbewältigung** als einen erfolgsentscheidenden Aspekt interdisziplinärer Produktentstehungsprojekte.

Der Vortrag verdeutlicht, dass eine isolierte Betrachtung der Zielerfüllung auf Disziplin- bzw. Sub-Systemebene, nicht zwangsläufig eine optimale Zielerfüllung des zu entwickelnden (Gesamt-)Systems zur Folge hat. Um eine **ganzheitliche Betrachtung der Zielerfüllung** zu ermöglichen, muss **Vertrauen** und **Transparenz** zwischen den Disziplinen bzw. Organisationseinheiten eines Unternehmens geschaffen werden.

Die Gruppe hob dabei die Bedeutung einer einheitlichen und durchgängigen **Objektivierung des Ziel- und Objektssystems** hervor. Diese Objektivierung könne dabei unter Zuhilfenahme mathematischer Zusammenhänge und Operatoren gelingen.

5.1.3 Zusammenfassung und Interpretation der Ergebnisse

Auf Basis der Vorträge, der Ergebnisse der Gruppenarbeit und der anschließenden Diskussion sowie der abschließenden Befragung per Blitzlichtrunde kann ein zusammenhängendes Bild der Erfolgsfaktoren und Barrieren der Interdisziplinarität entwickelt werden (vgl. Abbildung 5-2).

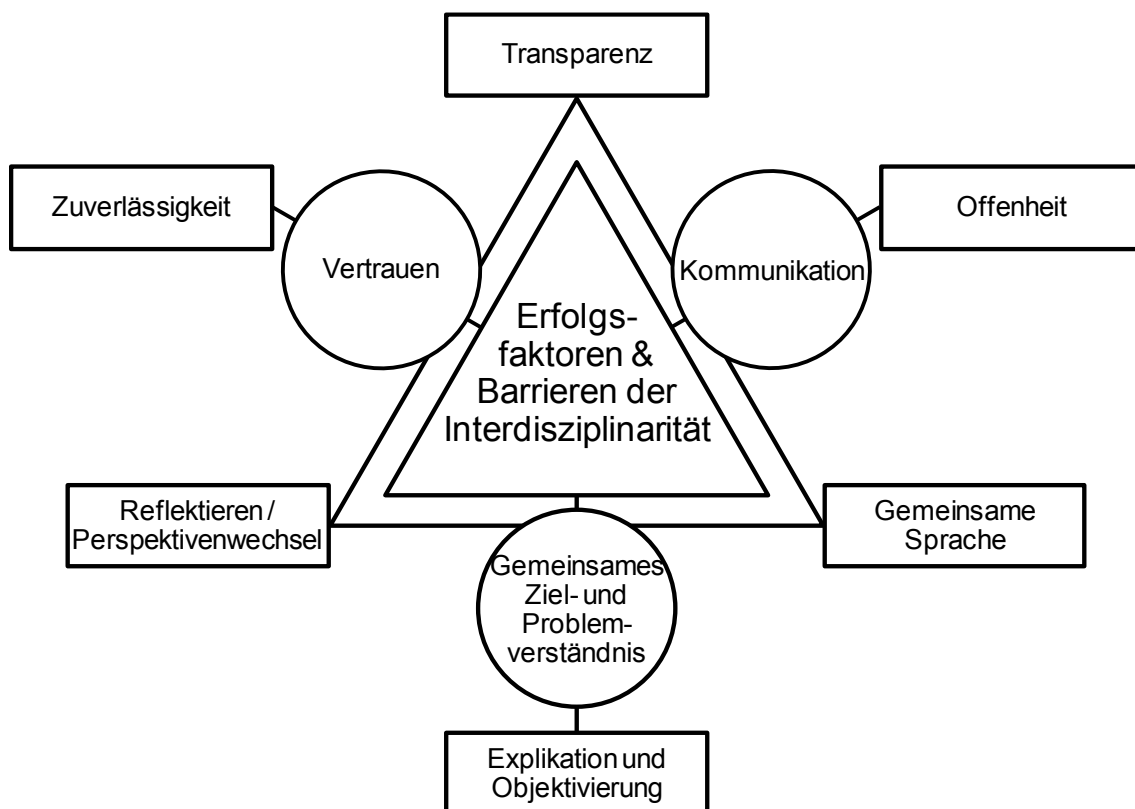


Abbildung 5-2: Erfolgsfaktoren und Barrieren der Interdisziplinarität

Dabei erhebt dieses Bild der erarbeiteten und extrahierten Erfolgsfaktoren und Barrieren der Interdisziplinarität nicht den Anspruch auf Vollständigkeit und universelle Gültigkeit. Die Relevanz der genannten Aspekte wird jedoch über die Zusammenstellung der Teilnehmer und die methodische Vorgehensweise im Rahmen des Expertenworkshops legitimiert. Aufgrund der vorgenommenen Fokussierung auf die Produktentstehung, ist von einer Gültigkeit zumindest für dieses Gebiet auszugehen.

In der Zusammenstellung der Erfolgsfaktoren und Barrieren der Interdisziplinarität (vgl. Abbildung 5-2) sollen sich die Relevanz einzelner Aspekte und deren Wechselwirkungen untereinander widerspiegeln. So sind Aspekte, die sich als besonders relevant heraus gestellt haben, in Kreisen an den Kanten des Dreiecks dargestellt.³²² Die Wechselwirkungen werden durch die Anordnung der Aspekte zueinander zum Ausdruck gebracht. So wird ein einzelner Aspekt von seinen unmittelbaren Nachbarn am stärksten beeinflusst. Hierdurch soll auch deutlich werden, dass alle aufgezeigten Aspekte mehr oder weniger stark miteinander in Wechselwirkung stehen und folgerichtig nicht der eine Erfolgsfaktor bzw. die eine Barriere der Interdisziplinarität existiert. Weiterhin wird deutlich, dass ein einzelner Aspekt nicht per se als Erfolgsfaktor oder Barriere eingestuft werden kann. Vielmehr ist es von der Ausprägung eines Aspekts abhängig, ob eher ein Erfolgsfaktor oder eine Barriere vorliegt. Auf Basis der gewählten Formulierung ist ein Aspekt als Erfolgsfaktor einzustufen, je stärker seine Ausprägung vorhanden ist. Im Umkehrschluss ist ein Aspekt eher eine Barriere interdisziplinärer Zusammenarbeit, je weniger stark er ausgeprägt bzw. vorhanden ist.

Im Kern zeigt sich, dass eine erfolgreiche interdisziplinäre Zusammenarbeit erheblich begünstigt wird durch Akteure denen es möglich ist, ein disziplinübergreifendes Systemverständnis des zu entwickelnden Produktes und der hierfür erforderlichen Prozesse aufzubauen. Hierbei haben drei übergeordnete Kategorien von Aspekten eine wesentliche Bedeutung:

- Unternehmenskultur und organisatorische Rahmenbedingungen
- Persönliche Eigenschaften und Fähigkeiten
- Methodische Unterstützung

Das Unternehmen als zentraler Ort interdisziplinärer Zusammenarbeit nimmt über die vorherrschende Unternehmenskultur und existierende organisatorische Rahmenbedingungen einen wesentlichen Einfluss auf die Erfolgsfaktoren und Barrieren der

³²² Der Aspekt *Gemeinsame Sprache* wird ebenfalls als Relevant eingestuft. Um die Konsistenz der Abbildung 5-2 zu wahren, wurde auf eine Hervorhebung der Relevanz dieses Aspekts verzichtet.

Interdisziplinarität. Dies betrifft insbesondere diejenigen Aspekte, welche in Abbildung 5-2 im oberen und mittleren Bereich angeordnet sind. Dabei hat die **Unternehmenskultur** einen grundlegenden Einfluss auf den Willen der Akteure, in kooperativer Art und Weise zusammen zu arbeiten und sich gegenseitig zu vertrauen. Über entsprechende **organisatorische Rahmenbedingungen**, die beispielsweise Auswirkungen auf die Transparenz und Kommunikation im Unternehmen haben, kann die Zusammenarbeit der einzelnen Akteure darüber hinaus positiv beeinflusst werden. Im Zentrum der Produktentstehung stehen jedoch die einzelnen Akteure, deren **persönliche Eigenschaften** und **Fähigkeiten** in besonderem Maße Erfolgsfaktoren, aber auch Barrieren interdisziplinärer Zusammenarbeit darstellen (Aspekte insbesondere im mittleren Bereich von Abbildung 5-2). So beeinflusst die Offenheit und Zuverlässigkeit der Akteure maßgeblich die erforderliche Kommunikation und das benötigte Vertrauen zwischen den einzelnen Akteuren bzw. innerhalb eines interdisziplinären Teams. Um ein gemeinsames Ziel- und Problemverständnis herstellen zu können, das maßgeblich über Erfolg oder Misserfolg eines Produktentstehungsprozesses entscheidet, ist ein ständiger **Perspektivenwechsel**, d.h. ein Denken in Systemen (Systemdenken) ebenso erforderlich, wie die **Reflektionsfähigkeit** der Akteure. Daneben müssen die Akteure in der Lage sein, zumindest teilweise eine **gemeinsame Sprache** zu sprechen. Daher müssen die einzelnen Akteure neben der Sprache ihrer eigenen Disziplin auch die Sprache anderer Disziplinen beherrschen. Um den Erfolg einer interdisziplinären Zusammenarbeit zu begünstigen, ist eine **methodische Unterstützung** der Akteure unverzichtbar. Hierbei zeigt sich, dass entsprechende Methoden insbesondere die Ausbildung eines **gemeinsamen Ziel- und Problemverständnisses** unterstützen können (Aspekte im unteren Bereich von Abbildung 5-2). So kann die Unterstützung der Reflektion und des Perspektivenwechsels dazu beitragen, dass einzelne Akteure relevante Zielsystemelemente und -beziehungen effektiver und effizienter identifizieren können. Weiterhin kann die Klassifizierung möglicher Zielsystemelemente und -beziehungen die Explikation und Objektivierung Zielsystem-relevanter Informationen unterstützen. Nicht zuletzt würde eine gemeinsame Sprache, insbesondere im Kontext der Zielsystemmodellierung dazu führen, dass die Kommunikation zwischen den Akteuren entscheidend verbessert wird.

5.2 Zwecke von Zielen³²³

Eine der zentralen Voraussetzungen zur korrekten Bildung und Nutzung eines Zielsystemmodells ist die **Bekanntheit des Modellierungszwecks**, da im Wesentlichen auf Basis des Zwecks der Modellierung auf den zulässigen Grad der Modellverkürzung geschlossen werden kann. Im Prozess der Modellbildung wird es hierdurch möglich, auf die Abbildung aller nicht-relevanten Aspekte des Originals bewusst zu verzichten und damit die Effizienz der Modellbildung erheblich zu verbessern, ohne dabei die Qualität des resultierenden Modells zu gefährden. Im Rahmen der Modellnutzung erlaubt die Bekanntheit des Modellierungszwecks eine Bewertung der vorgenommenen Modellverkürzungen und damit eine effektive Nutzung des Modells.

Das erweiterte ZHO-Modell (vgl. Abbildung 5-3) beschreibt die Produktentstehung als iterativen Prozess, in dem ein Ziel- und Objektsystem durch kontinuierliche Analyse- und Syntheseaktivitäten eines Handlungssystems co-evolutionär entwickelt werden.

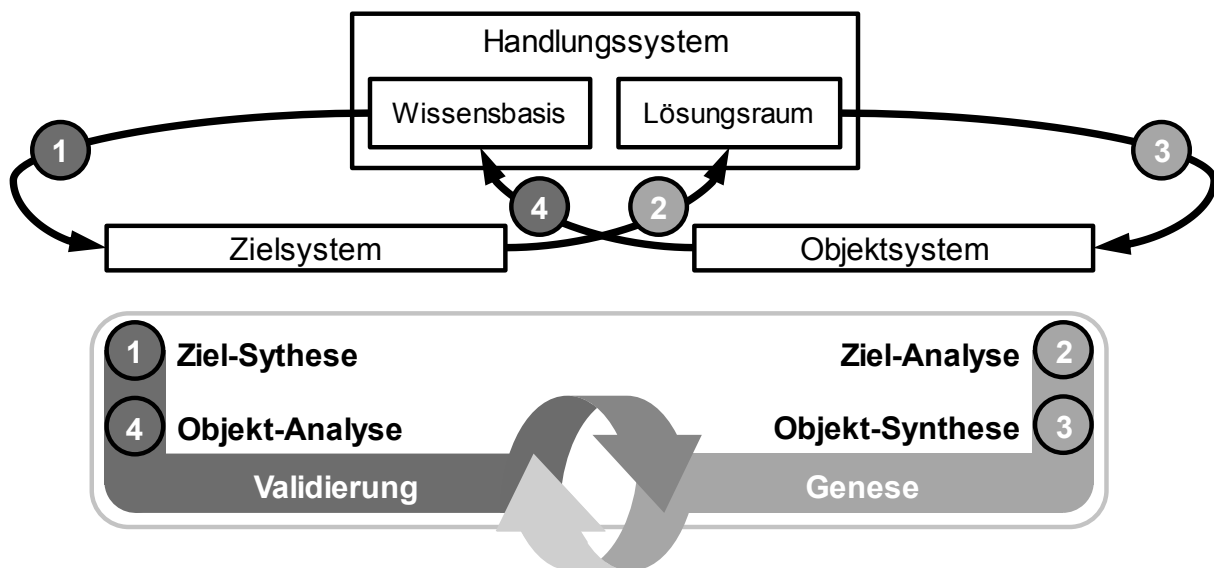


Abbildung 5-3: Generische Analyse- und Syntheseaktivitäten in der Produktentstehung

Auf Basis einer genaueren Untersuchung dieser generischen Analyse- und Syntheseaktivitäten im Hinblick auf deren Wechselwirkungen mit einzelnen Zielen, ist es möglich, auf übergeordnete Zwecke von Zielen in der Produktentstehung zu schließen. Diesem Schluss liegt die These zugrunde, dass jede Aktivität in der Produktentstehung vor dem Hintergrund einer gewissen Zielsetzung verfolgt wird³²⁴.

³²³ vgl. Albers, Ebel & Lohmeyer 2012

³²⁴ vgl. Lohmeyer 2012, S. 116 - 119

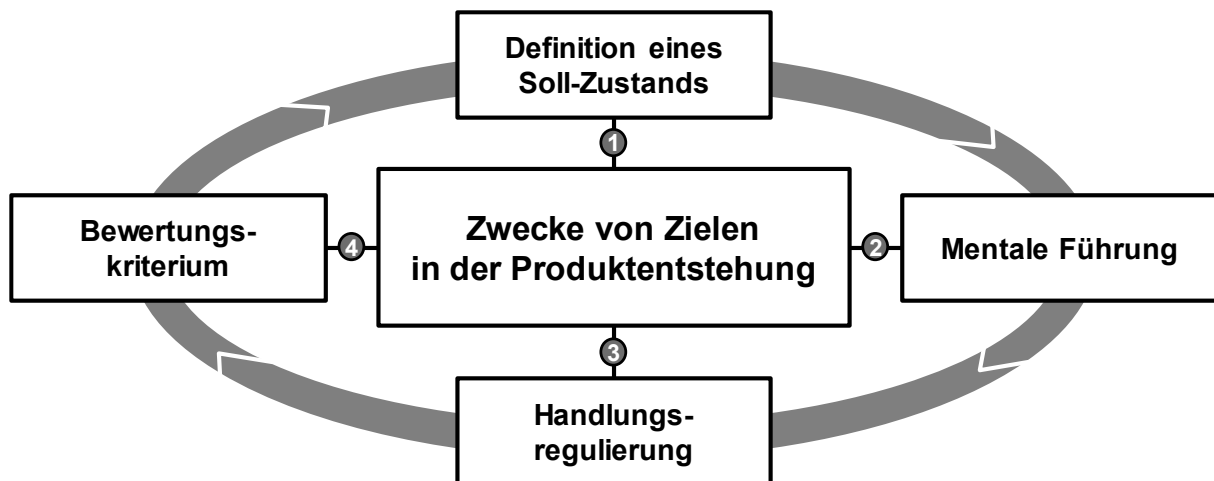


Abbildung 5-4: Übergeordnete Zwecke von Zielen in der Produktentstehung

Bei der **Synthese neuer Ziele** (Schritt 1 in Abbildung 5-3 und Abbildung 5-4) wird in erster Linie der Zweck verfolgt, einen zu **erreichenden Soll-Zustand** zu definieren. Der Zweck, der mit dieser Syntheseaktivität einhergeht, kommt in den meisten Definitionen des Zielbegriffs zum Ausdruck³²⁵. Hierbei gilt es klar zwischen dem eigentlichen Ziel (was soll erreicht werden) und einer möglichen Lösung und deren Eigenschaften (wie soll es erreicht werden) zu unterscheiden. Beispielsweise bildet die Kraftstoffverbrauchsreduzierung ein Ziel in der heutigen Fahrzeugentwicklung, das durch den Einsatz eines Turboladers (Anforderung) zum Zwecke des Downsizings erreicht werden kann. Dies ist in besonderem Maße relevant für die anschließende **Ziel-Analyse** (Schritt 2), in der die explizierten Ziele durch einzelne Akteure wahrgenommen, interpretiert und mental redefiniert werden. In diesem Zuge haben Ziele den Zweck, den Lösungsraum der einzelnen Akteure sinnvoll zu begrenzen, somit den Einsatz deren Wissensbasis zu steuern und die Akteure entsprechend mental zu führen. Die Wahrnehmung, Interpretation und Redefinition³²⁶ der explizit vorliegenden Ziele findet hierbei auf Basis der vorhandenen Wissensbasis und der mentalen Modelle (Perzeptions- und Kogitationsmodelle³²⁷) des einzelnen Akteurs statt, weshalb die **mentale Führung**, die durch explizierte Ziele hervorgerufen wird, individuell unterschiedlich ausfällt. Dies ist einerseits dann zu berücksichtigen, wenn die Wissensbasen und die mentalen Modelle der Ziel-analysierenden Akteure nicht oder nur teilweise bekannt sind, da in solchen Fällen die hervorgerufene mentale Führung nicht oder nur teilweise abgeschätzt werden kann und möglicherweise deutlich von der Intention des Ziel-synthetisierenden Akteurs abweicht. Andererseits führen ähnliche Wissensbasen und mentale Modelle

³²⁵ vgl. Abschnitt 2.5.1 und Definition 2-8

³²⁶ vgl. Abschnitt 2.2.3

³²⁷ vgl. Abschnitt 2.1.2 und Abbildung 2-5

zwischen Ziel-synthetisierenden und Ziel-analysierenden Akteuren dazu, dass die hervorgerufene mentale Führung im Wesentlichen der ursprünglichen Intention entspricht. Hierdurch wird es den einzelnen Akteuren möglich, unvollständig explizierter Zielsysteme schlüssig zu ergänzen bzw. zu konkretisieren und damit der vorherrschenden Unsicherheit³²⁸ erfolgreich zu begegnen. So kann die Analyse des Ziels der Kraftstoffverbrauchsreduzierung beispielsweise mit (System-)Leichtbau und einer gleichzeitigen Steigerung der Fahrdynamik assoziiert werden oder mit der Nutzung eines Hybridantriebs, um hierdurch gleichzeitig die Innovationskraft des Unternehmens darzustellen. Auf Basis eines zweckmäßig begrenzten Lösungsraums, kann im nächsten Schritt eine zielführende **Synthese neuer Objekte** erfolgen. In diesem Zuge haben Ziele den Zweck, Handlungen der Akteure auf mentaler, virtueller und physischer Ebene anzustoßen und zu regulieren (**Handlungsregulierung**)³²⁹. Hierdurch kann die Planung und Steuerung der zur Zielerreichung erforderlichen (Synthese-)Aktivitäten auf einem abstrakteren Niveau erfolgen, da die einzelnen Akteure in die Lage versetzt werden, aus den Zielen konkrete Aktivitäten zur Zielerreichung abzuleiten und diese selbständig auf Basis einer übergeordneten (Projekt-)Planung zu regulieren. Beispielsweise führt das erstmalig zu realisierende Ziel einer CFK-Karosserie in Großserienmaßstäben zu vollständig anderen Produktentstehungsprozessen, wie die bekannte und beherrschte Entwicklung einer konventionellen Stahlkarosserie. Im Anschluss an die Synthese neuer Objekte haben Ziele den Zweck, bei der **Analyse dieser Objekte** als Kriterien zur **Bewertung der Zielerreichung** zu dienen. Hierbei liegt die Herausforderung zum einen darin, geeignete Kriterien zu identifizieren, die tatsächlich die Zielerreichung bewerten und die gleichzeitig auf die synthetisierten Objekte anwendbar sind. Zum anderen müssen geeignete Randbedingungen definiert werden, unter denen die Überprüfung der Zielerreichung durchgeführt werden soll, da die Randbedingungen einen erheblichen Einfluss auf das Ergebnis der Überprüfung und dessen Gültigkeit haben. Die Herausforderung wächst sowohl mit der Unbekanntheit des zu synthetisierenden Objekts als auch mit der Unbekanntheit der zu definierenden Randbedingungen. Im Falle der erstmaligen Entwicklung einer CFK-Karosserie gilt es beispielsweise etablierte Crash-Tests und damit verbundene Simulationsverfahren zu hinterfragen bzw. neu zu definieren, da diese im Regelfall für konventionelle Stahlkonstruktionen entworfen wurden.

Ziele haben infolgedessen nicht nur den Zweck, einen gewünschten Soll-Zustand zu definieren. Vielmehr definieren Ziele indirekt auch den entsprechenden Prozess der

³²⁸ vgl. Abschnitt 2.3.2

³²⁹ vgl. Abschnitt 2.4

Überführung in diesen Soll-Zustand, da sie den betrachteten Lösungsraum, die Durchführung einzelner Aktivitäten sowie die Überprüfung der Zielerreichung wesentlich beeinflussen. Bei der Bildung und Nutzung eines Zielsystemmodells müssen folglich alle vier Zwecke berücksichtigt werden, um auf einen angemessenen Grad der Verkürzung schließen zu können. Hierbei entscheidet maßgeblich die Bekanntheit der Wissensbasis und der mentalen Modelle des Modellnutzers über die mögliche Verkürzung der Ziele, insbesondere der Darstellung des gewünschten Soll-Zustandes. So ist ein erfahrener Fahrzeugentwicklungsleiter sicherlich in der Lage, das Ziel der Kraftstoffverbrauchsreduzierung sinnvoll zu verorten und zielführend zu verfeinern, wohingegen ein unerfahrener Entwickler eines Zulieferers höchstwahrscheinlich eine weitaus detaillierte Zielsetzung benötigen würde. Zur Überprüfung der korrekten Interpretation und Redefinition eines Zielsystemmodells durch den Modellnutzer können zum einen die expliziten Verfeinerungen herangezogen werden, die der Modellnutzer im Zielsystemmodell vornimmt. Zum anderen können die geplanten Aktivitäten, des Modellnutzer beleuchtet werden, um auf die Korrektheit der Interpretation und Redefinition zu schließen. Beispielsweise könnte der (unerfahrene) Entwickler des Zulieferers gebeten werden, einen Projektplan und erste Lösungsvorschläge zu präsentieren. Einen großen Einfluss auf die korrekte Interpretation und Redefinition explizierter Ziele haben ebenfalls die entsprechenden Bewertungskriterien und Randbedingungen, unter denen eine Überprüfung der Zielerreichung durchgeführt wird. Sind diese Bewertungskriterien und Randbedingungen frühzeitig bekannt bzw. werden frühzeitig definiert, kann dies die Synthese gewünschter Objekte ebenso unterstützen, wie deren valide Überprüfung. So ist die Nennung eines bzw. mehrerer Test-Zyklen zur Kraftstoffverbrauchsermittlung neben der eigentlichen Vorgabe des Kraftstoffverbrauchs unerlässlich, um eine zielgerichtete Entwicklung zu ermöglichen.

Insgesamt sind alle vier Zwecke von Zielen relevant für die Realisierung eines gewünschten Soll-Zustandes. Die Effektivität und Effizienz der Modellbildung und -nutzung kann jedoch erheblich gesteigert werden, wenn alle vier Zwecke gemeinsam betrachtet werden und aufeinander abgestimmt sind.

5.3 Beurteilung von Zielen³³⁰

Die Beurteilung einzelner Ziele stellt eine wesentliche Voraussetzung für die Schaffung eines umfassenden Zielsystemverständnisses dar. Erst wenn einzelne Ziele richtig beurteilt und verstanden sind, können zielgerichtete Aktivitäten geplant, durchgeführt und überprüft werden. Dabei sind die Kriterien, unter denen eine

³³⁰ vgl. Albers, Lohmeyer & Ebel 2011

Beurteilung im einzelnen Fall vorgenommen werden muss, oftmals vielfältig, selten einfach überschaubar und nur teilweise auf andere Fälle übertragbar. Dies betrifft zum einen die **explizite Beurteilung von Zielen** anhand entsprechender Metriken, wodurch eine automatisierte Auswertung möglich wird und Schlüsse gezogen werden können, die nicht nur auf einzelnen Zielen basieren, sondern auf Zielsystemen³³¹. Zum anderen betrifft dies die **implizite Beurteilung von Zielen**, die jeder Akteur intuitiv und auf Basis seiner mentalen Modelle vornimmt. Eine Vielzahl der intuitiv vorgenommenen Beurteilungen von Zielen wird hierbei nicht expliziert, weil sich die einzelnen Akteure ihrer Bewertung unter Umständen nicht bewusst sind oder sie aus Gründen der Effizienz bewusst auf eine Explikation verzichten. Für den einzelnen Akteur ist dies eine effiziente und scheinbar zielführende Vorgehensweise, jedoch besteht die Gefahr, dass aufgrund einer mangelnden Explikation und Objektivierung kein gemeinsames Ziel- und Problemverständnis ausgebildet werden kann³³². Um dieser Gefahr entgegen zu wirken und gleichzeitig eine hohe Effizienz in der Beurteilung von Zielen realisieren zu können, bietet sich eine Etablierung gemeinsamer mentaler Modelle zur Beurteilung von Zielen an. Die Basis hierzu kann ein einfaches, generisches und damit flexibel einsetzbares Beurteilungsmodell für Ziele bilden. Hierzu werden zuerst vier nicht-orthogonale, jedoch klar differenzierbare **Beurteilungsdimensionen** von Zielen hergeleitet, die sequentiell auf Ziele angewendet werden können. Anschließend werden die vier separaten Beurteilungen zusammen geführt, indem jeweils zwei Beurteilungsdimensionen in einer Matrix gegeneinander aufgetragen werden. Die so entstehenden sechs Matrizen können ebenfalls sequentiell analysiert werden, wobei jede Matrix entsprechend ihrer Beurteilungsdimensionen und deren Ausprägung unterschiedliche methodische Hilfestellungen bereitstellt. Hierdurch kann sowohl ein Beitrag zu einer gemeinsamen Sprache geleistet werden als auch einzelne Akteure bei der Reflektion bzw. einem Perspektivenwechsel unterstützt werden³³². Auf Basis korrekt beurteilter und verstandener Ziele ist es weiterhin möglich, einen zweckdienlichen Verkürzungsgrad für die Explikation der Ziele festzulegen und damit die Effizienz der Modellbildung und Effektivität der Modellnutzung eines expliziten Zielsystemmodells zu steigern.

5.3.1 Beurteilungsdimensionen von Zielen

Im Nachfolgenden werden auf Basis der Grundlagen und des Standes der Forschung vier generische Beurteilungsdimensionen hergeleitet und erläutert, die insbesondere eine implizite Beurteilung von Zielen unterstützen sollen.

³³¹ vgl. Kapitel 7

³³² vgl. Abschnitt 5.1

Reifegrad von Zielen

Der Reifegrad bezieht sich in der Regel auf den Zustand eines Produkts hinsichtlich definierter Indikatoren beispielsweise der Anzahl erfüllter Anforderungen³³³. Hierdurch werden Aussagen über die (Un-)Vollständigkeit der Realisierung eines zu entwickelnden Produktes möglich. Zumeist sind hierzu jedoch konkrete Objekte erforderlich, anhand derer eine objektive Ermittlung der entsprechenden Indikatoren vorgenommen werden kann. Diese Objekte liegen bei der Analyse und insbesondere bei der Synthese von Zielen oftmals (noch) nicht vor. Da einer zielgerichteten Synthese von Objekten die Synthese und Analyse von Zielen vorausgehen muss, ist die Bewertung des Reifegrades eines Ziels mittels geeigneter Indikatoren eine zweckmäßige Ergänzung zu Reifegraden, die eher den Grad der Zielerfüllung bzw. Produktrealisierung abbilden. Als Indikatoren zur Bewertung des Reifegrades einzelner Ziele können die vorhandenen **Wissens- und Definitionslücken** (Unsicherheiten³³⁴) herangezogen werden, die bezüglich einzelner Ziele existieren. Zur Operationalisierung lassen sich die verschiedenen Zwecke von Zielen³³⁵ nutzen, die hinsichtlich vorhandener Wissens- und Definitionslücken bewertet werden. Der Reifegrad eines Ziels beschreibt damit die Vollständigkeit, mit der die Wissens- und Definitionslücken des Ziels geschlossen sind. Folglich korreliert der Reifegrad eines Ziels negativ mit der vorherrschenden Unsicherheit bezüglich des Ziels.

Definition 5.1: Reifegrad von Zielen

Der Reifegrad von Zielen beschreibt die Vollständigkeit, mit der die Wissens- und Definitionslücken eines Ziels geschlossen sind.

Die Unterteilung der Unsicherheit in Wissens- und Definitionslücken erlaubt eine differenziertere Beurteilung des Reifegrades eines Ziels, da sowohl ein unzureichendes Verständnis als auch eine unzureichende Explikation bzw. Dokumentation zu einer unzureichenden Gewissheit über ein Ziel führen können. Auf dieser Basis ist eine subjektive und implizite Beurteilung des Reifegrades eines Ziels und der entsprechend vorherrschenden Unsicherheit vergleichsweise intuitiv möglich. Eine Explikation und Kommunikation dieser Beurteilung erlaubt es den Akteuren darüber hinaus, ein intersubjektives und interdisziplinäres Zielverständnis aufzubauen, wodurch potenzielle Fehlentwicklungen frühzeitig identifiziert und vermieden werden können. Eine objektive und explizite Beurteilung des Reifegrades eines Ziels kann auf Basis der verschiedenen Zwecke von Zielen erfolgen. Da die Definition des

³³³ vgl. Abschnitt 2.4.3

³³⁴ vgl. Abschnitt 2.3.2

³³⁵ vgl. Abschnitt 5.2

Reifegrades von Zielen zugunsten einer intuitiv-impliziten Anwendbarkeit vergleichsweise generisch formuliert ist, gestaltet sich dessen Objektivierung aufwendiger und muss für eine Operationalisierung basierend auf den Zwecken einzelner Ziele weiter detailliert werden. Als Beispiel lässt sich das Ziel einer Kraftstoffverbrauchsreduzierung mittels Hybridantrieb anführen. Um die Zielvorgabe eines bestimmten Kraftstoffverbrauchs vollständig überblicken zu können, müssen eine Vielzahl weiterer Ziele und Anforderungen, beispielsweise das Fahrzeuggesamtgewicht oder die Kapazität der Batterien sowie die Wechselwirkungen dieser Ziele und Anforderungen in Erfahrung gebracht bzw. definiert werden.

Härtegrad von Zielen

Die zahlreichen Ziele, die durch ein Produkt erreicht werden müssen, sind bezüglich der Priorität der Zielerreichung in der Regel nicht gleichwertig. Aus verschiedenen Gründen werden Ziele durch unterschiedliche Interessensvertreter unterschiedlich priorisiert. Eine der zentralen Aufgaben in der Entwicklung mechatronischer Produkte ist die Abwägung und Normierung der unterschiedlichen Zielprioritäten, da die einzelnen Interessensvertreter dies in der Regel nicht (alleine) leisten können und tendenziell die von ihnen eingebrachten Ziele in deren Priorität überschätzen. Erst auf Basis einheitlich priorisierter Ziele ist eine sinnvolle Planung von Entwicklungsaktivitäten und Entwicklungskapazitäten möglich. Weiterhin haben die jeweiligen Ziele und deren Prioritäten einen wesentlichen Einfluss auf die mentale Führung und Handlungsregulierung der einzelnen Akteure³³⁶. Einheitlich priorisierte Ziele sind ferner für die erfolgreiche Bewältigung sich anbahnender oder vorhandener Zielkonflikte erforderlich, da sich Zielkonflikte oftmals nur durch Änderungen an einzelnen Zielen und Anforderungen auflösen lassen. Nicht zuletzt wirken sich priorisierte Ziele positiv auf die mögliche Optimierung eines Produkts hinsichtlich Zielerfüllung aus, da hierdurch mögliche Freiheitsgrade, die durch geringfügige Änderung einzelner Ziele entstehen, sichtbar und nutzbar werden.

Um die Priorität einzelner Ziele zum Ausdruck zu bringen, wird oftmals eine Unterteilung in solche Ziele vorgenommen, die unter allen Umständen erfüllt werden müssen und solche, die nach Möglichkeit berücksichtigt werden sollen³³⁷. Da die Priorisierung der Ziele de facto die mögliche Änderbarkeit eines Ziels impliziert, besteht durch eine binäre Priorisierung der Ziele die Gefahr, dass die Änderbarkeit eines Ziels ebenfalls als binär wahrgenommen wird. Dieser Gefahr kann entgegengewirkt werden, indem die scheinbar vorhandene Kopplung der Priorität

³³⁶ vgl. Abschnitt 5.2

³³⁷ vgl. Pahl et al. 2007, S. 215

und **Änderbarkeit eines Ziels** explizit aufgehoben wird, d.h. dass die Änderbarkeit eines Ziels separat definiert wird.

Definition 5.2: Härtegrad von Zielen

Der Härtegrad von Zielen beschreibt die definierte Änderbarkeit eines Ziels.

Der Härtegrad eines Ziels spiegelt damit den Willen der Akteure wieder, an einem bestehenden Ziel festzuhalten und ist folglich ein Maß für die Beständigkeit des Ziels im Prozess der Produktentstehung. Durch die Beurteilung des Härtegrads eines Ziels, ist es einem Akteur folglich möglich, die Beständigkeit bzw. Änderbarkeit eines Ziels gesondert zu hinterfragen. Hierdurch kann zum einen aufgedeckt werden, dass ein Ziel möglicherweise deutlich weniger beständig ist als angenommen, wodurch Anpassungen bei abgeleiteten Zielen und Lösungen hinsichtlich ihrer Robustheit gegen Zieländerungen nötig werden. Zum anderen können auf Basis einer identifizierten Änderbarkeit eines Ziels, das als unumstößlich expliziert bzw. interpretiert wurde, Freiheitsgrade resultieren, die wiederum zur Zielkonfliktbewältigung oder Optimierung eines Produkts genutzt werden können. Die Beurteilung des Härtegrades kann dabei sowohl implizit und subjektiv als auch explizit und objektiv vorgenommen werden, da der Härtegrad nicht zwingend aus Indikatoren abgeleitet werden muss und entsprechend im Rahmen einer beliebigen (numerischen) Skala definiert werden kann. Als Beispiel lässt sich abermals das Ziel einer Kraftstoffverbrauchsreduzierung nennen, das im Rahmen einer Fahrzeugentwicklung mit einem hohen Härtegrad, beispielsweise neun von zehn Punkten, bewertet wurde und damit unbedingt einzuhalten ist. Das hieraus (scheinbar) abgeleitete Ziel der Realisierung eines Hybridantriebs wurde lediglich mit zwei von zehn Punkten definiert, so dass der Hybridantrieb auch durch Leichtbaumaßnahmen ersetzt werden kann, sofern das definierte Kraftstoffverbrauchsziel weiterhin erreicht wird.

Auswirkung von Zielen

Einzelne Ziele sind in vielfältiger Art sowohl untereinander als auch mit abgeleiteten Anforderungen sowie den Funktionen und der Gestalt eines Produktes verknüpft³³⁸. Aufgrund dieser Verknüpfungen hat die Definition oder Änderung eines Ziels vielfältige Auswirkungen, die im Rahmen der Beurteilung des Ziels berücksichtigt werden müssen. Die Beurteilung der Auswirkungen ist jedoch nicht trivial. Zum einen müssen neben den unmittelbaren zumeist auch die mittelbaren Verknüpfungen berücksichtigt werden, weshalb die Anzahl der zu betrachtenden Elemente aufgrund des exponentiellen Zusammenhangs oftmals nur schwer zu erfassen und zu

³³⁸ vgl. Abschnitt 2.3.3

überblicken ist. Zum anderen müssen die individuellen Auswirkungen eines jeden verknüpften Elements voraus gedacht und beurteilt werden, um die Auswirkungen des definierten oder geänderten Ziels vollständig erfassen zu können.

Grundsätzlich können die Auswirkungen einer Zieldefinition oder -änderung in zwei Kategorien unterschieden werden. **Endogene Auswirkungen** beziehen sich auf das zu entwickelnde Produkt und umfassen die potenziell erforderlichen Zeit-, Ressourcen- und Kostenaufwände, die eine Zieldefinition oder -änderung nach sich zieht. **Exogene Auswirkungen** beziehen sich auf den Abnehmer bzw. Nutzer (Kunden) des Produkts und dessen Zufriedenheit in Folge einer Zieldefinition oder -änderung.

Definition 5.3: Auswirkung von Zielen

Die Auswirkung von Zielen beschreibt die Konsequenzen einer Zieldefinition oder -änderung hinsichtlich der erforderlichen Umsetzungsaufwände und der resultierenden Kundenzufriedenheit.

Potenzielle Zieldefinitionen und Zieländerungen sind daher sowohl hinsichtlich der erforderlichen Umsetzungsaufwände als auch der resultierenden Kundenzufriedenheit bzw. dem resultierenden Anbieternutzen zu beurteilen. Hierdurch wird eine ausgewogene Betrachtung der Auswirkungen von Zielen erreicht, da sowohl eine technische Sichtweise als auch eine Kundensichtweise eingenommen werden muss. Eine vollständige Objektivierung der Auswirkungen von Zielen gestaltet sich schwierig und ist Gegenstand aktueller Forschung. So lassen sich bestehende Beziehungen innerhalb bzw. zwischen einzelner Domänen (beispielsweise Ziele und Funktionen) durch Design-Structure-Matrices (DSM) bzw. Multiple-Design-Matrices (MDM) abbilden und anschließend mittels mathematischer Methoden (Graphentheorie) analysieren³³⁹. Der Aufwand der Modellierung ist jedoch beträchtlich, da jede mögliche Beziehung zwischen einzelnen Elementen explizit untersucht und bewertet werden muss, wodurch der Bewertungsaufwand folglich mit dem Quadrat der betrachteten Elemente skaliert. Oftmals ist es jedoch bereits auf Basis einer (inter-)subjektiven Beurteilung möglich, die relevanten Auswirkungen eines Ziels zu ermitteln und teilweise zu objektivieren. Eine explizite Beurteilung der Auswirkung eines Ziels mittels einer beliebigen (numerischen) Skala, kann beispielsweise im Rahmen einer Diskussion mit verschiedenen Akteuren unterschiedlicher Disziplinen erheblich zu einer Objektivierung der Auswirkungen des Ziels beitragen, da unbekannte Wechselwirkungen und fehlerhafte Annahmen auf Basis der Diskrepanzen in den Beurteilungen aufgedeckt werden können. Als Beispiel sei das

³³⁹ vgl. Lindemann & Maurer 2007

Vorhaben in der Fahrzeugentwicklung angeführt, ein definiertes Kraftstoffverbrauchsziel mittels der Realisierung eines Hybridantriebs zu erreichen. Dieses Ziel hat weitreichende Auswirkungen sowohl auf das Fahrzeug selbst, beispielsweise das zu erreichende Gewichtsziel (endogene Auswirkung), als auch auf den Endkunden, beispielsweise über den resultierenden Kaufpreis des Fahrzeugs (exogene Auswirkung).

Beeinflussbarkeit von Zielen

Das Zielsystem wird durch das Handlungssystem synthetisiert³⁴⁰. Zum Zweck der Operationalisierung werden in interdisziplinären Produktentstehungsprozessen einzelne Ziele von unterschiedlichen Akteuren unterschiedlicher Organisationseinheiten verantwortet³⁴¹. Die Fragmentierung des Zielsystems wird hierbei in der Regel auf Basis der Kompetenzen und Hierarchien der einzelnen Organisationseinheiten bzw. Akteure vorgenommen. Aufgrund der Komplexität heutiger mechatronischer Produkte werden die jeweils entwicklungsrelevanten Ziele einzelner Teilsysteme des Gesamtprodukts nicht mehr nur von einem Akteur bzw. einer Organisationseinheit verantwortet. Die einzelnen Akteure sehen sich daher regelmäßig mit der Herausforderung konfrontiert, zwar für das Ergebnis einer Teilsystementwicklung verantwortlich zu sein, jedoch nicht alle hierfür relevante Ziele vollständig beeinflussen zu können. Diese Herausforderung ist Chance und Risiko zugleich, insbesondere wenn noch nicht alle relevanten Ziele bekannt bzw. definiert sind. Auf der einen Seite kann durch eine geschickte Beeinflussung der Ziele sowohl die Entwicklung einzelner Teilsysteme als auch des Gesamtprodukts positiv beeinflusst werden. Auf der anderen Seite kann eine unausgewogene Beeinflussung einzelner Ziele zwar die Entwicklung einzelner Teilsysteme erheblich begünstigen, gleichzeitig jedoch die Entwicklung des Gesamtprodukts massiv behindern.

Definition 5.4: Beeinflussbarkeit von Zielen

Die Beeinflussbarkeit von Zielen beschreibt die Fähigkeit einzelner Akteure oder Organisationseinheiten, ein bestimmtes Ziel zu definieren bzw. zu ändern.

Die Beeinflussbarkeit eines Ziels ist oftmals kein absoluter Wert, sondern abhängig vom betrachtenden Akteur bzw. von der betrachtenden Organisationseinheit und dessen tatsächliche Fähigkeit der Einflussnahme auf ein Ziel. In der realen Produktentstehung kann folglich zwischen einer formellen und einer informellen Beeinflussbarkeit von Zielen unterschieden werden. Die formelle Beeinflussbarkeit

³⁴⁰ vgl. Abschnitt 2.2.3

³⁴¹ vgl. Abschnitt 2.5 und Abschnitt 5.1

eines Ziels definiert den Akteur bzw. die Organisationseinheit, der bzw. die formell Verfeinerungen bzw. Änderungen an einem bestimmten Ziel vornehmen darf. Die formelle Beeinflussbarkeit entspricht damit der **Autorisation** eines Ziels und kann entsprechend einfach expliziert und objektiviert werden. Die informelle Beeinflussbarkeit eines Ziels beschreibt hingegen die Fähigkeit der **informellen Einflussnahme** auf ein bestimmtes Ziel. Aufgrund des informellen Charakters gestaltet sich eine Objektivierung schwierig, da die einzelnen Akteure in der Regel ihre tatsächlichen Fähigkeiten der Einflussnahme auf einzelne Ziele nicht, unvollständig oder verfälscht explizieren. Dennoch oder gerade deswegen hat die informelle Beeinflussbarkeit eines Ziels einen erheblichen Einfluss auf die Fähigkeit insbesondere eines Akteurs, ein bestimmtes Ziel zu definieren oder zu ändern³⁴². Als Beispiel kann im Rahmen der Fahrzeugentwicklung der Kosten- oder Leistungsumfang eines Sub-Systems genannt werden, das der Fahrzeughersteller definieren kann, der Zulieferer jedoch nicht. Über entsprechende Beziehungen einzelner Akteure des Zulieferers können jedoch, beispielsweise im Falle eines Zielkonflikts oder technischer Probleme, Änderungen der definierten Ziele veranlasst werden.

5.3.2 Matrizen von Beurteilungsdimensionen

Die vier Beurteilungsdimensionen von Zielen (vgl. Abbildung 5-5) erlauben eine separierte Beurteilung eines Ziels aus unterschiedlichen Blickwinkeln. Die Separierung führt zu einer Vereinfachung der Beurteilung, da aufgrund der jeweils vorgenommenen Fokussierung nicht alle für eine umfassende Beurteilung erforderlichen Beurteilungskriterien gleichzeitig erfasst und präsent sein müssen. Die Vereinfachung wird jedoch durch eine Reduzierung der Komplexität erreicht, da relevante Wechselwirkungen zwischen den einzelnen Beurteilungsdimensionen ausgeblendet werden³⁴³. Um der Komplexität realer Beurteilungssituationen gerecht werden zu können, müssen daher auch die Wechselwirkungen zwischen den Beurteilungsdimensionen betrachtet werden. Hierzu werden jeweils zwei der vier Beurteilungsdimensionen in einer Matrix einander gegenüber gestellt, in ihren Ausprägungen jeweils als niedrig und hoch quantisiert und die hierdurch resultierenden Ausprägungskombinationen hinsichtlich ihrer Bedeutung analysiert. Auf Basis der resultierenden sechs Matrizen lassen sich hierdurch Wechselwirkungen zwischen den Beurteilungsdimensionen auf einer vergleichsweise generischen Ebene darstellen und darüber hinaus mögliche Handlungsempfehlungen ableiten.

³⁴² vgl. Jupp, Eckert & Clarkson 2009

³⁴³ vgl. Abschnitt 2.3.1

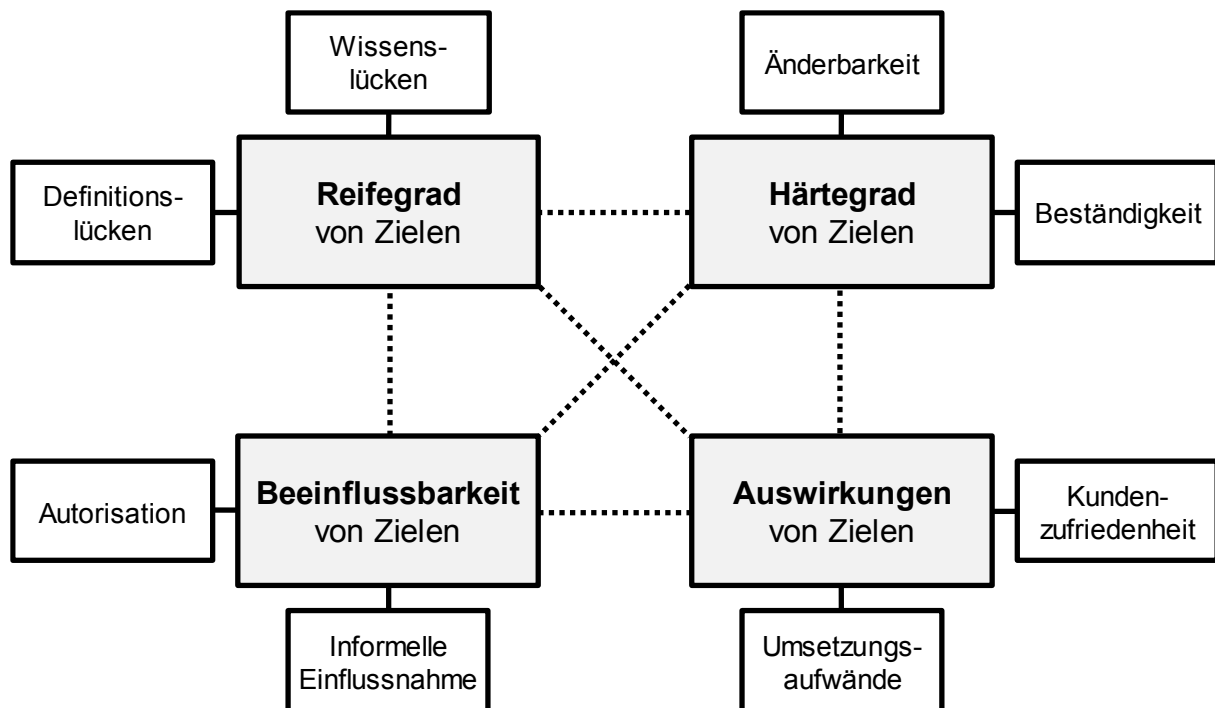


Abbildung 5-5: Beurteilungsdimensionen von Zielen

Kritikalitätsmatrix (Beeinflussbarkeit über Auswirkung)

Die Kritikalitätsmatrix entsteht durch eine Gegenüberstellung der Beeinflussbarkeit und der Auswirkung eines Ziels und ermöglicht Aussagen über die **Kritikalität des Ziels** (vgl. Abbildung 5-6). Die Matrix gibt dem einzelnen Akteur damit Hinweise, welche Ziele mit besonderer Aufmerksamkeit beobachtet werden müssen und folglich dem Akteur stets präsent sein sollten. Ziele, deren Auswirkungen eher als niedrig eingeschätzt werden und deren Beeinflussbarkeit als hoch angenommen wird, sind als trivial einzustufen und bedürfen keiner besonderen Aufmerksamkeit. Ziele mit niedriger Beeinflussbarkeit bei gleichzeitig niedrigen Auswirkungen sind zwar als vergleichsweise unwichtig einzuschätzen, sollten jedoch mit einem Mindestmaß an Aufmerksamkeit beobachtet werden, da eine Fehleinschätzung der Auswirkungen eines Ziels aufgrund der niedrigen Beeinflussbarkeit folgenschwere Konsequenzen haben kann. Ziele mit hoher Auswirkung sind definitionsgemäß relevante, wichtige Ziele, die einen erheblichen Einfluss auf das Produkt und den Produktentstehungsprozess haben. Auch im Fall hoher Beeinflussbarkeit sollte diesen Zielen ein erhebliches Maß an Aufmerksamkeit zukommen. Falls erforderlich, sind Änderungen an solchen Zielen zwar prinzipiell durchführbar, jedoch stets mit gewichtigen und erfolgsrelevanten Auswirkungen hinsichtlich Änderungsaufwand bzw. Kundenzufriedenheit. Im Fall einer niedrigen Beeinflussbarkeit sind Ziele mit hoher Auswirkung extrem kritisch, da auf erfolgsrelevante Definitionen oder Änderungen keinen oder wenig Einfluss genommen werden kann. Diese Ziele sind mit höchster Aufmerksamkeit zu beobachten. Darüber hinaus sollte in Erwägung

gezogen werden, Maßnahmen zur Senkung der Auswirkungen bzw. Erhöhung der Beeinflussbarkeit zu ergreifen.

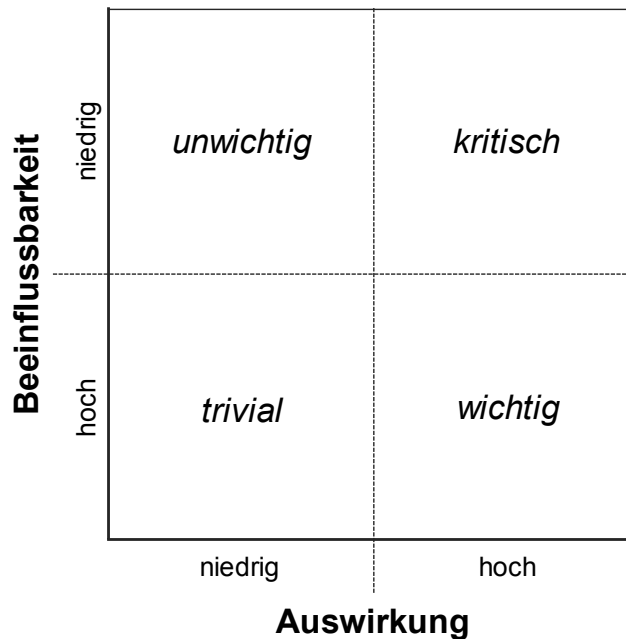


Abbildung 5-6: Kritikalitätsmatrix

So ist beispielsweise die Vorgabe der Kraftstoffbetankungsseite eines Fahrzeugs vergleichsweise trivial, die Vorgaben des Sonnenblendenspiegels im Regelfall unwichtig, das Fahrzeugdesign sehr wichtig und die Vorgaben an zugelieferte Komponenten des Motors (Steuerkette) kritisch.

Entscheidungsmatrix (Härtegrad über Reifegrad)

Die Entscheidungsmatrix resultiert durch Vergleich des Härtegrades eines Ziels mit dessen Reifegrad und bietet eine Entscheidungsunterstützung bei der Synthese neuer Ziele durch Adressierung des **Unsicherheitsdilemmas**³⁴⁴ (vgl. Abbildung 5-7). Das Unsicherheitsdilemma beschreibt die grundlegende Problematik unsicherheitsbehafteter Prozesse, die darin besteht, dass die rationale Definition eines Ziels problemspezifisches Wissen voraussetzt, das jedoch nur auf Basis zweckmäßig definierter Ziele erarbeitet werden kann. Bei einer Kombination aus niedrigem Reife- und Härtegrad, wie sie oftmals zu Beginn eines Produktentstehungsprozesses vorliegt, muss der einzelne Akteur prinzipiell zwischen zwei alternativen Vorgehensweisen wählen, die beide jeweils Vor- und Nachteile beinhalten. Die konservative Vorgehensweise sieht in einem ersten Schritt den Aufbau eines tieferen Verständnisses des Ziels vor, um in einem zweiten Schritt eine rationale Priorisierung des Ziels vornehmen zu können. Diese Vorgehensweise trägt erheblich zur Risikominimierung

³⁴⁴ vgl. Abschnitt 2.3.2 und Lohmeyer 2012, S. 13

bei und sollte wann immer möglich favorisiert werden. Aus der anfangs nur vagen bzw. weichen Zieldefinition können etliche Iterationen kleineren Umfangs resultieren, die sich in Summe jedoch erheblich negativ auf die benötigten Entwicklungszeiten und -ressourcen auswirken können. Da die Produktentstehung aufgrund der globalen Konkurrenzsituation nicht vollkommen risikofrei sein kann, ist es jedoch oftmals unumgänglich, einige Ziele bereits zu Beginn eines Produktentstehungsprozesses sehr hart zu definieren, obwohl noch eine erhebliche Unsicherheit bezüglich dieser Ziele vorherrscht. Dies ist zumeist bei Zielen mit hoher Auswirkung erforderlich, um eine zielgerichtete Entwicklung der in Beziehung stehenden Ziele zu ermöglichen. Dieses Vorgehen ist mit Risiken verbunden, da in Folge einer weiteren Konkretisierung der anfangs sehr hart definierten, unsicherheitsbehafteten Ziele auch Fehleinschätzungen aufgedeckt werden können, die zu Iterationen erheblichen Umfangs führen oder gar das Scheitern bedeuten können. Daher muss im Rahmen der Projektierung eines Projekts sichergestellt werden, dass insbesondere Ziele mit hohen Auswirkungen ein ausreichendes Maß an Gewissheit aufweisen. An beide Vorgehensweisen anschließend, muss die Umsetzung der Ziele in konkrete Objekte sichergestellt werden.

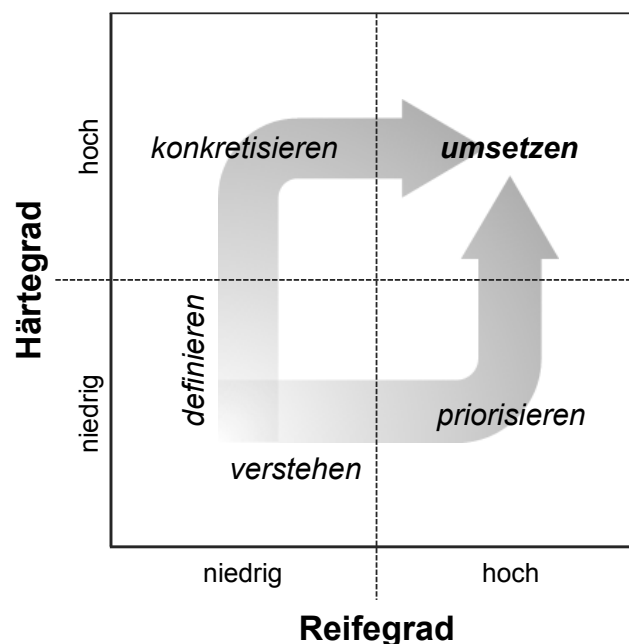


Abbildung 5-7: Entscheidungsmatrix

Als Beispiel kann die Zielvorgabe des Kraftstoffverbrauchs eines Fahrzeugs dienen, die bereits zu Beginn eines Produktentstehungsprozesses mit hohem Härtegrad definiert werden muss, da eine Vielzahl anderer Ziele von diesem Ziel und seiner Beständigkeit abhängig sind, die jedoch erst mit fortschreiten des Produktentstehungsprozesses sinnvoll konkretisiert werden kann. Demgegenüber steht beispielsweise das Kostenziel, das vor einer endgültigen Priorisierung bzw. Definition ausreichend verstanden, d.h. abgesichert sein sollte.

Definitionsmatrix (Auswirkung über Härtegrad)

Die Definitionsmatrix entsteht durch eine Gegenüberstellung der Auswirkungen und des Härtegrads eines Ziels, woraus sich Handlungsempfehlungen für die **Definition von Zielen** ableiten lassen (vgl. Abbildung 5-8). Ziele mit hoher Auswirkung sind dadurch gekennzeichnet, dass sie eine Vielzahl nach außen gerichteter Wechselwirkungen mit anderen Zielen aufweisen³⁴⁵ und folglich Änderungsmultiplikatoren darstellen³⁴⁶. Ziele mit niedriger Auswirkung haben entsprechend eine geringe Anzahl nach außen gerichteter Wechselwirkungen mit anderen Zielen und können daher als Änderungsabsorber wirken³⁴⁶. Für eine zielgerichtete und effiziente Entwicklung ist es daher erforderlich, Ziele mit hoher Auswirkung frühzeitig bzw. vorrangig zu anderen Zielen zu definieren. Hierbei sollte vor einer endgültigen Fixierung eines solchen Ziels validiert werden, ob das Ziel zu einer Befriedigung der Kundenwünsche bzw. Kundenbedürfnisse beiträgt und ob das Ziel einen ausreichenden Anbieternutzen realisiert. Ziele mit niedriger Auswirkung sollten hingegen nachrangig definiert bzw. fixiert werden, da hierdurch Freiheitsgrade zur Absorption notwendiger Änderungen erhalten bleiben, das Zielsystem somit an Robustheit gewinnt und unvorhergesehene bzw. unvorhersehbare Entwicklungen abgefangen werden können³⁴⁷.

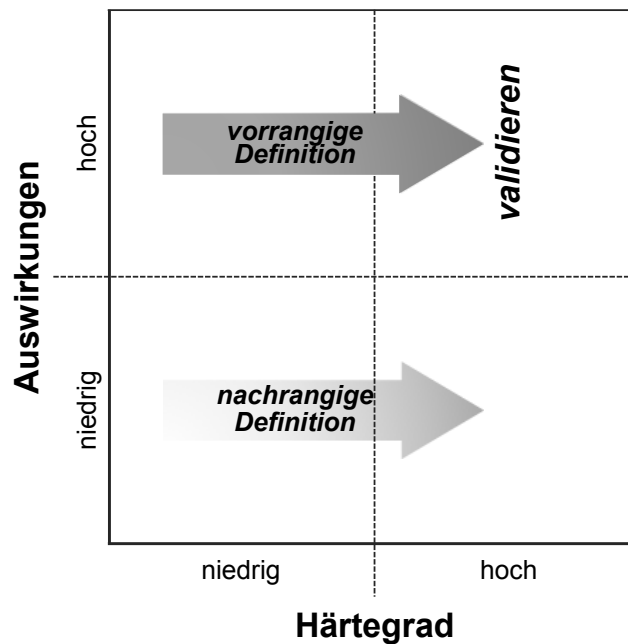


Abbildung 5-8: Definitionsmatrix

³⁴⁵ vgl. Abschnitt 2.5 und Abschnitt 5.3.1

³⁴⁶ vgl. Abschnitt 2.3.3 und Abbildung 2-15

³⁴⁷ vgl. Abschnitt 2.3.2

Als Beispiel einer vorrangigen Definition sei abermals das Kostenziel in der Fahrzeugentwicklung genannt, das vor einer endgültigen Definition ausreichend valide vorliegen sollte. Eine nachrangige Definition eines Ziels empfiehlt sich beispielsweise bei der Festlegung des Kraftstoffvorrats eines Fahrzeugs zur Erfüllung des Reichweitenziels.

Entwicklungsmatrix (Auswirkung über Reifegrad)

Die Entwicklungsmatrix resultiert durch Vergleich der Auswirkungen eines Ziels mit dessen Reifegrad, woraus sich Handlungsempfehlungen für die **Entwicklung von Zielen** ableiten lassen (vgl. Abbildung 5-9). Den Reifegrad eines Ziels zu erhöhen bedeutet, die vorherrschenden Wissens- und Definitionslücken aktiv durch Entwicklungsaktivitäten zu schließen. Hierbei empfiehlt es sich, Lösungen für Ziele mit hoher Auswirkung vorrangig zu entwickeln, um einerseits das Risiko fehlerhafter Annahmen zu reduzieren und andererseits die bestehenden Wechselwirkungen zu anderen Zielen besser zu verstehen und ggf. quantifizieren zu können. Hierdurch können insbesondere auch vorhandene Konflikte im Zielsystem aufgedeckt und frühzeitig aufgelöst werden, ohne dass Iterationen über einzelne Entwicklungsphasen hinweg erforderlich werden. Elemente, die aus einer zunehmenden Konkretisierung von Zielen mit hohen Auswirkungen resultieren, sollten dabei stets auf Zielerfüllung hin überprüft, d.h. verifiziert werden. Lösungen für Ziele mit niedriger Auswirkung sollten nachrangig entwickelt werden. Hierdurch kann die absorbierende Wirkung von Zielen mit niedriger Auswirkung genutzt werden, um Zielkonflikte effizient aufzulösen und umfangreiche Iterationen zu vermeiden. Hierbei muss jedoch berücksichtigt werden, dass die Auswirkungen eines Ziels mit steigender Reife in der Regel auch genauer beurteilt werden können. Ziele, deren Auswirkungen ursprünglich als gering beurteilt wurden, können sich bei zunehmender Konkretisierung folglich als Ziele mit deutlich höheren Auswirkungen heraus stellen, wodurch sie ggf. nicht mehr als Absorber, sondern als Multiplikatoren eines Änderungsverlaufs wirken. Ziele mit einer vermeintlich geringen Auswirkung dürfen folglich nicht gänzlich vernachlässigt werden und es gilt eine vernünftige Balance aus Aufwands- und Risikominimierung einzustellen.

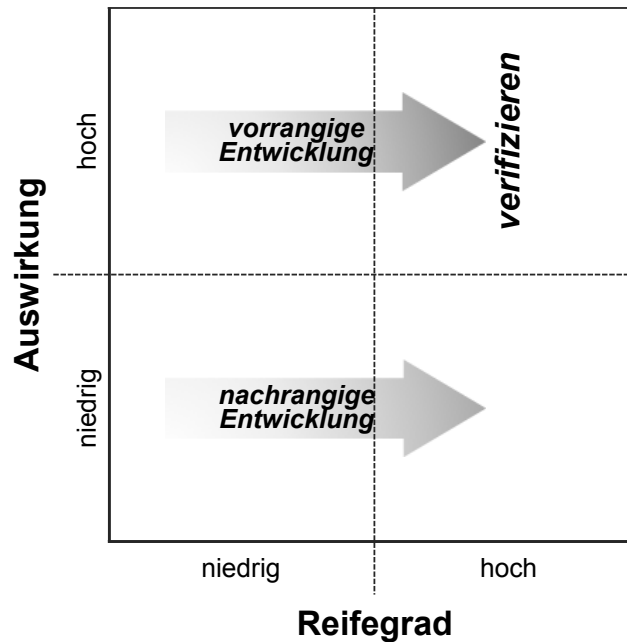


Abbildung 5-9: Entwicklungsmatrix

Als Beispiel kann die Entwicklung eines Fahrzeug-Hybridantriebs dienen. Hierbei sollte vorrangig die Topologie des Antriebsstrangs und der Antriebsstrang selbst entwickelt werden, bevor in die Entwicklung der konkreten Batterie investiert wird.

Partizipationsmatrix (Beeinflussbarkeit über Härtegrad)

Die Partizipationsmatrix entsteht durch Gegenüberstellung der Beeinflussbarkeit eines Ziels mit dessen Härtegrad woraus sich Handlungsempfehlungen für den **Definitionsprozess von Zielen** in der interdisziplinären Produktentstehung ableiten lassen (vgl. Abbildung 5-10). Die Änderbarkeit von Zielen muss im Prozess der Produktentstehung stetig eingeschränkt werden, da andernfalls keine Konvergenz und damit kein Produkt realisiert werden kann. Dies erfolgt zum einen kontinuierlich im Rahmen von Diskussionen und Verhandlungen zwischen einzelnen Akteuren über die Definition und Priorisierung einzelner Ziele und zum anderen diskret im Rahmen des Freezings³⁴⁸ etlicher Ziele zu entsprechenden Meilensteinen, beispielsweise der Spezifikations- bzw. Lasten- oder Pflichtenheftfreigabe. Werden Ziele definiert, die eine gewisse Auswirkung auf den Verantwortungsbereich eines Akteurs haben, sollte dieser Akteur im Vorfeld der Entscheidung bzw. im Rahmen der Entscheidungssituation an der Zieldefinition partizipieren. Andernfalls können die resultierenden Zieldefinitionen und -priorisierungen erheblich negative Auswirkungen auf den eigenen Verantwortungsbereich und ggf. auch auf die Qualität des zu entwickelnden Produktes haben, die in der Regel auch nicht ohne weiteres revidiert werden können.

³⁴⁸ vgl. Eger, Eckert & Clarkson 2005

Im Falle einer niedrigen Beeinflussbarkeit sollte ein Akteur entsprechend eine Partizipation an der Entscheidungsfindung von Zielen mit Auswirkungen auf den eigenen Verantwortungsbereich einfordern. Verantwortet ein Akteur ein Ziel, das er in hohem Maße selbst beeinflussen kann, sollte er mit Blick auf eine optimale Gesamtsystemlösung, die Partizipation anderer Akteure bzw. Interessensvertreter in Erwägung ziehen – nicht zuletzt auch, um bezüglich anderer Ziele eine Partizipation einfordern zu können.

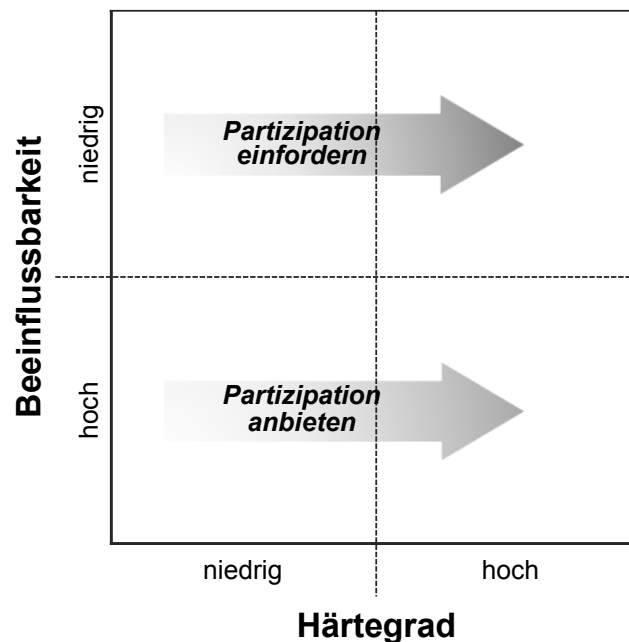


Abbildung 5-10: Partizipationsmatrix

Als Beispiel sei das auf Sub-Systeme heruntergebrochene Kosten- oder Gewichtsziel in der Fahrzeugentwicklung zu nennen, das von mehreren Akteuren verantwortet wird.

Konkretisierungsmatrix (Beeinflussbarkeit über Reifegrad)

Die Konkretisierungsmatrix entsteht durch Vergleich der Beeinflussbarkeit eines Ziels mit dessen Reifegrad woraus sich Handlungsempfehlungen für den **Konkretisierungsprozess von Zielen** in der interdisziplinären Produktentstehung ableiten lassen (vgl. Abbildung 5-11). Die einzelnen Ziele eines Zielsystems legen den zu erreichenden Soll-Zustand fest³⁴⁹. Aufgrund vorhandener Wechselwirkungen zwischen den einzelnen Zielen wird jedoch gleichzeitig der Raum möglicher Lösungen eingeschränkt³⁵⁰. Ist der Reifegrad eines Ziels vergleichsweise niedrig, sind in der Regel auch noch nicht alle Wechselwirkungen zu anderen Zielen bekannt

³⁴⁹ vgl. Abschnitt 2.5.1

³⁵⁰ vgl. Abbildung 2-35

bzw. können noch nicht genauer quantifiziert werden. Daher kann in der Regel auch nicht vollständig erfasst werden, in welchem Maße ein Ziel den Lösungsraum tatsächlich einschränkt. Dies gelingt erst mit zunehmender Konkretisierung eines Ziels. Hierbei entsteht im Falle einer hohen Beeinflussbarkeit eines Ziels die Möglichkeit, das Ziel derart zu konkretisieren, dass die einschränkende Wirkung des Ziels teilweise aufgehoben wird und neue Lösungsräume entstehen, die wiederum zur Optimierung des Produkts genutzt werden können. Oftmals ist für eine solche Optimierung nur eine geringe Anpassung einiger weniger Ziele erforderlich, die jedoch nicht umfassend beeinflusst werden können. Dennoch kann es einem Akteur gelingen, im Zuge des Konkretisierungsprozesses dieser Ziele, die erforderlichen Zielanpassungen und damit den benötigten Lösungsraum einzufordern, indem auf Basis einer Gesamtsystemoptimierung argumentiert wird.

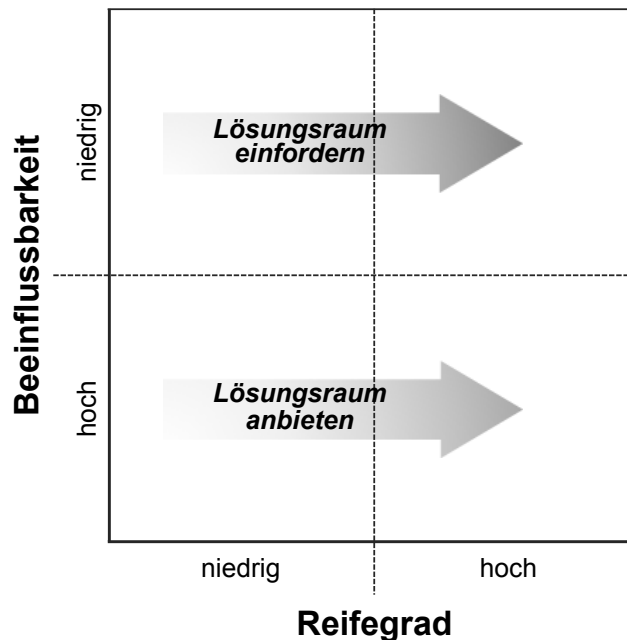


Abbildung 5-11: Konkretisierungsmatrix

Das Packaging in der Fahrzeugentwicklung, d.h. die Festlegung der von Sub-Systemen einzuhaltenden Bauräume, kann als Beispiel der Konkretisierungsmatrix angeführt werden.

6 Untersuchungen zur Modellierung von Zielsystemen

In diesem Kapitel wird die Modellierung von Zielsystemen in der industriellen Praxis anhand zweier Fallstudien untersucht. Die Fallstudien finden im Kontext realer Produktentwicklungsprojekte statt und haben zum Ziel, Erkenntnisse für die Strukturierung und Modellierung von Zielsystemen zu gewinnen. Hierbei wird ein wesentliches Augenmerk auf die **Anwendbarkeit** der Modellierungsmethode und des jeweils gewählten **Werkzeugs** durch die involvierten Entwickler gelegt. In beiden Fallstudien werden daher intuitiv handhabbare Werkzeuge eingesetzt, die sich entsprechend den Erfordernissen der Fallstudien flexibel anpassen lassen. Hierdurch wird sichergestellt dass die involvierten Entwickler die Modellierungsmethode in ihren Arbeitsalltag integrieren und selbstständig anwenden können.

6.1 Modellierung von Zielsystemen mittels Semantic MediaWikis³⁵¹

In der ersten Fallstudien wird das Zielsystem mittels eines Semantic MediaWiki abgebildet. „Ein Wiki ist eine frei erweiterbare Sammlung von verlinkten Web-Seiten, ein Hypertextsystem für die Speicherung und Modifizierung von Informationen – ein Datenbestand, dessen Web-Seiten von jedem Nutzer einfach mittels eines Web-Browsers editiert werden können.“³⁵² MediaWiki ist hierbei das populärste Softwarepaket zum Betrieb eines Wikis, das u.a. bei der Online-Enzyklopädie Wikipedia zum Einsatz kommt³⁵³. Der hohe Bekanntheitsgrad des MediaWikis verbunden mit einer hohen Benutzerfreundlichkeit und einer sehr guten Lesbarkeit führen in Summe zu einer hohen Akzeptanz seitens der Nutzer³⁵⁴.

6.1.1 Zielsetzung der Untersuchung

Die erste Fallstudie zur Modellierung von Zielsystemen verfolgt mehrere Ziele, die alle die Anwendbarkeit der Modellierungsmethode sowie des gewählten Werkzeugs durch die involvierten Entwickler adressieren.

- Explorative Suche nach geeigneten Strukturen des Zielsystems sowie angrenzender Partialmodelle.

³⁵¹ vgl. Albers, Ebel & Sauter 2010

³⁵² vgl. Leuf & Cunningham 2001, S. 14

³⁵³ vgl. <http://de.wikipedia.org>

³⁵⁴ vgl. Adler, Frost & Gross 2011

Im Rahmen der Fallstudie soll bewusst keine potenzielle Referenzstruktur des Zielsystems vorgegeben und bewertet werden. Vielmehr soll auf Basis der tatsächlichen Bedarfe eine geeignete Struktur identifiziert bzw. erarbeitet werden und somit die Bedarfssituation in der industriellen Praxis unverfälscht erfasst werden. Als einzige Orientierung soll die entsprechend dem ZHO-Modell³⁵⁵ vorgenommene Unterteilung in ein Ziel- und Objektsystem Berücksichtigung finden.

- Untersuchung der Zielsystementwicklung entlang des zeitlichen Verlaufs des Produktentstehungsprozesses.

Es ist zu erwarten, dass das Zielsystem mit fortschreitendem Produktentwicklungsprozess mehrere inhaltliche wie auch strukturelle Erweiterungen bzw. Änderungen erfahren wird. Hierbei ist weiterhin zu erwarten, dass ein Teil dieser Anpassungen auf neue Erkenntnisse über die Modellierungsmethode zurück zu führen sein wird. Ein anderer Teil der Anpassungen wird sich jedoch vermutlich aufgrund des Produktentstehungsprozesses und dessen Charakteristika ergeben. Beide Aspekte sind relevant für die Ableitung von Methoden zur Modellierung von Zielsystemen.

- Untersuchung möglicher Wechselwirkungen zwischen generellem Domänenwissen und fallspezifischem Wissen mit Auswirkungen auf die Modellierung von Zielsystemen.

Generelles Domänenwissen und fallspezifisches Wissen sind nach KLEIN³⁵⁶ neben Problemlösungs- und Kontrollwissen die Hauptkategorien des Wissens in der Produktentstehung. Fallspezifisches Wissen ist stark projektgebunden und umfasst im Wesentlichen Produkthanforderungen (Ziele) und Produktbeschreibungen (Objekte) wohingegen generelles Domänenwissen das nicht-fallspezifisch, übertragbare Wissen beschreibt, beispielsweise Beziehungen zwischen Strukturen, Funktionen und Verhalten oder Wissen über verschiedene Abstraktionsniveaus. Es ist davon auszugehen, dass eine starke Wechselwirkung zwischen diesen beiden Hauptkategorien des Wissens besteht, die insbesondere im Rahmen der Produktgenerationsentwicklung einen Einfluss auf die Modellierung von Zielsystemen hat, da hierbei große Teile bestehenden Wissens aus vorangegangener Produktgenerationen in die Entwicklung einer neuen Produktgeneration einfließt.

- Abschätzung des Potenzials semantischer Funktionalitäten im Rahmen der Zielsystemmodellierung.

³⁵⁵ vgl. Abschnitt 2.2.1

³⁵⁶ vgl. Klein 1998

Mittels semantischer Funktionalitäten können Bedeutungen von und zwischen Elementen des Zielsystems explizit abgebildet und als Ergänzung zur eigentlichen Information des Elements in einer Datenbank abgespeichert werden. Je strukturierter und expliziter die Semantiken (Bedeutungen) vorliegen, desto leistungsfähiger werden die Möglichkeiten, um mittels eines Computers mit dem modellierten Zielsystem zu arbeiten³⁵⁷. Im Fokus der Betrachtung stehen hierbei leistungsfähige Suchanfragen, die Informationen aus dem Zielsystem und angrenzender Partialmodelle extrahieren, die nicht unmittelbar auf der Hand liegen.

- Untersuchung der Tauglichkeit des Werkzeugs MediaWiki zur Zielsystemmodellierung.

Wikis haben ihre Popularität in der Vergangenheit insbesondere aufgrund ihrer einfachen Anwendbarkeit bei gleichzeitig hoher Leistungsfähigkeit erlangt (einfaches Editieren und einfache Dokumentation, einfache Veröffentlichung und Kommunikation, einfache Zusammenarbeit und Meinungsbildung).³⁵⁸ Daher gilt es einerseits zu klären, ob die einfache Anwendbarkeit und die hieraus resultierende Akzeptanz eines Wiki auch für den Fall der Modellierung von Zielsystemen gegeben sind. Andererseits muss geklärt werden, inwiefern die Modellierungsmöglichkeiten eines MediaWiki geeignet sind, um Zielsysteme im Verlauf eines Produktentstehungsprozesses zu modellieren.

6.1.2 Rahmenbedingungen der Untersuchung

Die Untersuchung fand im Rahmen eines Vorentwicklungsprojekts bei einem der weltweit führenden Hersteller von Bogenoffset-Druckmaschinen statt (vgl. Abbildung 6-1). Das Vorentwicklungsprojekt hatte zum Ziel, eine **alternative Bogenzuführungsidee** hinsichtlich Realisierbarkeit, Vor- und Nachteile zu bestehenden Bogenzuführungslösungen sowie Kosten belastbar zu bewerten. Hierdurch sollte untersucht werden, inwiefern die alternative Bogenzuführungsidee als innovative Neuerung in die nächste Produktgeneration von Bogenoffset-Druckmaschinen einfließen soll. Das Projekt wurde ab dem Zeitpunkt der Initiierung über **sechs Monate** hinweg begleitet. Das Projektteam bestand aus **drei Maschinenbauingenieuren** (inkl. dem Untersuchenden selbst), die zeitweise von verschiedenen **Experten angrenzender Disziplinen** (Elektrotechnik, Regelungstechnik und Informatik) unterstützt wurden. Im Zentrum der untersuchten Projektphase stand die Definition eines ersten, explizit vorliegenden Zielsystems der alternativen Bogenzuführung

³⁵⁷ vgl. Völkel 2006

³⁵⁸ vgl. Sauter 2012, S. 67

(initiales Zielsystem) sowie damit eng verbunden die Erarbeitung eines hierauf aufbauenden Bogenzuführungskonzepts.



Abbildung 6-1: Produktgenerationen von Bogenoffset-Druckmaschinen³⁵⁹

Das Wiki wurde lokal auf dem Laptop des Untersuchenden installiert, da eine Netzwerkinstallation einerseits zu erheblichen Komplikationen mit der Unternehmensinternen IT-Abteilung geführt hätte und andererseits der kollaborative Aspekt eines Wiki im Rahmen der Untersuchung nicht untersucht werden sollte. Entsprechend wurde das Wiki im Wesentlichen vom Untersuchenden selbst bedient.

6.1.3 Ergebnisse und Interpretation der Untersuchung

Die Ergebnisse der Untersuchung werden entsprechend der formulierten Zielsetzungen strukturiert und interpretiert und im Nachfolgenden dargelegt.

Strukturierung des Zielsystems und angrenzender Partialmodelle

Die Modellierung projektrelevanter Informationen wurde entsprechend dem ZHO-Modell der Systemtechnik³⁶⁰ prinzipiell strukturiert nach Zielen, die zu erfüllen sind und Objekten, die im Projektverlauf erzeugt wurden (vgl. Abbildung 6-2). Einzelne bzw. mehrere zusammengehörige Ziele und Objekte wurden dabei mittels einzelner Wiki-Seiten abgebildet, die aufgrund der gewählten Wiki-Umsetzung (MediaWiki) zunächst in keiner Beziehung zueinander stehen. Sämtliche Ziel- und Objektseiten wurden daher in eine **hierarchische Struktur** überführt, die sich wesentlich an der Funktions- später an der Baustruktur des Bogenzuführungssystems orientierte. Um die notwendig erscheinende Nachvollziehbarkeit zwischen Zielen und Objekten gewährleisten zu können, wurde zu jeder Funktion bzw. Baugruppe eine definierende **Ziel-Seite und eine korrespondierende Objekt-Seite**, die die entsprechende Ziel-Umsetzung beschreibt, angelegt. Hierdurch konnte mittels dreier **Hyperlinks** (Hyperlink-Header) auf jeder Wiki-Seite einfach sowohl durch die Hierarchie der Ziel- bzw. Objektseiten als auch zwischen einer Ziel- und Objektsicht navigiert werden

³⁵⁹ Mit freundlicher Genehmigung der Heidelberg Druckmaschinen AG

³⁶⁰ vgl. Abschnitt 2.2.1

(vgl. Abbildung 6-2 und Abbildung 6-3). Darüber hinaus wurden im Verlauf der einzelnen Wiki-Seiten vereinzelt Hyperlinks platziert, um Beziehungen zu anderen Wiki-Seiten herstellen zu können. Hierbei wurden neben Ziel- und Objektseiten auch Profil-Seiten einzelner Wissensträger sowie Dokumente, die in das Wiki geladen wurden oder im WorldWideWeb verfügbar sind, verlinkt.

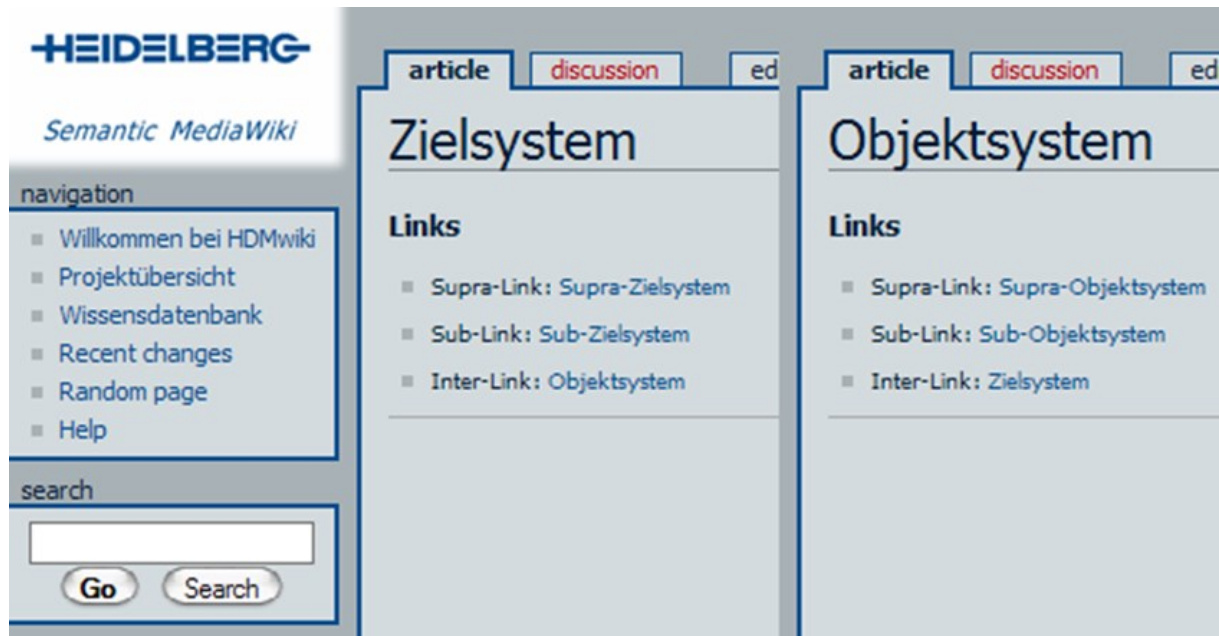


Abbildung 6-2: Vernetzung der Partialmodelle im MediaWiki-System über Hyperlinks

Die Inhalte der einzelnen **Ziel-Seiten** waren nach unterschiedlichen Kriterien strukturiert, beispielsweise nach den unterschiedlichen Projektzielen (vgl. Abbildung 2-29), nach der Quelle (vgl. Abbildung 2-37) sowie nach funktionaler und nichtfunktionaler Ausprägung. Ziele und zugehörige Anforderungen wurden stets gebündelt dokumentiert, so dass eine **Durchgängigkeit** von Zielen über Anforderungen bis hin zur umsetzenden Lösung realisiert werden konnte. Die Inhalte der einzelnen **Objekt-Seiten** wurden entsprechend dem Projektfortschritt strukturiert, d.h. neben dem aktuellen Lösungsstand wurden auch vergangene bzw. alternative Lösungen dokumentiert. Konkret orientierte sich die Strukturierung der Objekt-Seiten am Projekt-Entwicklungsprozess mit den Phasen Ideenfindung, Konzeptdefinition und Konzeptausgestaltung sowie innerhalb dieser Struktur anhand des SPALTEN-Prozesses³⁶¹. Hierbei wurde die Strukturierung nach SPALTEN insbesondere genutzt, um noch nicht entschiedene **alternative Umsetzungsmöglichkeiten** präsent zu halten (Alternative Lösungen) und **Begründungen** von getroffenen Entscheidungen zu dokumentieren (Lösungsauswahl).

³⁶¹ vgl. Abschnitt 2.4.2

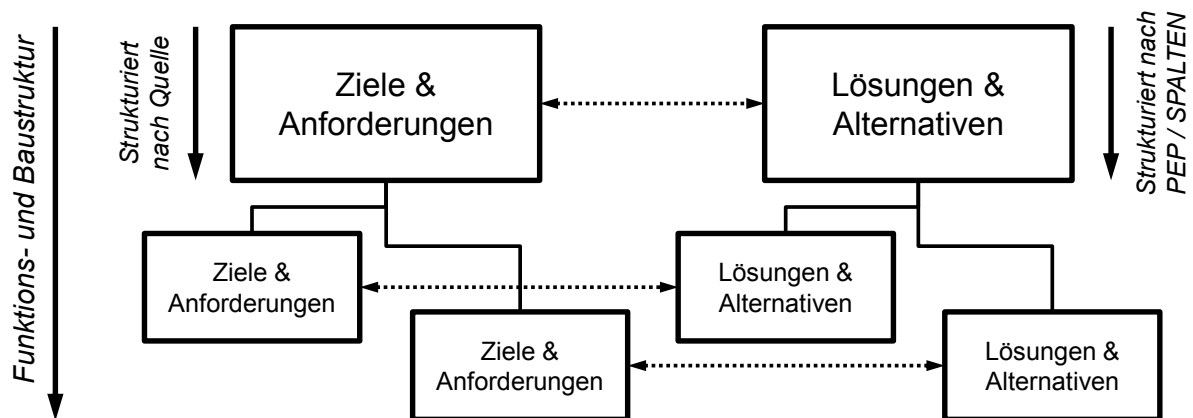


Abbildung 6-3: Strukturierung der Partialmodelle im MediaWiki-System

Zusammenfassend wurden im Rahmen der Untersuchung nachfolgende Partialmodelle genutzt, um die projektbezogenen und als relevant eingestuften Informationen abzubilden und damit das Zielsystem ganzheitlich und durchgängig modellieren zu können:

- **Ziele**
Ziele wurden zumeist in direkter Verbindung mit Anforderungen formuliert, sollten jedoch als eigenständiges Partialmodell aufgefasst werden. Hierbei bietet sich eine Strukturierung der Ziele nach den unterschiedlichen Projektzielen (vgl. Abbildung 2-29) bzw. Quellen der Ziele (vgl. Abbildung 2-37) an.
- **Anforderungen**
Anforderungen wurden zumeist in direkter Verbindung mit Zielen formuliert, sollten jedoch als eigenständiges Partialmodell aufgefasst werden. Hierbei bietet sich eine Strukturierung nach funktionalen (mit einer Funktion unmittelbar in Beziehung stehenden) und nichtfunktionalen (mit einer Funktion nur mittelbar in Beziehung stehenden) Anforderungen an (vgl. Abbildung 2-36), um hierdurch die zu realisierenden Eigenschaften zu definieren³⁶².
- **Funktionsstruktur**
Die Funktionsstruktur diene als strukturierendes Kriterium für Ziele, Anforderungen und Lösungen und wurde nicht explizit aufgestellt. Die Strukturierung der Objekt-Seiten nach funktionalen Gesichtspunkten hatte jedoch einen erheblichen Einfluss auf die Suche nach Lösungen. Daher wird die Funktionsstruktur als separates Partialmodell geführt.
- **Baustruktur**
Die Baustruktur diene neben der Funktionsstruktur als strukturierendes Kriterium für Ziele, Anforderungen und insbesondere Lösungen. Sämtliche Lösun-

³⁶² vgl. Weber 2012

gen, denen eine (physische) Gestalt zugeordnet werden kann, finden sich in der Baustruktur wieder. Insofern ist die Baustruktur ein wichtiges Partialmodell um (physische) Gestalt und die damit verbundenen Merkmale³⁶² abzubilden.

- **Projektphasen, Meilensteine und Arbeitsergebnisse**

Die einzelnen Projektphasen wurden zusammen mit den sie untergliedernden Projektaktivitäten genutzt, um einerseits den Projektfortschritt zu dokumentieren und andererseits die Projektplanung inkl. der zu liefernden Arbeitsergebnisse (Deliverables) präsent zu halten. Hierbei wurden insbesondere solche Arbeitsergebnisse (Deliverables) definiert, die zum jeweiligen Ende einzelner Projektphasen, d.h. zu Meilensteinen (Gates), vorliegen sollen. Da diese prozessrelevanten Informationen einen erheblichen Einfluss auf die Entwicklung des Systems hatten, sollten auch Partialmodelle, die eher dem Handlungssystem zuzuordnen sind, bei einer ganzheitlichen Zielsystemmodellierung berücksichtigt werden (Projektphasen sowie Meilensteine und Arbeitsergebnisse).

- **Projektaktivitäten (Entscheidungen)**

Einzelne Aktivitäten wurden hauptsächlich entsprechend der SPALTEN-Problemlösungsmethodik dokumentiert. Hierbei konnte zum einen gezeigt werden, dass eine Untergliederung der vergleichsweise grob ausdifferenzierten Projektphasen mittels weiterer (Produktentstehungs-)Aktivitäten notwendig ist, um die Projektplanung zu operationalisieren. Zum anderen konnte gezeigt werden, dass zu Entscheidungssituationen oftmals eine starke Informationsverdichtung bei gleichzeitig hoher Informationsqualität vorliegt, weshalb Entscheidungssituationen und damit verbundene Informationen bzw. Dokumente unbedingt modelliert werden sollten.

- **Stakeholder**

Die Zuordnung bzw. Verlinkung von Wiki-Seiten zu einzelnen Wissensträgern (Stakeholder) wurde in zweierlei Hinsicht verwendet. Zum einen wurden einzelne Informationen ihrem Wissensträger zugeordnet, so dass eine Verbindung zwischen der verkürzt vorliegenden expliziten Information und der jeweiligen Wissensquelle hergestellt werden kann. Zum anderen wurden einzelne Zuständigkeiten bzw. Verantwortlichkeiten, beispielsweise von Baugruppen oder Aktivitäten, über die Verlinkungen zugeordnet. Da die einzelnen Stakeholder eine wichtige Rolle im Produktentstehungsprozess einnehmen, empfiehlt es sich, ein separates Stakeholder-Partialmodell zu modellieren.

Zeitliche Entwicklung des Zielsystems

Das Zielsystem sowie die angrenzenden Partialmodelle haben im Verlauf der Untersuchung erwartungsgemäß sowohl strukturelle als auch inhaltliche Veränderungen erfahren.

Etliche strukturelle Anpassungen sind erforderlich geworden, da die notwendigen bzw. geeigneten Partialmodelle und deren jeweilige Struktur erst im Verlauf der Untersuchung deutlich geworden sind. Diese Anpassungen waren mit erheblichen (kognitiven) Aufwänden verbunden, die nur aufgrund der vorgenommenen Untersuchung und der damit verbundenen Unterstützung durch den Untersuchenden akzeptiert wurden. Folglich müssen in regulären Entwicklungsprojekten **von Beginn an geeignete Partialmodelle und Strukturen vorhanden sein**, da andernfalls keine individuelle Akzeptanz erreicht werden kann. Strukturelle Anpassungen sind weiterhin aufgrund aufgetretener **Iterationen** erforderlich geworden. Fehlerhafte Annahmen und ausgeschlossene Alternativen erforderten mehrmals einzelne Wiki-Seiten zu überarbeiten und deren Einbindung in das Netzwerk der übrigen Wiki-Seiten anzupassen. Diese Anpassungen wurden zum einen aufgrund der zunehmenden **Unübersichtlichkeit** notwendig, die durch eine undifferenzierte Aufbereitung von alternativen, nicht weiterverfolgten und fehlerhaften Lösungspfaden entstanden ist. Zum anderen zeigte sich, dass diese Lösungspfade oftmals zur **Begründung** des aktuell gültigen Lösungspfades erforderlich sind und daher nicht ohne weiteres aus dem Zielsystem entfernt werden dürfen. Strukturelle Änderungen des Zielsystems, die aufgrund der vorherrschenden Unsicherheit in Produktentstehungsprojekten resultieren, müssen folglich zugelassen und möglichst effizient durch geeignete Methoden und Werkzeuge unterstützt werden.

Auf einer makroskopischen Ebene betrachtet entwickelten sich das Zielsystem sowie die angrenzenden Partialmodelle beginnend bei Zielen, über Anforderungen und Funktionen bis hin zu Baugruppen und Bauteilen. Diese makroskopische Betrachtung wird jedoch der tatsächlich stattgefundenen Entwicklung nicht gerecht und würde zu falschen Schlüssen und damit zu einer kaum anwendbaren Methode zur Modellierung von Zielsystemen führen. Dies ist hauptsächlich dadurch begründet, dass die Entwicklung der einzelnen Partialmodelle nicht sequentiell und losgelöst voneinander stattgefunden hat, sondern vielmehr **iterativ und in starker Wechselwirkung** vorgenommen wurde. So wurde ein überwiegender Teil der Ziele erst im iterativen Wechselspiel mit der Definition von Anforderungen bzw. Suche nach Gestaltmerkmalen identifiziert bzw. expliziert. Auch die Definition, Änderung bzw. Verfeinerung von Anforderungen fand unter starker Wechselwirkung mit den gewählten Lösungen und deren Merkmalen statt. Hierbei kam der **Funktionsstruktur** die Rolle eines Mittlers zwischen Anforderungen und den sie realisierenden Lösungen in Form von Baugruppen und Bauteilen mit ihren jeweiligen Merkmalen zu.

Auch die Ausgestaltung des Projekt-Entwicklungsprozess mittels einzelner (Produktentstehungs-)Aktivitäten unterlag einer stark wechselseitigen Beziehung zur Entwicklung der Ziele, Anforderungen und nicht zuletzt der (möglichen) Lösungen. In Summe konnte beobachtet werden, dass der Projekt-Entwicklungsprozess trotz der ursprünglich definierten und sequenziell aufeinander aufbauenden drei Projektphasen vielmehr wie ein **Spiral-Prozess**³⁶³ ausgestaltet wurde. Für die Modellierung von Zielsystemen und angrenzender Partialmodelle bedeutet dies, dass die Reife aller relevanter Partialmodelle kontinuierlich im Verlauf des Produktentstehungsprozess erhöht werden muss und die einzelnen Partialmodelle folglich von Beginn an in einem iterativen Wechselspiel zu entwickeln sind.

Fallspezifisches Wissen und generelles Domänenwissen

Zu Beginn der Untersuchung wurde davon ausgegangen, dass eine starke Wechselwirkung zwischen generellem Domänenwissen und fallspezifischem Wissen besteht, die einen Einfluss auf die Modellierung von Zielsystemen hat. Daher wurde das MediaWiki-System als **Wissens- und Projektdatenbank** konzipiert, die innerhalb einer gemeinsamen Systemumgebung jeweils einen Bereich für die Abbildung von generellem Domänenwissen (unterteilt nach Wissensbereichen) und fallspezifischem Wissen (unterteilt nach Projekten) zur Verfügung stellt (vgl. Abbildung 6-4). Die Unterteilung in diese beiden Bereiche erfolgte dabei lediglich auf der Ebene der Wiki-Seiten mittels unterschiedlicher Kategorisierung und Ausgestaltung der jeweiligen Seiten sowie im Falle der Projektdatenbank über die Einbindung der Seiten mittels Hyperlinks in die Struktur der übrigen projektbezogenen Seiten. Hierdurch konnte die Voraussetzung für eine **Durchlässigkeit** zwischen fallspezifischem Wissen und generellem Domänenwissen geschaffen werden.

Im Verlauf der Untersuchung wurde deutlich, dass starke Wechselwirkungen zwischen generellem Domänenwissen und fallspezifischem Wissen bestehen und diese Wechselwirkungen auch einen erheblichen Einfluss auf die Modellierung von Zielsystemen und angrenzender Partialmodelle haben.

Hierbei konnten im Wesentlichen drei Arten von Wechselwirkungen identifiziert werden, die insbesondere im Rahmen der Zielsystemmodellierung relevant sind:

- Identifikation relevanter Variablen
- Ermittlung projektspezifischer Variablenwerte
- Ermittlung vorgegebener Variablenwerte

³⁶³ vgl. Abbildung 2-25

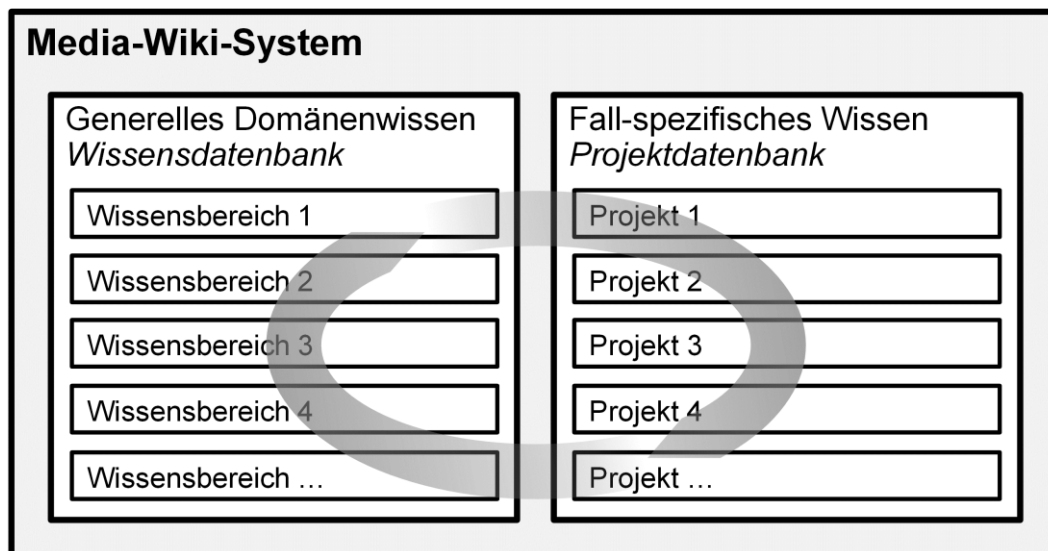


Abbildung 6-4: Wissen im MediaWiki-System

Die **Identifikation relevanter Variablen** erfolgte größtenteils auf Basis der Analyse bestehender Lasten- und Pflichtenhefte und hatte zum Ziel, alle relevanten Aspekte im Rahmen der Zielsystemmodellierung zu berücksichtigen. Hierbei sind insbesondere die verschiedenen nichtfunktionalen Anforderungen (vgl. Abbildung 2-36) zu nennen, die zum Zwecke der Operationalisierung branchen- und produktspezifisch ausdifferenziert sein müssen. Weiterhin wurde das bestehende generelle Domänenwissen genutzt, um **projektspezifische Werte** für die im Zielsystem zu berücksichtigenden Variablen zu ermitteln. Hierbei wurde zum einen **prozedurales Methoden- und Prozesswissen** zur projektspezifischen Parametrierung relevanter Variablen genutzt, beispielsweise Auslegungs- und Berechnungsvorschriften bzw. -programme. Zum anderen wurde auf **deklaratives Faktenwissen** zurückgegriffen wie beispielsweise einzuhaltende Normen und Gesetze, um konkrete, vorgegebene Werte von Variablen definieren zu können. Diese beobachteten Wechselwirkungen zwischen generellem Domänenwissen und fallspezifischem Wissen waren im Verlauf der Untersuchung keineswegs unidirektional, d.h. im Verlauf des Projekts wurde fallspezifisches Wissen erzeugt, das zu einem gewissen Anteil generalisierbar und damit der Wissensdatenbank zuführbar war (vgl. Abbildung 6-4).

Insgesamt zeigte sich, dass eine **Durchlässigkeit** zwischen generellem Domänenwissen und fallspezifischem Wissen und den jeweils abbildenden Methoden und Werkzeugen sowohl die Qualität und Effizienz der Zielsystemmodellierung einzelner Projekte erhöhen kann als auch zu einem langfristigen organisationalem Lernzuwachs beitragen kann. Hierdurch ist eine Organisation insbesondere im Rahmen der **Produktgenerationsentwicklung** in der Lage, erhebliche Wettbewerbsvorteile zu generieren, da die Entwicklungsressourcen vollumfänglich auf den

Modifikations- und Neuentwicklungsanteil einer neuen Produktgeneration gelegt werden können.

Potenzial semantischer Funktionalitäten

Zur Untersuchung des Potenzials semantischer Funktionalitäten im Kontext der Zielsystemmodellierung wurde das MediaWiki-System mit den Erweiterung Semantic MediaWiki³⁶⁴ und Halo-Extension³⁶⁵ ausgestattet. Hierbei stellt die Erweiterung Semantic MediaWiki die wesentlichen semantischen Funktionalitäten zur Verfügung. Im Kern sind dies die Definition expliziter Eigenschaften (beispielsweise *Has Requirement: Requirement123*) und Relationen (beispielsweise *Satisfies: Requirement123*) sowie die strukturierte Ausgabe von Suchanfragen mit semantischem Inhalt (beispielsweise *Satisfies: Requirement123*). Die Halo-Extension baut auf dieser Erweiterung auf und fügt insbesondere Funktionalitäten hinzu, die die Anwendbarkeit des semantischen MediaWiki-Systems verbessern. So werden beispielsweise eine Autovervollständigung der Suchfunktion (vgl. Abbildung 6-5) sowie ein OntologyBrowser (vgl. Abbildung 6-7) hinzugefügt, der eine Navigation durch die Eigenschaften und Relationen des Semantic MediaWiki ermöglicht.

Im Rahmen der Untersuchung wurden die semantischen Funktionalitäten nicht mit höchster Priorität verwendet, d.h. nicht alle modellierten Inhalte wurden entsprechend der Notation des Semantic MediaWiki expliziert und damit für das MediaWiki-System interpretierbar gestaltet. So wurden hauptsächlich die Ziele und Anforderungen sowie die einzelnen Phasen und Aktivitäten des Projekts mit expliziten Eigenschaften und Relationen versehen. Jedoch konnte bereits auf dieser Basis das große **Potenzial aufgezeigt** werden, das eine Modellierung des Zielsystems mit expliziten Eigenschaften und Relationen beinhaltet. So ließen sich beispielsweise alle alternativen Lösungen auflisten, die innerhalb der Projektphase Ideenfindung generiert wurden. Weiterhin konnte beispielhaft aufgezeigt werden, welche Lösungen zur Erfüllung (satisfy) einer Anforderung beitragen. Es konnte damit gezeigt werden, dass unter konsequenter Verwendung relevanter Eigenschaften und Relation es möglich ist, Informationen aus dem Zielsystem und angrenzender Partialmodelle zu extrahieren, die nicht unmittelbar auf der Hand liegen. Hierdurch kann ein entscheidender Mehrwert im Umgang mit dem explizierten Zielsystemmodell realisiert werden, der den Mehraufwand in der Modellierung zu rechtfertigen vermag.

³⁶⁴ <https://semantic-mediawiki.org/>

³⁶⁵ <https://semantic-mediawiki.org/wiki/Halo>

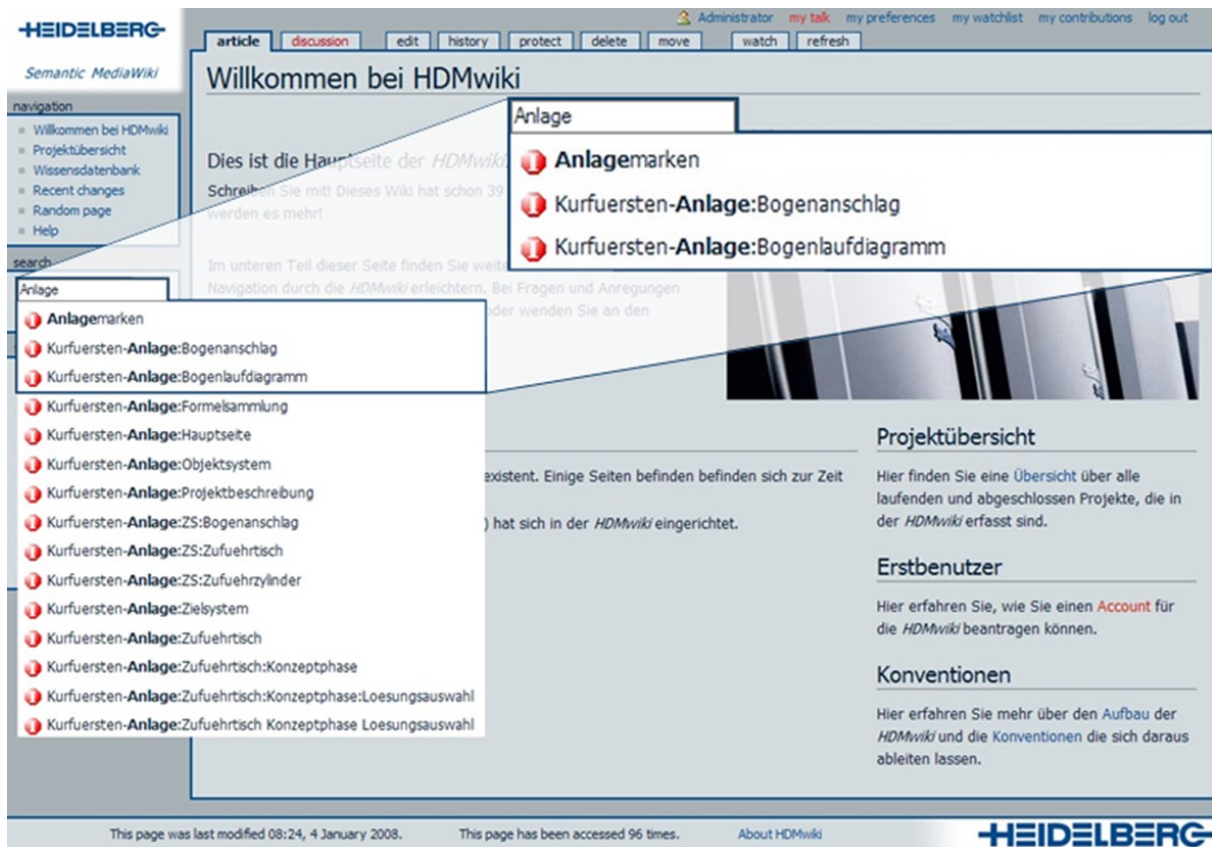


Abbildung 6-5: Autovervollständigung der Suchfunktion im MediaWiki-System

Die Untersuchung zeigte jedoch auch, dass die verwendeten Eigenschaften und Relationen dem Entwickler **nicht in Form einer abstrakten Ontologie**³⁶⁶ präsentiert werden dürfen, sondern fallspezifisch und entsprechend ausschnittsweise mittels eines geeigneten Werkzeugs zur Verfügung gestellt werden müssen. Hierbei empfiehlt sich, dass der Entwickler aus einer **geschlossenen Liste** relevanter Eigenschaften und Relationen auswählen kann und nicht wie im Fall der Semantic MediaWiki eine textuelle Eingabe sowohl der Eigenschaften als auch deren fallspezifischen Ausprägung vornehmen muss.

Tauglichkeit des MediaWiki zur Zielsystemmodellierung

MediaWiki ist ein freies Softwarepaket zum Betrieb eines Wikis, das ursprünglich für die Online-Enzyklopädie Wikipedia geschrieben wurde³⁶⁷. Dieser Ursprung ist dem Werkzeug inhärent und entsprechend den Erkenntnissen aus der Untersuchung der wesentliche Grund, weshalb ein MediaWiki-System **nur durchschnittlich geeignet** ist für die Modellierung von Zielsystemen in der interdisziplinären Produktentstehung. Prinzipiell lassen sich die meisten Defizite durch einen erhöhten Modellierungsauf-

³⁶⁶ vgl. Gruber 1993 "A specification of a representational vocabulary for a shared domain of discourse – definitions of classes, relations, functions, and other objects – is called an ontology."

³⁶⁷ vgl. <http://www.mediawiki.org/>

wand weitestgehend kompensieren, so dass der Nutzen eines entsprechend modellierten Zielsystems durchaus gegeben sein kann. Jedoch zeigten die Untersuchung und insbesondere die Aussagen der involvierten Entwickler, dass ein solcher (Mehr-)Aufwand im Kontext realer Produktentwicklung nicht geleistet werden kann bzw. wird.

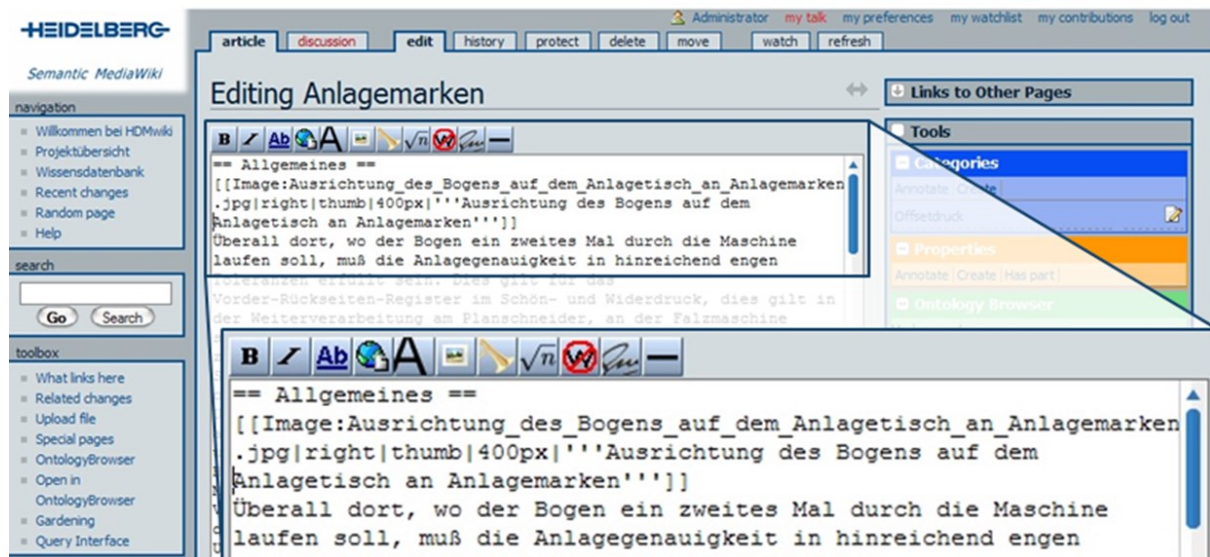
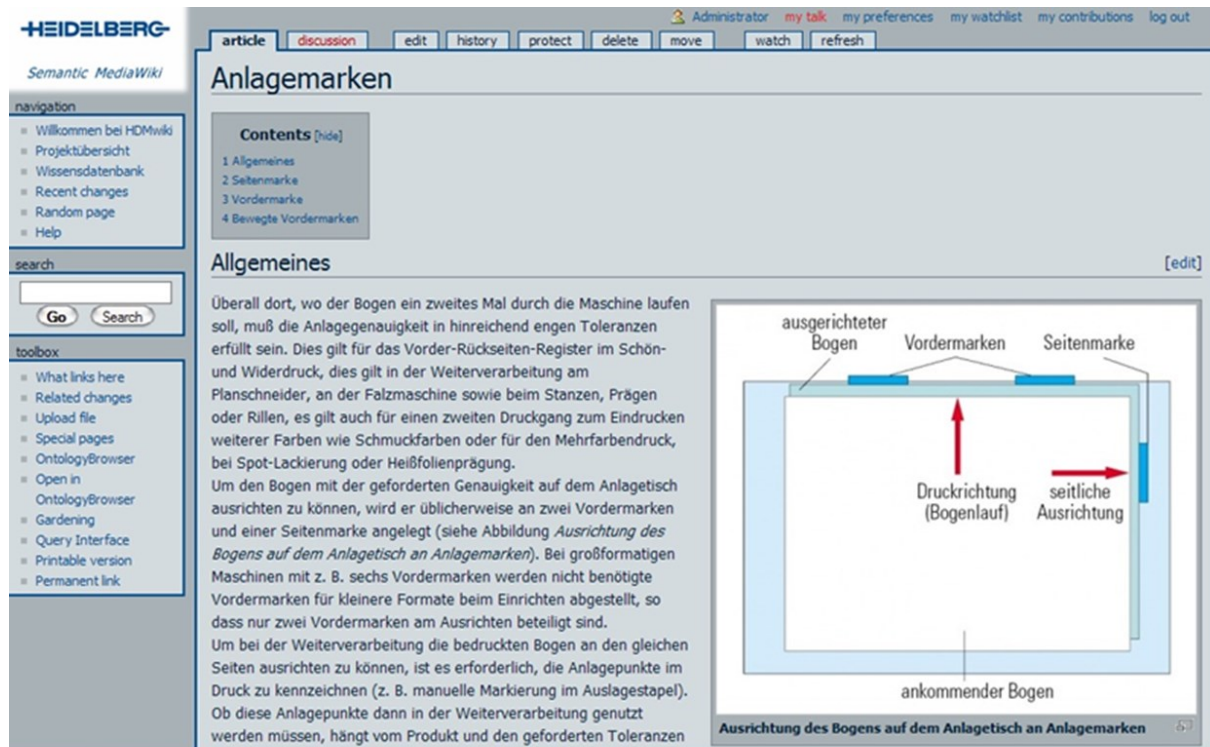


Abbildung 6-6: MediaWiki-Seite und zugehöriger MediaWiki-Code

Eine der wesentlichen Nachteile des MediaWiki-Systems besteht in der prinzipiell **nicht vorhandenen Hierarchie** der einzelnen Wiki-Seiten. Die Untersuchung zeigte, dass die Entwickler zwar in vernetzten Zielen und Anforderungen denken, sich innerhalb dieser Strukturen jedoch klare Hierarchien ausbilden, die die Entwickler

insbesondere als strukturierendes Kriterium für eine explizite Modellierung nutzen. Dies spiegelt sich auch in vorhandenen und etablierten Anforderungs- und Spezifikationsdokumenten wider, die im Wesentlichen hierarchisch gegliedert sind. Diese Dokumente und deren Strukturen dürfen nicht ignoriert werden, da sie einerseits einen erheblichen branchen- bzw. unternehmensspezifischen Erfahrungsschatz bergen und sie andererseits auch die Denkweise der Entwickler beeinflussen. Die folglich sinnvolle hierarchische Strukturierung der Inhalte des Zielsystems und angrenzender Partialmodelle sollte entsprechend werkzeugseitig unterstützt werden. Das MediaWiki-System konnte eine entsprechende Hierarchie durch die hervorragenden Möglichkeiten der Vernetzung einzelner Wiki-Seiten und Seiteninhalte via Hyperlinks realisieren. Jedoch musste die gewünschte Hierarchie auf jeder einzelnen Seite durch einen entsprechenden Hyperlink-Header (vgl. Abbildung 6-2) umgesetzt werden. Die Möglichkeiten der Vernetzung konnten weiterhin genutzt werden, um beliebige Relationen zwischen einzelnen Wiki-Seiten herzustellen und damit Strukturen, die notwendigerweise über die reine Hierarchie hinausgehen, zu schaffen. Hierbei stellte sich als nachteilig heraus, dass die resultierende Hierarchie bzw. **Struktur der Wiki-Seiten nicht unmittelbar überschaubar** ist, sondern erst auf Basis eines iterativen Folgens von Hyperlinks erfahrbar wird. Hierdurch ist es mittels eines MediaWiki nicht möglich, einen schnellen Überblick über das modellierte Zielsystem zu erhalten. Diesem mangelnden Überblick in der Breite steht gegenüber, dass einzelne Wiki-Seiten in der Tiefe **hervorragend und übersichtlich ausdetailliert** werden können. Der üblicherweise als Vorteil eines (Media)Wiki genannte Aspekt der **einfachen Editierbarkeit** einzelner Wiki-Seite konnte im Rahmen der Untersuchung **jedoch nicht bestätigt werden**. So wurde das Editieren einzelner Wiki-Seiten mittels Bearbeitung des zugehörigen Wiki-Codes (vgl. Abbildung 6-6) als wenig intuitiv und folglich als erhebliche Hürde wahrgenommen. Diese Hürde wurde insbesondere bei strukturellen Anpassungen des modellierten Zielsystems deutlich, da eine notwendige Umstrukturierung mit der Anpassung etlicher Hyperlinks auf verschiedenen Wiki-Seiten verbunden ist und in Summe erhebliche Aufwände erzeugt. Im Zuge der Umstrukturierung des modellierten Zielsystems zeigte sich auch, dass der Informationsgehalt je Wiki-Seite stark unterschiedlich ist. Beispielsweise wurde auf einer Wiki-Seite lediglich eine Anforderung modelliert, wohingegen auf einer anderen Wiki-Seite die Lösung für ein komplettes Teilsystem beschrieben war. Aus Effizienzgründen ist die Abbildung mehrerer Informationen auf ein und derselben Wiki-Seite zu empfehlen, da das Anlegen und Verwalten einer Wiki-Seite einen gewissen administrativen Aufwand erzeugt. Aus Gründen der eindeutigen Zuordnung einer Information zu einer Wiki-Seite und damit verbunden die Möglichkeit einer eindeutigen Modellierung der Eigenschaften und Relationen empfiehlt es sich jedoch, möglichst wenige oder gar nur genau eine Information je Wiki-Seite abzubilden. Im

Zuge der Untersuchung stelle sich heraus, dass dieser Konflikt mit einem MediaWiki-System nicht zufriedenstellen zu lösen ist, da der **Aufwand einer eindeutigen Modellierung den Nutzen klar übersteigt**.

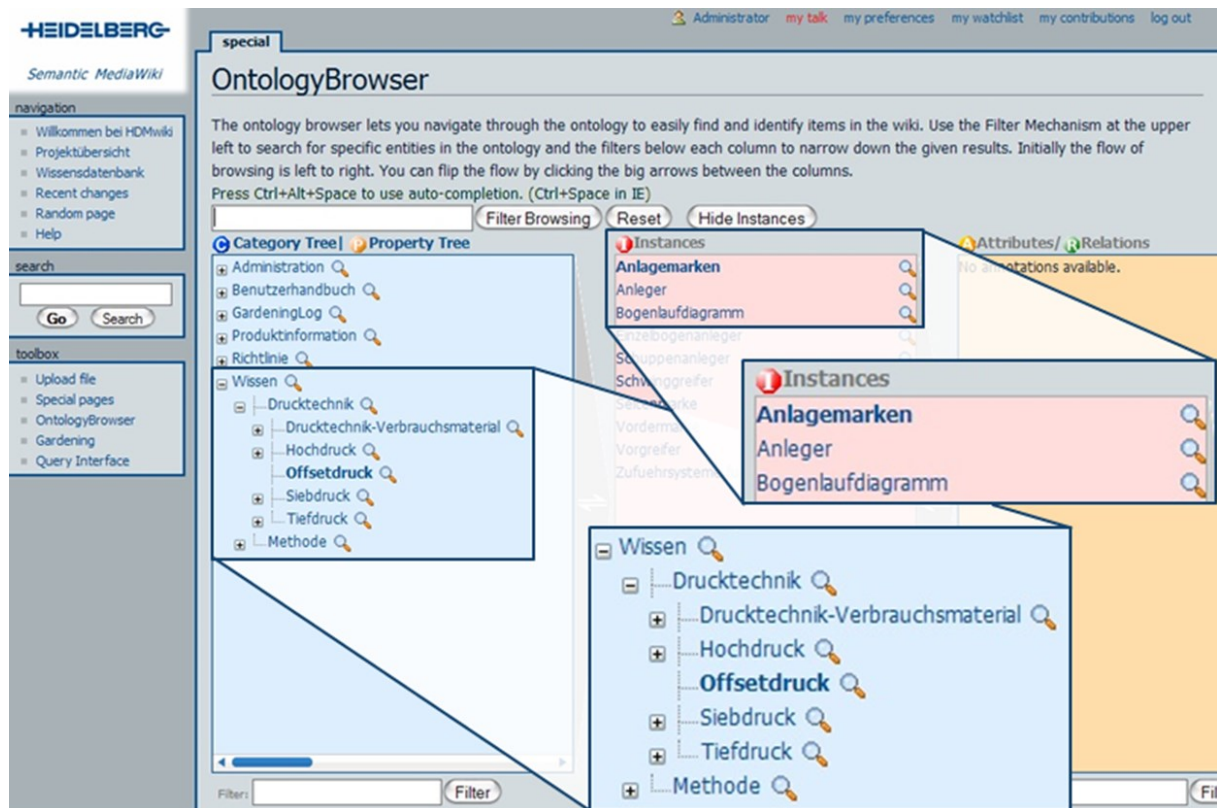


Abbildung 6-7: OntologyBrowser des MediaWiki-Systems

Das **Auffinden von modellierten Informationen** wurde durch das MediaWiki-System auch aufgrund der vorgenommenen Erweiterungen sehr gut unterstützt. So lieferte die **Suchfunktion** (mit Autovervollständigung – vgl. Abbildung 6-5) ebenso relevante Suchergebnisse wie auch das Navigieren durch den **OntologyBrowser** (vgl. Abbildung 6-7). Die Formulierung **semantischer Suchanfragen** gestaltete sich teilweise schwierig, da die zugrundeliegenden Eigenschaften und Relationen in einem Semantic MediaWiki nicht zentral verwaltet werden, sondern auf Basis der dezentralen Verwendung in den jeweiligen Wiki-Seiten gebildet werden (Bottom-Up-Ontologie). Hierdurch wurden zum einen Begrifflichkeiten uneinheitlich verwendet, zum anderen waren die Eigenschaften und Relationen nur teilweise aufeinander abgestimmt. Diese Problematik kann und sollte mittels fest vorgegebener Eigenschaften und Relationen, die durch den Benutzer nicht geändert oder erweitert werden können, abgefangen werden.

6.2 Modellierung von Zielsystemen mittels Mind-Maps³⁶⁸

Die Zielsetzung und Inhalte der zweiten Fallstudie bauen auf den Ergebnissen der ersten Fallstudie auf, wobei gezielte Variationen und Ergänzungen vorgenommen werden. Als wesentlich ist hierbei die Variation des Werkzeugs zur Modellierung des Zielsystems zu betrachten. Durch die Verwendung einer **Mind-Mapping-Software** wird sowohl Einfluss auf die Struktur bzw. Strukturierung des Zielsystems genommen als auch auf dessen inhaltliche Ausgestaltung. Dies ist zum einen darauf zurück zu führen, dass Mind-Maps durch ihre **hierarchische Baumstruktur** Eltern-Kind-Beziehungen ausbilden und somit als spezialisierte Form von Concept-Maps gelten, die wiederum stärker vernetzt sein können. Zum anderen sollen die **Inhalte einer Mind-Map auf einzelne Worte beschränkt** werden, um hierdurch das Assoziationsvermögen zu nutzen bzw. zu fördern, Inhalte grafisch zueinander in Beziehung zu setzen und hierdurch zu strukturieren.³⁶⁹ Für eine Mind-Map charakteristisch sind hierbei folgende Eigenschaften:³⁷⁰

- Der Gegenstand der Aufmerksamkeit kristallisiert sich in einem Zentralbild.
- Die Hauptthemen des Gegenstands strahlen vom Zentralbild wie Äste aus.
- Die Äste enthalten Schlüsselbilder oder Schlüsselworte, die auf einer mit dem Zentralbild verbundenen Linie in Druckbuchstaben geschrieben werden. Themen von untergeordneter Bedeutung werden als Zweige, die mit Ästen höheren Niveaus verbunden sind, dargestellt.
- Die Äste bilden ein Gefüge miteinander verbundener Knotenpunkte.

Im Rahmen der Untersuchung wird die Software MindManager der Firma Mindjet eingesetzt³⁷¹.

6.2.1 Zielsetzung der Untersuchung

Die zweite Fallstudie zur Modellierung von Zielsystemen verfolgt mehrere Ziele, die in weiten Teilen den Zielen der ersten Fallstudie entsprechen bzw. auf den Ergebnissen der ersten Fallstudie aufbauen.

- Explorative Suche nach geeigneten Strukturen des Zielsystems sowie angrenzender Partialmodelle.

Auch im Rahmen der zweiten Fallstudie soll keine Referenzstruktur des Zielsystems vorgegeben werden. Das Ziel ist bewusst nicht die Bestätigung der Strukturen und Partialmodelle, die im Rahmen der ersten Fallstudie identifi-

³⁶⁸ vgl. Albers, Klingler & Ebel 2013

³⁶⁹ vgl. Gronau 2009, S. 27

³⁷⁰ vgl. Buzan & Buzan 2012, S. 59

³⁷¹ vgl. <http://www.mindjet.com>

ziert wurden (vgl. Abschnitt 6.1.). Vielmehr soll diese Fallstudie dazu genutzt werden, weitere praxisrelevante Strukturen des Zielsystems sowie angrenzender Partialmodelle zu identifizieren.

- Untersuchung einer hierarchischen Modellierung des Zielsystems.
Durch die Verwendung einer Mind-Mapping-Software ist eine prinzipiell hierarchische Modellierung des Zielsystems und angrenzender Partialmodelle impliziert. Es gilt zu klären, wie sich dieser Umstand auf die Zielsystemmodellierung bzw. das resultierende Modell auswirkt und inwiefern die netzwerkartigen Strukturen einer Concept-Map erforderlich sind.
- Untersuchung der Eignung von Beurteilungsdimensionen von Zielen als explizite Eigenschaften.
Zur expliziten Charakterisierung von Elementen des Zielsystems sollen die erarbeiteten Beurteilungsdimensionen von Zielen³⁷² verwendet werden. Hierbei soll die explizite Verwendung der Beurteilungsdimensionen sowie der darauf aufbauenden Beurteilungsmatrizen in der praktischen Anwendung untersucht und bewertet werden.
- Berücksichtigung von Entscheidungen (Entscheidungssituation, Entscheidung, Entscheidungsbegründungen und Entscheidungsalternativen).
Im Rahmen der ersten Fallstudie wurde die Bedeutung von Entscheidungen im Produktentstehungsprozess hervorgehoben und eine entsprechende Abbildung der Entscheidungssituation, der Entscheidung, der Entscheidungsbegründung sowie der Entscheidungsalternativen gefordert. Daher soll untersucht werden, wie Entscheidungen in die Zielsystemmodellierung bzw. die Modellierung angrenzender Partialmodelle integriert werden kann. Hierauf aufbauend soll eine Bewertung der Auswirkungen bzw. des Nutzens einer solchen Modellierung erfolgen.
- Berücksichtigung von Änderungen im Zielsystem
Wie bereits in der ersten Fallstudie gezeigt werden konnte, unterliegen das Zielsystemmodell sowie die angrenzenden Partialmodelle im Verlauf eines Produktentstehungsprozesses ständiger Veränderung. Daher soll im Rahmen dieser Fallstudie untersucht werden, in welchem Maß die Aspekte Änderungsautorisation und Änderungsverlauf im Kontext der Zielsystemmodellierung (mit MindMaps) berücksichtigt werden müssen und können.
- Untersuchung der Tauglichkeit des Werkzeugs Mind-Map zur Zielsystemmodellierung

³⁷² vgl. Abschnitt 5.3

Der ursprüngliche Einsatzzweck von Mind-Maps ist die Unterstützung von Kreativitätstechniken, insbesondere zur Strukturierung von Ideen. Hierbei liegt ein wesentlicher Schwerpunkt von Mind-Mapping-Software auf der barrierefreien Erstellung von Inhalten³⁷³. Daher gilt es zu untersuchen, inwiefern Mind-Mapping im Allgemeinen für die Zielsystemmodellierung geeignet ist und inwiefern die gewählte Software-Lösung im Speziellen die Modellierung von Zielsystemen sowie angrenzender Partialmodelle unterstützt.

6.2.2 Rahmenbedingungen der Untersuchung

Die Untersuchung fand abermals bei einem der weltweit führenden Hersteller von Bogenoffset-Druckmaschinen statt. Das untersuchte Entwicklungsprojekt hatte zum Ziel, ein **alternatives Bogenzuführungskonzept zur Serienreife** zu entwickeln. Zu Beginn der Untersuchung lag als Ergebnis eines vorangegangenen Projekts bereits ein Lastenheft vor, das entsprechend den Unternehmensrichtlinien erstellt wurde. Das hieraus **abzuleitende Pflichtenheft** und damit verbunden die konkrete **Ausgestaltung der technischen Lösung** (inkl. der Anpassung an einen bestimmten Maschinentyp) stellte die erste Projektphase dar, die im Rahmen der **sechsmonatigen Untersuchung** im Wesentlichen vollständig begleitet wurde. Die zweite Projektphase hatte planmäßig die Ausgestaltung und Beauftragung des Bogenzuführungssystems zum Inhalt. Daran anschließend sollten im Rahmen der dritten Projektphase die Fertigung, Montage und der Test des Bogenzuführungssystems erfolgen. Projektphase zwei und drei waren hierbei nicht Gegenstand der Untersuchung. Das Projektteam bestand aus **vier Maschinenbauingenieuren** und beinhaltete den Projektmanager sowie den Untersuchenden selbst. Hierbei waren der Projektleiter und ein weiterer Maschinenbauingenieur bereits in der ersten Fallstudie beteiligt. Das Projektteam wurde im Verlauf der Entwicklung von **verschiedenen Experten** angrenzender Disziplinen wie Elektrotechnik, Regelungstechnik und Informatik ergänzt bzw. unterstützt.

Die Mind-Mapping-Software MindManager wurde auf den Rechnern aller permanenten Projektmitglieder installiert, so dass prinzipiell das gesamte Projektteam auf die Inhalte der Mind-Map zugreifen konnte. Hierzu wurde die Mind-Map-Datei an zentraler und prominenter Stelle im Projektlaufwerk abgelegt und allen permanenten Projektmitgliedern Lese- und Schreibrechte eingeräumt. Zu Beginn der Untersuchung wurde vereinbart, dass neue Inhalte nur durch den Projektleiter bzw. den Untersuchenden erstellt werden dürfen. Die übrigen Projektmitglieder sollten die

³⁷³ vgl. Gronau 2009, S. 26

Schreibrechte lediglich für Anmerkungen nutzen, die (spätestens) im Rahmen der wöchentlichen Projektsitzungen diskutiert werden sollten.

6.2.3 Ergebnisse und Interpretation der Untersuchung

Die Ergebnisse der Untersuchung werden entsprechend der formulierten Zielsetzungen strukturiert und interpretiert und im Nachfolgenden dargelegt.

Strukturierung des Zielsystems und angrenzender Partialmodelle

Die Strukturierung des Zielsystems und angrenzender Partialmodelle wurde stark durch die **hierarchische Modellierung mittels Mind-Mapping** geprägt (siehe Abschnitt Hierarchische Modellierung des Zielsystems und angrenzender Partialmodelle). Einige der Partialmodelle, die bereits im Rahmen der ersten Fallstudie (vgl. Abschnitt 6.1) identifiziert werden konnten, wurden auch in der zweiten Fallstudie verwendet. So wurden aus dem bereits bestehenden Lastenheft sowohl **Ziele** als auch **Anforderungen** extrahiert bzw. direkt übernommen. Eine Strukturierung der Ziele und insbesondere der Anforderungen nach Funktionen bzw. das explizite Ableiten einer Funktionsstruktur erfolgte hierbei nicht. Dies kann darauf zurück geführt werden, dass aufgrund des im Vorfeld durchlaufenen Vorentwicklungsprojekts (vgl. Abschnitt 6.1) bereits eine sehr genaue Vorstellung über den Aufbau des Bogenzuführungssystems etabliert werden konnte und auch das Zusammenspiel der disziplinspezifischen Lösungsanteile (Maschinenbau, Elektrotechnik, Regelungstechnik, Informatik) im Wesentlichen durchdrungen war. Daher wurden insbesondere die Objekte in Form von Lösungen und Alternativen entsprechend der **Baustruktur** angeordnet und beschrieben, wohingegen die Ziele und Anforderungen entsprechend der Vorgabe des existierenden Lastenheftes nach **funktionalen** und **nicht-funktionalen Anforderungen** untergliedert war. Hierbei wurden aus den bereits formulierten Anforderungen übergeordnete Ziele hergeleitet, die den Anforderungen in der Hierarchie übergeordnet wurden. Insofern wurden die **Ziele und Anforderungen nicht in zwei getrennten Partialmodellen gepflegt** und entsprechend vernetzt, sondern in einem umfassenden Partialmodell abgebildet, das zwar eine Durchgängigkeit von Zielen zu Anforderungen realisierte, das jedoch keine scharfe Trennung zwischen Zielen und Anforderungen zuließ. Der Entwicklungsprozess mit seinen Phasen, Produktentstehungsaktivitäten, Meilensteinen und Arbeitsergebnissen (Deliverables) sowie Stakeholdern wurde mit Ausnahme von **Entscheidungen** (siehe Abschnitt Modellierung von Entscheidungen) nicht explizit abgebildet.

Im Rahmen der zweiten Fallstudie konnten darüber hinaus zwei weitere Partialmodelle identifiziert werden, die im Kontext einer ganzheitlichen und durchgängigen Zielsystemmodellierung relevant sind.

- **Anwendungsszenarien und Anwendungsfälle**

Im Rahmen der Untersuchung konnte mehrfach die Relevanz explizierter Anwendungsszenarien und Anwendungsfälle aufgezeigt werden. Hierbei beschreiben Anwendungsszenarien und Anwendungsfälle die Interaktion des zu entwickelnden Produkts (Systems) mit dessen (System-)Nutzern. Durch gezieltes Hinterfragen, in welchen Anwendungsszenarien bzw. Anwendungsfällen einzelne Anforderungen des Lastenhefts von besonderer Bedeutung sind, konnten relevante Anwendungsszenarien und Anwendungsfälle identifiziert und hieraus bisher nicht ausreichend berücksichtigte Ziele und Anforderungen abgeleitet werden. Darüber hinaus waren die explizierten Anwendungsszenarien und Anwendungsfälle neben den formulierten Zielen und Anforderungen eine wichtige Quelle zur Definition von Testfällen.

- **Tests (Testfälle, Testumgebungen und Testinterpretation)**

Im Rahmen der Untersuchung wurde ebenfalls die Bedeutung von Tests für die Zielsystemmodellierung deutlich. Tests ermitteln hierbei die Eigenschaften des zu entwickelnden Produkts (Systems) und liefern dadurch Erkenntnisse über das Produkt (System) und die Erfüllung der zuvor definierter Ziele und Anforderungen. So musste auf Basis eines identifizierten Anwendungsszenarios eine bestehende Anforderung des Lastenheftes geringfügig verschärft werden, wodurch die Erfüllung der Anforderung mittels der angedachten Lösung nicht mehr mit ausreichender Sicherheit gewährleistet werden konnte. Aufgrund der Bedeutung dieser Lösung für die Funktionserfüllung des Bogenzuführungssystems musste infolgedessen ein ausreichend sicherer Nachweis geführt werden, dass die Lösung im späteren Produkt in der Lage ist, die verschärfte Anforderung sicher zu erfüllen. Daher musste neben einem geeigneten Testfall auch eine geeignete Testumgebung definiert werden, da das spätere Produkt bzw. ein entsprechender Vorgänger nicht zur Verfügung stand. Hierzu musste insbesondere die Testinterpretation vorausgedacht werden, da andernfalls der Testfall bzw. die Testumgebung keine verwertbaren Erkenntnisse geliefert hätte.

Aufgrund der **Relevanz von Anwendungsfällen und Tests für die Zielsystemmodellierung in der interdisziplinären Produktentstehung** werden diese und unmittelbar angrenzende Begrifflichkeiten als Aspekte des **XiL-Frameworks** nach ALBERS³⁷⁴ definiert.

³⁷⁴ vgl. Albers, Düser & Ott 2008, Albers & Düser 2009 und Albers, Behrendt & Ott 2010

Definition 6-1: Anwendungsfall³⁷⁵

Ein Anwendungsfall beschreibt genau ein Hauptszenario sowie zugehörige Alternativ- und Ausnahmeszenarien, die ein System bei der Interaktion mit einem Systemnutzer (Personen und Systeme) ausführt, um einen gewünschten Mehrwert zu erreichen.

Definition 6-2: Anwendungsszenario³⁷⁵

Ein Anwendungsszenario beschreibt Interaktionen zwischen einem System, den Systemnutzern (Personen und Systeme) und der Systemumgebung sowie hierfür relevante Kontextinformationen.

Ein Anwendungsszenario tritt entweder als Haupt-, Alternativ- oder Ausnahmeszenario auf.

Definition 6-3: Hauptszenario³⁷⁵

Ein Hauptszenario beschreibt eine Interaktionsfolge, die normalerweise ausgeführt wird, um eines oder mehrere mit dem Szenario assoziierte Ziele zu erfüllen.

Definition 6-4: Alternativszenario³⁷⁵

Ein Alternativszenario beschreibt eine Interaktionsfolge, die alternativ zu einem Hauptszenario ausgeführt wird, jedoch die gleichen Ziele erfüllt.

Definition 6-5: Ausnahmeszenario³⁷⁵

Ein Ausnahmeszenario beschreibt eine Interaktionsfolge, die ausgeführt wird, wenn in einem anderen Szenario ein Ereignis eintritt, das die Erfüllung eines oder mehrerer mit dem Szenario assoziierter Ziele verhindert.

³⁷⁵ vgl. Pohl 2007, S. 127-140

Definition 6-6: Test

Ein Test ermittelt Systemeigenschaften eines Systems unter Test (SUT) und liefert Erkenntnisse über das System, insbesondere ob das System zuvor definierte Ziele, Anforderungen oder Hypothesen ganz, teilweise oder nicht erfüllt.³⁷⁶

Ein Test umfasst stets einen Testfall, eine Testumgebung und eine Testinterpretation.

Definition 6-7: Testfall³⁷⁷

Ein Testfall beschreibt Randbedingungen, Eingangsgrößen und das erwartete Systemverhalten eines Systems unter Test.³⁷⁸

Definition 6-8: Testumgebung³⁷⁹

Eine Testumgebung beinhaltet die Gesamtheit aller physischen und virtuellen Modelle bzw. Originale, die notwendig sind, um einen oder mehrere Testfälle durchzuführen und das erwartete Systemverhalten zu erfassen.

Definition 6-9: Testinterpretation

Eine Testinterpretation überführt das in einem durchgeführten Testfall erfasste Systemverhalten in eine oder mehrere zugrundeliegende Systemeigenschaften und liefert Erkenntnisse über das System und den Test, insbesondere ob zuvor definierte Ziele, Anforderungen oder Hypothesen ganz, teilweise oder nicht erfüllt werden.

Die Untersuchungen im Rahmen der zweiten Fallstudie haben gezeigt, dass die Bildung von Anwendungs- und Testfällen in hohem Maße mit der Zielsystemmodellierung verknüpft ist (vgl. Abbildung 6-8).

³⁷⁶ Liegt der Fokus im Besonderen auf der Überprüfung von Hypothesen ist der Begriff Experiment zu favorisieren (vgl. Freudenmann 2014, S. 38).

³⁷⁷ vgl. Alt 2009, S. 25

³⁷⁸ Branchen- bzw. disziplinabhängig wird der Begriff Manöver häufig auch synonym zu Testfall verwendet (vgl. Freudenmann 2014, S. 128).

³⁷⁹ vgl. Alt 2009, S. 36

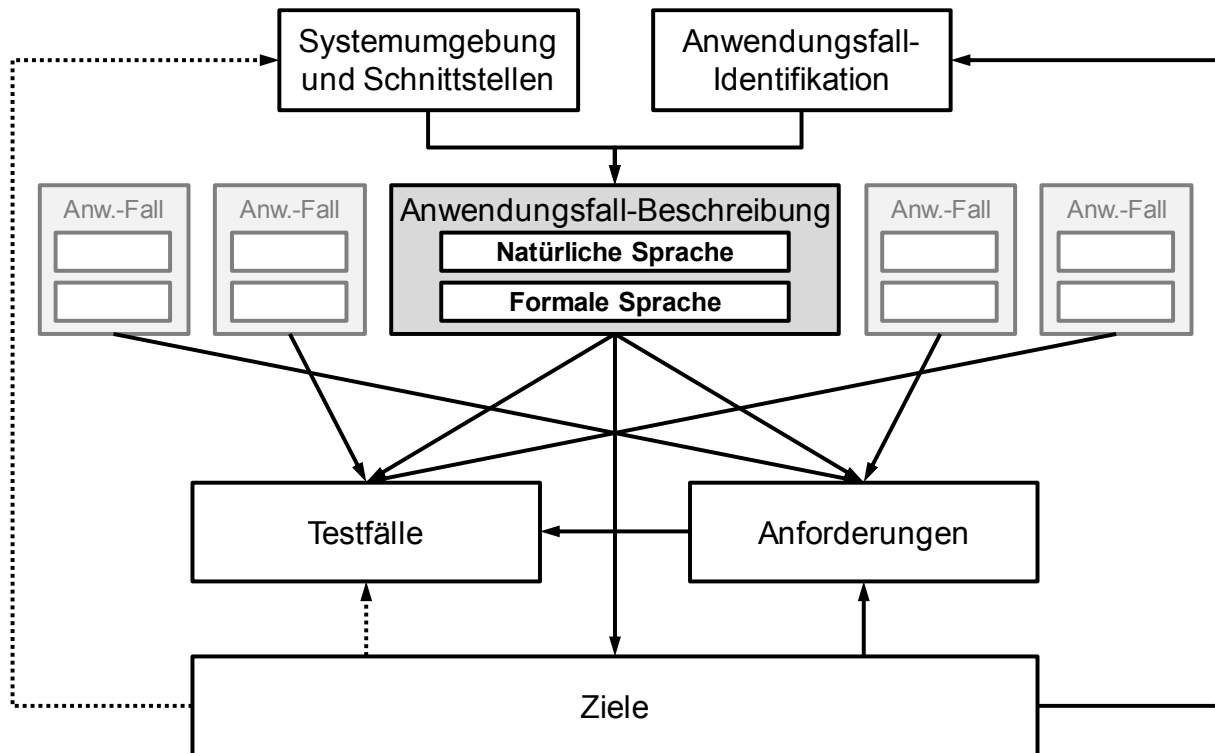


Abbildung 6-8: Bildung von Anwendungs- und Testfällen³⁸⁰

Basierend auf einer Anwendungsfallidentifikation und einer hierauf aufbauenden Festlegung der Systemumgebung (und damit der Systemgrenze) sowie der Schnittstellen zwischen dem betrachteten System und der Systemumgebung kann ein Anwendungsfall beschrieben werden. Hierbei erfolgt die Beschreibung entweder in **natürlicher oder formaler Sprache** (Sequenz-, Zustands- und Aktivitätsdiagramme). Wird ein Anwendungsfall in natürlicher Sprache formuliert, sollten nachfolgende Aspekte adressiert werden³⁸¹:

- Name
- Akteure
- Auslösendes Ereignis
- Kurzbeschreibung
- Vorbedingung
- Hauptszenario
- Trigger der Alternativ- und Ausnahmeszenarien
- Alternativ- und Ausnahmeszenarien
- Nachbedingung

³⁸⁰ eigene Darstellung in Anlehnung an Rupp & Queins 2003

³⁸¹ vgl. Rupp & Queins 2003

Auf Basis formulierter Anwendungsfälle können einerseits zu erreichende Ziele sowie zu erfüllende Anforderungen identifiziert bzw. abgeleitet werden. Andererseits bilden Anwendungsfälle neben Zielen und Anforderungen eine wesentliche Basis zur Definition von Testfällen. Hierbei müssen Anwendungsfälle in der Regel in Testfälle überführt werden. Der Grenzfall, dass sich Anwendungs- und Testfall exakt entsprechen bildet eine Ausnahme.

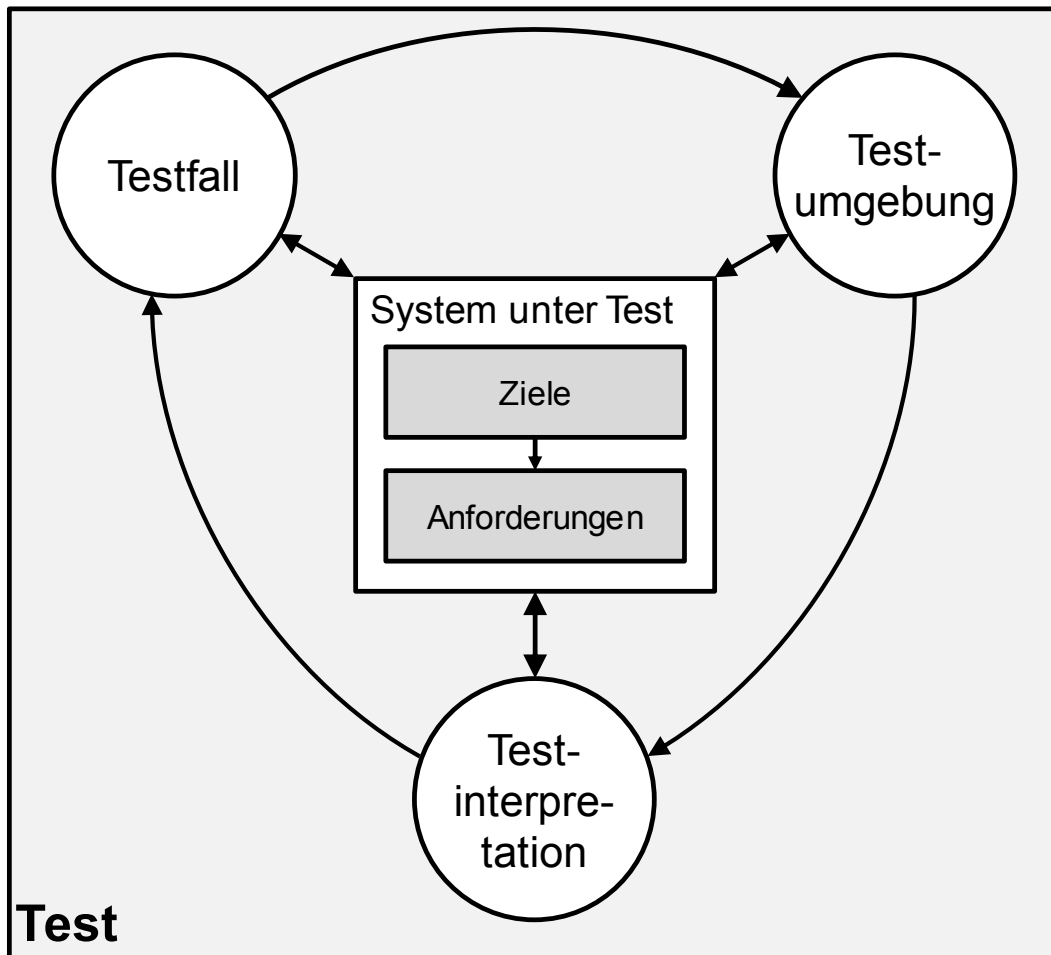


Abbildung 6-9: Interaktion zwischen Test, Zielen und Anforderungen

Die Durchführung von Tests zum Zwecke der **Verifizierung** bzw. **Validierung** eines Systems unter Test steht in enger Wechselwirkung mit den definierten bzw. zu definierenden Zielen und Anforderungen (vgl. Abbildung 6-9). Einerseits müssen ein Testfall, die durchführende Testumgebung sowie die Testinterpretation iterativ aufeinander und auf die durch das System zu erfüllenden Ziele bzw. Anforderungen abgestimmt werden. Hierzu muss eine Vielzahl von Faktoren berücksichtigt werden, wodurch die Definition zielführender Tests im Kontext der Entwicklung mechatronischer Produkte nicht trivial ist. Andererseits resultieren etliche Ziele und Anforderungen aus extern oder intern definierten Tests, die das zu entwickelnde Gesamtsystem oder Teilsysteme hiervon erfüllen müssen. Im Falle bereits bestehender Tests muss hierbei eine **Nachvollziehbarkeit** zwischen den einzelnen Tests, den hieraus

abgeleiteten Anforderungen sowie den ursprünglichen Zielen des Tests hergestellt werden. Gelingt dies nicht, ist lediglich eine Verifizierung des Systems unter Test auf Basis der abgeleiteten Anforderungen möglich. Für den Fall, dass zu einzelnen Zielen und Anforderungen noch keine Tests existieren, kann und sollte die Testdefinition (Testfall, Testumgebung und Testinterpretation) frühzeitig und in enger Abstimmung mit der Definition der Ziele und Anforderung erfolgen. Hierdurch kann die Qualität der Ziele und Anforderungen sowie die Qualität der Tests erheblich gesteigert werden und damit ein wesentlicher Beitrag zur erfolgreichen Entwicklung mechatronischer Produkte geleistet werden.

Hierarchische Modellierung des Zielsystems und angrenzender Partialmodelle

Die prinzipiell **hierarchische Modellierung** des Zielsystems wurde durch alle beteiligten Entwickler befürwortet und akzeptiert (vgl. Abbildung 6-10). Hierbei ließ sich das bestehende, hierarchisch strukturierte Lastenheft unmittelbar in der Mind-Map abbilden, wodurch die Akzeptanz der Modellierung von Zielsystemen mittels Mind-Maps aufgrund der unmittelbaren Abbildung bekannter Strukturen und Inhalte gesteigert werden konnte. Mit einer hierarchischen Modellierung des Zielsystems geht unmittelbar einher, dass neue Elemente innerhalb der bestehenden Hierarchie **eindeutig zugeordnet** werden müssen. Dies wurde prinzipiell als vorteilhaft empfunden, da hierdurch neue Elemente des Zielsystems nicht gedankenlos, kontextfrei und damit wirkungslos dem bereits bestehenden Zielsystem hinzugefügt werden können. Vielmehr mussten dem Zielsystem neue Elemente bewusst und oftmals im Diskurs hinzugefügt werden, da die Zuordnung zur bestehenden Hierarchie in vielen Fällen nicht eindeutig zu beantworten war. Dies kann zum einen auf die unterschiedlichen Perspektiven der beteiligten Entwickler zurückgeführt werden, die je nach individuellem Bedarf teils unterschiedlich strukturierte Hierarchien gleicher Inhalte ausprägen würden (**Sichten**). Zum anderen muss dies darauf zurückgeführt werden, dass einzelne **Elemente tatsächlich mehrfach im Zielsystem** Verwendung finden müssen (**Mehrfachverwendung**). Beiden Aspekten kann mit einer **rein hierarchischen Modellierung** (mittels Mind-Maps) **nicht Rechnung getragen werden**, was von den beteiligten Entwicklern als **klarer Nachteil** benannt wurde, da diese Nachteile auch bei einer rein dokumentenbasierten Modellierung von Zielsystemen existieren.

Zusätzlich zur rein hierarchischen Modellierung des Zielsystems wurden die Möglichkeiten genutzt, Relationen zwischen einzelnen Elementen des Zielsystems zu etablieren (vgl. Abbildung 6-10). Hierbei wurden im Wesentlichen Verträglichkeitsrelationen (begünstigend, hemmend) verwendet, die zu einem leicht gesteigerten Zielsystemverständnis beitragen konnten. Es zeigte sich jedoch, dass der Mehrwert der Relationen erst dann entstehen kann, wenn sie zu einer essenziellen Verbesserung der Nachvollziehbarkeit beitragen. Hierbei wird über die Hierarchie

bereits ein Teil der vertikalen Nachvollziehbarkeit³⁸², d.h. der Nachvollziehbarkeit innerhalb eines Partialmodells, realisiert. Diagonalbeziehungen³⁸³ innerhalb eines Partialmodells können über eine reine Hierarchie jedoch nicht abgebildet werden. Dies gilt ebenfalls für die horizontale Nachvollziehbarkeit³⁸² zwischen Elementen unterschiedlicher Partialmodelle. Für beide Fälle **müssen Relationen bereitgestellt werden, die eine bidirektionale Nachvollziehbarkeit ermöglichen** (Pre-Requirements-Specification-Traceability alias Backward-Traceability bzw. Rückwärts-Nachvollziehbarkeit sowie Post-Requirements-Specification-Traceability alias Forward-Traceability bzw. Vorwärts-Nachvollziehbarkeit³⁸⁴).

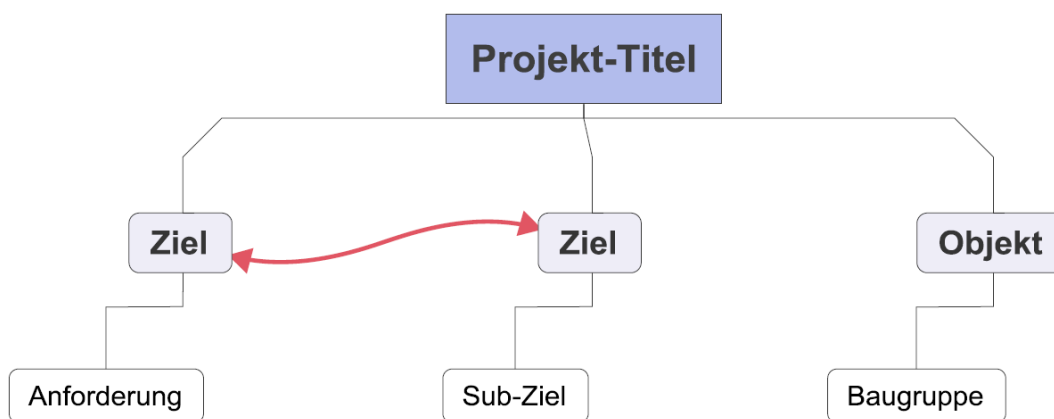


Abbildung 6-10: Hierarchische Modellierung des Zielsystems mittels Mind-Maps

Eignung von Beurteilungsdimensionen von Zielen

Im Rahmen der Fallstudie wurden zur expliziten Charakterisierung von Elementen des Zielsystems die Beurteilungsdimensionen von Zielen³⁸⁵ verwendet (vgl. Abbildung 6-11).

³⁸² vgl. Schuh 2012, S. 184

³⁸³ vgl. Abbildung 2-31

³⁸⁴ vgl. Rupp 2009, S. 401

³⁸⁵ vgl. Abschnitt 5.3

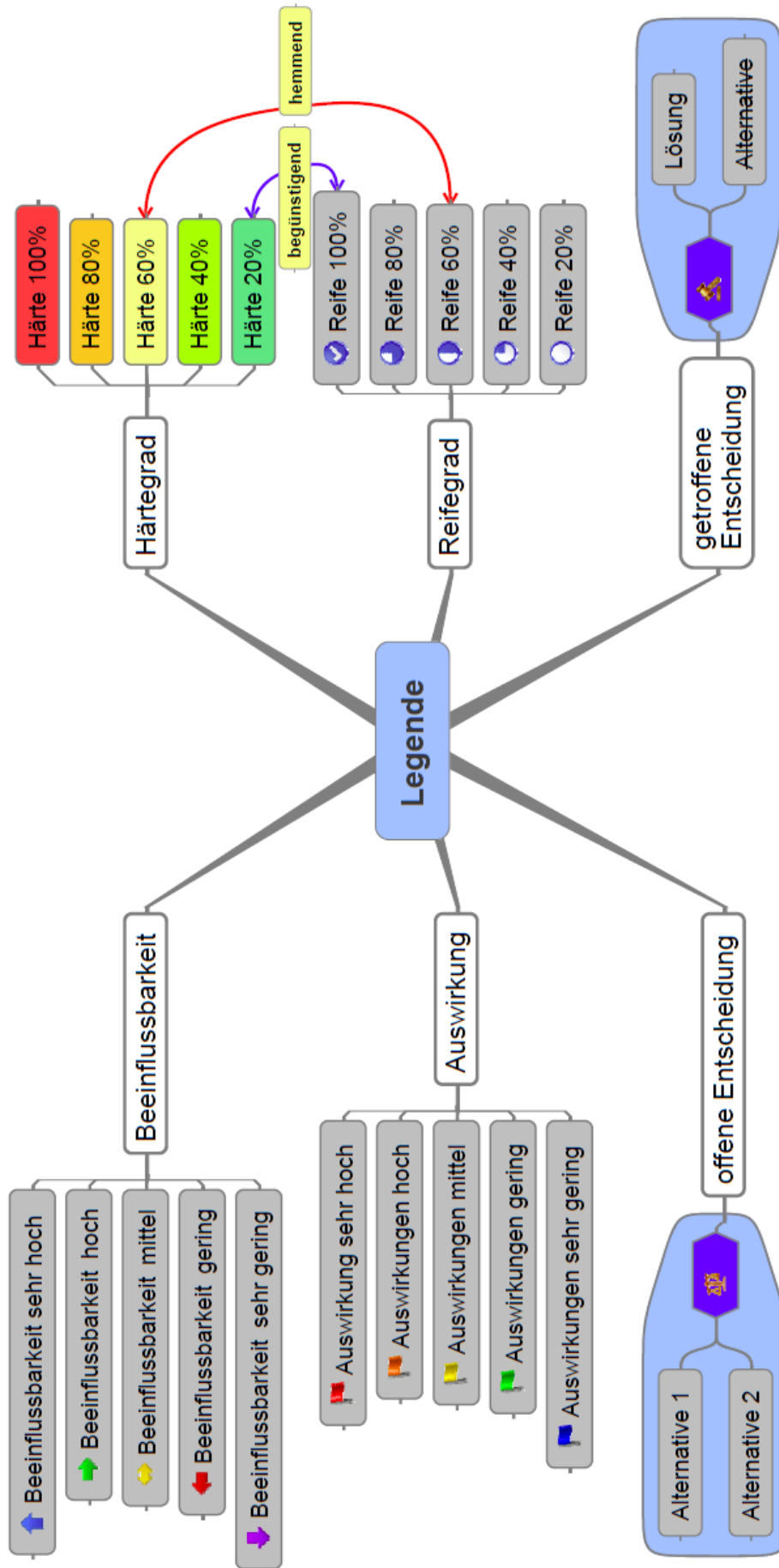


Abbildung 6-11: Abbildung von Beurteilungsdimensionen von Zielen und Entscheidungen

Hierzu wurden die involvierten Entwickler in einem ersten Schritt mit den vier Beurteilungsdimensionen von Zielen und den darauf aufbauenden Matrizen von Beurteilungsdimensionen konfrontiert, mit dem Ziel, gemeinsame mentale Modelle³⁸⁶ als elementare Basis einer gemeinsamen, expliziten Verwendung der Beurteilungsdimensionen zu schaffen. Hierbei wurden die Beurteilungsdimensionen und die darauf aufbauenden Matrizen in weiten Teilen als **nutzenstiftende Perzeptions- und Kogitationsmodelle**³⁸⁷ bewertet. Lediglich die Differenzierung des Reife- und Härtegrads eines Ziels fiel den involvierten Entwicklern schwer, da nach ihrem Verständnis der Härtegrad stets mit dem Reifegrad steigt. Dieses Verständnis kann darauf zurückgeführt werden, dass Entwickler (Ingenieure) stets bemüht sind, das Entwicklungsrisiko so gering wie möglich zu halten und folglich die Änderbarkeit eines Ziels erst dann einschränken, wenn die Unsicherheit des Ziels entsprechend niedrig ist. Folglich wurde zur expliziten Charakterisierung der Elemente des Zielsystems die **Verwendung des Reifegrads als wenig nutzenstiftend** empfunden, da der **Härtegrad als sehr nützliche Beurteilungsdimension** bereits die notwendige Information explizit zum Ausdruck bringt. Weiterhin wurden die **Auswirkungen eines Ziels als explizite Beurteilungsdimension sehr positiv wahrgenommen**, da aufgrund der (teils differierenden) Einschätzungen über die Auswirkung eines Ziels relevante Wechselwirkungen identifiziert werden konnten, die bis dato unbekannt waren. Darüber hinaus konnte mittels der Einschätzung der Auswirkungen eines Ziels die Aufmerksamkeit auf änderungssensitive Ziele gelenkt werden, ohne die Vielzahl an Wechselwirkungen explizit modellieren zu müssen. Die **Beeinflussbarkeit eines Ziels wurde zwar als eine prinzipiell wertvolle Beurteilungsdimension** eines Ziels empfunden, jedoch wurde im Rahmen der Zielsystemmodellierung **die Explikation als nicht sinnvoll** erachtet. Dies kann dadurch begründet werden, dass die Beeinflussbarkeit eines Ziels einerseits keine Eigenschaft des Ziels selbst darstellt, sondern sich erst im Zusammenspiel mit einzelnen Akteuren ergibt, d.h. dass die Beeinflussbarkeit eines Ziels personenabhängig ist. Zum anderen sind die einzelnen Akteure in der Regel nicht bereit, ihre tatsächliche (informelle) Beeinflussbarkeit eines Ziels offen zu explizieren bzw. zu dokumentieren. Insofern wird die Beeinflussbarkeit eines Ziels als explizite Eigenschaft auf die **Autorisation** reduziert. Auf Basis der explizierten Beurteilungsdimensionen sowie einer Filterfunktionalität der verwendeten Mind-Mapping-Software MindManager konnten Matrizen von Beurteilungsdimensionen gebildet werden. Hierdurch konnten zusätzliche Informationen aus dem bestehenden Zielsystemmo-

³⁸⁶ vgl. Abschnitt 2.4.1

³⁸⁷ vgl. Abbildung 2-5

dell extrahiert werden, die insbesondere im Fall der **Definitionsmatrix** (Auswirkung eines Ziels über dessen Härtegrad) **wertvolle Hinweise für die Schwerpunktlegerung der Validierungsaktivitäten** geliefert hat.

Modellierung von Entscheidungen

Begründungen sind ein wesentlicher Bestandteil des Zielsystems und liegen in der Regel (zumindest) in Entscheidungssituationen vor. Die Dokumentation bzw. Modellierung von Entscheidungssituationen sowie der zugehörigen Entscheidung, Entscheidungsbegründungen und Entscheidungsalternativen ist daher für die Zielsystemmodellierung von besonderer Bedeutung. Im Rahmen der zweiten Fallstudie wurden Entscheidungen durch die folgenden Aspekte abgebildet (vgl. Abbildung 6-11):

- Status der Entscheidung (offen oder getroffen)
- Begründung der Entscheidung (Notiz)
- Alternative Lösungen bzw. gewählte und nicht gewählte Lösungen
- Entscheidungsrelevante Dokumente (Hyperlink)

Dabei zeigte sich, dass die Modellierung von Entscheidungen nicht nur im Nachgang von Entscheidungssituationen nutzenstiftend ist, sondern bereits in deren Vorfeld einen wesentlichen Mehrwert bieten kann. So konnte durch die **Modellierung offener Entscheidungen ein Überblick über relevante Definitionslücken** und damit über bestehende Unsicherheiten³⁸⁸ geschaffen werden. Hierbei wurde auch deutlich, dass offene Entscheidungen die Komplexität im Zielsystem erheblich erhöhen, insbesondere wenn einzelne offene Entscheidungen sich gegenseitig beeinflussen. Somit trägt die **Modellierung offener Entscheidungen dazu bei, erfolgreich mit der Komplexität im Zielsystem umzugehen**. Weiterhin konnte der Status einer offenen Entscheidung im Vorfeld der Entscheidungssituation dazu genutzt werden, alternative Lösungen zu sammeln und damit verknüpfte Informationen abzubilden. Hierdurch konnten **Entscheidungssituationen optimal vorbereitet und unterstützt** werden. Noch innerhalb einzelner Entscheidungssituationen wurden die wesentlichen Aspekte der Entscheidungsbegründung notiert, um diese im Anschluss entsprechend auszuformulieren. In diesem Zuge zeigte sich, dass der Aufwand zur **(ausführlichen) Dokumentation von Entscheidungen hinsichtlich Aufwand-zu-Nutzen kritisch** gesehen wird. Dies wurde dadurch begründet, dass nur die Entscheidungen nochmals genauer betrachtet werden, die sich als fehlerhaft herausgestellt haben werden (aufgrund von Fehlern oder geänderter Randbedingungen). Da dies nur bei einem Bruchteil der Entscheidungen

³⁸⁸ vgl. Abschnitt 2.3.2

der Fall sei, ist ein hoher Dokumentationsaufwand für jede Entscheidung nur schwer zu vermitteln. Daher muss ein wesentliches Augenmerk auf die zukünftige **Verfügbarkeit von entscheidungsrelevanten Dokumenten** gelegt werden, die für eine Entscheidungssituation vorbereitet und genutzt wurden. Da mit der Relevanz einer Entscheidung in der Regel auch die Qualität der entscheidungsrelevanten Dokumente steigt, ist die Qualität der Entscheidungsbegründung stets in etwa der Relevanz der Entscheidung angemessen. Hierdurch kann der Aufwand zur Entscheidungsbegründung einer jeden Entscheidung auf konstant akzeptablen Niveau gehalten werden. Die **Modellierung von alternativen Lösungen** im Vorfeld einer Entscheidungssituation, sollte ebenfalls für die Entscheidungsdokumentation genutzt werden. Dies ist zum einen sinnvoll, um im Falle zukünftiger Komplikationen stets die möglichen Alternativen präsent zu halten. Zum anderen ist dies erforderlich, um Änderungen im Zielsystem, die sich aufgrund neuer Erkenntnisse ergeben haben, nachvollziehen zu können.

Berücksichtigung von Änderungen im Zielsystem

Auch in der zweiten Fallstudie hat sich gezeigt, dass im Verlauf eines Produktentstehungsprozesses das **Zielsystem stetig konkretisiert** wird. So wurden dem Zielsystem ab dem Zeitpunkt der Lastenhefterstellung etliche neue Elemente hinzugefügt sowie bestehende Elemente verfeinert bzw. geändert. Hierbei konnten zweierlei Bedarfe klar identifiziert werden. Zum einen muss es möglich sein, **spezifische Änderungsautorisationen für jedes einzelne Element des Zielsystems** zu definieren. Es genügt nicht, auf Basis der Ordner- bzw. Dateifreigabe des Computer-Betriebssystems (Windows) die Mind-Map-Datei vollumfänglich oder überhaupt nicht für Änderungen freizugeben, da innerhalb einer Mind-Map oftmals mehrere Verantwortlichkeiten vorliegen. Hierbei hat sich gezeigt, dass eine Regelung auf Basis von Absprachen langfristig nicht zielführend ist. Eine Rechtevergabe, die auf Datei-Ebene ansetzt, kann daher nur erfolgreich sein, wenn sämtliche Änderungen ausschließlich von einer zentralen Stelle/Person vorgenommen werden. Zum anderen muss es möglich sein, **Änderungen im Zielsystem zu bemerken und nachvollziehen zu können**. Neben der Realisierung einer Nachvollziehbarkeit von Änderungen besteht hierbei eine wesentliche Herausforderung darin, relevante Änderungen im vernetzten Zielsystem für die Entwickler sichtbar zu machen. Hierbei kann die Relevanz beispielsweise über den Abstand eines geänderten Elements entlang der Hierarchie und/oder Relationen im Zielsystem zu einem betrachteten bzw. verantworteten Element erfolgen.

Tauglichkeit einer Mind-Map zur Zielsystemmodellierung

Im Rahmen der zweiten Fallstudie hat sich gezeigt, dass ein Zielsystem prinzipiell mittels einer Mind-Map modelliert werden kann (vgl. Abbildung 6-12). Eine wesentliche Stärke von Mind-Maps besteht darin, dass **bestehende Lasten- und**

Pflichtenhefte bzw. deren Strukturen unmittelbar und intuitiv abgebildet werden können. Dem einzelnen Entwickler ist es hierbei ohne nennenswerte Hürde möglich, ihm bekannte Inhalte in eine neue Form zu überführen (eindimensionaler, linearer Text im Lasten- und Pflichtenheft im Vergleich zur zweidimensionalen Baumstruktur). Die gewählte Mind-Mapping-Software MindManager hatte hierbei sicherlich einen erheblichen Anteil an der intuitiven und flexiblen Art der Modellierung des Zielsystems. Hervorzuheben sind zum einen das **unkomplizierte Hinzufügen neuer Inhalte** sowie zum anderen die **effiziente Anpassbarkeit der Struktur** des Zielsystems. Diesen Vorteilen sind jedoch auch Nachteile gegenüberzustellen, die prinzipbedingt mit einer Modellierung von Zielsystemen mittels Mind-Maps einhergehen. Hierbei ist in erster Linie der **Verlust an Übersichtlichkeit** zu nennen, der bereits bei rund 100 modellierten Elementen eintritt. Der angenommene Vorteil einer Mind-Map, die Inhalte des Zielsystems effizient erfassen zu können, bestätigte sich nicht. Auch bei einer Mind-Map müssen die Inhalte eines praxisrelevanten Zielsystems explorativ erfasst werden. Die Eigenschaften einer Mind-Map wirken hierbei jedoch stark unterstützend. Zu nennen ist hierbei zum einen, dass die jeweiligen **Einträge in der Mind-Map kurz und prägnant formuliert** sein sollen und damit schnell erfassbar sind. Zum anderen ist neben dem jeweils explorierten Element auch dessen **Umgebung peripher sichtbar**, wodurch das Element stets in seinem Kontext wahrgenommen wird. Der Forderung nach einer kurzen und prägnanten Formulierung der Einträge konnte jedoch nicht immer entsprochen werden, da im Rahmen der Modellierung die **Zielsysteminhalte nicht immer derart verkürzt** werden können. Daher musste oft die **nicht leistungsfähige Notiz-Funktion** einzelner Einträge genutzt werden, wodurch die Explorierbarkeit der Mind-Map erheblich verschlechtert wurde. Nachteilig auf die Übersichtlichkeit der Mind-Map wirkte sich zudem aus, dass die zur Modellierung eines Zielsystems unbedingt **erforderlichen Relationen nur allumfassend aktiviert bzw. deaktiviert** werden können. Jedoch würden auch Relationen, die nur ausgehend von einem selektierten Element erscheinen, lediglich einen begrenzten Mehrwert liefern, da die in **Beziehung stehenden Elemente in der Regel nicht direkt sichtbar** sind. Mit der verwendeten Mind-Mapping-Software MindManager konnten darüber hinaus der geforderten Bedarfe hinsichtlich der **Berücksichtigung von Änderungen im Zielsystem nicht befriedigend** bedient werden.

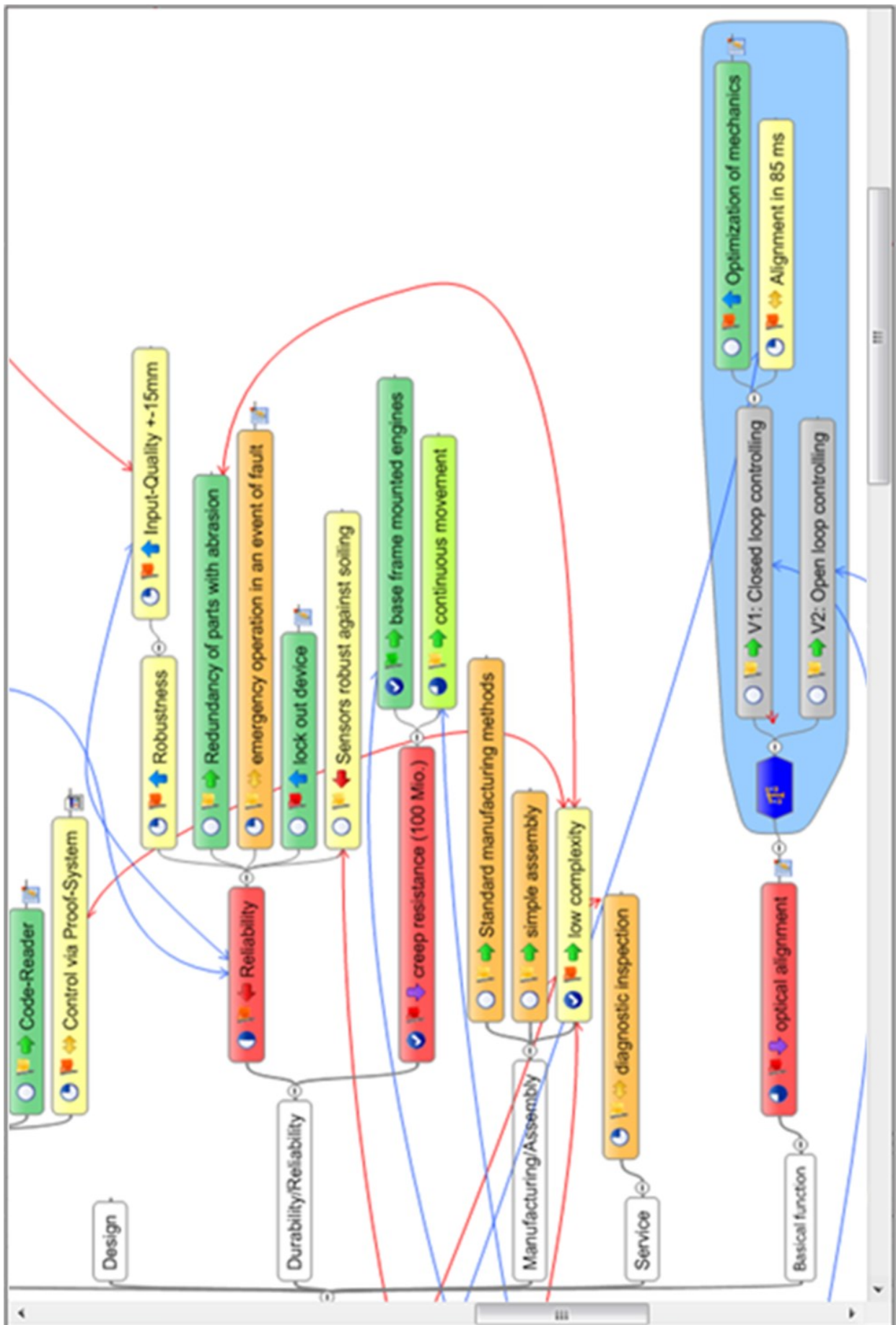


Abbildung 6-12: Modellierung von Zielsystemen mittels Mind-Maps

6.3 Erkenntnisse aus den Untersuchungen

Im Rahmen der Untersuchungen zur Modellierung von Zielsystemen in der industriellen Praxis konnten wesentliche Erkenntnissen gewonnen werden, die zur Beantwortung der formulierten Forschungsfragen unmittelbar beitragen³⁸⁹. Da diese Erkenntnisse eine zentrale Grundlage für die Methode zur Modellierung von Zielsystemen³⁹⁰ bilden, werden die Erkenntnisse aus den Untersuchungen im Folgenden nochmals verdichtet und strukturiert dargelegt.

Erforderliche Partialmodelle für die Modellierung von Zielsystemen

Die Untersuchungen haben gezeigt, dass für eine ganzheitliche und durchgängige Modellierung von Zielsystemen eine Reihe angrenzender Partialmodelle berücksichtigt werden müssen. Hierbei wurde deutlich, dass neben Aspekten des Ziel- und Objektsystems **insbesondere auch Aspekte des Handlungssystems** (Prozesse, Akteure, Ressourcen) abgebildet werden müssen³⁹¹. Folgende Partialmodelle sind entsprechend der Erkenntnisse aus den Untersuchungen für die Modellierung von Zielsystemen erforderlich:

- **Ziele**
- **Anforderungen**
- **Anwendungsfälle**
- **Funktionen (Funktionsstruktur)**
- **Gestalt/Implementation (Baustruktur)**
- **Tests (Testfälle und Testumgebung)**
- **Stakeholder (Akteure)**
- **Phasen und Produktentstehungsaktivitäten**
- **Meilensteine und Arbeitsergebnisse**

Hierbei können die jeweiligen Originale einzelner Partialmodelle derart verkürzt werden, wie es zum Zwecke der ganzheitlichen und durchgängigen Modellierung des Zielsystems erforderlich ist. Beispielsweise ist es oftmals nicht erforderlich, dass Modelle des Partialmodells Gestalt/Implementierung sämtliche Informationen beinhalten, die eine Fertigung der Modelle ermöglichen würden. Jedoch könnte es zweckdienlich sein, beispielsweise die Masse eines (Teil-)Systems abzubilden.

Weiterhin zeigte sich die besondere Bedeutung von Entscheidungen im Produktentstehungsprozess für die Zielsystemmodellierung³⁹², insbesondere hinsichtlich der

³⁸⁹ vgl. Abschnitt 4.1

³⁹⁰ vgl. Kapitel 7

³⁹¹ vgl. Abschnitt 2.2.1 und Abbildung 2-33

³⁹² vgl. Albers, Muschik & Ebel 2010

Begründung modellierter Elemente des Zielsystems. Entscheidungen werden hierbei innerhalb aller identifizierter Partialmodelle getroffen, so dass

- **Entscheidungen,**
- **Begründungen** und
- **relevante Dokumente**

als Elemente in allen Partialmodellen auftreten können.

Hierbei müssen die identifizierten Partialmodelle sowie geeignete Strukturen, Relationen und Attribute bereits zu Beginn eines Produktentstehungsprojekts in Form einer Zielsystem-Referenzstruktur zur Verfügung stehen, da andernfalls keine ausreichende Unterstützung und damit keine individuelle Akzeptanz seitens der Entwickler realisiert werden kann.

Struktur, Relationen und Attribute des Zielsystems

Die Untersuchungen haben ergeben, dass eine **primär hierarchische Strukturierung** des Zielsystems sowie angrenzender Partialmodelle eine hohe Akzeptanz erreichen kann. Hierbei muss es dem einzelnen Entwickler jedoch möglich sein, **individuelle Perspektiven** (Sichten) auf die Hierarchie bilden zu können, ohne dabei Redundanzen zu erzeugen. Darüber hinaus muss auch eine **mehrfache Verwendung** einzelner Elemente des Zielsystems an unterschiedlichen Positionen innerhalb der Hierarchie redundanzfrei möglich sein. In Verbindung mit **geeigneten Relationen** zwischen Elementen verschiedener bzw. gleicher Partialmodelle kann hierdurch eine bidirektionale Nachvollziehbarkeit im Zielsystem realisiert werden. Werden die jeweiligen Elemente zusätzlich mit **geeigneten Attributen** versehen, beispielsweise auf Basis der Beurteilungsdimensionen von Zielen, die als nutzenstiftendes Perzeptions- und Kogitationsmodell bewertet wurden, ist die Grundlage geschaffen, entscheidungsrelevante Informationen aus dem modellierten Zielsystem extrahieren zu können.

Extraktion entscheidungsrelevanter Informationen

Semantische Funktionalitäten sind ein wesentlicher Schlüssel, um auf Basis eines explizit modellierten Zielsystems Schlüsse zu ziehen, die für einen Entwickler nicht auf der Hand liegen. Hierbei haben beide Untersuchungen gezeigt, dass ein Computer auf Basis geeigneter und modellierter Relationen und Attribute in der Lage ist, praxisrelevante Suchanfragen mit entscheidungsrelevanten Antworten zu bedienen. Dem Entwickler darf dabei die hierfür zugrunde zu legende **Ontologie nicht in einer abstrakten Form** präsentiert werden, sondern muss auf Basis einer **geschlossenen Liste** relevanter Attribute und Relationen zu Verfügung gestellt werden. Schlussendlich muss der Entwickler in die Lage versetzt werden, mittels einer **intuitiv bedienbaren Suchfunktion**, die wesentliche Aspekte der Ontologie integriert, relevante Suchanfragen und damit Suchtreffer zu generieren.

Entwicklung des Zielsystems

Im Rahmen der Untersuchungen zur Modellierung von Zielsystemen zeigte sich, dass die einzelnen Partialmodelle nicht sequentiell und losgelöst voneinander entwickelt wurden, sondern vielmehr **iterativ und in starker Wechselwirkung**. Diese Beobachtung lässt sich unmittelbar durch den zugrundeliegenden Produktentstehungsprozess begründen, der trotz ursprünglich definierter und sequenziell aufeinander aufbauender Phasen (Stages) und Meilensteine (Gates) vielmehr wie ein **Spiral-Prozess** ausgestaltet war. Eine methodische Unterstützung muss dem Entwickler folglich dabei behilflich sein, Zielsysteme und angrenzende Partialmodelle im Verlauf des Produktentstehungsprozess **kontinuierlich und in einem iterativen Wechselspiel** zu entwickeln.

Werkzeug für die Modellierung von Zielsystemen

Die Untersuchungen haben gezeigt, dass die Akzeptanz für ein Werkzeug zur Modellierung von Zielsystemen insbesondere dann gegeben ist, wenn es **intuitiv bedienbar** ist und ein **Nutzen aus der Modellierung** gezogen werden kann, der über die reine Dokumentation hinausgeht und der bereits bei moderatem Modellierungsaufwand sichtbar wird (vgl. Abbildung 2-40 und Abbildung 2-41). Diesem Anspruch werden die Werkzeuge, die in den Untersuchungen verwendet wurden, teilweise gerecht. Dabei hat jedes einzelne der verwendeten Werkzeuge seine individuellen Stärken und Schwächen, die teilweise komplementär ausgeprägt sind. Im Fall der Darstellung von Elementen des Zielsystems und angrenzender Partialmodelle lassen sich die Stärken beider Werkzeuge vereinen und somit die jeweils vorherrschenden Schwächen in der Darstellung eliminieren:

- MediaWiki: Wiki-Seiten – **detaillierte Darstellung** einzelner Inhalte, **keine Übersicht** über die Summe der Inhalte
- Mind-Map: Baumstruktur – **gute Übersicht** über die Summe der Inhalte, **keine sinnvolle Detaillierung** der Inhalte

Hierbei gilt es, die beiden Darstellungsformen des MediaWiki und der Mind-Map als **zwei unterschiedliche Darstellungsformen auf den gleichen Datenbestand** zu interpretieren und einen **dynamischen Wechsel** zwischen beiden Darstellungsformen zu ermöglichen. Hierdurch können strukturelle Änderungen im Zielsystem, die sich aufgrund der vorherrschenden Unsicherheit in Produktentstehungsprozessen unvermeidlich ergeben, bestmöglich unterstützt werden.

Darüber hinaus hat sich gezeigt, dass ein Werkzeug zur Zielsystemmodellierung folgenden Aspekten Rechnung tragen sollte:

- **Wiederverwendbarkeit** von Zielsystemen bzw. Wiederverwendbarkeit von Erkenntnissen aus Projekten mittels einer Durchgängigkeit von unternehmensweiter Wissensbasis und projektspezifischen Zielsystemen.

- Berücksichtigung spezifischer **Änderungsautorisationen** für jedes einzelne Element des Zielsystems.
- **Identifikation relevanter Änderungen** bzw. Identifikation von Änderungen an relevanten Elementen des Zielsystems.

7 Modellierung von Zielsystemen

In diesem Kapitel werden die Ergebnisse und Erkenntnisse der beiden vorangegangenen Kapiteln genutzt, um eine Methode zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung abzuleiten, die den formulierten Bedarfen und Zielen gerecht wird.

Hierzu wird im Folgenden eine Möglichkeit der Strukturierung von Zielsystemen entwickelt, die eine ganzheitliche und durchgängige Modellierung von Zielsystemen erlaubt (Abschnitt 7.1). Hierauf aufbauend und unter Berücksichtigung der Erkenntnisse aus den beiden Fallstudien (Kapitel 6) werden Regeln zur Modellierung von Zielsystemen abgeleitet, die es dem Entwickler mechatronischer Produkte erlauben soll, effizient, selbstständig und im gesamten Verlauf des Produktentstehungsprozesses ein nutzenstiftendes Zielsystem zu modellieren (Abschnitt 7.2). Zur Umsetzung und Validierung der Methode zur Modellierung von Zielsystemen wird anschließend ein intuitiv anwendbares und flexibel einsetzbares Werkzeug entwickelt (Abschnitt 7.3). Hierdurch wird die Validierung der Methode möglich, die zuerst auf Basis einer retrospektiven Modellierung eines IPEK-internen Entwicklungsprojekts erfolgt und durch einen produktiven Einsatz der Methode in einem Entwicklungsprojekt eines Industrieunternehmens abgeschlossen wird (Abschnitt 7.4).

7.1 Strukturierung von Zielsystemen

Die Strukturierung von Zielsystemen wird anhand der erforderlichen Partialmodelle und deren Elemente, der Relationen zwischen den Elementen sowie der Attribute, die die Elemente charakterisieren, dargelegt.

7.1.1 Partialmodelle im Zielsystem

Die Untersuchungen im Rahmen der beiden Fallstudien³⁹³ haben gezeigt, dass eine **ganzheitliche und durchgängige Modellierung von Zielsystemen** auf Basis von neun Partialmodellen erfolgen kann (vgl. Abbildung 7-1). Ein wesentliches Charakteristikum dieser Kombination von Partialmodellen besteht darin, dass **alle drei Systeme des ZHO-Systemmodells** der Produktentstehung³⁹⁴ (Ziel-, Handlungs- und Objektsystem) gleichermaßen abgedeckt sind. Im Folgenden werden die einzelnen Partialmodelle erläutert.

³⁹³ vgl. Abschnitt 6.3

³⁹⁴ vgl. Abschnitt 2.2.1

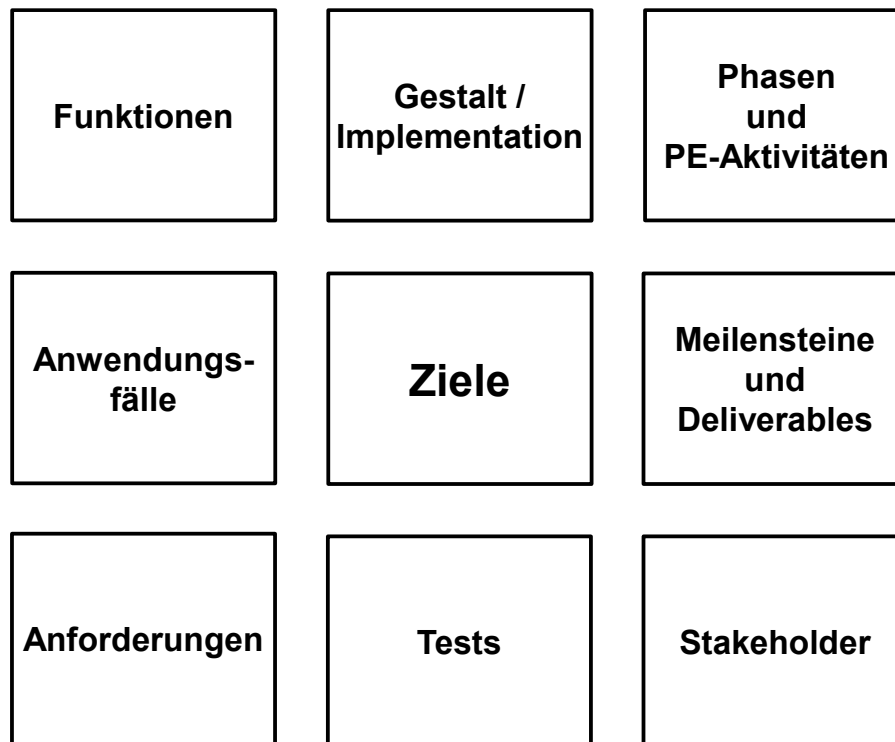


Abbildung 7-1: Partialmodelle für die Zielsystemmodellierung

Ziele

Dieses Partialmodell beinhaltet alle Ziele, die im Rahmen eines Entwicklungsprojekts erreicht werden sollen. Das Modell enthält damit die Gesamtheit der bewussten gedanklichen Vorwegnahmen eines zukünftigen Soll-Zustands, der durch alle beteiligten Akteure gemeinsam vereinbart und durch deren aktives Handeln erreicht bzw. vermieden wird. Die Inhalte des Modells stellen damit den **Sinn und Zweck eines Entwicklungsprojekts** dar, an dem sich sämtliche Inhalte aller anderen Partialmodelle ausrichten bzw. messen lassen müssen. Daher nimmt das Partialmodell der Ziele eine besondere, exponierte Position in der Darstellung der Partialmodelle ein (vgl. Abbildung 7-1). Die einzelnen Ziele innerhalb des Partialmodells können nach deren Quelle, insbesondere nach Stakeholdern und Produktlebenszyklusphasen, strukturiert werden. Stakeholder und Produktlebenszyklusphasen bilden auch wichtige Quellen zur Identifikation von Zielen. Weiterhin lassen sich Ziele aus bereits explizierten Anforderungen, identifizierten Anwendungsfällen oder zu erfüllenden Tests extrahieren.

Anforderungen

Dieses Partialmodell umfasst sämtliche Anforderungen, die das zu entwickelnde Produkt erfüllen muss, um die definierten Ziele zu erreichen. Das Partialmodell beinhaltet damit die explizierten **Eigenschaften und Bedingungen**, die das System und dessen Subsysteme aufweisen müssen, um ein formell vorgegebenes Dokument, beispielsweise einen Vertrag oder eine Norm, zu erfüllen. Hierbei beinhaltet das Partialmodell sowohl **funktionale wie auch nichtfunktionale**

Anforderungen (bzw. Qualitätsanforderungen) und kann entsprechend dieser Unterteilung strukturiert werden³⁹⁵. Quellen für Anforderungen sind in erster Linie formulierte Ziele. Weitere Quellen für Anforderungen können Stakeholder, Phasen im Produktlebenszyklus, die Produktumgebung sowie Anwendungsfälle und Tests sein. Generell kann auch die Untergliederung der nichtfunktionalen Anforderungen (bzw. Qualitätsanforderungen) zur Identifikation erforderlicher Anforderungen genutzt werden.

Anwendungsfälle

Das Partialmodell der Anwendungsfälle beinhaltet sämtliche (formalen) Beschreibungen der **Interaktionen zwischen dem zu entwickelnden System und den Systemnutzern** (Personen und Systeme). Hierbei werden sowohl die Interaktionsfolgen abgebildet, die zur Erfüllung der zugrundeliegenden Ziele führen (Haupt- und Alternativszenarien), als auch solche, die ausgeführt werden, wenn die zugrundeliegenden Ziele nicht erreicht werden können (Ausnahmeszenarien). Das Partialmodell beschreibt damit das **Verhalten eines Systems, das in unmittelbarer Interaktion mit den Systemnutzern stattfindet** und zur Erfüllung formulierter Ziele führt. Quellen von Anwendungsfällen können sowohl formulierte Ziele sein, als auch Systemnutzer und deren Wunsch bzw. Möglichkeiten mit dem System in Interaktion zu treten.

Funktionen

Dieses Partialmodell bildet sämtliche Funktionen und deren Verkettungen ab, die zur Realisierung der formulierten Ziele und Anforderungen erforderlich sind. Die einzelnen Funktionen lassen sich entsprechend ihrer Dekomposition in einer Funktionshierarchie abbilden. Die Abbildung der Verkettungen überführt die Funktionshierarchie in eine **Funktionsstruktur** auf deren Basis Wirknetzte³⁹⁶ als zustandsabhängige Verkettungen von Funktionen gebildet werden können (beispielsweise im Kontext eines Anwendungsfalls). Das Partialmodell kann sowohl teleologisch-normative Funktionen (Ziel-Funktionen³⁹⁶) als auch deskriptive Funktionen (Ist-Funktionen³⁹⁶) enthalten, wobei diese klar differenziert werden sollten. Funktionen lassen sich in erster Linie aus Anforderungen und Anwendungsfällen sowie aus bestehenden Realisierungen ableiten und können gemäß ihrem zugrundeliegendem Zweck oder ihrer realisierenden Gestalt bzw. Implementation strukturiert werden.

³⁹⁵ vgl. Abbildung 2-36

³⁹⁶ vgl. Albers & Wintergerst 2014

Gestalt/Implementation

Dieses Partialmodell beinhaltet die Gestalt bzw. Implementation der in den Anforderungen beschriebenen Eigenschaften mittels entsprechender Merkmale³⁹⁷. Die Elemente dieses Partialmodells repräsentieren folglich die **Verwirklichung der vereinbarten Ziele** und realisieren damit den gewünschten Mehrwert³⁹⁸. Aufgrund des unmittelbaren Bezugs zur physischen Verwirklichung erfolgt die Strukturierung des Partialmodells nach physischen Kriterien, beispielsweise der Baustruktur bzw. der Struktur der Stücklisten (BOM – Bill of Materials). Dies gilt in der Regel nicht für (Software-)Quelltext, der oftmals nach funktionalen Gesichtspunkten strukturiert ist.

Phasen und Produktentstehungsaktivitäten

Dieses Partialmodell umfasst die **Handlungen**, die im Rahmen eines Produktentstehungsprozesses erforderlich bzw. geplant sind, um die formulierten Ziele zu erreichen. Hierbei werden die Handlungen in (durch Meilensteine) abgrenzbare (Projekt-)Phasen sowie verfeinernde Produktentstehungsaktivitäten differenziert. Hierdurch kann der Einzigartigkeit jedes Produktentstehungsprozesses Rechnung getragen werden³⁹⁹, da insbesondere einzelne Produktentstehungsaktivitäten flexibel angepasst und im Rahmen der Produktgenerationsentwicklung⁴⁰⁰ wiederverwendet werden können. Da auf eine separate Modellierung der handlungsbefähigenden Ressourcen (mit Ausnahme der Stakeholder und Testumgebungen) verzichtet wird, werden die einzelnen Produktentstehungsaktivitäten falls erforderlich unmittelbar mit den assoziierten Ressourcen modelliert. Für den Fall, dass ein explizites Ressourcenmanagement eine große Bedeutung für die Erreichung der formulierten Ziele hat, ist zum Zwecke einer integrativen Betrachtung von Prozess und Prozessressourcen ein Partialmodell der (Projekt-)Ressourcen zu ergänzen.

Meilensteine und Deliverables

Das Partialmodell der Meilensteine und Deliverables beinhaltet sämtliche **Zeitpunkte** bzw. Ereignisse, zu denen der Projektfortschritt explizit zu überprüfen ist und Entscheidungen mit hoher Tragweite getroffen werden müssen. Hierzu beinhaltet das Partialmodell (meilensteinspezifisch) **Indikatoren**, die zur Messung des Projektfortschritts herangezogen werden, entsprechende Methoden zur Erhebung der Indikatoren, meilensteinspezifische **Sollwerte der Indikatoren, Maßnahmen**, die

³⁹⁷ vgl. Weber 2012

³⁹⁸ Im Fall der Gestalt sind dies alle relevanten geometrischen und stofflichen Eigenschaften eines Systems. Im Fall der Implementation ist dies (Quell-)Code, der mittels eines Übersetzers (Assembler, Compiler) unmittelbar durch eine Prozessoreinheit ausgeführt werden kann (vgl. Thau 2013, S. 6 und IEEE 610.12 1990).

³⁹⁹ vgl. Albers 2010

⁴⁰⁰ vgl. Albers, Bursac & Wintergerst 2015

bei bestimmten Soll-Ist-Abweichungen einzelner bzw. mehrerer Indikatoren einzuleiten sind sowie berichtende und entscheidungsbefugte Stakeholder. Hierbei bilden Deliverables (Arbeitsergebnisse) besondere Indikatoren, die zum Zeitpunkt eines Meilensteins überprüft werden. Zum einen beinhalten Deliverables eine Outcome- bzw. Outputorientierung, wodurch die Greifbarkeit des Indikators gegenüber anderen (abstrakten) Indikatoren erheblich verbessert ist. Zum anderen sind Deliverables in Unternehmen oftmals bereits etabliert, so dass mit dem Deliverable assoziierte Ziele und Prozesse in der Regel bekannt und ebenfalls etabliert sind. Hierdurch ist eine weitere Untergliederung der Prozesse auf informeller Ebene möglich, wodurch eine kontinuierlichere Überprüfung des Projektfortschritts möglich wird.

Stakeholder

Dieses Partialmodell umfasst alle Personen, Gruppen oder Organisationen, die ein berechtigtes Interesse am Verlauf oder Ergebnis des Produktentstehungsprozesses haben. Das Partialmodell beinhaltet damit in erster Linie die involvierten Menschen als **Akteure im Produktentstehungsprozess**. Neben entsprechenden Rumpfdaten der Akteure, wie Name, Firmen- bzw. Abteilungszugehörigkeit und Kontaktdaten können weitere Aspekte, beispielsweise Erfahrungen und Kompetenzen oder Ressourcen- bzw. Auslastungslage abgebildet werden. Das Partialmodell beinhaltet darüber hinaus auch solche Personen, Gruppen oder Organisationen, die nicht aktiv in den Produktentstehungsprozess eingreifen, sondern auf der Basis einzuhaltender Randbedingungen, beispielsweise Normen oder Gesetze, interagieren (Gesetzgeber, Berufsgenossenschaften, TÜV und weitere).

Tests

Dieses Partialmodell beinhaltet alle **Tests**, die im Rahmen eines Produktentstehungsprozesses durchgeführt werden, um zuvor formulierte Ziele, Anforderungen aber auch Hypothesen hinsichtlich ihrer Erfüllung zu überprüfen. Entsprechend beinhaltet das Partialmodell sowohl **Testfälle** als auch (testspezifische) **Testumgebungen** und **Testinterpretationen**. Hierbei wird ein spezifischer Test durch Kombination eines spezifischen Testfalls mit einer spezifischen Testumgebung und einer spezifischen Testinterpretation (ab-)gebildet. Erforderliche Tests werden zum einen vorgegeben und müssen aufgrund vertraglicher oder gesetzlicher Vorgaben durchgeführt und bestanden werden. Zum anderen resultieren Tests aus dem Bedarf, Unsicherheiten im Produktentstehungsprozess (frühzeitig) zu verringern und damit das Risiko zu reduzieren, dass das Produkt des Entstehungsprozesses die vorhandenen Ziele und Anforderungen nicht erfüllt.

7.1.2 Elementtypen im Zielsystem

Die projektspezifischen Inhalte der einzelnen Partialmodelle werden über konkrete Ausprägungen (Elemente) von Elementtypen gebildet, wobei Elementtypen nicht per se einzelnen Partialmodellen zugeordnet sind. So beinhaltet beispielsweise das Partialmodell der Ziele Elemente des Elementtyps Ziel. Es ist jedoch auch möglich, dass Elemente des Elementtyps Ziel (Ziele) in anderen Partialmodellen wie beispielsweise dem Partialmodell der Phasen und Produktentstehungsaktivitäten oder der Tests vorkommen. Hierdurch entsteht eine große Flexibilität in der Modellierung des Zielsystems, wodurch sich vergleichsweise einfache Modellierungsregeln ableiten lassen, die in der Modellbildung dynamisch angewendet werden können⁴⁰¹. Im Folgenden werden Elementtypen definiert, die innerhalb der identifizierten Partialmodelle⁴⁰² Verwendung finden können, um hierdurch eine ganzheitliche und durchgängige Zielsystemmodellierung zu ermöglichen (vgl. Abbildung 7-2)

| | | | |
|----------------|-------------------------------|--------------------------|---------------------------|
| Partialmodelle | Ziele | Anforderungen | Anwendungsfälle |
| | Funktionen | Gestalt / Implementation | Phasen und PE-Aktivitäten |
| | Meilensteine und Deliverables | Stakeholder | Tests |
| Elementtypen | Ziel | Anforderung | Anwendungsfall |
| | Funktion | Gestalt / Implementation | Phasen / Aktivitäten |
| | Meilenstein | Deliverable | Stakeholder |
| | Test | Entscheidung | Begründung |
| | Dokument / Information | Schnittstelle | |
| Relationen | Dekomposition | Verfeinerung | Ungerichtete Relation |
| | Gerichtete Relation | Unterstützung | Konflikt |
| | Inkonsistenz | Ableitung | Erfüllung |
| | Verifizierung | Validierung | Verantwortlichkeit |
| | Treiber | | |
| Attribute | Bezeichner | Status | Wichtigkeit |
| | Autorisation | Änderungsinformation | Freitext |

Abbildung 7-2: Übersicht über die Strukturierung von Zielsystemen

⁴⁰¹ vgl. Abschnitt 7.2

⁴⁰² vgl. Abschnitt 7.1.1

Ziel

Ein Element des Typs Ziel enthält genau ein Ziel, das durch aktives Handeln erreicht oder vermieden werden soll.

Der Elementtyp kann entsprechend der Projektziele weiter in Finanzziele, Markt-/Vertriebsziele, Produktziele, Produktionsziele sowie Projektmanagementziele differenziert werden⁴⁰³. Ein Beispiel für ein Element des Typs Ziel in der Fahrzeugentwicklung bildet der zu erreichende Kraftstoffverbrauch oder der SOP (Start of Production).

Anforderung

Ein Element des Typs Anforderung enthält genau eine Anforderung, die ein System oder Systemkomponente aufweisen muss, um ein Ziel zu erreichen.

Der Elementtyp kann beispielsweise weiter entsprechend des FURPS+-Modells⁴⁰⁴ in die verschiedenen produktarchitekturrelevanten Anforderungsarten untergliedert werden. Ein Beispiel für ein Element des Typs Anforderung in der Entwicklung von Hybridfahrzeugen bildet die Kapazität des erforderlichen Energiespeichers.

Anwendungsfall

Ein Element des Typs Anwendungsfall enthält einen Anwendungsfall, ein Hauptszenario oder zugehörige Alternativ- und Ausnahmeszenarien, die ein System bei der Interaktion mit einem Systemnutzer ausführt, um einen gewünschten Mehrwert zu erreichen.

Der Elementtyp kann entsprechend in Anwendungsfall, Haupt-, Alternativ- sowie Ausnahmeszenario untergliedert werden. Ein Beispiel für ein Element des Typs Anwendungsfall stellt die Anfahrt eines Hybridfahrzeugs mit Anhänger bei zwölfprozentiger Steigung dar.

Funktion

Ein Element des Typs Funktion enthält genau eine gewünschte oder tatsächliche Beziehung zwischen Attributen (Inputs, Outputs, Zustände) eines Systems.

Der Elementtyp kann entsprechend weiter in Ziel-Funktion und Ist-Funktion differenziert werden. Ein Beispiel für ein Element des Typs Funktion könnte im Bereich der Fahrzeugentwicklung die Messung der Raddrehzahl an jedem Rad sein.

⁴⁰³ vgl. Abbildung 2-29

⁴⁰⁴ vgl. Abschnitt 2.5.1

Gestalt/Implementation

Ein Element des Typs Gestalt/Implementation bildet entweder stoffliche und geometrische Eigenschaften eines Systems ab (Gestalt) oder Code, der mittels eines Übersetzers (Assembler, Compiler) unmittelbar durch eine Prozessoreinheit ausgeführt werden kann.

Hierbei kann eine Differenzierung des Elementtyps entsprechend einer disziplin- oder domänenspezifischen (fachgebietsspezifischen) Zuordnung hilfreich sein. Ein Beispiel für ein Element des Typs Gestalt könnte ein Sensor zur Messung der Raddrehzahl, beispielsweise ein Hall-Sensor oder ein Impulsrad sein. Ein Beispiel für ein Element des Typs Implementation könnte der entsprechende Softwarecode sein, der die Information des Sensors in eine Drehzahl überführt.

Phase/Aktivität

Ein Element des Typs Phase/Aktivität beinhaltet eine Handlung, die im Rahmen eines Produktentstehungsprozesses erforderlich bzw. geplant ist, um die formulierten Ziele zu erreichen.

Der Elementtyp kann in Phase und (Produktentstehungs-)Aktivität differenziert werden. Als Beispiel für ein Element des Typs Phase bildet der abschließende, mehrwöchige Test in der Fahrzeugentwicklung unter widrigen Bedingungen (Wüste, Eislandschaft). Ein Element des Typs Aktivität dieser Phase könnte beispielsweise die Definition bzw. Auswahl der zu durchlaufenden Tests und deren Sequenz sein.

Meilenstein

Ein Element des Typs Meilenstein beinhaltet einen Zeitpunkt bzw. ein Ereignis, zu dem der Projektfortschritt anhand definierter Indikatoren von definierten Stakeholdern berichtet und bewertet wird und Entscheidungen mit hoher Tragweite getroffen werden. Hierzu enthält ein Meilenstein spezifische Sollwerte der Indikatoren und zusätzlich Maßnahmen, die bei bestimmten Soll-Ist-Abweichungen einzelner bzw. mehrerer Indikatoren einzuleiten sind. Ein Element des Typs Meilenstein ist typischerweise die Verabschiedung eines vertragswirksamen Pflichtenhefts.

Deliverable

Ein Element des Typs Deliverable definiert einen Indikator, der im Rahmen eines Meilensteins zur Messung des Projektfortschritts herangezogen wird sowie die entsprechende Methode zur Erhebung des Indikators.

Der Elementtyp Deliverable kann durch einen Elementtyp Indikator ergänzt werden, auf dessen Basis eine kontinuierliche Bewertung des Projektfortschritts auch zwischen den Meilensteinen möglich wird. Ein Beispiel für ein Element des Typs Deliverable könnte das Bestehen eines definierten Tests oder auch die Verausgabung eines gewissen Anteils des Budgets zur Prüfstandsfertigung sein.

Stakeholder

Ein Element des Typs Stakeholder beinhaltet eine Person, Gruppe oder Organisation, die ein berechtigtes Interesse am Verlauf oder Ergebnis des Produktentstehungsprozesses hat sowie relevante Informationen über diese Person, Gruppe oder Organisation.

Der Elementtyp kann entsprechend in Person, Gruppe und Organisation differenziert werden. Ein Element des Typs Stakeholder könnte beispielsweise der projektverantwortliche Entwickler sein.

Test

Ein Element des Typs Test beinhaltet einen Test, einen Testfall, eine Testumgebung oder eine Testinterpretation, die Erkenntnisse über das System unter Test liefern, insbesondere ob das System zuvor definierte Ziele, Anforderungen oder Hypothesen ganz, teilweise oder nicht erfüllt.

Der Elementtyp kann entsprechend in Test, Testfall, Testumgebung sowie Testinterpretation untergliedert werden. Entsprechend könnte ein Element des Typs Test beispielsweise die Überprüfung der Festigkeit eines Biegestabs sein. Der Testfall würde hierbei die Einspannung des Biegestabs sowie der örtliche und zeitliche Verlauf der aufzuprägenden Kraft sein. Die Testumgebung könnte entweder ein physischer Prüfstand oder ein bestehendes FE-Modell sein.

Entscheidung

Ein Element des Typs Entscheidung enthält eine offene bzw. getroffene Entscheidung, die eine Auswahl aus bestehenden Alternativen vorbereitet bzw. dokumentiert sowie relevante Informationen bezüglich der Entscheidungssituation, beispielsweise den Zeitpunkt der Entscheidung sowie involvierte Stakeholder. Ein Element des Typs Entscheidung beinhaltet damit beispielsweise die (noch zu treffende) Entscheidung über die Auswahl eines Antriebsmotors für ein Hybridfahrzeug sowie die zugehörigen entscheidungsrelevanten Informationen.

Begründung

Ein Element des Typs Begründung beinhaltet die Begründung(en) eines bestehenden Sachverhalts oder einer getroffenen Entscheidung. Ein Typ des Elements Begründung ergänzt bzw. begründet beispielsweise eine dokumentierte Anforderung.

Dokument/Information

Ein Element des Typs Dokument/Information beinhaltet eine abgrenzbare bzw. abgegrenzte Menge an Informationen bzw. ein Dokument, das entsprechende Informationen enthält. Ein Element des Typs Dokument/Information könnte beispielsweise das Datenblatt eines Antriebsmotors für ein Hybridfahrzeug enthalten.

Schnittstelle

Ein Element des Typs Schnittstelle enthält dedizierte Informationen über (bestimmte) Wechselwirkungen zwischen zwei anderen Elementen, die über eine einfache Relation hinausgehen. Beispielsweise könnte ein Element des Typs Schnittstelle die Pinnbelegung einer Steckerverbindung beinhalten, die zwei andere Systeme zum Informationsaustausch benötigen.

7.1.3 Relationen im Zielsystem

Explizite Relationen ermöglichen die Modellierung von **Wechselwirkungen** im Zielsystem und sind damit neben den Partialmodellen und Elementtypen ein weiterer essenzieller Bestandteil der Struktur von Zielsystemen. Um eine Durchgängigkeit im Zielsystem realisieren zu können, müssen jedoch sämtliche relevante Relationen zwischen allen Partialmodellen bzw. Elementtypen des Zielsystems explizit verfügbar sein. Hierbei besteht der Anspruch, sowohl **Quer- als auch Diagonalbeziehungen** bidirektional innerhalb von und zwischen Partialmodellen ausbilden zu können. Entsprechend dem Ziel einer möglichst schlanken Methode zur Zielsystemmodellierung gilt es, die Anzahl an Relationen auf ein sinnvolles Minimum zu reduzieren (vgl. Abbildung 7-2). Hierdurch wird sowohl die Nachvollziehbarkeit des Zielsystems eines Produktes als auch die Wiederverwendbarkeit einzelner Sub-Zielsysteme im Rahmen der Produktgenerationsentwicklung realisiert.

Die Relationen beruhen hierbei zum einen auf dem aktuellen Stand der Forschung⁴⁰⁵ zum anderen aus einer Gegenüberstellung aller identifizierter Partialmodelle und Elementtypen. Auf dieser Basis können vielzählige, potenzielle Relationen identifiziert werden, wobei sich aufgrund von Ähnlichkeiten bzw. mehrfacher Verwendbarkeit von Relationen einzelne Relationen bzw. Cluster von Relationen als besonders geeignet herausbilden. Im Folgenden werden die Relationen vorgestellt, die mindestens benötigt werden, um alle relevanten Wechselwirkungen zwischen den vorhandenen Partialmodellen bzw. Elementtypen abzubilden. Dabei werden die identifizierten Relationen bewusst nicht den einzelnen Partialmodellen bzw. Elementtypen per se zugeordnet und damit entsprechende Restriktionen geschaffen. Vielmehr sollen die Relationen auf Basis klarer Definitionen zweck- und bedarfsorientiert (pragmatisch) verwendet werden.

Relationen entlang der Hierarchie

Die Relationen entlang der Hierarchie im Zielsystem stellen eine besondere Wechselwirkung zwischen Elementen dar, die dadurch gekennzeichnet ist, dass sie

⁴⁰⁵ vgl. Abschnitt 2.5

die jeweils als **dominant definierte Wechselwirkung** eines Elements repräsentiert. Hierbei bildet auch die Hierarchie Wechselwirkungen zwischen einzelnen Elementen aus, die zwar nicht zwingend expliziert werden müssen, über deren Bedeutungen jedoch Klarheit herrschen sollte.

| Relationstyp | Relation | Inverse Relation |
|---------------|------------------|------------------|
| Dekomposition | beinhaltet | Teil von |
| Verfeinerung | verfeinert durch | verfeinert |

Die Dekompositionsrelation besagt, dass ein Element ein anderes (vollständig) enthält und greift damit das hierarchische Konzept der Systemtheorie auf. Die Verfeinerungsrelation besagt, dass ein Element ein anderes Element genauer, verfeinernd beschreibt. Die Ableitungsrelation beschreibt, dass sich ein Element logisch oder begründet aus einem anderen Element ableitet bzw. ergibt.

Generische Relationen

Generische Relationen setzen zwei **Elemente in Beziehung**, ohne die Art der Beziehung genauer zu benennen. Dies ist zum einen erforderlich, wenn die genaue Beziehung zwischen zwei Elementen (noch) nicht bekannt ist oder nicht genauer modelliert werden soll. Zum anderen können mittels generischer Relationen auch bisher unbekannte Wechselwirkungen (notdürftig) modelliert werden.

| Relationstyp | Relation | Inverse Relation |
|-----------------------|--------------|----------------------|
| Ungerichtete Relation | in Beziehung | |
| Gerichtete Relation | beeinflusst | wird beeinflusst von |

Die Ungerichtete Relation besagt hierbei lediglich, dass zwei Elemente in einer gewissen Beziehung zueinander stehen, wohingegen die Gerichtete Relation zusätzlich eine Aussage über die gegenseitige Einflussnahme beinhaltet.

Verträglichkeits-Relationen

Verträglichkeits-Relationen beschreiben die **Kompatibilität** zwischen zwei Elementen und erlauben entsprechend Aussagen über die Ausgeglichenheit bzw. Ausgewogenheit der modellierten Elemente.

| Relationstyp | Relation | Inverse Relation |
|---------------|-----------------|------------------|
| Unterstützung | unterstützt | |
| Konflikt | in Konflikt mit | |
| Inkonsistenz | inkonsistent zu | |

Die Unterstützungs-Relation sagt aus, dass sich zwei Elemente (gegenseitig) begünstigen und deutet damit ein Synergiepotenzial an. Die Konflikt-Relation

entspricht dem Gegenteil der Unterstützungs-Relation und besagt, dass sich zwei Elemente (gegenseitig, in ihrer Umsetzung) behindern. Die Inkonsistenz-Relation zeigt einen Widerspruch auf, der durch zwei Elemente bzw. deren Ausprägungen entsteht.

Ziel-Objekt-Relationen

Ziel-Objekt-Relationen explizieren Wechselwirkungen zwischen Zielen, Anforderungen, Funktionen und der Gestalt bzw. Implementation. Sie stellen daher essenzielle, produktbezogene Relationen zur **Sicherstellung der Nachvollziehbarkeit** im Zielsystem dar.

| Relationstyp | Relation | Inverse Relation |
|--------------|-----------|--------------------|
| Ableitung | leitet ab | abgeleitet von |
| Erfüllung | erfüllt | wird erfüllt durch |

Die Ableitungsrelation beschreibt, dass sich ein Element logisch oder begründet aus einem anderen Element ableitet bzw. ergibt. Diese Relation kann in Ausnahmefällen auch entlang der Hierarchie des Zielsystemmodells verwendet werden. Die Erfüllungsrelation besagt, dass ein Element ein anderes Element (und dessen Forderung) erfüllt bzw. zu dessen Erfüllung beiträgt.

Verifizierungs- und Validierungsrelationen

Die Verifizierung und Validierung ist zentraler Bestandteil der Produktentstehung, weshalb die Relationstypen Verifizierung und Validierung, die stets mit Tests (Aktivitäten) in Beziehung stehen, eine besondere Rolle für die Nachvollziehbarkeit im Zielsystem einnehmen. Erst auf Basis dieser Relationstypen ist es möglich, nachzuvollziehen (sicherzustellen), dass formulierte **Ziele und Anforderungen tatsächlich erfüllt** sind.

| Relationstyp | Relation | Inverse Relation |
|---------------|-------------|-------------------|
| Verifizierung | verifiziert | verifiziert durch |
| Validierung | validiert | validiert durch |

Die Verifizierungsrelation besagt, dass ein Element durch ein anderes Element hinsichtlich dessen Erfüllung einer Anforderung überprüft wird. Die Validierungsrelation hingegen beschreibt, dass ein Element durch ein anderes Element hinsichtlich dessen Erfüllung eines Ziels überprüft wird.

Handlungsrelationen

Handlungsrelationen beschreiben **Wechselwirkungen zu Elementen des Handlungssystems**, insbesondere zu Akteuren bzw. Stakeholdern und Aktivitäten.

Sie sind für das Management eines Produktentstehungsprozesses von besondere Wichtigkeit.

| Relationstyp | Relation | Inverse Relation |
|--------------------|--------------------|------------------|
| Verantwortlichkeit | verantwortlich für | verantwortlich |
| Treiber | treibt | getrieben von |

Die Verantwortlichkeitsrelation ordnet einem Element einen verantwortlichen Akteur (bzw. Stakeholder) zu. Die Treiberrelation ordnet einem Element eine es treibende Aktivität oder Akteur (bzw. Stakeholder) zu.

7.1.4 Attribute im Zielsystem

Explizite Attribute ermöglichen eine (genauere) **Beschreibung einzelner Elementtypen**, indem explizite Eigenschaften bzw. Merkmale sowie zulässige Eigenschafts- bzw. Merkmalsausprägungen definiert werden. Hierdurch können einerseits relevante Informationen strukturiert erfasst und abgebildet werden und andererseits auch gezielt entsprechend der definierten Attribute abgerufen werden.

Die Attribute beruhen im Wesentlichen auf Erkenntnissen, die im Rahmen der Untersuchungen zur Modellierung von Zielsystemen⁴⁰⁶ gewonnen wurden. Hierbei sind etliche Attribute denkbar und möglicherweise auch sinnvoll. Jedoch wurden bewusst solche Attribute ausgewählt, die auf alle Elementtypen angewandt werden können, um somit bei minimaler Anzahl an Attributen einen maximalen Mehrwert für die Zielsystemmodellierung erzeugen zu können (vgl. Abbildung 7-2).

Bezeichner

Jedes Element benötigt zum Zwecke der Identifikation einen Bezeichner. Dabei muss zwischen einem eindeutigen Bezeichner (ID bzw. Kennung) und dem Namen eines Elements unterschieden werden. Der **eindeutige Bezeichner** ist zur eindeutigen Identifikation eines Elements (durch ein verarbeitendes System) erforderlich, wohingegen der **Name** eines Elements zur effizienten Einordnung und Erfassung des Elements durch einen menschlichen Akteur genutzt wird.

Status

Der Status eines Elements stellt eine **zweckdienliche Vereinigung des Reife- und Härtegrades** als Beurteilungsdimensionen von Zielen dar. Dabei orientiert sich der Status eines Elements aus Gründen der Akzeptanz an praxisetablierten Freigabestatus, die im Rahmen des Freigabemanagements von PDM-Systemen zum Einsatz

⁴⁰⁶ vgl. Kapitel 6

kommen⁴⁰⁷. Hierbei orientieren sich die jeweiligen Status-Ausprägungen am Lebenszyklus des bezeichnenden Elements. In Abhängigkeit des jeweiligen Elements und dessen Einordnung in die Systematik des ZHO-Modells, kann der Status unterschiedliche Ausprägungen annehmen:

| Ziel & Objektsystem | Handlungssystem |
|---------------------|-----------------|
| Entwurf | |
| Eingefroren | Eingeplant |
| Freigegeben | In Durchführung |
| Bestätigt | Abgeschlossen |
| Zurückgezogen | |

Der Lebenszyklus und damit Status eines jeden Elements beginnt mit dessen Entwurf. In diesem Status hat/nimmt ein Element (per Definition) noch keinen Einfluss auf andere Elemente bzw. den Produktentstehungsprozess und kann damit beliebig (oft) geändert werden. Elemente des Ziel- und Objektsystems werden in einem nächsten Schritt eingefroren. Hierdurch wird einerseits signalisiert, dass (vorerst) keine weiteren Änderungen am betreffenden Element mehr vorgenommen werden und andererseits wird hierdurch der Freigabeprozess des Elements angestoßen. Durch eine entsprechende Freigabe erlangt ein Element den Status Freigegeben, wodurch es auf Basis einer (begründeten) Entscheidung fortan zu berücksichtigen ist und nicht mehr geändert werden darf. Ein freigegebenes Element kann den Status Bestätigt erlangen, indem die Korrektheit des Elements durch eine Verifizierung bzw. Validierung bestätigt wurde. Müssen an einem freigegebenen oder bestätigten Element Änderungen vorgenommen werden, fällt es auf den Status Entwurf zurück. Elemente des Handlungssystems, insbesondere Produktentstehungsaktivitäten, können nach dem Status Entwurf entweder direkt zur Durchführung eingeplant werden oder durchlaufen zuerst die Status Eingefroren, Freigegeben und optional Bestätigt. Eingeplante Elemente des Handlungssystems wechseln in den Status In Durchführung, sobald ein definierter Starttermin erreicht ist oder Ressourcen abgerufen werden. Entsprechend erfolgt ein Wechsel in den Status Abgeschlossen, sobald ein Endtermin erreicht ist oder keine Ressourcen mehr abgerufen werden. Müssen an einem eingeplanten oder in Durchführung befindlichen Element Änderungen vorgenommen werden, fällt es auf den Status Entwurf zurück. Elemente, die keine Gültigkeit (mehr) besitzen und deren Inhalt und Status sich voraussichtlich nicht mehr ändern werden, erhalten den Status

⁴⁰⁷ vgl. Stockinger 2008, S. 15

Zurückgezogen und sind nur noch zum Zwecke der Nachvollziehbarkeit zu berücksichtigen.

Wichtigkeit

Das Attribut Wichtigkeit leitet sich indirekt aus der Kritikalitätsmatrix⁴⁰⁸ ab und indiziert, welche Elemente von **besonderer Bedeutung** sind und entsprechend mit besonderer Aufmerksamkeit beobachtet werden müssen. Die Unterteilung der Wichtigkeit kann entweder anhand der vier Niveaus der Kritikalitätsmatrix erfolgen (unwichtig, trivial, wichtig, kritisch) oder auf Basis einer (beliebig) zu wählenden numerischen Skala, beispielsweise eins (nicht wichtig) bis zehn (sehr wichtig).

Autorisation

Das Attribut Autorisation zeigt an, welche **Rechte** ein Akteur (Stakeholder) bezüglich eines bestimmten Elements hat, beispielsweise Schreib- oder Leserechte. Auf Basis der Untersuchungen zur Modellierung von Zielsystemen⁴⁰⁹ kann davon ausgegangen werden, dass bereits ein erheblicher Mehrwert entsteht, wenn etwaige Änderungsrechte eines Elements auf einen bestimmten Akteur eingeschränkt werden können. Folglich kann ein bestimmtes Element entweder von allen Akteuren geändert werden, oder von nur genau einem.

Änderungsinformationen

Informationen über Änderungen an einzelnen Elementen sollten mindestens in einer Form vorliegen, dass eine Änderung bemerkt werden kann. Daher sollte mindestens ein Attribut vorhanden sein, dass das **Datum der letzten Änderung** eines Elements sowie den ändernden Akteur (Stakeholder) beinhaltet, sodass zumindest auf kommunikativer Basis der Änderungsinhalt ermittelt werden kann.

Freitext

Das Attribut Freitext eines Elements stellt eine pragmatische Lösung dar, um die **inhaltliche Ausgestaltung** der einzelnen Elemente zu realisieren. Damit die Anschlussfähigkeit an bestehende bzw. alternative Formen der Informationsmodellierung gegeben ist, kann das Attribut Freitext neben Zeichenfolgen auch Hyperlinks beinhalten. Hierdurch können sämtliche Informationen bzw. Dokumente eingebunden werden, die im Inter- und Intranet sowie in (Netzwerk-)Laufwerken verfügbar sind. Damit können auch alternative Formen der Informationsmodellierung genutzt werden (z.B. Grafiken oder auch PowerPoint).

⁴⁰⁸ vgl. Abschnitt 5.3.2

⁴⁰⁹ vgl. Kapitel 6

7.2 Regeln zur Modellierung von Zielsystemen

Auf Basis der gesammelten Erfahrungen aus den Untersuchungen zur Modellierung von Zielsystemen⁴¹⁰ sowie basierend auf der Strukturierung von Zielsystemen⁴¹¹ werden im Folgenden Regeln und Empfehlungen zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung abgeleitet. Hierbei sollen möglichst **wenige und gleichzeitig einfache Regeln/Empfehlungen** gebildet werden, die in der eigentlichen Modellbildung flexibel durch den Modellerschaffer angewandt werden können. Hierdurch soll die Eigenkomplexität der Methode minimiert und die individuelle Akzeptanz in der Anwendung der Methode erhöht werden. Die hieraus resultierenden 25 Regeln und Empfehlungen werden im Folgenden vorgestellt (vgl. Abbildung 7-3).

7.2.1 Allgemeine Modellierungsregeln

Die allgemeinen Modellierungsregeln sind nicht auf einzelne Partialmodelle, Elementtypen, Relationen oder Attribute ausgerichtet. Sie formulieren vielmehr, wie die Strukturierung des Zielsystems⁴¹¹ prinzipiell dazu genutzt werden kann, ein nutzenstiftendes Zielsystem zu modellieren.

Unreife Inhalte modellieren / Bestehende Inhalte nutzen

Für die Zielsystemmodellierung ist es unerlässlich, dass Inhalte abgebildet werden, die noch nicht mit absoluter oder ausreichender Sicherheit vorliegen. Der Anspruch einer absoluten Sicherheit würde zwangsläufig dazu führen, dass kein einziges neues Element explizit abgebildet wird. Jedoch ist bereits der Anspruch einer ausreichenden Sicherheit kritisch einzuschätzen, da aufgrund der vorherrschenden Wechselwirkungen im Zielsystem diese Sicherheit nur erreicht werden kann, wenn wechselwirkende Inhalte modelliert, bekannt und auf dieser Basis berücksichtigt sind. Entsprechend gilt es, die **Zielsystemmodellierung als Prozess** zu verstehen, der auf Basis einer relativen Unsicherheit beginnt und über den gesamten Produktentstehungsprozess (und darüber hinaus) andauert. Im Rahmen der Produktgenerationsentwicklung sind daher **bestehende (Sub-)Zielsysteme**, die in das Zielsystem der nächsten Produktgeneration aufgenommen werden können, eine äußerst wertvolle Ausgangsbasis um die Qualität des Zielsystems und die Effizienz der Zielsystemmodellierung zu steigern sowie eventuelle Hemmnisse der expliziten Modellierung neuer Zielsysteminhalte abzubauen.

⁴¹⁰ vgl. Kapitel 6

⁴¹¹ vgl. Abschnitt 7.1

| | |
|-------------------------------------|---|
| Allgemeine Modellierungsregeln | Unreife Inhalte modellieren / Bestehende Inhalte nutzen |
| | Regelmäßig konsolidieren |
| | Generelles Domänenwissen auslagern und vernetzen |
| | Abbildung von Dekompositionen und Verfeinerungen über die Hierarchie realisieren |
| | Mehrfachverwendung und Sichten von Elementen mittels assoziierter Kopien realisieren |
| | Durchgängigkeit über Relationen herstellen |
| | Alle Partialmodelle, Elementtypen und Attribute (optional) nutzen |
| | Übergeordnete Inhalte im gleichnamigen Partialmodell abbilden |
| | Spezifische Inhalte unterhalb des betroffenen Elements abbilden |
| | Freitext zur Ausdetaillierung eines Elements nutzen |
| | Elemente für Inhalte und als Container nutzen |
| Spezifische Modellierungsregeln | Zweckbasierte Verwendung von Elementtypen |
| | Verantwortlichkeiten definieren |
| | Tests frühzeitig definieren und vernetzen |
| | Meilensteine mit geeigneten Deliverables versehen |
| | Phasen mittels PE-Aktivitäten verfeinern und operationalisieren |
| | Entscheidungen zur Vorbereitung und Dokumentation von Entscheidungssituationen nutzen |
| | Anwendungsfälle zur Extraktion von Zielen und Anforderungen nutzen |
| | Funktionen als Mittler zwischen Anforderungen und Gestalt / Implementation nutzen |
| | Wichtige Begründungen und Dokumente / Informationen separat modellieren |
| | |
| Extraktion relevanter Informationen | Baumstruktur zur Exploration / Elementsicht für Details nutzen |
| | Suchbegriffe für einfache Suchanfragen nutzen |
| | Elementtypen und Attribute für komplexere Suchanfragen nutzen |
| | Änderungsdatum zur Identifikation geänderter / neuer Inhalte nutzen |
| | Knotensuche als zusätzliche Einschränkung nutzen |

Abbildung 7-3: Regeln und Empfehlungen zur Modellierung von Zielsystemen

Regelmäßig konsolidieren

Die (notwendige) Modellierung unreifer Inhalte hat unmittelbar zu Folge, dass eine **regelmäßige Konsolidierung** des modellierten Zielsystems erforderlich ist. Dies ist letztendlich darauf zurück zu führen, dass jeder Produktentstehungsprozess einzigartig ist⁴¹² und im Vorfeld nicht vollständig geplant werden kann⁴¹³. Entsprechend können die Strukturen und Inhalte eines projektspezifischen Zielsystems nur teilweise vorausgedacht werden – Änderungen und Anpassungen sind immanenter Bestandteil der Zielsystemmodellierung. Hierbei gilt es eine Balance zwischen Korrektheit und Aktualität des Zielsystems und dem entsprechenden Konsolidierungsaufwand zu finden. Hierbei können die Attribute Status und Wichtigkeit dabei unterstützen, eine konsensbasierte Konsolidierung des Zielsystems vorzunehmen.

Generelles Domänenwissen auslagern und vernetzen

Generelles Domänenwissen, das im Kontext eines konkreten Produktentstehungsprozesses benötigt oder generiert wird, sollte in einem vom Projekt unabhängigen Partialmodell gepflegt werden (**Partialmodell Wissensbasis**). Darüber hinaus muss eine Durchlässigkeit zwischen fallspezifischem Wissen und generellem Domänenwissen geschaffen werden. Mit einer hierarchischen Modellierung kann dies realisiert werden, indem das Partialmodell Wissensbasis auf gleicher Ebene wie das **Partialmodell Projekte** angesiedelt ist⁴¹⁴. Hierdurch werden Inhalte aus der Wissensbasis einfach und effizient in konkreten Produktentstehungsprojekten (wieder-)verwendbar (Produktgenerationsentwicklung). Gleichzeitig wird die Hemmschwelle zur Aufbereitung und Ablage von prinzipiell wiederverwendbarem Wissen, das in konkreten Projekten entstanden ist, reduziert.

Abbildung von Dekompositionen und Verfeinerungen über die Hierarchie realisieren

Die Hierarchie im Zielsystem sollte (ausschließlich) zur **Modellierung von Dekompositions- und Verfeinerungsbeziehungen** verwendet werden. Entsprechend sollte ein (Kind-)Element einem anderen (Eltern-)Element untergeordnet werden, wenn es Teil von diesem ist oder es verfeinert. Die Wurzel eines konkreten Produktentstehungsprojekts sollte die Projektbezeichnung bzw. die Projektzielsetzung beinhalten.

Elemente, die sich nach dieser Regel nicht eindeutig einem (Eltern-)Element zuordnen lassen, müssen per Konsens bzw. Beschluss zugeordnet werden.

⁴¹² vgl. Albers 2010

⁴¹³ vgl. Lohmeyer 2012, S. 13

⁴¹⁴ vgl. Abbildung 6-4

Mehrfachverwendung und Sichten von Elementen mittels assoziierter Kopien realisieren

Elemente, die aus inhaltlichen (Mehrfachverwendung) oder methodischen (Sichten) Gründen an mehreren verschiedenen Positionen innerhalb der Hierarchie angesiedelt sein sollen, dürfen nicht als voneinander losgelöste Elemente modelliert werden. Um Mehrfachverwendung und Sichten zu ermöglichen, ist eine Position in der Hierarchie zu definieren, die das editierbare Original-Element enthält und auf dessen Basis assoziierte Kopien an allen übrigen Positionen hinzugefügt werden können. Eine **assozierte Kopie** ist dadurch gekennzeichnet, dass sie die gleichen Eigenschaften wie das Original-Element besitzt, jedoch nicht editiert werden kann und stattdessen eine Relation des Relationstyps Assoziierte Kopie beinhaltet. Hierdurch lassen sich insbesondere auch Elemente des Partialmodells Wissensbasis in konkrete Projekte einbinden und damit im Zuge der Produktgenerationsentwicklung nutzbar machen.

| Relationstyp | Relation | Inverse Relation |
|-------------------|-----------------|------------------|
| Assoziierte Kopie | spezifiziert in | verwendet in |

Durchgängigkeit über Relationen herstellen

Die erforderliche Durchgängigkeit bzw. Nachvollziehbarkeit im Zielsystem ist über eine angemessene Verwendung der zur Verfügung stehenden **Relationen** sicherzustellen. Können die Wechselwirkungen zwischen zwei Elementen mittels einer Relation nicht ausreichend abgebildet werden, kann der Elementtyp **Schnittstelle** verwendet, um die Wechselwirkung genauer zu beschreiben. Das Element vom Typ Schnittstelle ist beiden in Beziehung zu setzenden Elementen hinzuzufügen, wobei ein Element das Original des Schnittstellenelements beinhaltet und das andere Element eine assoziierte Kopie davon.

Alle Partialmodelle, Elementtypen und Attribute (optional) nutzen

Sämtliche zur Verfügung stehenden Partialmodelle, Elementtypen und Attribute sollten **pragmatisch** und dem Modellierungszweck dienlich eingesetzt werden. Die Partialmodelle dienen dabei sowohl als geeignete Systematik zum Ablegen und Wiederfinden projektrelevanter Informationen als auch als (grobes) Zielsystem-Referenzmodell, das branchen- bzw. firmenspezifisch weiter verfeinert werden kann und projektspezifisch entwickelt werden muss. Hierbei sind alle relevanten Partialmodelle von Beginn des Produktentstehungsprozesses kontinuierlich, im

Sinne eines Spiral-Prozesses⁴¹⁵, voranzutreiben. Im Zweifelsfall bezieht dies alle vorhandenen Partialmodelle mit ein.

Übergeordnete Inhalte im gleichnamigen Partialmodell abbilden

Inhalte bzw. Elemente, die klar einem Partialmodell zugeordnet werden können und die eine **Relevanz für eine (relativ) große Anzahl anderer Elemente** besitzen, sollten im gleichnamigen Partialmodell des jeweiligen Elementtyps modelliert werden.

Beispielsweise sollte eine Anforderung an das zu entwickelnde System im Partialmodell Anforderungen modelliert werden. Weiterhin sollten (Produktentstehungs-)Aktivitäten, die eine erste Verfeinerung definierter Projektphasen darstellen, im Partialmodell Phasen und Produktentstehungsaktivitäten abgelegt werden.

Spezifische Inhalte unterhalb des Elements abbilden

Inhalte bzw. Elemente, die klar einem Element zugeordnet werden können und die primär eine **Relevanz für genau dieses Element besitzen**, sollten unterhalb dieses Elements modelliert werden und ggf. mit korrespondierenden Inhalten bzw. Elementen des zugehörigen Partialmodells vernetzt werden.

Beispielsweise sollten lösungsspezifische Anforderungen unterhalb der jeweiligen Lösung modelliert werden. Denkbar wäre die (lösungsspezifische) Lagerbelastbarkeit eines Schaltroboters (Elementtyp Anforderung), die sich aufgrund einer definierten Endeffektorbelastbarkeit und der gewählten Kinematik (Partialmodell Gestalt/Implementation) an einem bestimmten Freiheitsgrade ergibt. Ein weiteres Beispiel sind Offene-Punkte-Listen (Elementtyp Aktivität), die nicht innerhalb des Partialmodells Phasen und Produktentstehungsaktivitäten abgebildet werden müssen, sondern auch unterhalb des sie adressierenden Elements abgelegt werden können. Hierbei sollte die Offene-Punkte-Liste jedoch eine (Ableitungs-)Relation zum entsprechenden Element des Partialmodells Phasen und Produktentstehungsaktivitäten haben.

Freitext zur Ausdetaillierung eines Elements nutzen

Jedes modellierte Element sollte einen kurzen und **prägnanten Bezeichner** (Namen) besitzen. Eine **detaillierte Beschreibung** des Elements soll entsprechend im Rahmen des Attributs Freitext erfolgen. Hierbei sind neben Zeichenfolgen auch Hyperlinks zu verwenden.

⁴¹⁵ vgl. Abbildung 2-25

Elemente für Inhalte und als Container nutzen

Einzelne Elemente können sowohl **konkrete Inhalte** repräsentieren (Inhalts-Element) als auch lediglich als **strukturierender Container** dienen (Container-Element). Der erste Fall liegt vor, wenn das Attribut Freitext einen Inhalt besitzt, der zweite, wenn das Attribut Freitext leer ist. Beide Fälle sind zulässig und sollten zweckdienlich eingesetzt werden.

7.2.2 Spezifische Modellierungsregeln

Die spezifischen Modellierungsregeln beziehen sich unmittelbar auf ein Partialmodell, eine Relation, ein Attribut oder Kombinationen dieser. Hierbei haben die spezifischen Modellierungsregeln deutlich **empfehlenderen Charakter**, als die zuvor formulierten Allgemeinen Modellierungsregeln, die prinzipiell einzuhalten sind.

Zweckbasierte Verwendung von Elementtypen

Elementtypen sind nicht per se einzelnen Partialmodellen zugeordnet, d.h. prinzipiell kann **jeder Elementtyp in jedem Partialmodell verwendet werden**. Weiterhin können Elemente eines Elementtyps verschiedene Elemente anderer Elementtypen beinhalten. Insofern ist die Verwendbarkeit bzw. Kompatibilität der Elementtypen nicht explizit geregelt wodurch die Flexibilität der Modellierung gesteigert wird.

Verantwortlichkeiten definieren

Verantwortlichkeiten müssen definiert werden. Hierzu kann einerseits das Attribut **Autorisation** verwendet werden, da im Falle genau eines autorisierten Akteurs/Stakeholders dieser auch verantwortlich für die Inhalte des jeweiligen Elements ist. Andererseits können Verantwortlichkeiten über die **Verantwortlichkeitsrelation** festgelegt werden, indem die Relation zwischen einem Element und einem verantwortlichen Akteur/Stakeholder definiert wird. Zuständigkeiten lassen sich von Verantwortlichkeiten differenzieren, indem die Treiberrelation anstatt der Verantwortlichkeitsrelation verwendet wird.

Tests frühzeitig definieren und vernetzen

Tests zum Zwecke der Verifizierung und Validierung sind essenzieller Bestandteil eines schlussendlich erfolgreichen Produktentstehungsprozesses. Daher ist es unbedingt erforderlich, Tests (Testfälle, Testumgebungen und Testinterpretationen) **frühzeitig im Produktentstehungsprozess zu definieren**. Insbesondere für modellierte Ziele, Anforderungen und Meilensteine bzw. Deliverables sollten frühzeitig entsprechende Tests (mit)definiert werden und mit den korrespondierenden Elementen vernetzt werden, beispielsweise Funktionsnachweise, Simulationsergebnisse oder Abnahmetests.

Meilensteine mit geeigneten Deliverables versehen

Meilensteine sind für die Steuerung von Produktentstehungsprozessen von besonderer Bedeutung, da sie die **Überprüfung des Projektfortschritts** zu bestimmten Zeitpunkten ermöglichen. Hierzu müssen Meilensteine mit geeigneten Deliverables versehen werden. In heutigen interdisziplinären Produktentstehungsprozessen besteht hierbei der Anspruch, dass die Deliverables eines jeden Meilensteins einerseits die Bewertung des Projektfortschritts unter Berücksichtigung sämtlicher Entwicklungsaktivitäten ermöglichen. Andererseits müssen Deliverables zergliedert werden können, um eine Bewertung des Projektfortschritts auch zwischen Meilensteinen zu ermöglichen. Im Sinne des Spiral-Prozesses empfiehlt es sich daher, sämtliche Meilensteine mit Deliverables aus (möglichst) allen Partialmodellen zu versehen. So sollte ein Meilenstein nicht nur Deliverables in Form von Tests physischer Gestalt beinhalten, sondern beispielsweise auch verabschiedete Anwendungsfälle oder Funktionsstrukturen.

Phasen mittels PE-Aktivitäten verfeinern und operationalisieren

Projektphasen untergliedern ein Projekt in der Regel (nur) in vergleichsweise grobe Abschnitte. Eine Operationalisierung der Projektphasen auf der Ebene der Akteure bzw. Stakeholder ist hierdurch jedoch nicht bzw. nur unmittelbar gegeben. Daher müssen Projektphasen mittels Produktentstehungsaktivitäten verfeinert und hierdurch **operationalisiert** werden. Hierbei gilt es unbedingt zu berücksichtigen, dass die verfeinernden Produktentstehungsaktivitäten entsprechend dem integrierten Produktentstehungsmodell **iPeM** zu einem **Spiral-Prozess** führen und nicht zu einem (rein) sequenziellen Stage-Gate-Prozess. Weiterhin sollten einzelne Aktivitäten mit Verantwortlichkeiten versehen werden.

Entscheidungen zur Vorbereitung und Dokumentation von Entscheidungssituationen nutzen

Mit Hilfe des Elementtyps Entscheidung können **Entscheidungssituationen sowohl vorbereitet als auch dokumentiert** werden. Im Vorfeld einer Entscheidungssituation können entsprechende Inhalte (alternative Lösungen, Bewertungen, etc.) unterhalb eines Elements vom Typ Entscheidung gesammelt werden. Um den offenen Status der Entscheidung zu modellieren, wird das Attribut Status genutzt. Wurde eine Entscheidung getroffen, werden die Positionen in der Hierarchie des Entscheidungselements und der gewählten Lösung vertauscht und der Status des Entscheidungselements entsprechend angepasst. Hierdurch wird sowohl ein kontinuierlicher Ideenspeicher realisiert, als auch Entscheidungen inkl. vorhandener Alternativen dokumentiert.

Anwendungsfälle zur Extraktion von Zielen und Anforderungen nutzen

Anwendungsfälle sind in besonderem Maße dazu geeignet, **Ziele und Anforderungen zu identifizieren**, da sie die Interaktionen des zu entwickelnden Systems mit den Systemnutzern in den Mittelpunkt der Betrachtung rücken. Hierdurch rücken die möglichen technischen Lösungen zur Realisierung des Anwendungsfalls vorerst in den Hintergrund, wodurch eine gute Basis zur Integration verschiedener Disziplinen sowie des Kunden in die Anwendungsfallmodellierung geschaffen ist. Auf der Basis modellierter Anwendungsfälle lassen sich Ziele und Anforderungen extrahieren, modellieren und mit den Anwendungsfall vernetzen.

Funktionen als Mittler zwischen Anforderungen und Gestalt/Implementation nutzen

Das Partialmodell der Funktionen und die hierin abbildbare Funktionshierarchie und Funktionsstruktur kann als **Mittler** zwischen Anforderungen und der realisierenden Gestalt bzw. Implementation genutzt werden. Dies kann insbesondere in Projekten hilfreich sein, in denen bisher unbekannte Anforderungen realisiert werden müssen oder eine unbekannte Gestalt bzw. Implementation zur Realisierung der Anforderungen Verwendung finden soll. Dabei kann über die Funktionshierarchie bzw. -struktur eine Integration der verschiedenen am Produktentstehungsprozess beteiligten Disziplinen erfolgen, wodurch eine frühzeitige Überprüfung der Anforderungen sowie eine interdisziplinäre Suche nach geeigneten Lösungen zur Realisierung der Anforderungen möglich werden. Die Funktionshierarchie bzw. -struktur kann hierbei auf Basis der formulierten (funktionalen) Anforderungen oder der bestehenden Gestalt/Implementation hergeleitet werden.

Wichtige Begründungen und Dokumente/Informationen separat modellieren

Im Rahmen des Attributs Freitext lassen sich eine Vielzahl relevanter Informationen bezüglich eines Elements modellieren. Hierunter fallen auch Begründungen über die (aktuelle) Ausprägung des Elements sowie mittels Hyperlinks eingebundene Informationen bzw. Dokumente. Sind diese Begründungen bzw. **Dokumente/Informationen von besonderer Wichtigkeit**, sollten die entsprechenden Inhalte in einem separaten Element des Typs Begründung bzw. Dokument/Information modelliert werden. Hierdurch wird die Präsenz der Inhalte wesentlich erhöht, wodurch eine zielgerichtete Extraktion der jeweiligen Inhalte möglich wird.

7.2.3 Extraktion entscheidungsrelevanter Informationen

Die Regeln zur Extraktion entscheidungsrelevanter Informationen stellen Empfehlungen dar, wie auf modellierte Inhalte des Zielsystems und angrenzender Partialmodelle zugegriffen werden kann, um auf Basis der **extrahierten Informationen Schlüsse ziehen** zu können, die für einen Entwickler nicht auf der Hand liegen.

Baumstruktur zur Exploration/Elementsicht für Details nutzen

Auf Basis der hierarchischen Baumstruktur können die modellierten Inhalte des Zielsystems und angrenzender Partialmodelle **effizient exploriert werden**. Dies ist zum einen aufgrund der kurzen und prägnanten Benennung der Elemente möglich, die eine schnelle Erfassung der (möglichen) Inhalte eines Elements erlauben. Zum anderen führen die Relation zwischen hierarchisch benachbarten Elementen dazu, dass entlang der Hierarchie eine Konkretisierung bzw. Abstraktion der Inhalte stattfindet, wodurch die Exploration teilweise vorausgesehen werden kann und entsprechend zielgerichteter erfolgt. Die effiziente Exploration (bzw. Navigation) durch die hierarchische Baumstruktur wird ergänzt durch die Möglichkeit, die jeweiligen Inhalte eines Elements (Attribut Freitext) in einer alternativen Darstellungsform unmittelbar explorieren zu können (vgl. Abschnitt 7.3).

Suchbegriffe für einfache Suchanfragen nutzen

Soll auf ein bestimmtes und bekanntes Element bzw. dessen Inhalt zugegriffen werden, bietet sich die Verwendung einer Suchanfrage mittels eines Suchbegriffs an. Hierbei kann die Voraussetzung der **Bekanntheit des gesuchten Elements** dadurch abgeschwächt werden, indem eine automatische Vervollständigung der eingegebenen Zeichen eines Suchbegriffs auf Basis bestehender Elementbezeichnungen erfolgt (Autovervollständigung). Weiterhin können auch nur Fragmente eines Suchbegriffs verwendet werden, um die Liste der Suchergebnisse zu vergrößern. Ebenso können mehrere Suchbegriffe verwendet werden, um auf Basis einer Und-Verknüpfung der Suchbegriffe die Liste der Suchergebnisse gezielt einzuschränken.

Elementtypen und Attribute für komplexere Suchanfragen nutzen

Komplexere Suchanfragen sind insbesondere dann erforderlich, wenn die möglichen Suchergebnisse nicht antizipiert werden können. Dies ist in der Regel der Fall, wenn nicht ein Element gefunden, sondern eine **Antwort gegeben werden soll**. In solchen Fällen sollten die Elementtypen und Attribute bzw. deren Ausprägungen genutzt werden, um komplexere Suchanfragen zu formulieren. Hierbei resultieren insbesondere aus der kombinierten Verwendung des Elementtyps und der verschiedenen Attribute komplexe Suchanfragen, auf deren Basis relevante Antworten aus dem modellierten Zielsystem extrahiert werden können.

Änderungsdatum zur Identifikation geänderter/neuer Inhalte nutzen

Zur Identifikation geänderter bzw. neuer Inhalte kann eine Suchanfrage nur auf Basis oder in Kombination mit einer **Datumsangabe** erfolgen, auf dessen Basis die entsprechenden Elemente als Suchergebnis ausgegeben werden.

Knotensuche als zusätzliche Einschränkung nutzen

Die Baumstruktur kann auf Basis der zugrundeliegenden Hierarchie und der einhergehenden Konkretisierung der Inhalte genutzt werden, um Suchanfragen

gezielt auf **Bereiche der Hierarchie** unterhalb eines bestimmten Knotens zu beschränken.

7.3 Werkzeug zur Modellierung von Zielsystemen

Das Werkzeug zur Modellierung von Zielsystemen baut auf den Erkenntnissen aus den Untersuchungen zur Modellierung von Zielsystemen⁴¹⁶ auf und ist darauf ausgerichtet, die Methode zur Strukturierung und Modellierung von Zielsystemen⁴¹⁷ möglichst einfach und schlank in Form eines **Prototyps** umzusetzen. Hierdurch soll das Werkzeug eine Anwendung und damit die Validierung der entwickelten Methode ermöglichen. Das (Software-)Werkzeug muss hierbei nicht dem Anspruch einer kommerziellen Verwertbarkeit genügen, jedoch muss es für einen Entwickler einfach und intuitiv anwendbar sein und zur Laufzeit ausreichend stabil und fehlerfrei arbeiten.

Die Entwicklung des Werkzeugs orientiert sich an einigen wesentlichen Zielen und hieraus abgeleiteter Anforderungen, die erreicht bzw. realisiert werden sollten. Aus diesen Zielen und Anforderungen leiten sich die wesentlichen Funktionen der Software und deren konkrete Implementation ab. Daher wird im Folgenden auf die Ziele des Werkzeugs als Ausgangspunkt der Entwicklung eingegangen. Anschließend wird das Ergebnis der Werkzeugentwicklung anhand der grafischen Benutzeroberfläche der Software dargelegt.

7.3.1 Ziele der Werkzeugentwicklung

Die formulierten Ziele sind für die Entwicklung des Software-Werkzeugs im Wesentlichen bestimmend. Daneben müssen einige Randbedingungen berücksichtigt werden, insbesondere die zur Verfügung stehenden Entwicklungsressourcen.

Windows PC – Desktop und Tablet

Im geschäftlichen Umfeld verzeichnet das Betriebssystem Windows von Microsoft den größten Marktanteil⁴¹⁸, weshalb die Implementation primär auf dieses Betriebssystem ausgerichtet ist und die Implementierung in C# (C-Sharp) vorgenommen wird. Da ein zunehmender Trend hin zu mobilen, insbesondere Tablet-PCs zu verzeichnen ist, soll die Software sowohl auf einem **Desktop-PC als auch auf einem Tablet-PC** verwendet werden können, wodurch die Software beispielsweise unmittelbar in Sitzungen zum Einsatz kommen kann. Aufgrund der zur Verfügung stehenden Entwicklungsressourcen und der Einheitlichkeit der Bedingung

⁴¹⁶ vgl. Kapitel 6

⁴¹⁷ vgl. Abschnitt 7.1 und 7.2

⁴¹⁸ vgl. Statista 2014

soll dieses Ziel nicht mit verschiedenen grafischen Benutzeroberflächen realisiert werden. Entsprechend muss die Benutzeroberfläche sowohl für Desktop-Betrieb (großer Bildschirm, Maus- und Tastatureingabe) als auch für Tablet-Betrieb (kleinerer Bildschirm, Touchscreeneingabe) geeignet sein.

Einfaches und einheitliches Bedienkonzept

Ein wesentlicher Schlüssel für einen intuitiven Umgang mit einer Software liegt in der Einfachheit der Bedienung. Hierbei liefert ein **einheitliches Bedienkonzept** einen entscheidenden Beitrag. Entsprechend soll die Software ein möglichst einfaches, in jedem Fall jedoch ein einheitliches Bedienkonzept aufweisen. So sollen Interaktionen mittels Maus bzw. Touchscreen einheitlich entweder eine Aktion ausführen (linke Maustaste bzw. kurzes Antippen des Touchscreens) oder ein Kontextmenü zur Bearbeitung öffnen (rechte Maustaste bzw. langes Antippen des Touchscreens).

Einheitlichkeit der Elemente

Um einen intuitiven und flexiblen Umgang mit den Elementen des Zielsystems realisieren zu können, soll der Aufbau aller Elemente sowie der hierauf anwendbaren Methoden **einheitlich und vom Elementtyp bzw. dessen Attributen und Relationen unabhängig** sein (beispielsweise Ziel oder Phase/Aktivität). Hieraus resultiert, dass ein Element erst aufgrund seiner (veränderlichen) Parametrisierung charakterisiert wird, wodurch eine erhebliche Flexibilität erreicht werden kann. Weiterhin wird folglich nicht zwischen einem Element mit vorhandenem Inhalt (Inhalts-Element) und einem Element ohne vorhandenem Inhalt (Container-Element), das lediglich zur Strukturierung dient, unterschieden. Hierdurch wird ebenfalls die Flexibilität der Software gesteigert, da sich ein Element von einem Inhalts- zu einem Container-Element (und umgekehrt) entwickeln kann, je nach Bedarf der Modellierung.

Struktur- und Elementansicht

Die Untersuchungen zur Modellierung von Zielsystemen haben gezeigt, dass (mindestens zwei) unterschiedliche Darstellungsformen bzw. Ansichten der Zielsystem-Elemente erforderlich sind. Eine Ansicht muss dabei einen Überblick über die vorhandenen Elemente des Zielsystems geben, indem die mittelbare Systemumgebung eines bestimmten Elements dargestellt wird (**Strukturansicht**⁴¹⁹). Eine andere Ansicht muss hingegen auf ein bestimmtes Element fokussieren, so dass dessen Inhalt, Attribute und unmittelbare Systemumgebung sichtbar werden

⁴¹⁹ vgl. Abschnitt 6.2

(**Elementansicht**⁴²⁰). Hierbei müssen beide Ansichten auf Basis eines bestimmten Elements ineinander überführbar sein.

Die Strukturansicht soll ausgehend von einem bestimmten Element, **das Element selbst und dessen Systemumgebung** mittels einer Baumstruktur darstellen. Hierbei sollen die einzelnen Elemente lediglich auf Basis ihres Bezeichners (Name) dargestellt werden. Weiterhin soll nicht die vollständige Systemumgebung eines Elements dargestellt werden, da dies stets das ganze Zielsystem umfassen würde. Um die Systemumgebung auf eine **Menge von erfassbaren Elementen** zu beschränken, werden neben dem bestimmten Element nur solche Elemente angezeigt, die bezüglich des bestimmten Elements eine der folgenden Regeln erfüllen (vgl. Abbildung 7-4):

- Ein Sub-, Subsub oder Subsubsubelement des bestimmten Elements
- Das Superlement des bestimmten Elements
- Ein Sub-, Subsub oder Subsubsubelement des Superlements
- Das Supersuperelement des bestimmten Elements

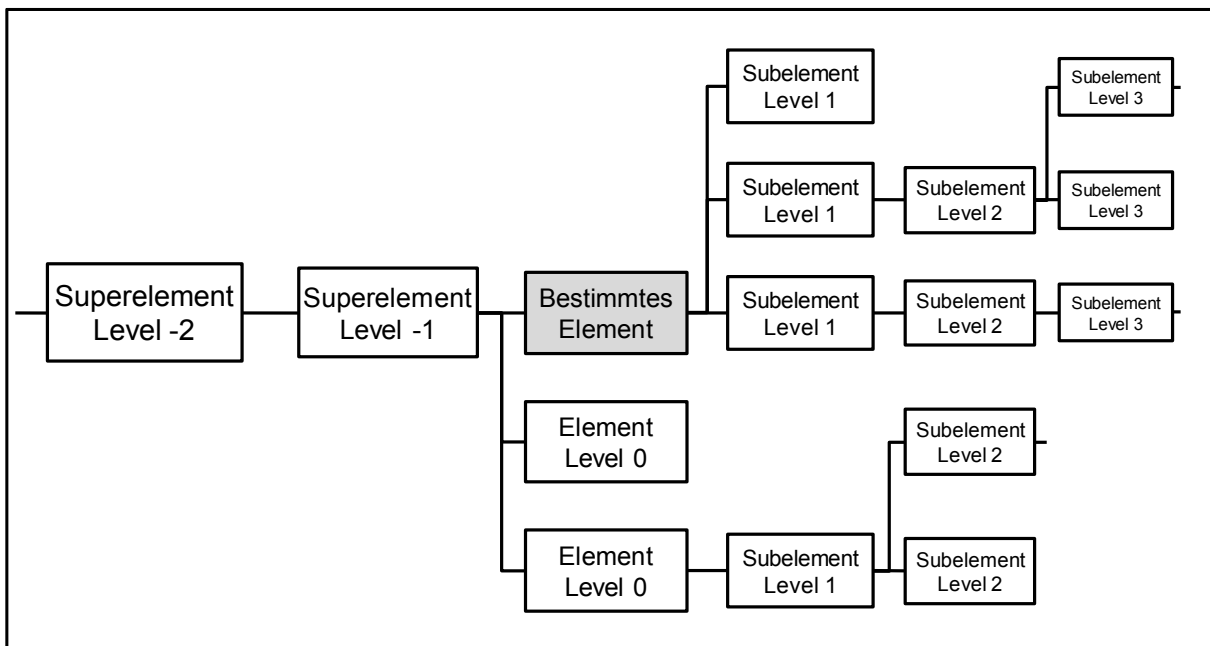


Abbildung 7-4: Strukturansicht des Werkzeugs

Grenzen an ein Element weitere Super- oder Subelemente an, die jedoch aufgrund dieser Regeln nicht dargestellt werden, soll zur Indikation stattdessen eine (halbe) Verbindungslinie dargestellt werden. Weiterhin soll die Darstellungsgröße der Elemente entsprechend ihrer Hierarchiestufe bezüglich dem bestimmten Element skaliert werden. Hierdurch nehmen das bestimmte Element verfeinernde Elemente

⁴²⁰ vgl. Abschnitt 6.1

umso weniger Raum ein, je weiter sie das bestimmte Element detaillieren, wodurch die Übersichtlichkeit der Baumstruktur verbessert werden soll (vgl. Abbildung 7-4). Ein beliebiges, dargestelltes Element soll per linker Maustaste bzw. kurzem Antippen des Touchscreens zum bestimmten Element werden.

Die **Elementansicht** soll primär ein **bestimmtes Element darstellen**, d.h. neben dem Elementnamen soll hauptsächlich der Inhalt des Attributs Freitext dargestellt werden (vgl. Abbildung 7-5). Weiterhin soll der Name des Superelements sowie die Namen von bis zu sechs Subelemente dargestellt werden, um **effizient durch die Hierarchie der Elemente navigieren** zu können. Sind mehr als sechs Subelemente vorhanden, sollen jeweils bis zu sechs weitere Subelemente mittels Bildlauf dargestellt werden. Zusätzlich sollen die elementspezifischen Ausprägungen der Attribute Status, Wichtigkeit, Autorisation und Zeitpunkt der letzten Änderung angezeigt werden. Ein beliebiges, dargestelltes Element soll per linker Maustaste bzw. kurzem Antippen des Touchscreens zum bestimmten Element werden.

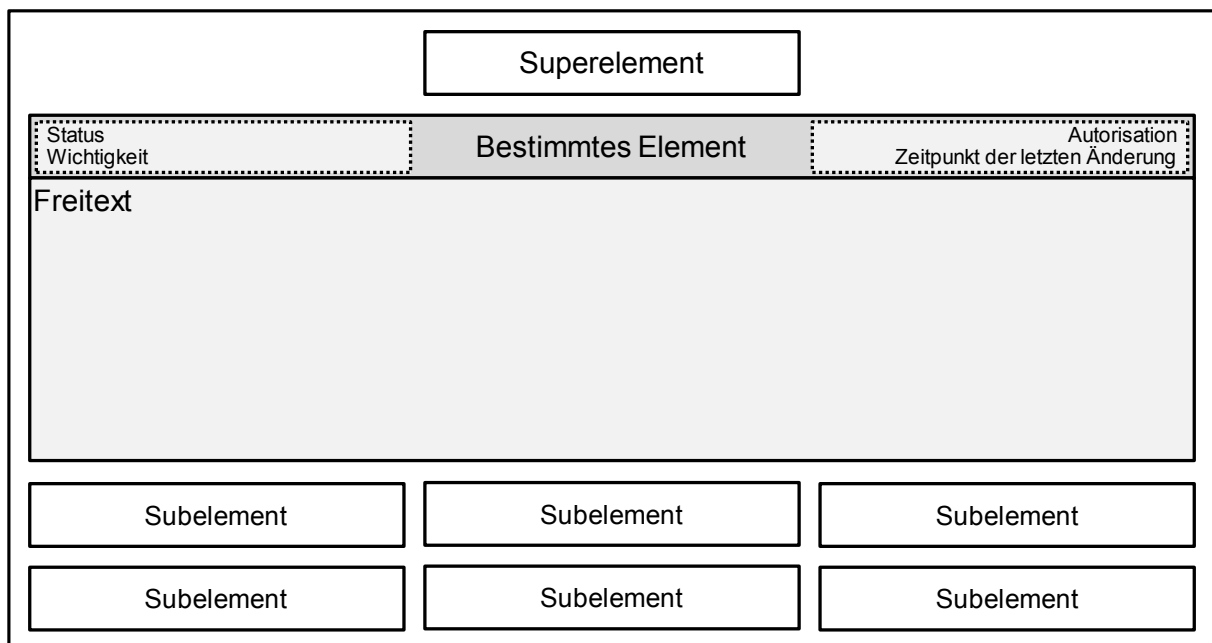


Abbildung 7-5: Elementansicht des Werkzeugs

Wird auf das bestimmte Element ein Doppelklick per linker Maustaste bzw. Touchscreen durchgeführt, soll das bestimmte Element in der alternativen Ansicht dargestellt werden.

Nutzbare Relationen

Die explizite Abbildung von Beziehungen zwischen Elementen des Zielsystems ist für die Zielsystemmodellierung ein immanenter Bestandteil. **Relationen müssen daher einfach und intuitiv genutzt** werden können. Insbesondere müssen die in Beziehung stehenden Elemente eines bestimmten Elements sowie die jeweiligen Relationen unmittelbar ersichtlich sein. Daher sollten die Namen der in Beziehung

stehenden Elemente sowie die jeweiligen Relationen zusammen mit der Struktur- bzw. Elementansicht dauerhaft dargestellt werden können (vgl. Abbildung 7-6). Durch diese Darstellung soll es darüber hinaus möglich sein, ein in Beziehung stehendes Element per linker Maustaste bzw. kurzem Antippen des Touchscreens als bestimmtes Element zu definieren. Hierdurch soll effizient durch das Beziehungsgeflecht von Elementen des Zielsystems navigiert werden können.

Suchfunktion

Die Suchfunktion ist ein wichtiger Bestandteil der Software, da sie einen wesentlichen Beitrag zur Extraktion entscheidungsrelevanter Informationen leistet (vgl. Abbildung 7-6). Hierbei muss die Suchfunktion die Suche mittels mehrere **alternativer Kriterien** erlauben und auch **Kombinationen von Kriterien** unterstützen:

- Suchbegriff(e) inkl. Autovervollständigung auf Basis der Namen der vorhandenen Elemente. Mehrere Suchbegriffe sind UND-verknüpft auszuwerten.
- Ausprägungen der Elementtypen sowie der Attribute Status, Wichtigkeit und Autorisation
- Datum der letzten Änderung (listet alle Elemente mit diesem oder jüngerem Datum der letzten Änderung auf)
- Knotensuche (listet alle Elemente unterhalb dieses Knotens auf, inkl. den Knoten selbst)

Werden einzelne Suchkriterien miteinander kombiniert, sind diese UND-verknüpft auszuwerten, d.h. es ist die Schnittmenge der Suchergebnisse aller Suchkriterien zu bilden.

Das Suchergebnis soll anstelle der Struktur- bzw. Elementansicht dargestellt werden. Hierbei sollen die einzelnen Suchtreffer auf Basis der Namen der jeweiligen Elemente tabellarisch aufgelistet werden. Die Auflistung soll zuerst nach Elementtyp, dann alphabetisch sortiert sein. Ein so dargestelltes Element soll per linker Maustaste bzw. kurzem Antippen des Touchscreens zum bestimmten Element werden und in der zuletzt genutzten Ansicht (Struktur- oder Elementansicht) dargestellt werden.

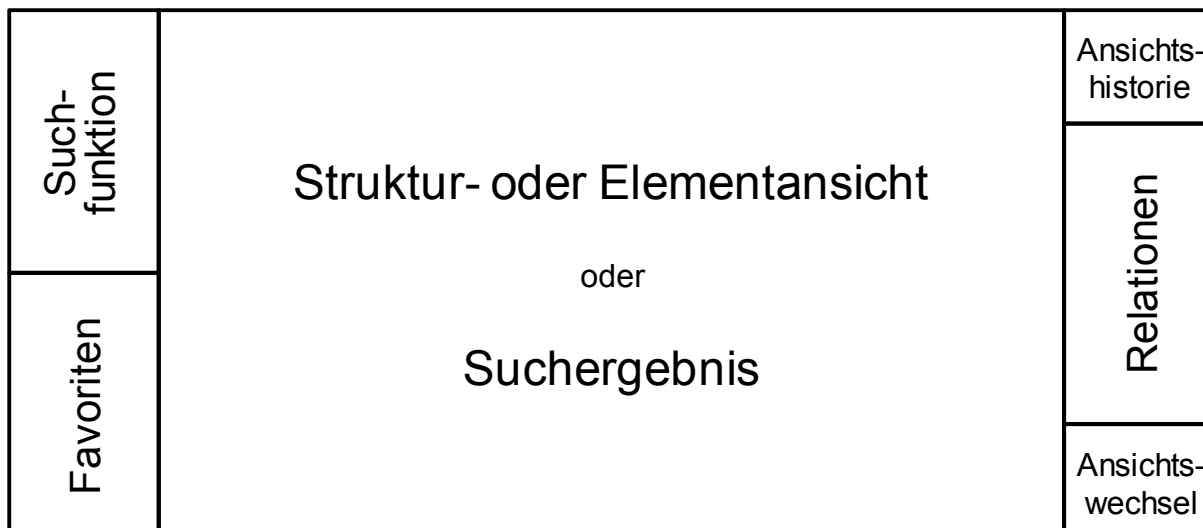


Abbildung 7-6: Elemente der grafischen Benutzeroberfläche des Werkzeugs

Favoriten

Unterschiedliche Entwickler haben entsprechend ihrer Zuständigkeit bzw. Verantwortlichkeit einen unterschiedlichen Fokus auf das modellierte Zielsystem. Folglich ist davon auszugehen, dass einzelne Entwickler schwerpunktmäßig an bestimmten Elementen des Zielsystems ein besonderes Interesse haben. Da in einer hierarchischen Baumstruktur die Elemente von Interesse nicht zwangsläufig unmittelbar beieinander liegen, sollen **nutzerspezifische Verlinkungen** zu einzelnen Elementen des Zielsystems (Favoriten) erstellt werden können. Hierdurch sollen favorisierte Elemente effizient aufgerufen werden können.

Ansichtshistorie

Im Rahmen der Modellierung von Zielsystemen ist davon auszugehen, dass neben den hierarchischen Beziehungen zwischen einzelnen Elementen auch etliche Relationen modelliert sind, so dass die Konnektivität im Zielsystemmodell in Ansätzen dem tatsächlichen Beziehungsgeflecht im Zielsystem entspricht. Aufgrund dieser Konnektivität der Elemente im Zielsystemmodell ist weiterhin davon auszugehen, dass im Zuge der Exploration des Zielsystems stark nicht-lineare Elementsequenzen auftreten, die zu einem Orientierungsverlust führen können. Daher muss es möglich sein, **entlang vollzogener Elementsequenzen navigieren** zu können, ohne die Sequenz selbständig reproduzieren zu müssen (Ansichtshistorie). Entsprechend ist eine Lösung zu realisieren, wie sie bereits von gängigen Internet-Browsern zur Rückwärts- und Vorwärtsnavigation besuchter Internetseiten bekannt ist. Hierbei muss eine konkrete Suchanfrage und das resultierende Suchergebnis wie ein Element behandelt werden, so dass zwischen Suchergebnis und einzelnen Suchtreffern vor- und zurücknavigiert werden kann.

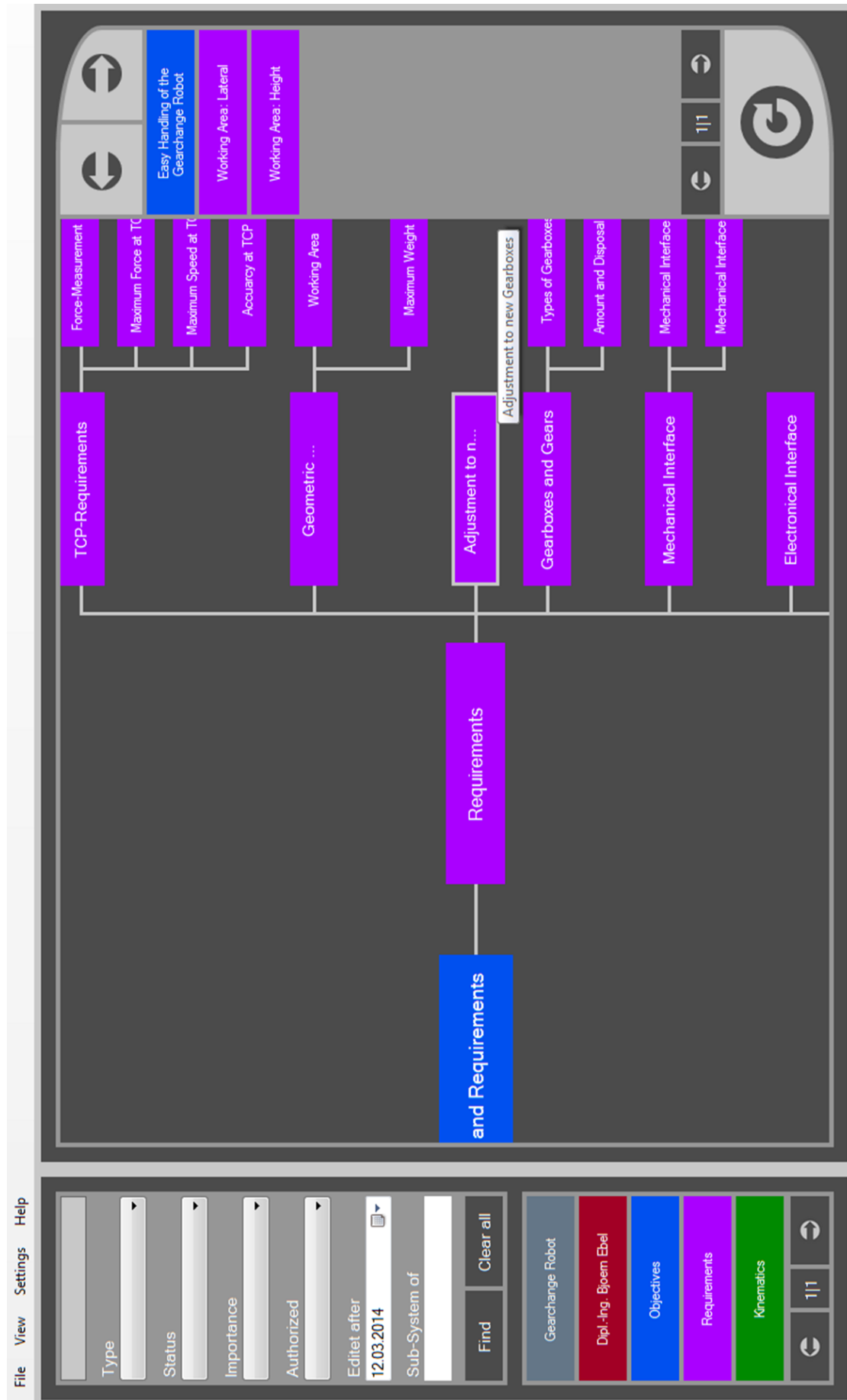


Abbildung 7-7: Strukturansicht des SystemModelers

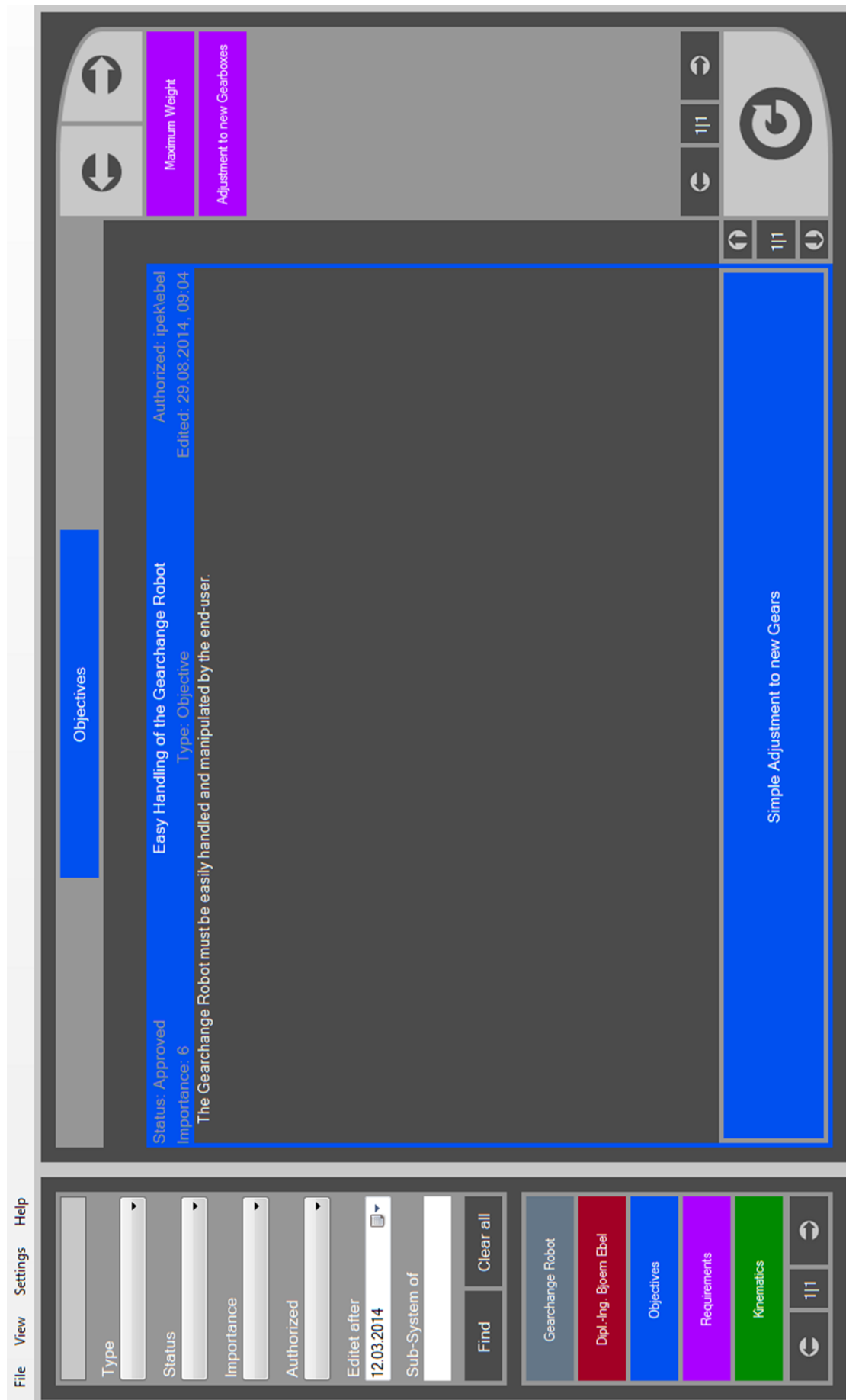


Abbildung 7-8: Elementansicht des SystemModelers

7.3.2 Ergebnis der Werkzeugentwicklung

Das Ergebnis der Werkzeugentwicklung ist eine Software (**SystemModeler**), die primär für die Modellierung von Zielsystemen ausgelegt und geeignet ist, die jedoch aufgrund ihrer generischen bzw. anpassbaren Funktionalitäten prinzipiell auch zur Modellierung beliebiger Systeme eingesetzt werden kann (vgl. Abbildung 7-7, Abbildung 7-8 und Abbildung 7-9).

Im Zuge der Implementierung wurde auf eine **Anpassbarkeit** bzw. Erweiterbarkeit der Software insbesondere zur Laufzeit Wert gelegt. So können die Ausprägungen der Attribute Elementtyp, Status und Wichtigkeit projektspezifisch definiert werden. Weiterhin können die Struktur- und Elementansicht nach individuellen Bedürfnissen angepasst werden, um insbesondere auf unterschiedliche Bildschirmgrößen und Betrachtungsabstände reagieren zu können. Um auch eine Anpassbarkeit bzw. Erweiterbarkeit auf der Ebene des Programmcodes sicher zu stellen, wurde eine modulare und möglichst transparente Programmstruktur realisiert (objektorientierte Programmierung). Projektspezifische Daten werden in (Projekt-)Dateien gespeichert, wobei das **Datenformat auf dem XML-Standard** basiert. Hierdurch wird ebenfalls die Transparenz und Erweiterbarkeit der Software unterstützt, jedoch resultiert hieraus unmittelbar, dass einzelne Projektdaten **zeitgleich nur von einem Benutzer** bearbeitet werden können.

Auf eine weitere Beschreibung der Software wird an dieser Stelle verzichtet und stattdessen in den **Anhang** verwiesen, da die Software primär als Werkzeug zur Umsetzung und Validierung der Methode zur Modellierung von Zielsystemen dient.

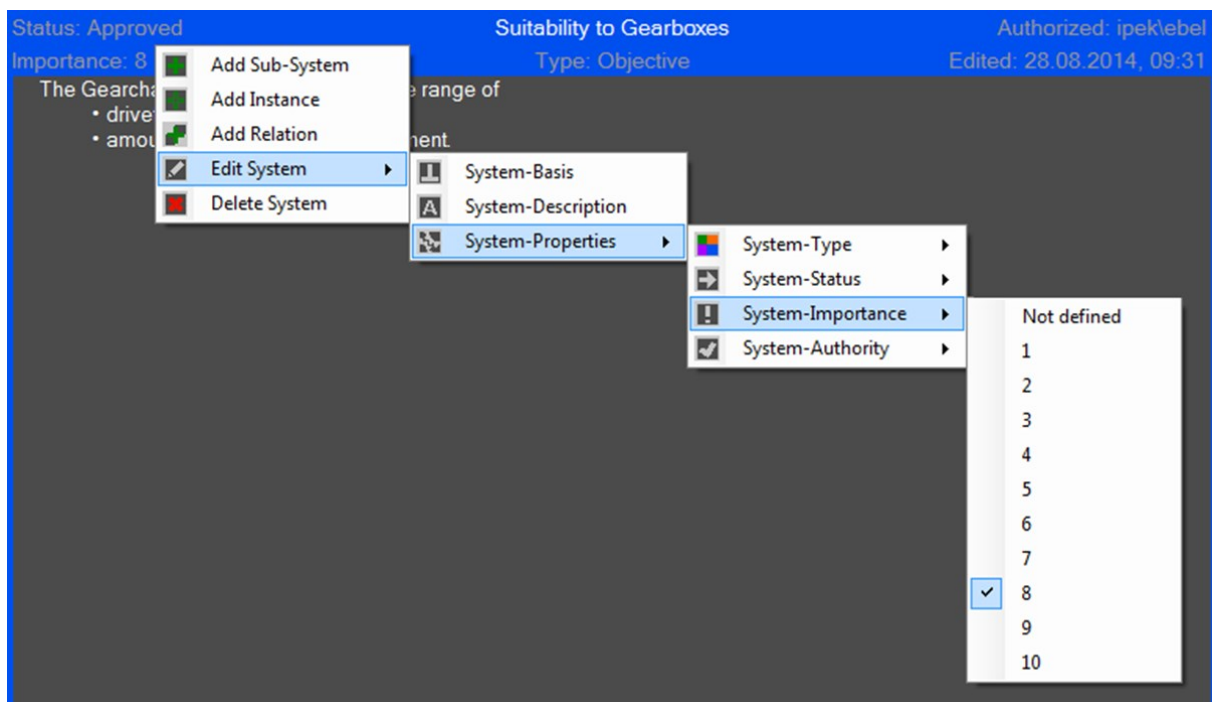


Abbildung 7-9: Kontextmenü des SystemModeler

7.4 Validierung der Methode zur Modellierung von Zielsystemen

Die Validierung der Methode zur Modellierung von Zielsystemen findet in zwei Schritten statt und schließt die Entwicklung der Methode ab. In einem ersten Schritt wird die Methode dazu genutzt, das Zielsystem eines IPEK-internen Entwicklungsprojekts (Schaltroboter) **retrospektiv zu modellieren**. Hierdurch sollen insbesondere mögliche Inkonsistenzen der Methode zur Modellierung von Zielsystemen aufgedeckt werden. Weitere bzw. umfassendere Erkenntnisse sind auf Basis einer retrospektiven Modellierung nicht zu erwarten. Jedoch bildet die Gewissheit einer prinzipiell konsistenten Methode zur Zielsystemmodellierung eine solide Basis für deren **Einsatz in der Entwicklungspraxis** und damit dem zweiten Schritt der Validierung. Hierbei wird die Methode im Rahmen eines Entwicklungsprojekts bei der AVL List GmbH im Bereich der Li-Ion-Batterieentwicklung eingesetzt. Durch den produktiven Einsatz unter Praxisbedingungen sollen relevante Aussagen zur Tauglichkeit der Methode zur Modellierung von Zielsystemen abgeleitet werden.

7.4.1 Retrospektive Modellierung

Rahmenbedingungen der Untersuchung

Die erste Untersuchung der Methode zur Modellierung von Zielsystemen findet anhand einer retrospektiven Modellierung des **Zielsystems eines Schaltroboters** zur Prüfstandsautomatisierung statt (vgl. Abbildung 7-10). Das Projekt hatte zum Ziel, einen Schaltroboter als Ergänzung der Prüfstandsautomatisierung des IPEK-Antriebssystemprüfstands PLP (Powertrain-in-the-Loop-Prüfstand) zu entwickeln, der State-of-the-Art-Schaltrobotern in den Belangen Aktordynamik und Sensorauflösung überlegen ist und dessen Schnittstellen dem IPEK vollständig zugänglich sind. Das Projekt wurde von wissenschaftlichen **Mitarbeitern des IPEK** verantwortet und durchgeführt, wodurch ein tiefer Einblick in die tatsächlich durchlaufenen Entwicklungsaktivitäten möglich ist. Darüber hinaus war der Untersuchende im ersten Drittel des zwölfmonatigen Projekts vollständig involviert und verantwortete die **Spezifikation des Schaltroboters** (integrierte Lasten- und Pflichtenhefterstellung). Die retrospektive Modellierung des Zielsystems findet nach Abschluss des Projekts zusammen mit dem Projektleiter statt und erfolgt im Rahmen von drei halbtägigen Sitzungen. Hierbei wird das Zielsystem auf Basis der Erinnerungen des Projektleiters und des Untersuchenden in Kombination mit vorhandenen Dokumenten modelliert. Die Zielsetzung der Modellierung besteht in der nachvollziehbaren Dokumentation der Entwicklungsaktivitäten sowie dem Resultat der Entwicklung.



Abbildung 7-10: IPEK-Schaltroboter zur Prüfstandsautomatisierung

Ergebnisse der Untersuchung

Im Zuge der retrospektiven Modellierung des Zielsystems wurden rund **100 Elemente und etwa 50 Relationen** modelliert, wodurch schätzungsweise ein Drittel der Elemente und Relationen abgebildet wurden, die für die Entwicklung des Schaltroboters tatsächlich erforderlich gewesen sind. Hierbei wurden jedoch **alle definierten Partialmodelle, Elementtypen, Relationen und Attribute genutzt**.

Im Rahmen der retrospektiven Modellierung konnte die **Konsistenz nachgewiesen** und die prinzipielle Tauglichkeit der Methode zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung gezeigt werden. Alle relevant erscheinenden Aspekte konnten einfach und effizient in das Modell des Zielsystems eingebunden werden. Dies ist zum einen auf die Performanz der Methode sowie des Werkzeugs zur Zielsystemmodellierung zurückzuführen. Zum anderen kann dies auch darauf zurückgeführt werden, dass das Modell des Zielsystems vom Entwickler der Methode selbst erstellt wurde. Im Zuge der Modellierung und (fiktiven) Nutzung des Modells zeigten sich kleinere Verbesserungspotenziale, die jedoch nicht zwingend im Vorfeld des Einsatzes der Methode in der Entwicklungspraxis umgesetzt werden müssen.

- Partialmodell Gestalt/Implementation disziplinspezifisch untergliedern
Das Partialmodell Gestalt/Implementation kann entsprechend der disziplinspezifischen Lösungsanteile untergliedert werden in **Mechanik-Struktur, Elektronik-Struktur, Regler-Struktur und Software-Struktur**. Hierdurch

können die disziplinspezifischen Lösungsanteile separat verantwortet und modelliert werden, wodurch eine effizientere Modellierung realisiert werden kann. Gleichzeitig muss jedoch verstärkt auf eine Koordination der Lösungsanteile sowie eine Kommunikation der Verantwortlichen geachtet werden.

- Test in Testfall und Testumgebung untergliedern
Sowohl das Partialmodell als auch der Elementtyp **Test kann in Testfall und Testumgebung untergliedert** werden. Hierdurch wird eine klarere Trennung zwischen dem Ablauf eines Test (Testfall) und der ausführenden Hard- und/oder Software (Testumgebung) möglich, wodurch insbesondere die Wiederverwendbarkeit einzelner Elemente verbessert wird. Im Rahmen der Modellierung kann der Aspekt der Testfallinterpretation mit in den Testfall aufgenommen werden. Hierbei muss jedoch beachtet werden, dass die Testfallinterpretation ggf. abhängig ist von der ausführenden Testumgebung.
- Suchfunktion erweitern
Die Suchfunktion des Werkzeugs zur Zielsystemmodellierung ermöglichte einen effizienten Zugriff auf die modellierten Zielsysteminhalte. Hierbei zeigte sich, dass eine **Integration von booleschen Operatoren und Vergleichsoperatoren** die Leistungsfähigkeit der Suchfunktion weiter steigern kann.

Insgesamt zeigen die Ergebnisse der Untersuchung, dass die Methode und das Werkzeug zur Zielsystemmodellierung die Modellierung eines nutzenstiftenden Zielsystems erlauben.

7.4.2 Einsatz in der Entwicklungspraxis

Rahmenbedingungen der Untersuchung

Die zweite Untersuchung der Methode zur Modellierung von Zielsystemen findet im Rahmen der **Li-Ion-Batterieentwicklung bei der AVL List GmbH** statt. Im Rahmen eines Kundenprojekts nutzt ein funktionsverantwortlicher **Projektmanager** die Methode, um die für ihn relevanten technischen und entwicklungsprozessspezifischen Informationen abzubilden und zu vernetzen. Der Projektmanager erwartet sich hierdurch einerseits ein verbessertes Verständnis des Zielsystems und auf Basis des explizit vorliegenden Zielsystemmodells die Extraktion entscheidungsrelevanter Informationen. Zum anderen soll das Zielsystemmodell als Dokumentation der Entwicklungsaktivitäten dienen, die vom Projektmanager verantworteten werden.

Die Methode zur Modellierung von Zielsystemen wird über einen Zeitraum von (mindestens) **vier Monaten** genutzt. In diesem Zeitraum befindet sich das Kundenprojekt in der **ersten von drei Prototypenentwicklungszyklen**, die durch eine stetige Verfeinerung des Zielsystems sowie einen hohen Anteil virtueller Entwicklungs- und Validierungsaktivitäten gekennzeichnet ist. Zu Beginn der vier

Monate wird dem Projektmanager via Webkonferenz die Methode und das Werkzeug in einer zweistündigen Sitzung anhand eines bestehenden Zielsystemmodells (Schaltroboter) erläutert. In der Folge finden drei weitere, jeweils etwa einstündige Webkonferenzen statt, um den Fortschritt der Zielsystemmodellierung zu diskutieren und Unklarheiten in der Modellierung zu beseitigen. Zum Abschluss des viermonatigen Untersuchungszeitraums findet eine zweistündige Webkonferenz zwischen dem Projektmanager und dem Untersuchenden statt, um die Ergebnisse der Untersuchungen zusammen zu fassen und zu bewerten.

Ergebnisse der Untersuchung

Die Ergebnisse der Untersuchung werden im Wesentlichen im Rahmen einer zweistündigen Webkonferenz zusammengefasst und bewertet. Hierbei wird die Webkonferenz anhand eines **Fragenkatalogs** strukturiert, der die relevanten Aspekte hinsichtlich der Methode und des Werkzeugs zur Zielsystemmodellierung⁴²¹ abfragt. Folgende Fragen wurden gestellt:

- War(en) die Strukturierung/die Modellierungsregeln/das Werkzeug zur Zielsystemmodellierung generell nutzenstiftend? Wo liegen Stärken, wo Schwächen? Welche Verbesserungen werden vorgeschlagen?
- Waren die einzelnen Aspekte der Strukturierung/der Modellierungsregeln/des Werkzeugs nutzenstiftend? Wo liegen Stärken, wo Schwächen? Welche Verbesserungen werden vorgeschlagen?
- Wie beurteilen Sie den Aufwand und den Nutzen sowie das Verhältnis aus Nutzen und Aufwand der Methode zur Modellierung von Zielsystemen?
- Welche übergreifenden Verbesserungsvorschläge für die Methode zur Modellierung von Zielsystemen können Sie nennen?
- Worin sehen Sie aktuelle und zukünftige Herausforderungen der Zielsystemmodellierung?

Hierbei wurde der Fragebogen im Vorfeld der Webkonferenz an den Projektmanager übersandt. Im Verlauf der Webkonferenz wurde der Fragebogen zur Strukturierung des Interviews genutzt (**semistrukturiertes Interview**). In die folgende Auflistung der Untersuchungsergebnisse fließen neben den Inhalten der abschließenden Webkonferenz auch die Inhalte der drei einstündigen Webkonferenzen mit ein.

Die Erläuterung der Methode zur Modellierung von Zielsystemen im Rahmen der zweistündigen Webkonferenz zu Beginn der Untersuchung war dem Projektmanager zufolge vollkommen ausreichend, um selbständig ein nutzenstiftendes Zielsystem modellieren zu können. Dies kann zum einen auf die **einfache und intuitive**

⁴²¹ vgl. Abschnitt 7.1, 7.2 und 7.3 sowie die jeweiligen Unterabschnitte.

Anwendbarkeit der Methode (sowie des Werkzeugs) zurückgeführt werden. Zum anderen ist der funktionsverantwortliche Projektmanager als äußerst erfahren in der Produkt- und Prozessmodellierung einzustufen. Entsprechend wurden die drei einstündigen Webkonferenzen im Verlauf der Untersuchung hauptsächlich zur Diskussion und kaum zur Beseitigung von Unklarheiten genutzt. Im Verlauf des Untersuchungszeitraums war der Projektmanager insgesamt in der Lage, rund **250 Elemente und 500 Relationen** zu modellieren.

Die **Strukturierung des Zielsystems sowie der angrenzenden Partialmodelle** (vgl. Abbildung 7-1) **wurde als äußerst nutzenstiftend eingeschätzt**, da sämtliche als relevant eingestuft Informationen sinnvoll und strukturiert durch den Projektmanager abgebildet werden konnten. Hierbei wurde insbesondere die **hierarchische Strukturierung innerhalb der Partialmodelle in Kombination mit dem Konzept der Relationen und assoziierten Kopien als wertvoll erachtet**. Das Test-Partialmodell wurde im Rahmen der Untersuchung lediglich dazu genutzt, die zu erfüllenden Tests, die in Relation zu Deliverables stehen, abzubilden. Dies ist dem aktuellen Projektfortschritt geschuldet, da zum Zeitpunkt des Interviews noch keine (physischen) Tests durchgeführt wurden und entsprechend auch noch keine Testergebnisse vorliegen, die eine Auswirkung auf definierte Ziele und Anforderungen haben. Der Projektmanager plant jedoch, die Tests sowie die Testinterpretationen zu modellieren und mit den betreffenden Zielen und Anforderungen in Beziehung zu setzen. Zu Beginn der Untersuchung vermutete der Projektmanager, dass ein explizites **Ressourcen-Partialmodell** erforderlich sei. Im Rahmen der viermonatigen Untersuchung wurden jedoch lediglich Stakeholder, denen Verantwortlichkeiten zugewiesen wurden, abgebildet. Dieser Umstand lässt sich dadurch begründen, dass sämtliche monetären Ressourcen (Budget) über ein separates System abgebildet werden, das mit dem firmenweiten ERP-System (Enterprise-Ressource-Planing) interagiert. Damit einher geht auch die Verwaltung von Personalressourcen, die unmittelbar mit dem System zur Verwaltung der monetären Ressourcen interagiert. Insofern scheint ein **Stakeholder-Partialmodell vorerst ausreichend** zu sein. Das Partialmodell der Funktionen wurde zum einen dazu genutzt, eine **Funktionshierarchie** als Beitrag zur (virtuellen) Testdefinition zu erstellen. Weiterhin wurde das Partialmodell genutzt, um in einem separaten Bereich relevante **Zustände und Zustandswechsel** zu modellieren.

Die **Elementtypen** konnten in weiten Teilen zur Modellierung des Zielsystems verwendet werden. Hierbei wurden in 86% aller Fälle die Elementtypen Deliverable (21%), Gestalt/Implementation (14%), Anforderung (13%), Funktion (10%), Dokumente/Information (10%), Stakeholder (7%), Anwendungsfall (6%) und Ziel (5%) verwendet. Aufgrund der Notwendigkeit, vom Auftraggeber definierte Randbedingungen (Anforderungen mit sehr hohem Härtegrad) als solche abbilden zu

können, wurde der **Elementtyp Randbedingung** hinzugefügt. Des Weiteren wurde durch den Projektmanager der Wunsch geäußert, zwischen **lösungsneutralen, vertragswirksamen Kundenanforderungen (Kundenzielen) und lösungsspezifischen Anforderungen**, die sich erst im Verlauf des Projekts ergeben, differenzieren zu können. Hierdurch kann klarer unterschieden werden, was erreicht und wie es realisiert werden soll. Darüber hinaus untergliederte der Projektmanager den **Elementtyp Phase/Aktivität in seine beiden Bestandteile** und fügte mit dem **Action-Item** ein Elementtyp hinzu, der die Phasen und (Produktentstehungs-)Aktivitäten nochmals verfeinert. Dies wurde erforderlich, um aktuell offene Kundenanfragen modellieren und nachverfolgen zu können. Der Elementtyp **Entscheidung wurde nicht genutzt**, da Entscheidungen von großer Tragweite nicht vom Projektmanager verantwortet wurden. Darüber hinaus hielt der Projektmanager die **Modellierung ganzer Entscheidungen für zu aufwendig** und beschränkte sich entsprechend auf die Verlinkung entscheidungsrelevanter Dokumente in das modellierte Zielsystem. Zusätzlich wurde der Elementtyp **Dokument/Information** genutzt, um relevante Zusatzinformationen zu einzelnen Elementen hinzuzufügen (Heftnotiz).

Die **Relationen** konnten ohne jegliche Änderungen zur Modellierung der Wechselwirkungen zwischen den Elementen des Zielsystems genutzt werden. Hierbei wurden in 91% aller Fälle die Relationen Gerichtete Relation (24%), Erfüllungsrelation (23%), Verantwortlichkeitsrelation (19%) Treiberrelation (11%), Unterstützungsrelation (8%) und Ableitungsrelation (6%) verwendet. Ergänzend wurde durch den Projektmanager eine **Vorgänger/Nachfolgerrelation** (gefolgt von/folgt) definiert, um einzelne Aktivitäten in eine zeitliche bzw. kausale Abhängigkeit setzen zu können. Die **Verifizierungs- und Validierungsrelation wurde nicht genutzt**, da Tests (noch) nicht ausreichend konzipiert bzw. modelliert wurden, um diese Relation sinnvoll einsetzen zu können.

Die **Attribute** zur Charakterisierung der einzelnen Elemente des Zielsystems wurden insgesamt als sinnvoll und geeignet bewertet. Aufgrund der Tatsache, dass das Modell des Zielsystems nur vom Projektmanager selbst genutzt wurde, entschied sich der Projektmanager die Attribute **Autorisation, Wichtigkeit und Status nicht zu nutzen**. Das Attribut Freitext und hierbei insbesondere die Möglichkeit **Hyperlinks** nutzen zu können, wurden für äußerst zweckdienlich gehalten. In einigen Situationen wünschte sich der Projektmanager jedoch, **formatierten Text, Tabellen und Bilder** einfügen zu können.

Die **Regeln zur Modellierung** von Zielsystemen wurden im Verlauf der Untersuchung vollumfänglich angewandt und in diesem Zuge als hilfreich und korrekt eingeschätzt. Hierbei wurde insbesondere die Regel bezüglich **Mehrfachverwendung und Sichten von Elementen mittels assoziierter Kopien als äußerst**

nutzenstiftend bewertet. Ebenso wurden die Regeln zur **Extraktion entscheidungsrelevanter Informationen genutzt und als hilfreich** sowie korrekt beurteilt.

Das **Werkzeug** zur Modellierung von Zielsystemen wurde insgesamt als **äußerst geeignet** angesehen, die entsprechende Methode einfach und intuitiv umzusetzen. Auf Basis der eigenen Erfahrung des Projektmanagers mit alternativen Modellierungswerkzeugen, insbesondere Microsoft Word, Excel, Sharepoint und Atego Artisan Studio (SysML) wurde der Software eine **sehr gute Benutzerfreundlichkeit bei gleichzeitig vorhandener Leistungsfähigkeit** bescheinigt. Der Projektmanager hob insbesondere die **Struktur- und Elementansicht** sowie deren Kombination als Besonderheit hervor (vgl. Abbildung 7-4 und Abbildung 7-5). Hierbei vermittelt die **Strukturansicht einen guten Überblick** über die Umgebung eines bestimmten Elements ohne dabei den Benutzer mit zu vielen Informationen (Elementen) zu überfordern. Die **Elementansicht setzt den Inhalt eines Elements in den Mittelpunkt** der Betrachtung und erlaubt darüber hinaus über die Darstellung der Super- und Subelemente eine **effiziente Navigation** durch die Baumstruktur. Ebenfalls positiv bewertet wurde die (dauerhafte) **Darstellung der Relationen** eines Elements. Hierbei wurde jedoch negativ bewertet, dass der **Relationstyp einer Relation nur als Tooltip** (Mous Over) angezeigt wird und nicht dauerhaft in Verbindung mit dem in Beziehung stehenden Element. Weiterhin wurde die **Verschachtelung des Kontextmenüs** bemängelt, die zu einer Effizienzverringerung in der Bearbeitung des Zielsystemmodells führt (vgl. Abbildung 7-9). Hingegen wurde die **Leistungsfähigkeit der Suchfunktion** positiv hervorgehoben. Durch die Kombination aus Suchbegriff(en), gewählten Attributen, der Knotensuche sowie dem Datum der letzten Änderung konnten entscheidungsrelevante Informationen extrahiert werden. Hierbei wäre eine Extraktion ohne das Zielsystemmodell bzw. das Werkzeug prinzipiell möglich gewesen, jedoch nur unter hohem Zeitaufwand und unter Zuhilfenahme anderer Methoden und Werkzeuge. Hierbei wurde deutlich, dass die Leistungsfähigkeit der Suchfunktion nochmals erheblich gesteigert werden kann, wenn **Relationen als Teil einer Suchanfrage** verwendet werden können. Die Favoriten und die Ansichtshistorie wurden regelmäßig genutzt und als unauffälliger, jedoch äußerst nützlicher Bestandteil der Software gesehen.

Insgesamt wurde die Methode zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung als **nutzenstiftend** bewertet. Hierbei wurde auch das Software-Werkzeug zur Umsetzung der Methode gelobt, das trotz einiger **Schwächen in der Benutzerfreundlichkeit** (Usability) bereits eine **produktive Nutzung der Methode in der Entwicklungspraxis** erlaubt. Hierbei ist der Projektmanager davon überzeugt, dass der investierte **Aufwand zur Modellierung des Zielsystems gerechtfertigt** ist, da entsprechende (nur schwer quantifizierbare) Mehrwerte entstehen. Diese Mehrwerte beruhen dabei auf einer **integrativen**

Modellierung aller relevanten Aspekte des ZHO-Modells. Entsprechend dieser Überzeugung wird der Projektmanager die Methode bzw. das Software-Werkzeug auch weiterhin im Projekt nutzen, um für ihn relevante technische und entwicklungsprozessspezifische Informationen abzubilden und zu vernetzen.

Um zukünftigen Herausforderungen der Zielsystemmodellierung gewachsen zu sein, müssten geeignete **(Zielsystem-)Referenzstrukturen** geschaffen werden, die auf den vorliegenden Partialmodellen, Elementtypen, Relationen und Attributen aufbauen und diese weiter verfeinern. Auf Basis einer modellbasierten Modellierung könne hierdurch sowohl eine **Wiederverwendbarkeit** der Strukturen und ggf. der Inhalte realisiert werden, als auch eine **leistungsfähige Funktionalität zur Extraktion entscheidungsrelevanter Informationen** geschaffen werden.

7.4.3 Abschließende Bewertung

Die beiden Untersuchungen haben die **Tauglichkeit der Methode** zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung gezeigt. Ein Entwickler mechatronischer Produkte ist mit Hilfe der Methode in der Lage, selbstständig und effizient ein Zielsystem zu modellieren und auf dieser Basis einen Mehrwert zu schaffen.

Die Methode ermöglicht über die vorhandene Strukturierung bzw. über die Partialmodelle und Elementtypen die **einfache und intuitive Abbildung Zielsystem-relevanter Informationen**. Gleichzeitig dienen die Partialmodelle als Indikatoren, welche **Wissens- und Definitionslücken** im Verlauf eines Produktentstehungsprozesses noch geschlossen werden müssen. Mit Hilfe der definierten Relationen als Teil einer Zielsystem-Referenzstruktur lassen sich relevante Beziehungen zwischen den einzelnen Elementen des Zielsystems abbilden, wodurch eine **Nachvollziehbarkeit bzw. Durchgängigkeit im Zielsystem** realisiert werden kann. Hierbei können die gewählten Attribute dazu beitragen, eine genauere Charakterisierung der modellierten Elemente vorzunehmen. Die Untersuchungen zeigen, dass eine **weitere Differenzierung bzw. Detaillierung der Zielsystem-Referenzstruktur** dazu führen kann, heutigen und zukünftigen Herausforderungen der Zielsystemmodellierungen adäquat zu begegnen. Hierbei müssen die Regeln zur Zielsystemmodellierung weiterhin einfach und eindeutig sein und sich in der Modellbildung dynamisch anwenden lassen, um mit der Komplexität heutiger und zukünftiger Zielsysteme umgehen zu können. Die Untersuchungen haben weiterhin gezeigt, dass das eingesetzte **Werkzeug bereits in der Lage ist, die Methode zur Modellierung von Zielsystemen umzusetzen** und in der Entwicklungspraxis die Erstellung nutzenstiftender Modelle von Zielsystemen und angrenzender Partialmodelle zu ermöglichen. Hierbei realisiert das Werkzeug über die vorhandene **Suchfunktion bereits Mehrwerte**, die über die reine Dokumentation des

Zielsystems hinausgehen (Extraktion entscheidungsrelevanter Informationen). Dennoch hat auch das Werkzeug noch erhebliche **Verbesserungspotenziale**, die in Kombination mit einer Weiterentwicklung der Methode zur Modellierung von Zielsystemen gehoben werden müssen.

8 Zusammenfassung und Ausblick

In diesem abschließenden Kapitel werden die Ergebnisse der Arbeit zusammengefasst und ein Ausblick auf mögliche weiterführende Forschungsaktivitäten der Zielsystemmodellierung gegeben.

8.1 Zusammenfassung

Im Rahmen der vorliegenden Arbeit wurde eine Methode zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung entwickelt. Die Methode basiert auf der Systemtheorie, greift wesentliche Aspekte der Modelltheorie auf und berücksichtigt, dass der Mensch im Mittelpunkt der Produktentstehung steht. Hierdurch wird der Komplexität heutiger Produkte und entsprechender Produktentstehungsprozesse Rechnung getragen und eine Anwendbarkeit der Methode in der interdisziplinären Produktentstehung sichergestellt.

Die identifizierten Erfolgsfaktoren der Interdisziplinarität bilden hierbei das Fundament der vorliegenden Arbeit. Nur auf Basis der erfolgreichen Zusammenarbeit diverser Menschen unterschiedlicher Disziplinen kann ein relevantes Zielsystem entstehen und damit ein letztendlich erfolgreiches Produkt entwickelt werden. Hierbei liefern die Erfolgsfaktoren einen wichtigen Beitrag zur Gestaltung einer erfolgreichen interdisziplinären Zusammenarbeit zwischen verschiedenen Menschen. Auf dieser Basis ist es möglich, das Verständnis von Zielen und Zielsystemen in der interdisziplinären Produktentstehung zu verbessern und damit einen Beitrag zu einer effizienten und effektiven Modellierung von Zielsystemen zu leisten. Durch die Identifikation und Darlegung der verschiedenen Modellierungszwecke von Zielen in Verbindung mit einer Methode zur expliziten wie auch impliziten Beurteilung von Zielen ist es dem einzelnen Entwickler mechatronischer Produkte möglich, selbstständig und effizient ein nutzenstiftendes Zielsystem zu modellieren.

Um die Anwendbarkeit der Methode zur Zielsystemmodellierung in der industriellen Praxis sicherstellen zu können, wurden zwei Untersuchungen im Kontext realer Produktentwicklungsprojekte durchgeführt. Auf dieser Basis wurden relevante Einflüsse auf die Modellierung von Zielsystemen identifiziert, die erst im tatsächlichen Prozess der interdisziplinären Produktentstehung sichtbar werden. Hierdurch konnten relevante Erkenntnisse über die Zielsystemmodellierung in der Entwicklungspraxis gewonnen werden, die insbesondere die Strukturierung von Zielsystemen sowie das konkrete Modellieren und Extrahieren relevanter Informationen mittels eines entsprechenden Werkzeugs determinieren.

Auf Basis dieser wichtigen Zwischenergebnisse wurde die Methode zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung entwickelt. Hierzu wurden praxisrelevante Partialmodelle, Elementtypen und Relationen definiert, deren Kombination eine geeignete Zielsystem-Referenzstruktur bilden und die prinzipiell in der Lage sind, sämtliche Zielsystem-relevanten Informationen abzubilden. Darüber hinaus wurden Attribute identifiziert, die eine ergänzende Charakterisierung der Elemente erlauben und damit zur Operationalisierung der Zielsystem-Referenzstruktur beitragen. Hierbei ermöglicht die explizit vorliegende Struktur von Zielsystemen, sämtliche Zielsystem-relevanten Informationen ganzheitlich und durchgängig zu modellieren und das resultierende Zielsystemmodell zur Extraktion entscheidungsrelevanter Informationen zu nutzen. Um den Entwickler bei der Modellierung des Zielsystems im Rahmen eines interdisziplinären Produktentstehungsprozesses zu unterstützen, wurden Regeln und ein Werkzeug zur Zielsystemmodellierung entwickelt, die einfach und intuitiv anwendbar und auf die Zielsystem-Referenzstruktur abgestimmt sind.

Die Tauglichkeit der entwickelten Methode zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung wurde abschließend im Rahmen einer Validierung in der Entwicklungspraxis bestätigt. Neben einigen Verbesserungspotenzialen der Methode und insbesondere des Werkzeugs wurden hierbei auch zukünftige Herausforderungen der Zielsystemmodellierung aufgezeigt.

8.2 Ausblick

Weiterführende Forschungsaktivitäten der Zielsystemmodellierung müssen an der weiter steigenden Komplexität technischer Produkte und korrespondierender Produktentstehungsprozesse ausgerichtet sein und gleichzeitig den Menschen als zentralen Gestalter der Produktentstehung berücksichtigen. Eine der zentralen Herausforderungen wird darin bestehen, den wachsenden Ansprüchen und Aufwänden der Zielsystemmodellierung einen adäquaten Nutzen der Modellierung gegenüber zu stellen. Hieraus lassen sich drei unterschiedliche, jedoch parallel zu verfolgende Forschungsaktivitäten ableiten.

Erstens sollten Forschungsaktivitäten durchgeführt werden, die eine Minimierung der Ansprüche der Zielsystemmodellierung zum Ziel haben. Noch scheint nicht geklärt zu sein, ob die Zielsystemmodellierung zukünftig zentral, dezentral oder in einer Mischform erfolgen wird. Entsprechend ist noch nicht geklärt, ob notwendige Kompetenzen zur Zielsystemmodellierung zukünftig zentral bei einzelnen Experten oder verteilt im Unternehmen bzw. der Entwicklungsabteilung vorhanden sein müssen. In beiden Fällen ist es jedoch wünschenswert, dass die Modellierung von Zielsystemen möglichst einfach und intuitiv erfolgt. Dies kann möglicherweise realisiert werden, indem die Komplexität der Methode zur Zielsystemmodellierung

dynamisch auf Basis der Komplexität des zu entwickelnden Produkts bzw. dessen Teilaspekte skaliert werden kann.

Zweitens sollten Forschungsaktivitäten durchgeführt werden, die eine Minimierung des Aufwands der Zielsystemmodellierung zum Ziel haben. Ein wesentlicher Schlüssel dürfte hierbei in der durchgängigen, modellbasierten Abbildung aller (Zielsystem-)relevanter Aspekte liegen. Hierdurch können einmal modellierte Strukturen und Inhalte konsequent wiederverwendet werden, ohne dass Redundanzen und damit Inkonsistenzen entstehen können. Die Aufwandsreduzierung kann hierbei zum einen innerhalb eines bestimmten Produktentstehungsprozesses wirksam werden, zum anderen jedoch insbesondere im Rahmen der Produktgenerationsentwicklung, die eine Wiederverwendung der überwiegenden Strukturen und Inhalte des Zielsystems erlaubt und damit einen kontinuierlichen Reifeprozess der Zielsystem-Referenzstruktur ermöglicht.

Drittens sollten Forschungsaktivitäten durchgeführt werden, die den Nutzen einer Zielsystemmodellierung erhöhen. Hierbei entsteht der Nutzen einer (Zielsystem)Modellierung insbesondere dadurch, dass Erkenntnisse über ein Original gewonnen werden können, die ohne das Modell nicht zugänglich sind. Hierbei wird die Komplexität zukünftiger technischer Produkte dazu führen, dass auch interdisziplinär kooperierende Menschen die Komplexität nicht mehr alleine überschauen können. Eine explizite und durchgängige Modellierung von Zielsystemen wird folglich notwendig werden, um rechnerunterstützt mit der vorherrschenden Komplexität umgehen zu können. Hierbei kommt der Extraktion von Informationen aus dem modellierten Zielsystem eine besondere Bedeutung zu, um den Mensch, der auch zukünftig im Mittelpunkt der Produktentstehung stehen wird, mit für ihn (entscheidungs-)relevanten Informationen zu versorgen. Im Anschluss an die vorliegende Arbeit sollte daher an folgender Forschungsfrage geforscht werden:

Ausblick-Forschungsfrage: Extraktion relevanter Informationen

Wie können relevante Informationen aus einem Zielsystemmodell mittels semantischer Funktionalitäten extrahiert werden?

- Welche Elementtypen, Relationen und Attribute müssen zusätzlich definiert werden und wie sind diese im Rahmen der Zielsystemmodellierung zu verwenden?
- Wie müssen Suchanfragen formuliert und ausgewertet werden, um relevante Informationen aus einem Zielsystemmodell zu extrahieren?
- Wie müssen relevante Informationen aufbereitet werden, um möglichst effizient und fehlerfrei von einem Menschen erfasst und genutzt werden zu können?

9 Literaturverzeichnis

Adler, Frost & Gross 2011

Adler, F., Frost, I., und Gross, D. (2011) *Die Qual der Wiki-Wahl - Wikis für Wissensmanagement in Organisationen*. In Open Journal of Knowledge Management, IV/2011.

Albers 2010

Albers, A. (2010) *Five Hypotheses and a Meta Model of Engineering Design Processes*. In 8th International Symposium on Tools and Methods of Competitive Engineering TMCE 2010. Ancona, Italy.

Albers 2011

Albers, A. (2011) *Der Entwickler im Zentrum des Systems der Produktentstehung*. In Stuttgarter Symposium für Produktentwicklung SSP 2011, Stuttgart.

Albers, Behrendt & Ott 2010

Albers, A., Behrendt, M. und Ott, S. (2010) *Validation - Central Activity to Ensure Individual Mobility*. In Proceedings Fisita 2010 World Automotive Congress, Budapest, Hungary.

Albers & Braun 2011a

Albers, A. und Braun, A. (2011) *Der Prozess der Produktentstehung*. In Henning, F. und Moeller, E. (Hrsg.) *Handbuch Leichtbau – Methoden, Werkstoffe, Fertigung*. Hanser, München, S. 5-30.

Albers & Braun 2011b

Albers, A. und Braun, A. (2011) *A Generalized Framework to Compass and to Support Complex Product Engineering Processes*. In International Journal of Product Development, Bd. 15, Nr. 1/2/3, S. 6-25.

Albers & Braun 2012

Albers, A. und Braun, A. (2012) *Towards Handling Complexity - Testing the iPeM Process Modeling Approach*. In 9th International Symposium on Tools and Methods of Competitive Engineering TMCE 2012. Karlsruhe, Germany.

Albers, Braun & Ebel 2013

Albers, A., Braun, A., Ebel, B. (2013) *Aktivitätenbasierte Analyse von Produktentstehungsprozessen*. In Maurer, M.; Schulze, S.-O. (Hrsg.) *Tag des Systems Engineerings 2013*. Carl Hanser Verlag, München, S. 415-424.

Albers, Braun & Muschik 2010

Albers, A., Braun, A. und Muschik, S. (2010) *Uniqueness and the Multiple Fractal Character of Product Engineering Processes*. In Heisig, P., Clarkson, P. J., Vajna, S. (Hrsg.) *Modelling and Management of Engineering Processes*. Springer, London, S. 14-26.

Albers et al. 2005

Albers, A., Burkardt, N., Meboldt, M. und Saak, M. (2005) *SPALTEN Problem Solving in the Product Development*. In 15th International Conference on Engineering Design ICED'05, Melbourne, Australia.

Albers, Bursac & Wintergerst 2015

Albers, A., Bursac, N. und Wintergerst, E. (2015) *Produktgenerationsentwicklung – Bedeutung und Herausforderungen aus einer entwicklungsmethodischen Perspektive*. In Stuttgarter Symposium für Produktentwicklung SSP 2015, Stuttgart. Abstract angenommen.

Albers & Düser 2009

Albers, A., Düser, D. (2009) *Integration von Simulation und Test am Beispiel Vehicle-in-the-loop auf dem Rollenprüfstand und im Fahrversuch*. In 3. Internationales Symposium für Entwicklungsmethodik – Herausforderungen im Spannungsfeld neuer Antriebskonzepte. Wiesbaden.

Albers & Düser 2010

Albers, A. und Düser, T. (2010) *A New Process for Configuration and Application of Complex Validation Environments Using the Example of Vehicle-in-the-Loop at the Roller Test Bench*. In International Mechanical Engineering Congress & Exposition IMECE 2010, Vancouver, Canada.

Albers, Düser & Ott 2008

Albers, A., Düser, T. und Ott, S. (2008) *X-in-the-loop als integrierte Entwicklungsumgebung von komplexen Antriebsystemen*. In 8. Tagung Hardware-in-the-loop-Simulation. Haus der Technik, Kassel.

Albers, Deigendesch & Meboldt 2008

Albers, A., Deigendesch, T. und Meboldt, M. (2008) *Handling Complexity – A Methodological Approach Comprising Process and Knowledge Management*. In 7th International Symposium on Tools and Methods of Competitive Engineering TMCE 2008, Izmir, Turkey, S. 1247-1258.

Albers, Ebel & Alink 2011

Albers, A., Ebel, B. und Alink, T. (2011) *Erfolgsfaktoren der Interdisziplinarität. Ein Bericht*. In Banse, G., Fleischer, L.G. (Hrsg) *Wissenschaft im Kontext: Inter- und Transdisziplinarität in Theorie und Praxis*. Trafo, Berlin.

Albers, Ebel & Lohmeyer 2012

Albers, A., Ebel, B. und Lohmeyer, Q. (2012) *Systems of Objectives in Complex Product Development*. In 9th International Symposium on Tools and Methods of Competitive Engineering TMCE 2012, Karlsruhe.

Albers, Ebel & Sauter 2010

Albers, A., Ebel, B. und Sauter, C. (2010) *Combining Process Model and Semantic Wiki*. In 11th International Design Conference DESIGN2010, Dubrovnik.

Albers, Klingler & Ebel 2013

Albers, A., Klingler, S. und Ebel, B. (2013) *Modeling Systems of Objectives in Engineering Design Practice*. In Proceedings of the 19th International Conference on Engineering Design ICED'13, Seoul, South-Korea.

Albers & Lohmeyer 2012

Albers, A. und Lohmeyer, Q. (2012) *Advanced Systems Engineering – Towards a Model-Based and Human-Centered Methodology*. In 9th International Symposium on Tools and Methods of Competitive Engineering TMCE 2012, Karlsruhe, Germany.

Albers, Lohmeyer & Ebel 2011

Albers, A., Lohmeyer, Q. und Ebel, B. (2011) *Dimensions of Objectives in Interdisciplinary Product Development Projects*. In Proceedings of the 18th International Conference on Engineering Design ICED'11, Copenhagen, Denmark, Vol.2, S. 256-265.

Albers & Meboldt 2006

Albers, A. und Meboldt, M. (2006) *A new Approach in Product Development, based on systems engineering and systematic problem solving*. AEDS 2006 WORKSHOP, Pilsen – Czech Republic.

Albers & Meboldt 2007

Albers, A. und Meboldt, M. (2007) *IPEMM – Integrated Product Development Process Management Model Based on Systems Engineering and Systematic Problem Solving*. In 16th International Conference on Engineering Design ICED'07, Paris, France, No. 537.

Albers, Muschik & Ebel 2010

Albers, A., Muschik, S. und Ebel, B. (2010) *Einflüsse auf Entscheidungsprozesse in frühen Aktivitäten der Produktentstehung*. In Symposium für Vorausschau und Technologieplanung, Berlin.

Albers, Saak & Burkardt 2002

Albers, A., Saak, M. und Burkardt, N. (2002) *Gezielte Problemlösung bei der Produktentwicklung mit Hilfe der SPALTEN-Methode*. In 47. Internationales Wissenschaftliches Kolloquium: Maschinenbau und Nanotechnik – Hochtechnologie des 21. Jahrhunderts. Kretzschmar Verlag, Illmenau, S. 83-84.

Albers & Sadowski 2014

Albers, A. und Sadowski, E. (2014) *The Contact and Channel Approach (C&C²-A) – relating a system's physical structure to its functionality*. In Chakrabarti, A. und Blessing, L. (Hrsg.) *An Anthology of Theories and Models of Design: Philosophy, Approaches and Empirical Explorations*. Springer.

Albers, Sadowski & Marxen 2011

Albers, A., Sadowski, E. und Marxen, L. (2011) *A new Perspective on Product Engineering Overcoming Sequential Process Models*. In Birkhofer, H. (Hrsg.) *The Future of Design Methodology*. Springer, London.

Albers, Turki & Lohmeyer 2012

Albers, A., Turki, T. und Lohmeyer, Q. (2012) *Transfer of Engineering Experience by Shared Mental Models*. In 14th International Conference on Engineering and Product Design Education E&PDE 2012, Antwerp, Belgium, No.5101.

Albers & Wintergerst 2014

Albers A. und Wintergerst E. (2014) *The Contact and Channel Approach (C&C²-A): relating a system's physical structure to its functionality*. In Blessing, L. und Chakrabarti, A. (Hrsg.) *An Anthology of Theories and Models of Design: Philosophy, Approaches and Empirical Explorations*. Springer, London.

Albers & Zingel 2013

Albers, A. und Zingel, C. (2013) *Extending SysML for Engineering Designers by Integration of the Contact & Channel – Approach (C&C²-A) for Function-Based Modeling of Technical Systems*. In Proceedings of the Conference on Systems Engineering Research. Atlanta, Georgia, USA.

Alink 2010

Alink, T. (2010) *Bedeutung, Darstellung und Formulierung von Funktion für das Lösen von Gestaltungsproblemen mit dem C&C-Ansatz*. Dissertation, IPEK, Karlsruher Institut für Technologie (KIT).

Almefelt et a. 2006

Almefelt, L., Berglund, F., Nilsson, P. und Malmqvist, J. (2006) *Requirements management in practice: findings from an empirical study in the automotive industry*. In *Research in Engineering Design*, vol. 17, no. 3, S. 113-134.

Alt 2009

Alt, O. (2009) *Car Multimedia Systeme Modell-basiert testen mit SysML*. Vieweg+Teubner, Wiesbaden.

Alt 2012

Alt, O. (2012) *Modellbasierte Systementwicklung mit SysML*. Carl Hanser, München.

Angermeier 2015

Angermeier, G. (2015) *Aktivität*. URL <https://www.projektmagazin.de/glossarterm/aktivitaet>. Aktualisierungsdatum 27.01.2015.

Ashby 1974

Ashby, W. R. (1974) *Einführung in die Kybernetik*. Suhrkamp, Frankfurt am Main.

Avgoustinov 2007

Avgoustinov, N. (2007) *Modelling in Mechanical Engineering and Mechatronics: Towards Autonomous Intelligent Software Models*. Springer, London.

Badke-Schaub, Daalhuizen & Roozenburg 2011

Badke-Schaub, P., Daalhuizen, J. und Roozenburg, N. (2011) *Towards a Designer-Centred Methodology: Descriptive Considerations and Prescriptive Reflections*. In Birkhofer, H. (Hrsg.) *The Future of Design Methodology*, London, Springer, S. 181-197.

Baldwin & Clark 2000

Baldwin, C. Y., Clark, K. B. (2000) *Design rules: The power of modularity*. The MIT Press, Cambridge, MA.

Balzert 2000

Balzert, H. (2000) *Lehrbuch der Software-Technik*. 2. Auflage. Spektrum, Heidelberg.

Bertalanffy 1949

Bertalanffy, L. (1949) *Zu einer allgemeinen Systemlehre*. In *Biologia Generalis*, Nr.195, S. 114-129 (Nachdruck in Bleicher, K. (1972) *Organisation als System*. Gabler, Wiesbaden, S. 29-46).

Bertalanffy 1969

Bertalanffy, L. (1969) *General System Theory – Foundations, Development, Applications*. George Braziller, New York.

Biedermann, Diepold & Lindemann 2010

Biedermann, W., Diepold, K. J. und Lindemann, U. (2010) *Modellabstufungen zur Betrachtung struktureller und dynamischer Komplexität*. In Tag des Systems Engineerings 2010, Gesellschaft für Systems Engineering e.V. (Hrsg.), Carl Hanser, München.

Birkhofer & Jänsch 2003

Birkhofer, H. und Jänsch, J. (2003) *Interaction between Individuals*. In Lindemann, U. (Hrsg.) *Human Behaviour in Design – Individuals, Teams, Tools*, Springer, Berlin, S. 105-110.

Blessing & Chakrabarti 2009

Blessing, L. T. und Chakrabarti, A. (2009) *DRM, a Design Research Methodology*. Springer, London.

Boehm 1988

Boehm, B. (1988) *A Spiral Model of Software Development and Enhancement*. IEEE Computer, 1988, S. 61-72.

Bröhl 1995

Bröhl, A. P. (1995) *Das V-Modell - der Standard für die Softwareentwicklung*. 2. Auflage. München, Oldenbourg.

Brown & Blessing 2005

Brown, D. C. und Blessing, L. (2005) *The Relationship between Function and Affordance*. Proceedings of the ASME 2005 IDETC/CIE Conference, 24.-28. September 2005, Long Beach, California, USA.

Browning, Fricke & Negele 2006

Browning, T. R., Fricke, E. und Negele, H. (2006) *Key Concepts in Modelling Product Development Processes*. In *System Engineering*, Vol. 9, Nr. 2, S. 104-128, Wiley InterScience.

Burghardt 2008

Burghardt, M. (2008) *Projektmanagement – Leitfaden für die Planung, Überwachung und Steuerung von Projekten*. 8. Auflage, Publics Corporate Publishing, Erlangen.

Buzan & Buzan 2012

Buzan, T. und Buzan, B. (2002). *Das Mind-Map-Buch – Die beste Methode zur Steigerung Ihres geistigen Potenzials*. 5. Auflage. mvg, München.

Chalupnik, Wynn & Clarkson 2009

Chalupnik, M. J., Wynn D.C. und Clarkson P. J. (2009) *Approaches to Mitigate the Impact of Uncertainty in Development Processes*. In 17th International Conference on Engineering Design ICED'09, Stanford, USA, 2009, Vol.1, S. 459-470.

Chalupnik, Wynn & Clarkson 2013

Chalupnik, M.J., Wynn D. C. and Clarkson P. J. (2013) *Comparison of illities for protection against uncertainty in system design*. In *Journal of Engineering Design*. Taylor & Francis, London, UK.

Conant & Ashby 1970

Conant, R. C., Ashby, W. R. (1970) *Every Good Regulator of a System must be a model of that System*. In *International Journal of Systems Science*, Vol. 1, No. 2, S. 89-97.

Cooper 1990

Cooper, R. G. (1990) *Stage-Gate Systems: A New Tool for Managing New Products*. In *Business Horizons*, Vol. 33, No. 3, S. 44-54.

Cooper 1994

Cooper, R. G. (1994) *Perspective third-generation new product processes*. In *Journal of Product Innovation Management*, Bd. 11, Nr. 1, S. 3-14.

Cooper 2001

Cooper, R. G. (2001) *Winning at New Products*. 3rd edition, Perseus Publishing, Cambridge.

Daenzer & Huber 1996

Daenzer, W. F. und Huber, F. (1996) *Systems Engineering*, 10. Auflage. Orell Füssli Verlag Industrielle Organisation, Zürich.

Darlington & Culley 2002

Darlington, M. J. und Culley, S. J. (2002) *Current Research in the Engineering Design Requirement*. In *Journal of Engineering Manufacture*, Vol. 216, No. 3, S. 375-388.

Deming 1982

Deming, W. E. (1982) *Out of the Crisis*. MIT Press, Cambridge, MA.

Dörner 1998

Dörner, D. (1998) *Logik des Misslingens – Strategisches Denken in komplexen Situationen*. Rowohlt, Reinbeck bei Hamburg.

Dörner 2003

Dörner, D. (2003) *Die Logik des Misslingens. Strategisches Denken in komplexen Situationen*. 10. Auflage. Hamburg: Rowohlt Taschenbuch 2003.

Dutke 1994

Dutke, S. (1994) *Mentale Modelle: Konstrukte des Wissens und Verstehens. Kognitionspsychologische Grundlagen für die Software-Ergonomie*. Verlag für Angewandte Psychologie, Göttingen.

Earl, Johnson & Eckert 2005

Earl, C., Johnson, J. und Eckert, C. (2005) *Complexity*. In Clarkson, J. und Eckert, C. (Hrsg.) *Design Process Improvement - A Review of Current Practice*, Springer, London, S. 174-197.

Ebert 2012

Ebert, C. (2012) *Systematisches Requirements Engineering: Anforderungen ermitteln, spezifizieren, analysieren und verwalten*. dpunkt, Heidelberg.

Eckert et al. 2011

Eckert, C., Alink, T., Ruckpaul, A. und Albers, A. (2011) *Different notions of function: results from an experiment on the analysis of an existing product*. In *Journal of Engineering Design*. Taylor & Francis, London, UK.

Eckert, Clarkson & Zanker 2004

Eckert, C., Clarkson, P. J. und Zanker, W. (2004) *Change and customisation in complex engineering domains*. In *Research in Engineering Design*, Vol. 15-1, S. 1-21.

Edwards & Howell 1991

Edwards, M. und Howell, S. (1991) *A Methodology for System Requirements Specification and Traceability for Large Real-Time Complex Systems*. Technical report, U.S. Naval Surface Warfare Center, Dahlgren Division, Dahlgren.

Eger, Eckert & Clarkson 2005

Eger, T., Eckert, C. und Clarkson, P. J. (2005) *The Role of Design Freeze in Product Development*. In *Proceedings of the 15th International Conference on Engineering Design (ICED'05)*, Melbourne, 15.-18. August 2005.

Ehrlenspiel 2009

Ehrlenspiel, K. (2009) *Integrierte Produktentwicklung – Denkabläufe, Methodeneinsatz, Zusammenarbeit*. 4. Auflage, Hanser, München..

Eiletz 1999

Eiletz, R. (1999) *Zielkonfliktmanagement bei der Entwicklung komplexer Produkte – am Beispiel PKW-Entwicklung*. Shaker, Aachen. Dissertation, TU München.

Eppinger & Salminen 2001

Eppinger, S. und Salminen, V. (2001) *Patterns of product development interactions*. In *Proceedings of the 13th International Conference on Engineering Design (ICED'01)*, Glasgow, 21.-23. August 2001.

Erden et al. 2008

Erden, M. S., Komoto, H., van Beek, J., D'Amelio, V., Echavarria, E und Tomiyama, T. (2008) *A review of function modeling: approaches and applications*. In *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 22 (2), S. 147-169.

Felgen 2007

Felgen, L. (2007) *Systemorientierte Qualitätssicherung für mechatronische Produkte*. Dr. Hut, München. Dissertation, TU München.

Flath et al. 2000

Flath, M., Kespohl, H., Möhringer, S. und Oberschelp, O. (2000) *Entwicklung mechatronischer Systeme*. In Gausemeier, J. et al. (Hrsg.) *Entwicklungsumgebungen Mechatronik – Methoden und Werkzeuge zur Entwicklung mechatronischer Systeme*. Paderborn, HNI-Verlagsschriftenreihe, Band 80.

Frank 2006

Frank, U. (2006) *Spezifikationstechnik zur Beschreibung der Prinziplösung selbstoptimierender Systeme*. Dissertation. Universität Paderborn.

Freudenmann 2014

Freudenmann, T. (2014) *Ontologien zur Validierung von Produkten basierend auf dem Contact & Channel – Ansatz (C&C²-Ansatz)*. Dissertation, IPEK, Karlsruher Institut für Technologie (KIT).

Gausemeier, Lanza & Lindemann 2012

Gausemeier, J., Lanza, G., Lindemann, U. (2012) *Produkte und Produktionssysteme integrativ konzipieren – Modellbildung und Analyse in der frühen Phase der Produktentstehung*. Hanser, München.

Gausemeier et al. 2000

Gausemeier, J., Lindemann, U., Reinhart, G. und Wiendahl, H.-P. (2000) *Kooperatives Produktengineering - Ein neues Selbstverständnis des ingenieurmäßigen Wirkens*. HNI-Verlagsschriftenreihe, Bd. 79, Heinz Nixdorf Institut, Paderborn.

Gausemeier, Plass & Wenzelmann 2009

Gausemeier, J., Plass, C. und Wenzelmann, C. (2009) *Zukunftsorientierte Unternehmensgestaltung – Strategien, Geschäftsprozesse und IT-Systeme für die Produktion von morgen*. Hanser, München.

Gausemeier, Tschirner & Dumitrescu 2013

Gausemeier, J., Tschirner, C., und Dumitrescu, R. (2013). *Der Weg zu Intelligenten Technischen Systemen*. *Industrie Management*, 29(1).

Gebauer 2001

Gebauer, M. (2001) *Kooperative Produktentwicklung auf Basis verteilter Anforderungen*. Dissertation, Universität Karlsruhe (TH), Shaker Verlag, Aachen.

Gerst 2002

Gerst, M. (2002) *Strategische Produktentscheidungen in der integrierten Produktentwicklung*. Dissertation, TU München. Dr. Hut, München.

Gilz & Eigner 2013

Gilz, T und Eigner, M. (2013) *Ansatz zur integrierten Verwendung von SysML Modellen in PLM zur Beschreibung der funktionalen Produktarchitektur* In Maurer, M., Schulze, S.-O. (Hrsg.) *Tag des Systems Engineerings 2013*. Carl Hanser, München, S. 293-302.

Glinz 2007

Glinz, M. (2007) *On Non-Functional Requirements*. In 15th IEEE International Requirements Engineering Conference, 15-19 Oktober, Neu Delhi, Indien.

Göpfert 1998

Göpfert, J. (1998) *Modulare Produktentwicklung: Zur gemeinsamen Gestaltung von Technik und Organisation*. Deutscher Universitäts-Verlag, Wiesbaden. Dissertation, TU München.

Grady 1992

Grady, R. (1992) *Practical Software Metrics for Project Management and Process Improvement*. Prentice Hall.

Greniewski & Kempisty 1963

Greniewski, H. und Kempisty, M. (1963) *Kybernetische Systemtheorie ohne Mathematik*. Dietz, Berlin.

Gronau 2009

Gronau, N. (2009) *Anwendungen und Systeme für das Wissensmanagement: ein aktueller Überblick*. 3. Auflage. GITO, Berlin.

Gruber 1993

Gruber, T. R (1993) *A Translation Approach to Portable Ontology Specifications*. In *Knowledge Acquisition*, Vol. 5, S. 199-220.

Haberfellner et al. 2012

Haberfellner, R., de Weck, O., Fricke, E. und Vössner, S. (2012) *Systems Engineering – Grundlagen und Anwendung*. 12. Auflage. Orell Füssli, Zürich.

Hall & Fagen 1956

Hall, A. D und Fagen, R. E. (1956) *Definition of systems. General Systems 1*.

Hastings & McManus 2004

Hastings, D. und McManus, H. (2004) *A Framework for Understanding Uncertainty and its Mitigation and Exploitation in Complex Systems*. In Engineering Systems Symposium 2004, Massachusetts, USA.

Hellenbrand 2013

Hellenbrand, D. (2013) *Transdisziplinäre Planung und Synchronisation mechatronischer Produktentwicklungsprozesse*. Dr. Hut, München. Dissertation, TU München.

Heylighen 1999

Heylighen F. (1999) *The Growth of Structural and Functional Complexity during Evolution*. In Heylighen, F, Bollen, J und Riegler, A. (Hrsg.) *The Evolution of Complexity*, Kluwer Academic, Dordrecht, S. 17-44.

Hood et al. 2007

Hood, C., Mühlbauer, S., Rupp, C. und Versteegen, G. (2007) *iX-Studie Anforderungsmanagement*. Heise, Hannover.

Hubbard 2007

Hubbard, D. W. (2007) *How to Measure Anything*. John Wiley & Sons, Inc., Hoboken.

IEEE 610.12 1990

The Institute of Electrical and Electronics Engineers (1990) *IEEE Standard Glossary of Software Engineering Terminology*. New York.

Iwanek et al. (2013)

Iwanek, P., Kaiser, L., Dumitrescu, R. und Nyßen, A. (2013) *Fachdisziplinübergreifende Systemmodellierung mechatronischer Systeme mit SysML und CONSENS*. In Maurer, M., Schulze, S.-O. (Hrsg.) *Tag des Systems Engineerings 2013*. Carl Hanser, München, S. 337-346.

Jansen 2007

Jansen, S. (2007) *Eine Methodik zur modellbasierten Partitionierung mechatronischer Systeme*. Aachen, Shaker. Dissertation Ruhr-Universität Bochum.

Jupp, Eckert & Clarkson 2009

Jupp J. R., Eckert C. und Clarkson P. J. (2009) *Dimensions of Decision Situations in Complex Product Development*. In International Conference on Engineering Design, ICED'09, Stanford, Vol. 3, S. 239-250.

Kallmeyer 1998

Kallmeyer, F. (1998) *Eine Methode zur Modellierung prinzipieller Lösungen mechatronischer Systeme*. Dissertation, Universität Paderborn.

Keuth 1978

Keuth, H. (1978) *Realität und Wahrheit. Zur Kritik des kritischen Rationalismus*. Tübingen.

Klaus 1962

Klaus, G. (1967) *Kybernetik in philosophischer Sicht*. 2. Auflage, Berlin.

Klaus 1967

Klaus, G. (1967) *Wörterbuch der Kybernetik*. Berlin.

Klein 1998

Klein, R. (1998) *A Knowledge Level Theory of Design and Engineering*. In *Globalization of manufacturing in the digital communications era of the 21st century: innovation, agility, and the virtual enterprise*. Dordrecht; Kluwer Academic Publishers, S. 271-286.

Klir 1972

Klir, J. G. (1972) *Trends in general systems theory*. John Wiley & Sons Limited, Canada.

Kluge 2002

Kluge, F. (2002) *Etymologisches Wörterbuch der deutschen Sprache*. De Gruyter, Berlin.

Kosiol 1961

Kosiol, E. (1961) *Modellanalyse als Grundlage unternehmerischer Entscheidungen*. In Zeitschrift für handelswissenschaftliche Forschung 13, S. 318-334.

Kreimeyer 2010

Kreimeyer, M. F. (2010) *A Structural Measurement System for Engineering Design Processes*. Dr. Hut, München. Dissertation, TU München.

Kreye, Goh & Newnes 2011

Kreye, M. E., Goh, Y. M. und Newnes, L. B. (2011) *Manifestation of Uncertainty – A Classification*. In 18th International Conference on Engineering Design ICED'11, Copenhagen, Denmark.

Kruse 1996

Kruse, P. J. (1996) *Anforderungen in der Systementwicklung – Erfassung, Aufbereitung und Bereitstellung von Anforderungen in interdisziplinären Entwicklungsprojekten*. Dissertation, TU Clausthal, VDI-Verlag, Düsseldorf.

Langan-Fox, Anglim & Wilson 2004

Langan-Fox, J., Anglim, J. und Wilson, J. R. (2004) *Mental Models, Team Mental Models, and Performance: Process, Development and Future Directions*. In Human Factors and Ergonomics in Manufacturing, Vol. 14, No. 4, S. 331-352.

Lange 1965

Lange, O. (1965) *Wholes and parts: A general theory of system behaviour*. Pergamon, Oxford/London/Warschau.

La Porte 1975

La Porte, T. R. (1975) *Explication of the Concept of Organized Complexity: Studies in its Effects*. In La Porte, T. R. (Hrsg.). *Organized Social Complexity – Challenge to Politics and Policy*. Princeton, University Press, S. 3-39.

Lévárdy & Browning 2009

Lévárdy, V. und Browning, T. R. (2009) *An Adaptive Process Model to Support Product Development Project Management*. IEEE Transactions on Engineering Management 56, S. 600-620.

Leuf & Cunningham 2001

Leuf, B. und Cunningham, L. (2001) *The Wiki Way - Collaboration and Sharing on the Internet*. Addison-Wesley, Boston.

Lindemann 2009

Lindemann, U. (2009) *Methodische Entwicklung technischer Produkte*. 3. Auflage, Springer, Berlin, Heidelberg.

Lindemann & Lorenz 2008

Lindemann, U. und Lorenz, M. (2008) *Uncertainty Handling in Integrated Product Development*. In 10th International Design Conference DESIGN 2008, Dubrovnik, Croatia, S. 175-182.

Lindemann & Maurer 2007

Lindemann, U. und Maurer, M. (2007) *Facing Multi-Domain Complexity in Product Development*. In *The Future of Product Development – Proceedings of the 17th CIRP Design Conference*, Berlin, Springer.

Lindemann, Maurer & Braun 2009

Lindemann, U., Maurer, M. und Braun, T. (2009) *Structural complexity management: An approach for the field of product design*. Springer, Berlin.

Lohmeyer 2012

Lohmeyer, Q. (2012) *Menschzentrierte Modellierung von Produktentstehungssystemen unter besonderer Berücksichtigung der Synthese und Analyse dynamischer Zielsysteme*. Dissertation, IPEK, Karlsruher Institut für Technologie (KIT).

Luhmann 1979

Luhmann, N. (1979) *Trust and Power*. Wiley, Chichester.

Luhmann 1994

Luhmann, N. (1994) *Die Wissenschaft der Gesellschaft. 2. Auflage*. Suhrkamp, Frankfurt a.M.

Malik 2003

Malik, F. (2003) *Strategie des Managements komplexer Systeme*. Haupt, Bern.

Maurer 2007

Maurer, M. S. (2007) *Structural awareness in complex product design*. Dr. Hut, München. Dissertation, TU München.

McConell 1996

McConnel, S. (1996) *Rapid Development: Taming Wild Software Schedules*. Microsoft Press, Redmond.

Meboldt 2008

Meboldt, M. (2008) *Mentale und formale Modellbildung in der Produktentstehung – als Beitrag zum integrierten Produktentstehungs-Modell (iPeM)*. Dissertation, IPEK, Universität Karlsruhe (TH).

Meboldt, Matthiesen & Lohmeyer 2012

Meboldt, M., Matthiesen, S. und Lohmeyer, Q. (2012) *The Dilemma of Managing Iterations in Time-to-Market Development Processes*. In 2nd International Conference on Modelling and Management of Engineering Processes MMEP 2012, Cambridge, UK.

Mesarovic & Macko 1969

Mesarovic, M. D. und Macko, D. (1969) *Foundations for a scientific theory of hierarchical systems*. In White/Wilson/Wilson (Hrsg.) *Hierarchical structures*, New York.

Miller 1956

Miller, G. A. *Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information*. *The Psychological Review*, Vol. 63, S. 81-97.

Miller, Galanter & Pribram 1960

Miller, G. A., Galanter, E. und Pribram, K. A. (1960) *Plans and the Structure of Behavior*. Holt, Rhinehart, & Winston, New York.

Müller 1983

Müller, R. (1983) *Zur Geschichte des Modelldenkens und des Modellbegriffs*. In Stachowiak, H. (Hrsg.) *Modelle – Konstruktion der Wirklichkeit*. München, S. 17-86.

Müller 2007

Müller, A. (2007) *Iterative Zielklärung und Handlungsplanung als Faktoren erfolgreichen Gruppenhandels bei der Lösung komplexer Probleme*. Dissertation, TU München.

Muschik 2011

Muschik, S. (2011) *Development of Systems of Objectives in Early Product Engineering*. Dissertation, IPEK, Karlsruher Institut für Technologie (KIT).

Negele et al. 1999

Negele, H., Fricke, E., Schrepfer, L. und Härtlein, N. (1999) *Modelling of Integrated Product Development Processes*. In 9th Annual Symposium of INCOSE, UK.

de Neufville 2004

de Neufville, R. (2004) *Uncertainty management for engineering systems planning and design*. In 1st Engineering Systems Symposium, MIT, Cambridge, MA.

Neumann, Badke-Schaub & Lauche 2006

Neumann, A., Badke-Schaub, P. und Lauche, K. (2006) *A Framework for Measuring Team Mental Models in Design*. In 9th International Design Conference DESIGN 2006, Dubrovnik, Croatia.

Oerding 2009

Oerding, J. (2009) *Ein Beitrag zum Modellverständnis der Produktentstehung - Strukturierung von Zielsystemen mittels C&CM*. Dissertation, IPEK, Universität Karlsruhe (TH).

Pahl et al. 2007

Pahl, G., Beitz, W., Feldhusen, J. und Grote, K.-H. (2007) *Pahl/Beitz Konstruktionslehre – Grundlagen erfolgreicher Produktentwicklung - Methoden und Anwendung*. 7. Auflage, Springer, Berlin, Heidelberg.

Patzak 1982

Patzak, G. (1982) *Systemtechnik, Planung komplexer innovativer Systeme. Grundlagen, Methoden, Techniken*. Springer, Berlin.

Pfeiffer 1965

Pfeiffer, W. (1965) *Absatzpolitik bei Investitionsgütern der Einzelfertigung – Möglichkeiten und Grenzen des Einsatzes absatzpolitischer Instrumente im Sondermaschinenbau*. Poeschel, Stuttgart.

Pfeifer-Silberbach 2005

Pfeifer-Silberbach, U. (2005) *Ein Beitrag zum Monitoring des Reifegrades in der Entwicklung eines Produktes*. Shaker, Aachen. Dissertation, TU Darmstadt.

Pohl 2007

Pohl, K. (2007) *Requirements Engineering – Grundlagen, Prinzipien, Techniken*. dpunkt, Heidelberg.

Ponn & Lindemann 2011

Ponn, J. und Lindemann, U. (2011) *Konzeptentwicklung und Gestaltung technischer Produkte - Systematisch von Anforderungen zu Konzepten und Gestaltungsformen*. 2. Auflage, Springer, Berlin, Heidelberg.

Puhl 1999

Puhl, H. (1999) *Komplexitätsmanagement: Ein Konzept zur ganzheitlichen Erfassung, Planung und Regelung der Komplexität in Unternehmensprozessen*. Dissertation, Universität Kaiserslautern.

Pulm 2004

Pulm, U. (2004) *Eine systemtheoretische Betrachtung der Produktentwicklung*. Dissertation, TU München.

Ramesh & Jarke 2001

Ramesh, B. und Jarke, M. (2001) *Toward Reference Models for Requirements Traceability*. In IEEE Transactions on Software Engineering. Vol. 27, no. 1, S. 58- 93.

Robertson & Robertson 2006

Robertson, S. und Robertson, J. (2006) *Mastering the Requirements Process*. Addison Wesley, New York.

Ropohl 1975

Ropohl, G. (1975) *Einleitung in die Systemtechnik*. In Systemtechnik – Grundlagen und Anwendung, Hanser, München, S. 1-77.

Ropohl 2009

Ropohl, G. (2009) *Allgemeine Technologie – Eine Systemtheorie der Technik*. Universitätsverlag Karlsruhe.

Rupp & Queins 2003

Rupp, C. und Queins, S. (2003) *Vom Use-Case zum Test-Case*. In OBJEKTSpektrum, Vol. 4.

Rupp 2009

Rupp, C. (2009) *Requirements-Engineering und -Management*. 5. Auflage. Hanser. München.

Saak 2007

Saak, M. (2007) *Entwicklung eines Konzeptes und eines Prototypen für ein rechnergestütztes Werkzeug zum effizienten Einsatz der Problemlösungsmethodik „SPALTEN“*. Dissertation, IPEK, Universität Karlsruhe (TH).

Sauter 2012

Sauter, C. (2012) *Ein Beitrag zur Integration von Wikis und Social Tagging in die Produktentstehung*. Dissertation, IPEK, Karlsruher Institut für Technologie (KIT).

Sander, Kellner & Albers 2013

Sander, C., Kellner, M., und Albers, A. (2013) *Design for Energy Efficiency: Structural Optimization Process for Mechanical Parts in Dynamic Systems*. In NAFEMS World Congress.

Schlitt 2004

Schlitt, M. (2004) *Grundlagen und Methoden für Interpretation und Konstruktion von Informationssystemmodellen*. Dissertation, Universität Bamberg.

Schuh & Schwenk 2001

Schuh, G. und Schwenk, U. (2001) *Produktkomplexität managen*. Hanser, München.

Schuh 2005

Schuh, G. (2005) *Produktkomplexität managen: Strategien, Methoden, Tools*. Hanser. München.

Schuh 2012

Schuh, G. (2012) *Innovationsmanagement - Handbuch Produktion und Management* 3. 2. Auflage. Springer. Berlin, Heidelberg.

Simon 1994

Simon, H. A. (1994) *Die Wissenschaften vom Künstlichen*. Springer, Wien.

Smith & Reinertsen 1992

Smith, P. G. und Reinertsen D. G. (1992) *Stortening the Product Development Cycle*. In *Research-Technology Management*, May-June, S. 44-49.

Sommerville 2004

Sommerville, I. (2004) *Software Engineering*. Pearson, Addison-Wesley. Bosten, Munich.

Stachowiak 1973

Stachowiak, H. (1973) *Allgemeine Modelltheorie*. Springer, Wien.

Stachowiak 1983

Stachowiak, H. (1983) *Erkenntnisstufen zum Systematischen Neopragmatismus und zur Allgemeinen Modelltheorie*. In Stachowiak, H. (Hrsg.) *Modelle – Konstruktion der Wirklichkeit*. München, S. 87-146.

Statista 2014

Statista (2014) *Marktanteile der führenden Betriebssystemversionen in Deutschland von Januar 2009 bis Juli 2014*. URL <http://de.statista.com/statistik/daten/studie/158102/umfrage/marktanteile-von-betriebssystemen-in-deutschland-seit-2009/>. Aktualisierungsdatum 31.08.2014.

Stechert 2010

Stechert, C. (2010) *Modellierung komplexer Anforderungen*. Dissertation, TU Braunschweig.

Stempfle 2007

Stempfle, H. G. (2007) *Systemtheorie im Brückenbau*. Dissertation, Eidgenössische Technische Hochschule Zürich.

Stockinger 2008

Stockinger, S. (2008) *Konzept zur Optimierung des PDM-Einsatzes in der Automobilindustrie*. Dissertation, RWTH Aachen.

Stoff 1969

Stoff, V. A. (1969) *Modellierung und Philosophie*. Akademie-Verlag, Berlin.

Suh 1999

Suh, N. P. (1999) *A Theory of Complexity, Periodicity and the Design Axioms*. In *Research in Engineering Design*, Vol. 11, S. 116-131.

Thau 2013

Thau, S. (2013) *Heuristiken zur Analyse und Synthese technischer Systeme mit dem C&C²-Ansatz auf Basis von Entwicklungsprojekten im industriellen Umfeld*. Dissertation, IPEK, Karlsruher Institut für Technologie (KIT).

Thomas 2005

Thomas, O. (2005) *Das Modellverständnis in der Wirtschaftsinformatik: Historie, Literaturanalyse und Begriffsexplikation*. Veröffentlichungen des Instituts für Wirtschaftsinformatik, Hrsg. Prof. Dr. Dr. h.c. mult. August-Wilhelm Scheer, Heft 184.

Thunnissen 2005

Thunnissen, D. P. (2005) *Propagating and Mitigating Uncertainty in the Design of Complex Multidisciplinary Systems*. Dissertation, California Institute of Technology.

Ulrich & Eppinger 2000

Ulrich, K. T. und Eppinger, S. D. (2000) *Product Design and Development*. 2nd edition, McGraw Hill, Boston.

Unger 2003

Unger, D. W. (2003) *Product Development Process Design: Improving Development Response to Market, Technical and Regulatory Risks*. Dissertation, MIT.

Vajna 2005

Vajna, S. (2005) Workflow for design. In Clarkson, J. und Eckert, C. (Hrsg.) *Design Process Improvement - A Review of Current Practice*, Springer, London, S. 366-385.

VDI 2221 1993

Verein Deutscher Ingenieure (1993) *Richtlinie VDI 2221, Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte*. Beuth Verlag.

VDI 2206 2004

Verein Deutscher Ingenieure (2004) *Richtlinie VDI 2206, Entwicklungsmethodik für mechatronische Systeme*. Beuth Verlag.

VDI 2519 2001

Verein Deutscher Ingenieure (2001) *Richtlinie Vorgehensweise bei der Erstellung von Lasten-/Pflichtenheften*. VDI 2519, Beuth Verlag.

Völkel 2006

Völkel, M. (2006) *Personal Knowledge Models for More Productive Knowledge Workers*. In Wissensmanagement, S. 299-308.

Weber 2012

Weber C. (2012) *Produkte und Produktentwicklungsprozesse abbilden mit Hilfe von Merkmalen und Eigenschaften – eine kritische Zwischenbilanz*. In 23. DfX-Symposium in Bamberg, S. 25-62.

de Weck, Eckert & Clarkson 2007

de Weck, O., Eckert, C. und Clarkson J. (2007) *A Classification of Uncertainty for Early Product and System Design*. In 16th International Conference on Engineering Design ICED'07, Paris, France, No. 480.

Wegehaupt 2004

Wegehaupt, P. (2004) *Führung von Produktionsnetzwerken*. Dissertation, RWTH Aachen.

Weinzierl 2006

Weinzierl, J. (2006) *Produktreifegrad-Management in unternehmensübergreifenden Entwicklungnetzwerken - Ein ganzheitlicher Ansatz zur Entscheidungsunterstützung im strategischen Anlaufmanagement*. Verlag Praxiswissen, Dortmund. Dissertation, Universität Dortmund.

Wiebel et al. 2013

Wiebel, M., Eifler, T., Mathias, J., Kloberdanz, M., Bohn, A. und Birkhofer, H. (2013) *Modellierung von Unsicherheit in der Produktentwicklung*. In Jeschke, H., Jakobs, E.-M., Dröge, A. (Hrsg.) *Exploring Uncertainty – Ungewissheit und Unsicherheit im interdisziplinären Diskurs*. Springer Gabler, Wiesbaden.

Weiß 2006

Weiß, S. (2006) *Konzept und Umsetzung eines Navigators für Wissen in der Produktentwicklung*. Fortschritt-Berichte VDI.

Wiener 1949

Wiener, N. (1949) *Cybernetics – or Control and Communication in the Animal and the Machine*. MIT, John Wiley & Sons, Inc., New York.

Wintgen 1968

Wintgen, G. (1968) *Zur mengentheoretischen Definition und Klassifizierung kybernetischer Systeme*. In: *Wissenschaftliche Zeitschrift der Humboldt-Universität zu Berlin, Gesellschafts- und sprachwissenschaftliche Reihe* 17, S. 867-885

Wöhe & Döring 2013

Wöhe, G. und Döring, U (2013) *Einführung in die Allgemeine Betriebswirtschaftslehre*. Vahlen, München.

Wynn 2007

Wynn, D. C. (2007) *Model-Based Approaches to Support Process Improvement in Complex Product Development*. Dissertation, University of Cambridge (UK).

Wynn, Eckert & Clarkson 2007

Wynn, D. C., Eckert, C. M. und Clarkson, P. J. (2007) *Modelling Iteration in Engineering Design*. In 16th International Conference on Engineering Design ICED'07, Paris, France, No. 561.

Zingel 2013

Zingel, J. C. (2013) *Basisdefinition einer gemeinsamen Sprache der Produktentwicklung und einer Modellierungstechnik für Zielsystem und Objektsystem technischer Systeme in SysML auf Grundlage des ZHO-Prinzips*. Dissertation, Karlsruher Institut für Technologie (KIT).

Zschocke 1995

Zschocke, D. (1995) *Modellbildung in der Ökonomie*. München.

10 Anhang – Anleitung des Werkzeugs SystemModeler

Einführung

Vielen Dank für die Nutzung des SystemModeler. Dieses Handbuch soll Ihnen als kurze Einführung in die Konzepte und Funktionalitäten des SystemModeler dienen.

Der SystemModeler wurde im Rahmen einer Promotion am IPEK – Institut für Produktentwicklung des Karlsruher Instituts für Technologie (KIT) entwickelt. Die Entwicklung und das Programm als Ergebnis dieser Entwicklung sind auf eine prototypische Umsetzung und Validierung einer Methode zur Modellierung von Zielsystemen in der interdisziplinären Produktentstehung ausgerichtet. Entsprechend wurde eine kommerzielle Verwertung des Programms zu keinem Zeitpunkt als Zielsetzung formuliert, weswegen zum einen keinerlei Support gewährt werden kann und zum anderen keinerlei Haftung für die Nutzung des Programms übernommen wird. Die Nutzung des Programms erfolgt ausdrücklich auf eigenes Risiko des Anwenders.

Der SystemModeler ist ein Werkzeug zur Modellierung von Zielsystemen in der Produktentstehung. Fokus des Werkzeugs liegt zum einen auf der ganzheitlichen und durchgängigen Abbildung aller zum Zwecke der Zielsystemmodellierung relevanter Informationen sowie deren Verknüpfungen. Zum anderen wurde auf eine einfache, intuitive und flexible Anwendbarkeit des Werkzeugs durch den Benutzer Wert gelegt. Hierbei ist zu berücksichtigen, dass im Rahmen dieses Prototyps lediglich textuelle Informationen modelliert werden können.

Das Programmfenster

Das Programmfenster des SystemModeler öffnet sich nach Starten des Programms durch Ausführung der Datei SystemModeler.exe. Das Fenster lässt sich in einen Bereich unterteilen, der systemspezifische Informationen (Struktur- oder Elementansicht und Relationen) anzeigt und einen Bereich der systemunspezifische Informationen darstellt (restliche Bereiche in Abbildung 1).

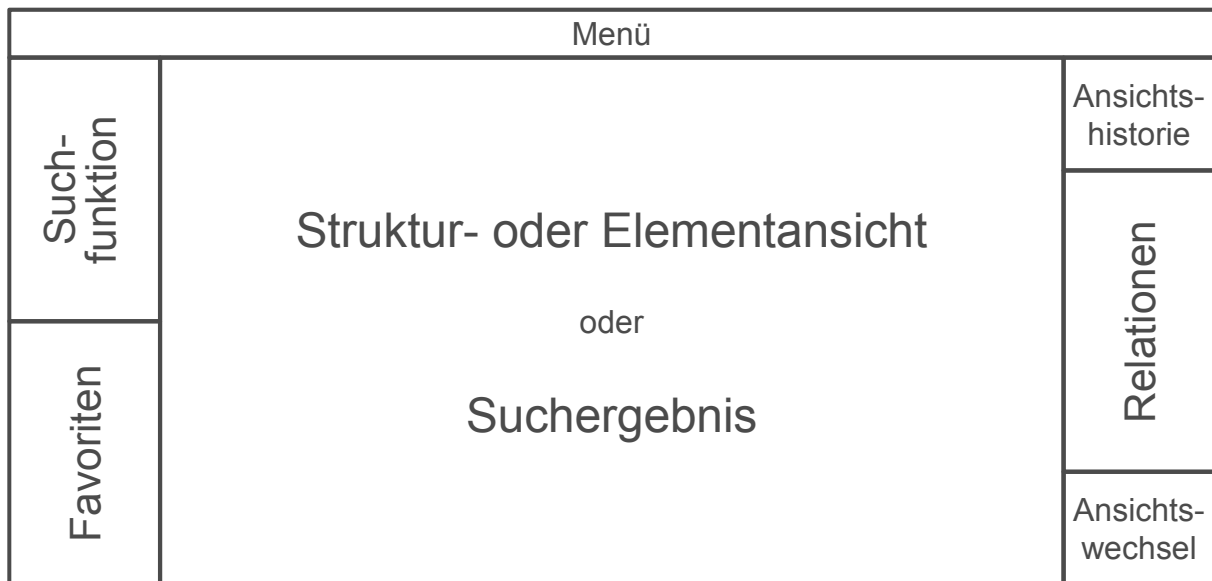


Abbildung 1 - Schematische Darstellung des Programmfensters

Im Menü-Bereich sind sämtliche übergreifenden Befehle und Einstellungen zusammengestellt. File enthält alle dateispezifischen Befehle (Create, Open, Save und Close Project). View enthält die beiden Einträge System View und Structural View und ermöglicht die Auswahl einer der beiden Ansichten auf das modellierte Gesamtsystem (vgl. Abbildung 2 und Abbildung 3). Settings deckt sämtliche projektspezifischen Einstellungen ab und beinhaltet die Definition der Elementtypen (System-Types), Relationen (System-Relations) sowie der Attribute System-Status (System-Status) und System-Wichtigkeit (System-Importances). Zusätzlich enthält es den Eintrag System-Ansicht (System-View), der eine Anpassung der Darstellung im Programmfenster ermöglicht. Help enthält den Eintrag View Documentation und öffnet die vorliegende Dokumentation des SystemModeler, sofern diese im gleichen Verzeichnis wie SystemModeler.exe abgelegt ist.

Der Programmfenster-Bereich Struktur- oder Elementansicht stellt das bzw. die Systemelemente dar (vgl. Abbildung 2 und Abbildung 3) und dient ebenfalls zur Anzeige von Suchergebnissen. Im Bereich Suchfunktion können einzelne Suchanfragen mittels der Eingabe von Freitext und der Wahl bestimmter, weiterer Attribute definiert werden. Der Bereich Favoriten enthält benutzerdefinierte und benutzerspezifische Element-Favoriten, die stets unmittelbar verfügbar sind. Im Programmfensterbereich Relationen werden sämtliche Relationen und Assoziierte Kopien eines Systemelements angezeigt und sind damit ebenfalls unmittelbar verfügbar. Der Programmfenster-Bereich Ansichtshistorie ermöglicht die Navigation durch die Historie der angezeigten Systemelemente. Der Bereich Ansichtswchsel erlaubt den Wechsel zwischen Struktur- und Elementansicht.

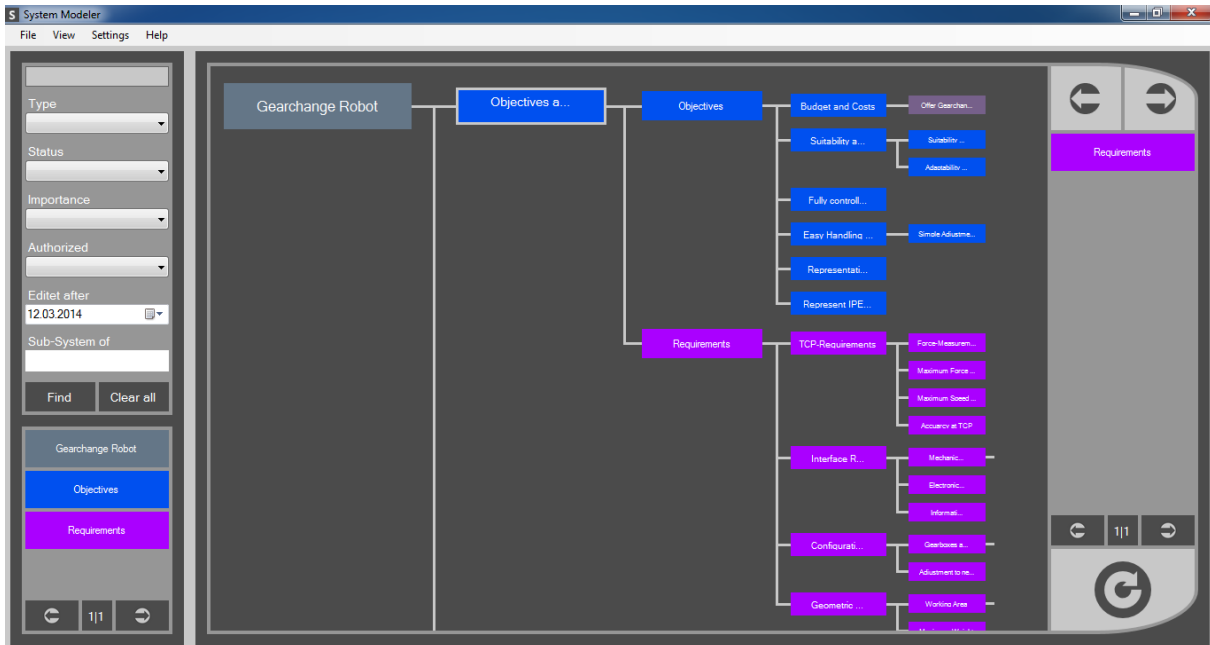


Abbildung 2: Darstellung des Programmfensters in der Strukturansicht (Structural View)

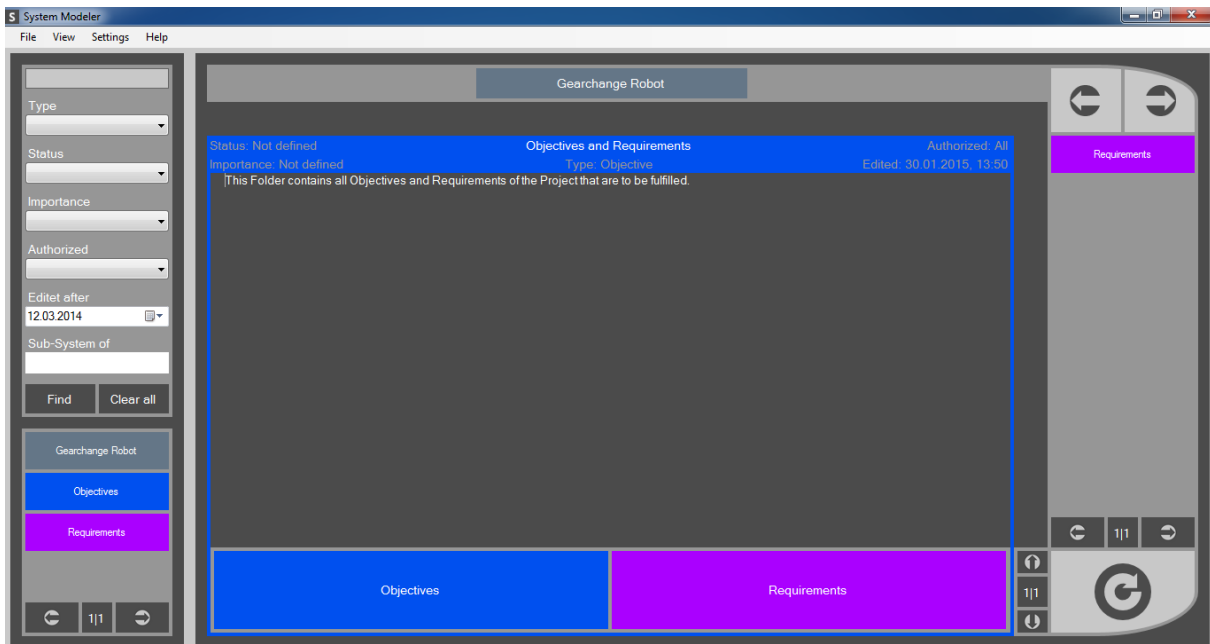


Abbildung 3: Darstellung des Programmfensters in der Elementansicht (System View)

Ein Projekt anlegen/öffnen und speichern

Ein neues Projekt kann je nach Bedarf auf zwei unterschiedliche Weisen angelegt werden. Soll ein neues Projekt ohne projektspezifische Einstellungen erstellt werden, öffnen Sie das Programm (schließen Sie es ggf. zuvor) und wählen im Menü-Bereich File→Create Project. Im daraufhin folgenden Fenster geben Sie den Namen des Projekts bzw. des Gesamtsystems ein und bestätigen Sie mit OK (vgl. Abbildung 4).

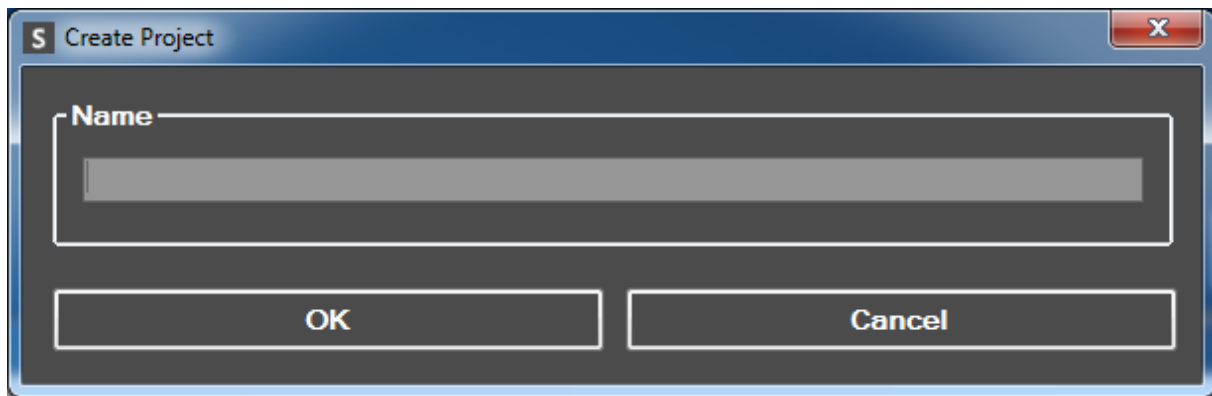


Abbildung 4: Anlegen eines neuen Projekts

Soll ein Projekt mit vordefinierten projektspezifischen Einstellungen erstellt werden, öffnen sie das Programm und öffnen Sie zunächst über File→Open Project das Projekt, dessen Einstellungen übernommen werden sollen. Öffnen Sie das Projekt Default.smp, um die Defaulteinstellungen zu übernehmen. Schließen Sie daraufhin das Projekt ohne Änderungen vorzunehmen über File→Close Project und wählen Sie nun File→Create Project um ein neues Projekt mit den gewählten projektspezifischen Einstellungen zu erstellen.

Ein bestehendes Projekt öffnen Sie über File→Open Project. Zur Auswahl einer SystemModeler-Projektdatei (Dateiendung smp) navigieren Sie durch die bestehende Windows-Ordnerstruktur. Ein bereits geöffnetes Project speichern Sie über File→Save Project. Soll ein geöffnetes Projekt geschlossen werden, ohne dass dabei das Programm beendet werden soll, wählen Sie File→Close Project.

Projektspezifische Einstellungen vornehmen

Jedes Projekt umfasst eine Menge projektspezifischer Einstellungen, die individuell angepasst werden können. Projektspezifische Einstellungen können nur bei geöffnetem Projekt vorgenommen werden.

Elementtypen festlegen/ändern

Jedes Systemelement muss im Rahmen der Modellierung mit SystemModeler von genau einem Elementtyp aus einer geschlossenen Liste sein (beispielsweise Anforderung oder Stakeholder). Die Einträge dieser Liste und damit die in einem Projekt zur Verfügung stehenden Elementtypen definieren Sie über Settings→System-Types (vgl. Abbildung 5).

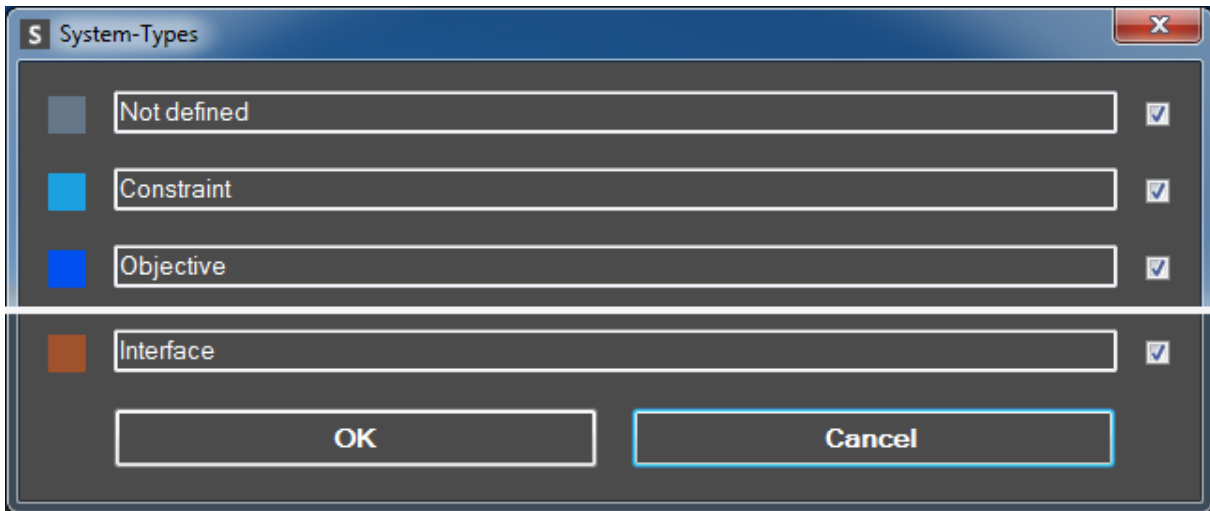


Abbildung 5: Festlegen von Elementtypen

Hierbei kann der Name und die Farbe jedes Elementtyps festgelegt werden sowie einzelne Elementtypen aktiviert (deaktiviert) werden, so dass diese Einträge (nicht) in der Liste der auswählbaren Elementtypen erscheint. Zur Anpassung eines Elementtyps muss der Elementtyp aktiviert, d.h. die Checkbox ausgewählt sein. Anschließend können Sie den Namen sowie die Farbe des Elementtyps durch Auswahl des farbigen Kastens festlegen. Insgesamt können maximal 15 Elementtypen definiert werden.

Relationen festlegen/ändern

Einzelne Systemelemente können im Rahmen der Modellierung mit SystemModeler in Beziehung zueinander gesetzt werden. Hierzu kann aus einer geschlossenen Liste von Relationen gewählt werden, die über Settings→System-Relations definiert werden kann (vgl. Abbildung 6).

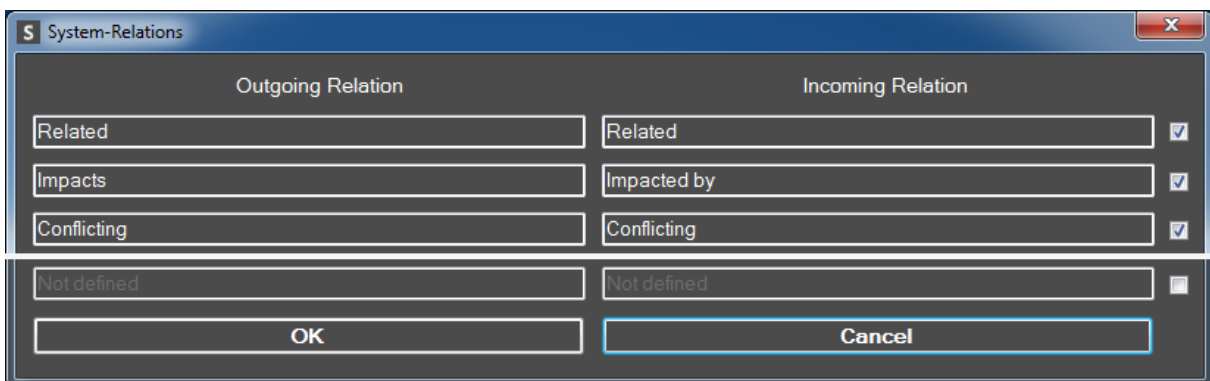


Abbildung 6: Festlegen von Relationen

Um bidirektionale Relationen abbilden zu können, wird eine Relation jeweils definiert über eine ausgehende und eine eingehende Beziehung (vgl. Impacts und Impacted by in Abbildung 6). Zur Anpassung einzelner Relationen müssen daher jeweils die Outgoing Relation sowie die Incoming Relation individuell festgelegt bzw. geändert

werden. Soll eine unidirektionale Relation definiert werden, sind die Felder Outgoing Relation sowie Incoming Relation gleich zu benennen (vgl. Related in Abbildung 6). Zur Anpassung und Nutzung einzelner Relationen müssen diese aktiviert, d.h. die Checkbox ausgewählt sein. Insgesamt können maximal 15 Relationen definiert werden.

System-Status festlegen/ändern

Einem Systemelement kann ein Status zugewiesen werden (vgl. Freigabe- und Änderungsmanagement). Die Einträge der Liste zur Verfügung stehender Status können über Settings→System-Status definiert werden (vgl. Abbildung 7).

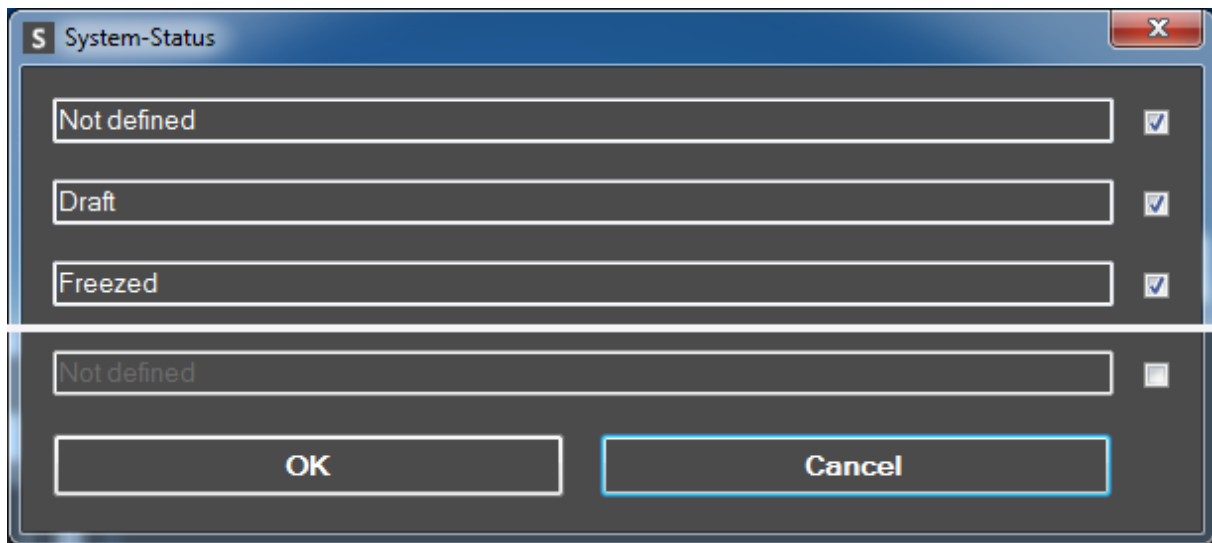


Abbildung 7: Festlegen von System-Status

Die einzelnen Namen der (aufeinander folgenden) Status können definiert sowie einzelne Status mittels Checkbox aktiviert (deaktiviert) werden, so dass diese Einträge (nicht) in der Liste der auswählbaren System-Status erscheinen. Zur Anpassung eines System-Status muss der Status aktiviert sein. Es können insgesamt maximal 15 Status definiert werden.

System-Wichtigkeit festlegen/ändern

Einzelnen Systemelementen kann eine Wichtigkeit zugeordnet werden. Hierzu kann aus einer geschlossenen Liste von Wichtigkeiten gewählt werden, die über Settings→System Importance definiert werden kann. Die einzelnen Namen der (aufeinander folgenden) Wichtigkeiten können definiert sowie einzelne Wichtigkeiten mittels Checkbox aktiviert (deaktiviert) werden, so dass diese Einträge (nicht) in der Liste der auswählbaren System-Wichtigkeiten erscheinen. Zur Anpassung einer System-Wichtigkeit muss die Checkbox aktiviert sein. Es können insgesamt maximal 15 Wichtigkeiten definiert werden.

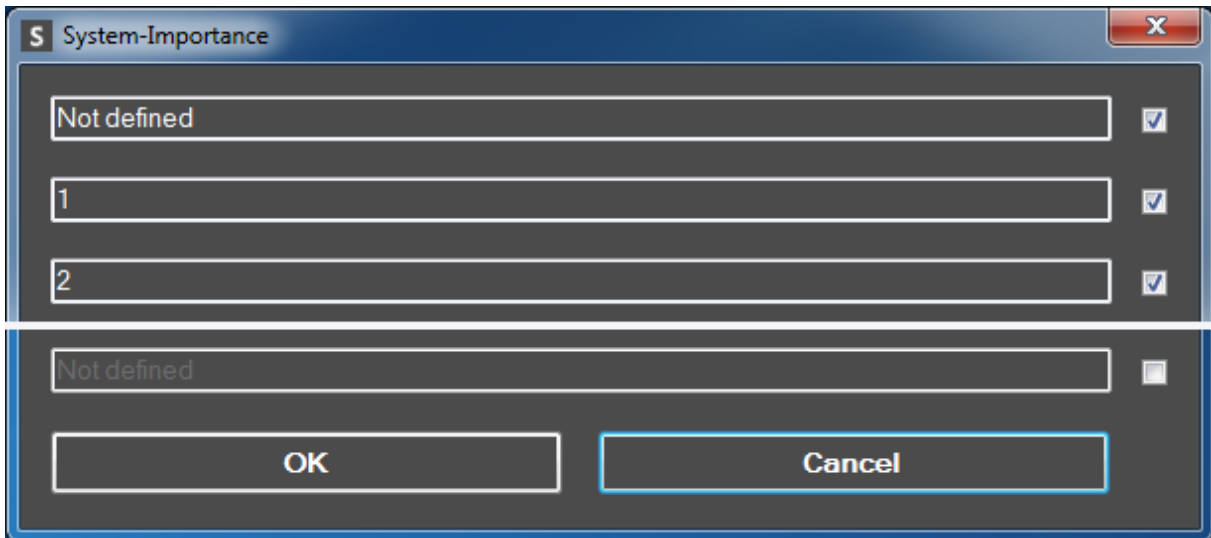


Abbildung 8: Festlegen von System-Wichtigkeit

Ansicht nutzen und wechseln

Der SystemModeler bietet über die Struktur- und Elementansicht zwei unterschiedliche Ansichten zur Darstellung modellierter Systeme (vgl. Abbildung 9 und Abbildung 10).

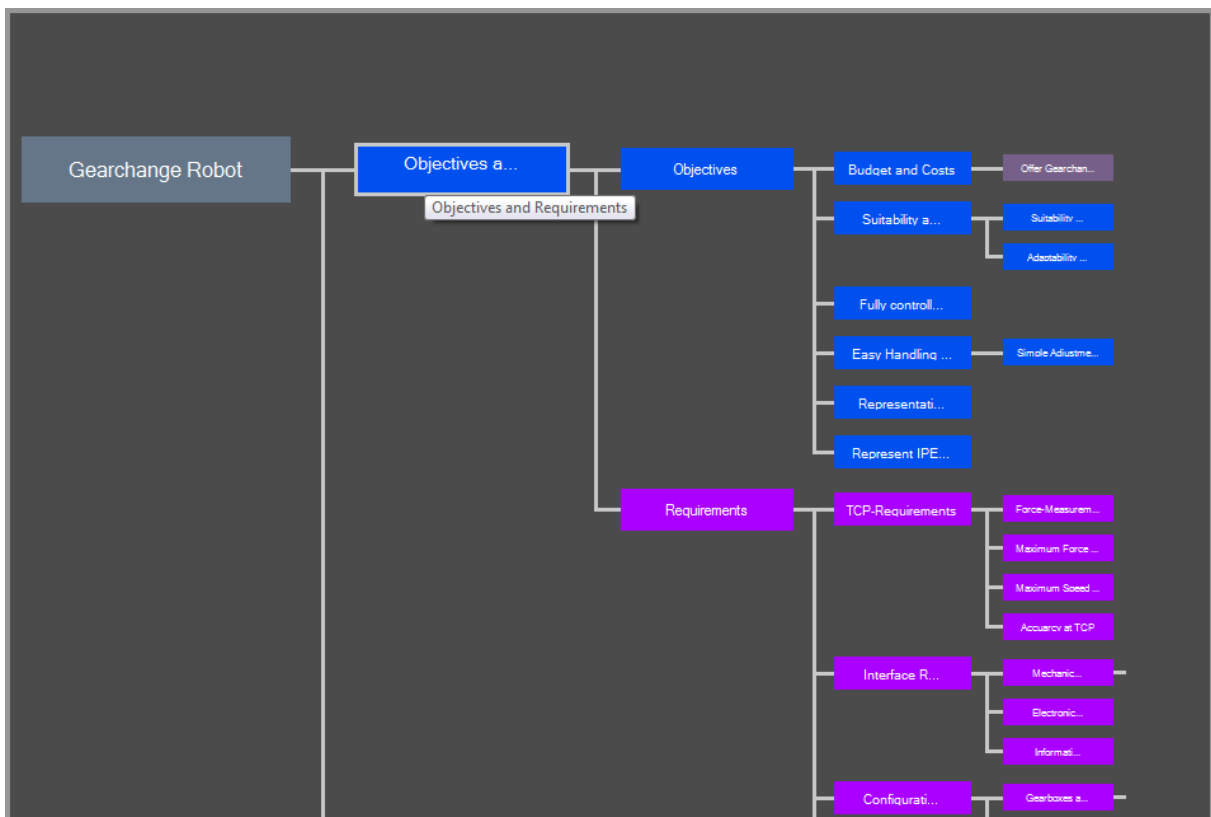


Abbildung 9: Strukturansicht des SystemModelers

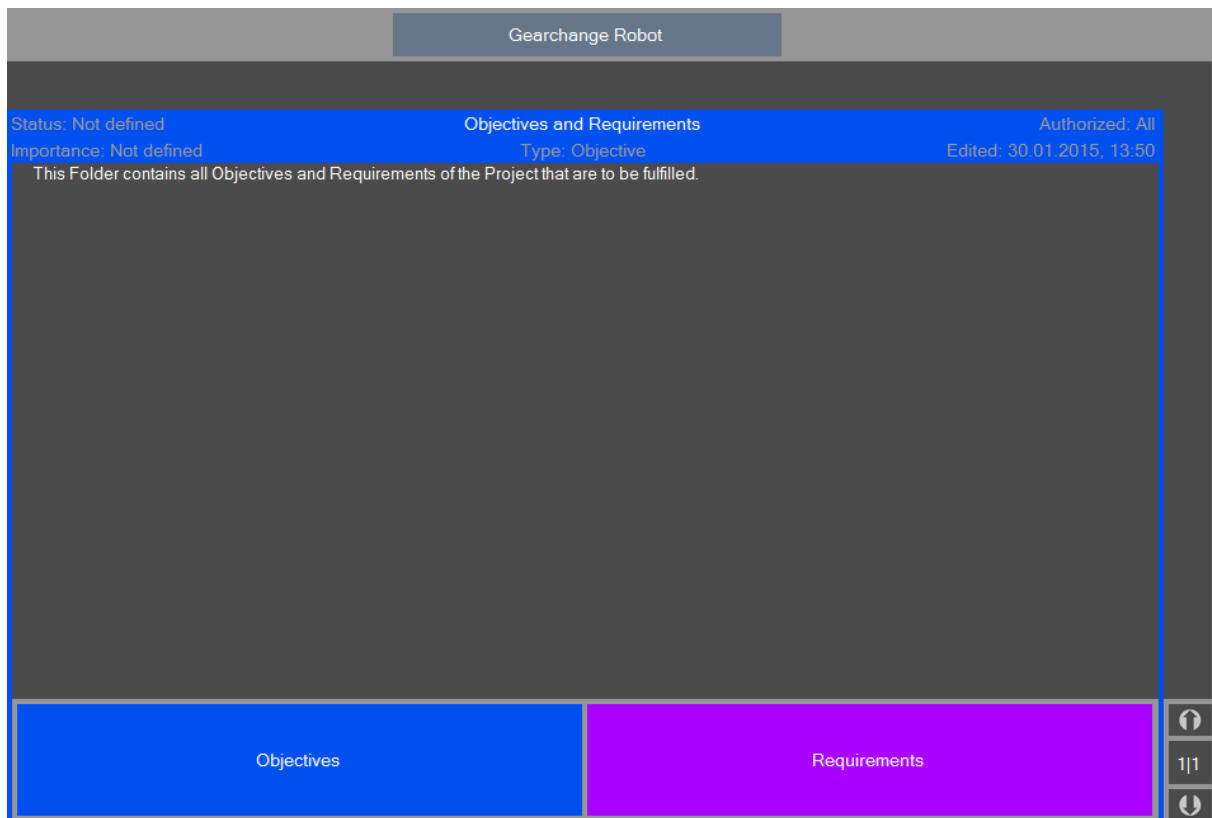


Abbildung 10: Elementansicht des SystemModelers

Systemelement selektieren

In beiden Ansichten steht das selektierte Systemelement im Fokus der Darstellung (in Abbildung 9 und Abbildung 10 das Systemelement Objectives and Requirements). Dies bedeutet, dass das selektierte Systemelement sowie die unmittelbare (Elementansicht) bzw. mittelbare (Strukturansicht) Umgebung des Elements dargestellt wird. Ein Systemelement wird selektiert, indem ein einfacher (Maus-)Linksklick auf das zu selektierende Systemelement ausgeführt wird. Die aktuell gewählte Ansicht (Struktur- oder Elementansicht) baut sich daraufhin mit dem (neu) selektierten Systemelement auf. Hierbei ist die Selektion eines Systemelements nicht nur auf Systemelemente der Struktur- und Elementansicht beschränkt – ebenso können Favoriteneinträge und in Beziehung stehende Systemelemente selektiert werden (siehe auch Abschnitt Favoriten erstellen und nutzen sowie Relationen eines Systems bearbeiten).

Strukturansicht nutzen

Die Strukturansicht des SystemModelers ist vergleichbar der Darstellungsform einer MindMap, d.h. die hierarchisch strukturierten Systemelemente werden in einer Baumansicht dargestellt (vgl. Abbildung 9).

Hierbei werden nicht alle vorhandenen Systemelemente dargestellt, sondern nur solche, die in einer mittelbaren Beziehung zum selektierten Systemelement stehen.

Konkret werden Systemelemente angezeigt, die ein Sub-, Subsub oder Subsubsubelement des selektierten Elements sind, das Superlement des selektierten Elements, ein Sub-, Subsub oder Subsubsubelement des Superlements sowie das Supersuperelement des selektierten Elements. Sind weitere Systemelemente vorhanden, die jedoch aufgrund dieser Regeln in Verbindung mit dem selektierten Element nicht angezeigt werden, wird stattdessen eine halbe Verbindungslinie dargestellt.

Der jeweils dargestellt Ausschnitt von Systemelementen in der Strukturansicht kann auf verschiedene Weisen angepasst werden. Durch einfachen (Maus-)Linksklick auf eine freie, d.h. nicht durch ein Systemelement belegte Stelle der Strukturansicht, wird das Zentrum der Ansicht auf den gewählten Punkt festgelegt und die Ansicht neu aufgebaut. Weiterhin lässt sich der Ausschnitt der Strukturansicht mittels der (Tastatur-)Pfeiltasten verschieben. Darüber hinaus kann der Ausschnitt der Strukturansicht in vertikaler Richtung mittels Mausrad verschoben werden.

Elementansicht nutzen

Die Elementansicht des SystemModelers zeigt das selektierte Systemelement im Detail (Attribute und textuelle Beschreibung des Systemelements) sowie dessen unmittelbare Super- und Subsystemelemente (vgl. Abbildung 10).

Im obersten Teil der Ansicht wird das jeweilige Supersystemelement dargestellt. Im oberen Teil der Ansicht (unterhalb des Supersystemelements) wird das selektierte Systemelement und dessen Attribute Elementtyp, Status, Wichtigkeit sowie Autorisierung und Änderungszeitpunkt dargestellt.

Im mittleren Teil der Ansicht wird die textuelle Beschreibung des Systemelements dargestellt. Eventuell vorhandene Hyperlinks (blaue Schrift) können mittels einfachem (Maus-)Linksklick geöffnet werden. Das Ziel des Hyperlinks wird hierbei mit der im Windows-System hinterlegten Anwendung geöffnet (beispielsweise InternetExplorer oder Adobe Acrobat).

Im unteren Teil der Ansicht werden bis zu sechs Subsystemelemente gleichzeitig dargestellt. Sind mehr als sechs Subsystemelemente vorhanden, wird dies über einen Zähler rechts neben den dargestellten Subsystemelementen angezeigt. Über die Pfeiltasten ober- und unterhalb des Zählers kann mittels einfachem (Maus-)Linksklick durch die Seiten der vorhandenen Subsystemelemente navigiert werden.

Ansicht wechseln

Die beiden Ansichten (Struktur- und Elementansicht) können unmittelbar gewechselt werden. Hierzu ist ein einfacher (Maus-)Linksklick auf den Ansichtswechsel-Knopf im

rechten-unteren Programmfensterbereich ausreichend (vgl. Abbildung 1 und Abbildung 2).

In der Strukturansicht kann die Ansicht zusätzlich durch doppelten (Maus-)Linksklick auf ein beliebiges Systemelement gewechselt werden. Hierdurch wird das gewählte Systemelement selektiert und in der Elementansicht dargestellt.

In der Elementansicht kann die Ansicht zusätzlich durch doppelten (Maus-)Linksklick auf das selektierte Systemelement in die Strukturansicht gewechselt werden.

Ansicht anpassen

Die Struktur- und Elementansicht kann in engen Grenzen angepasst werden. Das Menü zur Anpassung der Ansicht erreichen Sie unter Settings→System-View.

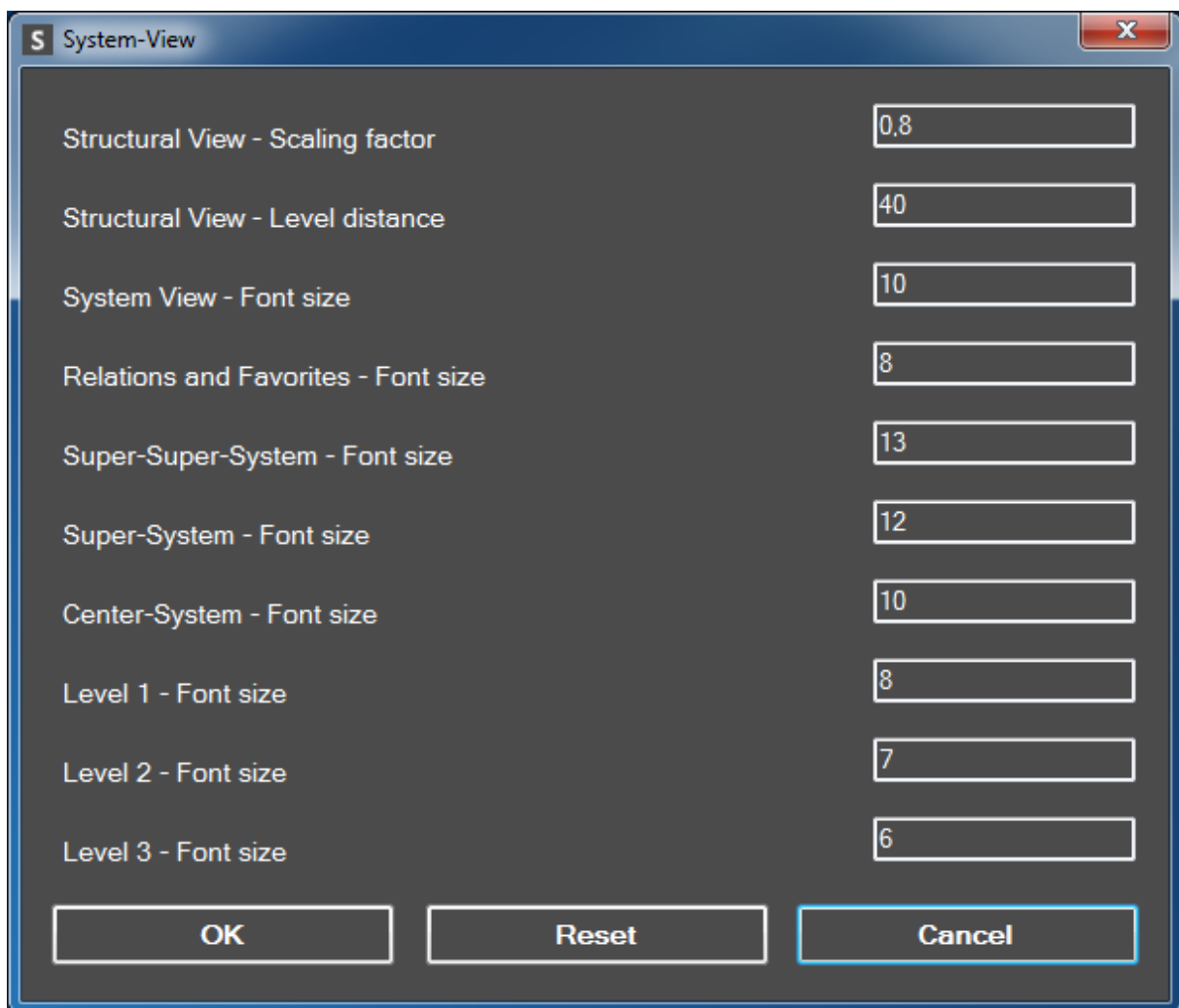


Abbildung 11: Anpassung der Ansicht

Ein System anlegen, editieren und löschen

Ein neues System anlegen

Ein neues Systemelement muss stets als Subsystemelement eines bestehenden Systemelements eingefügt werden.

In der Elementansicht ist hierzu ein einfacher (Maus-)Rechtsklick auf das selektierte Element erforderlich. Im erscheinenden Kontextmenü wählen Sie Add Sub-System und geben Sie im darauffolgenden Fenster den Namen des Systems ein (vgl. Abbildung 12).

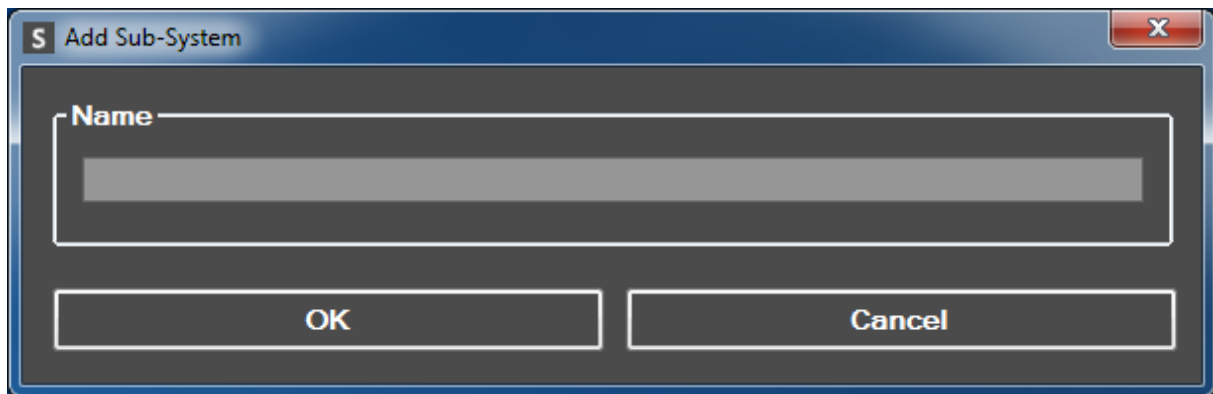


Abbildung 12: Benennung eines anzulegenden Systemelements

Analog kann in der Strukturansicht zu jedem angezeigten Systemelement ein neues (Sub-)Systemelement hinzugefügt werden, sofern das neue Systemelement entsprechend dem selektierten Systemelement und der zugrundeliegenden Anzeigeregeln dargestellt werden wird.

Ein bestehendes System editieren

Ein bestehendes System kann in beiden Ansichten (mit Ausnahme des Supersystems in der Elementansicht) per einfachem (Maus-)Rechtsklick→Edit-System→System-Basis editiert werden (vgl. Abbildung 13).

Im darauffolgenden Fenster kann zum einen der Name des gewählten Systemelements editiert werden. Zum anderen kann das Supersystemelement des gewählten Systemelements gewechselt werden. Hierbei kann aus der Liste aller bestehenden Systemelemente gewählt werden. Diese Liste kann über Type-Filter nach einem Elementtyp gefiltert werden.

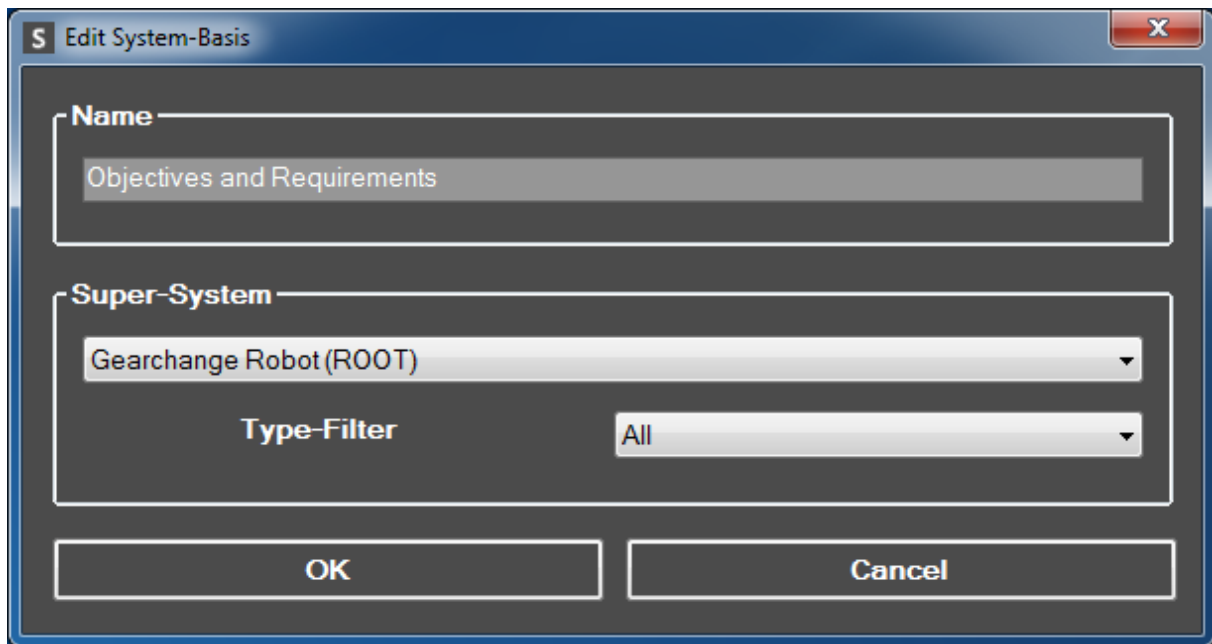


Abbildung 13: Editieren eines bestehenden Systemelements

Editieren der Reihenfolge bestehender Systeme

Die Reihenfolge angezeigter Systemelemente kann sowohl in der Struktur- als auch in der Elementansicht (hier lediglich Subsystemelemente) mittels einfachem (Maus-)Rechtsklick auf ein Systemelement gefolgt von Change Position verändert werden (vgl. Abbildung 14).

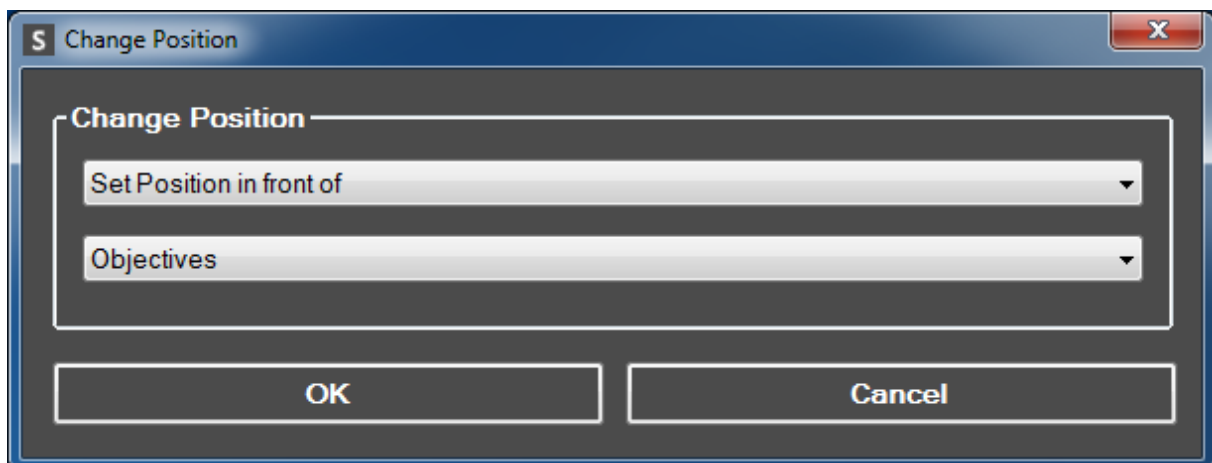


Abbildung 14: Editieren der Reihenfolge angezeigter Systemelemente

Im darauffolgenden Fenster kann definiert werden, ob das gewählte Systemelement vor (Set Position in front of) oder nach (Set Position behind of) einem anderen Systemelement des gleichen Supersystems (in Abbildung 14 Objectives) positioniert werden soll.

Ein bestehendes System löschen

Ein bestehendes Systemelement kann über einfachen (Maus-)Rechtsklick auf das Systemelement und Delete System gelöscht werden. Im darauffolgenden Fenster kann entschieden werden, ob das gewählte Systemelement inklusive aller seiner Subsystemelemente gelöscht werden soll oder nicht oder ob der Vorgang abgebrochen werden soll (vgl. Abbildung 15). Sollen neben dem gewählten Systemelement nicht auch alle Subsystemelemente gelöscht werden, so werden diese Subsystemelemente unterhalb des Supersystemelementes des gerade gelöschten Systemelementes angeordnet.

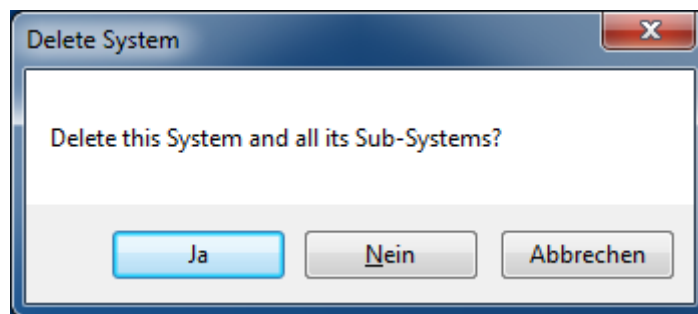


Abbildung 15: Optionen beim Löschen eines bestehenden Systemelements

Eigenschaften eines Systems bearbeiten

Ein Systemelement kann neben seinem Namen durch weitere Eigenschaften näher beschrieben bzw. charakterisiert werden (Elementtyp, Status, Wichtigkeit, Autorisierung und Freitext).

Elementtyp, Status und Wichtigkeit bearbeiten

Der Elementtyp eines Systemelements wird einmalig automatisch zum Zeitpunkt der Erstellung definiert und entspricht dem Systemtyp des zugeordneten Supersystemelements. Der Elementtyp kann mittels einfachem (Maus-)Rechtsklick auf das Systemelement und Edit System→System Properties→System Type→Elementtyp aus Liste gewählt werden.

Der Status eines Systemelements wird einmalig automatisch zum Zeitpunkt der Erstellung definiert und entspricht dem ersten Eintrag der Liste zur Verfügung stehender Status. Der Status kann mittels einfachem (Maus-)Rechtsklick auf das Systemelement und Edit System→System Properties→System Status→Status aus Liste gewählt werden.

Die Wichtigkeit eines Systemelements wird einmalig automatisch zum Zeitpunkt der Erstellung definiert und entspricht dem ersten Eintrag der Liste zur Verfügung stehender Wichtigkeiten. Die Wichtigkeit kann mittels einfachem (Maus-)Rechtsklick

auf das Systemelement und Edit System→System Properties→System Importance→Wichtigkeit aus Liste gewählt werden.

Autorisation bearbeiten

Jedes bestehende Systemelement kann mit einer Autorisation versehen werden, d.h. das Systemelement kann nur noch durch genau einen autorisierten Benutzer editiert werden. Die Autorisation eines Systemelements wird einmalig automatisch zum Zeitpunkt der Erstellung als All definiert, wodurch jeder Benutzer Änderungen an diesem Systemelement vornehmen kann. Eine Änderung der Autorisation kann von jedem berechtigten Benutzer gesetzt bzw. gelöscht werden. Zum Setzen einer Berechtigung muss die Autorisation eines Systems zuvor auf All definiert sein. Mittels einfachem (Maus-)Rechtsklick→Edit System→System Properties→System Authority→Set Authority kann die Berechtigung auf eine Person eingeschränkt werden, die bereits ein System im bestehenden Projekt angelegt hat (vgl. Abbildung 16). Standardmäßig wird der eigene Windows-Account vorgeschlagen.

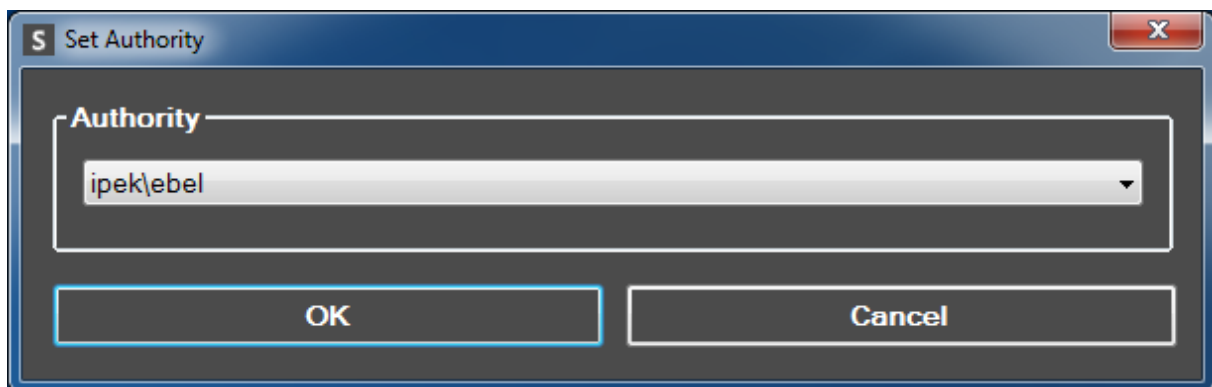


Abbildung 16: Festlegen der Autorisation eines Systemelements

Eine bestehende Autorisation eines Systemelements kann, sofern der eigene Windows-Account hierzu berechtigt ist, über einfachem (Maus-)Rechtsklick→Edit System→System Properties→System Authority→Delete Authority entfernt werden. Hierdurch wird die Autorisation des Systemelements wieder auf All gesetzt.

Freitext bearbeiten

Zur Detaillierung einzelner Systemelemente wird im Rahmen der Modellierung mit SystemModeler auf Freitext zurückgegriffen. Der Freitext eines Systemelements kann in der Elementansicht entweder durch einfachen (Maus-)Rechtsklick→Edit System→System-Description auf das selektierte Systemelement erfolgen oder durch einfachen (Maus-)Rechtsklick→System-Description auf den Bereich der textuellen Beschreibung des selektierten Systemelements. Im darauf folgenden Fenster können Änderungen der textuellen Beschreibung des Systemelements vorgenommen werden (vgl. Abbildung 17).

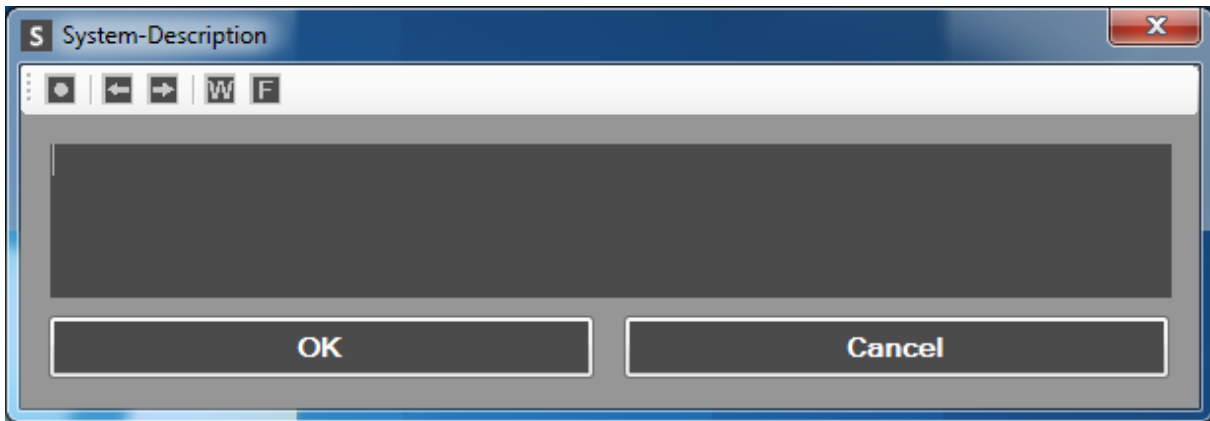


Abbildung 17 Bearbeiten von Freitext

Im oberen Bereich des Freitext-Bearbeitungsfensters (Abbildung 17) stehen Formatierungshilfen zur Verfügung (von links nach rechts: Bullet-Point einfügen, Texteinzug nach links, Texteinzug nach rechts, Hyperlink in das WorldWideWeb, Hyperlink in das Dateisystem).

Relationen eines Systems bearbeiten

Einzelne Systemelemente können im Rahmen der Modellierung mit SystemModeler in Beziehung zueinander gesetzt werden. Hierbei stehen die im Projekt definierten Relationen zur Verfügung.

Eine Relation hinzufügen

Relationen können sowohl in der Struktur- als auch in der Elementansicht lediglich dem selektierten Systemelement hinzugefügt werden. Um eine Relation hinzuzufügen, führen Sie einen einfachen (Maus-)Rechtsklick auf das selektierte Systemelement aus und wählen Sie Add Relation.

Im darauffolgenden Fenster kann eine Relation aus der Liste definierter Relationen sowie das in Beziehung zum selektierten Systemelement zu setzende Systemelement gewählt werden. Hierbei werden in der Liste der Relationen alle ein- und ausgehenden Relationen entsprechend der zuvor definierten Reihenfolge angezeigt. Sind ein- und ausgehende Relation äquivalent (unidirektionale Beziehung) wird diese nur einmal anstatt zweimal in der Liste angezeigt. Die Liste des in Beziehung zu setzenden Systemelements kann über Type-Filter nach einem Elementtyp gefiltert werden.

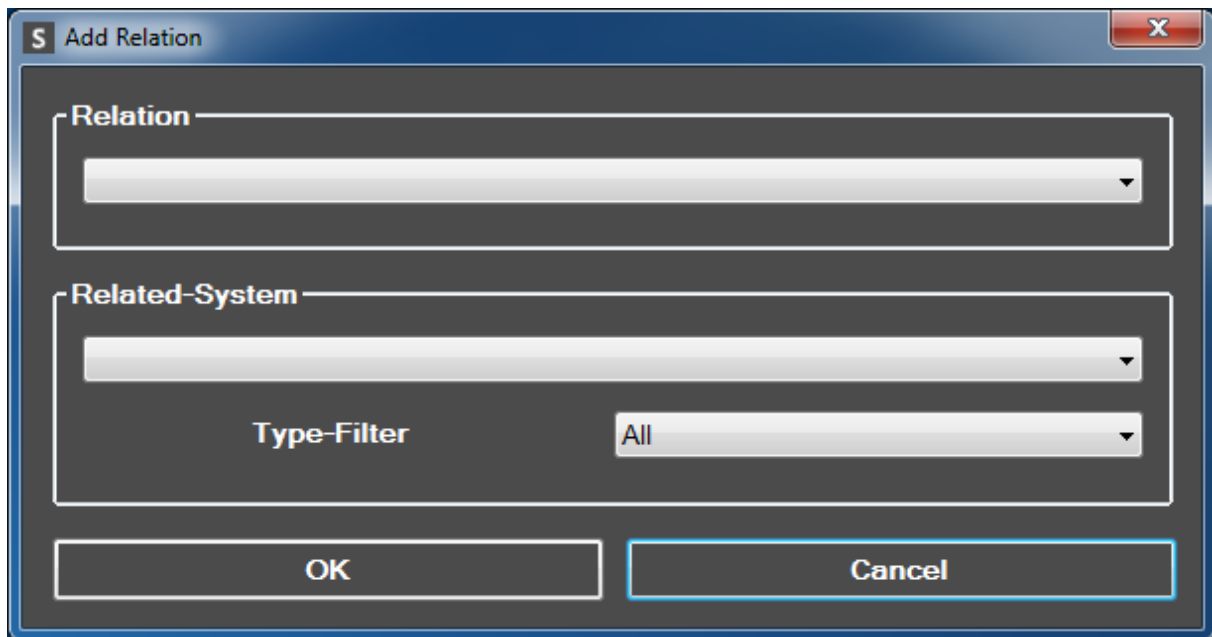


Abbildung 18: Hinzufügen einer Relation

Eine Relation nutzen, bearbeiten und löschen

Bestehende Relationen eines selektierten Systemelements werden im rechten-mittleren Bereich des Programmfensters angezeigt (vgl. Abbildung 1, Abbildung 2 und Abbildung 3). Sind mehr Relationen vorhanden, als gleichzeitig angezeigt werden können, wird dies über einen Zähler unterhalb der aufgelisteten Relationen angezeigt. Über die Pfeiltasten links und rechts des Zählers kann mittels einfachem (Maus-)Linksklick durch die Seiten der vorhandenen Relationen navigiert werden.

Durch einen einfachen (Maus-Rechtsklick) auf eine bestehende Relation, wird das in Beziehung stehende Systemelement selektiert. Wird die Maus über das in Beziehung stehende Element bewegt, wird per Tool-Tip die konkrete Relation angezeigt. Ein einfacher (Maus-)Rechtsklick auf ein in Beziehung stehendes Systemelement öffnet das Kontextmenü, sodass die Relation bearbeitet (→Edit Relation), die Reihenfolge der Relation geändert (→Change Position) oder die Relation gelöscht werden kann (→Delete Relation).

Assoziierte Kopien eines Systems bearbeiten

Eine Assoziierte Kopie stellt eine besondere Kopie bzw. Verknüpfung eines bestehenden Systemelements dar. Eine Assoziierte Kopie kann (wie ein neues Systemelement auch) einem bestehenden (Super-)Systemelement hinzugefügt werden. Jedoch kann eine Assoziierte Kopie weder weitere Subsystemelemente beinhalten, noch können Relationen zur Assoziierten Kopie hinzugefügt werden. Auch eine Bearbeitung der Eigenschaften einer Assoziierten Kopie ist nicht möglich.

Eine Assoziierte Kopie entspricht stets ihrem Original, mit dem einzigen Unterschied, dass sie sich an einer anderen Stelle der Hierarchie einordnet. Hierdurch ist es möglich, die Mehrfachverwendung eines Systemelements abzubilden, ohne unzusammenhängende Kopien zu erstellen. Weiterhin können mittels Assoziierter Kopien individuelle Sichten erstellt werden.

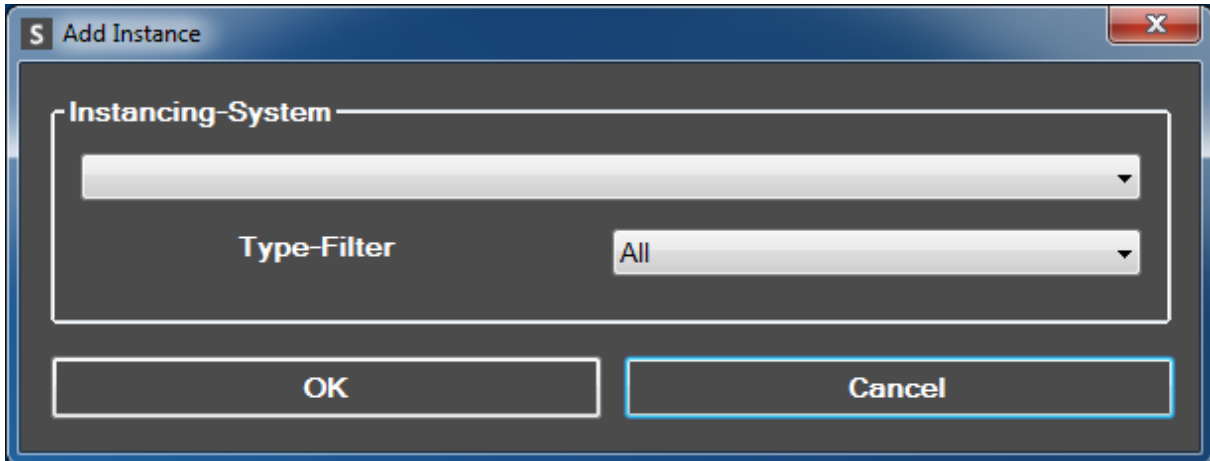


Abbildung 19: Hinzufügen einer Assozierten Kopie

Eine Assoziierte Kopie hinzufügen

Eine Assoziierte Kopie kann jedem Systemelement hinzugefügt werden, dem auch ein neues Systemelement hinzugefügt werden könnte: einfacher (Maus-)Rechtsklick auf das entsprechende Supersystemelement und Add Instance.

Im daraufhin erscheinenden Fenster kann eingegeben werden, von welchem Original eine Assoziierte Kopie eingefügt werden soll (Instancing-System – vgl. Abbildung 19). Die Liste der (Original-)Systemelemente kann über Type-Filter nach einem Elementtyp gefiltert werden.

Eine Assoziierte Kopie nutzen bearbeiten und löschen

Vorhandene Assoziierte Kopien unterscheiden sich in der Darstellung von Systemelementen durch eine graue Füllung (vgl. Abbildung 20).

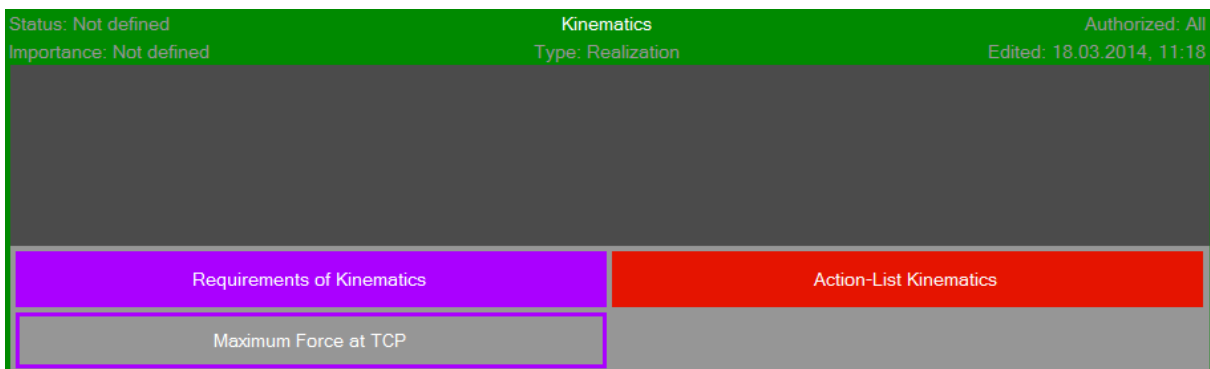


Abbildung 20: Darstellung von Assoziierten Kopien

Durch einfachen (Maus-)Linksklick auf eine Assoziierte Kopie wird dessen Original selektiert und in der aktuell gewählten Ansicht dargestellt (vgl. Abbildung 21). Hierbei werden Assoziierte Kopien, die zum selektierten Systemelement existieren, im Bereich der Relationen dargestellt, wobei diese grau gefüllt sind und der Tool-Tip hierzu Used-in lautet.

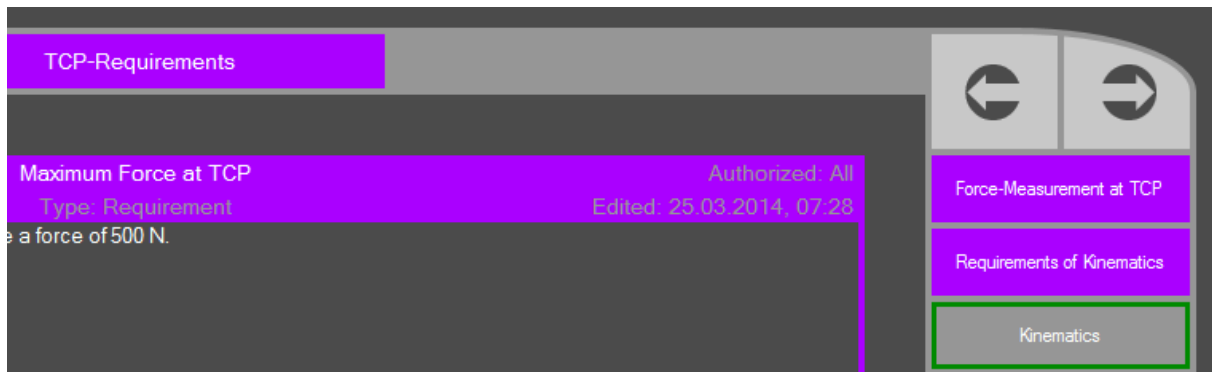


Abbildung 21: Darstellung verwendeter Assoziierter Kopien eines Systemelements

Ein einfacher (Maus-)Rechtsklick auf eine bestehende Assoziierte Kopie öffnet ein Kontext-Menü, über das die Reihenfolge der Assoziierten Kopie geändert (→Change Position) oder die Assoziierte Kopie gelöscht werden kann (→Delete Instance).

Navigieren durch die Ansichtshistorie

Die Ansichtshistorie des SystemModelers ist vergleichbar mit der Seitenhistorie eines Internet-Browsers wie InternetExplorer. Entsprechend erlaubt die Ansichtshistorie einen Wechsel des selektierten Systemelements auf Basis bereits (in einer Programmsitzung des SystemModeler) selektierter Systemelemente. Zur Navigation durch die Ansichtshistorie ist jeweils ein einfacher (Maus-)Linksklick auf einen der Navigationspfeile im rechten-oberen Programmfensterbereich erforderlich (vgl. Abbildung 1, Abbildung 2 und Abbildung 3). Der linke Navigationspfeil führt rückwärts, der rechte vorwärts durch die Historie.

Favoriten erstellen und nutzen

Der SystemModeler bietet die Möglichkeit, benutzerspezifische Favoriten zu erstellen, so dass einzelne Systemelemente, die an unterschiedlichen Orten in der Hierarchie angesiedelt sind, unmittelbar und damit effizient selektiert werden können (vgl. Abbildung 22).

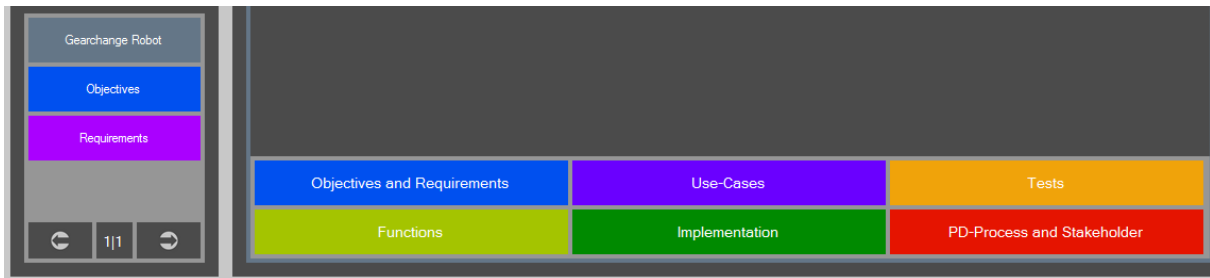


Abbildung 22: Favoriten im SystemModeler

Ein Systemelement kann zur Liste der Favoriten hinzugefügt werden, indem im Bereich der Favoriten ein einfacher (Maus-)Rechtsklick ausgeführt und anschließend Add Favorite gewählt wird. Im daraufhin erscheinenden Fenster kann das als Favorit hinzuzufügende Systemelement aus einer Liste aller vorhandenen Systemelemente ausgewählt werden, wobei als Vorauswahl das aktuell selektierte Systemelement vorgeschlagen wird. Die Liste der Systemelemente kann über Type-Filter nach einem Elementtyp gefiltert werden (vgl. Abbildung 23).

Bestehende Favoriten können über einen einfachen (Maus-)Linksklick selektiert werden, wodurch das entsprechende Systemelement selektiert wird. Weiterhin können bestehende Favoriten in ihrer Reihenfolge bearbeitet sowie gelöscht werden: einfacher (Maus-)Rechtsklick auf einen Favoriten gefolgt von Change Position bzw. Delete Favorite.

Sind mehr Favoriten vorhanden, als gleichzeitig angezeigt werden können, wird dies über einen Zähler unterhalb der aufgelisteten Favoriten angezeigt. Über die Pfeiltasten links und rechts des Zählers kann mittels einfachem (Maus-)Linksklick durch die Seiten der vorhandenen Favoriten navigiert werden.

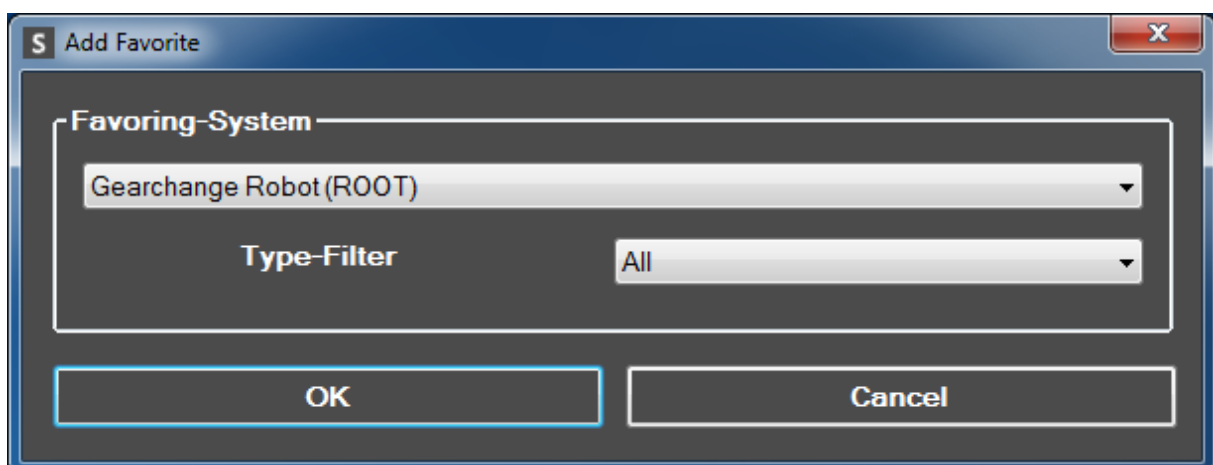


Abbildung 23: Hinzufügen eines Favoriten

Die Suchfunktion nutzen

Über die Suchfunktion des SystemModelers können Systemelemente identifiziert und aufgelistet werden, die einer zuvor definierten Kombination bestimmter Kriterien entsprechen.

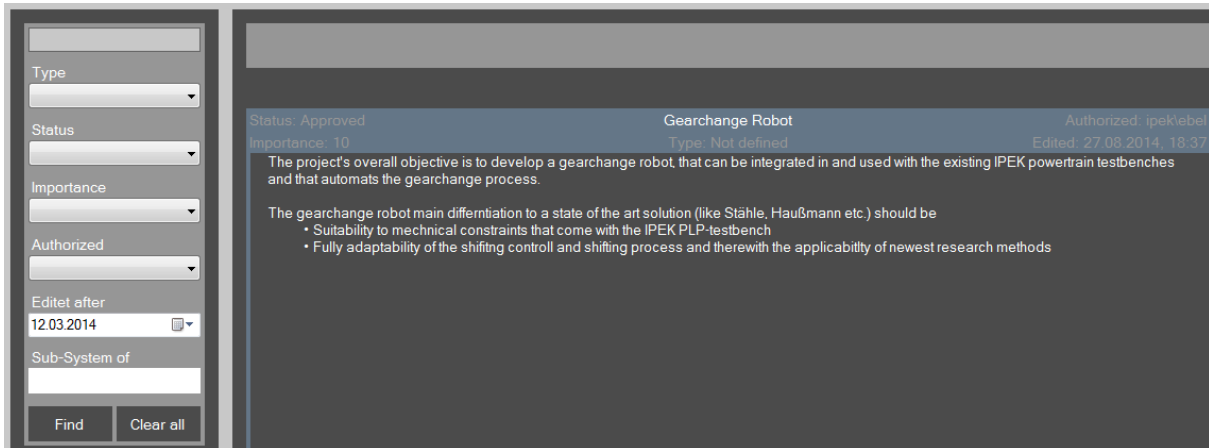


Abbildung 24: Die Suchfunktion des SystemModelers

Eines der wesentlichen Kriterien, das zur Suche nach Systemelementen genutzt werden kann, bildet die Freitext-Suche im oberen-linken Programmfensterbereich. Durch eine Auto-Vervollständigung unterstützt können hier Systemnamen oder auch Fragmente davon eingegeben und zur Suche genutzt werden. Hierbei sind sämtliche mit Leerzeichen abgegrenzten Fragmente logisch UND-Verknüpft, d.h. jedes weitere Fragment verringert in der Regel die Menge der Suchtreffer. Neben der Freitextsuche können die Systemelement-Eigenschaften Elementtyp (Type), Status (Status), Wichtigkeit (Importance) und Autorisation (Authorized) genutzt werden. Zusätzlich kann das Suchergebnis auf Systemelemente eingeschränkt werden, die nach einem bestimmten Datum editiert wurden (Edited after) und die bezüglich der Hierarchie der Systemelemente unterhalb eines bestimmten Systemelements liegen (Sub-System of).

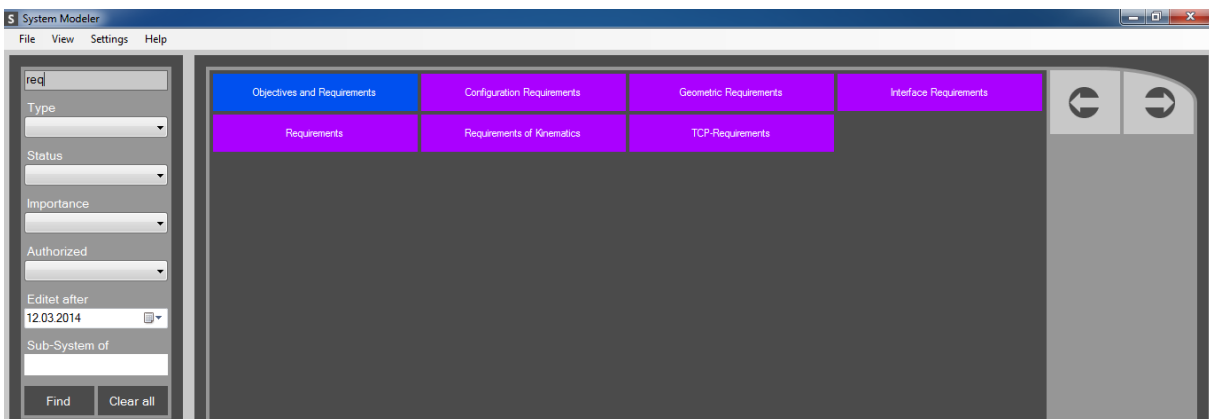


Abbildung 25: Darstellung von Suchergebnissen im SystemModeler

Die Suche nach vorhandenen Systemelementen wird über einen einfachen (Maus-)Linksklick auf den Finden-Knopf (Find) gestartet. Hierbei werden sämtliche Eingaben der Suchfunktion UND-Verknüpft ausgewertet und in einer tabellarischen Ansicht dargestellt (vgl. Abbildung 25). Hierbei werden die Suchergebnisse erst nach Elementtyp und anschließend nach Systemelementname sortiert.

Durch einfachen (Maus-)Linksklick auf ein aufgelistetes Systemelement wird dieses selektiert. Um wieder zur Auflistung der Suchergebnisse bzw. zum zuvor selektierten Systemelement zurück zu kehren, kann die Navigation durch die Ansichtshistorie genutzt werden.

Sämtliche Eingaben der Suchfunktion können über einen einfachen (Maus-)Linksklick auf den Löschen-Knopf (Clear all) gelöscht werden.