

**Band
432**

Verlagsschriftenreihe des Heinz Nixdorf Instituts
Prof. Dr.-Ing. Iris Gräßler (Hrsg.)
Produktentstehung

Dominik Wiechel

Modellbasierte Wirkketten- analyse zur Bewertung technischer Änderungen

Dominik Wiechel

***Modellbasierte Wirkkettenanalyse zur
Bewertung technischer Änderungen***

***Model-based effect chain analysis for
the evaluation of engineering change***

Bibliografische Information Der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

Band 432 der Verlagsschriftenreihe des Heinz Nixdorf Instituts

© Heinz Nixdorf Institut, Universität Paderborn – Paderborn – 2025

ISSN (Online): 2365-4422

ISBN: 978-3-947647-51-4

Das Werk einschließlich seiner Teile ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung der Herausgeber und des Verfassers unzulässig und strafbar. Das gilt insbesondere für Vervielfältigung, Übersetzungen, Mikroverfilmungen, sowie die Einspeicherung und Verarbeitung in elektronischen Systemen.

Als elektronische Version frei verfügbar über die Digitalen Sammlungen der Universitätsbibliothek Paderborn.

Satz und Gestaltung: Dominik Wiechel

Modellbasierte Wirkkettenanalyse zur Bewertung technischer Änderungen

zur Erlangung des akademischen Grades eines
DOKTORS DER INGENIEURWISSENSCHAFTEN (Dr.-Ing.)
der Fakultät Maschinenbau
der Universität Paderborn

genehmigte
DISSERTATION

von
M.Sc. Dominik Wiechel
aus Altena (Westf.)

Tag des Kolloquiums:
Referent:
Korreferent

17.10.2025
Univ.-Prof. Dr.-Ing. Iris Gräßler
Univ.-Prof. Dipl.-Ing. Dr.techn. Hannes Hick

Vorwort

Die Dissertation habe ich während meiner Zeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für Produktentstehung des Heinz Nixdorf Instituts der Universität Paderborn verfasst. Diese umfasst die Ergebnisse meiner wissenschaftlichen Arbeit in verschiedenen Forschungs- und Industrieprojekten, die ich während meiner Tätigkeit am Lehrstuhl leiten und begleiten durfte.

Zunächst gilt mein ausdrücklicher Dank Frau Prof. Dr.-Ing. Iris Gräßler für die kontinuierliche Unterstützung während meiner Tätigkeit am Lehrstuhl. Insbesondere möchte ich mich für das entgegengebrachte Vertrauen bedanken, welches mir als Gruppenleiter für das Thema Model-Based Systems Engineering und als Projektleiter zugesprochen wurde.

Ein herzlicher Dank gilt ebenfalls Herrn Prof. Dipl.-Ing. Dr.techn. Hannes Hick von der Technischen Universität Graz für die Übernahme des Korreferats.

Des Weiteren gilt mein besonderer Dank Herrn Dr.-Ing. Jens Pottebaum für die bereichernde Zusammenarbeit in den Jahren am Lehrstuhl. Das freundschaftliche Miteinander und den Rückhalt, den ich erfahren habe, empfand ich als sehr wertvoll für meine Weiterentwicklung.

Ebenfalls ist es mir ein Anliegen, dem gesamten Lehrstuhl-Team zu danken! Die Zusammenarbeit mit den Kolleginnen und Kollegen hat die Zeit als wissenschaftlicher Mitarbeiter einzigartig gemacht. Mein besonderer Dank gilt Christian Oleff, Daniel Roesmann und Henrik Thiele für die prägende Zusammenarbeit und die allzeit guten Ratschläge. Zusätzlich danke ich meiner Gruppe Model-Based Systems Engineering– Benedikt Grewe, Jan Pfeifer, Sven Rarbach und Deniz Özcan. Ich danke euch für die spannenden Diskussionen, die unvergesslichen Erinnerungen und das freundschaftliche Zusammenarbeiten! Ebenfalls danke ich herzlich den studentischen Hilfskräften Mara Spork, Oliver Schön, Marius Sauerwald und Thomas Hesse für ihren großartigen Einsatz.

Abschließend danke ich meinen Eltern Wolf-Dietrich Wiechel und Silvia Wiechel und meiner Freundin Hanna Mersch vom ganzen Herzen. Ihr habt mich auf verschiedenen Wegen unermüdlich unterstützt. Dabei wart ihr stets motivierend und verständnisvoll. In diesem Sinne – danke für alles!

Paderborn, im Dezember 2025

Dominik Wiechel

Vorveröffentlichungen

- [GW21] GRÄBLER, I.; WIECHEL, D.: Systematische Bewertung von Auswirkungsanalysen des Engineering Change Managements. In: DS 111: Proceedings of the 32nd Symposium Design for X (DFX2021). 2021. S. 1-10.
- [GWR21] GRÄBLER, I.; WIECHEL, D.; ROESMANN, D.: Integrating human factors in the model based development of cyber-physical production systems. In: *Procedia CIRP*, 2021.
- [GWP21] GRÄBLER, I.; WIECHEL, D.; POTTEBAUM J.: Role model of model-based systems engineering application. In: 19th Drive Train Technology Conference (ATK 2021). IOP Conference Series: Materials Science and Engineering, Vol. 1097, No. 1, 2021.
- [GWK+22] GRÄBLER, I.; WIECHEL, D.; KOCH, A.-S.; PREUB, D.; OLEFF, C.: Model-based effect chain analysis for complex systems. *Proceedings of the Design Society*, 2022, 2. Jg., S. 1885-1894.
- [MWR+22] MENNINGER, B.; WIECHEL, D.; RACKOW, S.; HÖPFNER, G.; OLEFF, C.; BERROTH, J.; GRÄBLER, I.; JACOBS, G.: Modellierung und Analyse funktionaler Varianz komplexer technischer Systeme, In: DS 119: Proceedings of the 33rd Symposium Design for X (DFX2022). Hamburg, 2022. S. 1-10.
- [GWO22] GRÄBLER, I.; WIECHEL, D.; OLEFF, C.: Extended RFLP for complex technical systems. In: 2022 IEEE International Symposium on Systems Engineering (ISSE). IEEE, 2022. Vienna, S. 1-8.
- [GWT22] GRÄBLER, I.; WIECHEL, D.; THIELE, H.: Fortschrittskontrolle der Modellierung mechatronischer Produkte. In: Tagungsband der Fachtagung VDI-Mechatronik, 2022.
- [GWK+23] GRÄBLER, I.; WIECHEL, D.; KOCH, A. S.; STURM, T.; MARKFELDER, T.: Methodology for Certification-Compliant Effect-Chain Modeling. MDPI Systems, 2023.
- [GW23] GRÄBLER, I.; WIECHEL, D.: Customized impact analyses for technical engineering changes. In: 2023 18th Annual System of Systems Engineering Conference (SoSe). IEEE, 2023. S. 1-6.
- [GBW+23] GRÄBLER, I.; BODDEN, E.; WIECHEL, D.; POTTEBAUM, J.: Defense-in-Depth als neues Paradigma der sicherheitsgerechten Produktentwicklung: interdisziplinäre, bedrohungsbewusste und lösungsorientierte Security. In: Zeitschrift Konstruktion Nr 11-12, 2023.
- [GWR24] GRÄBLER, I.; WIECHEL, D.; RARBACH, S.: Model-based impact analysis in dynamic Systems-of-Systems. In: *Procedia CIRP*, 2024.
- [GWO+24] GRÄBLER, I.; WIECHEL, D.; ÖZCAN, D.; TAPLICK, P.: Tailored metrics for assessing the quality of MBSE models. In: *Proceedings of the Design Society*, 2024. (
- [GWG24] GRÄBLER, I.; WIECHEL, D.; GREWE, B.: Bedarfsgerechte Wirkkettenmodellierung und -analyse komplexer akusto-mechatronischer Systeme. In: *Zeitschrift Konstruktion*, 2024.
- [GPP+24] GRÄBLER, I.; POTTEBAUM, J.; PREUB, D.; WIECHEL, D.; OLEFF, C.: KI-gestützte Anforderungsentwicklung für komplexe technische Systeme. In: *Zeitschrift Konstruktion*, 2024. (eingereicht)
- [GWR24] GRÄBLER, I.; WIECHEL, D.; RARBACH, S.: Model-based impact analysis in dynamic System of Systems. In: *Procedia CIRP*, 128, S. 585-590, 2024.
- [GPH+24] GRÄBLER, I.; POTTEBAUM, J.; HOLLAND, M.; WIECHEL, D.; DICKOPF, T.; STJEPANDIĆ, J.: Leveraging Data Ecosystems in Model-Based Systems Engineering for Ecological, Circular Added Value. In: *Engineering For Social Change*, S. 175-184, 2024.
- [GRW24] GRAESSLER, I.; RARBACH, S.; WIECHEL, D.: Artifact-Oriented Tailoring Approach for Model-Based Impact Analysis. In: *IEEE International Symposium on Systems Engineering (ISSE)*, S. 1-8, 2024.

Zusammenfassung

Der unsystematische und ineffiziente Umgang mit technischen Änderungen führt zu hohen Personalkosten durch Mehraufwand, einer reduzierten Entscheidungsqualität und kann in Projektfehlschlägen resultieren. Herausforderungen beim Umgang mit technischen Änderungen liegen in der Berücksichtigung relevanter Ursache-Wirkungs-Zusammenhänge, der quantitativen und qualitativen Bewertung der Auswirkungen und der effizienten prozessualen Einbindung. Mithilfe eines modellbasierten Lösungsansatzes lassen sich die Auswirkungen technischer Änderungen schneller bewerten und fundierte Entwicklungsentscheidungen treffen. In der vorliegenden Dissertation wird eine Methodik für die Entwicklung modell-basierter Wirkkettenanalysen zur Bewertung technischer Änderungen beschrieben. Durch die modellbasierte Wirkkettenanalyse können direkt oder indirekt von der Änderung betroffene Modellelemente identifiziert werden. Das Datenmodell wird mithilfe von Abfragesprachen analysiert und die Auswirkungen werden mithilfe von integrierten Metriken quantifiziert. Die Ergebnisse der Wirkkettenanalyse werden in rollenspezifischen Sichten bereitgestellt, wodurch eine bedarfsgerechte Entscheidungsunterstützung gewährleistet wird. Die Validierung anhand von vier industriellen Fallstudien belegt die Praxistauglichkeit der Methodik. Die Methodik verbindet das Änderungsmanagement mit den Vorteilen einer modellbasierten Systementwicklung und bildet somit einen synergetischen neuen Forschungsbereich.

Abstract

The unsystematic and inefficient handling of technical changes leads to high personnel costs due to additional work, reduced decision-making quality and can result in project failures. The challenges in dealing with technical changes arise in the consideration of relevant cause-and-effect relationships, the quantitative and qualitative assessment of the effects and the efficient process integration. With the help of a model-based solution approach, the effects of technical changes can be assessed more quickly and profound development decisions can be made. This dissertation describes a methodology for the development of model-based effect chain analyses for the evaluation of technical changes. Model-based effect chain analyses can be used to identify model elements that are directly or indirectly affected by the change. The data model is analysed using query languages and the effects are quantified using integrated metrics. The results of the impact chain analysis are provided in role-specific views, which ensures needs-based decision support. The validation based on four industrial case studies demonstrates the practical applicability of the methodology. The methodology combines change management with the advantages of model-based system development and thus forms a synergetic new area of research.

Modellbasierte Wirkkettenanalyse zur Bewertung technischer Änderungen

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation und Zielsetzung	1
1.2	Wissenschaftliches Vorgehen.....	6
2	Grundlagen und Stand der Forschung	9
2.1	Grundlagen.....	9
2.1.1	Technische Änderungen in dem Produktlebenszyklus	9
2.1.2	Änderungsmanagement	10
2.1.3	Systems Engineering.....	13
2.1.4	Model-based Systems Engineering	18
2.1.5	Wirkkettenmodellierung	20
2.1.6	Graphentheorie.....	22
2.2	Stand der Forschung	23
2.2.1	Lösungsansätze für EC-Auswirkungsanalysen.....	23
2.2.2	Lösungsansätze für die durchgängige Modellierung	31
2.2.3	Lösungsansätze für die Integration in den ECM Prozess	35
2.3	Forschungslücke und Handlungsbedarf	37
3	Anforderungen an die Methodik.....	39
4	Konzeption der Methodik zur modellbasierten Wirkkettenanalyse. 43	
4.1	Konzeptionelles Architekturmodell.....	43
4.2	Metadatenmodell	47
4.3	MBSE-Rollenmodell.....	50

5	Entwicklung der Methodik zur modellbasierten Wirkkettenanalyse technischer Änderungen	51
5.1	Schritt 1: Zieldefinition der Wirkkettenanalyse	53
5.2	Schritt 2: Ermittlung von Informationen	62
5.3	Schritt 3: Modellierung des Wirkkettenmodells	64
5.4	Schritt 4: Durchführung der Wirkkettenanalyse	71
5.5	Software-gestützte Implementierung	76
5.6	Einbindung in die Produktentwicklung	83
6	Validierung in der industriellen Praxis	87
6.1	Fallstudien aus den Bereichen Landtechnik, Spritzgusstechnik und Sondermaschinenbau	89
6.1.1	Fallstudie 1: Raupenlaufwerk einer Landmaschine	90
6.1.2	Fallstudie 2: Auswuchtmaschine	92
6.1.3	Fallstudie 3: Spritzgussartikel für E-Mobilität	95
6.2	Fallstudie 4: Fensterheber aus der Automobilindustrie	97
6.3	Validierung der Anforderungen	99
6.3.1	Eingangsdaten	100
6.3.2	Methoden, Modelle und Hilfsmittel	102
6.3.3	Anwendung	106
6.3.4	Ergebnisse	110
6.4	Methodik-Erfolg	114
7	Zusammenfassung & Ausblick	117
	Literaturverzeichnis	121
	Anhang	135

Anhang

A1	Übersicht über Forschungsprojekte und Workshops mit Industrievertretern	139
A2	Literaturrecherchen.....	141
A2.1	Identifikation relevanter Literatur- und Fallstudien	141
A2.2	Liste an Ansätzen zur modellbasierten Auswirkungsanalyse	144
A2.3	Kriterien zur Auswahl der EC-Auswirkungsanalyse-Ansätze.....	150
A2.4	Steckbrief EC-Auswirkungsanalyse Methoden	152
A2.5	Gegenüberstellung der Ansätze anhand der Elemente	167
A2.6	Liste an Ansätzen zu Modellierungsmethoden	168
A2.7	Steckbriefe MBSE-Modellierungsmethoden	170
A2.8	Fragebogen zur Ermittlung der Herausforderungen	173
A3	Anforderungen und Anwendungsfälle.....	176
A3.1	Anwendungsfälle der Auswirkungsanalyse.....	176
A3.2	Nummerierung der Anwendungsfäll	177
A3.3	Anforderungsworkshops zur Wirkkettenanalyse	178
A3.4	Interviews für Anforderungen an die Wirkkettenmodellierung in Automobilindustrie	179
A4	Konzeption der MECA-Methodik	180
A4.1	Vorgehen zur Erstellung des Rollenmodells	180
A4.2	Modellierungsregeln	181
A4.3	Ergebnisdimensionen	181
A4.4	RFLPV Handouts.....	184
A4.5	Vollständige Ansicht der Customizations.....	186
A4.6	RFLPV ² -Profil	186
A4.7	Metamodell Graphdatenbank neo4j.....	188

A4.8	Ursache-Wirk-Muster für Wirkkettenanalysen	190
A4.9	Basisalgorithmen	191
A4.10	MECA Poster	195
A4.11	Lessons Learned	200
A5	Fallstudie kollaborativer 6-achsiger Roboterarm Nyrio Ned 2.....	202
A5.1	Übersicht der Fallstudie	202
A5.2	Modelle Roboterarm (Nyrio Ned 2)	204
A5.3	Greifvorgang bei Pick-and-Place-Bewegung	210
A5.4	Neo4j Import Roboterarm	212
A5.5	Dashboards Roboterarm	213
A6	Zusatzinformationen ImPaKT	216
A6.1	Übersichtsposter Fallbeispiele	216
A6.2	Übersicht zur Anwendung der MECA-Methodik in Fallstudien	216
A6.3	Partialmodell je Fallstudie	218
A7	Fallstudie CLAAS und Eisengiesserei Baumgarte	220
A8	Fallstudie Hofmann	226
A9	Fallstudie Hadi Plast.....	238
A10	Validierung	247
A10.1	Bewertungsmethoden nach Hevner	247
A10.2	Aufbau Prüfstand	248
A10.3	Kriterien Planung des Prüfablaufs	249
A10.4	Validierungs-Workshops (Zyklus 1-3)	250
A10.5	Validierungs-Fragebogen	251
A10.6	Auswertung der Fragebögen	259

A11	Code.....	260
A11.1	Code zum Import des Metadatenmodells	260
A11.2	Dashboards Roboterarm	267

Abkürzungsverzeichnis

AECIA	Advanced Engineering Change Impact Approach
BPMN	Business Process Modeling Notation
CAD	Computer Aided Design
C-FAR	Change Favorable Representation
CIT	CLAAS Industrietechnik GmbH
CO ²	Kohlenstoffdioxid
CPM	Change Prediction Method
CPM-PDD	Characteristics-Properties Modeling / Property-Driven Development
CPS	Cyber Physisches System
D	Defizit
DSM	Design Structure Matrix
EB	Eisengiesserei Baumgarte GmbH
EC	Engineering Change
ECM	Engineering Change Management
ECO	Engineering Change Order
ECR	Engineering Change Request
EDS	Ergebnisdimensionsschablone
ERP	Enterprise Ressource Planning
FS	Fallstudie
FAS4M	
HMF	Hofmann Mess- und Auswuchttechnik GmbH
HP	Hadi Plast GmbH & Co. KG Kunststoff-Verarbeitung
KSKM	Kaiserslauterner Systemkonkretisierungsmodell
MBSE	Model-based Systems Engineering
M-CAD	Mechanical Computer-Aided Design
MECA	Model-based Effect Chain Analysis

OCL	Object Constraint Language
OOSEM	Object-oriented Systems Engineering Method
PLM	Product Lifecycle Management
RC-Car	Remote Control Car
RFLP	Requirements, Functions, Logical Elements, Physical Elements
RM-SEA	Role Model for Systems Engineering Application
SAL	Smart Automation Laboratory
SE	Systems Engineering
SoI	System of Interest
SoS	System of Systems
SQL	Structured Query Language
SuS	System Usability Scale
SW	Software
SysML	Systems Modeling Language
SysMOD	System Modeling Toolbox
TIM	Traceability Information Modellen
TSM	Total System Model
UML	Unified Modeling Language
VPE	Virtuelle Produktentwicklung
z. B.	zum Beispiel

Symbolverzeichnis

α	Einlagerungswinkel zur Berechnung der Hertz'schen Pressung beim Auswuchtprozess
A_{proj}	Projizierte Fläche eines Spritzgussartikels
B	Abstand der Laufrollen
k_W	Anzahl der Kavitäten eines Spritzgusswerkzeugs
P_W	Werkzeuginnendruck eines Spritzgusswerkzeugs
R_R	Radius des Rotors
R_S	Radius der Laufrollen
Z	Zuhaltekraft

1 Einleitung

1.1 Motivation und Zielsetzung

Technische Änderungen sind während der Entwicklung komplexer technischer Systeme unvermeidbar [HCC13, S. 473]. Technische Änderungen sind Modifikationen an Entwicklungsartefakten wie Zeichnungen und technische Spezifikationen, die im Rahmen des Entwicklungsprozesses bereits freigegeben sind [Las21, S. 919ff.]. Das übergreifend vertretene Ziel eines Unternehmens beim Umgang mit technischen Änderungen ist die schnelle Umsetzung der technischen Änderung ohne Reduzierung der Wirtschaftlichkeit [IRA20, S. 4520]. Bei der operativen Handhabung von technischen Änderungen stehen Entwickler¹ vor der Herausforderung, Änderungsauswirkungen effektiv und effizient abzuschätzen [LWM12, S. 32]. Eine technische Änderung kann unterschiedliche Auswirkungen haben, beispielsweise eine Erhöhung der Lastanforderungen, dadurch notwendige konstruktive Anpassungen der Wandstärken und eine Aktualisierung des Softwarecodes. Die Änderungsauswirkungen müssen von den Entwicklern frühzeitig und präzise bewertet werden, da die Kosten für die Umsetzung einer technischen Änderung mit jeder Phase des Produktlebenszyklus exponentiell um den Faktor 10 steigen [FGN00, S. 173]. In Bild 1-1 wird verdeutlicht, dass frühzeitige technische Änderungen zu geringeren Aufwänden führen, beispielsweise da die Eigenschaftsabsicherung noch nicht durchgeführt wurde oder noch keine Investitionen in Produktionsanlagen getätigt wurden [SMS12, S. 292]. Für Unternehmen ist daher die frühzeitige Identifikation und effiziente und effektive Bewertung der Änderungsauswirkungen essenziell, um bei Änderungsanfragen gezielt vorzugehen und Kosten zu minimieren.

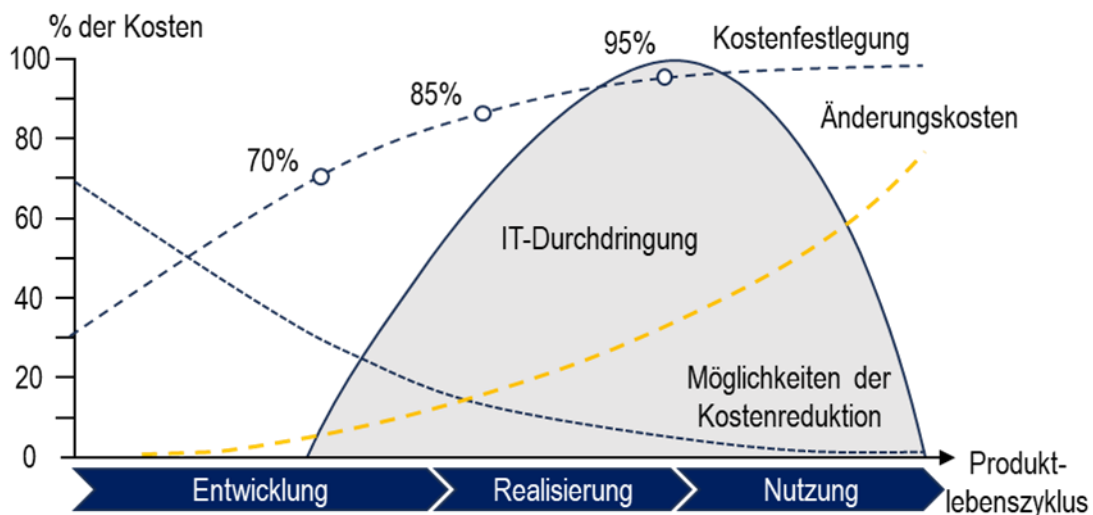


Bild 1-1: Änderungskosten im Produktlebenszyklus nach Eigner [Eig09, S. 23]

¹ Zugunsten der Verständlichkeit der wissenschaftlichen Inhalte wird in dieser Arbeit das generische Maskulinum verwendet. Sämtliche Personenbezeichnungen gelten für alle Geschlechter.

Im Folgenden werden vier Defizite beschrieben, die Entwickler an einem frühzeitigen, effektiven und effizienten Umgang mit technischen Änderungen hindern. Zur Identifikation der Defizite wurde eine Literaturstudie nach MACHI und MCEVOY [MM12] durchgeführt (siehe Anhang A2.1, S. 141).

Ein Defizit (D1) ist die **manuelle subjektive Bewertung der Änderungsauswirkungen**. Änderungsbedarfe müssen nach dem Auftreten von einem oder mehreren Akteuren im Unternehmen manuell analysiert werden. Die manuelle Analyse führt bei einer Vielzahl von technischen Änderungen während eines Entwicklungsprojekts zu einer hohen Kapazitätsbindung. Ein Beispiel ist die Bearbeitung von 40.000 Änderungsbedarfen während eines dokumentierten Entwicklungsprojekts eines Sensor-Systems aus dem Militärbereich [GWB09, S. 1]. Die Heterogenität dieser Änderungsbedarfe führt dazu, dass die Informationen zur Bewertung der Änderung an die Art der technischen Änderung angepasst werden müssen [SLG15, S. 208]. Ergebnisse aus drei Studien zeigen, dass der Umgang mit technischen Änderungen 20 % bis 30 %, in Ausnahmefällen sogar bis zu 70 % der zeitlichen Entwicklungskapazität bindet [LWM12, S. 2, DKJ05, S. 4, LR98, S. 2]. Daher müssen Entwickler die Auswirkungen technischer Änderungen schnell abschätzen und auf diese reagieren, um im Wettbewerb mit anderen Unternehmen bestehen zu können [KJ16, S. 2]. Aktuell hängt die Bewertung der Auswirkungen in hohem Maße von subjektivem Expertenwissen ab [FGN00, S. 174, IRS18, S. 146, BS22, S. 74]. Die objektive Bewertung der Änderungsauswirkungen anhand exakter Werte ist herausfordernd [HYM03, S. 483]. Interviews mit 17 Chefsingenieuren und Systemverantwortlichen belegen, dass die Bewertung von technischen Änderungen nicht ausschließlich auf der Grundlage des Expertenwissens Einzelner erfolgen sollte, sondern auf Basis des kollektiven Wissens. [CSE04, S. 790]. Die Abhängigkeit von dem subjektiven Wissen einzelner Experten limitiert die Aussagekraft bei der Bewertung von Auswirkungen durch voreingenommene Sichtweisen der Experten [IRA20, S. 4522]. Daher besteht der Bedarf, die Bewertung der Auswirkungen basierend auf objektiven Daten und Bewertungskriterien durchzuführen [GW21, S. 8].

Das zweite Defizit (D2) ist die **unsystematische Berücksichtigung von Änderungsausbreitungen** (Änderungspropagation) [CSE04, S. 788]. Eine initiale Änderung an einem Systemelement kann sich entlang von Ursache-Wirkungs-Zusammenhängen auf weitere Bestandteile des Systems ausbreiten [Ham13, S. 5]. Innerhalb des entstehenden Ursache-Wirkungs-Musters ist die initiale Änderung die Ursache für nachfolgende Änderungen [SMS12, S. 292]. Effekte wie Propagationseffekte [BW23, S. 117], Lawinen-Effekte [ECZ04, S. 18] und Rückkopplungsschleifen [FGN00, S. 172] müssen in Abhängigkeit von dem EC betrachtet werden. Anhand eines exemplarischen Beispiels aus einem bilateralen Forschungsprojekt (Paragon in Anhang A1, S. 139) wird gezeigt, wie eine Änderungspropagation verläuft:

Die Funktion des Einklemmschutzes eines Dachantriebs stellt sicher, dass das Schiebedach eines Kraftfahrzeugs automatisch zurückläuft, wenn eine Klemmkraft von mehr als 100 Newton gemessen wird (vgl. UN ECE R 21 [Eur03]). Bei ersten Tests während der Verifikation hat sich herausgestellt, dass der Antriebsmotor durch die negativ gerichtete Klemmkraft beschädigt wird. Als technische Änderung soll ein neuer Motorantrieb integriert werden, der die Kraft ohne Beschädigung aufnehmen kann. Durch die Integration des neuen Motors muss das Gehäuse des Dachantriebs geometrisch angepasst werden. Die konstruktive Anpassung des Gehäuses führt zu Anpassungen der Entkopplungselemente, welche den Körperschall des Motorantriebs dämpfen. Ebenfalls muss die Software des Steuergeräts aktualisiert werden, um die automatischen und manuellen Schließfunktionen weiterhin durch Bedienelemente wie eine Fernbedienung, einen manuellen Schalter oder ein Borddisplay zu realisieren. Das Beispiel zeigt, dass die Konsequenzen der Klemmkräfte nicht nur Auswirkungen auf den Antriebsmotor haben, sondern durch Propagation auch das Gehäuse, die Entkopplungselemente und die Software betreffen.

Aktuell werden Änderungspropagationen nicht systematisch entlang interdisziplinärer Ursache-Wirkungs-Beziehungen abgebildet. Für die systematische Analyse der Propagationspfade müssen Entwicklungsartefakte bis auf parametrischer Ebene abgebildet werden [MJL17, S. 678]. Sofern die Propagationen nicht explizit abgebildet werden, können Änderungsausbreitungen nicht objektiv und automatisiert analysiert werden.

Ein weiteres Defizit (D3) ist die **fehlende Durchgängigkeit der Datengrundlage**. Technische Änderungen können sich auf alle Daten und Informationen auswirken, die während des Produktentwicklungsprozesses entstehen [BW23, S. 117]. Aktuell werden Daten und Informationen in separaten Datensilos abgelegt und verwaltet, wodurch keine Verknüpfungen zwischen den Daten existieren und nur ungenaue Analyseergebnisse gefolgert werden können [IRA20, S. 4507]. Die Datengrundlage ist daher nicht durchgängig miteinander vernetzt, wodurch die Auswirkungen technischer Änderungen auf einer unvollständigen Datengrundlage bewertet werden. Ein Beispiel ist die fehlende Berücksichtigung von Verknüpfungen zwischen Modellen auf Systemarchitekturebene und weiteren Modellen aus der disziplinspezifischen Implementierung, beispielsweise Geometriemodellen, Simulationsmodellen und Kalkulationsmodellen [GP22, S. 14]. Änderungsprozesse ohne IT-Unterstützung wurden bereits von den Unternehmen im Jahr 1999 als unzureichend und ineffizient anerkannt [HM99, S. 31]. IT-Systeme können die notwendige Infrastruktur bereitstellen, um Entwickler im technischen Änderungsmanagement zu unterstützen [KJ16, S. 8]. Aufgrund der Vielzahl von separaten Daten und Partialmodellen müssen sowohl existierende IT-Lösungen zur Unterstützung des Änderungsprozesses integriert werden [PM98, S. 11], als auch eine Struktur zur Vernetzung der Daten festgelegt werden [JL18, S. 181]. Zur Gewährleistung der Durchgängigkeit müssen die IT-Systeme Schnittstellen zwischen den Modellen bereitstellen und standortunabhängige Zugriffe auf die Daten gewährleisten [LWM12, S. 32]. Daher besteht der Bedarf zur Verbesserung der Kollaboration und Kommunikation zwischen verschiedenen Stakeholdern im Änderungsprozess [LWM12, S. 33, IRS18, S. 149].

Zusammenfassend müssen Entwickler bei der systematischen und IT-gestützten Erstellung einer durchgängigen Datengrundlage entlang von technischen Ursache-Wirkungs-Beziehungen unterstützt werden.

Ein weiteres Defizit (D4) sind **fehlende maßgeschneiderte Analyseergebnisse** zur Bewertung der Machbarkeit einer technischen Änderung durch beteiligte Personen. Die Personen aus unterschiedlichen Disziplinen, Bereichen und Unternehmen sind in den Änderungsmanagement-Prozess eingebunden und müssen Auswirkungen technischer Änderungen aus ihrer individuellen Perspektive bewerten. [HYM03, S. 484] Studienergebnisse zeigen, dass bei unkritischen Änderungen unternehmensübergreifend durchschnittlich fünf Personen eingebunden sind, bei kritischen Änderungen sogar acht Personen [LWM12, S. 8]. Dies resultiert in hoher Kapazitätsbindung und organisatorischen Aufwänden, welche zu einer langen Bearbeitungszeit der Änderung führen. – im Durchschnitt zwischen 25,5 und 39,8 Arbeitstagen [LWM12, S. 9]. Die lange Bearbeitungszeit resultiert aus operativen Tätigkeiten zur Analyse der Änderung und aus ressourcenintensiven Abstimmungen, beispielsweise in Änderungskomitees [PM98, S. 3]. Daher besteht der Bedarf nach flexibel anpassbaren Analysefunktionalitäten [GW21, S. 7] und Hilfsmitteln [Sta22, S. 410]. Hierzu müssen relevante Informationen über den EC an die Bedarfe der Stakeholder angepasst werden [SLG15, S. 209]. Maßgeschneiderte Analyseergebnisse bieten das Potenzial, die Informationsbedarfe der beteiligten Personen automatisiert zu folgern und so überflüssige Abstimmungsaufwände zu reduzieren.

In Bild 1-2 werden die genannten Defizite visualisiert und in den logischen Ablauf einer technischen Änderung eingeordnet.

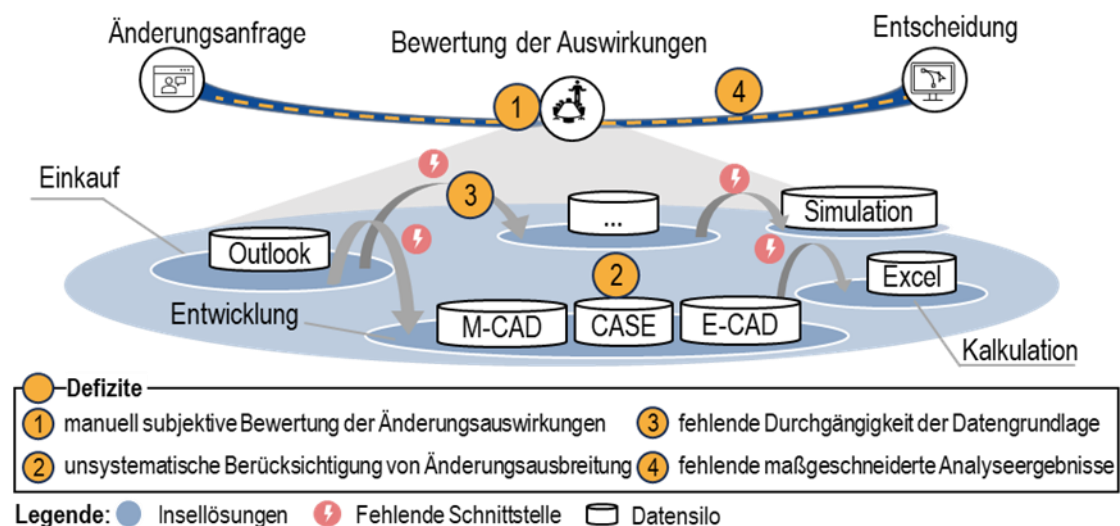


Bild 1-2: Defizite bei der Bewertung von Änderungsauswirkungen

Existierende Ansätze werden nur selten im industriellen Anwendungszusammenhang validiert, wodurch eine Lücke zwischen Theorie und Praxis entsteht und die prozessuale Integration vernachlässigt wird [IRS18, S. 150, BW23, S. 144]. Daher wird die Relevanz der Defizite durch die Analyse von acht Umfragen mit industriellen

Experten (siehe Anhang A2.8, S. 173) abgesichert, die vom Verfasser im Rahmen des erfolgreich abgeschlossenen Verbundforschungsprojekts ImPaKT (in Anhang A1, S. 139 und [Uni24]) durchgeführt wurden.

Laut einer Studie ist die Formalisierung der Analyse von Änderungsauswirkungen ein zentraler Erfolgsfaktor, um eine Leistungssteigerung im Umgang mit technischen Änderungen zu ermöglichen [Abe07, S. 20]. Die Formalisierung umfasst sowohl den Prozess zur Analyse der Auswirkungen eines Änderungsbedarfs als auch die systematische Durchführung der Auswirkungsanalyse. Die Auswirkungsanalysen unterstützen Entwickler bei der Entscheidungsfindung im Umgang mit Änderungsbedarfen, indem potenzielle Effekte einer Änderung identifiziert und resultierende Folgen abgeschätzt werden [GW21, S. 3]. Bisherige Modelle, Methoden und Hilfsmittel zur Auswirkungsanalyse technischer Änderungen beruhen auf ähnlichen Konzepten, um Änderungen zu modellieren und zu analysieren [BW23, S. 117]. Die genannten Defizite werden dabei nicht durch einen ganzheitlichen Ansatz gelöst. Das subjektive Expertenwissen der Anwender wird nicht gezielt durch objektive Analyseergebnisse unterstützt (D1). Entwickler erhalten keine methodischen Hilfsmittel zur eigenständigen Erstellung einer durchgängigen Datengrundlage ihrer Systeme, obwohl diese notwendig ist, um Änderungsauswirkungen und zugehörige Änderungspropagationen abzubilden und aufwandsarm zu bewerten (D2/D3). Aus prozessualer Sicht werden die Auswirkungsanalysen nicht individuell an die Informationsbedarfe der Anwender angepasst (D4).

Die Defizite verdeutlichen, dass kein ganzheitlicher Ansatz existiert, welcher die Auswirkungen technischer Änderungen auf Basis von Ursache-Wirkungs-Beziehungen bewertet und sogenannte Wirkkettenanalysen¹ umsetzt. Ausgehend von den beschriebenen Defiziten wird folgende Zielsetzung für die Dissertation abgeleitet:

Entwicklung einer Methodik zur modellbasierten Wirkkettenanalyse zur Bewertung von Änderungsauswirkungen in der Produktentwicklung

Zur Strukturierung des weiteren Lösungsvorgehens werden die drei folgenden Forschungsfragen aus der Zielsetzung abgeleitet:

Forschungsfrage 1: Welche Aktivitäten sind notwendig, um Auswirkungen technischer Änderungen entlang von Ursache-Wirkungs-Beziehungen zu bewerten?

Forschungsfrage 2: Welche Modellierungsansätze eignen sich, um Änderungspropagationen systematisch und durchgängig entlang von Ursache-Wirkungsbeziehungen abzubilden?

Forschungsfrage 3: Was sind geeignete Analyseverfahren, um Auswirkungen technischer Änderungen maßgeschneidert und objektiv zu bewerten?

¹ Im Folgenden wird der Begriff Wirkkettenanalyse verwendet, um den vom Autor verfolgten ganzheitlichen Ansatz gegenüber existierenden Lösungsansätzen abzugrenzen.

1.2 Wissenschaftliches Vorgehen

Die Dissertationsschrift wird im Bereich der ingenieurwissenschaftlichen Forschung erarbeitet. In dem Bereich existieren wissenschaftliche Vorgehensweisen wie der Ansatz nach ULRICH [Ulr81], die Design Research Methodology nach BLESSING und CHAKRABARTI [BC09] und Design Research in Information Systems nach HEVNER und CHATTERJEE [HCG10]. In dem Promotionsvorhaben werden industrielle Endanwender eingebunden, um praxistaugliche Lösungen zu entwickeln. Daher wird auf dem Ansatz nach ULRICH [Ulr81] aufgebaut, welcher den Praxiszusammenhang als Hauptaufgabe in die Vorgehensschritte einbezieht [Ulr81, S. 10]. Das Vorgehen ist in Bild 1-3 dargestellt.

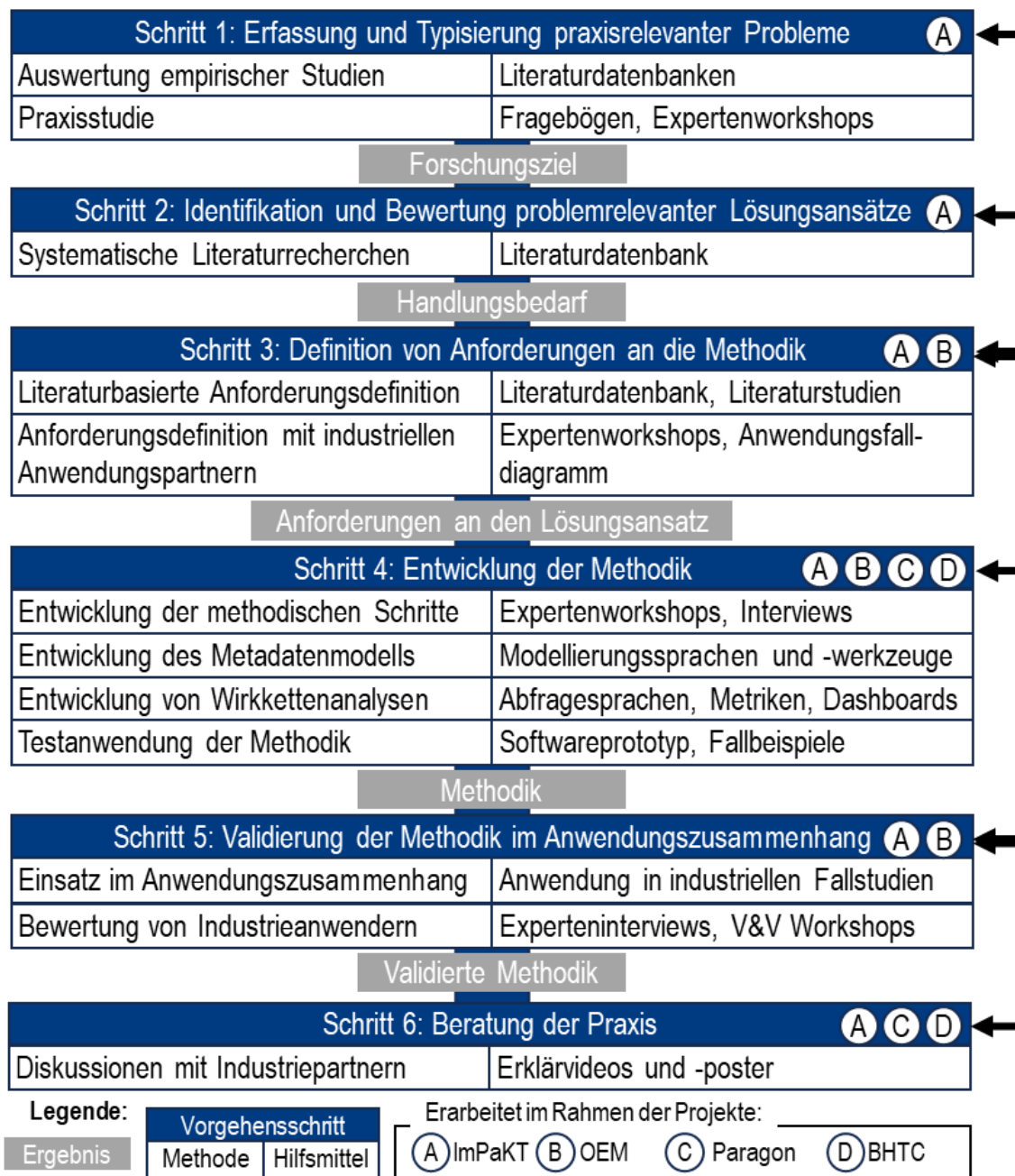


Bild 1-3: Wissenschaftliches Vorgehen des Promotionsvorhabens nach Ulrich [Ulr81]

Zu Beginn wird in Schritt 1 die **Erfassung und Typisierung praxisrelevanter Probleme** durchgeführt. Hierzu werden in einem zweiteiligen Vorgehen Herausforderungen von Auswirkungsanalysen im Engineering Change Management identifiziert. Zunächst werden Literaturstudien und literaturbasierte Anwendungsfälle identifiziert und beschriebene Probleme erfasst und typisiert. Anschließend werden Umfragen mit Industrieanwendern durchgeführt, in denen die Probleme aus der Praxisperspektive reflektiert und bestätigt werden.

Anschließend wird in Schritt 2 die **Identifikation und Bewertung problemrelevanter Lösungsansätze** durchgeführt. Hierzu werden Literaturrecherchen im Bereich Engineering Change Management in Hinblick auf Wirkkettenanalysen und im Bereich MBSE in Hinblick auf Ansätze zur durchgängigen Modellierung durchgeführt. Die identifizierten Lösungsansätze werden anhand der praxisrelevanten Probleme analysiert, um Herausforderungen und Limitationen aktueller Ansätze zu analysieren. Durch die Analyse der identifizierten Ansätze werden Kriterien zur Bewertung der Ansätze definiert und Handlungsbedarfe abgeleitet. Abschließend werden die betrachteten Ansätze und der Lösungsansatz dieser Arbeit anhand von hergeleiteten Bewertungskriterien verglichen und abgegrenzt.

Aufbauend darauf wird in Schritt 3 die **Erfassung und Definition von Anforderungen an die Methodik** durchgeführt. Anhand der Handlungsbedarfe werden literaturbasiert problemrelevante Anforderungen an die Methodik erhoben. Zur Sicherstellung der Praxistauglichkeit wurden die ermittelten Anforderungen in Workshops mit Industrieanwendern diskutiert und bestätigt.

Ausgehend von den Anforderungen wird in Schritt 4 die **Methodik** zur modellbasierten Wirkkettenanalyse entwickelt. Hierzu werden in einer Konzeption grundlegende Modelle entwickelt, die die Entwicklung der ganzheitlichen Methodik strukturieren. Aufbauend auf den Modellen werden eine Methode und Hilfsmittel zur Analyse und Bewertung der Auswirkungen entwickelt. Die so entstehende Methodik wird softwaregestützt implementiert und am Beispiel eines Roboterarms demonstriert.

In Schritt 5 wird die **Validierung der Methodik im Anwendungszusammenhang** beschrieben. Die Validierung wird unterteilt in den Einsatz der Methodik im industriellen Anwendungszusammenhang im Rahmen von vier Fallstudien, die Bewertung der Anforderungen und die Bewertung des Erfolgs der Methodik durch Industrieanwender aus einem Verbundforschungsprojekt. Die vier industriellen Fallstudien zur Anwendung der Methodik sind: ein Raupenlaufwerk für Landmaschinen, eine Auswuchtmaschine, ein Spritzgussartikel eines Ladesteckers für elektrische Kraftfahrzeuge und ein Fensterheber eines Kraftfahrzeugs. Zur Validierung der Anforderungen und zur Demonstration der Softwarefunktionalitäten werden Experimente mit Industrieanwendern durchgeführt, in denen die Methodik anhand von Praxisbeispielen angewendet wird und die Anforderungserfüllung und der Methodik-Erfolg in Fragebögen bewertet wird.

Abschließend wird in Schritt 6 die validierte Methodik für die **Beratung der Praxis** eingesetzt. Hierzu werden Erklärvideos und -poster (siehe A4.10, S. 195) erstellt, die Anwender zur eigenständigen Entwicklung modellbasierter Wirkkettenanalysen befähigen.

2 Grundlagen und Stand der Forschung

Im Folgenden werden Grundlagen und Begrifflichkeiten innerhalb des Betrachtungsbereichs beschrieben (Kapitel 2.1). Anschließend werden problemrelevante Lösungsansätze aus den Themenfeldern *Auswirkungsanalysen im Engineering Change Management* und *durchgängige Modellierungsansätze* identifiziert und diskutiert (Kapitel 2.2), um abschließend die Forschungslücke und Handlungsbedarfe herzuleiten (Kapitel 2.3).

2.1 Grundlagen

Aufbauend auf einer grundlegenden Einordnung des Dissertationsthemas in den Produktlebenszyklus (Kapitel 2.1.1) werden Vorgehensweisen des Änderungsmanagements (Kapitel 2.1.2) beschrieben. Das Änderungsmanagement erfolgt im Rahmen übergeordneter Entwicklungsmethodik - dem Systems Engineering (Kapitel 2.1.3) und dem Model-based Systems Engineering (Kapitel 2.1.4). Ansätze der Wirkkettenmodellierung (Kapitel 2.1.5) und Graphentheorie (Kapitel 2.1.6) umfassen weitere Grundlagen für Wirkkettenanalysen.

2.1.1 Technische Änderungen in dem Produktlebenszyklus

Das Thema der Dissertation lässt sich in die Lebenszyklusphase „Entwicklung“ einordnen (siehe Bild 2-1). Der Bereich „Entwicklung“ beinhaltet die theoretischen Grundlagen des Systems Engineerings und des Änderungsmanagements. Der Lösungsansatz beginnt bei der Herleitung von Anforderungen während der Entwicklung und endet bei der Übergabe des entwickelten Systems an die Realisierung. Informationen, die während der Realisierung und der Betriebs- und Leistungserbringung entstehen, können über Informationskreisläufe für die Bewertung der Änderungsauswirkungen verwendet werden.

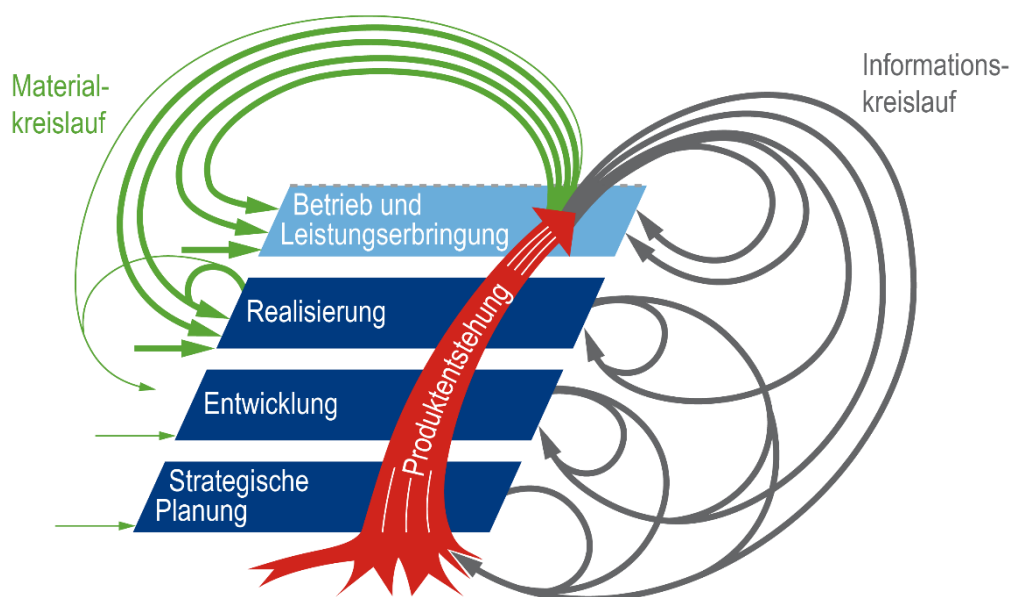


Bild 2-1: Generischer Produktlebenszyklus nach Gräßler [GP21, S. 13]

2.1.2 Änderungsmanagement

(...) without an adequate change management only two alternatives exist: to die of changes, or to miss the chance of a successful product.
[FGN00, S. 170]

WICKEL [Wic17, S. 18] definiert eine **technische Änderung** (*Engineering Change – EC*) als Modifikationen an Produkten, Produktmodellen sowie zugehörigen Bestandteilen, unter Anwendung eines definierten Änderungsprozesses, nachdem diese für die weitere Entwicklung und Produktion freigegeben sind. Die Definition nach JARRETT ET AL. [JEC11, S. 103] hebt die zu beachtenden Wechselwirkungen hervor: Eine Änderung kann jede Veränderung der Form, der Passform und/oder der Funktion des Produkts (...) umfassen und kann die Wechselwirkungen und Abhängigkeiten der Bestandteile des Produkts verändern. Folgende Begriffe werden im Anwendungskontext unterschieden: ECs werden durch Änderungsimpulse (*Trigger*) ausgelöst [JEC11, S. 107] und durch eine formale **Änderungsanfrage** (*Engineering Change Request - ECR*) dokumentiert. Auf Basis von Entscheidungsalternativen wird eine Entscheidung über die Ablehnung oder Genehmigung einer Änderung getroffen. [JEC11, S. 108ff.]. Im Falle einer Genehmigung wird ein **Änderungsantrag** (*Engineering Change Order - ECO*) erstellt, durch den die Durchführung einer technischen Änderung veranlasst wird [VDA09a, S. 5].

Technische Änderungen und die auslösenden Änderungsimpulse sind heterogen, da sie unterschiedliche Größen und Arten haben, unterschiedlich viele Rollen im Projekt involvieren und die Umsetzung unterschiedlich lang dauern kann [SMS12, S. 291, JEC11, 106]. Änderungen werden anhand der Auswirkungen auf das System in **Änderungskategorien** unterteilt, beispielsweise Auswirkungen auf Kosten, Zeitplan, wichtige Schnittstellen, Sicherheit, Leistung und Zuverlässigkeit. [KSF20, S. 480] Aufgrund der Heterogenität technischer Änderungen werden auch eine Vielzahl von Entwicklungsdaten benötigt, um die Auswirkungen technischer Änderungen zu bewerten. Zwei Charakteristika zur Unterscheidung von technischen Änderungen sind die Ursache und Herkunft der Änderung [ECZ04, S. 20]. **Ursachen** für technische Änderungen sind beispielsweise Änderungen der Kundenspezifikation, Fehler bei der Umsetzung von Kundenwünschen, festgestellte Schwächen bei der Prüfung eines Prototyps, Qualitätsprobleme bei einem Systemelement oder Weiterentwicklungen für künftige Produktrevisionen [PM98, S. 2]. Die **Herkunft** technischer Änderungen wird unterschieden in Änderungen, die von dem Produkt ausgehen und während der Produktentwicklung entstehen und Änderungen, die von externen Stakeholdern außerhalb des Unternehmens gefordert werden [UTY16, S. 28]. Eine effiziente und effektive Handhabung technischer Änderungen kann daher nur sichergestellt werden, wenn Entwickler befähigt werden, Auswirkungen heterogener technischer Änderungen präzise bewerten zu können [PM98, S. 2].

Für den Umgang mit technischen Änderungen werden fünf **Strategien** unterschieden: Vermeidung, Vorverlagerung, effektiver Umgang, effizienter Umgang und Lernen. Die Strategie *Vermeidung* zielt darauf ab, technische Änderungen abzuwenden und die Anzahl aufkommender Änderungen systematisch zu reduzieren. Die Strategie *Vorverlagerung* umfasst, Änderungen frühzeitig zu erkennen und zu einem frühen Zeitpunkt umzusetzen. Die Steigerung der *Effektivität* ist durch eine spezifischere Bewertung von technischen Änderungen möglich. *Effizienz* umfasst die zielgerichtete Nutzung vorhandener Ressourcen zur Reduzierung der entstehenden kapazitiven und monetären Aufwände. Beim *Lernen* liegt der Fokus auf der Erhöhung der Effektivität und Effizienz durch das Lernen aus dem Umgang mit vergangenen Änderungen. [FGN00, S. 173]

Im **technischen Änderungsmanagement** (*Engineering Change Management – ECM*) werden diese Strategien umgesetzt. Das ECM umfasst die Organisation, Kontrolle und Ausführung von ECs [JEC11, S. 105] entlang des gesamten Produktlebenszyklus [HCC13, S. 476]. In der Praxis stehen Entwickler trotz der ECM Ansätze vor der Herausforderung, nicht mehr selbst abschätzen zu können, welche Auswirkungen technische Änderungen aufeinander haben [FES03, S. 1]. Eine initiale technische Änderung kann weitere unerwartete Änderungen auslösen, sodass sich die Auswirkungen auf große Teile des betrachteten Systems ausweiten [CSE04, S. 788]. Technische Änderungen bilden durch die Ursache-Wirkungs-Beziehungen zwischen den Elementen des Systems ein Netzwerk, welches von FRICKE ET AL. [FGN00, S. 170] als das Ursache-Wirkungs-Netz von technischen Änderungen bezeichnet wird. Ein Änderungsimpuls kann zu einer initialen Änderung führen. Die initiale Änderung kann sich fortpflanzen und einen Impuls für weitere Änderungen darstellen. Die eingeleitete Änderung ist demnach die Hauptursache für die Ausbreitung auf weitere Elemente, wodurch die Wirkung der Änderung zur Ursache weiterer Ausbreitungen wird [KJ16, S. 5]. Dieser Effekt wird **Änderungspropagation** oder Änderungsfortpflanzung genannt. Gerade durch Änderungspropagation können sich Änderungsdurchlaufzeiten erhöhen und unvorhersehbare Kosten entstehen [UTY15]. Das Verhalten der Änderungspropagation kann typisiert werden. ECKERT ET AL. [ECZ04, S. 13] unterscheiden Systemelemente nach ihrem Absorptions- oder Propagationsverhalten in fünf Arten:

- *Konstanten* sind Elemente, die weder Änderungen absorbieren noch hervorrufen.
- *Absorber* sind Elemente, die mehr Änderungen absorbieren als auslösen.
- *Träger* absorbieren eine ebenso große Anzahl von Änderungen, wie sie hervorrufen.
- *Multiplikatoren* generieren mehr Änderungsbedarfe als sie hervorrufen.
- *Puffer* können einen Teil an Änderungen absorbieren, aber auch weitere hervorrufen.

Die Arten des Propagationsverhaltens werden im linken Teil von Bild 2-2 in die zwei Achsen des Grads der Absorption und des Grads der Propagation eingeordnet. Je nach Änderung können durch das Propagationsverhalten der Systemelemente unterschiedlich viele Änderungen über die Zeit resultieren, wodurch unterschiedliche Änderungsmuster

entstehen [ECZ04, S. 17f.]. ECKERT ET AL. bezeichnen das Muster, wenn mehrere Systemelemente aus nachvollziehbaren Gründen betroffen sind und geändert werden, als kontrollierbare *Änderungswellen*. Bei *Änderungsblüten* steigt die Anzahl der resultierenden Änderungen zunächst stark an, flacht aber nach einer absehbaren Zeit wieder ab. Bei *Änderungslawinen* nimmt die Anzahl resultierender Änderungen stetig zu und die planmäßige und zeitnahe Umsetzung ist gefährdet. Die Muster werden rechts in Bild 2-2 dargestellt. [ECZ04, S. 17f.]

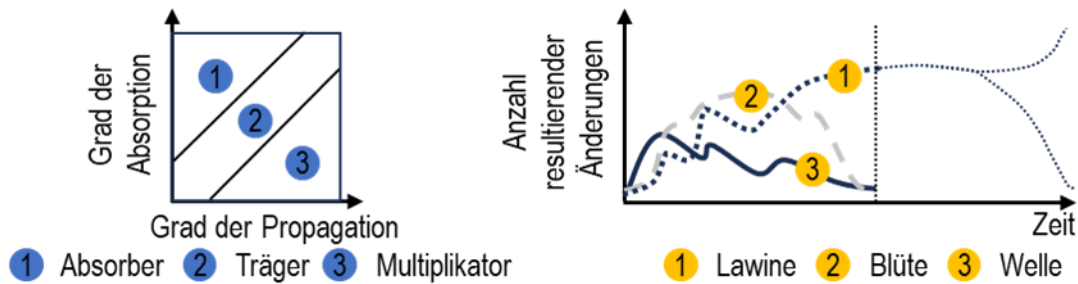


Bild 2-2: Absorptions-/Propagationsverhalten [ECZ04, S. 13] und resultierende Muster [ECZ04, S. 18]

Seit den 1990er-Jahren beschäftigten sich Forscher mit der Modellierung und Analyse von Änderungspropagation und zugehörigen Propagationspfaden [BW23, S. 117]. Bei der Analyse von Propagationspfaden technischer Änderungen werden zwei fundamentale Konzepte unterschieden [BW23, S. 123]: (1) die Begrenzung der Propagation auf eine diskrete Anzahl von betroffenen Elementen unterschiedlicher Art sowie (2) die Ausbreitung der Änderung entlang dieser Elemente. Aufbauend auf einer umfassenden Literaturanalyse erarbeiteten die Autoren BRAHAM und WYNN **Bereiche der Änderungspropagation**, die in der Produktentwicklung betroffen sein können [BW23, S. 123f.]. Hierzu zählen unter anderem Anforderungen, Funktionen, Subsysteme und Systemkomponenten sowie zugehörige Parameter und Geometrien. Die Art und Weise wie zwei Elemente verknüpft sind, hat Einfluss auf die Bewertung der Auswirkung in Hinblick auf den Propagationseffekt. Im Bereich der Produktentwicklung können solche Propagationspfade anhand der Daten und Informationen aus Entwicklungsartefakten und zugehörigen Relationen abgebildet werden. Informationssysteme bilden hierbei die Grundlage für einen effizienten und effektiven Umgang [KJ16, S. 8].

Werkzeuge zur Bewertung der Ausbreitung und der Auswirkungen von Änderungen dienen dazu, Abhängigkeiten in einem System zu ermitteln und Auswirkungen systematisch entlang der Abhängigkeiten zu bewerten [IRA20, S. 4509]. Funktionalitäten zur Unterstützung von Auswirkungsanalysen können in dedizierte Werkzeuge umgesetzt oder in existierende Werkzeuge integriert werden [HM99, S. 32]. Zusätzlich müssen für die Analyse Daten und Modelle aus IT-Systeme vernetzt werden, beispielsweise aus CAD (Computer Aided Design) Systemen und PLM (Product Lifecycle Management) Systemen [HST23, S. 118]. Die Werkzeuge unterstützen Entwickler bei der Durchführung von **Engineering Change Auswirkungsanalysen** (EC-Auswirkungsanalysen).

„Die EC-Auswirkungsanalyse für das technische Änderungsmanagement ist die Aktivität der Identifikation potenzieller Effekte einer Änderung – einschließlich initialer und konsekutiver Effekte – und die Abschätzung der resultierenden (lokalen und kollektiven) Folgen, mit dem Ziel, eine effektive Entscheidung über Änderungsbedarfe zu unterstützen.“ [GW21, S. 3]

2.1.3 Systems Engineering

Etablierte Konstruktionssystematiken, wie beispielsweise PAHL und BEITZ [PBF07] oder die Richtlinie 2221 [Ver19] des Vereins Deutscher Ingenieure (VDI) strukturieren die Entwicklungsaufgabe systematisch ausgehend von der Mechanik und dem jeweils verfügbaren Bauraum. Die unterschiedlichen regionalen Schulen der Konstruktionssystematik wurden erstmals 1993 in der VDI-Richtlinie 2221 [Ver93] harmonisiert und inhaltlich zusammengeführt. 2019 wurde die VDI-Richtlinie 2221 aktualisiert, um der Herausforderung zunehmender Volatilität gerecht zu werden [Ver19]. **Systems Engineering Ansätze** unterstützen Entwickler aus interdisziplinären Fachbereichen bei der Handhabung der steigenden Komplexität, indem Produkte als Systeme betrachtet werden. Die Komplexität umfasst die Systemkomplexität und die Lebenszykluskomplexität [GO22b, 5]. Zur Umsetzung des Entwicklungsprojekts wird im Systems Engineering eine kontinuierliche Interaktion mit dem Projektmanagement vorgesehen [INC15, S104, HKS21, S. 250]. Ein **System** ist „(...) eine eindeutig abgegrenzte Einheit und besteht aus verschiedenen Subsystemen und Systemelementen (Hardware, Software, Dienstleistungen, Personen etc.) sowie deren Verknüpfungen“ [VDI21, S. 11]. Diese Verknüpfungen umfassen unter anderem die aus der Konstruktionslehre bekannten Energie-, Stoff-, und Signalflüsse, können aber auch andere logische Wirkzusammenhänge darstellen. Jedes physische Systemelement kann Eigenschaften und Funktionen besitzen oder erfüllen [HWF19, S. 4]. Die Systemelemente und Relationen formen ein Konstrukt, die sogenannte Systemstruktur [HWF19, S. 5]. Das zu entwickelnde System, dessen Lebenszyklus betrachtet wird, wird im Systems Engineering als System of Interest (SoI) bezeichnet. [INC15, S. 265]. Diese muss von dem Systemumfeld abgegrenzt werden, um den Betrachtungsbereich des Systems festzulegen. Daher besitzt das System „(...) eine **Systemgrenze**, die durch den Entwickler unter Berücksichtigung der jeweiligen Perspektive zu definieren ist [VDI21, S. 11].“ Eine beispielhafte abstrakte Abgrenzung eines Systems ist in Bild 2-3 dargestellt. Ein übertragbares Praxisbeispiel aus einem zugrunde liegenden bilateralen Forschungsprojekt ist die Betrachtung eines Kraftfahrzeugs im Straßenverkehr. Das Kraftfahrzeug ist das SoI und besteht selbst aus einzelnen Elementen und Relationen wie dem Dachantrieb, einer Batterie, einem Ladestecker und Sensoren. Das SoI interagiert aber auch mit Systemen und Elementen im Umfeld, beispielsweise mit einem Ladesäulensystem oder einem Straßenschild.

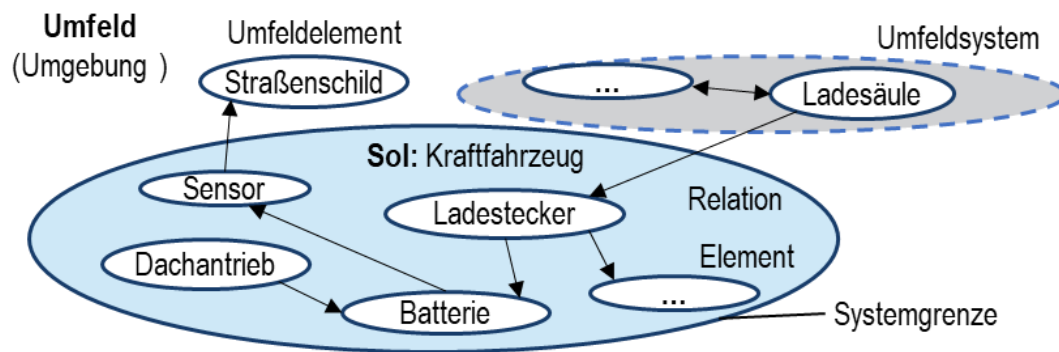


Bild 2-3: Systemgrenze des System of Interest [HWF19, S. 5]

In Systems Engineering Ansätzen wird die Entwicklung solcher technischer Systeme systematisiert. Systems Engineering lässt sich daher wie folgt definieren:

*„**Systems Engineering** beruht auf systemischem Denken und umfasst eine Methodik für die Entwicklung komplexer technischer Systeme zur Erzielung eines disziplinübergreifenden Optimums innerhalb eines zuvor festgelegten Zeit- und Kostenrahmens. Hierfür werden die Disziplinen modelltechnisch strukturiert und miteinander vernetzt.“*
 ([GO22b, S. 16] aufbauend auf [Grä15, S. 5])

Systems Engineering (SE) besteht aus drei Kernelementen: dem Systemdenken, der Entwicklungsmethodik und dem Systems Engineer [GO22b, S. 16f.] (siehe Bild 2-4).



Bild 2-4: SE Kernelemente Systemdenken, Entwicklungsmethodik und Systems Engineer

Das **Systemdenken** beschreibt eine ganzheitliche Denkweise, welche bei der Identifikation von komplexen systemischen Ursache-Wirkungs-Zusammenhängen und der zielgerichteten Lösungsfindung unterstützt [GO22b, S. 16].

Systemdenken ist ein ganzheitlicher Analyseansatz, der sich auf die Art und Weise konzentriert, wie die Bestandteile eines Systems miteinander in Beziehung stehen und wie Systeme im Laufe der Zeit und im Kontext größerer Systeme funktionieren. [Eig21, S. 67].

Ausgehend von dem Verständnis der Definition leiten ARNOLD und WADE vier Bereiche des Systemdenkens ab: die Denkweise („Wie geht man an das systematische Problem heran“), den Inhalt („Was ist im System“), die Struktur („Wie ist das System organisiert“) und das Verhalten („Was passiert, wenn Inhalte der Struktur miteinander interagieren“)

[AW17, S. 11]. Das Systemdenken kann in drei Dimensionen unterteilt werden: „System abgrenzen“, „Zusammenhänge und Wechselwirkungen erkennen“ und „Systemsichten und Perspektiven einnehmen“ [GO22d, S. 94]. In der ersten Denkdimension wird das SoI wie oben gezeigt von den Umfeldsystemen und Umfeldeinflüssen abgegrenzt. Die Systemgrenze beschreibt daher, was zum System gehört und was nicht [INC15, S. 6]. Die Anwendung des Systemdenkens unterstützt Entwickler in der zweiten Denkdimension bei der ganzheitlichen Betrachtung von Ursache-Wirkungs-Beziehungen und der zugehörigen Abgrenzung, Strukturierung und Dekomposition von Systemen [HWF19, S. 100]. Durch das systematische Erkennen von Zusammenhängen und Wechselwirkungen wird die Grundlage geschaffen, um relevante Ursache-Wirkungs-Beziehungen frühzeitig zu berücksichtigen. In der dritten Dimension werden die Struktur und das inhärente Verhalten aus unterschiedlichen Perspektiven und Systemsichten betrachtet [GO22c, 102]. Diese Perspektiven und Sichtweisen unterstützen die Anwender bei der Handhabung der Komplexität sowie bei der Fokussierung auf eigene SE-Aktivitäten. Systemsichten können miteinander kombiniert werden. Beispielsweise kann eine abstrakte Black-Box Sichtweise durch eine detaillierende White-Box Darstellung ergänzt werden [HWF19, S. 16]. In Bild 2-5 wird illustriert, dass die Festlegung von Systemgrenzen bei der Einschränkung betrachteter Elemente und Wirkzusammenhänge aus der Perspektive eines Anwenders unterstützt. Beispielsweise liegen für den Entwickler des Dachantriebs die internen Wirkzusammenhänge des Dachantriebs sowie Wechselwirkungen zum Kraftfahrzeug innerhalb des Betrachtungsbereichs. Wechselwirkungen zur Ladesäule oder zum Straßenschild können vernachlässigt werden.

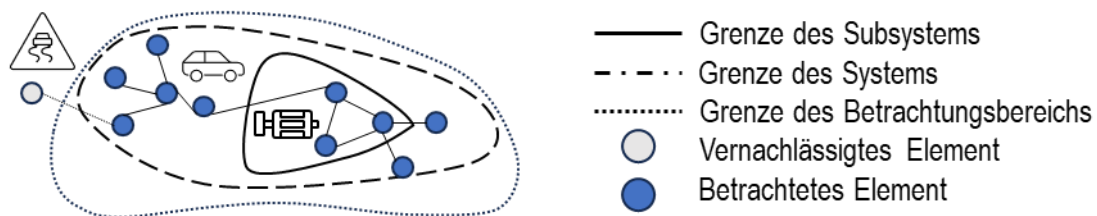


Bild 2-5: Systemgrenzen und Betrachtungsbereiche in Anlehnung an [HWF19, S. 178]

Durch eine **Entwicklungsmethodik** werden bedarfsgerecht Modelle, Methoden und Hilfsmittel bereitgestellt, welche Entwickler bei der Lösung der Aufgabenstellung unterstützen [GO22b, S. 16]. Der Begriff Methodik beschreibt ein System von Modellen, Methoden und Hilfsmitteln zur Lösung einer Aufgabenstellung [Lau96, Boc96, Hey99, Grä00]. Zur Anwendung von SE werden Normen und Richtlinien wie die VDI/VDE Richtlinie 2206 [VDI21] und ISO/IEC 1588:2015 [ISO11b] genutzt, aber auch SE Ansätze wie das INCOSE Handbook [INC15], NASA Systems Engineering Handbook [NAS07], Systems Engineering nach HABERFELLNER [HWF19] oder auch System Lifecycle Management nach EIGNER [Eig21]. Die Entwicklungsmethodiken enthalten unterschiedliche Modelle, mit welchen das Vorgehen in der Entwicklung visualisiert werden kann. Ein Beispiel dafür ist das V-Modell, welches im SE weit verbreitet ist und SE-Aktivitäten in einer V-Form anordnet. [INC15, S. 33]

Die Richtlinie VDI/VDE 2206:2021 [VDI21] beschreibt eine Entwicklungsmethodik für die interdisziplinäre Entwicklung mechatronischer und cyber-physischer Systeme. Innerhalb des V-Modells werden die sachlogischen Verknüpfungen von Aufgaben der interdisziplinären Systementwicklung visualisiert [VDI21, S. 20f.]. Das V-Modell besteht dabei aus drei Strängen, die während der Entwicklung parallel durchlaufen werden [VDI21, S. 21]. Die drei Stränge werden durch horizontale Kontrollpunkte untergliedert, welche Kontrollfragen umfassen und Entwicklern bei der Strukturierung und Planung der Entwicklungsaufgaben helfen. Im Vergleich zu Meilensteinen sind die Kontrollpunkte zeitlich unabhängig und nicht relevant für die Vergabe von Freigaben [VDI21, S. 24].

Zur Anwendung von SE müssen die Entwickler unterschiedliche Rollen im Unternehmen einnehmen. Alle notwendigen Verantwortlichkeiten und Rollen werden durch den **Systems Engineer** repräsentiert [GO22b, S. 17]. Zur Definition der Rollen und Verantwortlichkeiten werden Rollenmodelle definiert, die die Rollen übersichtlich und prägnant zusammenfassen. SHEARD [She96, S. 479] definiert zwölf Systems Engineering Rollen. Ausgehend von dem Rollenmodell nach SHEARD und Erkenntnissen aus der praktischen Anwendung entwickeln GRÄBLER ET AL. ein Modell mit 15 Rollen. [GOH19, S. 1272], in dem unter anderem der System Architect und Implementation Engineer definiert werden.

Zentraler Bezugspunkt des Systems Engineers ist die **Systemarchitektur**. In dem Standard ISO/IEC/IEEE 40210 [ISO11b] werden zentrale Grundkonzepte einer Systemarchitektur und einer Architekturbeschreibung definiert. Eine Architekturbeschreibung stellt eine Konzeptualisierung der Systemarchitektur dar und trägt zum Verständnis des Wesens des Systems und der Schlüsseleigenschaften bei. Architekturbeschreibungen werden für unterschiedliche Zwecke genutzt, unter anderem als Basis für die Systementwicklung, zur Kommunikation zwischen Stakeholdern, zum Nachweis bei Zertifizierungen oder als Bewertungsgrundlage für Analysen und Reviews des Systems entlang der Lebenszyklusphasen. Eine Systemarchitektur umfasst Konzepte und Eigenschaften eines Systems, die in seinen Elementen, Beziehungen und in den Grundsätzen seiner Gestaltung und Entwicklung zum Ausdruck kommen. [ISO11b, 5 ff.] Die Konzepte einer Systemarchitektur werden in Bild 2-6 veranschaulicht.

2.1.4 Model-based Systems Engineering

„Gute Modellbildung bedeutet: Das Richtige weglassen.“ - VAJNA ET AL. [VWZ18, S. 188]

Das dokumentenzentrierte Systems Engineering ist von silo-orientierten Datenablagen und dateibasierten Datenaustauschen mit gängigen Austauschformaten geprägt [Eig21, S. 69]. Die **Modellbildung** ist ein Schlüsselfaktor des Systems Engineerings, um fundierte Entscheidungen entlang des Systemlebenszyklus auf Basis einer modellbasierten Systemdarstellung zu treffen [PDH11, S. 14]. Modellbildung umfasst die Erstellung und Analyse von disziplinspezifischen und disziplinübergreifenden Modellen entlang des Produktlebenszyklus [GO22d, S. 35]. Die Nutzung von Modellen unterstützt Entwickler dabei, implizites Expertenwissen zu explizieren und deklarativ in Modellen abzubilden, wodurch silo-orientierte Datenablagen aufgelöst werden. Das deklarative Wissen umfasst dabei die Beschreibung von Objekten und Beziehungen zwischen diesen [VWZ18, S. 419]. Ein Anwendungsfall ist die Identifikation, Modellierung und Analyse von Systemelementen und derer Wechselwirkungen untereinander. Bei der Modellbildung werden die Struktur und das Verhalten von Systemen, Subsystemen und Systemelementen durch Modelle beschrieben [VDI21, S. 35]. Ausgehend von den deklarativen Modellen kann der Entwickler durch sein prozedurales Wissen die Objekte innerhalb eines problemspezifischen Kontexts in Beziehung setzen [VWZ18, S. 419]. Die Zusammenführung von Modellbildung und SE wird als Model-based Systems Engineering bezeichnet.

*„**Model-based Systems Engineering (MBSE)** ist die formalisierte Anwendung der Modellierung zur Unterstützung von Anforderungserhebung, Entwurf, Analyse, Verifikation und Validierung. MBSE beginnt in der Konzeptionsphase und erstreckt sich über die gesamte Entwicklung sowie spätere Lebenszyklusphasen“ (übersetzt aus dem Englischen aus [INC, S. 15]).*

DELLIGATTI [Del14, S. 3] ergänzt, dass das primäre Artefakt von MBSE-Aktivitäten ein integriertes, kohärentes und konsistentes **Systemmodell** ist, das mithilfe eines Systemmodellierungswerkzeugs erstellt wird. Ein Systemmodell wird mit dem Ziel erstellt, ein System als Ganzes abzubilden, wohingegen spezifische Modelle einen spezifischen Aspekt des Systems abbilden [HBF19, S. 11f.]. HICK ET AL. [HBF19, S. 14] visualisieren die Anordnung von Systemmodellen und spezifischen Modellen in einem dreidimensionalen Würfel. Die Breite des Würfels beschreibt die an der Entwicklung beteiligten Disziplinen. Die Weite des Würfels beschreibt die an der Entwicklung beteiligten technischen Bereiche (z. B. Anforderungen, Struktur, Verhalten, Verifizierung und Validierung usw.). Die Tiefe des Würfels beschreibt den Detaillierungsgrad. [HBF19, S. 7] Die Anordnung verdeutlicht, dass Systemmodelle das Gesamtsystem in der Breite und Weite abdecken, wohingegen spezifische Modelle einen Aspekt des Modells in der Tiefe abbilden (siehe Bild 2-7). Demnach ist es wichtig, dass Systemmodelle und Systemansichten passend für ein Entwicklungsziel angepasst werden [HBF19, S. 12] Durch die

Kombination von Sichten und Perspektiven wird der Blickwinkel bewusst gesteuert und an die Aufgabe angepasst. [GO22d, S. 102]. Im Vergleich zur modellbasierten Entwicklung von Systemen steht bei der Bewertung eines Systems die Analyse im Vordergrund, beispielsweise die Analyse und Bewertung von Auswirkungen technischer Änderungen.

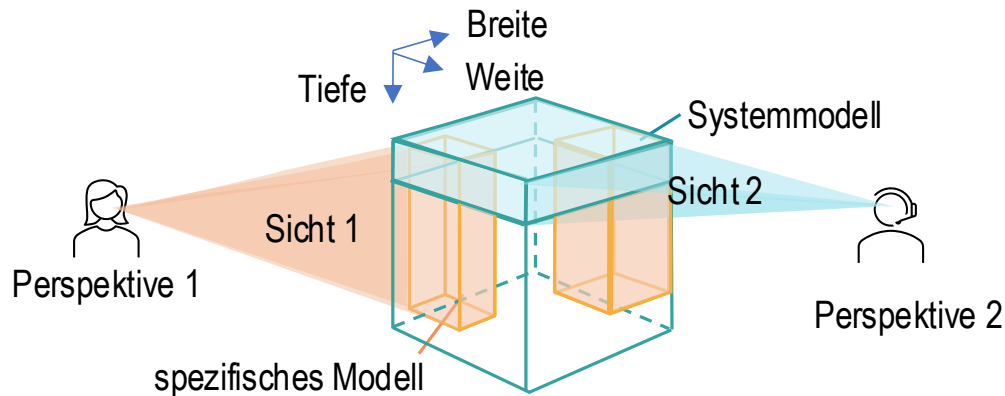


Bild 2-7: SysML-Systemmodell und spezifische Modelle (angelehnt an [GO22a, S. 55])

Ein Systemmodell wird mithilfe einer Modellierungssprache, eines Modellierungswerkzeugs und einer Modellierungsmethode entwickelt [Del14, S. 4]. Durch die **Modellierungssprachen** wird die verwendbare Semantik und Syntaktik zur Erstellung des MBSE-Systemmodells vorgegeben. Die Semantik beschreibt die inhaltliche Bedeutung. Die Syntaktik oder Syntax befasst sich mit der Beziehung zwischen den Zeichen eines Sprachsystems, also den durch Konvention festgelegten Regeln einer Sprache, nach denen einzelne Zeichen zu gültigen Ausdrücken und Sätzen kombiniert werden können. [Krc15, S. 16] Die meistverbreitete Modellierungssprache ist die Systems Modeling Language (SysML) [OMG19], welche über die Object Management Group spezifiziert wurde. SysML ist eine Profilerweiterung der Unified Modeling Language [Obj17]. Profile sind der Standard UML Mechanismus zur spezifischen Erweiterung der Modellierungssprache [Fri14, S. 351]. Modellierungssprachen werden über **Modellierungswerkzeuge** für Anwender nutzbar gemacht. Modellierungswerkzeuge für die Anwendung von SysML sind unter anderem Catia Magic von D'assault Systèmes, Rhapsody von IBM und Enterprise Architect von Sparxx Systems. Eine alternatives MBSE-Modellierungswerkzeug ist beispielsweise Eclipse Capella. **Modellierungsmethoden** unterstützen Anwender, getrennte Modelle zu einem konsistenten Ganzen zusammenzuführen [MMR09]. Die Modellierungsmethoden beruhen auf obligatorischen Schritten. Hierzu zählt das Identifizieren des Modellzwecks, das Identifizieren der Eingangs- und Ausgangsgrößen, die Wahl der Modellart und das Repräsentieren der Relationen und Systemelemente [PDH11, S. 102]. Die wichtigste Überlegung bei der Modellierung ist die Festlegung des Zwecks des Modells [PDH11, S. 103]. Abhängig vom Zweck werden die Systemmodelle auf die betrieblichen Anforderungen zugeschnitten [SZS21]. Durch eine vorgelagerte Problembeschreibungsphase werden Stakeholder unterstützt, erst die Probleme zu explizieren, bevor voreilig Lösungen entwickelt werden [PDH11, S. 292].

Die **Rückverfolgbarkeit** (eng. Traceability) zwischen Entwicklungsartefakten ist ein Vorteil von Systemmodellen. Traceability beschreibt den Grad einer Relation zwischen zwei oder mehreren Produkten aus einem Produktentstehungsprozess [IEE, S. 78]. Aus interdisziplinärer Entwicklungsperspektive entsteht durch Traceability die Fähigkeit zur Rückverfolgung von Entwicklungsartefakten und deren Beziehungen entlang der Lebenszyklusphasen. Durch die rechner-interpretierbare und explizite Modellierung solcher Entwicklungsartefakte lässt sich die Rückverfolgbarkeit in sogenannten Traceability Information Modellen (TIM) abbilden. REMPEL und MÄDER unterscheiden drei Elemente von TIMs: Trace-Relation-Klassen (*trace link class*), Artefaktklassen (*artifact class*) und Trace-Pfad-Klassen (*trace path class*) [RM15, S. 177]. Artefaktklassen repräsentieren Modell-elemente, die als eigene Entität mit einer bestimmten Anzahl von Eigenschaften innerhalb der digitalen Entwicklungsumgebung betrachtet werden (in Anlehnung an [Tor22, S. 34]). Innerhalb spezifischer Metadatenmodelle wird die Semantik festgelegt, in welcher das TIM definiert wird [RJ01, S. 66] Ein Beispiel ist die Definition der Artefakt-klassen Anforderung und Verifikation, die durch die Relationsklasse verify verknüpft werden. Durch die Kombination von mehr als zwei Artefakten und Relationen entsteht ein rückverfolgbarer Pfad (*Trace-Pfad*). Durch die Modellierung von rückverfolgbaren Pfaden ergibt sich der **Digital Thread**. Der Digital Thread (oder auch digitaler roter Faden [BBW17, S. 151]) umfasst die Gesamtheit der Artefakte eines Systems und ihrer Beziehungen, die durch Trace-Links im Laufe des Produktlebenszyklus erfasst werden [Tor22, S. 34]. Im Digital Thread werden Informationen aus technischen Modellen aus mehreren Ablageorten und Partialmodellen zusammengeführt [BBW17, S. 151]. Ein Beispiel dafür ist die Integration von Informationen aus CAD-Modellen und Systemmodellen [HST23, S. 118]. Die Verbindung von Modellen ist für die Realisierung des digital Threads unerlässlich [BFK21, S. 1044]. Im Änderungsmanagement können durch den Digital Thread direkt oder indirekt betroffene Elemente identifiziert werden [Eig21, S. 61]. Exemplarische Best Practices bei der Erstellung von Digital Threads sind nach TORIELLO [Tor22, S. 37]: (1) Klare und konsistente Richtung von Links über alle Artefakte hinweg, (2) Unidirektionale Links je Relationen, um reflexive Schleifen zu vermeiden, (3) Erstellung der Links auf gleicher, möglichst tiefer Detaillierungsebene und (4) Vernetzen der Artefakte auf gleicher Abstraktionsebene.

2.1.5 Wirkkettenmodellierung

Neben Modellierungsansätzen, die die Systemmodellierung unterstützen, gibt es Modellierungsansätze, die den Fokus auf die Systemanalyse legen. Ansätze der **Wirkkettenmodellierung** werden angewandt, um Ursache-Wirkungs-Beziehungen zu modellieren und die betroffenen Elemente und Abhängigkeiten zu analysieren [GWK22, S. 1885] (siehe Bild 2-8). Beispielsweise müssen Auswirkungen einer Anforderungsänderung auf physische Systemelemente und Verifikationselemente abgeschätzt werden [INC15, S. 196].

Wirkkettenmodelle haben folgende Gemeinsamkeiten [GWK22, S. 1888]:

- Es wird bewusst ein separater Bereich des Systemmodells betrachtet.
- Die Länge einer Wirkkette hängt von dem abgebildeten Detaillierungsgrad ab.
- Um eine Wirkkette zu modellieren, muss die Menge der Systemelemente und Relationen bekannt und in der Modellierungssprache abgebildet sein.

Der Begriff Wirkkettenmodellierung ist je nach Disziplin vorgeprägt. FREI verwendet den Begriff aus der mechanischen Perspektive auf Basis der VDI 2221 zur Beschreibung der Wirkzusammenhänge zwischen (Teil-)Funktionen und mechanischen Wirkflächen, Wirkflächenpaarungen und Wirkräumen [Fre00, S. 2]. KRAMER UND MÜNZENBERG hingegen beschreiben die Wirkkettenmodellierung als nachvollziehbaren Datenfluss innerhalb eines Systems zur Identifikation der Datenweitergabe und der Datenverarbeitung am Beispiel von Fahrerassistenzsystemen [KM10, S. 8]. ALT beschreibt technische Wirkkettenmodelle als einen funktionalen Verbund der Systemelemente [Alt12, S. 96]. Die funktionale Sichtweise auf die Wirkketten wird dabei als Ergänzung zu der technisch-physikalischen Architektur gesehen, um einen ganzheitlichen Überblick über das betrachtete System zu erhalten [Alt12, S. 96]. Basierend auf den Definitionen lassen sich Wirkketten aus einer interdisziplinären Sichtweise wie folgt definieren:

***Modellbasierte Wirkketten** mechatronischer und cyber-physischer Systeme sind Ursache-Wirkungs-Beziehungen, die Elemente und ihre Abhängigkeiten untereinander abbilden. ([GO22a, S. 71] aufbauend auf [GWK22, S. 1887])*

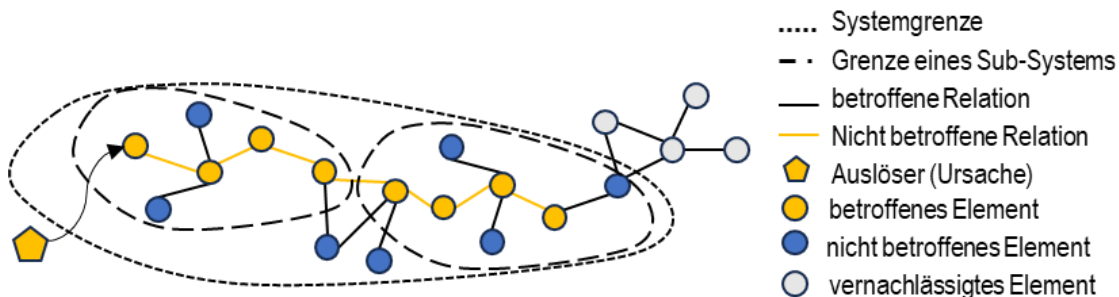


Bild 2-8: Schematische Darstellung einer Wirkkette

Die Modellierung von Wirkketten zielt darauf ab, systematisch die notwendige Menge an Elementen und Abhängigkeiten abzubilden, um spezifische Ursache-Wirkungs-Beziehungen zu erfassen [GWK22, S. 1887]. Erst die Berücksichtigung der Ursache-Wirkungs-Beziehungen bietet das Potenzial, ganzheitlich und unternehmerisch zu handeln [Sch14, S. 29]. Die sukzessive Ergänzung der Wirkketten um weitere Entwicklungsartefakte und Relationen führt zu einem ganzheitlichen Systemmodell [GO22a, S. 71].

2.1.6 Graphentheorie

Neben der Modellierung mit (semi)-formalen objektorientierten Modellierungssprachen wie SysML können Graphmodelle verwendet werden, um Systeme modellbasiert abzubilden und zu analysieren. **Graphen** können zur Kodierung von Daten verwendet werden, wobei Knoten Objekte des betrachteten Bereichs und Kanten Beziehungen zwischen diesen Objekten darstellen [AAB18, S. 4]. Ein Wissensgraph wird definiert als ein Graph von Daten, der dazu dient, Wissen über die reale Welt zu akkumulieren und zu vermitteln, dessen Knoten Entitäten von Interesse darstellen und deren Kanten potenziell unterschiedliche Beziehungen zwischen diesen Entitäten repräsentieren [HBC22, 73]. Im Vergleich zu objektorientierten Modellierungssprachen wie UML und SysML, in denen Klassen oder Blöcke die zentralen Elemente des Modells sind, liegt in Graphansätzen die Relevanz sowohl auf den Knoten als auch auf den verbindenden Kanten [FGG18, S. 1434]. Durch die Festlegung eines Identifikators und Eigenschaft-Wert-Paaren (Attributen) werden zusätzliche Metainformationen definiert [AAB18, S. 6]. Ein Property-Graph G ist somit ein Tupel $(V, E, \rho, \lambda, \sigma)$, in dem [AAB18, S. 7]:

- V eine endliche Menge an Knoten beschreibt.
- E eine endliche Menge an Kanten beschreibt.
- $\rho: E \rightarrow (V \times V)$ die Zuordnung der Kanten E zu Paaren von Knoten beschreibt.
- $\lambda(V \cup E)$ die Zuordnung eines Sets von Labels zu Knoten und Kanten beschreibt.
- $\sigma(V \cup E) \times \text{Prop}$ die Zuordnung von Eigenschaften zu Knoten und Kanten beschreibt.

Ein existierender Graph G mit einer Knotenmenge V und Kantenmenge E kann in induzierte Untergraphen unterteilt werden [Tur10, S. 23]. In der Graphentheorie werden **homogene** und **heterogene Relationen** unterschieden. Homogene Relationen sind Relationen zwischen Elementen ein und derselben Menge; Heterogene Relationen sind Relationen zwischen Elementen aus verschiedenen Mengen [Sch89, S. 5]. (siehe Bild 2-9).

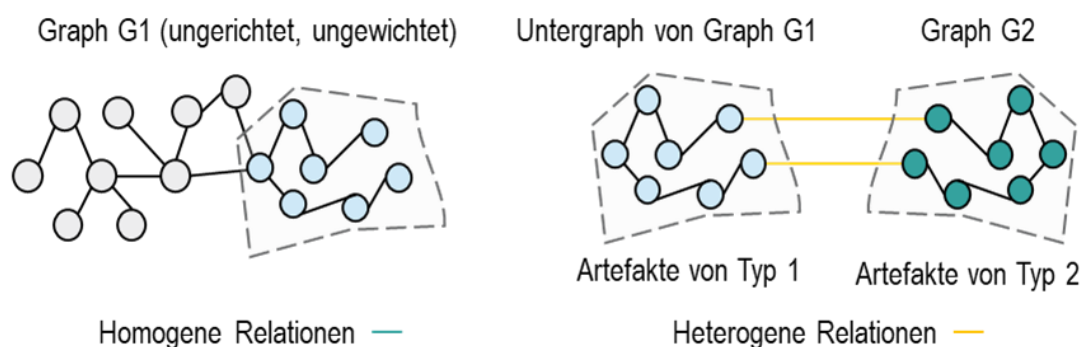


Bild 2-9: Darstellung eines Graphs und eines abgeleiteten Untergraphs

Graphalgorithmen unterstützen den Entwickler bei bestimmten Operationen auf der Datenstruktur des Graphs, beispielsweise bei der Identifikation von benachbarten Knoten oder der Abfrage, ob zwei Knoten miteinander verbunden sind [Tur10, S. 27]. Eine

Abfrage (*Query*) beschreibt eine Operation zum Abrufen eines strukturierten Graphmusters (*graph pattern*), welches in einer Graphdatenbank gesucht wird [AAB18, S. 8]. Die Abfragen können in vier Kategorien unterteilt werden: Lokale Abfragen einzelner Knoten (1), Nachbarschaftsabfragen angrenzender Knoten (2), Traversalabfragen entlang eines Unter-Graphen (3) und globale Abfragen innerhalb des gesamten Graphen (4) [BGP24, S. 10]. In **Abfragesprache** werden Graphmuster syntaktisch ausgedrückt und folgen dem intuitiven Verständnis, Knoten und Kanten mit Pfeilen zwischen ihnen zu kodieren [AAB18, S. 17]. Im Vergleich zu den festen Vorgaben innerhalb einer Abfrage eines strukturierten Graphmusters bieten navigationale Suchabfragen (*Navigational Queries*) zusätzliche Flexibilität [AAB18, S. 21]. Beispiele sind Algorithmen zur Identifizierung des kürzesten Pfads (*shortest path*) wie der Dijkstra Algorithmus, Floyd-Warshall Algorithmus und der Bellman-Ford Algorithmus [KH13, S. 99].

2.2 Stand der Forschung

Ausgehend von der Zielsetzung der Arbeit sowie den Grundlagen innerhalb der relevanten Themenfelder werden im Stand der Forschung problemrelevante Lösungsansätze identifiziert und anhand aktueller Herausforderungen analysiert. Aufbauend auf den identifizierten Defiziten (Kapitel 1.1) werden systematische Literaturrecherchen durchgeführt, um Lösungsansätze aus dem Bereich EC-Auswirkungsanalysen (Kapitel 2.2.1), MBSE-Modellierungsmethoden (Kapitel 2.2.2) und ECM Prozessen (Kapitel 2.2.3) zu identifiziert und zu analysieren. Am Ende der Unterkapitel werden Abgrenzungskriterien definiert, durch die die Forschungslücke und der Handlungsbedarf konkretisiert und das eigene Vorhaben abgegrenzt werden (Kapitel 2.3).

2.2.1 Lösungsansätze für EC-Auswirkungsanalysen

Im Folgenden werden die relevanten Lösungsansätze für EC-Auswirkungsanalysen identifiziert und in Hinblick auf die identifizierten Herausforderungen analysiert. Die systematische Literaturrecherche wurde nach den ersten vier Schritten des Vorgehens von MACHI & MCEVOY [MM12] durchgeführt und in den *Proceedings des Design for X Symposiums 2020* vorveröffentlicht [GW21]. Im Jahr 2024 wurde das Vorgehen erneut durchgeführt, um hinzugekommene Ansätze seit 2020 zu ergänzen. Ziel der Literaturrecherche ist die Identifikation von Ansätzen der modellbasierten EC-Auswirkungsanalyse. Basierend auf der Analyse und Bewertung von Literaturplattformen nach [GH20, 205 ff.] GUSENBAUER und HADDAWAY wurde folgende Auswahl getroffen: Science Direct als hauptsächliche Literaturplattform zur Identifikation von Konferenz- und Journalbeiträgen sowie IEEE Explore Library und Google Scholar als ergänzende Plattformen zur Vervollständigung der Suche. Die verwendeten Suchstrings, Eingrenzungen und KO-Kriterien sowie die Anzahl der identifizierten Ansätze werden in Bild 2-10 illustriert.

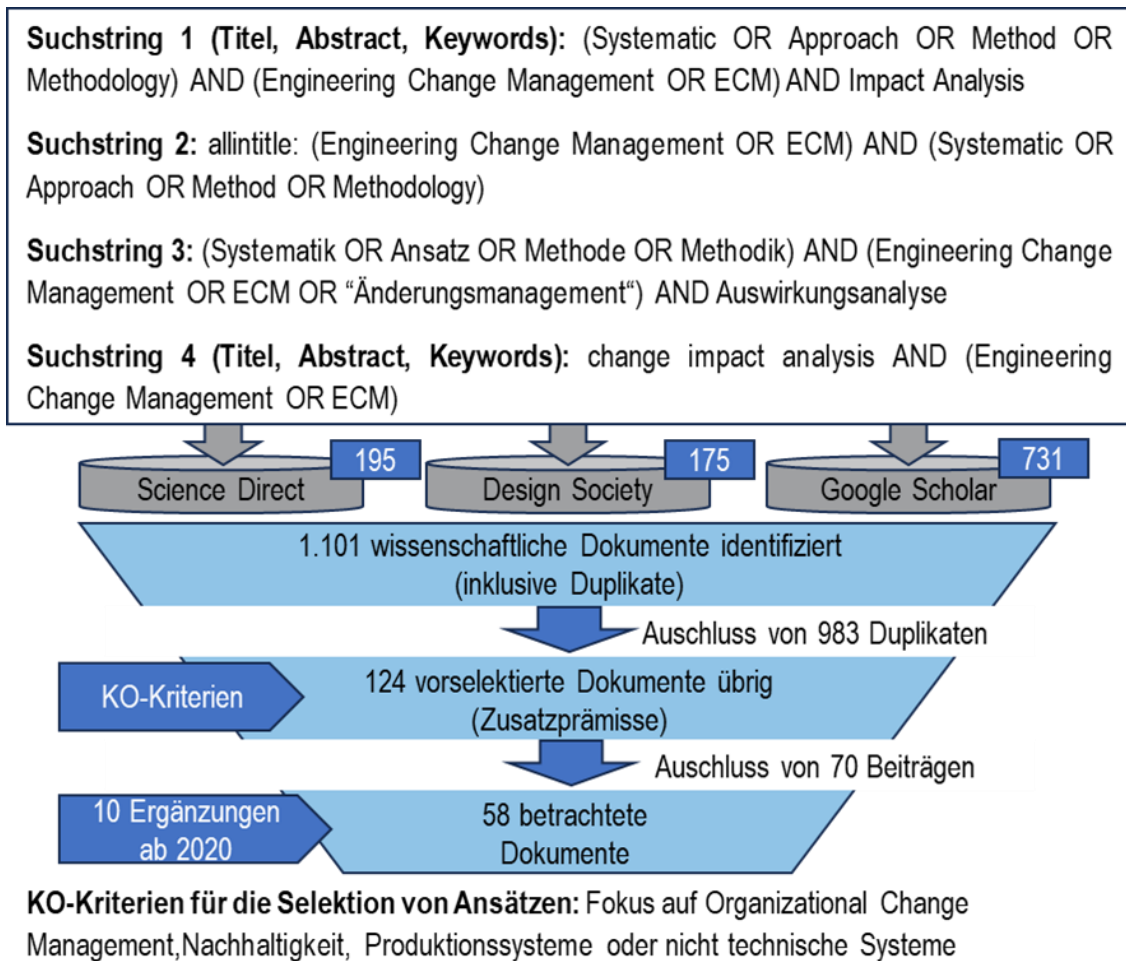


Bild 2-10: Darstellung der systematischen Literaturrecherche in Anlehnung an Wortmann [WBC20, S. 72]

Insgesamt wurden 58 Ansätze für EC-Auswirkungsanalyse identifiziert (siehe Anhang A2.2, S. 144). Im Folgenden werden ausgewählte Ansätze beschrieben, welche anhand von KO-Kriterien (siehe A2.3 S. 150) wie der Relevanz, Aktualität und Interdisziplinarität ausgewählt wurden. Die ausgewählten Ansätze werden anhand des zugrunde liegenden Datenmodells in drei Kategorien vorgestellt: matrixbasierte Ansätze, graphenbasierte Ansätze und objektorientierte Modellierungsansätze. Neben den folgenden Zusammenfassungen der Ansätze werden ausgewählten Ansätze in Steckbriefen detailliert beschrieben (siehe Anhang A2.4, S. 152).

Matrizenbasierte Ansätze

Quadratische Matrizen werden in der Entwicklung eingesetzt, um Zusammenhänge zwischen Modellelementen abzubilden, beispielsweise als Design Structure Matrix (DSM) [Epp12, S. 6, Epp91] oder von Abhängigkeiten zwischen Tätigkeiten der Produktentstehung [ES99, S. 126]. Jede Zelle der Matrix enthält eine numerische oder binäre Information über die Abhängigkeit zwischen den betrachteten Artefakten der Zeile und

Spalte [CSE04, S. 789]. DSMs können als Datenmodell genutzt werden, um Änderungsauswirkungen zu analysieren. In Bild 2-11 wird das Vorgehen konzeptionell dargestellt.

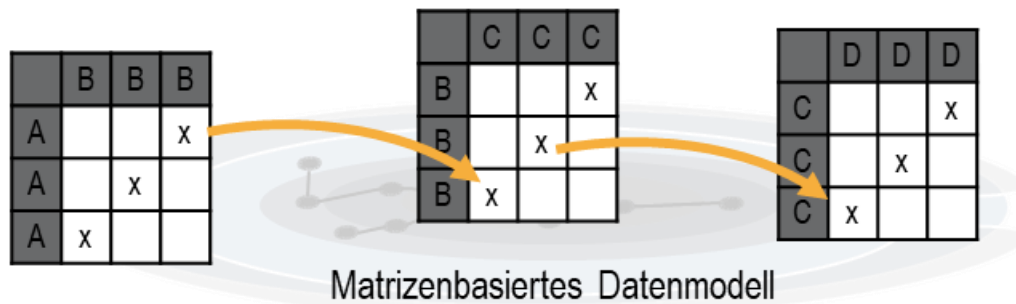


Bild 2-11: Schematische Darstellung eines matrizenbasierten Datenmodells

CLARKSON ET AL. [CSE04] veröffentlichten im Jahr 2004 die Change Prediction Method (CPM) zur Vorhersage von Änderungsauswirkungen. Basierend auf DSMs zur Abbildung von Abhängigkeiten zwischen Systemelementen wird das direkte und indirekte Propagationsrisiko berechnet. Zur Berechnung des direkten Änderungsrisikos werden die Änderungswahrscheinlichkeit und -auswirkung einer Änderung multipliziert. Indirekte Änderungsrisiken werden mithilfe eines Propagationsbaums abgeschätzt.

Aufbauend auf dem Characteristics-Properties Modeling / Property-Driven Development (CPM-PDD) Ansatz nach WEBER [Web05] entwickeln KÖHLER ET AL. [Köh09] einen matrizenbasierten CPM-PDD Ansatz, um die Relationen zwischen den Charakteristika und Eigenschaften zu repräsentieren und Auswirkungen technischer Änderungen zu bewerten [KCW08, S. 172] – den PD/CIRA Ansatz [Köh09, S. 196]. In diesem Ansatz werden von einem ECR betroffene Merkmale innerhalb der Beziehungsmatrizen zwischen Merkmalen und Charakteristika identifiziert. Der Ansatz unterstützt sowohl bei der Analyse der Auswirkungen eines ECRs als auch bei der Vorbereitung einer Engineering Change Order.

WICKEL definiert eine Methodik zur datenbasierten Analyse der Änderungsausbreitung. Die Methodik besteht aus drei Bestandteilen: (1) einem Vorgehen zur Erstellung eines Änderungsstrukturmodells, (2) einer Analyse der Änderungsstruktur zur Identifikation von Handlungsfeldern des ECMs und (3) einer Analyse zur Vorhersage von Änderungsausbreitungen [Wic17, S. 67]. Mithilfe eines sechsschrittigen Vorgehens [Wic17, S. 70] werden die Ausbreitungswahrscheinlichkeit und -häufigkeit basierend auf Abhängigkeiten in Multi-Domain-Matrizen bewertet [Wic17, S. 103].

GRÄBLER ET AL. [GOS20] entwickeln einen systematischen Ansatz zur Bewertung des Risikos von Anforderungsänderungen. Das Risiko wird auf Basis der Änderungswahrscheinlichkeit und der Änderungsauswirkung berechnet. Die Änderungswahrscheinlichkeit wird in endogene und exogene Änderungswahrscheinlichkeit unterteilt. Mithilfe von definierten Bewertungsregeln und -kennzahlen wird das

Änderungsrisiko quantifiziert. Die Anwendbarkeit der Methode wird durch einen software-gestützten Prototypen unterstützt und in Fallstudien nachgewiesen. [GOP22]

MORKOS ET AL. [MSS12] untersuchen die Nutzung von DSMs höherer Ordnung zur Vorhersage von Propagationseffekten bei Anforderungsänderungen. In zwei durchgeführten Studien werden Propagationseffekte in modellierten DSMs analysiert. In der ersten Studie wurden syntaktisch-subjektive Beziehungen verwendet; in der zweiten Studie verschiedene Arten von Relationen basierend auf Schlüsselwörtern. Ergebnisse der Studien zeigen, dass sich Änderungen zweiter Ordnung am wahrscheinlichsten ausbreiten und zum Zeitpunkt der Änderung schwer vorherzusehen sind. Ebenso müssen Beziehungsgewichtungen entwickelt werden, um eine Gleichstellung der Relationen zwischen Anforderungen zu verhindern und die vorgeschlagenen Propagationspfade einer initial betroffenen Anforderung zu begrenzen.

Analyse der matrizenbasierten Ansätze

Matrizenbasierte Ansätze werden genutzt, um Abhängigkeiten zwischen Elementen der Entwicklungsartefakte zu identifizieren und das Expertenwissen der Anwender zu formalisieren. Die einzelnen Analysefunktionalitäten der matrizenbasierten Ansätze sind determiniert. Die Anpassbarkeit der zu analysierenden Propagationspfade auf weitere Entwicklungsartefakte wird nicht unterstützt. Zur Erstellung und Bewertung der Daten wird subjektives Expertenwissen benötigt [CSE04, S. 792]. Einzelne objektive Bewertungskriterien wie die Berechnung des direkten und indirekten Änderungsrisikos [CSE04, S. 792, GOS20, S. 14] oder Maßzahlen [Wic17, S. 50f.] werden vorgeschlagen, aber nicht an die Art des betrachteten ECs angepasst. Als Ergebnissichten werden vordefinierte Sichten auf die Matrizen [Köh09, S. 178], Tabellen [CSE04, S. 796] oder Portfolios [GOS20, S. 15] verwendet, welche nicht an die individuellen Informationsbedarfe der Nutzer angepasst werden. Verwendete IT-Lösungen werden nicht genannt oder stammen aus dem akademischen Umfeld. Eine hauptsächliche Limitation der matrizenbasierten Datenmodelle ist, dass die Matrizen in sich geschlossene Modelle sind, die festgelegte Artefakte wie Komponenten [CSE04, S. 791], Charakteristika und Merkmale [Köh09, S. 124] oder Anforderungen [MSS12, S. 909, GOS20, S. 7] abbilden oder separat zu pflegende Änderungsstrukturmodelle [Wic17, S. 103] erzeugen. Die Integration weiterer Partialmodelle wird in den betrachteten Ansätzen nicht umgesetzt, wodurch relevante Entwicklungsdaten von den DSMs separiert sind [GW21, S. 8]. Zusätzlich wird die Anzahl der betrachteten Elemente zur Reduzierung des Aufwands limitiert, beispielsweise auf eine Anzahl von 50 Komponenten [CSE04, S. 792]. Die prozessuale Integration und die kollaborative Anwendung werden lediglich von einzelnen Ansätzen expliziter adressiert, beispielsweise durch die Umsetzung von Funktionalitäten zum Datenaustausch [GOS20, S. 20].

Graphenbasierte Ansätze

Graph-Datenmodelle werden genutzt, um Änderungsauswirkungen zu identifizieren und zu bewerten. Im Folgenden werden graphenbasierte Lösungsansätze beschrieben und analysiert. Ansätze der Netzwerktheorie werden mit betrachtet, da die Forschungsgebiete eng verwoben sind [Wic17, S. 44]. Das Konzept wird in Bild 2-12 illustriert.

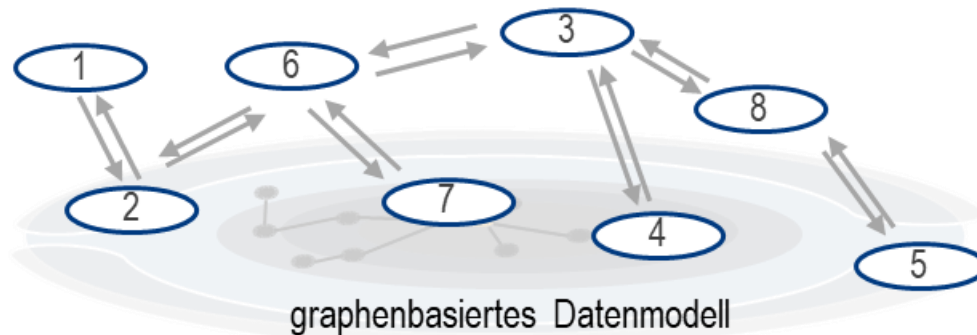


Bild 2-12: Schematische Darstellung eines graphenbasierten Datenmodells

PASQUAL und DE WECK [PW12] entwickeln ein Netzwerkmodell, welches die drei Ebenen durch sogenannte Inter-Layer-Relationen verknüpft: die Produktebene, die Änderungsebene und die soziale Ebene. Das Netzwerkmodell wird mithilfe von Werkzeugen und Metriken analysiert, um Propagationseffekte zu identifizieren und Auswirkungen zu bewerten [PW12, S. 305]. Zur Bewertung der Auswirkungen werden einzelne Metriken wie die Ausbreitungs-Direktheit (*Propagation Directness*) angewendet, mit welcher die Anzahl der internen Schnittstellen auf Produktebene identifiziert werden, die von einer Instanz einer Eltern-Kind-Propagation überspannt werden [PW12, S. 310].

YANG und DUAN [YD12] definieren eine Methode zur Identifikation von Propagationspfaden in einem graphenbasierten Produktmodell auf parametrischer Ebene, dem sogenannten Parameter Linkage Network (PLN). In diesem werden Parameter (Knoten) durch fundamentale und restriktive Verknüpfungen (Kanten) verbunden. Parameter auf der untersten Ebene des PLN beziehen sich beispielsweise auf Durchmesser, Grenzspannungen und Temperaturen, die direkt von den Entwicklern angepasst werden können. Basierend auf dem Modell werden Entwickler methodisch bei der Identifikation und Bewertung von negativen und positiven Auswirkungen der Propagationspfade entlang der parametrischen Beziehungen unterstützt.

CHENG und CHU [CC12] entwickeln einen Ansatz zur Erstellung eines gewichteten Graphnetzwerkmodells, welches mithilfe von drei Veränderlichkeitsindizes analysiert wird. In dem Graphnetzwerkmodell wird das betrachtete Produkt strukturell in Baugruppen und Komponenten dekomponiert, Relationen zwischen den Komponenten durch Expertenwissen modelliert und Gewichtungsfaktoren der Relationen festgelegt [CC12, S. 1421]. Die Änderungsauswirkungen innerhalb des Graphnetzwerkmodells werden unter anderem anhand des Grads der Veränderlichkeit bewertet [CC12, S. 1419].

FEI ET AL. [FGO11] entwickeln eine Methode zur Bewertung von Änderungsauswirkungen zwischen Kundenanforderungen, funktionalen Anforderungen und physischen Komponenten. Die Methode wird in drei Schritte unterteilt: Systemmodellierung, Propagationsanalyse und wissensbasiertes Lösen von Entwicklungskonflikten [FGO11, S. 541]. Im ersten Schritt werden funktionale Anforderungen und physische Komponenten in Partialmodellen wie einem SysML-Modell und einem mechanischen Computer-Aided Design (M-CAD) Modell abgebildet. Im zweiten Schritt werden die Relationen in einer mehrdimensionalen Matrix zusammengeführt, wodurch analysierbare Propagationspfade entstehen. Im letzten Schritt werden potenzielle Konflikte bei der Umsetzung einer technischen Änderung auf Basis eines Wissensgraphen bewertet, welcher historische Änderungen abbildet. In dem Ansatz wird demonstriert, wie mehrere Arten von Datenmodellen kombiniert werden können, um Änderungsauswirkungen zu bewerten.

OLLINGER und STAHOVICH [OS01] entwickeln eine software-gestützte Lösung, um Änderungen von Konstruktionsparametern zu bewerten. Das betrachtete System wird mithilfe der von den Autoren entwickelten Softwarelösung RedesignIT in einem gerichteten Graphmodell abgebildet. Knoten repräsentieren relevante Parameter des Systems, Kanten beschreiben Beschränkungen oder kausale Einflüsse zwischen den Knoten. Das entstehende Modell wird verwendet, um unerwünschte Auswirkungen vorherzusagen und den potenziellen Auswirkungen durch Änderungsentscheidungen entgegenzuwirken.

Analyse der graphenbasierten Ansätze

Die graphenbasierten Datenmodelle bieten die Möglichkeit, Auswirkungsanalysen automatisiert und objektiv durchzuführen. Die Analysefunktionalitäten sind determiniert und werden nicht an die Art des ECRs angepasst. Erste grundlegende Konzepte existieren, um beteiligte Rollen in die Auswirkungsanalyse mit einzubeziehen, beispielsweise durch die Betrachtung der sozialen Ebene [PW12, S. 306]. Die Bewertung der Auswirkungen wird vereinzelt durch objektive Bewertungskriterien wie die Veränderlichkeits-Metriken [CC12, S. 1419] oder die Ausbreitungsdirektheit [PW12, S. 310] unterstützt, beruhen sonst aber weiterhin auf subjektivem Expertenwissen. Die Aussagekraft der quantifizierten Ergebnisse der Auswirkungsanalysen wird nicht durch Anwender bestätigt. Die Ergebnisse der graphenbasierten Auswirkungsanalysen werden in vorgegebenen Formaten präsentiert, beispielsweise Unter-Graphen [YD12, S. 371] oder Tabellen [FGO11, S. 551, CC12, S. 1430]. Auch wenn einzelne IT-Lösungen wie die Softwarelösung Redesign-IT [OS01, S. 197] entwickelt werden, beruht keiner der Ansätze auf frei zugänglichen Graphdatenbanken. Graphenbasierte Ansätze bieten das Potenzial, Daten durchgängig in einem Modell abzubilden und separierte Datengrundlagen zusammenzuführen. Hierbei können Daten in der Tiefe und Breite an die benötigten Informationsbedarfe angepasst werden, beispielsweise auf funktionaler Ebene [FGO11, S. 539] oder auf Parameterebene [OS01, S. 206]. Die Integration disziplinspezifischer Partialmodelle wird nur in Einzelfällen ermöglicht [GW21, S. 8], beispielsweise bei der Integration von SysML-

Modellen und M-CAD Modellen [FGO11, S. 541]. Die Anzahl der zu beachtenden Parameter des Systems ist im Vergleich zu interdisziplinären komplexen Systemen gering [YD12, S. 367]. Bei graphenbasierten Ansätzen handelt es sich häufig um separate Modelle, die aktuell selten als zentrales Entwicklungsartefakt fungieren. Der manuelle Aufwand zur Erstellung und Pflege des separaten Modells ist demnach hoch. Die standortunabhängige Kollaboration ist aufgrund fehlender Kollaborations-Software limitiert.

Objektorientierte Modellierungsansätze

Neben matrizenbasierten und graphenbasierten Ansätzen existieren objektorientierte Modellierungsansätze, die eine Analyse von technischen Änderungen ermöglichen. Objektorientierte Modellierungsansätze nutzen existierende Modellierungssprachen wie UML [Obj17] und SysML [OMG19], um die Bestandteile eines Systems systematisch und formalisiert abzubilden. Die Auswirkungsanalysen basieren auf der Semantik und Syntax der Modellierungssprache, welche bedarfsgerecht erweitert werden kann, beispielsweise in einem UML-Profil. Das Konzept wird in Bild 2-13 dargestellt.

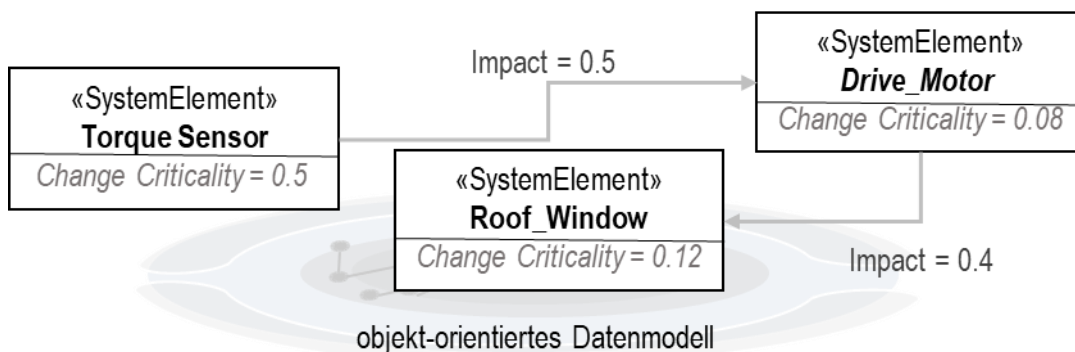


Bild 2-13: Konzeptionelles objektorientiertes Metadatenmodell in Anlehnung an [WKI20, S. 1172]

WILMS ET AL. [WKI20] entwickeln ein Konzept für ein domänenübergreifendes Produktmodell für das ECM. Das Konzept umfasst die Zusammenführung der funktionalen Struktur, der Produktstruktur und der Produktportfoliostruktur in einem SysML-Systemmodell [WKI20, S. 1168f.]. Es werden drei Sichten des Entwicklers unterschieden, die bei der Nutzung des Modells für das ECM unterstützen sollen: die funktionale Sicht, die strukturelle Sicht und die Auswirkungsanalysesicht [WKI20, S. 1169]. Durch die Nutzung weiterer Attribute wie der Änderungskritikalität und der Verantwortlichkeit wird die Detaillierungstiefe der Funktionen und Komponenten erhöht [WKI20, S. 1171f.].

MARTIN ET AL. [MKP22] entwickeln den Advanced Engineering Change Impact Approach (AECIA) für die holistische Betrachtung und Analyse von Änderungspropagationen. Die Aktivitäten des Ansatzes umfassen die Prüfung der Validität einer Änderungsanfrage, die Analyse und Modellierung der Änderungsausbreitung und -auswirkung sowie die Kommunikation und Dokumentation von Änderungsinformationen [MKP22,

S. 4]. Grundlage der Modellierung ist die IPEK-MBSE-Methodik [MMA22], welche um Elemente des Engineering Change Managements erweitert wird. Durch die Berücksichtigung von Kommunikationsaufwänden wird die prozessuale Integration in den Engineering Change Management Prozess vorgesehen [MKP22, S. 6].

JAGLA ET AL. [JJS21] entwickeln eine Methode auf Basis von SysML-Systemmodellen, um den Aufwand von technischen Änderungen abschätzen zu können. In dem SysML-Systemmodell werden sowohl Baugruppen und Komponenten abgebildet als auch konstruktive Elemente, beispielsweise Kraftflüsse zwischen zwei Komponenten. Ausgehend von dem SysML-Systemmodell werden Verknüpfungen zu Simulationsmodellen wie einer Finiten Elemente Analyse erstellt, wodurch die Detaillierungstiefe erhöht wird [JJS21, S. 93]. Zur Bewertung technischer Änderungen werden Abfragen mithilfe des im Modellierungswerkzeug integrierten Metachain-Navigator formuliert [JJS21, S. 91].

BRIAND ET AL. [BLO06] entwickeln einen UML-basierten Ansatz, indem Änderungsauswirkungen mithilfe von Regeln in der Object Constraint Language (OCL) deklariert und auf UML-Diagramme des Systems angewendet werden. [BLO06, S. 339] Die Autoren definieren eine Änderungstaxonomie und 97 Regeln zur Bewertung von Änderungen. Die Regeln beziehen sich auf unterschiedliche Änderungskategorien, beispielsweise Änderungen an Strukturelementen [BLO06, S. 342].

COHEN ET AL. [CNF00] entwickeln die Methodik C-FAR (Change Favorable Representation), in der existierende Produktdaten analysiert werden, um Auswirkungen auf das Produkt zu berechnen. Die Anwendung umfasst die Repräsentation der Produktdaten und die Identifikation und qualitative Bewertung der Propagationseffekte. Grundlage zur Abbildung des Ansatzes ist das EXPRESS Schema, in welchem Entitäten, Attribute und Relationen unterschieden werden. Aufbauend auf dem EXPRESS Schema werden Produktdaten genutzt, um Propagationspfade zu identifizieren und Abhängigkeiten zwischen Attributen durch Vektormultiplikationen zu analysieren [CNF00, S. 336].

Analyse der objektorientierten Modellierungsansätze

Objektorientierte Modellierungsansätze bieten das Potenzial, Auswirkungsanalysen automatisiert durchzuführen und dabei an die Art der Änderungsanfrage anzupassen. Limitationen in der Anwendung der Auswirkungsanalysen liegen in der Nutzung zusätzlicher Abfragesprachen und Funktionalitäten wie OCL [BLO06, S. 339] oder Metachains [JJS21, S. 91], für welche Vorkenntnisse benötigt werden. Die Bewertung der Ergebnisse beruht auf dem Expertenwissen der Anwender, beispielsweise bei der Interpretation alphabetischer oder numerischer Werte aus Vektormultiplikationen [CNF00, S. 336] und bei der Interpretation des Änderungsrisikos [WKI20, S. 1172]. Die Herleitung und Gewichtung weiterer Bewertungskriterien ist möglich [MKP22, S. 5]. Einzelne Autoren nutzen die Konzepte von Sichten und Perspektiven auf das Modell, um differenzierte vordefinierte Ergebnissichten zu erzeugen, beispielsweise

funktionale und strukturelle Sichten [WKI20, S. 1168]. Zur Modellierung nutzen Ansätze lizenz-basierte existierende IT-Lösungen wie den Cameo Systems Modeler von D'assault Systèmes [JJS21, S. 94]. Objektorientierte Modellierungsansätze können zur durchgängigen Modellierung genutzt werden und separate Datengrundlagen zusammenführen, beispielsweise indem Simulationsmodelle integriert werden [JJS21, S. 93]. Die zu erreichende Detaillierungstiefe ist abhängig von der gewählten Modellierungssprache. UML- und SysML-Ansätze [JJS21, S. 91, BLO06, S. 339, WKI20, S. 1172] sind durch die Nutzung von Profilerweiterungen nicht limitiert, EXPRESS [CNF00, S. 322f.] hingegen schon. Ein Vorgehen zur prozessualen Integration wird nur vereinzelt beschrieben, beispielsweise durch die Betrachtung der Kommunikation und Dokumentation [MKP22, S. 4]. Vorteile zur Verbesserung der Kommunikation im ECM werden beschrieben [MKP22, S. 6], aber nicht quantitativ nachgewiesen.

Fazit und Abgrenzungskriterien

Die Analyse der Ansätze verdeutlicht, dass die aktuellen Analysefunktionalitäten die Defizite bei der frühzeitigen, effizienten und effektiven Bewertung von Änderungsauswirkungen nicht hinreichend lösen. Daraus resultieren Kriterien (K), die von künftigen Ansätzen gelöst werden müssen. Existierende Ansätze sind in der Analyse auf eine determinierte Anzahl auswertbarer Propagationspfade limitiert. Daher müssen Auswirkungsanalysen **maßgeschneiderte Analysefunktionalitäten** bereitstellen, um Propagationspfade unterschiedlicher Art zu analysieren und betroffene Elemente frühzeitig zu identifizieren (K1). Die Bewertungsgrundlage und die Bewertungskriterien aktueller Ansätze basieren vorrangig auf subjektivem Expertenwissen. Für die Auswirkungsanalyse müssen **objektive Bewertungskriterien** ergänzt werden, um die Subjektivität der Expertenangaben zu reduzieren und Fehleinschätzungen zu vermeiden (K2). Einzelne Ansätze unterstützen nur festgelegte Sichtweisen, welche nicht an die Bedarfe der Anwender angepasst werden. Die Ergebnisse der Auswirkungsanalyse müssen in **anwenderspezifischen Sichten** (K4) dargestellt werden, die an die individuellen Bedürfnisse der Nutzer angepasst werden können. Zusätzlich bietet die explizite Modellierung von Ursache-Wirkungs-Zusammenhängen das Potenzial der **Automatisierbarkeit** (K3), um die Anwendbarkeit der IT-Lösungen für EC-Auswirkungsanalysen zu steigern. Dazu müssen sowohl die Identifikation betroffener Elemente als auch die Bewertung der Auswirkungen durch eine **IT-gestützte Implementierung** der Funktionalitäten gewährleistet sein (K5).

2.2.2 Lösungsansätze für die durchgängige Modellierung

Die Analyse der Ansätze für EC-Auswirkungsanalysen zeigt, dass das Datenmodell ausschlaggebend für die Vielfalt, Aussagekraft und Qualität der Ergebnisse einer EC-Auswirkungsanalyse ist. MBSE-Modellierungsansätze bieten Potenziale, um Ursache-Wirkungs-Beziehungen zwischen interdisziplinären Entwicklungsartefakten explizit in

durchgängigen Datenmodellen abzubilden. Folgend werden daher aktuelle MBSE-Modellierungsansätze identifiziert und analysiert, um Potenziale für EC-Auswirkungsanalysen abzuschätzen. Zur Identifikation der Lösungsansätze wurde eine Literaturrecherche durchgeführt und dokumentiert (siehe Anhang A2.6, S. 168 und A2.7, S. 170). Die Suchstrategie umfasst die Zielfestlegung, die Ableitung der Suchstrings, die Auswahl von Literaturdatenbanken und die Schneeballsuche in den identifizierten Literaturstudien von ESTEFAN [Est08, 15 ff.], WEILKIENS ET AL. [WSD16, 1 ff.] und MADNI [MS18, 175 ff.]. Die Modellierungsmethode SYSMOD von WEILKIENS [Wei07, S. 23] ist ein SysML-basierter, anwendungsfall-orientierter Ansatz zur Erstellung eines Systemmodells. Ausgehend von Anwendungsfällen wird die Anforderungserhebung durchgeführt und die Systemarchitektur erstellt. Zur Anwendung der Modellierungsmethode wird ein SysML Profil mit Stereotypen bereitgestellt. SYSMOD kann ebenfalls erweitert werden, beispielsweise mit Fokus auf funktionale Architekturen [MGW16] oder auf die Variantenmodellierung [Wei16]. In Hinblick auf die Detaillierungstiefe werden drei Stufen der Modellierungsintensität unterschieden (siehe Bild 2-14).

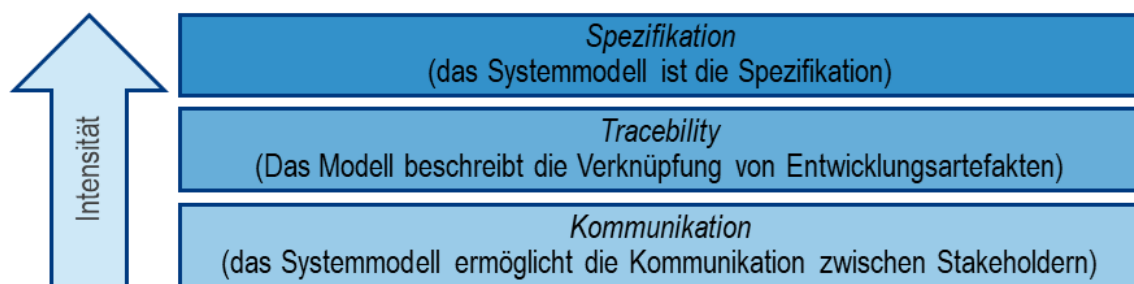


Bild 2-14: Modellierungsintensität nach Weilkiens

FRIEDENTHAL [Fri14] entwickelt die Object-oriented Systems Engineering Method (OOSEM) zur Erstellung von SysML-Systemmodellen. OOSEM ist ein szenario-getriebener Modellierungsansatz, welcher Entwickler bei der Analyse, Spezifikation, Entwicklung und Verifikation eines Systems unterstützt [Fri14, S. 405]. OOSEM umfasst Modellierungsschritte wie die Black-Box und White-Box Betrachtung, die logische Dekomposition [Fri14, S. 406]. Die Anwendung von OOSEM zum Aufbau eines detaillierten Systemmodells umfasst 46 Aktivitäten, die 6 hauptsächlichen Prozessschritten zugeordnet werden (siehe Bild 2-15). Bei der Festlegung der notwendigen Modellierungsintensität werden die Dimensionen Modellierungsbreite, Modellierungstiefe und Modellierungsgenauigkeit unterschieden. [Fri14].

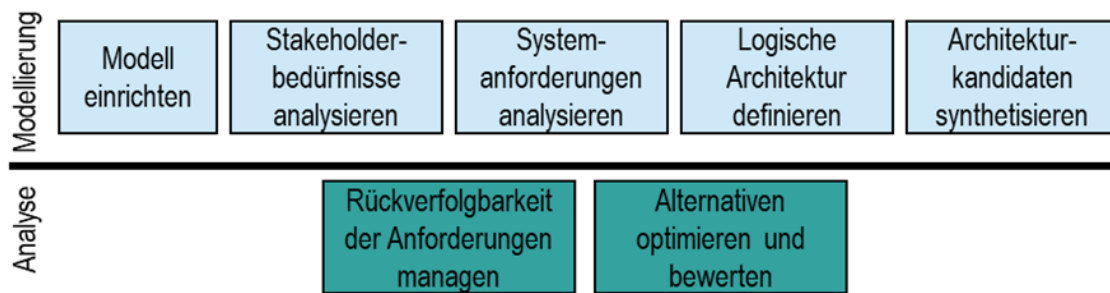


Bild 2-15: Prozessschritte von OOSEM nach Friedenthal [Fri14, S. 411]

EIGNER [Eig21] entwickelt die Virtuelle Produktentwicklung (VPE) System Development Methodology, welche sich an einem V-Modell orientiert und in der Entwicklungsartefakte auf vier Ebenen betrachtet werden: das Kontext Level, das funktionale Level, das logische Level und das physische Level. Hierbei handelt es sich um eine Modifikation des weitverbreiteten RFLP-Ansatzes [Eig21, S. 83]. In RFLP-Ansätzen werden Abhängigkeiten zwischen den Entwicklungsartefakten Anforderungen (*Requirements*), Funktionen (*Functions*), Logische Elemente (*Logical Elements*) und physische Elemente (*Physical Elements*) betrachtet [GWO22, S. 1, KK13, S. 93, ZBL14, S. 247, HST23, S. 120]. Die Konkretisierungslevel werden vom Anforderungsraum, Administrationsraum und Verifikations- und Validierungsraum umspannt und bilden so die zentrale Abbildung des Kaiserslauterner Systemkonkretisierungsmodells (KSKM) (siehe Bild 2-16). Das KSKM wird in der VPE System Development Methodik durch drei weitere Bausteine ergänzt: (1) MVPE-Modell als Prozess, (2) ein IT-Architekturkonzept und (3) die Abbildung der Makro-Methodik auf eine detaillierter Mikro-Methodik [Eig21, S. 80].

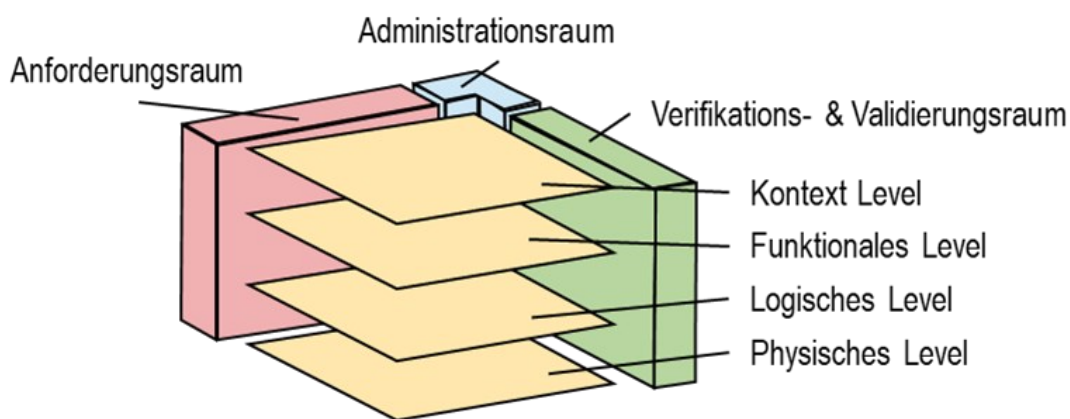


Bild 2-16: Schematische Darstellung der Ebenen des KSKM nach Eigner [Eig21, S. 84]

ALT [Alt12] entwickelt einen Ansatz zur Wirkkettenmodellierung und unterscheidet zwischen zwei praxisrelevanten Architektursichten: die technisch-physikalische Architektur und die technische Wirkkettenarchitektur [Alt12, S. 95]. Die technische Wirkkettenarchitektur stellt die Software- und Hardwarekomponenten in Form von Wirkketten in einem funktionalen Verbund dar [Alt12, S. 96]. Die Kombination beider Sichten auf die Architektur ergibt ein ganzheitliches Abbild des Systems [Alt12, S. 96]. Hierbei

empfiehlt der Autor eine Einteilung der Systemelemente und Relationen in die Kategorien Eingabe, Verarbeitung, Ausgabe (EVA-Logik) [Alt12, S. 97]. Die Anwendung der Methode wird in Bild 2-17 illustriert.

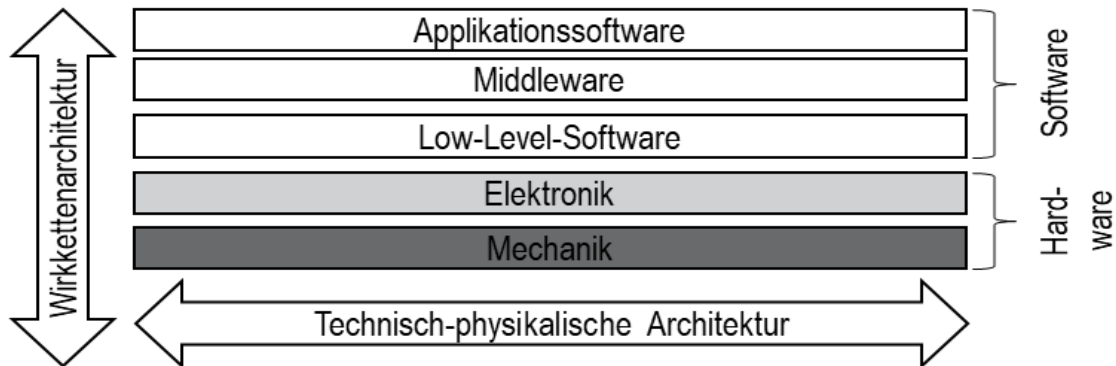


Bild 2-17: Schichtenmodell der Systemarchitektur nach Alt 2012 [Alt12, S. 101]

BAJAJ ET AL. [BBW17, BH18] entwickeln ein graphenbasiertes Systemmodell, welches lebenszyklusbegleitend Daten und Informationen aus unterschiedlichen Ablageorten verknüpft – das Total System Model [BBW17, S. 151]. Der Grundgedanke bei der Entwicklung des Total System Model ist „best in Class“: Für die Erstellung der Modelle werden die geeigneten IT-Werkzeuge genutzt und kombiniert, anstatt alle Entwicklungsartefakte in einem integrierenden Systemmodell abzubilden [BBW17, S. 167]. Ausgehend von der Interpretation, dass alle Partialmodelle separate Graphausschnitte darstellen, fungiert das TSM als „graph-of-graph“ [BBW17, S. 154]. Durch diesen abstrakten Ansatz entstehen unterschiedliche Graph-Topologien, beispielsweise mit führenden Modellen wie einem SysML-Modell oder stark verteilte Graphen, welche aus einer Vielzahl von Partialmodellen bestehen [BBW17, S. 153f.]. Das Konzept wird in Bild 2-18 verdeutlicht.

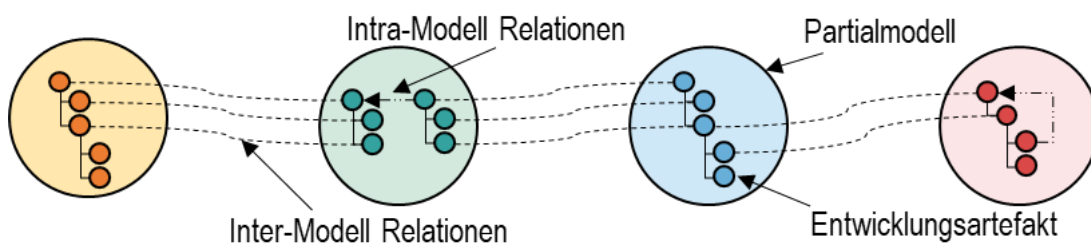


Bild 2-18: Konzeptionelle Darstellung des Total System Models nach [BBW17, S. 154]

Analyse von Lösungsansätzen für die durchgängige Modellierung

MBSE-Modellierungsansätze bieten das Potenzial, separate Datengrundlagen durch ein durchgängiges Systemmodell aufzulösen. SysML-basierte Ansätze nutzen die Semantik und Syntax der semiformalen Modellierungssprache, um Modellelemente auf unterschiedlichen Detaillierungstiefen abzubilden. Die Nutzbarkeit der SysML-basierten

Ansätze für EC-Auswirkungsanalysen sind bisher limitiert, da lediglich in SysML abgebildete Systemelemente einbezogen werden und weitere Partialmodelle vernachlässigt werden. Limitationen der Ansätze SysMOD [Wei07] und OOSEM [Fri14] bestehen in der Vielzahl der durchzuführenden Zwischenschritte, der fehlenden Einbindung von weiteren Entwicklungsdaten und -informationen aus Partialmodellen sowie die fehlende Hilfestellung bei der prozessualen Integration in existierende Entwicklungsprozesse. Im Vergleich dazu unterscheidet sich der Ansatz nach Eigner [Eig21] durch die generische, übersichtliche Struktur entlang der Entwicklungsartefakte. Der Ansatz zur Wirkkettenmodellierung verdeutlicht die Nutzung von den Konzepten Sicht und Perspektive (siehe Kapitel 2.1.4), beispielsweise zur Unterscheidung der technisch-physikalischen Sicht und der Sicht auf eine separate Wirkkette, [Alt12, S. 95]. Der Ansatz von Bajaj et al. verdeutlicht das Potenzial, graphenbasierte Modellierungsansätze zu verwenden, um Systeme durchgängig abzubilden. Ein Vorteil des durchgängigen Graphmodells ist die semantische Anpassbarkeit an die betrachteten Entwicklungsartefakte. Potenziale für Auswirkungsanalysen werden erkannt, beispielsweise für Anforderungsanalysen [Wei07, S. 63] oder Performance Analysen [Fri14, S. 394ff.]. Die Potenziale für das ECM werden nicht erkannt oder lediglich angedeutet [Eig21, S. 164]. Die kollaborative Anwendung der Ansätze führt häufig zu steigenden Lizenzkosten durch zusätzliche IT-Lösungen oder weitere Lizenzschlüssel.

Fazit und Abgrenzungskriterien

Die betrachteten MBSE-Modellierungsansätze bieten das Potenzial, Systeme in einem durchgängigen Datenmodell abzubilden und unterschiedliche Entwicklungsartefakte zu verknüpfen und so als Grundlage für EC-Auswirkungsanalysen zu fungieren. Die Nutzung der entstehenden Modelle für das ECM wird angedeutet, aber nicht weiter detailliert. Basierend auf dem identifizierten Potenzial und den Erkenntnissen der Analyse werden Kriterien mit Fokus auf das Datenmodell festgelegt, die für die Befähigung modellbasierter Wirkkettenanalysen benötigt werden. Zur Sicherstellung der Nutzbarkeit der IT-Lösung müssen Ansätze die **interdisziplinären Entwicklungsartefakte** (Mechanik, Elektrik/Elektronik, Software, ...) abbilden (K6). Diese Entwicklungsartefakte aus separaten Datengrundlagen müssen systematisch in einem **durchgängigen Datenmodell** verknüpft werden können, um Wirkkettenanalysen anwenden zu können (K7). Ebenfalls muss die Detaillierungstiefe mithilfe der **Integration von Partialmodellen** gewährleistet sein, um aussagekräftige Ergebnisse zu folgern (K8).

2.2.3 Lösungsansätze für die Integration in den ECM Prozess

Im ECM Prozess müssen die entstehenden Kosten, der Nutzen sowie die Risiken einer möglichen Implementierung von Änderungen in Hinblick auf ihre Auswirkungen auf

Zeitplan, Budget und Produktqualität abgewogen werden [PW12, S. 305]. Ein beispielhaftes Modell ist der Änderungsprozess nach VDA 4965 [VDA09a, S. 18] und das Modell der Normen DIN 199-4 [DIN22]. Die benötigten Informationen zur Durchführung des ECM sind in verschiedenen Unternehmen gleichbleibend [PM98, S. 11]. Das heißt, ECM Prozessschritte sind auf Makro-Ebene vergleichbar und können vereinheitlicht werden [JEC11, S. 106]. Auf Mikro-Ebene kann es jedoch Unterschiede geben, abhängig von ECM-relevanten Kontextbedingungen. Basierend auf der Erkenntnis, dass die offiziellen ECM Prozesse bei den Unternehmen auf einheitlichen Konzepten beruhen [JEC11, S. 106], haben JARRATT ET AL. ein generisches ECM-Modell entwickelt (siehe Bild 2-19). Zur praktischen Umsetzung werden häufig interdisziplinäre Änderungsmanagement-Komitees gebildet, um über ECRs zu diskutieren und abzustimmen [PM98, S. 2]. Änderungsmanagement-Komitees sollten interdisziplinär aufgestellt sein, beispielsweise bestehend aus Entwicklern, Vertrieblern und Einkäufern [IRA20, S. 4508].

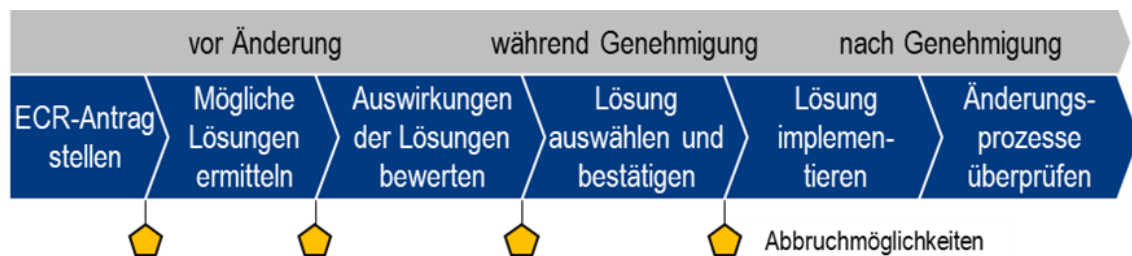


Bild 2-19: Generischer ECM Prozess (nach Jarratt [JEC11, S. 107] [JCE05])

Analyse der Lösungsansätze für die Integration in den ECM-Prozess

Entwickler können in ihren Entscheidungen während des ECM-Prozesses durch Ergebnisse von EC-Auswirkungsanalysen unterstützt werden. Generische Ansätze beinhalten Schnittstellen für EC-Auswirkungsanalysen. Trotz der vorhandenen Schnittstellen werden Herausforderungen bei der Integration von EC-Auswirkungsanalysen in den ECM-Prozess in aktuellen Ansätzen nicht betrachtet. Ebenfalls müssen die IT-Systeme von EC-Auswirkungsanalysen an die prozessual vorgegebenen Kommunikationswege angepasst werden und für die kollaborative Anwendung ausgelegt sein.

Fazit und Abgrenzungskriterien

Zur Gewährleistung des Praxistransfers muss in Ansätzen der EC-Auswirkungsanalyse die **kollaborative Anwendbarkeit** (K9) und die **Integration in Entwicklungsprozesse** (K10) sichergestellt werden. Ebenso muss die **praktische Anwendbarkeit** (K11) nachgewiesen werden, indem der Ansatz in realen Fallstudien von Endnutzern angewendet wird.

2.3 Forschungslücke und Handlungsbedarf

Der Vergleich von existierenden Lösungsansätzen (siehe Tabelle 2-1) verdeutlicht, dass wesentliche Herausforderungen von EC-Auswirkungsanalysen bisher ungelöst sind. Existierende Ansätze umfassen entweder einzelne Auswirkungsanalyse-Funktionalitäten für vordefinierte Datenmodelle (Kapitel 2.2.1) oder bieten Ansätze für durchgängige Datenmodelle, ohne Auswirkungen von ECs zu analysieren (Kapitel 2.2.2). Die unternehmensspezifische Integration in ECM Prozesse (Kapitel 2.2.3) wird von den Lösungsansätzen nebensächlich behandelt.

Tabelle 2-1: Abgrenzung des eigenen Vorhabens von Ansätzen und Forschungsprojekten

Kriterien	Prof. Eigner Eigner [Eig21]	Alt (promoviert bei Prof. Schür) Alt [Alt12]	Bajaj et al. [BBW17]	Prof. J. P. Clarkson Clarkson et al. [Cla04]	Prof. Bley Köhler et al. [KWC08]	Prof. Lindemann Dissertation Wikel [Wic17]	Prof. De Weck (MIT) Pasqual und De Weck [PW11]	Yang und Duan [YD12]	Cheng und Chu [CC12]	Prof. Vietor Wilms et al. [WK120]	Prof. Albers Martin et al. [MKP22]	Prof. Jacobs Jagla et al. [JJS21]	Eigenes Vorhaben
K1: Maßgeschneiderte Analysefunktionalitäten	○	◐	◐	◐	◐	◐	◐	○	○	—	○	◐	●
K2: Objektive Bewertungskriterien	—	—	—	○	—	○	—	◐	○	◐	—	◐	●
K3: Automatisierbarkeit	—	—	—	○	○	◐	○	○	○	○	○	◐	●
K4: anwender-spezifische Sichten	—	—	—	○	○	○	◐	—	—	○	—	—	●
AK5: IT-gestützte Implementierung	○	—	◐	○	—	◐	○	○	○	◐	◐	◐	●
AK6: Interdisziplinäre Entwicklungsartefakte	◐	◐	●	○	○	○	○	○	○	◐	◐	◐	●
AK7: Integration von Partialmodellen	◐	—	◐	—	—	○	—	○	◐	—	○	◐	◐
AK8: durchgängiges Datenmodell	●	◐	●	—	—	○	◐	○	○	◐	◐	○	●
K9: Kollaborative Anwendung	◐	○	◐	—	—	○	—	—	—	○	◐	○	●
K10: Integration in Entwicklungsprozesse	—	—	—	○	—	◐	—	—	○	○	◐	○	●
K11: praktische Anwendbarkeit	—	—	—	○	—	◐	○	○	○	○	○	○	●

Basierend auf den Erkenntnissen der Analyse und den Abgrenzungskriterien wird ein eigener Lösungsansatz entwickelt, mit dem Entwickler eigenständig durchgängige Modelle entlang existierender Ursache-Wirkungs-Muster abbilden und EC-Auswirkungsanalysen mit objektiven Bewertungskriterien umsetzen. Konzepte der Wirkkettenmodellierung werden durch Elemente für maßgeschneiderte EC-Auswirkungsanalysen hin zu sogenannten **Wirkkettenanalysen** zusammengeführt.

Ausgehend von einer systematischen Analyse der Lösungsansätze und den Erkenntnissen aus den Workshops werden **Anforderungen** an den eigenen Lösungsansatz definiert. Einzelne Anforderungen wurden im *MDPI Systems Journal* [GWK23, 8] und auf der *IEEE System of Systems Engineering Konferenz 2023* [GW23, S. 2] veröffentlicht. Die erhobenen Anforderungen wurden in Diskussionen mit industriellen Experten praxisorientiert abgesichert. Für diese praxisorientierte Absicherung der erhobenen Anforderungen wurden zwei digitale sechzigminütige Workshops (WS-2) durchgeführt, in denen die literaturbasierten Anforderungen mit vier industriellen Anwendern, fünf IT-Experten und zwei akademischen Experten diskutiert und bestätigt wurden (siehe Anhang A3.3, S. 178). Zusätzlich wurden vom Verfasser drei Interviews (WS-15) im Kontext der Wirkkettenmodellierung in der deutschen Automobilindustrie durchgeführt, um die erhobenen Anforderungen an die Modellierung zu ergänzen. Befragt wurden eine MBSE-Expertin des Automobilherstellers sowie Projektmanager und ein Partner eines deutschen Systems Engineering Consulting Unternehmens. In den neunzigminütigen Workshops wurden die entwickelten Anforderungen diskutiert und aus praxisorientierter Sicht bestätigt (siehe Anhang A3.4, S. 179).

A1: Die Verfügbarkeit von Daten: Die Durchgängigkeit der Datengrundlage stellt eine Herausforderung dar, nicht nur entlang der späten Lebenszyklusphasen. Die Durchführung der Modellierung mit bestehenden Daten, Informationen und Partialmodellen reduziert den Anwendungsaufwand. *Die Methodik muss so gestaltet sein, dass sie mit bestehenden Entwicklungsdaten durchgeführt werden kann.*

A2: Unterschiedliche Arten der Änderung: Die Heterogenität von Änderungsursachen führt zu der Herausforderung, Analysefunktionalitäten flexibel an die Fragestellung anzupassen. Zusätzlich führt die variierende benötigte Detaillierungstiefe der Daten dazu, dass für die Analyse unterschiedlicher Arten von Änderung unterschiedliche Daten benötigen. *Der Änderungsbedarf muss sich auf unterschiedliche Modellelemente beziehen können.*

A3: Tiefe, Breite und Weite des Wirkkettenmodells: Abhängig von der Analysefunktionalität wird eine variierende Detaillierungstiefe der Daten benötigt. Die Durchgängigkeit der Datengrundlage kann nur sichergestellt werden, indem Partialmodelle eingebunden werden. *Mit der Methodik müssen Daten auf verschiedenen Detaillierungsebenen (Tiefe), entlang von relevanten Entwicklungsartefakten (Weite) und aus verschiedenen Disziplinen (Breite) abgebildet werden.*

A4: Änderungsausbreitung: Unterschiedliche Änderungsarten führen zu unterschiedlichen Ursache-Wirkungs-Mustern. Wirkkettenanalysen müssen die für eine Änderungsart relevanten Ursache-Wirkungs-Muster analysieren, entlang welcher die Änderung propagiert. *Die Wirkkettenanalyse muss spezifische Propagationspfade analysieren.*

A5: Objektive Bewertung: In existierenden Ansätzen werden nur einzelne objektive Kriterien formuliert und angewendet, um Änderungsauswirkungen zu bewerten. Für aussagekräftige Bewertungen von Propagationseffekten muss das subjektive

Expertenwissen durch objektive Bewertungskriterien unterstützt werden. *Die Auswirkungen der technischen Änderung müssen objektiv, qualitativ und quantitativ bewertet werden können.*

A6: Prozessuale Verankerung: Fehlende Ansätze zur maßgeschneiderten prozessualen Integration und mangelnde Berücksichtigung der Kollaboration führen zu einer geringen Akzeptanz in der Praxis. *Die Wirkkettenanalyse muss in die unternehmensspezifischen Entwicklungsprozesse integriert werden können.*

A7: Akzeptabler zeitlicher Anwendungsaufwand: Ein Grund für die fehlende praktische Anwendung von IT-Lösungen ist der zeitliche Anwendungsaufwand. *Der zeitliche Modellierungs- und Anwendungsaufwand der Methodik muss in der industriellen Praxis akzeptabel sein.*

A8: Einfache Anwendbarkeit: Die fehlende praktische Anwendung von IT-Lösungen ist ebenfalls darauf zurückzuführen, dass die meisten Ansätze nur mit Forschungsbeispielen getestet werden, jedoch nicht in der Industrie. *Die Methodik muss so gestaltet sein, dass industrielle Anwender die Anwendbarkeit als einfach bewerten.*

A9: Kollaborative Anwendbarkeit: Je nach Änderungsart werden mehrere Rollen in die Bewertung der Auswirkungen mit einbezogen. Zur Sicherstellung der Praxistauglichkeit müssen Wirkkettenanalysen bereichs- und unternehmensübergreifend durchführbar sein. *Die Datenbereitstellung und -auswertung muss kollaborativ möglich sein.*

A10: Maßgeschneiderte Analyseergebnisse: Die fehlende Anpassbarkeit der Ergebnisse an den ECR führt zu fehlender Akzeptanz der Anwender. *Die Ergebnisse der Methodik müssen an die rollenspezifischen Bedarfe angepasst werden können.*

A11: Ergebnisqualität: Fehlende Flexibilität der Analysefunktionalitäten und die Anpassbarkeit der Ergebnisse an den ECR führen zu einer verringerten Ergebnisqualität. *Die Ergebnisse der Methodik müssen die qualitativen Ansprüche der industriellen Praxis erfüllen.*

A12: Nachvollziehbarkeit: Abhängig von den verwendeten Technologien ist die Herleitung der Ergebnisse vollständig oder nur teilweise nachvollziehbar. Beispielsweise führt der Einsatz von Ansätzen der Künstlichen Intelligenz dazu, dass einzelne Analyseschritte nicht vollständig nachvollziehbar sind. *Die Ergebnisse der Methodik müssen für Anwender nachvollziehbar sein.*

Die Anforderungen werden in vier Kategorien eingeteilt (angelehnt WICKEL [Wic17, 46], aufbauend auf HAMRAZ [Ham13, S. 65]): (1) Eingangsdaten, (2) Methoden, Modelle und Hilfsmittel, (3) Anwendbarkeit, (4) Ergebnis. Jede Anforderung wird durch Identifikationsnummer (ID), Anforderungsnamen, Anforderungstext und Quelle beschrieben. Bei der Quelle wird zwischen Literatur (L) Workshops (WS) Interviews (I) unterschieden. Die Anforderungen werden in Tabelle 3-1 aufgelistet.

Tabelle 3-1: Anforderungen an die Methodik zur modellbasierten Wirkkettenanalyse

K	Anforderungsname	Beschreibung	Quelle
Eingangsdaten	A1: Verfügbarkeit von Daten	Die Methodik muss so gestaltet sein, dass sie mit bestehenden Entwicklungsdaten durchgeführt werden kann.	L, WS
	A2: Unterschiedliche Arten der Änderung	Der Änderungsbedarf muss sich auf unterschiedliche Modellelemente beziehen können.	L, WS
Methoden, Modelle, Hilfsmittel	A3: Tiefe, Breite und Weite des Wirkkettenmodells	Mit der Methodik müssen Daten auf verschiedenen Detaillierungsebenen (Tiefe), entlang von relevanten Entwicklungsartefakten (Weite) und aus verschiedenen Disziplinen (Breite) abgebildet werden.	L, WS, I
	A4: Änderungsausbreitung	Die Wirkkettenanalyse muss spezifische Propagationspfade analysieren.	L, WS, I
	A5: Objektive Bewertung	Die Auswirkungen der technischen Änderung müssen objektiv, qualitativ und quantitativ bewertet werden können.	L, WS
	A6: Prozessuale Verankerung	Die Wirkkettenanalyse muss in die unternehmensspezifischen Entwicklungsprozesse integriert werden können.	L, WS
Anwendbarkeit	A7: Akzeptabler zeitlicher Anwendungsaufwand	Der zeitliche Modellierungs- und Anwendungsaufwand der Methodik muss in der industriellen Praxis akzeptabel sein	L
	A8: Einfache Anwendbarkeit	Die Methodik muss so gestaltet sein, dass industrielle Anwender die Anwendbarkeit als einfach bewerten.	L, I
	A9: Kollaborative Anwendbarkeit	Die Datenbereitstellung und -auswertung muss kollaborativ möglich sein.	L, I
Ergebnisse	A10: Maßgeschneiderte Ergebnisse (Tailoring)	Die Ergebnisse der Methodik müssen an die rollenspezifischen Bedarfe angepasst werden können.	L, WS
	A11: Ergebnisqualität	Die Ergebnisse der Methodik müssen die qualitativen Ansprüche der industriellen Praxis erfüllen.	L, I
	A12: Nachvollziehbarkeit	Die Ergebnisse der Methodik müssen für Anwender nachvollziehbar sein.	L

4 Konzeption der Methodik zur modellbasierten Wirkkettenanalyse

Ausgehend von den Anwendungsfällen und Anforderungen (Kapitel 3) wird ein Konzept der Methodik zur modellbasierten Wirkkettenanalyse entwickelt. Ziel der Konzeption ist die Definition notwendiger Bestandteile modellbasierter Wirkkettenanalysen, um eine ganzheitliche Methodik zu entwickeln, die Konzepte der MBSE-Modellierung synergetisch mit Konzepten der Wirkkettenanalyse verknüpft und Entwickler bei der frühzeitigen, effizienten und effektiven Bewertung von Änderungsauswirkungen unterstützt.

Für die Entwicklung des Konzepts wurden neben den Erkenntnissen aus dem Stand der Forschung auch Erkenntnisse aus geplanten, konzipierten und durchgeführten Workshops in vier Forschungsprojekten im Anwendungszusammenhang (siehe WS in Anhang A1, S. 139) abgeleitet: einem dreijährigen Verbundforschungsprojekt für modellbasierte Auswirkungsanalysen (ImPaKT), bilateralen Forschungsprojekten zur Wirkkettenmodellierung bei einem Automobilhersteller (Automobil-OEM) und einem Automobilzulieferer (Paragon) sowie einem bilateralen Forschungsprojekt zur Sicherstellung der Modellqualität (BHCT).

Die Konzeption wird entlang der drei Kernelemente des Systems Engineerings [GO22b, S. 16f.] nach GRÄBLER ET AL. strukturiert: Durch das Systemdenken wird ein **konzeptionelles Architekturmodell** für modellbasierte Wirkkettenanalysen (Kapitel 4.1) hergeleitet. Anschließend werden existierende Entwicklungsmethodiken untersucht, um ein **Metadatenmodell** für modellbasierte Wirkkettenanalysen zu definieren, welches die zu analysierenden Daten durchgängig abbildet (Kapitel 4.2). Abschließend werden die Rollen des Systems Engineers in einem **MBSE Rollenmodell** konkretisiert, die durch die Wirkkettenanalyse rollenspezifische Ergebnisse über die Änderungsauswirkungen erhalten (Kapitel 4.3).

4.1 Konzeptionelles Architekturmodell

In existierenden Lösungsansätzen für EC-Auswirkungsanalysen werden unterschiedliche Vorgehensweisen beschrieben. In der Literaturrecherche nach MACHI und MCEVOY [MM12] wurden 58 Ansätze für EC-Auswirkungsanalysen identifiziert (siehe Kapitel 2.2.1) und hinsichtlich des Datenmodells kategorisiert. Für die Analyse werden Steckbriefe der 15 selektierten Ansätze aus Kapitel 2.2.1 erstellt (siehe Anhang A2.4, S. 152). Die Ansätze wurden vom Verfasser analysiert, um relevante Aspekte einer modellbasierten Wirkkettenanalyse herzuleiten. Die Herleitung und Selektion der relevanten Aspekte wird in einer Vorveröffentlichung innerhalb der *Proceedings of the 32nd Symposium Design for X* detailliert beschrieben und als „wiederkehrende Elemente“ bezeichnet [GW21, 6]. Im Folgenden werden die Aspekte mit ihren charakteristischen Merkmalen zu einem Grobkonzept der zu entwickelnden Methodik zusammengeführt und in Bild 4-1 dargestellt.

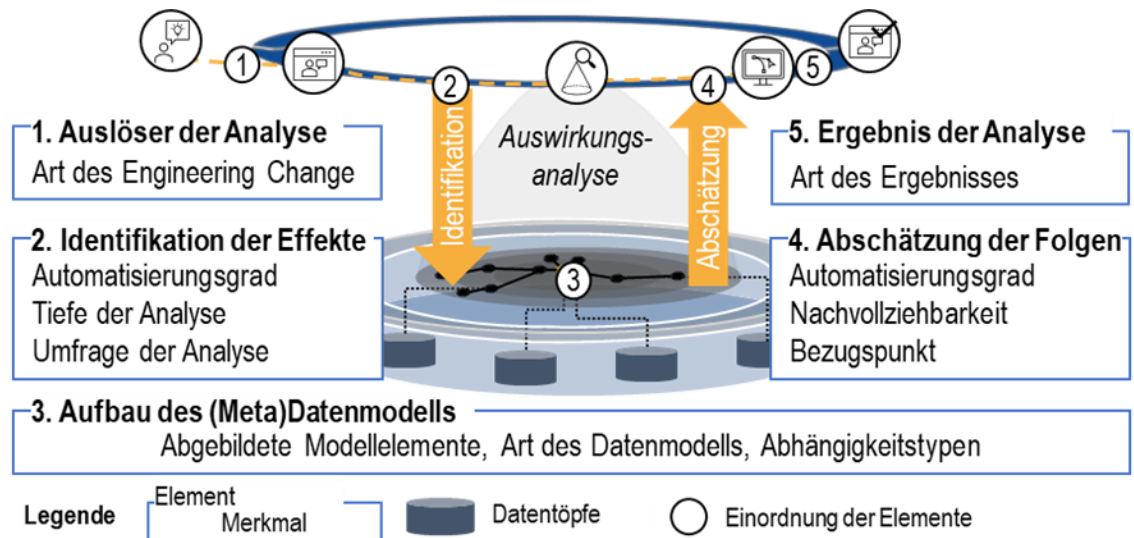


Bild 4-1: Grobkonzept von Wirkkettenanalysen [GW21, S. 6]

Die fünf Elemente des Grobkonzepts wurden vom Verfasser wie folgt definiert:

Die Heterogenität von Änderungsursachen führt dazu, dass unterschiedliche Auslöser zu einer Wirkkettenanalyse führen können. Der **Auslöser der Analyse** wird charakterisiert durch die Art des ECs mit den Attributen Quelle, Richtung, Ebene, Formalität und Zeitpunkt.

Die Wirkkettenanalyse beginnt mit der **Identifikation der Effekte** eines ECs. Die Identifikation der Effekte unterscheidet sich durch den Automatisierungsgrad (nicht-automatisiert, teil-automatisiert, voll-automatisiert), die Tiefe der Analyse (initiale Effekte, konsekutive Effekte, kollektive Effekte) und den Umfang der Analyse (Art und Anzahl der Abhängigkeiten und Systemelemente).

Die Wirkkettenanalyse operiert auf den durch ein Metadatenmodell zusammengeführten Entwicklungsdaten eines Systems. Das **Metadatenmodell** wird durch die abgebildeten Modellelemente (Anzahl und Art), die Art des Datenmodells (matrizenbasiert, objektorientierte Modellierung, graphenbasiert angelehnt an [Kre10]) sowie die verwendeten Abhängigkeitstypen (Richtung und Gewichtung) charakterisiert.

Neben der Identifikation der Effekte müssen die Folgen einer technischen Änderung objektiv abgeschätzt werden. Die **Abschätzung der Folgen** wird durch die Merkmale des Automatisierungsgrads (nicht-automatisiert, teil-automatisiert, voll-automatisiert), der Nachvollziehbarkeit (nicht nachvollziehbar, teilweise nachvollziehbar, nachvollziehbar) und des Bezugspunkts der Abschätzung (Qualität, Kosten, Zeit) charakterisiert.

Wirkkettenanalysen müssen die erwarteten Ergebnisse für eine Rolle folgern. Je nach Rolle werden unterschiedliche Ergebnisse benötigt, um die technische Änderung zu bewerten. Das **Ergebnis der Analyse** kann anhand des Merkmals der Art des Ergebnisses (qualitatives Ergebnis, quantitatives Ergebnis) unterschieden werden, welches der Anwender zur Bewertung der Auswirkungen nutzen soll.

Ausgehend von den identifizierten Elementen wird das konzeptionelle Architekturmodell von modellbasierten Wirkkettenanalysen entwickelt. Hierzu werden die Elemente für die modellbasierte Wirkkettenanalyse ausgeprägt und innerhalb einer adaptierten Architekturbeschreibung nach ISO 42010 [ISO11b, S. 11] verankert (siehe Kapitel 2.1.4). Das konzeptionelle Modell der Architekturbeschreibung bildet die Struktur des ganzheitlichen Lösungsansatz entlang von drei Ebenen:

Prozessuale Ebene: Für das **SoI** müssen Wirkkettenanalysen durchgeführt werden, um **Stakeholder** bei der Bewertung der Änderungsauswirkungen zu unterstützen. Jeder **Auslöser** (Element 1) führt zur Bewertung von separaten Propagationspfaden. Die Attribute eines ECs bestimmen die Ausgestaltung der Wirkkettenanalyse. Die Heterogenität möglicher Auslöser technischer Änderungen bestimmt die Vielzahl von Propagationspfaden und die damit einhergehende Notwendigkeit maßgeschneiderter Wirkkettenanalysen für verschiedene Rollen umzusetzen. Die **erwarteten Ergebnisse** (Element 5) der Wirkkettenanalyse sind abhängig von den Rollen und Perspektiven der Stakeholder. Jeder **Stakeholder** hat ein bestimmtes Interesse an dem SoI und nimmt eine oder mehrere **Rollen** bei der Analyse der Auswirkungen ein. Zur Bewertung der Auswirkungen benötigt der Stakeholder ein oder mehrere **Ergebnisse** der Wirkkettenanalyse, welche in einer rollenspezifischen **Ergebnissicht** zusammengefasst werden. Die Rolle bestimmt die **Perspektive** des Stakeholders und muss bei der Entwicklung der Ergebnissicht berücksichtigt werden. Zur Herleitung der Ergebnissichten werden die erwarteten Ergebnisse in einen rechnerinterpretierbaren **Wirkpfad** transformiert (siehe Bild 4-2). Hierzu werden die Konzepte der Wirkkettenmodellierung (siehe Kapitel 2.1.5) und die Elemente eines TIMs nach REMPEL und MÄDER [RM15, S. 177] (siehe Kapitel 2.1.4) auf Propagationspfade übertragen. Der Wirkpfad ist die Abbildung eines Ursache-Wirkungs-Musters [SMS12, S. 292] in einer definierten Reihenfolge an **Artefakttypen** und **Relationstypen**.

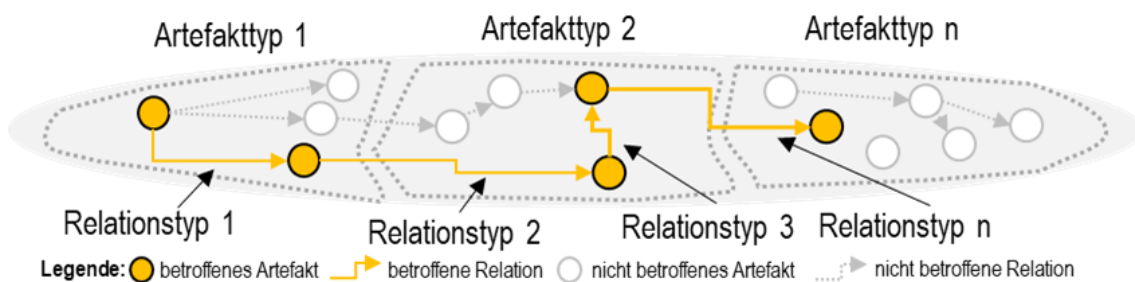


Bild 4-2: Darstellung eines Wirkpfads zur Abbildung eines Wirkmusters

Analyse-Ebene: Ein Wirkpfad umfasst kollektive Propagationseffekte entlang definierter Ursache-Wirkungs-Beziehungen und wird in einer **Wirkkettenanalyse** automatisiert und IT-gestützt ausgeführt. Die Wirkkettenanalyse besteht aus zwei Teilen: der Identifikation betroffener Elemente (Element 2) und der Bewertung der Auswirkungen (Element 4). Ausgehend von dem Auslöser wird ein fest definierter Wirkpfad analysiert, um betroffene Elemente zu identifizieren. Zur Bewertung der Auswirkungen werden Bewertungskriterien in die Wirkkettenanalyse integriert.

Modellebene: Die Wirkkettenanalyse operiert auf der Basis des **Wirkkettenmodells**, welches alle notwendigen Wirkpfade beinhalten muss. Die erwarteten Ergebnisse sind daher frühzeitig zu bestimmen, um die Wirkkettenmodelle an die erwarteten Ergebnisse anzupassen und maßgeschneiderte Wirkkettenanalysen zu befähigen. Die Erweiterbarkeit von objektorientierten und graphenbasierten Modellierungsansätzen wird genutzt, um maßgeschneiderte Wirkkettenmodelle herzuleiten. Zur Erstellung des Wirkkettenmodells werden existierende Entwicklungsdaten verwendet, die in einem oder mehreren Datenmodellen beschrieben werden. **Datenmodelle** beschreiben hierbei den syntaktischen Aufbau der Entwicklungsdaten und werden auf semantischer Ebene durch ein Metadatenmodell zusammengeführt. Im **Metadatenmodell** (Element 3) werden die benötigten Artefakt- und Relationstypen abgebildet. In dem Metadatenmodell wird definiert, welche Artefakt- und Relationstypen für die Wirkkettenmodellierung verwendet werden und wie die Datenmodelle interpretiert werden. Die Konzepte und Abhängigkeiten werden in einem konzeptionellen Modell der modellbasierten Wirkkettenanalyse dargestellt. (siehe Bild 4-3) Die hergeleiteten Elemente werden für die modellbasierte Wirkkettenanalyse ausgeprägt und durch architektonische Elemente in einem neuen und ganzheitlichen Architekturmodell ergänzt. Das ganzheitliche Architekturmodell verbindet die prozessuale Ebene, die Modellierungsebene und die Analyseebene und vereint Konzepte der MBSE-Modellierung mit Konzepten der Wirkkettenanalyse.

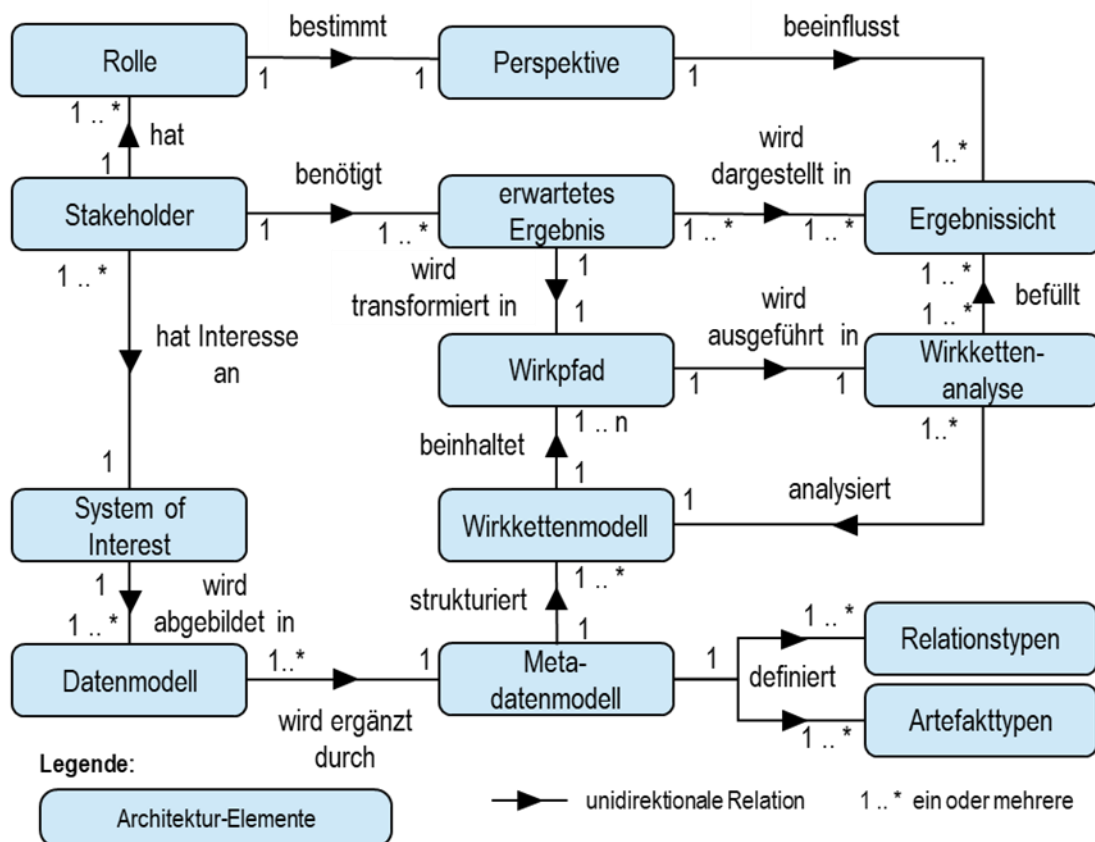


Bild 4-3: Konzeptionelles Architekturmodell für eine ganzheitliche modellbasierter Wirkkettenanalyse (aufbauend auf [ISO11b, S. 11])

4.2 Metadatenmodell

Das Metadatenmodell der modellbasierten Wirkkettenanalyse (Element 3) muss allgemeingültige Entwicklungsaktivitäten wie die Strukturierung von Anforderungen [BG21a, S. 200f.] oder das Aufgliedern einer Gesamtfunktion in Teilfunktionen [GBP21, S. 243ff.] abbilden. Neben der aktivitäts-orientierten Perspektive ist ebenfalls die artefakt-orientierte Perspektive bedeutsam, da die Systemmodelle und spezifischen Modelle aus Modellelementen und Relationen bestehen. Um Aspekte beider Perspektiven im Metadatenmodell zu berücksichtigen, wurden aktivitätsorientierte Entwicklungsmethodiken und artefakt-orientierte Modellierungsansätze abgeglichen. Die Ergebnisse des analytischen Abgleichs wurden auf dem *IEEE International Symposium on Systems Engineering 2022* vom Verfasser vorgestellt [GWO22] und vorveröffentlicht.

Zur Entwicklung des Metadatenmodells wird der weitverbreitete artefakt-orientierte RFLP-Ansatz [GWO22, S. 1, KK13, S. 93, ZBL14, S. 247, Eig21, S. 73] (siehe Kapitel 2.2.1) mit der aktivitäts-orientierten Perspektive des V-Modells abgeglichen. Im Vergleich zu existierenden Abgleichen [HST23, S. 118, Eig21, R82], wurde das V-Modell der VDI/VDE 2206:2021 [VDI21] verwendet. Gründe für die Auswahl der Richtlinie VDI/VDE 2206:2021 sind die Auslegung der Entwicklungsmethodik auf mechatronische und cyber-physische Systeme (CPS), der klare verständliche Aufbau entlang der Kernaufgaben und Kontrollpunkte sowie die weitverbreitete Art der Visualisierung entlang des V-Modells. Die Ergebnisse des Abgleichs wurden in einer Vorveröffentlichung auf dem *IEEE International Symposium on Systems Engineering 2022* vom Verfasser vorgestellt und veröffentlicht [GWO22].

Aufbauend auf den Elementen eines Traceability Information Models nach REMPEL und MÄDER [RM15, S. 177] (Kapitel 2.1.4) wird der Richtlinien text analysiert, um relevante Artefakttypen zu durchgängigen Abbildungen der Produktentwicklung zu identifizieren und als zentrale Entwicklungsartefakte für modellbasierte Wirkkettenanalysen festzulegen. Die identifizierten Artefakttypen der VDI/VDE Richtlinie 2206:2021 werden in Bild 4-4 dargestellt. Die fünf Kernaufgaben für die Entwicklung mechatronischer und cyber-physischer Systeme sind nach VDI/VDE 2206:2021 die *Anforderungserhebung*, die *Systemarchitektur*, die *Implementierung der Systemelemente*, die *Systemintegration und Verifikation* und die *Validierung und Übergabe*. Eingangsdaten für den Entwicklungsprozess sind die Restriktionen aus nachgelagerten Entwicklungsphasen, das Geschäftsmodell und der Entwicklungsauftrag. Während der Anforderungserhebung werden Kundenbedürfnisse zu Kundenanforderungen und anschließend zu Systemanforderungen transformiert. Die Kunden- und Systemanforderungen werden in der Spezifikation zusammengetragen und dokumentiert. Die anschließende Entwicklung der Systemarchitektur umfasst lösungsneutrale Funktionen und logische Elemente. Durch die Auswahl mehrerer logischer Elemente werden Architekturalternativen entwickelt, die die Funktionen erfüllen. Die Architekturalternativen werden gegenübergestellt und bewertet. Innerhalb der disziplinspezifischen Implementierung der Systemelemente werden die logischen Elemente zu physischen Elementen und zugehörigen Schnittstellen

transformiert. Bei der anschließenden Systemintegration wird die Verifikation auf verschiedenen Systemebenen durchgeführt. Bei der abschließenden Validierung wird die Erfüllung der Kundenbedürfnisse überprüft, um die Eigenschaften abzusichern und die erfolgreiche Übergabe des Systems zu gewährleisten. Je nach den Entwicklungsartefakten können Parameter ergänzt werden, da Komponenten durch Parameter spezifiziert werden und Ursache-Wirkungs-Beziehungen zwischen diesen bestehen [SMS12, S. 292, UTY15, S. 1683].

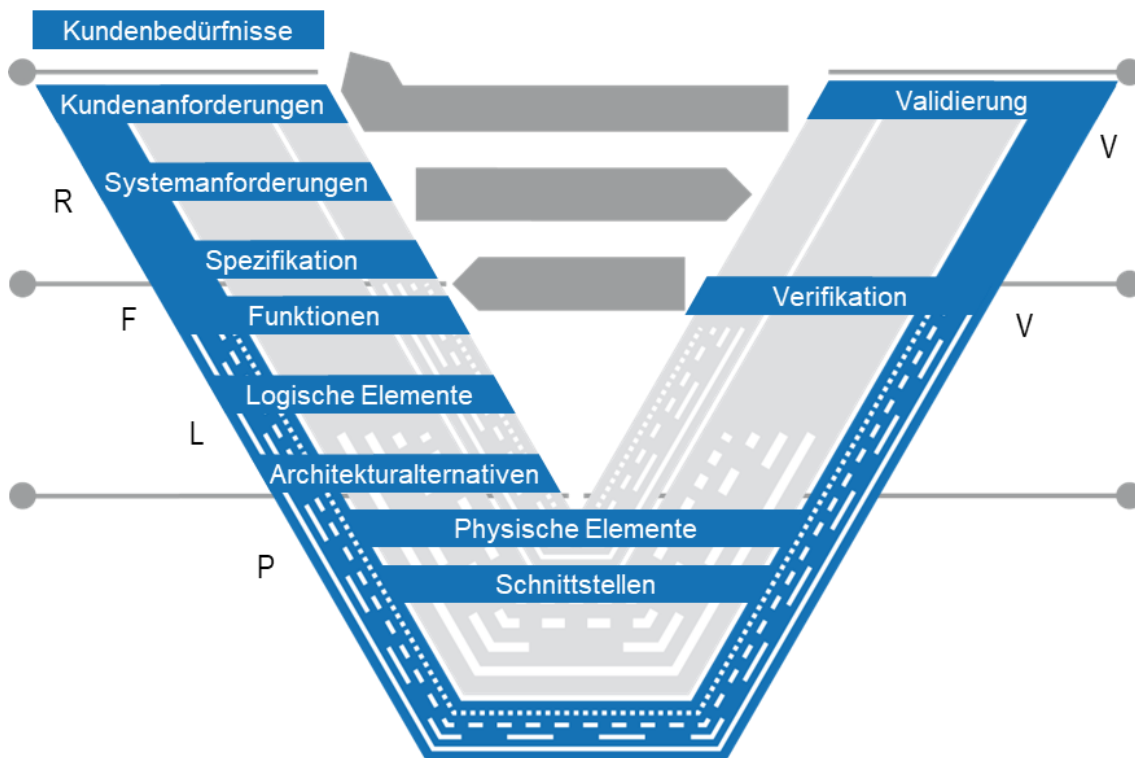


Bild 4-4: Identifizierte Entwicklungsartefakte (Eigene Darstellung mit Bezug zum V-Modell der VDI/VDE 2206:2021 [VDI21, S. 22])

Der Abgleich des RFLP Ansatzes mit der VDI/VDE Richtlinie 2206:2021 [VDI21, S. 22] verdeutlicht, dass alle Artefakte Ergebnisse der ersten drei Kernaktivitäten sind. Die Verifikation und die Validierung sind implizite Bestandteile des RFLP Ansatzes, weshalb die RFLP Artefakte um die Verifikation (RFLPV) [GWO22, S. 1] und die Validierung (RFLPV²) [GO22a, S. 71] erweitert werden.

Ausgehend von den identifizierten Entwicklungsartefakten werden neun Relationstypen zwischen den RFLPV²-Artefakten entwickelt (siehe Bild 4-5), die die Ursache-Wirkungs-Beziehungen zwischen den Artefakten darstellen. Hierzu werden die Relationstypen nach TORIELLO [Tor22, S. 35] um vier Relationstypen erweitert: derives, requires, specifies und realizes. Aufbauend auf den Unterscheidungen in Intra- und Interrelationen nach BAJAJ ET AL. [BBW17, S. 154f.] sowie homogene und heterogene Relationen nach SCHMIDT [Sch89, S. 5] werden die Relationen in drei Kategorien unterteilt: vertikale, horizontale und diagonale Relationen. Homogene Artefakte können horizontal

und vertikal vernetzt werden. **Vertikale Relationen** sind hierarchische Abhängigkeiten zwischen zwei Elementen eines homogenen Artefakttyps. **Horizontale Relationen** sind Abhängigkeiten zwischen zwei Elementen eines homogenen Artefakttyps auf derselben hierarchischen Ebene. **Diagonale Relationen** treten zwischen Elementen mit heterogenen Artefakttypen auf.

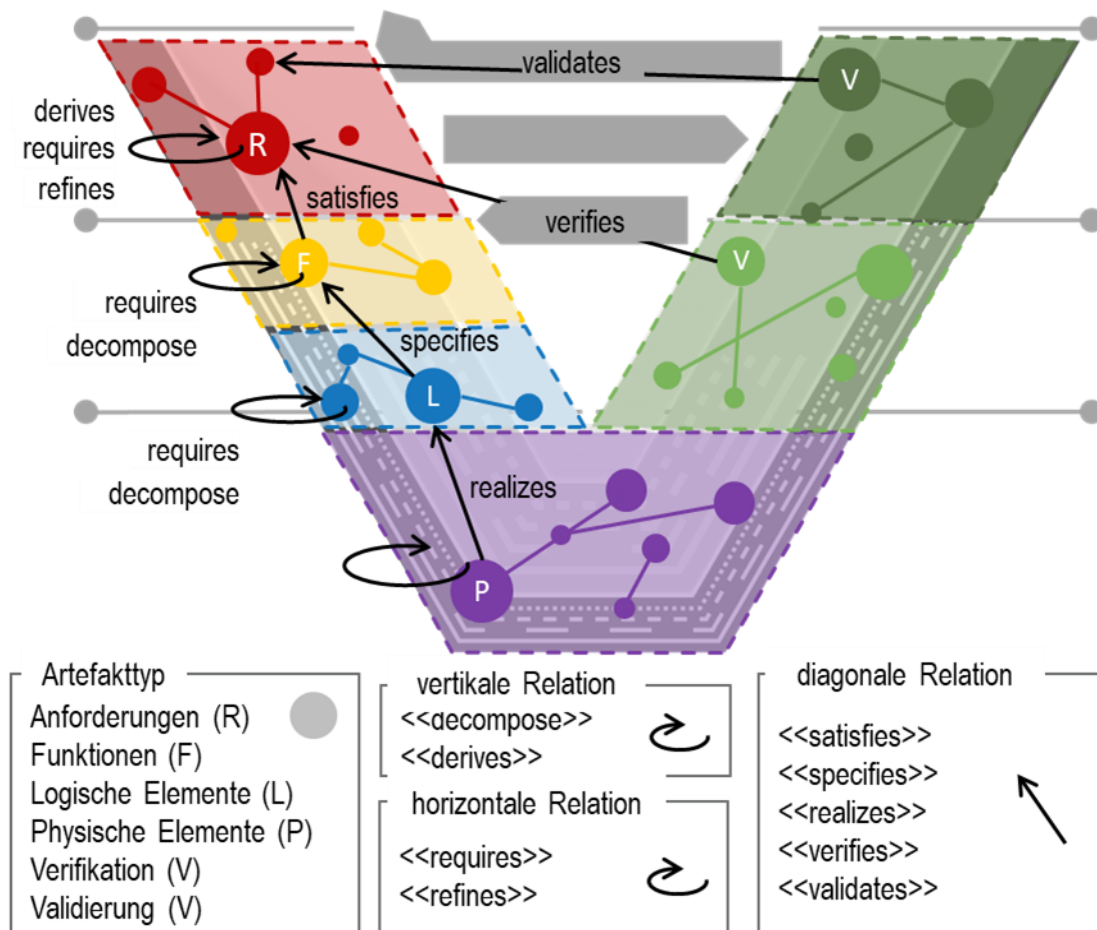


Bild 4-5: Konzipierung der Datenstruktur entlang von RFLPV² und V-Modell

Eine beispielhafte vertikale Relation für das Fallbeispiel des Dachantriebs sind die hierarchische Unterteilung von Anforderungen [Rup21, S. 453], die Unterteilung einer Gesamtfunktion und Teilfunktionen [BG21b, S. 246] und die strukturelle Unterteilung eines Systems in Subsysteme und Systemelemente [HWF19, S. 4]. Horizontale Relationen treten beispielsweise bei sich gegenseitig bedingenden Anforderungen, Energie-, Stoff-, und Informationsflüssen zwischen Teilfunktionen oder physikalischen Zusammenhängen einzelner Systemelemente auf. Beispielhafte diagonale Relationen sind die Verifikation einer Anforderung an den Dachantrieb durch einen Testfall, die Spezifizierung einer Funktion durch ein logisches Element oder die Realisierung eines logischen Elements durch ein physisches Element. Bei der Festlegung der Relationen werden die Best Practices nach TORIELLO [Tor22, S. 37] angewendet, indem eine klare und konsistente Richtung der unidirektionalen Relationen vorgegeben wird. Die gerichteten Relationen können für die Wirkkettenanalyse ebenfalls gewichtet werden,

beispielsweise unterschieden in zwanghafte Propagation und mögliche Propagation (in Anlehnung an [OLE22, S. 61]). Das Ergebnis ist ein Metamodell für die modellbasierte Wirkkettenanalyse, bestehend aus sechs grundlegenden Entwicklungsartefakte und neun Relationen. Diese grundlegenden Entwicklungsartefakte und Relationen bilden eine skalierbare und fundierte Ausgangslage, um Auswirkungen technischer Änderungen in der Entwicklung zu analysieren und zu bewerten. Das Metadatenmodell folgt dem Gedanken eines minimalen Informationsmodells, welches nur die benötigten Informationen beinhaltet, um definierte Aufgaben in bestimmten Entwicklungsphasen zu adressieren [MHH17, S. 1]. Ausgehend von dem Metadatenmodell können maßgeschneiderte Metadatenmodelle für Anwendungsfälle hergeleitet werden [GRW24, S. 1].

4.3 MBSE-Rollenmodell

Durch die Wirkkettenanalysen müssen individuelle Ergebnisse für beteiligte Stakeholder gefolgert werden, die an ihre rollenspezifische Sicht angepasst werden. Zur Berücksichtigung der Bedarfe werden MBSE Rollen identifiziert, die die Wirkkettenanalyse initiieren (Element 1) und die Analyseergebnisse erhalten (Element 2). Das MBSE-Rollenmodell beruht auf der Analyse von existierenden Rollenmodellen aus den Bereichen SE, Organisationsmanagement und Projektmanagement und wurde auf der *Konferenz zur Antriebstechnik (ATK 2021)* vorgestellt und vorveröffentlicht [GWP21]. Die analysierten Rollen umfassen sowohl die zwölf Systems Engineering Rollen nach SHEARD [She96] als auch die Rollen des Role-Model for Systems Engineering Application (RM-SEA) nach GRÄBLER ET AL. [GOH19]. Aufbauend auf der Analyse und dem Modell der RM-SEA [GOH19] wurden zwei MBSE-Rollen ergänzt und eine Rolle angepasst (siehe Bild 4-6).

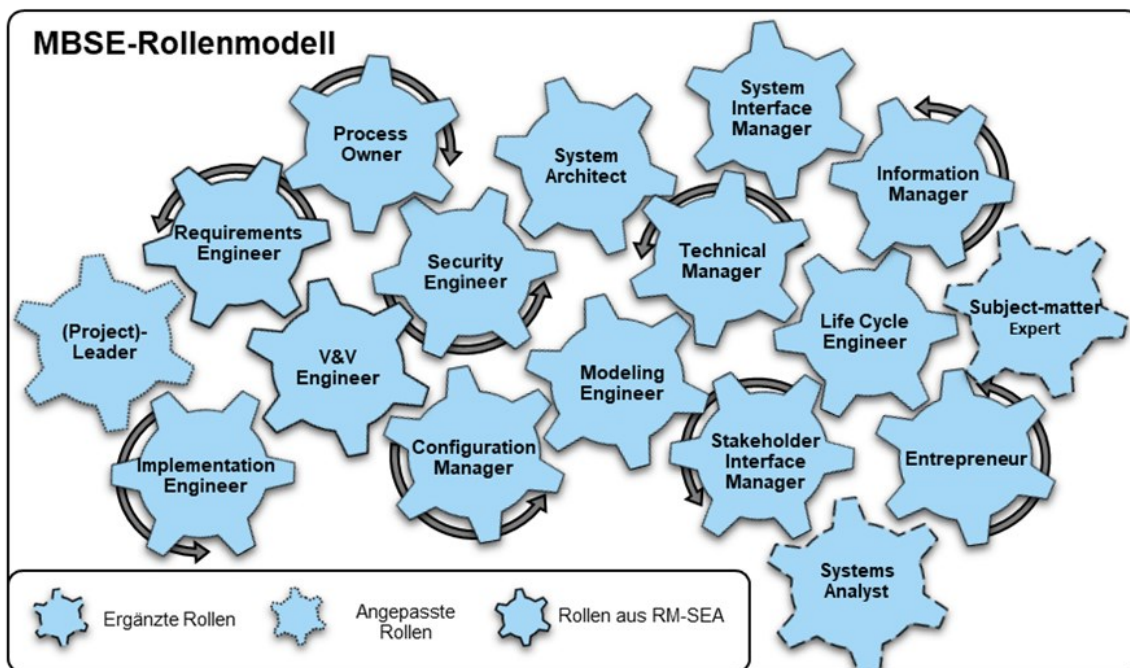


Bild 4-6: MBSE-Rollenmodell [GWP21, S. 8], aufbauend auf RM-SEA [GOH19]

5 Entwicklung der Methodik zur modellbasierten Wirkkettenanalyse technischer Änderungen

Aufbauend auf der Konzeption (Kapitel 4) wird in diesem Kapitel die resultierende Methodik zur modellbasierten Wirkkettenanalyse (MECA Model-based Effect Chain Analysis) beschrieben. Ausgangspunkt für die Entwicklung der Methodik ist das Konzept (siehe Kapitel 4), welches durch die Methodik operativ anwendbar wird. Die Methodik setzt sich aus Modellen, Methoden und Hilfsmitteln zusammen (siehe Kapitel 2.1.3). Basierend auf den identifizierten Elementen (siehe Kapitel 4.1) werden vier methodische Schritte festgelegt, die in Bild 5-1 verortet werden.

Schritt 1: Zieldefinition der Wirkkettenanalyse

Durch eine frühzeitige Zieldefinition der Wirkkettenanalysen werden zu analysierende Wirkpfade für unterschiedliche Auslöser technischer Änderungen (Element 1) definiert und notwendige Ergebnissichten (Element 5) abgeleitet.

Schritt 2: Ermittlung von Informationen

Die notwendigen Daten und Informationen für die Wirkkettenanalyse werden frühzeitig und zielgerichtet in Abhängigkeit von den zu analysierenden Wirkpfaden erfasst und zusammengetragen.

Schritt 3: Modellierung der Wirkketten

Bei der anschließenden Modellierung der Wirkketten wird das konzipierte Metadatenmodell erweitert, um alle benötigten Wirkpfade abzubilden und Limitationen disziplinspezifischer Modellierungen aufzuheben.

Schritt 4: Durchführung der Wirkkettenanalyse

Zur Durchführung der Wirkkettenanalyse werden Abfragen zur Analyse des Wirkpfads entwickelt und durch integrierte objektive Bewertungskriterien ergänzt. Durch die Ergebnisdarstellung in rollenspezifischen Sichten unterstützt die Wirkkettenanalyse das subjektive Expertenwissen der Anwender.

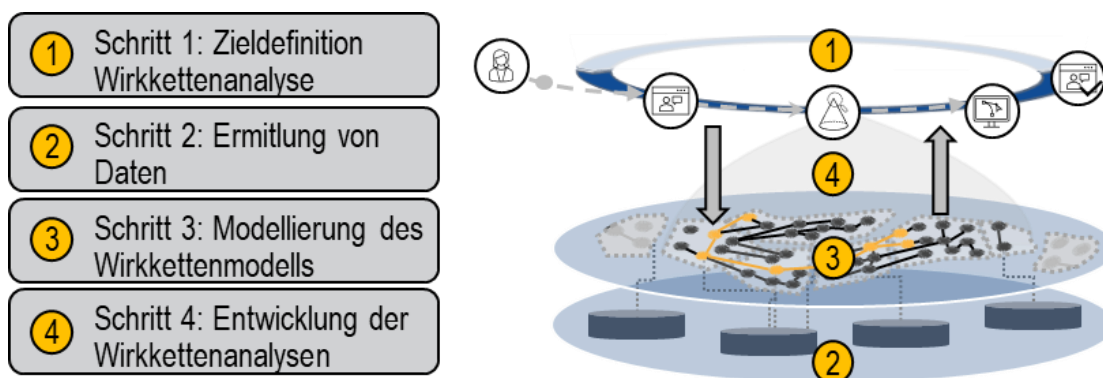


Bild 5-1: Einordnung der methodischen Schritte in das Grobkonzept

Die grundlegenden vier Schritte der Methode wurden den *Proceedings of the Design Society 2022* [GWK22] vorveröffentlicht und auf Basis aus Erfahrungen aus dem Anwendungszusammenhang weiterentwickelt. Jeder Schritt enthält konkrete Aktivitäten, die für die Entwicklung der Wirkkettenanalysen durchgeführt werden (siehe Bild 5-2). Jede Aktivität wurde vom Verfasser hergeleitet, ausgestaltet und auf Basis von Erfahrungen aus dem industriellen Anwendungszusammenhang (ImPaKT, Automobil-OEM, Paragon) verfeinert. Initiale Aktivitäten der Methodik wurden im *MDPI Journal Systems* [GWK23] vorveröffentlicht und anschließend weiterentwickelt. Die MECA-Methodik wird software-gestützt implementiert (Kapitel 5.5) und im Anwendungszusammenhang getestet, um Limitation in der Praxis zu reduzieren. Die prozessuale Integration und kollaborative Anwendbarkeit wird durch die Entwicklung eines Referenzprozesses sichergestellt (Kapitel 5.6). Durch die MECA-Methodik werden drei Strategien zum Umgang mit technischen Änderungen [FGN00, S. 173] adressiert: die effektive Abschätzung von Auswirkungen, der effiziente Einsatz von Ressourcen zur Handhabung der ECRs und die Vorverlagerung technischer Änderungen in frühe Entwicklungsphasen. Im Folgenden wird die MECA-Methodik anhand eines beispielhaften ECRs erklärt – einer Änderung der Griffweite eines kollaborativen 6-Achsen Roboterarms während einer Pick-and-Place Bewegung zum Greifen der Karosserie eines Remote Control Cars (RC-Cars).

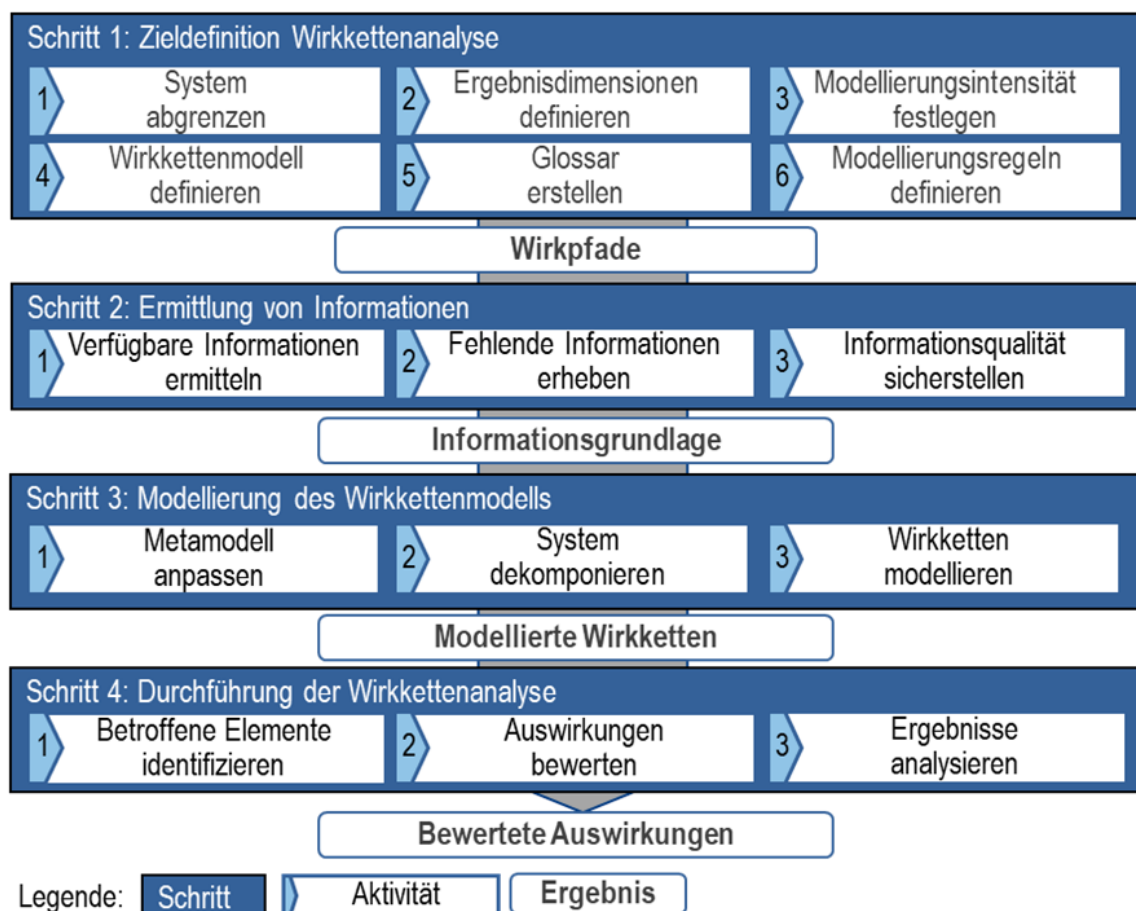


Bild 5-2: Überblick über die methodischen Schritte der MECA-Methodik

5.1 Schritt 1: Zieldefinition der Wirkkettenanalyse

Im ersten Schritt der MECA-Methodik werden die Ziele der Wirkkettenanalyse systematisch von den Anwendern festgelegt. Hierzu werden sechs Aktivitäten durchgeführt: drei für die Analyse des Ziels und drei für die Synthese (siehe Bild 5-3). Das Fundament zur Definition der Aktivitäten beruht auf Bestandteilen des konzeptionellen Modells (siehe Kapitel 4.1) und folgt dem Denkmuster, dass Stakeholder das Problem frühzeitig beschreiben und Fragestellungen explizieren müssen [PDH11, S. 292]. Durch die Problembeschreibung wird der Zweck der Modellierung aus Sicht der Stakeholder definiert [Fri14, S. 22]. Die Durchführung der Aktivitäten erfolgt durch die MBSE-Rollen (siehe Kapitel 4.3), die die Auswirkungen des ECRs bewerten.

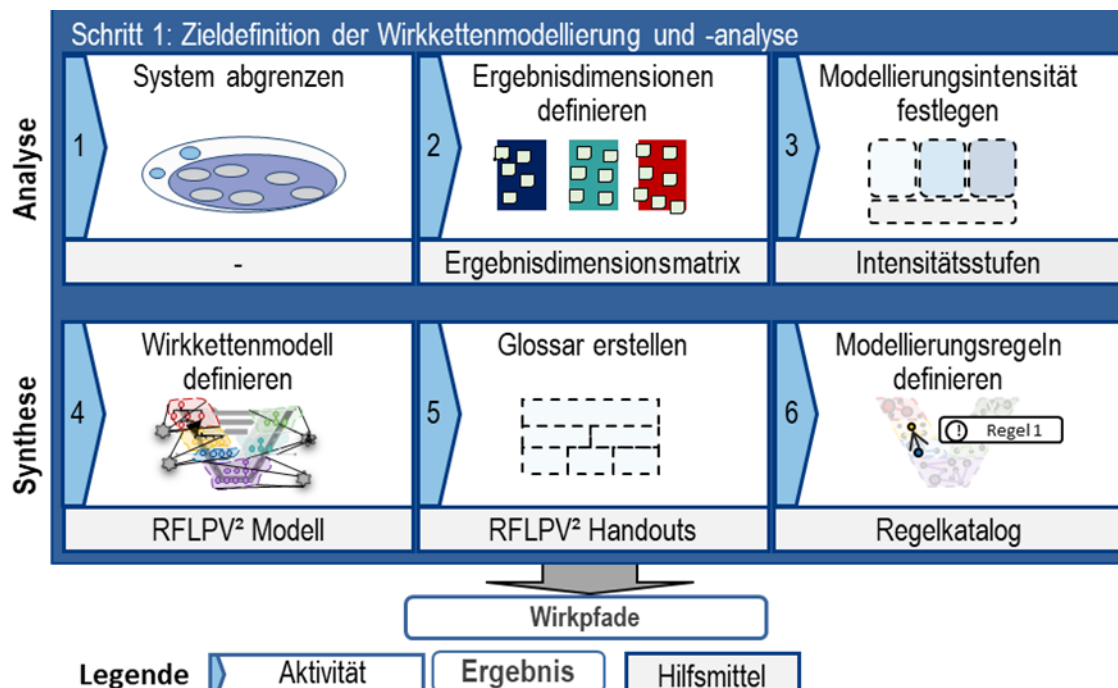


Bild 5-3: Detaillierte Darstellung von Schritt 1 der MECA-Methodik

In der initialen Analyseaktivität *System abgrenzen* wird die erste Denkdimension des Systemdenkens nach GRÄBLER operationalisiert [GO22d, S. 94]. Das betrachtete SoI wird von Umfeldsystemen und Einflussfaktoren abgegrenzt, um den Betrachtungsbereich der Wirkkettenanalyse festzulegen [HWF19, S. 178]. Der Roboterarm ist ein System im Smart Automation Laboratory (SAL) des Lehrstuhls für Produktentstehung und besteht aus verschiedenen Subsystemen wie der Basis, dem Unter- und Oberarm und dem Greifersystem zur Durchführung der Pick-and-Place Bewegung. Bei der Pick-and-Place-Bewegung wird die 3D-gedruckte Karosserie eines Remote Control Cars (RC-Cars) gegriffen und zur Beförderung in ein Shuttle des Schienensystems des SAL gelegt. Daher werden das Shuttle und die Karosserie des RC-Cars als Umfeldsysteme berücksichtigt. Weitere Maschinen der Produktionslinie im SAL werden vernachlässigt. Eine detaillierte und bildliche Darstellung des Fallbeispiels ist angehängt (siehe Anhang A5.1, S. 202).

Durch die Adaption des Konzepts eines Untersuchungsbereichs nach HABERFELLNER [HWF19, S. 177] werden der Modellierungsaufwand reduziert und die Wirkkettenanalysen maßgeschneidert für das SoI entwickelt. Zusätzlich wird durch die Systemabgrenzung deutlich, welche Stakeholder an der Entwicklung der Wirkkettenanalysen beteiligt sind. Ergebnis der Aktivität *System abgrenzen* ist der abgegrenzte Untersuchungsbereich der Wirkkettenanalysen und die identifizierten Stakeholder und ihrer Rollen (siehe Bild 5-4).

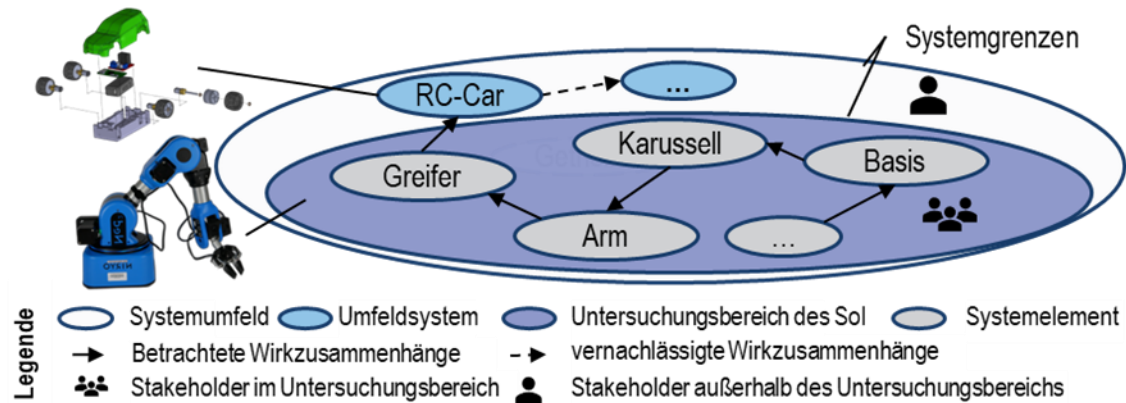


Bild 5-4: Systemabgrenzung in Anlehnung an Haberfellner [HWF19, S. 178]

Jeder Stakeholder betrachtet einen ECR aus einer rollen-spezifischen Perspektive. Ein Entwickler benötigt beispielsweise Informationen über Auswirkungen auf das konstruierte System, wohingegen ein Manager an dem ursächlichen Fehler interessiert ist [DKJ05, 11]. Daher werden zur Bewertung der Änderungsauswirkungen eines ECRs rollen-spezifische Ergebnissichten (Element 5) benötigt. Zur Herleitung der Ergebnissichten werden in der Aktivität *Ergebnisdimension definieren* die betroffenen Stakeholder befragt, um die erwarteten Ergebnisse zu explizieren. Eine **Ergebnisdimension** beschreibt ein erwartetes Ergebnis einer Wirkkettenanalyse aus der Perspektive eines Stakeholders [GW23, S. 3]. Es werden vier Bestandteile einer Ergebnisdimension definiert: eine Fragestellung, einen daraus abgeleiteten Wirkpfad, Bewertungsmetriken und eine Visualisierungsart. Das Vorgehen wird folgend an einem ECRs des Roboterarms erläutert (siehe Bild 5-5 und Anhang A5, S. 202), für welchen drei Fragen aus Sicht beteiligter MBSE-Rollen [GWP21, S. 8] definiert werden:

ECR: „Die Karosserie des RC-Cars muss von 65 mm auf 73 mm verbreitert werden, um die Demontage zu vereinfachen und die Materialzirkularität zu ermöglichen“.

- Frage des Anforderungsentwicklers: „Hat die Anforderungsänderung der Breite der Karosserie Auswirkungen auf weitere Anforderungen?“
- Frage des Systemarchitekten: „Welche sicherheitskritischen Funktionen sind von der initial geänderten Anforderung betroffen?“
- Frage des technischen Managers: „Wie hoch sind die Kosten der betroffenen und auszutauschenden mechanischen Systemelemente des Roboterarms?“

Die Ergebnisdimension überträgt das Goal Question Metrik Paradigma (GQM) nach BASILI [BC94, S. 2] auf technische Änderungen. Das übergeordnete Ziel der Wirkkettenanalyse (*Goal*), wird durch die Beantwortung der stakeholder-spezifischen Fragestellungen in Form von Ergebnisdimensionen (*Question*) erreicht und durch Metriken (*Metrics*) quantifiziert. Anhand der beispielhaften Fragen wird verdeutlicht, wie unterschiedlich die rollenspezifischen **Fragestellungen** und die damit gestellten Erwartungen an die Ergebnisse der Wirkkettenanalyse sind. Aus jeder Frage wird ein **Wirkpfad** abgeleitet. Der Wirkpfad wird durch eine oder mehrere **Bewertungsmetriken** zur Quantifizierung des Ergebnisses ergänzt. Eine beispielhafte Metrik für die Frage des Systemarchitekten ist die Anzahl an betroffenen sicherheitskritischen Funktionen. Je Ergebnisdimension wird abschließend eine **Visualisierungsart** festgelegt. Bei der Festlegung der Visualisierungsart werden Prinzipien zur Darstellung von Daten nach NUSSBAUMER [Nus15] berücksichtigt. Beispielhafte Visualisierungsarten für die Ergebnisse von Wirkkettenanalysen sind Tabellen, Modellausschnitte oder Balkendiagramme. Erfahrungen aus der Anwendung zeigen, dass je Rolle 1-5 Ergebnisdimensionen für einen ECR formuliert werden können.

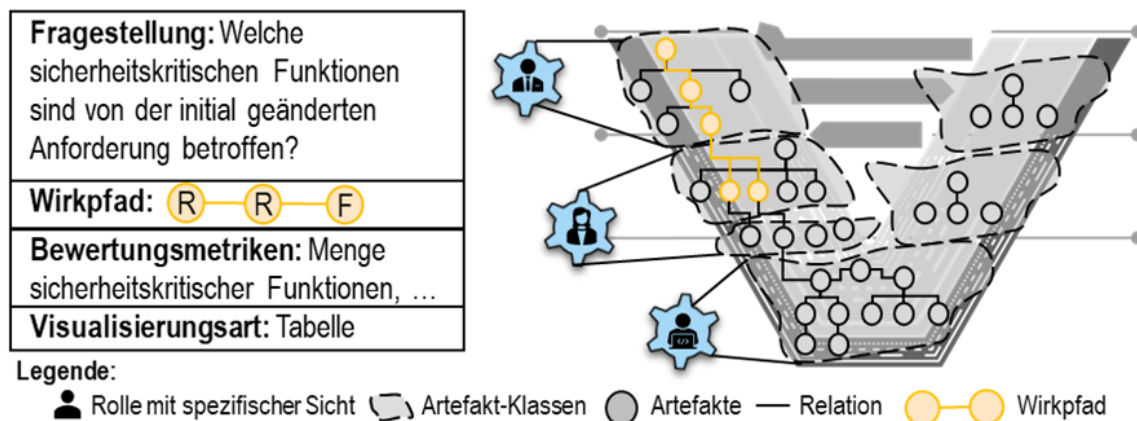


Bild 5-5: Darstellung der Ergebnisdimension

Als Hilfsmittel zur Herleitung von Ergebnisdimensionen wird eine **Ergebnisdimensionsschablone** (EDS) entwickelt. Schablonen sind bewährte Hilfsmittel in der Produktentstehung, beispielsweise bei der Anforderungsformulierung [Rup21, S. 361] oder zur Definition von Prozessen [Rup21, S. 382]. Die EDS unterstützt bei der Festlegung, welche Ergebnisdimension („Was“) mit welchen Bewertungskennzahlen und („Womit“) in welcher Visualisierungsart („Wie“) abgebildet wird. Die EDS wurde in Expertenworkshops mit betroffenen Stakeholdern angewendet, um die praktische Anwendbarkeit zu bestätigen. Ausgehend von den definierten Ergebnisdimensionen wird der Informationsbedarf zur Beantwortung der Fragestellungen abgeleitet. Hierzu wird jede Fragestellung in einen Wirkpfad transformiert, der im konzipierten Metadatenmodell abgebildet werden muss (Kapitel 4.2). Zur Herleitung der Wirkpfade werden drei Regeln definiert:

- Jede Frage einer Ergebnisdimension beschreibt einen individuellen Wirkpfad
- Subjekte in Fragestellungen weisen auf die benötigten Artefakttypen hin
- Verben in Fragestellungen weisen die benötigten Relationstypen hin

In Bild 5-6 wird eine exemplarische ausgefüllte EDS für die Fragestellung des technischen Managers dargestellt.

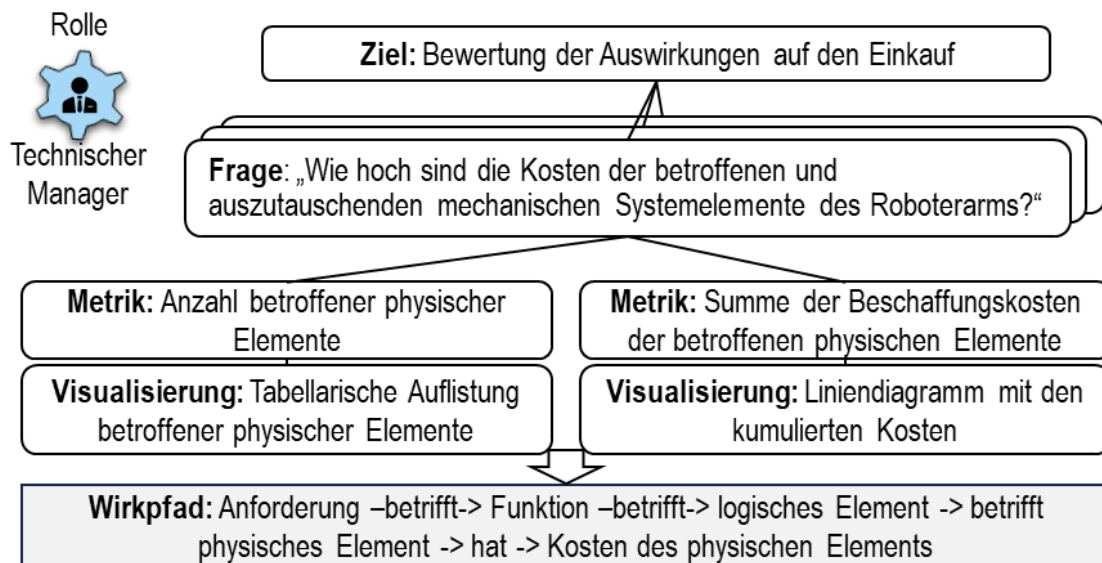


Bild 5-6: Exemplarisch ausgefüllte Ergebnisdimensionsschablone für einen technischen Managers

Das Beispiel aus Bild 5-6 verdeutlicht die Transformation von implizitem Expertenwissen in eine explizite Fragestellung und einen deklarativ beschriebenen Wirkpfad, der in einem Datenmodell abgebildet werden kann (in Anlehnung an [VWZ18, S. 419]). Jede Fragestellung der Ergebnisdimension wird durch Metriken und Visualisierungsarten spezifiziert. Die Fragestellungen, die Metriken und die Visualisierungsarten verdeutlichen die Perspektive des Stakeholders und bestimmen notwendige Sichten auf das Wirkkettenmodell, in dem die Daten zur Herleitung der Ergebnisse der Wirkkettenanalyse abgebildet werden. Die Menge der benötigten Wirkpfade definiert den Informationsbedarf, welcher im Metadatenmodell abgebildet werden muss. Ergebnis der Aktivität *Ergebnisdimensionen definieren* ist eine Sammlung von Ergebnisdimensionen und deren Informationsbedarfen.

Ausgehend von den identifizierten Ergebnisdimensionen und Informationsbedarfen wird die Aktivität *Modellierungsintensität festlegen* durchgeführt. Bei einer hohen Anzahl heterogener ECRs, die durch die Wirkkettenanalyse analysiert werden sollen, entstehen Ergebnisdimensionen, die eine divergierende Tiefe an benötigten Informationen aus dem Wirkkettenmodell benötigen. Erfahrungen aus der MBSE-Modellierung in bilateralen Forschungsprojekten (Paragon, Automobil-OEM) haben gezeigt, dass Variationen in der Modellierungsintensität zu fehlerhaften und inkonsistenten Modellen führen. Zur Sicherstellung eines fehlerfreien und konsistenten Wirkkettenmodells wird vor der Modellierung eine einheitliche Modellierungsintensität festgelegt. Die Modellierungsintensität beschreibt, wie detailliert ein Effekt und seine Wechselwirkungen in dem Wirkkettenmodell abgebildet werden (in Anlehnung an LOCHBICHLER [Loc20, S. 16]). Die Festlegung der

minimal notwendigen Modellierungstiefe reduziert den Aufwand für die Modellierung und die nachfolgende Analyse, welche eng mit der Modellierungsintensität des Modells verbunden ist [VWZ18, S. 177].

Als Hilfsmittel zur Festlegung der Modellierungsintensität wurde ein Modell mit drei Intensitätsstufen entwickelt und in den Modellierungworkshops für Wirkkettenanalysen mit Anwendern im ImPaKT Projekt angewendet (in WS 5-7). Zusätzlich wurden die Erkenntnisse nach TORIELLO aufgegriffen, nach dem Artefakte auf möglichst tiefen Granularitätsniveaus bei gleichzeitig konstanten Abstraktionsniveaus zu vernetzen sind [Tor22, S. 37]. Die drei Intensitätsstufen unterscheiden sich in Hinblick auf den Umfang an benötigten Informationen je Artefakttyp und die Anzahl von Relationstypen je Artefakttyp. Die sechs RFLPV²-Artefakttypen und die neun Relationstypen werden in drei Modellierungsintensitätsstufen eingeteilt. Am Beispiel der Artefakttypen Anforderungen und Funktionen werden die drei Intensitätsstufen beschrieben: In Intensitätsstufe 1 wird lediglich die Relation zwischen einer Funktion und einer Anforderung beschrieben. Hierzu wird die Relation <<satisfies>> eingesetzt. Intensitätsstufe 2 erweitert die Abbildung von Wirkzusammenhängen durch vertikale Relationen, wodurch Anforderungen und Funktionen hierarchisch unterteilt werden. In Intensitätsstufe 3 werden horizontale Relationen ergänzt, die Abhängigkeiten zwischen Artefakten auf einer Hierarchieebene abbilden. Zusätzlich zu den Intensitätsstufen können die Artefakte in allen Intensitätsstufen durch zusätzliche Attribute detailliert werden, beispielsweise anhand der Hauptmerkmale der Hauptmerkmaliste [VDI21, S. 38, GDB19, S. 4], Identifikationsnummern und Prioritätskennzahlen. Die Intensitätsstufen werden in Bild 5-7 dargestellt. Für das Beispiel des Roboterarms wird Intensitätsstufe 3 gewählt, da die Modellelemente horizontal, vertikal und diagonal vernetzt werden müssen, um die definierten Ergebnisdimensionen beantworten zu können und die betroffene Pick-and-Place-Funktionalität detailliert abzubilden.

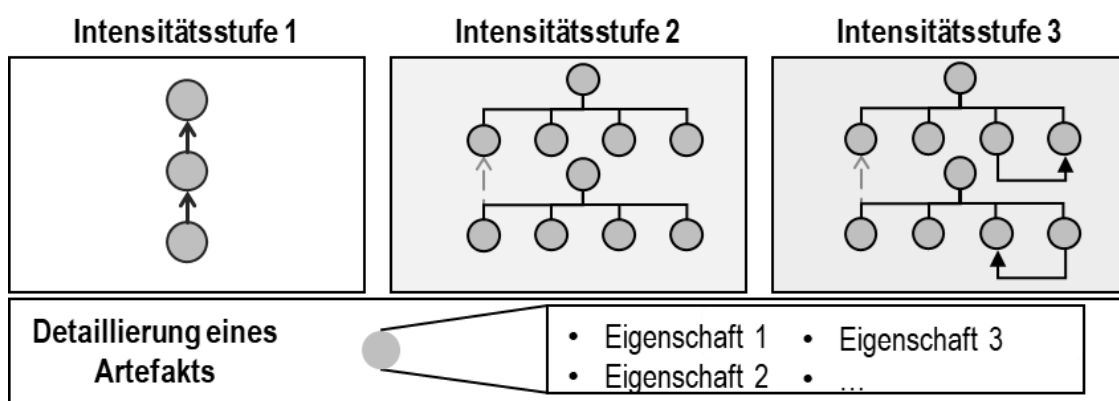


Bild 5-7: Intensitätsstufen der Modellierung für die Wirkkettenanalyse

Ergebnis der Aktivität ist eine festgelegte Stufe der Modellierungsintensität, die eingehalten werden muss, um alle erwarteten Ergebnisse folgern zu können. Nach der

Festlegung einer Modellierungsintensität werden aufbauend auf GRÄBLER und OLEFF [GO22a, S. 70] drei Leitfragen zum Abschluss der Analysephase beantwortet:

- Sind alle Ergebnisdimensionen der Wirkkettenanalyse klar definiert? (Nutzen)
- Sind das System und der Modellierungsumfang festgelegt? (Systemkomplexität)
- Stehen hinreichend Ressourcen für die Modellierung und Entwicklung der Wirkkettenanalyse zur Verfügung? (Zu leistender Modellierungsaufwand)

Die beschriebenen drei Aktivitäten der Analyse resultieren aus den hergeleiteten voneinander abhängigen Elementen des konzeptionellen Modells (siehe Kapitel 4.1). Das **SoI** wird in der ersten Aktivität von seinem Systemumfeld abgegrenzt. Für das SoI wird ein **Wirkkettenmodell** entwickelt, um eine **Wirkkettenanalyse** technischer Änderungen zu ermöglichen. Zur Festlegung der Inhalte des Wirkkettenmodells werden alle **Stakeholder** einbezogen, die ein Interesse an Ergebnissen der Wirkkettenanalyse haben. Jeder Stakeholder äußert unterschiedliche Anliegen, die durch **Ergebnisdimensionen** in Form von Fragestellungen, Metriken und Visualisierungsarten formalisiert werden. Zur Abbildung einer rollenspezifischen Ergebnisdimension wird die **Perspektive** der Stakeholder konkretisiert. Jeder Stakeholder kann in unterschiedlichen Rollen eine eigene Perspektive einnehmen. Grundlage zur Herleitung der Perspektiven sind die konzipierten MBSE-Rollen (Kapitel 4.3). Die Fragestellung einer Ergebnisdimension wird durch eine oder mehrere Ergebnissichten auf das Wirkkettenmodell beantwortet. Eine **Ergebnissicht** zeigt einen Teilausschnitt des Wirkkettenmodells in Übereinstimmung mit einer Perspektive. Jede Ergebnisdimension erfordert einen spezifischen **Wirkpfad**, der entlang definierter **Artefakttypen** und **Relationstypen** verläuft. Eine Ergebnissicht repräsentiert diesen **Wirkpfad** und stellt der Rolle damit die benötigten Informationen bereit. Die Menge der Wirkpfade definiert den abgeleiteten **Informationsbedarf**. Dieser umfasst somit alle Ergebnisdimensionen, von allen Stakeholdern in Bezug auf das SoI. Der Informationsbedarf bestimmt die notwendige Modellierungsintensität und muss in der Synthesephase in ein Datenmodell transformiert werden.

Erfahrungen aus den Modellierungsworkshops (WS 5-7) im ImPaKT Projekt und Workshops in der deutschen Automobilindustrie (WS 15) haben bestätigt, dass die Metadatenmodelle an die Bedarfe des Betrachtungsraums und die erwarteten Ergebnissichten anzupassen sind. Daher werden die identifizierten Informationsbedarfe innerhalb der **Synthesephase** in ein maßgeschneidertes Metadatenmodell (Element 3) transformiert. Hierzu werden die drei Syntheseaktivitäten *Metadatenmodell definieren*, *Glossar erstellen* und *Modellierungsregeln definieren* durchgeführt, um drei Fragen zu beantworten:

- Welche Artefakttypen werden durch welche Relationstypen verknüpft?
- Wie werden die Artefakttypen und Relationstypen definiert?
- Welche Regeln müssen bei der Modellierung eingehalten werden?

In der Aktivität *Metadatenmodell definieren* werden die benötigten Artefakttypen und Relationstypen definiert und in ein Metadatenmodell überführt, welches an die spezifischen Informationsbedarfe angepasst wird. Die Definition des Metadatenmodells folgt der Intention der Wirkkettenmodellierung, systematisch die tatsächlich geforderte Menge an Elementen und Abhängigkeiten abzubilden, um spezifische Ursache-Wirkungs-Beziehungen zu erfassen [GWK22, S. 1887]. Die identifizierten Wirkpfade aus der Analysephase werden somit in der Synthesephase in ein deklaratives Metadatenmodell transformiert. Durch die Abbildung der Wirkpfade wird die zweite Dimension des Systemdenkens adressiert: Das Erkennen von Zusammenhängen und Wechselwirkungen [GO22d, S. 94]. Basierend auf dem Dreiklang aus Tracepfad, Artefaktklasse und Relationsklasse nach REMPEL und MÄDER [RM15, S. 177] wird der Wirkpfad in separate Artefakttypen und Relationstypen dekomponiert. Ausgangspunkt ist das RFLPV²-Metadatenmodell (vgl. Kapitel 4.2), welches bereits in Vorveröffentlichungen im industriellen Anwendungszusammenhang angewendet wurde [GW23, S. 4]. Je nach identifiziertem Informationsbedarf wird das Modell bedarfsgerecht erweitert und kann in zukünftigen Projekten wiederverwendet werden. Das so entstehende Metadatenmodell beruht somit auf den zentralen RFLPV-Artefakten, kann aber im Sinne des Tailorings für den Anwendungsfall maßgeschneidert werden. Im Beispiel des Roboterarms werden Use Cases als separate Knoten ergänzt und die Liste an Parametern auf physischer Ebene erweitert, um die Software (SW) des Roboterarms mit abzubilden. Die ergänzten Artefakte sind in Bild 5-8 abgebildet und entlang der RFLPV²-Artefakte dargestellt.

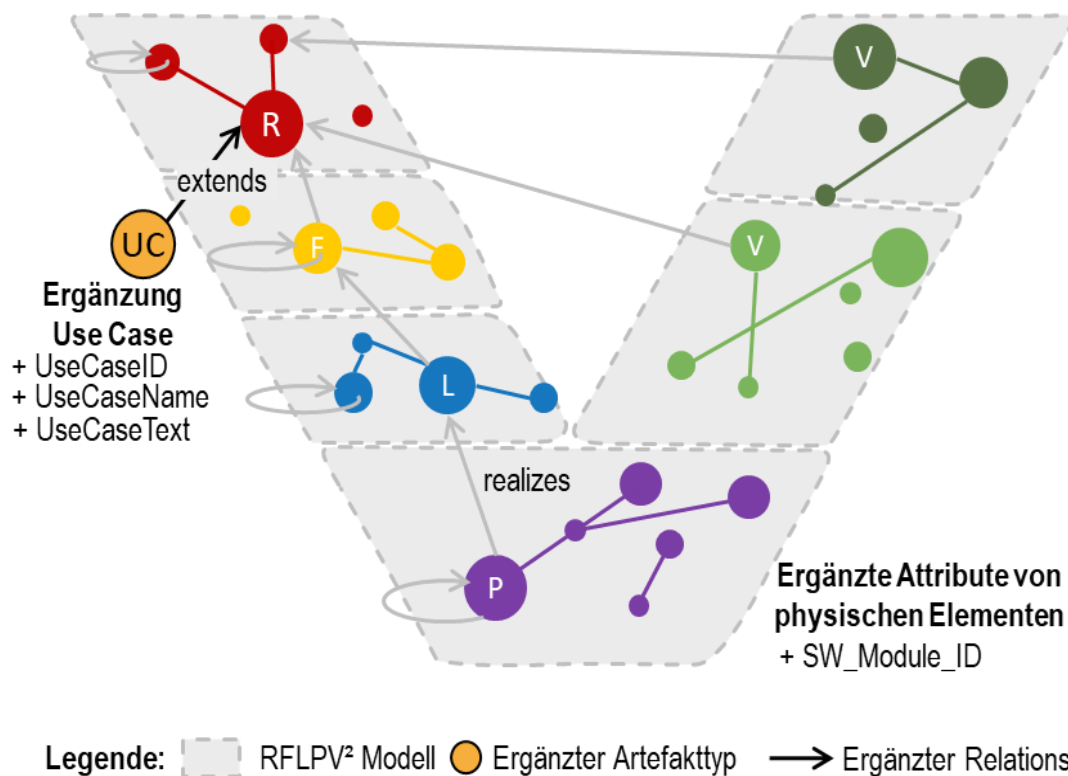


Bild 5-8: Darstellung der Artefakte und Relationen eines maßgeschneiderten Metadatenmodells für den Roboterarm entlang des V-Modells [VDI21, S. 21]

Die Zusammenhänge zwischen den Artefakttypen und Relationstypen lassen sich am V-Modell visualisieren, um allen Stakeholdern ein übersichtliches Bild über das Metadatenmodell bereitzustellen. Die Darstellung entlang des V-Modells hat sich in den bilateralen Forschungsprojekten (F2, F3, F4, F5) bewährt. Ergebnis der Aktivität *Metadatenmodell definieren* ist das spezifische Metadatenmodell für die Wirkkettenanalyse des SoI.

Zusätzlich zum definierten Metadatenmodell werden zwei ergänzende Aktivitäten durchgeführt, die die Semantik und Syntaktik [Krc15, S. 16] zur Anwendung des Metadatenmodells spezifizieren und so zur Beschreibung der Architektur beitragen. Erfahrungen aus Modellierungen in Forschungsprojekten (ImPaKT, Paragon, Automobil-OEM, BHTC) zeigen, dass ein einheitliches Verständnis essenziell für effiziente Wirkkettenanalysen ist. Aufbauend auf dem Grundgedanken, eine einheitliche Projektsprache zu fördern [Wei07, S. 85], wird auf semantischer Ebene die Aktivität *Glossar erstellen* durchgeführt. In dem Glossar werden die verwendeten Artefakttypen und Relationstypen einheitlich definiert. Als Hilfsmittel zur Erstellung eines Glossars werden Steckbriefe je Artefakt- oder Relationstyp erstellt. Ein Steckbrief beschreibt einen Artefakttyp oder Relationstyp durch eine Definition, durchzuführende Prozessaktivitäten, Befähiger, Kontrollmöglichkeiten und zugehörige Schlüsselrollen (siehe Bild 5-9). Die Steckbriefe helfen, unternehmensspezifische Interpretationen der Artefakte und Relationen zu reduzieren und ein gemeinsames Verständnis sicherzustellen [GWO22, S. 4]. Die Anwendung der entwickelten Steckbriefe wird in einer Vorveröffentlichung weiter spezifiziert [GWO22]. Das Ausfüllen der Steckbriefe wird für das Artefakt Anforderungen durch die Entwickler des Roboterarms durchgeführt (siehe Bild 5-9). Weitere Steckbriefe befinden sich im Anhang (siehe Anhang A4.4, S. 184).

...		
Anforderungen		
Definition: Eine Anforderung ist eine Aussage, die ein Merkmal oder eine Einschränkung eines Systems, eines Produkts oder eines Prozesses identifiziert, die bestimmte Qualitätskriterien erfüllt, überprüfbar ist und als notwendig für die Akzeptanz durch die Interessengruppen erachtet wird.		Prozessaktivitäten: Erhebung, Dokumentation, Vereinbarung und Management
Befähiger: Liste der Hauptmerkmale, Anforderungsvorlage, Qualitätskriterien	Kontrollen: IEEE STD 830, IEEE 1233, VDI 2519	Schlüsselrollen: Requirements Engineer

Bild 5-9: Exemplarischer Steckbrief für eine Anforderung [GWO22, S. 4], basierend auf Definitionen nach [INC15, S. 264] und [PR21, S. 13]

Neben der Festlegung der Begriffe auf semantischer Ebene, werden auf syntaktischer Ebene Modellierungsregeln festgelegt. Die Erfahrungen aus der Anwendung der MECA

Methodik sowie Erfahrungen weiterer Autoren von Modellierungsansätzen wie ALT [Alt12, S. 185] und FRIEDENTHAL [Fri14, S. 413] zeigen, dass Modellierungsregeln notwendig sind, um Modelle kollaborativ und fehlerfrei zu modellieren. Die Modellierungsregeln der MECA-Methodik umfassen sowohl Namenskonventionen, Vorgaben zur Strukturierung entlang der Systemhierarchieebenen als auch Regeln zur syntaktischen Verknüpfung der Artefakttypen und Relationstypen. Im Folgenden werden beispielhafte Modellierungsregeln beschrieben, die sich auf das RFLPV²-Metadatenmodell anwenden lassen und in Forschungsprojekten (ImPaKT, Automobil-OEM) erprobt wurden:

- Modellierungsregel 1: Modelliere nie mehr als 7 Artefakte innerhalb einer Sicht
- Modellierungsregel 2: Jede funktionale Anforderung muss durch eine Funktion erfüllt werden
- Modellierungsregel 3: Jedes logische Element muss durch physische Elemente realisiert werden

Als Hilfsmittel werden Modellierungsregeln für das RFLPV²-Metadatenmodell bereitgestellt (siehe Anhang A4.2, S. 180). Eine exemplarische Regel ist in Bild 5-10 dargestellt.

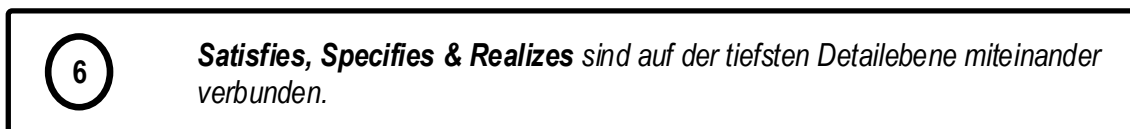


Bild 5-10: Darstellung einer exemplarischen Modellierungsregel für RFLPV²

Durch die Durchführung der Aktivitäten in der Synthese wird der Informationsbedarf durch ein spezifisches Metadatenmodell ersetzt. Die Architekturbeschreibung wird durch die festgelegte Modellierungsintensität sowie die zu nutzenden Artefakt- und Relationstypen mitsamt einem Glossar mit Steckbriefen und Modellierungsregeln detailliert. Nach Abschluss der Zieldefinition der Wirkkettenanalyse wird ein **Leitsatz** etabliert: Jede Folgeaktivität der Methodik muss auf die Erbringung der erwarteten Ergebnisse ausgerichtet sein und im Einklang mit den definierten Ergebnisdimensionen stehen (angelehnt an [GWK23, S. 10], aufbauend auf [Hol21, S. 41]).

Im Vergleich zu existierenden Ansätzen (vgl. Kapitel 3.2-3.4), entsteht durch das Frontloading [GO22d, S. 50] eine detaillierte Zieldefinition der Wirkkettenanalyse, die alle Stakeholder miteinschließt. Durch die frühzeitige Konkretisierung des Informationsbedarfs und des spezifischen Metadatenmodells werden Entwickler befähigt, maßgeschneiderte Wirkkettenanalysen zu entwickeln. Im Sinne des Tailorings [INC15, S. 265] werden die Aktivitäten der Modellbildung und Analyse an die Zieldefinition angepasst, wodurch sich der Modellierungs- und Analyseaufwand reduzieren und der Nutzen erhöht.

5.2 Schritt 2: Ermittlung von Informationen

Im zweiten Schritt werden die Informationen ermittelt, die dem Informationsbedarf für die Wirkpfade entsprechen und für die Wirkkettenmodellierung benötigt werden. Hierzu werden drei Aktivitäten durchgeführt, welche durch Hilfsmittel ergänzt werden (siehe Bild 5-11). Erfahrungen aus Workshops in der Automobilindustrie (WS-15 in Anhang A1, S. 139) haben gezeigt, dass der Aufwand zur Informationserhebung den Aufwand der Modellierung übersteigen kann [GWK22, S. 1892f.]. Der Aufwand der Informationserhebung resultiert aus unzureichender Informationsqualität, Heterogenität in der Datenhaltung und veralteten Versionsständen.

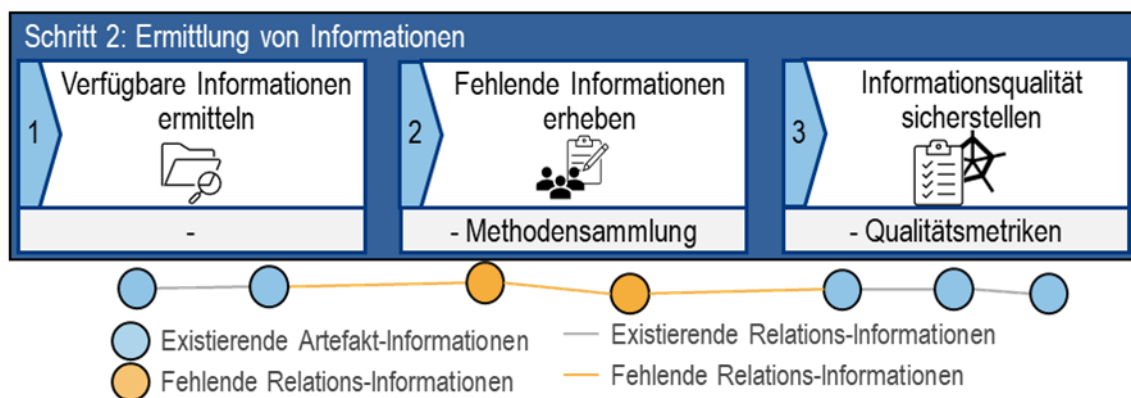


Bild 5-11: Darstellung des Schritts Identifikation von Informationen

In der ersten Aktivität *verfügbare Informationen ermitteln* werden alle vorhandenen Informationen zu den Artefakttypen und Relationstypen zusammengetragen, um den IST-Informationsstand zu bestimmen. Die Stakeholder werden eingebunden, um die vorliegenden Informationen bereitzustellen. Alle identifizierten Informationen werden nach Artefakt- und Relationstypen an einem zentralen Ablageort gespeichert. Zusätzlich werden die Metadaten des Titels, der Quelle, des Typs und der Relation dokumentiert. Informationen aus spezifischen Modellen des Roboterarms sind in Bild 5-12 dargestellt. Hierzu zählen Anforderungslisten, Funktionslisten, Stücklisten aus einem PLM-System und geometrische Parameter aus M-CAD Modellen. Durch die Bestimmung des IST-Informationszustands werden fehlende Daten und Informationen zu Artefakt- und Relationstypen aufgedeckt. Eine ganzheitliche Darstellung der vorhandenen Daten des Roboterarms ist im Anhang dargestellt (siehe Anhang A5.2, S. 204)

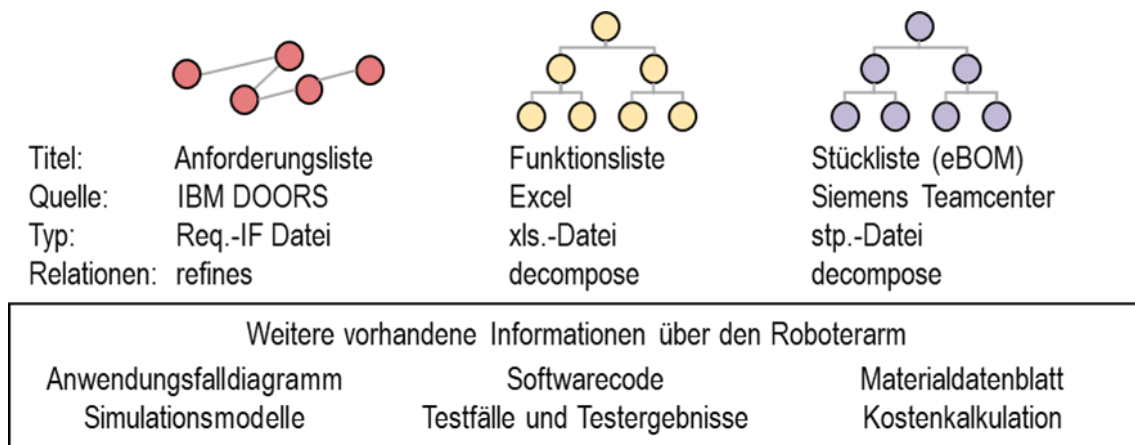


Bild 5-12: Darstellung von exemplarischen identifizierten Informationen

Die fehlenden Informationen müssen systematisch mit Methoden und Hilfsmitteln der Informationserhebung zusammengetragen werden. Hierbei werden drei Möglichkeiten unterschieden: indirekte Wissensakquisition, direkte Wissensakquisition und automatische Wissensakquisition [VWZ18, S. 421]. Bei der indirekten Wissensakquisition wird der Entwickler beim Explizieren des Wissens unterstützt. In diesem Kontext werden Design Structure Matrizen [Epp12] in den Modellierungsworkshops (WS 5-7, WS 15) als praktikables Hilfsmittel zur Erhebung von Relationen empfohlen. Weitere empfohlene Methoden und Hilfsmittel sind Experteninterviews [BLM09], Fokusgruppen [PT07], Fragebögen [KP10] oder virtuelle Whiteboards [KW15]. Bei der direkten Wissensakquisition kann der Entwickler das Wissen eigenständig formalisieren und im Wirkkettenmodell hinterlegen. Ein Beispiel ist die direkte Modellierung von Anforderungen in SysML. Am Beispiel des Roboterarms werden Modellelemente aus den oben genannten Modellen in SysML übertragen und dort von den Entwicklern verknüpft. Nutzbare Methoden für die automatisierte Wissensakquisition sind beispielsweise die Extraktion von Anforderungen aus Lastenheften [GÖ23, S. 169]. Alle neu erhobenen Informationen werden in dem zentralen Ablageort gespeichert.

In der dritten Aktivität *Informationsqualität sicherstellen* werden Kriterien der Informationsqualität [Krc15, S. 143] festgelegt und angewendet, die die Informationsqualität quantifizieren und nachweisen. Als Hilfsmittel werden existierende Standards und Rahmenwerke verwendet, die die Informationsqualität anhand verschiedener Kriterien und Metriken bewerten. Ein weitverbreitetes Rahmenwerk der Deutschen Gesellschaft für Informations- und Datenqualität (DGIQ) [RKM15, S. 28] definiert 15 Dimensionen der Datenqualität. Ein weiterer Standard beschreibt das Qualitätsmodell der System and Software Quality Requirements and Evaluation (SQuaRE) der ISO/IEC 25010 [ISO11a], welches in Bild 5-13 dargestellt wird. Für den Roboterarm werden die Aktualität, die Vollständigkeit und die Objektivität der Daten durch die beteiligten Stakeholder geprüft.

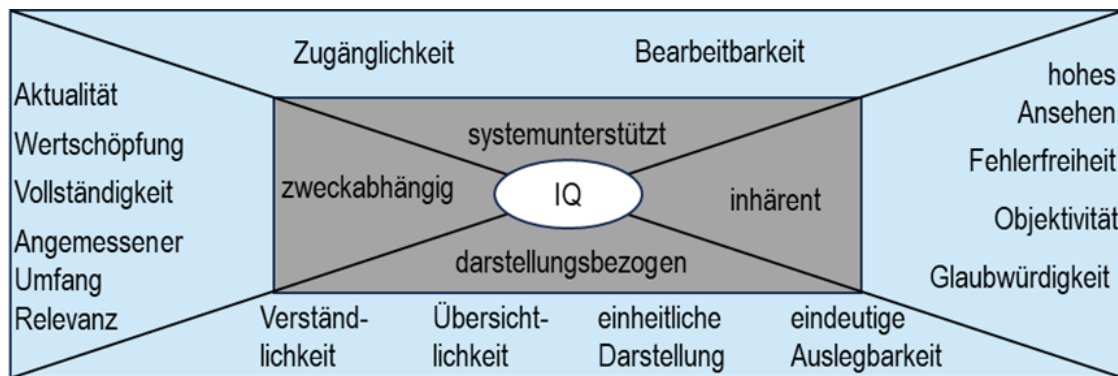


Bild 5-13: DIGQ nach ISO/IEC 25010 (angelehnt an [RKM15, S. 28])

5.3 Schritt 3: Modellierung des Wirkkettenmodells

Im dritten Schritt der MECA-Methodik wird das Wirkkettenmodell des SoIs auf Basis des spezifischen Metadatenmodells (Schritt 1) und der Informationen (Schritt 2) modelliert. Mithilfe der MECA-Methodik werden maßgeschneiderte Wirkkettenmodelle erstellt, welche immer nur die benötigten Modellelemente und Ursache-Wirkungs-Beziehungen zwischen Artefakten abbilden und die Start- und Endpunkte der Wirkpfade enthalten [GWK22, S. 1888]. Da das Wirkkettenmodell lediglich als Grundlage für die Wirkkettenanalyse genutzt wird, liegt der Modellierungsschwerpunkt auf den für die Stakeholder relevanten Aspekten [GO22a, S. 71] und der einfachen, fehlerfreien und schnellen Modellierung.

Zur Wirkkettenmodellierung werden eine Modellierungssprache und ein Modellierungswerkzeug ausgewählt. Die einzuhaltende Voraussetzung bei der Auswahl ist: Alle Artefakt- und Relationstypen müssen dargestellt werden können; alle Wirkpfade müssen durchlaufen werden können [GW23, S. 3]. Für die Wirkkettenanalyse werden objektorientierte und graphenbasierte Modellierungsansätze kombiniert, da sie sich als Analysefundament eignen (vgl. [GW21, S. 7] und Anhang A2.5, S. 167). Im Folgenden wird die Wirkkettenmodellierung für den Roboterarm anhand der objektorientierten Modellierungssprache SysML [OMG19] in Kombination mit einem Property-Graph-Modell durchgeführt. Die Wirkkettenmodellierung umfasst drei Aktivitäten: *Metamodell anpassen*, *System dekomponieren*, *Wirkketten modellieren* (siehe Bild 5-14).

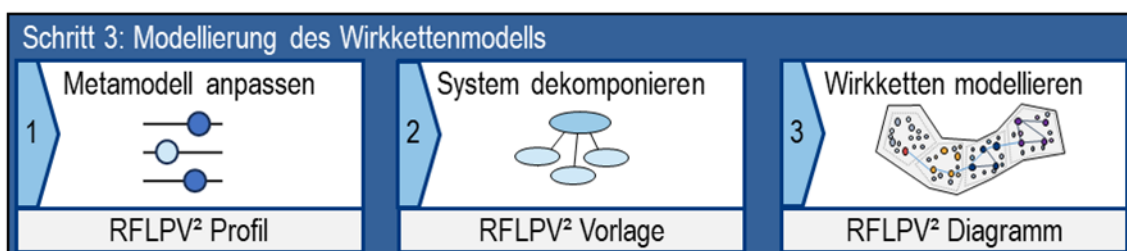


Bild 5-14: Aktivitäten zur Modellierung des Wirkkettenmodells

In der ersten Aktivität *Metamodell anpassen* wird die ausgewählte Modellierungssprache auf semantischer und syntaktischer Ebene an die Wirkkettenarchitektur angepasst. In diesem Schritt wird die Wahl der Modellart und das damit verbundene Repräsentieren der Relationen und Systemelemente festgelegt [PDH11, S. 102]. Bei der Modellierungssprache SysML wird ein UML Profil Diagramm erstellt, in dem <<Stereotypen>> für die nicht existierenden Artefakt- und Relationstypen erstellt werden. Das Profil enthält Stereotypen, Metaklassen und die Verbindungen zwischen diesen [Fri14, S. 354]. Vorgegebene Elemente der Modellierungssprache werden so übernommen und bedarfsgerecht erweitert. Für die Darstellung des Metadatenmodells des Roboterarms existieren in SysML beispielsweise Objekte für Anforderung und Anwendungsfälle, aber kein Objekt für Funktionen, logische Elemente, physische Elemente und Testfälle der Verifikation und Validierung. Daher werden für alle nicht existierenden benötigten Artefakt- und Relationsklassen Stereotypen entwickelt und bereitgestellt. Je Stereotyp werden spezifische Eigenschaften definiert, die Metadaten über ein stereotypisiertes Modellelement repräsentieren [Fri14, S. 356]. Für das Fallbeispiel des Roboterarms wurde beispielsweise die Eigenschaft *safetyCritical* ergänzt, um sicherheitskritische Funktionen zu kennzeichnen (siehe Bild 5-15). Gleiches gilt für die Erstellung und Anpassung eines Metamodells eines Property-Graphen. Dort werden je Artefakttyp ein Knoten und je Relationstyp eine Kante definiert. Wirkzusammenhänge zwischen zwei Artefakttypen lassen sich als Tupel aus einem Ausgangsknoten, einer Relation und einem Eingangsknoten darstellen. Ein Beispiel ist die Verbindung eines Knoten Verifikation per verifies-Relation zu einem Requirement (siehe Bild 5-15).

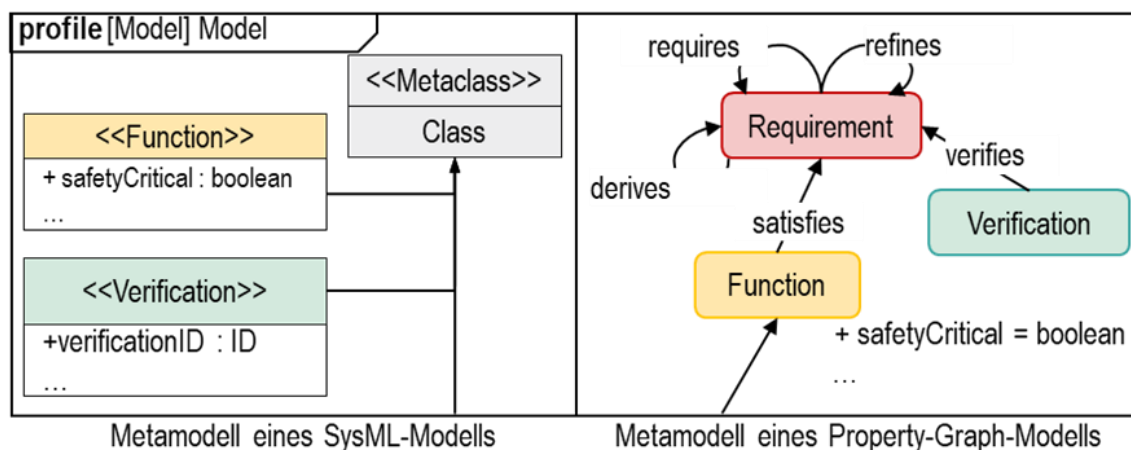


Bild 5-15: Aktivität *Metamodell anpassen*

Als Hilfsmittel werden zwei Metamodellerweiterungen definiert und bereitgestellt: eine Metadatenmodellerweiterung in Form eines UML-Profiles und ein Metadatenmodell für eine Property-Graph-Datenbank entlang der RFLPV²-Artefakt- und Relationstypen. Die festgelegten Stereotypen und Anpassungen werden im Folgenden erläutert. Zur Definition der Relationstypen werden die neuen Begriffe als Erweiterung einer <<Dependency>> hergeleitet. Die UML-Metaklasse <<Abstraction>> eignet sich, da sie zwei

Konzepte auf unterschiedlichen Abstraktionsniveaus verbindet, welche in einer Sender-Empfänger-Beziehung zueinander stehen [UML, S. 51]. Die Stereotypen <<Refines>> und <<Requires>> haben als Metaklasse Dependency, da sie auf einer Abstraktionsebene angewendet werden, beispielsweise zwischen mehreren Teilfunktionen. Ebenfalls werden im RFPLV² Profil Icons erstellt, um die Relationstypen einfach und eindeutig unterscheiden zu können (siehe Bild 5-16). Das vollständige Profil ist angehängt (siehe Anhang A4.6, S. 186).

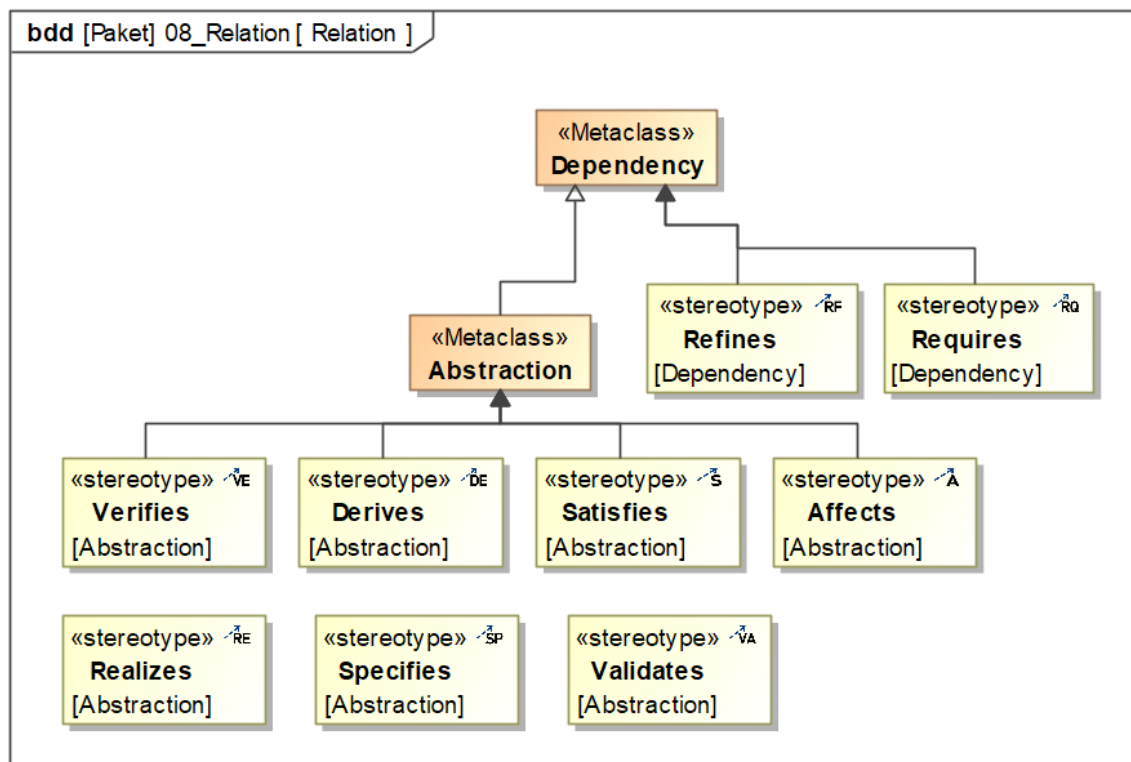


Bild 5-16: Relationstypen im neuen RFPLV²-Profil

Neben den neun Stereotypen der Relationstypen werden neun Stereotypen für Artefakttypen erstellt. Dies umfasst die Stereotypen für die Artefakttypen „Kundenbedürfnis“, „Kundenanforderung“, „Systemanforderung“, „Funktion“, „Logisches Element“, „Physisches Element“, „Verifikation“, „Validierung“ und „Engineering Change Request“. Jeder Artefakttyp wird durch Eigenschaften detailliert. Die Auswahl der vordefinierten Eigenschaften beruht auf der Analyse benötigter Eigenschaften aus 144 Ergebnisdimensionen (siehe Anhang A4.3, S. 181), Erfahrungen aus den Modellierungsworkshops (WS 5-7) und Standards und Normen. Ein Beispiel für die Nutzung von Normen ist die Definition der Eigenschaften eines ECRs nach VDA 4965 [VDA09b, S. 64f.]. Zwei Prinzipien zur Differenzierung werden unterschieden: die Modellierung separater Stereotype, beispielsweise für Kunden- und Systemanforderungen, und die Nutzung von Eigenschaften, beispielsweise die Unterscheidung der Disziplinen eines physischen Elements. Die Umsetzung der zentralen Artefakte ist in Bild 5-17 dargestellt.

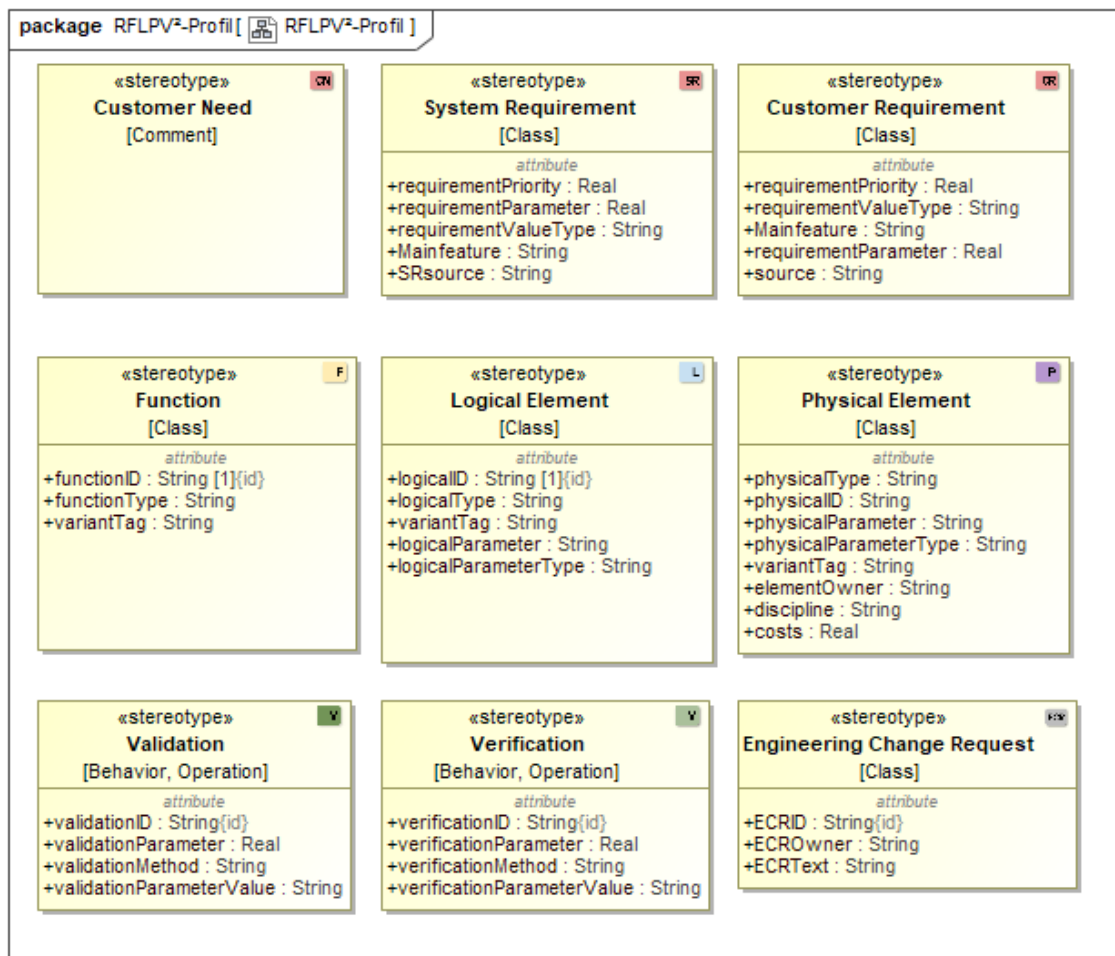
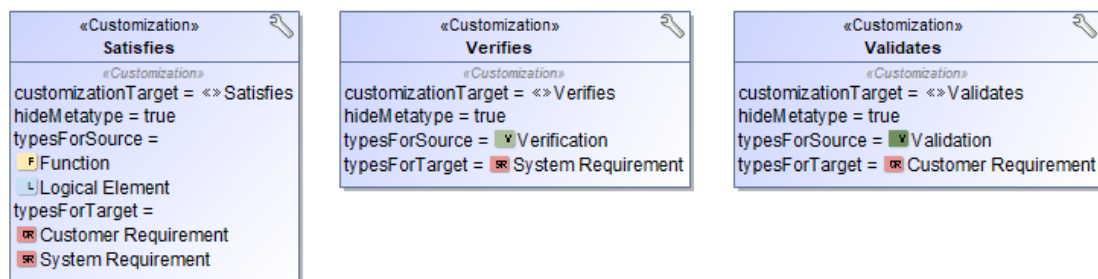


Bild 5-17: Artefakttypen im neuen RFLPV²-Profil

Das SysML-Profil wird neben stereotypisierten Objekten der Artefakt- und Relationstypen ebenfalls ergänzt durch Customizations, die die Modellierungsregeln (Aktivität 1.6) syntaktisch in das Metadatenmodell integrieren. Durch das Objekt <<Customization>> können Restriktionen und Reglementierungen für die Nutzung der Artefakt- und Relationstypen vorgenommen werden. Eine beispielhafte Restriktion ist, dass Testfälle zur Verifikation nur per <<verifies>> mit Systemanforderungen verknüpft werden dürfen. Andere Relationsarten sind nicht erlaubt und können nicht modelliert werden. In dem Metadatenmodell eines Property-Graph-Modells wird dies vorgenommen, indem Wirkzusammenhänge in festen Tupel festgehalten werden. In Bild 5-15 ist beispielsweise nur eine Verknüpfung einer Funktion per <<satisfies>> Relation zu einer Anforderung möglich. Drei exemplarische Customizations werden in Bild 5-18 dargestellt. Ein Beispiel ist die Anpassung des Relationstyps <<derives>>, welcher nur zwischen Kundenbedürfnissen, Kundenanforderungen und Systemanforderungen angewendet werden darf. Durch das RFLPV²-Profil wird die Anzahl der benötigten Systemelemente stark reduziert. Im Vergleich zu über 250 Modellelementen in SysML benötigt der Entwickler lediglich 19 Modellelemente, um ein Wirkkettenmodell zu erstellen.

Bild 5-18: Beispielhafte Anpassungen im RFLPV²-Profil

Neben dem SysML-Profil wird ebenfalls ein Metadatenmodell für die Graphdatenbank neo4j bereitgestellt, welches das einfache Erstellen eines RFLPV²-Graphen ermöglicht (siehe Anhang A4.7, S. 188). Dieses wird um die benötigten Knoten zur Abbildung des Metadatenmodells des Roboterarms ergänzt (siehe Anhang A5.4, S. 212). Ergebnis des ersten Schritts sind ein oder mehrere angepasste Metadatenmodelle, um die fallbeispiel-spezifischen Wirkkettenmodelle zu erstellen.

In der zweiten Aktivität *System dekomponieren* wird die grundlegende hierarchische Struktur des Wirkkettenmodells festgelegt. Hierzu wird das SoI (Schritt 1) aus hierarchischer Sicht dekomponiert. Die schrittweise Dekomposition des Systems auf den Hierarchieebenen ermöglicht die Detaillierung des Systems und die Verschachtelung einzelner Wirkketten. Basierend auf der Modellierungsintensität (Schritt 1) wird aus artefakt-orientierter Sicht festgelegt, welche Artefakt- und Relationstypen auf welcher Hierarchieebene modelliert werden. Für das Fallbeispiel des Roboterarms wird entschieden, dass das System in die vier hierarchischen Ebenen nach Haberfellner [HWF19, S. 8] unterteilt wird: Systemumfeld (RC-Car, Shuttle, ...), SoI (Roboterarm), Subsystem (Greifer-System, Karussell, Arm, Vision Set, ...) und Systemelement (Greiferbacke, Servomotor, ...). Einzelne Artefakttypen werden anschließend zugewiesen. Spezifische Funktionen wie *Greifkraft aufbringen* werden auf einer niedrigeren Hierarchieebene einem oder mehreren Systemelementen (*Servomotor*) zugewiesen. Die Teilfunktionen werden von der untersten Ebene mit den Systemanforderungen verknüpft (siehe Bild 5-19).

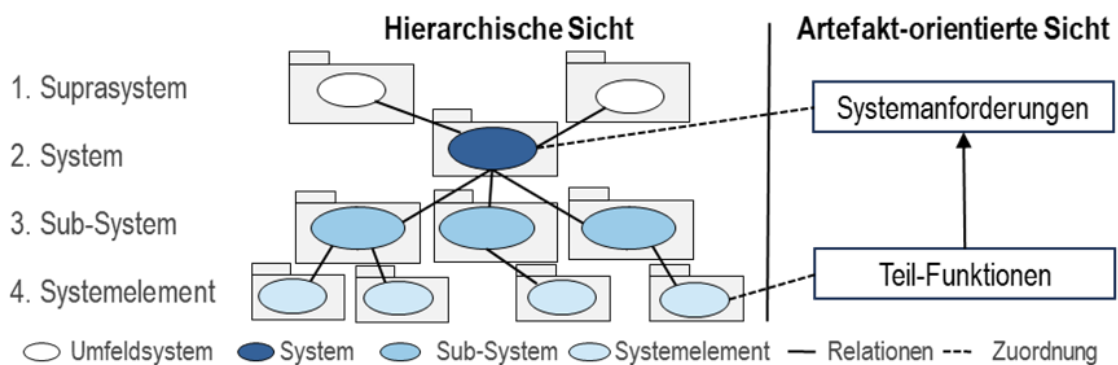


Bild 5-19: Aktivität System dekomponieren

Die Strukturierung und Modellierung des Wirkkettenmodells erfolgt in einer entwickelten RFLPV² Vorlage für SysML Modelle. Die Vorlage umfasst eine generische Struktur entlang von RFLPV², welche je Hierarchieebene verwendet werden kann. Die Elemente der SysML-Vorlage des RFLPV²-Metadatenmodells werden in Bild 5-20 dargestellt. Je Artefakttyp wird ein Paket (<<package>>) mit angepassten Icons erstellt und mithilfe eines Ordnungsschemas sortiert (Punkt 1). Das SysML Profil wird als separate Quelle integriert (Punkt 2). Zur visuellen Anordnung der SysML-Pakete werden die Inhalte auf einem Bild des V-Modells der VDI/VDE Richtlinie 2206:2021 dargestellt (Punkt 3).

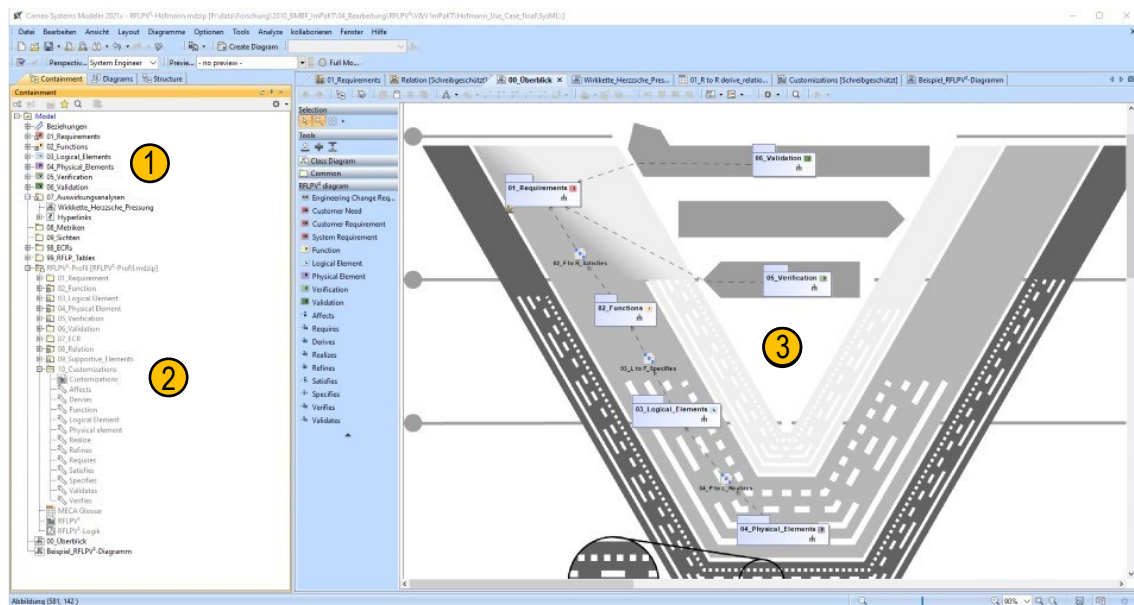


Bild 5-20: SysML Modellvorlage für RFLPV²

Das Ergebnis der Dekomposition ist eine unausgefüllte, hierarchisch unterteilte Strukturierung des Modells, beispielsweise als Paketstruktur in SysML. Je Systemebene wird eindeutig beschrieben und festgelegt, welche Artefakttypen und Relationstypen modelliert werden müssen.

In der dritten Aktivität *Wirkketten modellieren* werden die Artefakte und Relationen basierend auf dem angepassten Metamodell modelliert, wodurch das spezifische Wirkkettenmodell des SoI entsteht. Bei der Modellierung muss sichergestellt werden, dass die Modellierungsregeln eingehalten werden und so nur Wirkpfade entstehen, die analysiert werden können. Durch die Aktivität wird die vollständige Modellierung der hergeleiteten Wirkpfade sichergestellt.

Als Hilfsmittel zur Wirkkettenmodellierung in SysML wird ein separates RFLPV²-Diagramm auf Basis eines Blockdefinitionsdiagramms entwickelt. Die Besonderheiten des RFLPV²-Diagramms werden in Bild 5-21 dargestellt. Durch die Entwicklung des RFLPV²-Diagramms reduziert sich die Anzahl von Diagrammen im Vergleich zur vollumfänglichen SysML von neun Diagrammen auf ein Diagramm. Bei der Entwicklung des Diagramms wird die Werkzeugleiste angepasst (1). Die Werkzeugleiste umfasst nur

noch die definierten Artefakt- und Relationstypen. Zusätzlich werden bei der Schnellauswahl von Relationen für einen Artefakttyp die in dem Wirkkettenmodell festgelegten Relationsarten hinterlegt, wodurch schneller modelliert werden kann und Entwicklungskapazität eingespart wird (2). Durch die vorgenommenen Anpassungen (<<Customization>>) wird sichergestellt, dass keine fehlerhaften Relationen modelliert werden (3).

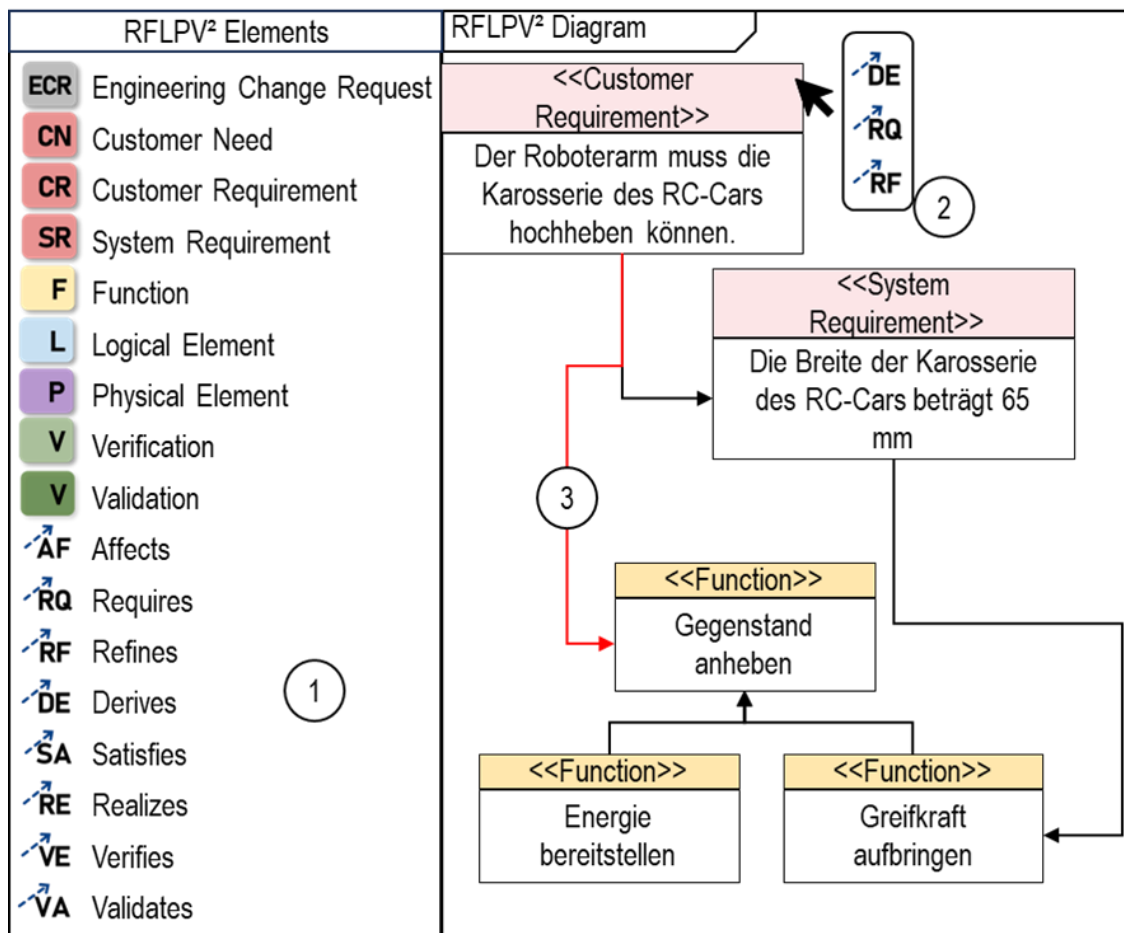


Bild 5-21: Wirkketten modellieren am Beispiel des Roboterarms

Das Ergebnis der drei Aktivitäten zur Wirkkettenmodellierung ist das spezifische Wirkkettenmodell, welches zur Analyse der Auswirkungen technischer Änderungen genutzt werden kann. Durch die Reduzierung der nutzbaren Elemente wird die Einfachheit der Wirkkettenmodellierung sichergestellt. Fehlerfreiheit wird durch die Implementierung von Modellierungsregeln gewährleistet. Schnelligkeit entsteht durch die zusätzlichen Anpassungen wie Schnellzugriffe und das Modellieren innerhalb eines Diagramms. Zur Kontrolle werden drei Fragen nach FRIEDENTHAL ET AL. [Fri14] beantwortet:

- Reicht der Umfang des Modells aus, um den beabsichtigten Zweck zu erfüllen?
- Ist das Modell in Bezug auf seinen Anwendungsbereich vollständig?
- Ist das Modell konsistent und verständlich?

5.4 Schritt 4: Durchführung der Wirkkettenanalyse

Im letzten Schritt werden die Ergebnisdimensionen (Schritt 1) in automatisierte Wirkkettenanalysen transformiert und rollenspezifische Ergebnissichten zur Bewertung der Änderungsauswirkungen entwickelt. Hierzu werden die in Bild 5-22 dargestellten Aktivitäten durchgeführt: *betroffene Elemente identifizieren*, *Auswirkungen bewerten* und *Analyseergebnisse darstellen*. Durch die Aktivitäten werden die Identifikation der Effekte (Element 2), die Abschätzung der Folgen (Element 4) und Ergebnisdarstellung (Element 5) umgesetzt. Methodische Grundlagen der Wirkkettenanalyse wurden auf der *IEEE System of Systems Engineering Conference 2023* präsentiert und vorveröffentlicht [GW23].

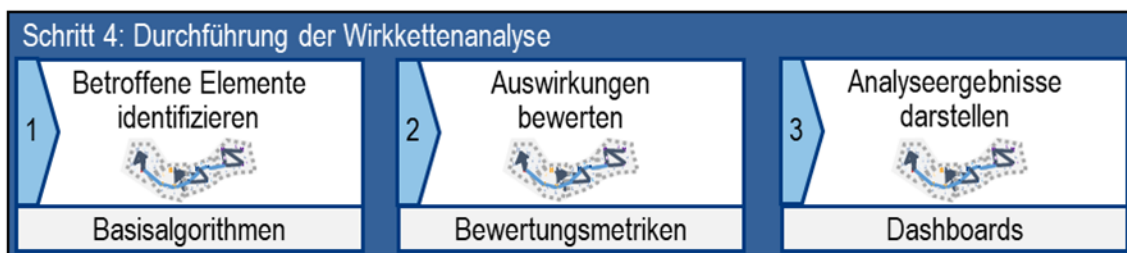


Bild 5-22: Aktivitäten zur Analyse der resultierenden Auswirkungen

Grundlage zur Entwicklung der maßgeschneiderten Wirkkettenanalysen sind die Wirkpfade aus den Ergebnisdimensionen (Schritt 1). Der Startpunkt zur Identifizierung der betroffenen Elemente entlang des Wirkpfades ist das von dem ECR initial betroffene Modellelement. Ausgehend von dem initial betroffenen Element wird der Wirkpfad (Propagationspfad) entlang der festgelegten Artefakttypen und Relationstypen durchlaufen und analysiert.

Der schematische Verlauf der Aktivität zur Identifikation betroffener Elemente wird in Bild 5-23 am Beispiel des Roboterarms dargestellt. Auslöser der technischen Änderung ist der technische Manager des RC-Cars, der die Karosserie des Fahrzeugs von 65 mm auf 73 mm verbreitern möchte. Zunächst wird der ECR von einem Systemarchitekten modellbasiert abgebildet und formalisiert. Anschließend wird die Wirkkettenanalyse von dem technischen Manager auf der Prozessebene gestartet. Hierzu wird das initial betroffene Element des ECRs festgelegt - die Anforderung „Breite des Greifobjekts“. Ausgehend von dem initial betroffenen Element werden durch die Wirkkettenanalyse automatisiert die Wirkpfade der Ergebnisdimension analysiert. Bei dem Durchlaufen des Wirkpfads werden Ursache-Wirkungs-Muster [SMS12, S. 292] in Abhängigkeit von dem Auslöser [PM98, S. 2] und Art der technischen Änderung [JEC11, 106] analysiert. Durch die Wirkkettenanalyse wird ein Anwender auf der Prozessebene befähigt, das entwickelte Wirkkettenmodell auszuwerten.

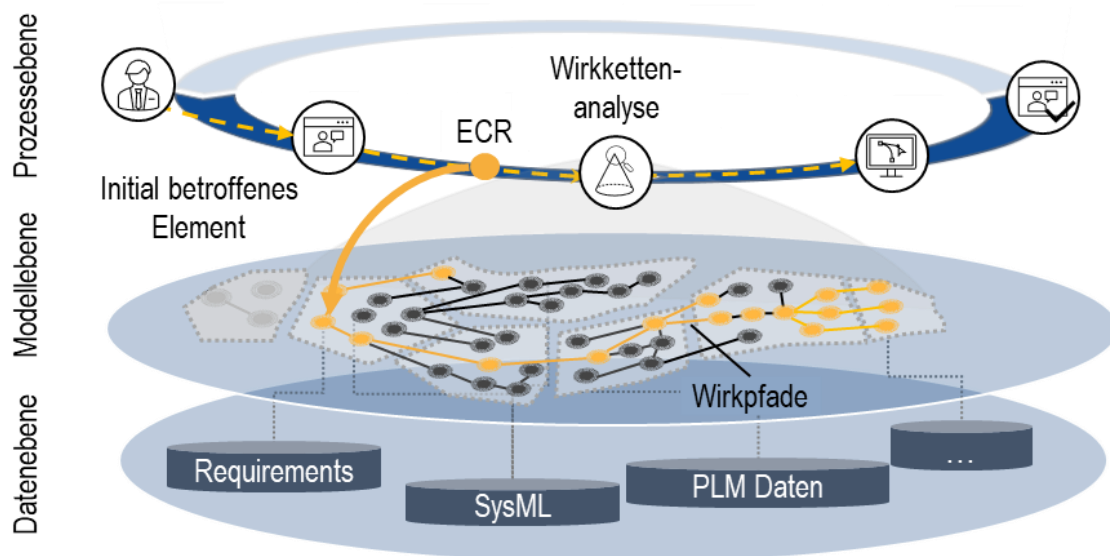


Bild 5-23: Identifikation betroffener Elemente

Jeder Wirkpfad (Schritt 1) wird als Graphmuster (siehe Kapitel 2.1.4) in einer Query abgebildet. Das Ablaufen des Wirkpfads erfolgt automatisiert mithilfe von Abfragesprachen (*Querysprachen*). Für die Wirkkettenanalyse des Roboterarms wird Cypher gewählt, da die Abfragesprache einfach verständlich ist [Kem15, S. 15] und sich für die eigenständige Entwicklung von maßgeschneiderten Wirkkettenanalysen eignet [GW23, S. 5]. Cypher wird verwendet, um neo4j Property-Graphen zu analysieren und bietet Funktionen zur Abfrage und Änderung von Daten und zur Angabe von Schemadefinitionen [FGG18, S. 1434]. Durch die Programmierung mit Cypher werden die vier Kategorien von Graph-Abfragen nach BESTA ET AL. [BGP24, S. 10] für die Wirkkettenanalyse nutzbar: lokale Abfragen, Nachbarschaftsabfragen, Traversalabfragen entlang von Untergraphen sowie globale Abfragen innerhalb eines gesamten Graphen. Die Nutzung navigationaler Queries für die Wirkkettenanalyse wurde aufgrund von Erfahrungen aus der Anwendung vernachlässigt, da nicht alle Ursache-Wirkungs-Muster erkannt werden.

Für die systematische Transformation der Fragestellung in einen rechnerinterpretierbaren Wirkpfad wird die Fragestellung schrittweise nach dem Prinzip „Walking the path step by step“ nach MILLER analysiert [Mil13, S. 145]. Hierzu wird die Fragestellung in separate Tupel eines Wirkpfads aufgeteilt, bestehend aus einem ausgehenden Artefakttyp, einem Relationstyp und einem betroffenen Artefakttyp [GW23, S. 3]. Ein beispielhaftes Tupel ist eine betroffene Systemanforderung, welche per Refines-Relation mit anderen Anforderungen verknüpft wird. In Bild 5-24 wird aus dem Tupel eine Cypher Query für den Anforderungsentwickler hergeleitet. In den ersten zwei Zeilen des Cypher-Codes wird der zu analysierende Wirkpfad abgebildet. Der Operator MATCH leitet den abzuführenden Wirkpfad ein. Artefakte des Wirkpfads werden in runden Klammern spezifiziert, Relationen werden in eckigen Klammern spezifiziert. Je Artefakt- und Relationstyp wird eine Variable definiert und durch einen Doppelpunkt getrennt. Nach dem Operator RETURN wird festgelegt, welche Variablen ausgegeben werden sollen.

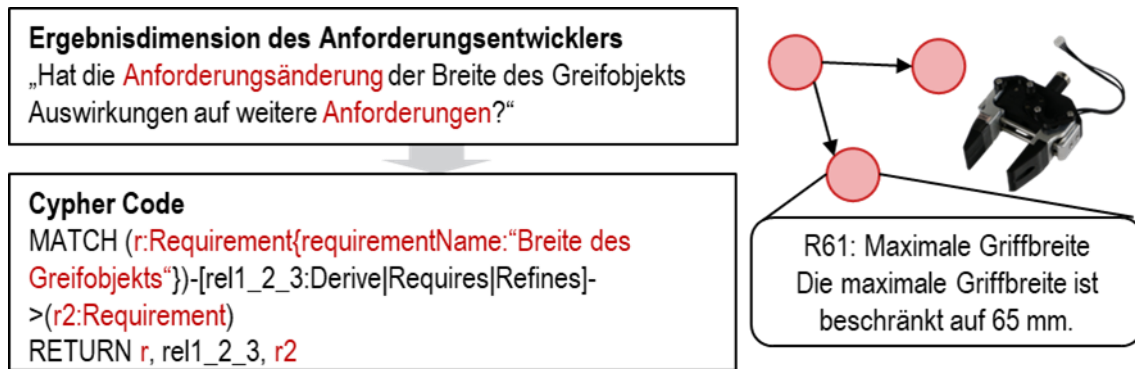


Bild 5-24: Wirkkettenanalyse für die Frage des Anforderungsentwicklers

Durch die Wirkkettenanalyse wird festgestellt, dass die Griffbreite des Greifers auf 65 mm begrenzt ist und das RC-Car mit der Karosseriebreite von 73 mm nicht mehr gegriffen werden kann (siehe Anhang A5.3, S. 210). Aufbauend auf dem Ergebnis prüft der Systemarchitekt, ob sicherheitskritische Funktionen betroffen sind (siehe in Bild 5-25).

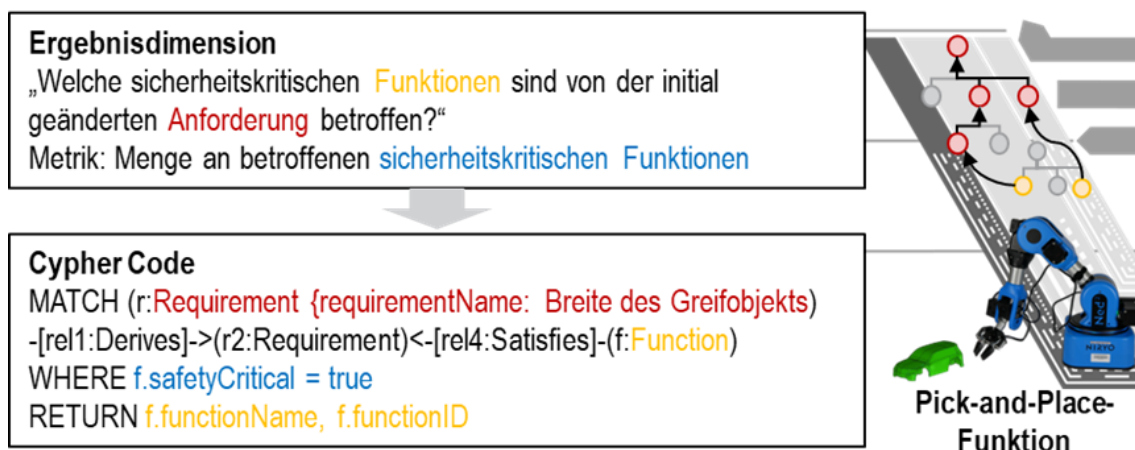


Bild 5-25: Wirkkettenanalyse für die Frage des Systemarchitekten

Durch die Query wird identifiziert, dass die sicherheitsrelevante Pick-and-Place-Funktion zum Anheben der Karosserie eines RC-Cars betroffen ist, da Fehler zu Verletzungen von Shopfloor-Mitarbeitern führen können. Resultat der Wirkkettenanalysen ist, dass ein neues Greifersystem entwickelt werden muss. Die Beispiele verdeutlichen, dass Wirkpfade durch maßgeschneiderte Abfragen analysiert werden können. Zusätzlich können Teile der entwickelten Abfragen wiederverwendet werden. Daher werden als Hilfsmittel wiederverwendbare Anfragen entwickelt – sogenannte Basisalgorithmen [GW23, S. 3]. Basierend auf den Erfahrungen aus der Analyse von 144 Ergebnisdimensionen (WS-3) wurden 26 Basisabfragen für die MBSE-Rollen entwickelt, die wiederverwendet werden können (siehe Anhang A4.9, S. 191). Die wiederverwendbaren **Basisalgorithmen** analysieren relevante identifizierte Ursache-Wirkungs-Muster (siehe Anhang A4.8, S. 190) der Änderungspropagation und stellen ein nutzbares Analyseverfahren dar.

Basierend auf den identifizierten betroffenen Systemelementen können die Auswirkungen qualitativ und quantitativ bewertet werden. Im Vergleich zu existierenden Ansätzen werden experten-basierte Bewertungen durch objektive Bewertungen auf Basis des Wirkkettenmodells ersetzt. In der Aktivität *Auswirkungen bewerten* werden die vordefinierten Metriken aus Aktivität 1.2 rechnerinterpretierbar in die Abfragen der betroffenen Elemente (Aktivität 4.1) eingebunden. Für jede **Bewertungsmetrik** wird ein Datentyp zur Repräsentation gewählt werden. Typische Datentypen sind: String, Boolean, Real, Integer, Float. Ein Beispiel für eine Metrik ist die Berechnung der kumulierten Materialkosten der betroffenen mechanischen Elemente (siehe Bild 5-26). Ausgehend von dem initial betroffenen Element wird ein Wirkungspfad entlang der betroffenen Funktionen, logischen Elemente und physischen Elemente abgelaufen. Die Materialkosten der betroffenen mechanischen Elemente werden durch den Cypher-Operator SUM summiert und als Wert (Real) der Variable Costs ausgegeben. Die kumulierten Kosten der betroffenen physischen Elemente entsprechen 85,5 € und sind unter anderem auf die Greiferbacken und den Servomotor des Greifers zurückzuführen. Zur Reduzierung der Änderungskosten soll der Servomotor im neuen Greifersystem mit einer Griffbreite über 73 mm wiederverwendet werden, wodurch ein neues Greifersystem entsteht (siehe Anhang A5.3, S. 210).

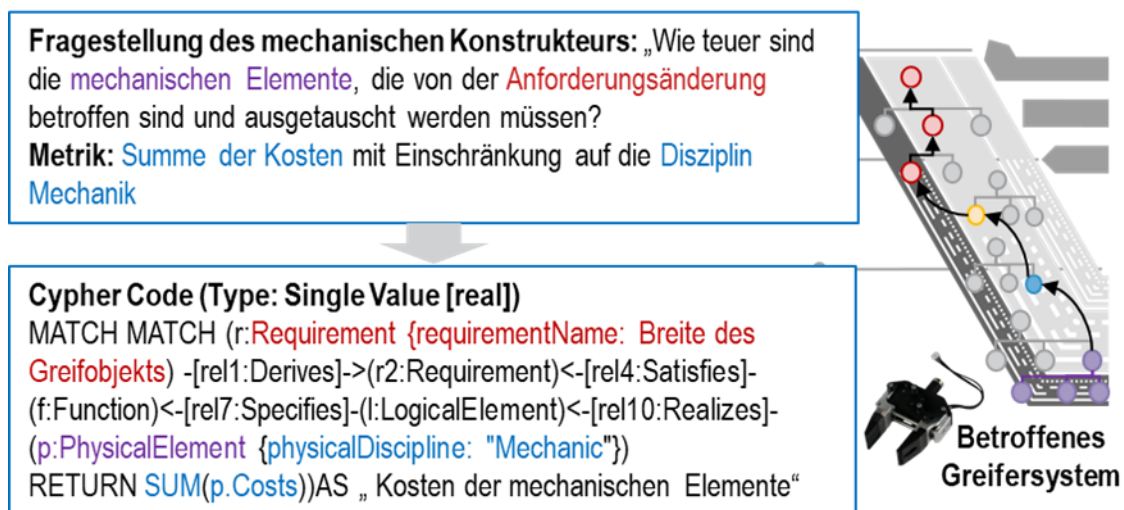


Bild 5-26: Wirkkettenanalyse für die Frage des Konstrukteurs

In der letzten Aktivität *Analyseergebnisse darstellen* werden die entwickelten Abfragen in rollenspezifischen Ergebnissichten dargestellt. Abhängig von den individuellen Bedarfen der Rolle können die notwendigen Informationen durch zusätzliche Informationen und Metriken ergänzt werden. Die Aktivität adressiert die dritte Dimension des Systemdenkens, indem die Komplexität für die Perspektive der Stakeholder durch maßgeschneiderte Ergebnissichten reduziert wird [GO22d, S. 102]. Je nach Ergebnis können unterschiedliche Repräsentationsarten gewählt werden, beispielsweise Teilausschnitte des Datenmodells, Tabellen, numerische Zahlenwerte, Diagramme oder Matrizen. Bei der Entwicklung der Ergebnissicht werden die Visualisierungsarten der Stakeholder berücksichtigt, welche in den Ergebnisdimensionen (Aktivität 1.2) definiert wurden.

Zur übersichtlichen Darstellung mehrerer rollenspezifischer Ergebnisse werden **Dashboards** entwickelt. In Bild 5-27 werden die Ergebnissichten der Stakeholder des Roboterarms schematisch in drei Dashboards dargestellt. Der Anforderungsentwickler wählt die betroffene Anforderung mit einer Drop-Down-Funktionalität aus. Abhängig von der Auswahl werden unterhalb weiterer betroffener Anforderungen angezeigt. Der Systemarchitekt identifiziert die betroffenen sicherheitskritischen Funktionen in dem Unter-Graph. Durch die Auswahl einer Funktion innerhalb des Unter-Graphen werden Metadaten der Funktion angezeigt. Dem technischen Manager werden die Namen der betroffenen Elemente tabellarisch dargestellt. Die dazugehörigen Einzelkosten (orange) und kumulierten Kosten (blau) werden in einem Liniendiagramm visualisiert. Durch Wirkkettenanalysen werden die Ergebnissichten an die rollenspezifischen Informationsbedarfe zur Bewertung technischer Änderungen angepasst. Die Einbindung von Bewertungsmetriken ergänzt existierende Abfragen und ermöglicht die quantitativen Analysen der Änderungsauswirkungen. Durch die Kombination der gefolgerten Ergebnissichten auf Basis des deklarativen Datenmodells und des vorhandenen prozeduralen Expertenwissens der Anwender können Änderungsauswirkungen effizient und effektiv bewertet werden. Ergebnis der Aktivität ist ein Satz an rollenspezifischen Ergebnissichten, welche für die qualitative und quantitative Bewertung der Auswirkungen genutzt werden.

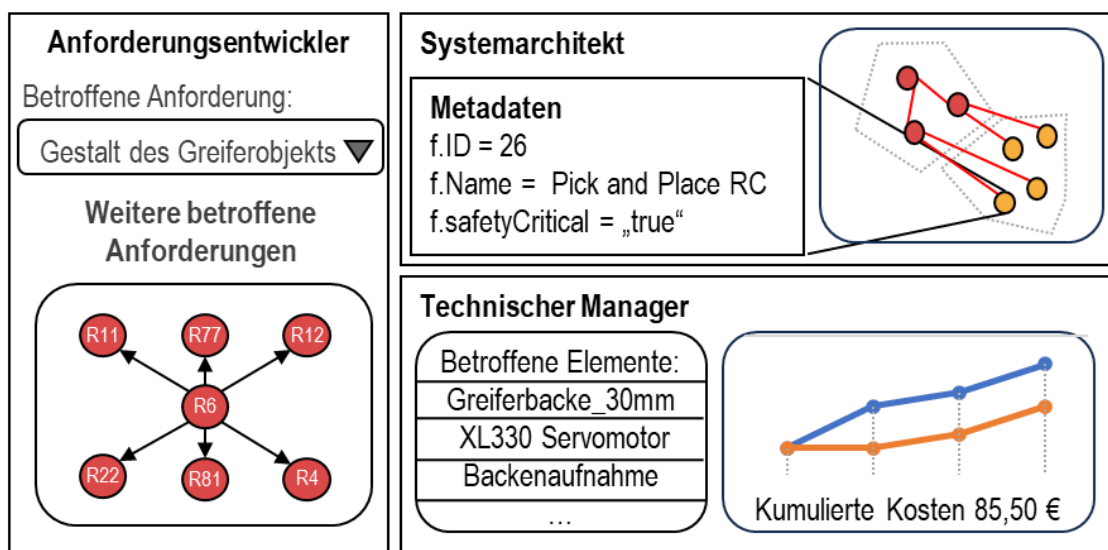


Bild 5-27: Schematische Darstellung der Dashboards für den Roboterarm

Insgesamt unterstützen die Wirkkettenanalysen Anwender bei der frühzeitigen, effizienten und effektiven Bewertung technischer Änderungen. Durch die entwickelte Methodik entstehen alle fünf Elemente für eine funktionsfähige modellbasierte Wirkkettenanalyse. Die Wirkkettenanalyse wird systematisch an die Bedarfe eines Unternehmens und seiner MBSE-Rollen angepasst. Im ersten Schritt werden alle technischen Änderungen formalisiert (Element 1), die durch die Wirkkettenanalyse bewertet werden sollen. Durch die frühzeitige Herleitung von Ergebnisdimensionen wird das erwartete Ergebnis jeder Rolle spezifiziert (Element 5). Aus den aufgenommenen Ergebnisdimensionen werden der Informationsbedarf hergeleitet und ein übergreifendes

Metamodell des Wirkkettenmodells abgeleitet (Element 3). Im zweiten Schritt werden alle benötigten Informationen über die Artefakt- und Relationstypen zusammengetragen, um ein vollständiges und konsistentes Wirkkettenmodell entwickeln zu können. Im dritten Schritt entsteht das Wirkkettenmodell durch eine Reihe von Modellierungsschritten. Das Wirkkettenmodell bildet alle Artefakt- und Relationstypen ab, wodurch Wirkpfade entstehen. Als Hilfsmittel werden ein SysML-Profil, eine Modellvorlage und ein angepasstes Diagramm entlang der zentralen Entwicklungsartefakte RFLPV² bereitgestellt. Im letzten Schritt werden Query Algorithmen zur Identifikation betroffener Elemente (Element 2) und Metriken zur qualitativen und quantitativen Bewertung (Element 4) integriert, um die rollenspezifischen Ergebnisse in den Ergebnissichten darzustellen. Die Methodik hilft Entwicklern dabei, Wirkkettenanalysen umzusetzen und informierte Entscheidungen über die Auswirkungen technischer Änderungen auf Basis einer objektiven Entscheidungsgrundlage zu treffen. Die Aktivitäten der MECA-Methodik werden auf DIN-A0-Postern übersichtlich zusammengefasst (siehe Anhang A4.10, S. 195).

5.5 Software-gestützte Implementierung

Die Anwendung der MECA-Methodik wird durch eine software-gestützte Implementierung unterstützt. Der Software-Prototyp umfasst vier Ebenen von IT-Werkzeugen und das RFLPV²-Metadatenmodell. Die vier IT-Werkzeuge der Ebenen sind (1) ein SysML-Modellierungswerkzeug mit RFLPV²-Profil zur Modellierung des spezifischen Wirkkettenmodells, (2) unternehmensspezifische IT-Werkzeuge und Verwaltungssysteme zur Ergänzung weiterer Ursache-Wirkungs-Beziehungen aus spezifischen Partialmodellen, (3) eine Property-Graph-Datenbank mit RFLPV²-Metadatenmodell zur Abbildung und Analyse des Wirkkettenmodells und (4) Graph-Applikationen zur Erstellung von rollenspezifischen Dashboards (siehe Bild 5-28). Die Auswahl der IT-Werkzeuge für die software-gestützte Implementierung basiert auf dem Prinzip „best in class“ nach BAJAJ ET. AL. [BBW17, S. 167]. Bei einer unternehmensspezifischen Adaption des Ansatzes ist die Auswahl der IT-Werkzeuge abhängig von den vorhandenen Lizenzen im Unternehmen. Beispielsweise können unterschiedliche IT-Werkzeuge wie Cameo Systems Modeler von D’sassault Systéms oder IBM Rhapsody zur SysML-Modellierung genutzt werden.

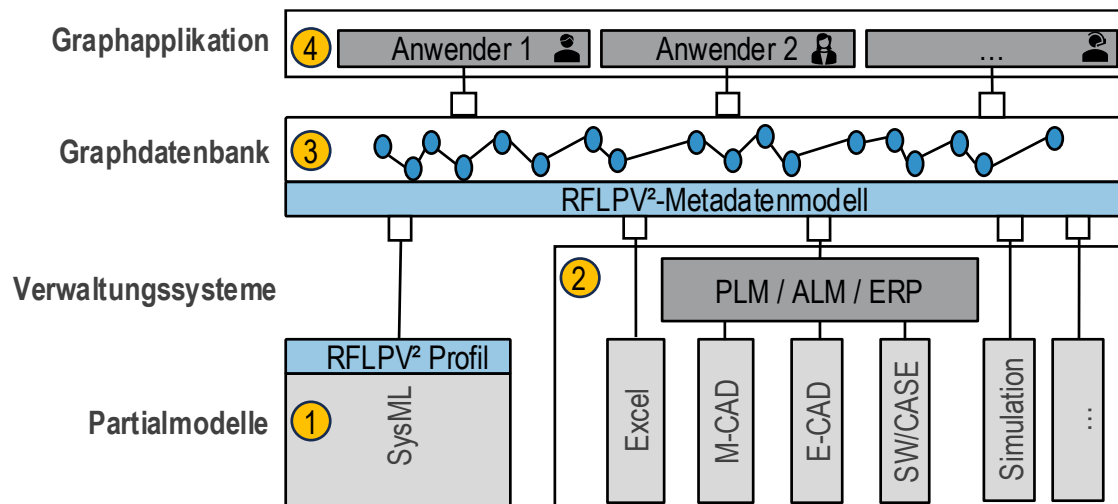


Bild 5-28: Struktur des Software-Prototypen für modellbasierte Wirkkettenanalysen entlang von vier Ebenen (in Anlehnung an [Eig09, S. 49])

Ausgangspunkt der Wirkkettenmodellierung ist ein SysML-Modell. Ausgehend von den erhobenen Ergebnisdimensionen (Schritt 1) und den zugehörigen Informationen (Schritt 2) wird ein maßgeschneidertes Wirkkettenmodell in SysML erstellt. Zur Erstellung des Wirkkettenmodells des Roboterarms wird das IT-Werkzeug Cameo Systems Modeler von D'assault Systèmes eingesetzt. Zur Vereinfachung der Modellierung im IT-Werkzeug werden drei Hilfsmittel entwickelt und bereitgestellt: eine Modellvorlage, ein SysML-Profil und ein Wirkketten-Diagramm (siehe Kapitel 5.3). In der folgenden textuellen Beschreibung werden numerische Referenzen eingeführt, um die Vorteile der entwickelten Hilfsmittel in Bild 5-29 und Bild 5-30 zu kennzeichnen.

Die **RFLPV²-Vorlage** gibt die initiale Struktur des Wirkkettenmodells vor. In einem Paketdiagramm werden die Artefakte in separaten Paketen entlang des V-Modells dargestellt (1). Die Darstellung entlang des V-Modells als Orientierungshilfe hat sich bei der Anwendung der MECA-Methodik in den Modellierungswerkshops bewährt (WS 5-7, WS 16). Je Paket wird ein individuelles Vorschaubild des Pakets festgelegt, um die Navigation in den Diagrammen und in der Verzeichnisstruktur intuitiv zu gestalten (2). Zusätzlich werden durch die Modellvorlage Namenskonventionen vorgegeben, um die Konsistenz im Wirkkettenmodell sicherzustellen. Ein Beispiel ist die absteigende Nummerierung der Pakete (01_Requirements, 02_Functions, ...) (3). Das **RFLPV²-Profil** (siehe Kapitel 5.3) wird als separate SysML Datei (mdzip.) gespeichert, und als Modul in die Modellvorlage integriert (4). Durch die Integration werden die Stereotypen zur Erweiterung der Semantik und Customizations zur Restriktion der Syntax anwendbar.

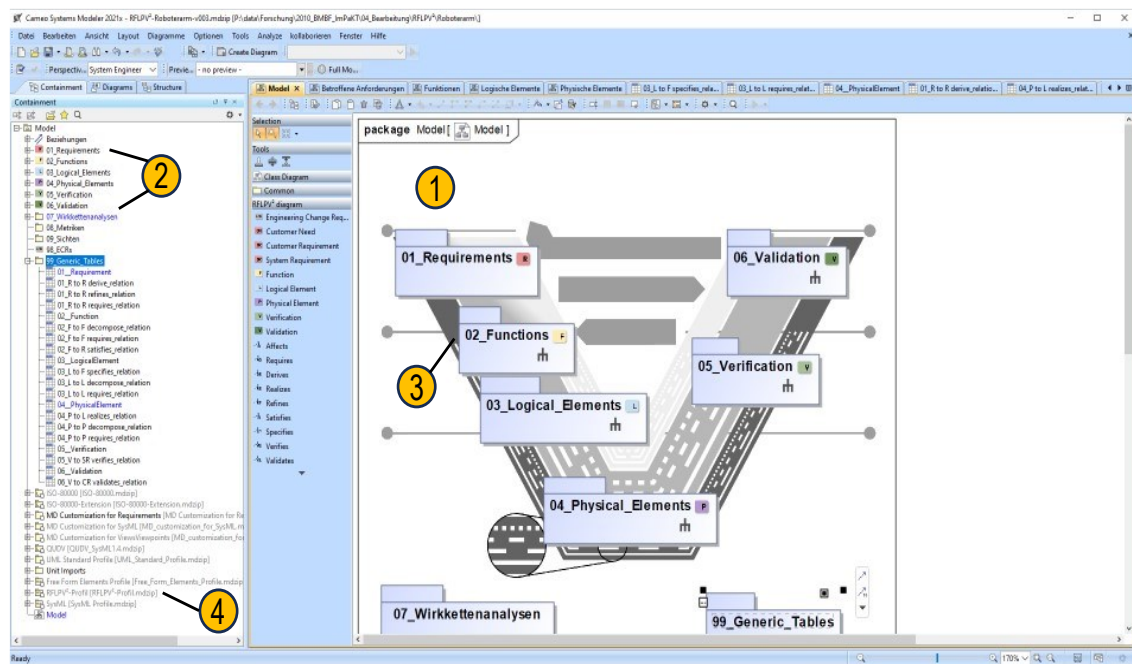


Bild 5-29: SysML Modellvorlage entlang des V-Modells der VDI 2206:2021

Zur einfachen Anwendung der neu definierten Stereotypen und Customizations wird ein individuelles SysML-Diagramm entwickelt, welches für die Wirkkettenmodellierung genutzt wird – das **RFLPV²-Diagramm**. Hierzu wird ein Klassendiagramm der UML an die Bedarfe der Wirkkettenmodellierung angepasst. In der Werkzeugleiste (5) werden neue Schaltflächen erzeugt, welche das unmittelbare Anlegen der Artefakt- und Relationstypen aus dem SysML Profil ermöglichen. Zusätzlich werden die Symboleigenschaften der Artefakte und Relationen angepasst, um eine eindeutige farbliche Unterscheidung hervorzurufen und eine intuitive Nutzung zu ermöglichen (6). Im Hintergrund des Wirkketten-Diagramms werden farbige Flächen eingeführt, um die Modellierung eines Wirkpfads intuitiv und strukturiert zu gestalten (7). Durch die <<Customizations>> (siehe Anhang A4.5, S. 186) werden die Artefakt- und Relationstypen beim Modellieren der Wirkkette innerhalb des RFLPV²-Diagramms syntaktisch eingeschränkt (8). Die ausgewählte Kundenanforderung kann per <<Derives>>-Beziehung lediglich mit einer Systemanforderung verknüpft werden, nicht mit einer Funktion, einem logischen Element oder einem physischen Element. Modellierungsfehler und Inkonsistenz werden dadurch vermieden. Das entwickelte RFLPV²-Diagramm wird in Bild 5-30 dargestellt.

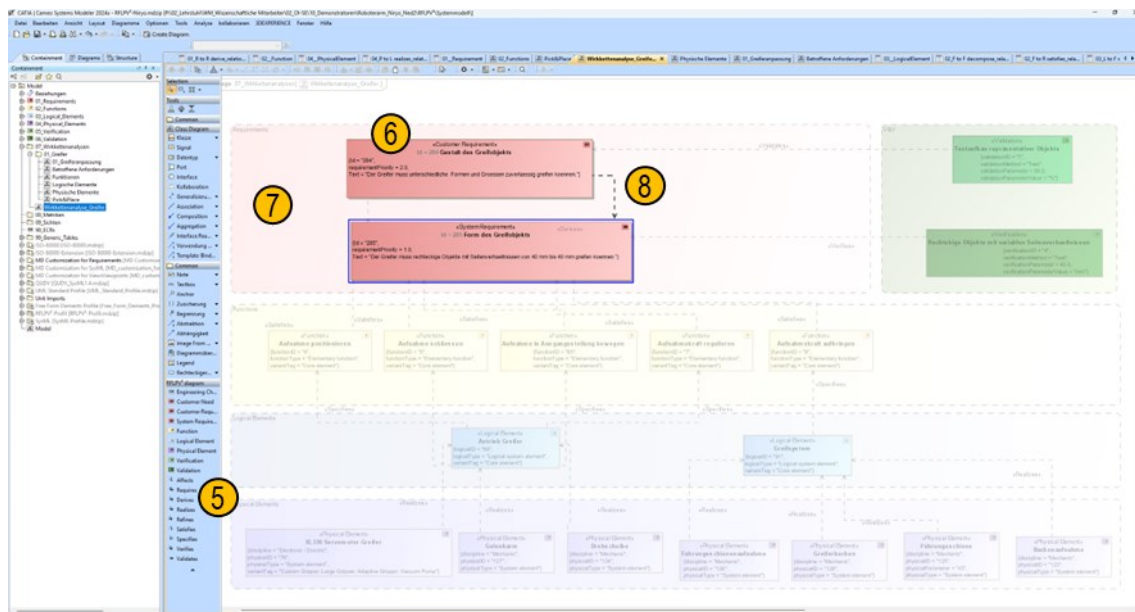


Bild 5-30: Semantische und syntaktische Anpassungen im Wirkkettenmodell

Da die Analyse der Lösungsansätze (siehe Kapitel 2.2.1) Limitationen von SysML-Modellierungswerkzeugen zur Auswertung von Änderungsauswirkungen aufgezeigt hat, wird eine Modelltransformation vorgenommen. Das SysML-Wirkkettenmodell wird exportiert, durch Daten und Informationen aus Partialmodellen ergänzt und in ein Property Graph Model überführt. Hierzu werden **Generic Tables** in die RFLPV²-Vorlage im Cameo Systems Modeler integriert, die alle Daten und Informationen der Artefakt- und Relationstypen abbilden und einen bidirektionalen In- und Export von csv.-Dateien ermöglichen. Die Struktur der generischen Tabellen wird an das Schema einer Property-Graph-Datenbank angepasst. Je Artefakt- und Relationstyp wird eine separate Tabelle erstellt. In den generischen Tabellen wird das Stereotyp als zu suchendes Element (Element type) sowie der Suchbereich (Scope) innerhalb des Wirkkettenmodells ausgewählt (1). Jede Tabelle beginnt mit einer Spalte zur Darstellung des Identifikators (ID) des Quellelements. Bei Artefakten werden in den folgenden Spalten die Attribute ergänzt (2); bei Relationen die ID der Zielelemente (3). Zur Herleitung der Tabellenspalten wird der Metachain-Navigator (4) genutzt. Je Spalte werden in einer **Metachain** Operationen für die Analyse des Wirkkettenmodells festgelegt, um die relevanten Daten des Artefakt- oder Relationstyps zusammenzutragen (5). Beispiele aus dem Wirkkettenmodell des Roboterarms sind in Bild 5-31 dargestellt und numerisch referenziert.

Tabelle für Artefakt

Criteria
Element Type: Physical Element Scope (optional): 04_Physical_Elements 1

Excel Import Status: ☐ New ☐ Updated ☐ Obsolete ☐ Unchanged

#	physicalID	physicalName	physicalDiscipline	physicalElementOwner	physicalParameter	physicalParameterType	physicalType	physicalCosts
1	27	Sechskantmutter M3 - DIN 934 - Messing	Mechanic	Deniz Oezcan	M3		System element	0.03
2	20	Netzteil 12V DC - 100W	Electronic / Electric	Daniel Preuss	100	W	System element	11.54
3	41	Zylinderstift - 8 x 80 mm - Stahl - gehärtet	Mechanic	Deniz Oezcan	80	mm	System element	0.11
4	32	Stirnrad aussenverzahnt - 60 mm - Modul 1,5 - zerschnitten	Mechanic	Deniz Oezcan	60	mm	System element	0.3
5	40	Zahnkranz innenverzahnt - 60 mm - Modul 1,5	Mechanic	Sven Rarbach	60	mm	System element	0.54
6	26	Schraube SHCS M3 x 40 mm	Mechanic	Sven Rarbach	40	mm	System element	0.02
7	7	Druckfeder - EN 10270-1 - 6 x 20 mm	Mechanic	Deniz Oezcan	20	mm	System element	0.02

Tabelle für Relation

Criteria
Element Type: Part Property Scope (optional): 04_Physical_Elements Filter: 2

#	physicalID	physicalID_decompose
1	1	37
2	2	36
3	3	1
4	4	1
5	5	10
6	6	38
7	7	15
8	10	38
9	11	39
10	12	15
11	13	36
12	14	15
13	42	15
14	15	35
15	16	8
16	17	8
17	18	8
18	19	21
19	19	10
20	19	39

3

Expression

Specify name and expression for the custom column.
Select and specify operations for calculating/gathering values of custom column. If several operations are specified, custom column will contain results of all of these operations.

Name: physicalID_decompose 4

Ausdruck:

- Expression
- Simple Navigation
- Metachain Navigation
- Find
- Implied Relation
- Create operation...

Metachain Navigation

Metaklasse oder Stereotyp	Eigenschaft oder Notiz
PartProperty [Property]	Gegenteil
Property	Typ
Physical Element [Class]	physicalID 5

OK Abbrechen Hilfe Evaluation Mode

Bild 5-31: Metachain Navigation für „Decompose“ Relation von Physischen Elementen

Die exportierten csv. Dateien aus dem SysML Modell werden durch Daten aus Partialmodellen ergänzt, welche entweder dezentral gespeichert oder in PLM- und ERP-Systemen verwaltet werden. Die Prämisse zur Integration der Daten ist, dass durch das IT-Werkzeug ein Export der Daten als csv.-Datei möglich ist. Für den Roboterarm können so Anforderungen aus Requirements Engineering Werkzeugen, Materialeigenschaften aus Geometriemodellen, Stücklisten aus einem PLM-System und Simulationsergebnisse aus Simulationswerkzeugen integriert werden (siehe Anhang A5.2, S. 204). Die exportierten csv.-Dateien müssen innerhalb des Metadatenmodells verortet werden, um in die Graphdatenbank des Eigenschaftsgraphens importiert zu werden.

Für die Wirkkettenanalyse wird die angereicherte Datengrundlage in eine **Property-Graph-Datenbank** überführt. Als Graphdatenbank wird neo4j Aura DB verwendet, da eine kostenfreie Instanz erstellt werden kann und Graphapplikationen zur Visualisierung und Analyse genutzt werden können. Auf der linken Seite des Import-Fensters werden die csv.-Dateien aufgerufen (1). In der Mitte ist das Metadatenmodell der Graphdatenbank abgebildet (2). Für den Import in Neo4j wird ein RFLPV²-Metadatenmodell bereitgestellt (siehe Bild 5-32). Basierend auf dem Metadatenmodell können die csv.-Dateien hochgeladen und in die Graphdatenbank überführt werden. Auf der rechten Seite können die Artefakt- und Relationstypen sowie deren Eigenschaften angepasst werden (3). Durch die Run Import Funktion wird das Property-Graph-Modell erzeugt.

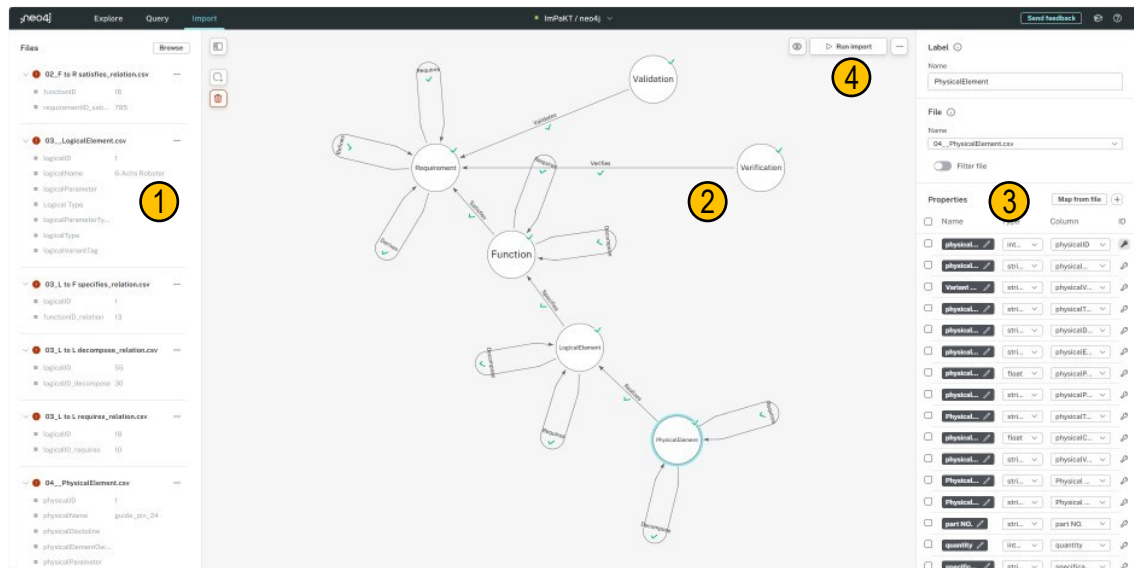


Bild 5-32: Exemplarische Integration des (Meta)datenmodells in Neo4j Aura

Das entstehende Graphmodell G des Roboterarms lässt sich in einem Tupel $(V, E, \rho, \lambda, \sigma)$ beschreiben (siehe Anhang A4.7, S. 188). Exemplarisch wird im Folgenden der Teilausschnitt in Bezug zu den Requirements und den Beziehungen untereinander beschrieben:

$V = \{Requirement\}$, $E = \{Requires, Refines, Dervies\}$, $\rho: = (rel1:Requires) = (Requirement, Requirement1)$, $(rel2:Refines) = (Requirement, Requirement1)$, $(rel3:Dervies) = (Requirement, Requirement1)$, λ : Die Labels der Knoten und Kanten entsprechen den Artefakt- und Relationstypen. σ : Requirement {requirementID, requirementName, requirementText, requirementType, Requirement Parameter, Requirement Value Type, Requirement Priority)

Nach dem Hochladen der Dateien in neo4j Aura können Wirkkettenanalysen in Cypher [Neo22] formuliert, ausgeführt und wiederverwendet werden.

Basierend auf dem so entstehenden Property-Graph-Modell in neo4j Aura DB wurden **Dashboards in der Graphapplikation** NeoDash entwickelt. Die rollen-spezifischen Dashboards bestehen aus einzelnen konfigurierbaren Komponenten (Widgets). Je Dashboard können neue einzelne Widgets angelegt werden (1), welche eine Abfrage eines Wirkpfads, Bewertungsmetriken und eine ausgewählte Visualisierungsart umfassen sowie eine automatisierte gefolgerte Ergebnisdimension repräsentieren (2). Die konfigurierbaren Dashboards erfüllen daher den steigenden Bedarf nach personalisierbaren IT-Werkzeugen, der auch im ECM vorherrscht [Sta22, S. 410]. Durch Cypher können ebenfalls Metriken in Abfragen integriert werden, um mathematische Berechnungen durchzuführen und Auswirkungen technischer Änderungen zu quantifizieren. Je Widget werden rollenspezifische Informationen zur Beantwortung einer oder mehrerer Ergebnisdimensionen bereitgestellt. Ebenfalls kann in Abhängigkeit der Ergebnisdimension eine passende Visualisierungsmöglichkeit gewählt werden. Exemplarisch dargestellte Visualisierungsmöglichkeiten sind Dropdown-Menüs zur Auswahl (3), einzelne

quantifizierte Werte (4) oder Graphausschnitte (5). Weitere Visualisierungsmöglichkeiten sind unter anderem Tabellen, Säulendiagramme, Kreisdiagramme, Liniendiagramme und Quellcode. Jedes Dashboard bildet somit verschiedene Sichten auf das Wirkkettenmodell ab und ermöglicht die qualitative und quantitative Analyse und Bewertung der Auswirkungen einer technischen Änderung. Beispielsweise werden in dem exemplarischen Dashboard in Bild 5-33 Sichten für Rolle 1, 2 und 3 vorgesehen (6).

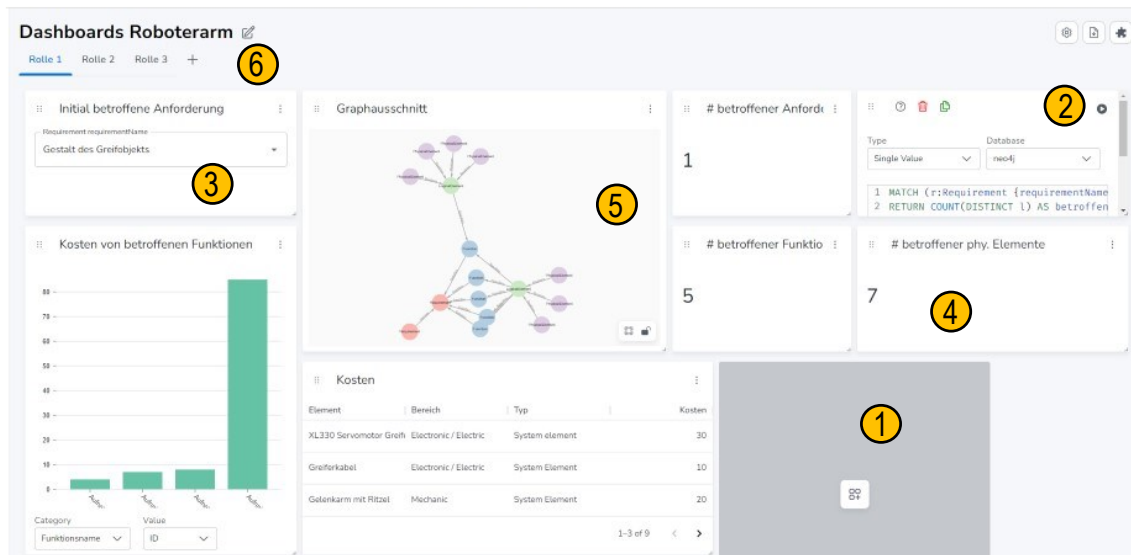


Bild 5-33: Exemplarisches Dashboard für Wirkkettenanalyse NeoDash

Der Code des Dashboards wird bereitgestellt (siehe Anhang A11, S. 260). Eine zusätzlich nutzbare Funktionalität ist die KI-gestützte Herleitung von Cypher-Abfragen durch die integrierten Schnittstellen zu Large-Language-Models (LLM), beispielsweise ChatGPT von OpenAI [Ope24]. Die KI-Modelle transformieren in den Anwendungsfall natürlichsprachlich Fragen automatisiert in die Abfragesprache Cypher.

Die beschriebene software-gestützte Implementierung vereint IT-Werkzeuge, um schnell und zielgerichtet ein Wirkkettenmodell zu erstellen, aussagekräftige Wirkkettenanalysen durchzuführen und Ergebnisse in rollenspezifischen Dashboards bereitzustellen. Das Erstellen des Wirkkettenmodells wird durch drei Hilfsmittel vereinfacht. Das RFLPV2-Metadatenmodell, generische Tabellen und das Datenaustauschformat csv. werden zur Ergänzung von Daten aus Partialmodellen und zur Transformation in ein Property-Graph-Modell genutzt. Die gewählte Abfragesprache Cypher eignet sich für das Erstellen eigener qualitativer und quantitativer Wirkkettenanalysen, unabhängig von der Art des ECRs. Das Speichern und die Wiederverwendung von Abfragen durch Basisalgorithmen reduzieren den Entwicklungsaufwand zur Erstellung der Wirkkettenanalysen. Mit Neodash werden rollen-spezifische Dashboards erstellt und an die individuellen Bedarfe der Nutzer angepasst, um die akteur-übergreifende Kollaboration zu gewährleisten.

Alle hergeleiteten Anwendungsfälle (Kapitel 3) werden durch die software-gestützte Implementierung funktional umgesetzt. Im Folgenden wird die funktionale Umsetzung der 18 Anwendungsfälle erklärt (siehe Nummerierung im Anhang A3.2, S. 177). In der software-gestützten Implementierung wird die Graphapplikation Neodash als *aufzufbares Analysetool* verwendet (1). Innerhalb der Graphapplikation kann eine *Analyse einfach oder mehrfach ausgeführt* werden (2). Das *Ziel der Analyse* (3) wird festgelegt, indem der initial betroffene Knoten mithilfe eines Drop-Down Menüs und einer Suchfunktion identifiziert wird oder explizit in der Abfrage beschrieben wird. Das *Festlegen der Analysedaten* (4) erfolgt durch den Upload der zusammengetragenen Informationen im csv.-Datenformat. Die *Daten des ECRs* werden innerhalb eines separaten ECR-Knotens *eingegeben* (5) und *gespeichert* (6). Die *Analyse der Auswirkungen* (7) erfolgt durch die Nutzung von Queries, welche die Wirkpfade und die festgelegten Bewertungsmetriken abbilden. Die *Anpassung der Analyse* (8) erfolgt durch die individuelle Festlegung des Wirkpfads und der Bewertungsmetriken, welche an die Bedarfe der Anwender angepasst werden. Das *Eingeben von Expertenwissen* (9) erfolgt durch separate Widgets, mit denen Knoten und Kanten manipuliert und aktualisiert werden können. Die *Ergebnisse* können *gespeichert* (10) und *exportiert* (11) werden, beispielsweise als Bilddatei des Graphen oder als json.-Datei zum rechnerinterpretierbaren Austausch der Ergebnisse. Das *Sichten der Ergebnisse* (12) erfolgt anhand der Dashboards. *Fehlende Informationen* (13) oder *nicht interpretierbare Daten* (14) werden durch Fehlermeldungen gekennzeichnet. Aufgrund der Nutzung der Abfragesprachen Cypher und der Flexibilität des zugrunde liegenden Metadatenmodells können *unterschiedliche Auswirkungen angezeigt* (15) werden. Hierzu zählen unter anderem *Auswirkungen auf die Anforderungserfüllung* (16) durch die Betrachtung von Ergebnissen aus Eigenschaftsabsicherungen, *Auswirkungen auf Kosten* (17) durch die Betrachtung der physisch betroffenen Elemente und *Auswirkungen auf Varianten* (18) durch die varianten-spezifische Differenzierung der Knoten. Alle hergeleiteten Anwendungsfälle können durch die prototypische Implementierung der modellbasierten Wirkkettenanalyse umgesetzt werden.

5.6 Einbindung in die Produktentwicklung

Für die Einbindung der MECA Methodik in die Produktentwicklung eines Unternehmens wird ein Referenzprozess bereitgestellt. Der Referenzprozess besteht aus den drei Schritten: Der Identifikation des Anwendungspotenzials (1), der Einbindung der Methodik in die Prozess- und IT-Landschaft (2) und die initiale Anwendung in einem Pilotprojekt (3) (siehe Bild 5-34).

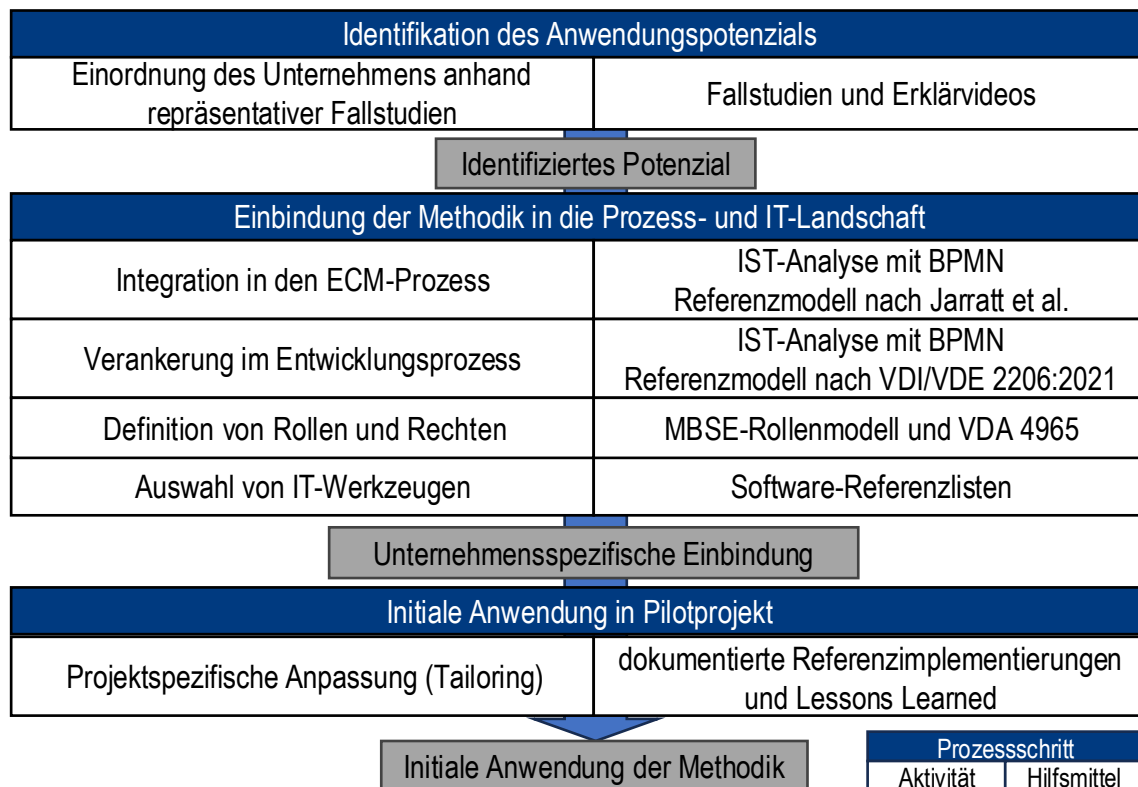


Bild 5-34: Referenzprozess zur Einbindung der Methodik in Unternehmensprozesse

Identifikation des Anwendungspotenzials: Vor der Anwendung der MECA-Methodik wird das unternehmensspezifische Anwendungspotenzial abgeschätzt. Hierzu wird eine Einordnung des Unternehmens anhand repräsentativer Fallstudien aus den Bereichen Landtechnik, Sondermaschinenbau, Spritzgusstechnik und Robotik vorgenommen. Die Anwendung der MECA-Methodik für technische Änderungen an den unterschiedlichen Beispielprodukten veranschaulicht die entstehenden Potenziale. Für jedes Fallbeispiel werden ein **Erklärvideo** und die zugehörige Präsentation bereitgestellt, um die Anwendung der MECA-Methodik nachzuvollziehen und das Potenzial abzuschätzen.

Einbindung der MECA-Methodik in die Prozess- und IT-Landschaft: Nach der Abschätzung des Potenzials wird entschieden, ob die MECA-Methodik angewendet und in die Unternehmensprozesse eingebunden werden soll. Das ECM ist ein begleitender Prozess während des Produktentstehungsprozesses. Daher müssen die unternehmensspezifischen Prozesse analysiert und angepasst werden. Für die IST-Analyse werden beide Prozesse mithilfe der formalen Modellierungssprache Business Process Modeling Notation 2.0 (BPMN) [Obj11] abgebildet. Zusätzlich werden Referenzmodelle der Prozesse verwendet, um eine Einordnung und eine Vergleichbarkeit der Prozesse zu gewährleisten. Für das ECM wird der generische **ECM-Prozess nach JARRATT ET AL.** [JEC11, S. 107] verwendet. Die Wirkkettenanalyse wird in den ECM Prozess integriert und fungiert als Hilfsmittel zur Entscheidungsunterstützung zum Umgang mit technischen Änderungen. Als Referenzprozess für die Produktentwicklung wird die **VDI/VDE 2206:2021** [VDI21] zugrunde gelegt. Im V-Modell lassen sich Zeitpunkte

identifizieren, zu denen Änderungen auftreten können und Wirkkettenanalysen durchgeführt werden müssen. Die schematische Durchführung der BPMN Modellierung sowie der Identifikation der Schnittstellen ist in Bild 5-35 dargestellt.

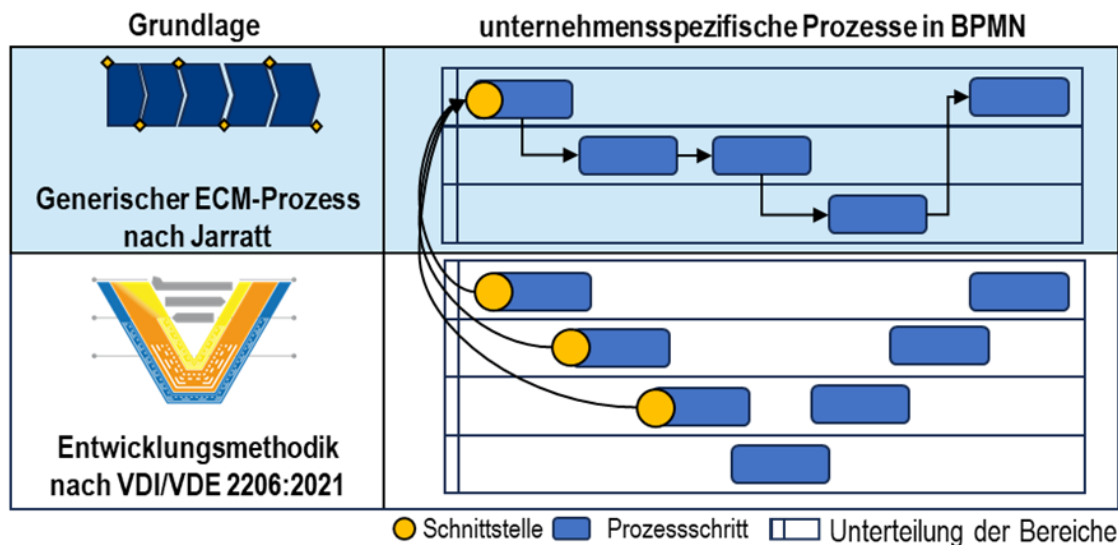


Bild 5-35: Produktentwicklungsmodell und Querschnittsaktivitäten (aufbauend auf Modellen nach [JEC11, S. 107, VDI21, S. 22])

In dem ImPaKT-Forschungsprojekt wurden die Produktentstehungsprozesse und ECM von vier industriellen Unternehmen untersucht, um Schnittstellen zwischen den Prozessen zu identifizieren, an denen regelmäßig technische Änderungen auftreten [TIB25, S. 5]. Folgende fünf Punkte wurden identifiziert: Der erste Punkt liegt bei der Erstellung der Spezifikation (1). Die Spezifikation wird iterativ mit dem Kunden erarbeitet, wodurch Anforderungsänderungen auftreten können. Der zweite Punkt liegt bei der Erstellung der Systemarchitektur, welche interdisziplinäre Funktionen und logische Elemente umfasst (2). Während der Entwicklung der Systemarchitektur treten regelmäßig technische Änderungen auf, beispielsweise durch eine Änderung einer Funktion oder die Auswahl eines alternativen logischen Elements. Der dritte Punkt liegt bei der Implementierung der Systemelemente (3). Beispiele sind die Änderung eines Durchmessers, die Änderung eines Schaltplans oder die Anpassung des Softwarecodes. Nach der anschließenden Integration können technische Änderungen zusätzlich der Durchführung von Verifikations- und Validierungsmaßnahmen dienen (4) und bei der Übergabe auftreten (5). Die Punkte sind in Bild 5-36 dargestellt und überschneiden sich mit Beobachtungen von PIKOSZ und MALMQUIST [PM98, S. 2]

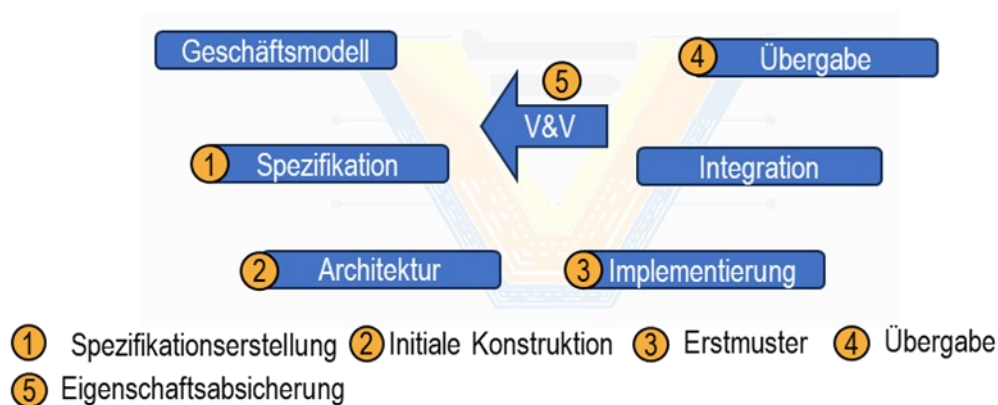


Bild 5-36: Änderungsanfragen entlang der Kontrollpunkte des V-Modells (in Anlehnung an [TIB25])

Durch die unternehmensspezifische Identifikation der Schnittstellen zwischen dem Entwicklungsprozess und dem ECM-Prozess wird die Wirkkettenanalyse in dem Entwicklungsprozess prozessual verankert. Zusätzlich zur prozessualen Betrachtung müssen organisatorische **Anpassungen der existierenden Rollen und Rechte** vorgenommen werden, um die beteiligten Personen in die Wirkkettenanalyse einzubinden. Basierend auf der VDA 4965 [VDA09b, S. 13f.] werden drei Rollen unterschieden und Personen zugeordnet: Der Antragsempfänger (1) nimmt Änderungsvorschläge entgegen und entscheidet, ob eine Wirkkettenanalyse durchgeführt werden soll. Der Änderungsantragsteller (2) erstellt den ECR in einer formalisierten und rechnerinterpretierbaren Art. Der EC Manager (3) ist für die Koordination des ECRs und die Koordination der Wirkkettenanalyse verantwortlich. Je nach Unternehmen können diese Verantwortlichkeiten aufgeteilt oder zentralen Schlüsselrollen zugeordnet werden.

Neben der prozessualen Einbindung müssen die Anwender ebenfalls **IT-Werkzeuge auswählen**, um die modellbasierte Wirkkettenanalyse durchzuführen.

Nach der Vorbereitung und Einbindung der MECA-Methodik in die Prozess- und IT-Landschaft wird die MECA-Methodik initial in einem Pilotprojekt angewendet. Durch den zielorientierten Aufbau der MECA-Methodik werden die Wirkkettenanalysen an die identifizierten Bedarfe und unternehmensspezifischen Rahmenbedingungen angepasst. Zusätzlich werden vier Referenzimplementierungen der MECA-Methodik bereitgestellt, welche Anwender bei der initialen Anwendung unterstützen. Hierzu zählen drei Fallstudien aus dem Forschungsprojekt ImPaKT und eine Fallstudie aus der Anwendung der MECA-Methodik im Automobilbereich. Die Referenzimplementierungen wurden in Erklärvideos zusammengefasst. Die Erklärvideos sind unter folgendem Link abrufbar: <https://www.impaktprojekt.de/impakt-erklervideos>

Die beschriebenen Aktivitäten und Referenzmodelle des Referenzprozesses zur Einbindung der MECA-Methodik unterstützen Anwender bei der Abschätzung der Potenziale für das Unternehmen, bei der Einordnung in die Prozess- und IT-Landschaft und bei der initialen Anwendung der Methodik in einem Pilotprojekt.

6 Validierung in der industriellen Praxis

In diesem Kapitel wird die MECA-Methodik mithilfe der software-gestützten Implementierung im Anwendungszusammenhang validiert. Grundlage der Validierung sind die hergeleiteten Anforderungen und Anwendungsfälle, welche praktische Defizite im Umgang mit den technischen Änderungen adressieren. Zur Durchführung der Validierung wird ein Validierungskonzept (siehe Tabelle 6-1) entwickelt. In dem Validierungskonzept wird je Anforderung ein Erfüllungskriterium und die Bewertungsgrundlage festgelegt. Für jede Bewertungsgrundlage werden Bewertungsmethoden aus den Sammlungen nach HEVNER ET AL. [HMP04, S. 87] und BLESSING UND CHAKRABARTI ([BC09, S. 243] aufbauend auf [Ble94]) ausgewählt (siehe Anhang A10.1, S. 247).

Da der Anforderungsbewertung häufig **Beobachtungen** im Anwendungszusammenhang zugrunde liegen, wurde die MECA-Methodik in **industriellen Fallstudien (FS) angewendet**. Ausgehend von den praktischen Herausforderungen der Anwender im Umgang mit technischen Änderungen wird in den Fallstudien ein **deskriptives Szenario** der Wirkkettenanalyse mit den Anwendungspartnern entwickelt. Jedes Szenario umfasst eine realistische komplizierte technische Änderung aus dem operativen Geschäft. Je deskriptivem Szenario werden die unternehmensspezifischen Ergebnisdimensionen definiert und die benötigten rollenspezifischen Wirkkettenanalysen mithilfe der MECA-Methodik entwickelt. Neben den beschriebenen Bewertungsmethoden wurden halbjährige Validierungs-Workshops durchgeführt (WS 4, 8, 11, 13). Im abschließenden Validierungsworkshop wurden **kontrollierte Experimente** durchgeführt, in denen die MECA-Methodik erklärt, von den Industriepartnern eigenständig angewendet und abschließend bewertet wurde. Zur Bewertung wurde eine **Befragung der Industrieanwender** mithilfe eines Validierungs-Fragebogens durchgeführt. Die Experimente und die Fragebögen dienen der Prüfung des Anwendungsaufwands, der Anwendbarkeit der softwaregestützten Implementierung, der Genauigkeit der Ergebnisse und des Mehrwerts der Methodik im Anwendungszusammenhang. Zusätzlich zu dem kontrollierten Experiment wurden funktionale und strukturelle Tests zur Bewertung der software-gestützten Implementierung durchgeführt. In den **funktionalen Tests** wird nachgewiesen, dass die geforderten Funktionalitäten aus den Anwendungsfällen (siehe Kapitel 3) software-gestützt umgesetzt wurden. In den **strukturellen Tests** wurde geprüft, ob alle Änderungskategorien bewertet, alle Ergebnisdimensionen transformiert und Wirkpfade abgelaufen werden können.

Folgend werden deskriptive Szenarien von drei Fallstudien aus dem ImPaKT Projekt dargestellt (Kapitel 6.1). Ergänzend wird ein deskriptives Szenario aus der Fallstudie zur Wirkkettenmodellierung eines Kraftfahrzeugs bei einem Automobil-OEM (Kapitel 6.2) betrachtet. Aufgrund der Vielzahl konzipierter, durchgeführter und analysierter Workshops im ImPaKT Projekt (siehe Anhang A1, S. 139) werden die ersten drei deskriptiven Szenarien bei der Bewertung priorisiert betrachtet. Aufbauend auf den Beschreibungen werden die Ergebnisse der Validierung je Anforderung beschrieben (Kapitel 6.3) und der Erfolg der Methodik validiert (Kapitel 6.4).

Tabelle 6-1: Validierungskonzept

	Anforderungsname	Erfüllungskriterium	Bewertungsgrundlage
Eingangsdaten	A1: Verfügbarkeit von Daten	Verfügbarkeit der Daten > 75 %	Verfügbarkeit der Daten in Modellierungsworkshops der Fallstudien 1-3; Befragung der Industrieanwender
	A2: Unterschiedliche Arten der Änderung	100 % aller Änderungsarten	Analyse der Queries und Basisalgorithmen von Fallstudie 1-3
Methoden, Modelle, Hilfsmittel	A3: Tiefe, Breite und Weite des Wirkkettenmodells	100 % der Kategorien zur Tiefe, Breite und Weite Likert-Skala > 0 1	Walkthrough durch Wirkkettenmodelle der Fallstudien 1-5; Befragung der Industrieanwender
	A4: Änderungsausbreitung	100 % der geforderten Wirkpfade	Funktionsprüfung der Wirkpfade in Fallstudien 1-3
	A5: Objektive Bewertung	100% Umsetzung objektiver Bewertungskriterien	Analyse der Bewertungskriterien von Fallstudien 1,2, 3 und 5
	A6: Prozessuale Verankerung	Nachweis der Integrierbarkeit Likert-Skala $\geq 0 1$	Prüfung der Integrierbarkeit in Fallstudien 1-3; Befragung der Industrieanwender
Anwendbarkeit	A7: Akzeptabler zeitlicher Anwendungsaufwand	Zeiten in kontrollierten Experimenten Likert-Skala $\geq 0 1$	Messung des Zeitaufwands in Fallstudien 1-3; Automatisierungsgrad; Befragung der Industrieanwender
	A8: Einfache Anwendbarkeit	SuS > 70 Punkte	Befragung der Industrieanwender durch SuS-Fragebogen, Analyse der Diskussionen im kontrollierten Experiment
	A9: Kollaborative Anwendbarkeit	Kollaborierende Rollen je Fallstudie > 2	Analyse der rollenspezifischen Dashboards in den Fallstudien 1,2,3 und 5
Ergebnisse	A10: Maßgeschneiderte Ergebnisse	100 % der Ergebnisdimensionen Likert-Skala $\geq 0 1$	Analyse rollenspezifischer Dashboards in Fallstudien 1,2, 3 und 5; Befragung der Industrieanwender
	A11: Ergebnisqualität	100 % der Ergebnisdimensionen Likert-Skala $\geq 0 1$	Vergleich mit Ergebnisdimensionen in Fallstudien 1-4; Befragung von Industrieanwendern
	A12: Nachvollziehbarkeit	Likert-Skala $\geq 0 1$	Walkthrough durch Wirkpfade in kontrollierten Experimenten der Fallstudien 1-3; Benchmarking, Befragung der Industrieanwender
Nummerierung der ausgewählten Fallstudien: 1: Raupenlaufwerk; 2: Auswuchtmaschine; 3: Spritzgussartikel 4: Fensterheber, 5: kollaborativer Roboterarm			

6.1 Fallstudien aus den Bereichen Landtechnik, Spritzgusstechnik und Sondermaschinenbau

Die MECA-Methodik wurde im Kontext des Verbundforschungsprojekts ImPaKT in drei industriellen Fallstudien in den Branchen Landtechnik, Spritzgusstechnik und Sondermaschinenbau angewendet. Die betrachteten Systeme sind ein Raupenlaufwerk einer Landmaschine, eine Auswuchtmaschine und ein Spritzgussartikel eines Ladesteckers aus dem Bereich Elektromobilität. Das **Raupenlaufwerk** des Unternehmens CLAAS Industrietechnik GmbH (CIT) wird in verschiedenen Landmaschinen wie Mähdreschern, Traktoren und Feldhäckslern verbaut. **Auswuchtmaschinen** werden von Unternehmen unterschiedlicher Branchen benötigt und in Produktionslinien integriert. Das Unternehmen Hofmann Mess- und Auswuchttechnik GmbH & Co. KG stellt unterschiedliche Varianten von Auswuchtmaschinen nach dem Baukastenprinzip her. In der dritten Fallstudie der Hadi-Plast GmbH & Co. KG liegt der Fokus auf einer technischen Änderung an einem thermoplastischen **Spritzgussartikel**, welcher für ein Ladesteckersystem zur Energieversorgung elektrisch betriebener Kraftfahrzeuge gefertigt wird. In Kooperation mit den Werkzeugbauern wird die Produktion des Spritzgussteils gewährleistet. Weitere Details zu der Fallstudie werden in Übersichtspostern zusammengefasst (siehe Anhang A6.1, S. 216). Die Beschreibung der deskriptiven Szenarien unterteilt sich in die Vorbereitung der Wirkkettenanalyse und die Durchführung der Wirkkettenanalyse für einen spezifischen ECR. Die Vorbereitung der Wirkkettenanalyse umfasst das Vorgehen zur Durchführung der ersten drei Schritte der MECA-Methodik: Zieldefinition (Schritt 1), Informationserhebung (Schritt 2) und Wirkkettenmodellierung (Schritt 3). Das Vorgehen wurde für alle Fallstudien identisch durchgeführt und wird daher zusammengefasst.

Anschließend wird je Szenario im letzten Schritt der MECA-Methodik die Wirkkettenanalyse der ausgewählten technischen Änderung beschrieben (Schritt 4). Eine vollständige Übersicht der Fallstudien ist im Anhang dargestellt (siehe Anhang 6.1, S. 216).

Zunächst wurden die Systemgrenzen in den Fallstudien definiert. Das Raupenlaufwerk wird als separates System betrachtet, ebenso die Auswuchtmaschine. Der Spritzgussartikel wird mitsamt dem Spritzgusswerkzeug und Einstellparameter der Spritzgussmaschine betrachtet. Je Fallstudie wird ein ECR festgelegt, für welchen Wirkkettenanalysen entwickelt werden (siehe Bild 6-1). Die Auswahl der technischen Änderungen beruht auf Diskussionen zu historischen ECRs (siehe A7-A9, S. 219) und aktuellen ECRs in den Unternehmen. Für den ausgewählten ECR wurden Ergebnisdimensionen aus der Sicht relevanter MBSE-Rollen (siehe Kapitel 4.3) definiert. Für die Modellierung wurde die höchste Modellierungsintensität ausgewählt und separate Metadatenmodelle auf Basis der RFLPV²-Artefakte definiert. Das Ergebnis des ersten Schritts sind ECRs für die Fallstudien sowie die Ausprägungen des Metadatenmodells.

Ausgehend von den allgemeinen Zieldefinitionen (Schritt 1) wurden Informationen in kontinuierlichen Modellierungsworkshops (MW-1, MW-2, MW-3) aufgenommen, ergänzt und hinsichtlich der Informationsqualität überprüft (Schritt 2 in Anhang A6.3).

S. 218). Für die Erhebung wurden Hilfsmittel wie Design Structure Matrizen [Epp12, S. 6] und Modellierungsfunktionalitäten in dem IT-Werkzeug Miro verwendet. Anschließend wurden die Wirkkettenmodelle der Fallbeispiele in SysML mithilfe der entwickelten Hilfsmittel (siehe Kapitel 5.3) und der software-gestützten Implementierung (siehe Kapitel 5.5) abgebildet und in die Graphdatenbank neo4j übertragen. Beispielbilder der Fallstudien, die betrachteten ECRs und Partialmodelle sind in Bild 6-1 dargestellt.


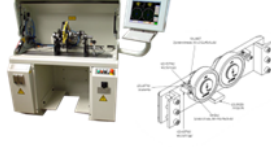
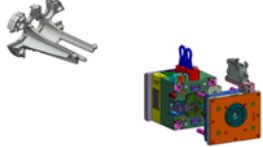
Name	Raupenlaufwerk	Auswuchtmaschine	Spritzgussartikel
Bild			
Betrachteter ECR	Änderung des Achsenabstands des Stützradpendels	Änderung des Rotordurchmessers	Änderung des Materials des Spritzgussartikels
Vorliegende Partialmodelle	Lastenheft, Stückliste, Geometriemodell, Kalkulationsmodell	Lastenhefte, Stückliste, ERP-Exporte, PLM-Exporte	Werkzeuglastenheft, Funktionslisten, Geometriemodelle, DIN-Normen, Kalkulation

Bild 6-1: Fallstudien aus dem ImPaKT Forschungsprojekt

6.1.1 Fallstudie 1: Raupenlaufwerk einer Landmaschine

Bei der Entwicklung des Raupenlaufwerks müssen Entwickler bei der Claas Industrietechnik GmbH (CIT) zum einen Anforderungen und Restriktionen der jeweiligen Landmaschinen berücksichtigen, zum anderen aber auch Restriktionen der eigenen Zulieferer. Das Raupenlaufwerk besteht aus verschiedenen Subsystemen wie dem Antriebs- und Abtriebsrad und zwei stützenden Rädern zwischen diesen – den sogenannten Midrollern (siehe Anhang A7, S. 220). Ein Zulieferer von CIT ist die Eisengiesserei Baumgarte GmbH (EB), welche unter anderem das Stützradpendel für die Midroller fertigt. In der Entwicklung von CIT wird festgelegt, dass das Raupenlaufwerk auf einer neuen Landmaschine angebracht werden soll. Da die Landmaschine eine neue Aufhängung besitzt, muss der Achsenabstand der Midroller angepasst werden. Hierzu wird die erste Wirkkettenanalyse durchgeführt (siehe Bild 6-2). Ausgehend von der initial betroffenen Anforderung *Achsenabstand* der Midroller (R4) ist die Teilfunktion *Kraft von Pendel an Band übertragen* (F4) betroffen, welche auf logischer Ebene durch den *Midroller links* (L1), *Midroller rechts* (L2) und *Pendel* (L3) abgebildet wird. Auf physischer Ebene wird festgestellt, dass die Baugruppe *Midroller RLG3* (P1, P2) und die dazugehörige Komponente *Stützradpendel 410 x 165 mm* (P3) betroffen ist, die von EB gefertigt wird. Das gusseiserne Stützradpendel verteilt die Aufstandskraft der Landmaschine auf bewegliche Rollen, die mit dem umlaufenden Band des Raupenlaufwerks verbunden sind. Das

Stützradpendel wird aktuell mit einem Achsabstand von 410 mm gefertigt, muss für die neue Landmaschine aber zukünftig auch mit einem Achsabstand von 380 mm gefertigt werden.

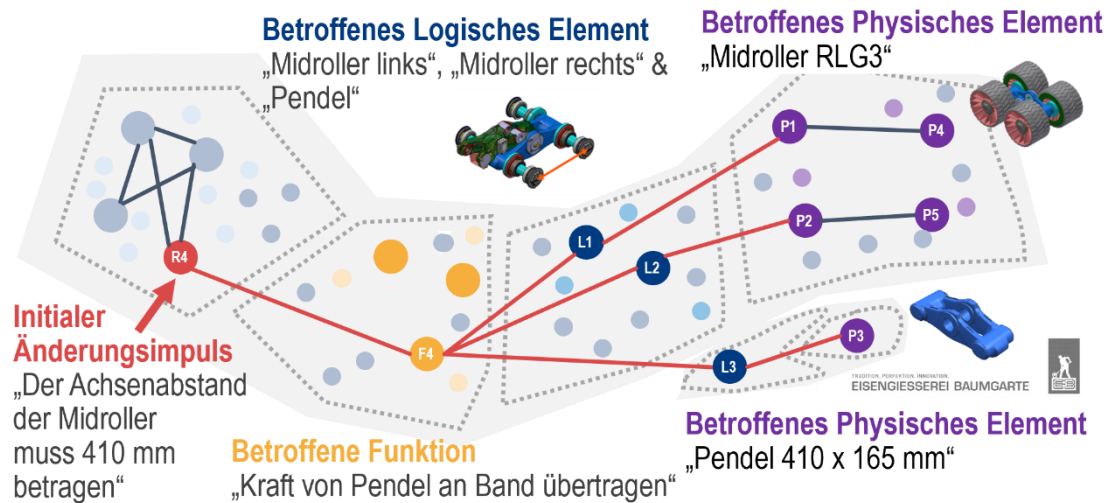


Bild 6-2: Fallbeispiel CLAAS Industrietechnik und Eisengiesserei Baumgarte GmbH

Bei dem Zulieferer Eisengiesserei Baumgarte (EB) wird das Pendel gegossen. Aufgrund des Änderungswunsches der CIT müssen die Bohrungspositionen konstruktiv angepasst werden, um den neuen Achsabstand durch eine neue Variante des Pendels umzusetzen. Da der Entwickler von CIT zusätzliche Varianz durch eine neue Variante des Pendels vermeiden will, wird in einer zweiten Wirkkettenanalyse geprüft, ob die variantenspezifische Nachbearbeitung eines Pendel-Rohlings kostengünstiger ist als die Fertigung von zwei Varianten des Pendels. Gemeinsam mit dem Kunden werden mithilfe von Wirkkettenanalyse die beiden Lösungsalternativen analysiert und ökonomisch verglichen:

Vergleich von zwei Lösungsalternativen: Die Fertigung eines Pendel-Rohlings (Alternative 1), welcher je Landmaschinenvariante nachbearbeitet werden kann. Oder die Fertigung zweier separater Pendel (Alternative 2) mit geringeren Stückzahlen.

Dabei werden Antworten auf folgende Ergebnisdimensionen automatisiert gefolgt:

- CIT Einkauf: Ist es kostengünstiger, eine weitere Variante des Stützradpendels zu gießen oder einen Pendel-Rohling nachzubearbeiten?
- EB Technische Betreuung: Was muss beim Gießprozess für die Lösungsalternativen geändert werden?
- EB Vertrieb: Für welche Kosten können beide Lösungsalternativen angeboten werden?

Ausgehend von den betroffenen Elementen aus der initialen Wirkkettenanalyse werden die beiden identifizierten Lösungsalternativen hinsichtlich der entstehenden Kosten verglichen (siehe Bild 6-3). Ergebnis der technischen Betreuung ist, dass für die Anpassung des Achsabstands bei Lösungsalternative 2 vier Kerne im Gusskasten verwendet werden, welche vom Vertrieb kalkulatorisch berücksichtigt werden müssen. Die daraus resultierenden Kosten für CIT liegen bei 104,55 €. Bei der Lösungsalternative 1 liegen die Kosten bei 92,00 € und jeweils 8,00 € für die variantenspezifische Nacharbeiten. Daher liegen die durchschnittlichen Kosten bei 100,00 € und unterschreiten die Kosten der zweiten Alternative. Die implementierten Dashboards sind im Anhang A7 (S. 220) aufgeführt.

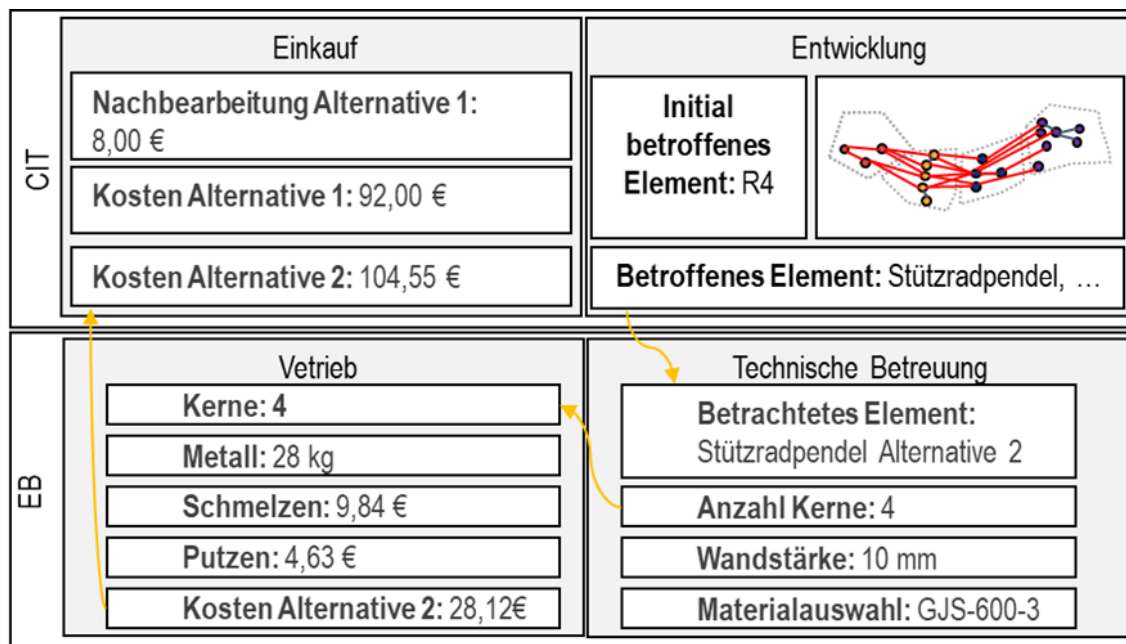


Bild 6-3: Dashboard zur ECR-Bewertung am Stützradpendel (mit fiktiven Zahlenwerten)

6.1.2 Fallstudie 2: Auswuchtmaschine

Rotierende Bauteile wie Kurbelwellen, Bohrer oder Turbinen werden auf die Auswuchtmaschinen des Unternehmens Hofmann Mess- und Auswuchttechnik GmbH (HFM) gelegt, um Unwuchten der Bauteile zu erkennen und zu beseitigen. Die Rotoren liegen auf zwei Lagerständern auf und werden durch einen Antrieb in Rotation versetzt (siehe Anhang A8, S. 226). Die Mess- und Bearbeitungsfunktionen werden durch mechanische, elektronische und softwaretechnische Komponenten realisiert. Die Anforderungen und Rahmenbedingungen an die Auswuchtmaschine basieren unter anderem auf dem auszuwuchtenden Rotor und dem Fabriklayout. Die Unwucht des Rotors wird durch die Messung der Schwingung durch Piezo-Messdosen innerhalb der Lagerständer berechnet. Die Konstruktion einer spezifischen Auswuchtmaschine ist stark abhängig von dem auszuwuchtenden Rotor. Relevante Parameter des Rotors sind der maximale Durchmesser, der Durchmesser des auf dem Lagerständer aufliegenden Rotorzapfens und das

Rotorgewicht. Aufgrund des Baukastensystems von Hofmann müssen bei technischen Änderungen an Standard-komponenten variantenübergreifende Auswirkungen berücksichtigt werden. Zusätzlich muss die Anforderungserfüllung stets gewährleistet sein. Im Folgenden wird die Bewertung der Auswirkungen eines realen ECRs mithilfe von Wirkkettenanalysen beschrieben:

Ein Endkunde fragt beim Einkaufsleiter von Hofmann an, ob es möglich ist, eine Motorwelle einer neuen Fahrzeugserie auf der existierenden Hofmann Auswuchtmaschine auszuwuchten. Zur Prüfung der Machbarkeit und Bewertung der Auswirkungen wird eine Wirkkettenanalyse durchgeführt. Im Vergleich zu der existierenden Motorwelle, für die die Auswuchtmaschine ausgelegt wurde, ist das Gewicht der neuen Kurbelwelle um 2 Kilogramm geringer und der Zapfendurchmesser um 5 mm geringer. Zur Bewertung der Änderung werden vier Ergebnisdimensionen der MBSE-Rollen beantwortet:

- Projektleiter: „Kann der Kunde den Rotor ohne Anpassungsbedarfe auswuchten?“
- Systemarchitekt: „Ist es technisch machbar, den Rotor auf der existierenden Maschine auszuwuchten?“
- Systemanalyst: „Wie hoch sind die Kosten einer Änderung?“
- Projektleiter: „In welcher Zeit sind zu beschaffende Bauteile lieferbar?“

Der Startpunkt der Wirkkettenanalyse ist die Anforderung Bauteilgröße (R8), in welcher der Wert des Parameters *Zapfendurchmesser* von 45 mm auf 40 mm angepasst werden soll. Ausgehend von dem initial betroffenen Element zeigen die Ergebnisse der ersten Wirkkettenanalyse, dass zehn weitere Anforderungen, eine Funktion, drei logische Elemente und acht physische Elemente von der Änderung betroffen sind (siehe Bild 6-4). Auf logischer Ebene ist der Lagereinsatz betroffen (L2), bei welchem der physikalische Effekt der Hertz'schen Pressung berücksichtigt werden muss. Auf physischer Ebene ist der Rollenlagereinsatz V2A 45 (P1) betroffen, auf dem der neue Rotor aufliegen würde.

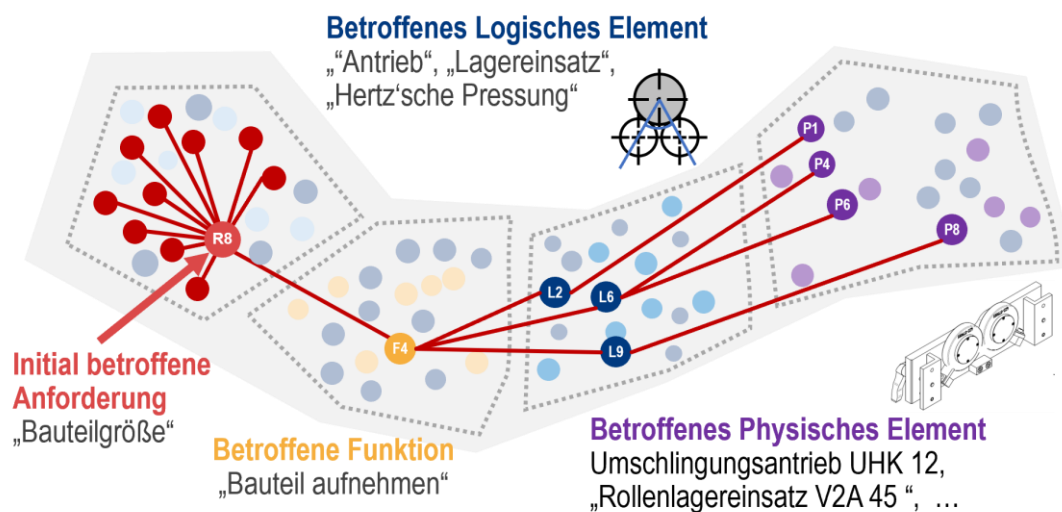


Bild 6-4: Wirkpfad entlang von RFLP im Fallbeispiel Hofmann

Bei einer Hertz'schen Pressung werden zwei Bauteile mit gekrümmten Oberflächen aufeinandergepresst, wodurch eine kreis- oder ellipsenförmige Berührungsfläche entsteht [Sau16, S. 90]. Bei einem Auswuchtprozess darf der entstehende Einlagerungswinkel (α) innerhalb des aufgespannten Dreiecks zwischen dem Zapfenmittelpunkt und den Mittelpunkten der Rollenlager einen Grenzwert von 120° nicht überschreiten, um die Belastung durch die Hertz'sche Pressung gering zu halten und plastische Verformungen am Rotor zu vermeiden. Der Einlagerungswinkel wird anhand des Abstands der Laufrollen (B), dem Radius des Rotors (R_R) und dem Radius der Laufrollen (R_S) berechnet.

Gleichung 1: Berechnung des Einlagerungswinkels beim Auswuchtprozess

$$\alpha = \text{asin}\left[\frac{B}{2 * (R_R + R_S)}\right]$$

Für die Konstrukteure wird ein maßgeschneidertes Dashboard bereitgestellt, in dem der entstehende Einlagerungswinkel durch die Änderung des Zapfendurchmessers automatisiert berechnet wird, sobald ein Parameter von einem ECR betroffen ist. Der Einlagerungswinkel, der sich zwischen dem Zapfen und dem aktuellen Rollenlagereinsatz V2A 45 (variantTag: M2) ergibt, liegt bei 134° . Da der Rotor mit dem aktuellen Rollenlagereinsatz nicht ausgewuchtet werden kann, werden weitere Varianten der Rollenlagereinsätze verglichen, um die technische Machbarkeit zu überprüfen. Dabei stellt sich heraus, dass das Rollenlager UHK 12.1 45 (variantTag: M3) zum Auswuchten des Rotors genutzt werden kann, da der Einlagerungswinkel bei 115° liegt. Schematisch sind die Dashboards in Bild 6-5 dargestellt; prototypisch implementiert im Anhang A8 (S. 226).

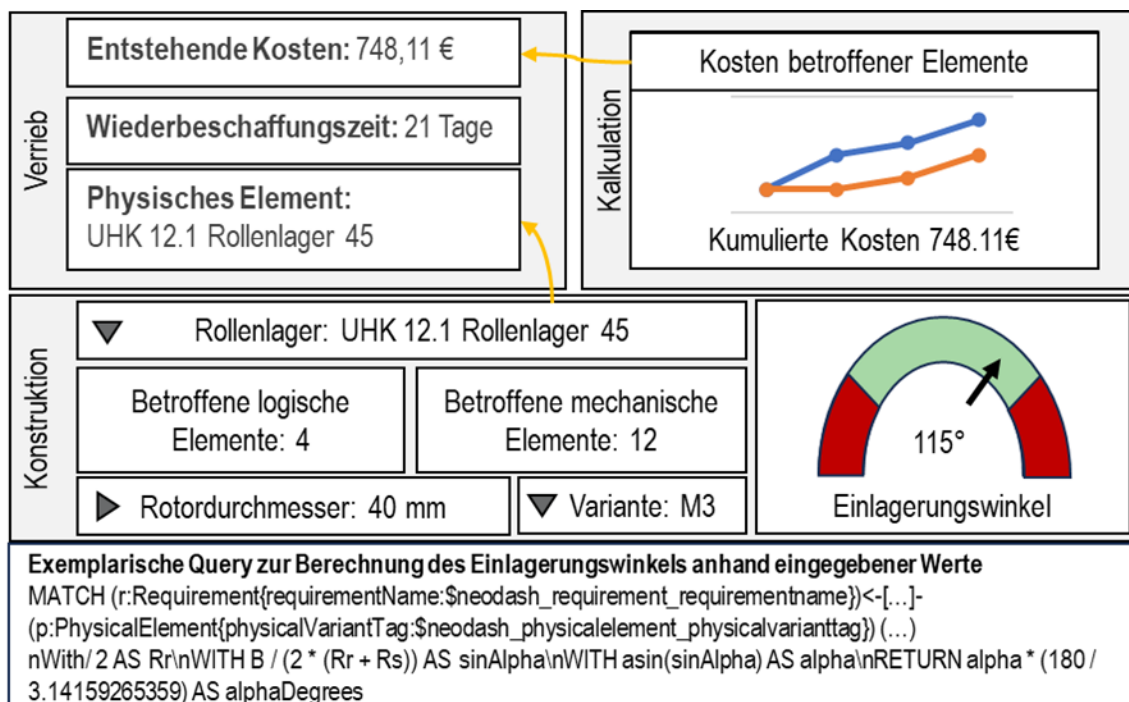


Bild 6-5: Vernetzte Dashboards im Fallbeispiel Auswuchtmaschine

Nachdem der Konstrukteur mithilfe der Wirkkettenanalyse eine technisch machbare Lösung identifiziert hat, informiert der Vertriebler den Kunden über die mögliche Lösung, für die ein alternatives Rollenlager beschafft werden muss. Der Kunde bestätigt die vorgeschlagene Lösung und fordert die Kosten und die Wiederbeschaffungszeit an. Im Dashboard der technischen Kalkulation werden die Kosten des neu ausgewählten Rollenlagers automatisiert kumuliert und angezeigt. Die technische Kalkulation prüft die Berechnung der Kosten und gibt diese für den Vertrieb frei. Die Beschaffung prüft die Wiederbeschaffungszeit des physischen Elements Rollenlagereinsatz UHK 12.1 45, welche 21 Tage beträgt. Die kumulierten Kosten und die Wiederbeschaffungszeit werden dem Vertrieb zur Kommunikation mit dem Kunden über die Dashboards bereitgestellt. Durch die Wirkkettenanalyse wurden betroffene Elemente identifiziert, die technische Machbarkeit überprüft und Lösungsalternativen zur Umsetzung der technischen Änderung geprüft und mit dem Kunden abgestimmt.

6.1.3 Fallstudie 3: Spritzgussartikel für E-Mobilität

Beim Spritzgussprozess wird Kunststoffgranulat einer Maschine zugeführt, verarbeitet und mit aufgebautem Druck in eine oder mehrere Werkzeugkavitäten gepresst. Die Kavität formt durch die geometrischen Elemente den gewünschten Spritzgussartikel. Das Unternehmen Hadi Plast GmbH & Co. KG Kunststoff-Verarbeitung (HP) entwickelt und produziert unterschiedliche Spritzgussartikel auf mehr als 50 Spritzgussmaschinen. Innerhalb des Fallbeispiels müssen daher **interdisziplinäre Zusammenhänge zwischen Maschine, Werkzeug und Artikeln** betrachtet werden (siehe Anhang A9, S. 238). Hierzu zählen auch mathematische Zusammenhänge zur Festlegung optimaler Produktionsparameter, beispielsweise des Einspritzdrucks, der Temperatur oder der aufzubringenden Zuhaltkraft des Werkzeugs. In der Fallstudie wurden Anforderungen an Spritzgussartikel, -maschine und -werkzeug abgebildet. Auf funktionaler Ebene wurden die Kernfunktionen des Spritzgusses abgebildet und horizontal und vertikal vernetzt. Auf logischer Ebene wurden die logischen Elemente eines Spritzgusswerkzeugs abgebildet, vorrangig formgebende Elemente wie Schnapphaken, Bohrungen und Fasen. Die physikalische Ebene wurde durch Stücklisten abgebildet, die Verifikation durch Prüfberichte.

Der Kunde möchte die Kosten eines bereits in Serie produzierten Ladesteckers senken und hierzu ein kostengünstigeres Kunststoffgranulat für den Produktionsprozess wählen. Daher wird eine **Materialänderung** von Ultraflow-K-FKGS6 zu Durethan BKVFN00 vorgeschlagen. Der Entwickler bei Hadi Plast führt eine Wirkkettenanalyse für die neue *Materialvorgabe* durch, um die Machbarkeit der technischen Änderung zu bewerten. Das initial betroffene Element ist die Anforderung „Materialvorgabe“ (1), welche auf Kundenwunsch geändert werden soll. Ausgehend von der initial betroffenen Anforderung werden mithilfe der Wirkkettenanalyse zwei weitere betroffene Anforderungen (A4, A6), drei betroffene Funktionen (F1-3), drei betroffene logische Elemente (L1-3) und zwei betroffene physische Elemente (P1, P2) identifiziert (siehe Bild 6-6). Auf logischer Ebene ist die mechanische Zuhaltkraft betroffen.

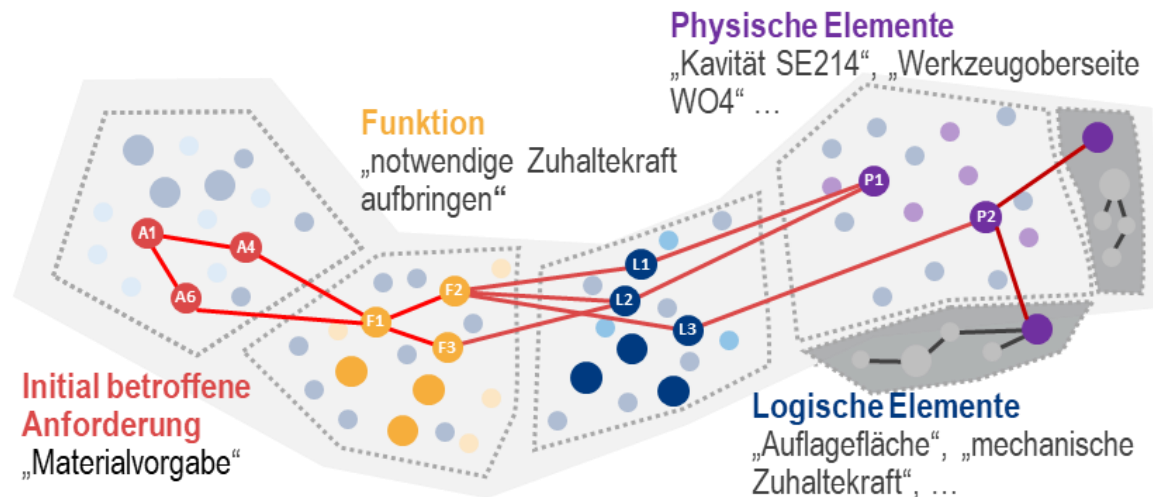


Bild 6-6: Wirkpfad des ECRs Änderung der Materialvorgabe

Die Materialänderung des Spritzgussartikels hat Einfluss auf den entstehenden Innendruck im Werkzeug, welcher wiederum die benötigte Zuhaltkraft beeinflusst, der durch die Maschine aufgebracht wird. Die Zuhaltkraft (Z) wird durch die Multiplikation des Werkzeuginnendrucks (P_W), der Anzahl der Spritzguss-Kavitäten (k_W) innerhalb des Spritzgusswerkzeugs und der projizierten Fläche (A_{proj}) berechnet (Gleichung 2).

$$Z = P_W \times k_W \times A_{proj}$$

Gleichung 2: Berechnung der erforderlichen Zuhaltkraft im Spritzgussprozess

Die Kavitäten sind ein oder mehrere formgebende Hohlräume innerhalb eines Spritzgusswerkzeugs [Jar19, S. 19]. Die projizierte Fläche ist die Fläche eines Artikels, die in die Formaufspannfläche eines Spritzgusswerkzeugs projiziert wird [Jar19, 44]. Zur Bewertung des ECRs der Materialänderung wird die Zuhaltkraft in einem maßgeschneiderten Dashboard des Entwicklers neu berechnet und mit der maximalen Zuhaltkraft der aktuell vorgesehenen Maschine verglichen. Neben den technischen Auswirkungen müssen ebenfalls die Auswirkungen auf die Qualität und die Kosten bewertet werden. Aus Sicht des Qualitätsmanagements (QM) muss sichergestellt werden, dass die Rahmenbedingungen weiterhin in die geforderte Toleranzgruppe TG 4 eingeordnet werden. Die Toleranzgruppe von Kunststoff-Formteilen wird nach ISO 20457 [DIN18] durch fünf Einzeleinflüsse (P) bestimmt: die Art des Verarbeitungsprozesses (P1), die Formsteifigkeit (P2), die Verarbeitungsschwindigkeit (P3), die Berücksichtigung der geometrie- und verfahrensbedingten Schwindungsunterschiede (P4) und die Bewertung des Fertigungsaufwands (P5). Durch den Materialwechsel ändern sich die Formsteifigkeit des Materials und die Verarbeitungsschwindigkeit, wodurch aus QM-Sicht eine erneute Berechnung der Toleranzgruppe notwendig wird. Aus Sicht der Kalkulation muss der Verkaufspreis neu berechnet werden, da das neue Material zu einem anderen Einkaufspreis erworben wird. Wie in Bild 6-7 dargestellt, nutzen die Rollen separate Dashboards, um die Änderungsauswirkungen schnell, objektiv und automatisiert zu bewerten.

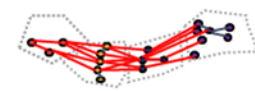
Einkauf/Kalkulation	Kunststoffgranulat: Durethan BKVFN00 ▼	Initial betroffenes Element: R3.1 
	Preis pro KG: 6,00 €	
	Artikelkosten: 0,17 € ✓	
QM	Geforderte Toleranzgruppe: 4	Kalkulation
	IST-Toleranzgruppe: 4 ✓	
	Maschine: Engel Victory ▼	
		Werkzeuginnendruck: 4,5 KN/cm ²
		Anzahl Kavitäten: 1 ▼
		Projizierte Fläche: 20 cm ² ▼
		IST-Zuhaltekraft: 90 KN ✓
		Zuhaltekraft Maschine: 100 KN
Exemplarische Query zur Berechnung der Zuhaltekraft anhand eingegebener Werte MATCH (cp:`machineparameters`{Kurzzeichen:\$neodash_machine_parameters_kurzzeichen}) WITH cp.Werkzeuginnendruck AS werkzeuginnendruck, toFloat(\$neodash_projizierte_flaeche) AS projizierte_flaeche, toFloat(\$neodash_anzahl_kavitaeten) AS anzahl_kavitaeten WITH werkzeuginnendruck, projizierte_flaeche, anzahl_kavitaeten, projizierte_flaeche * anzahl_kavitaeten * werkzeuginnendruck AS zuhaltekraft RETURN zuhaltekraft		

Bild 6-7: Vernetzte Dashboards im Fallbeispiel Spritzgussartikel

6.2 Fallstudie 4: Fensterheber aus der Automobilindustrie

Neben den ökonomischen Gründen Wirkketten zu modellieren und zu analysieren, steigt durch neue Regularien, Standards und Normen die Notwendigkeit, Wirkkettenmodelle explizit nachzuweisen, um Auswirkungen technischer Änderungen objektiv und frühzeitig zu bewerten. Ein praxisrelevantes Beispiel aus der Automobilindustrie ist die Einführung der UN ECE 156 Regularie [Uni21] der Europäischen Wirtschaftskommission. Die Regularie verpflichtet Automobilhersteller, ein Software Update Management System einzuführen, welches Software-Updates rückverfolgbar verwaltet und die Auswirkungen der Updates frühzeitig analysiert. Das Ziel der Wirkkettenanalyse im Beispiel der Automobilindustrie ist es daher, einerseits die Analyse von Auswirkungen von Software-Updates auf Funktionen aller Subsysteme und andererseits die Identifikation nicht-erfüllter Regularien nach einem Software-Update darzustellen. Falls die seit Januar 2021 geltende Regularie nicht nachweislich eingehalten wird, ist die Homologation einer Automobilsérie im europäischen Markt nicht möglich. [GWK23, S. 1]. In einem industrienahen Forschungsprojekt in der Automobilindustrie wurden über 100 Subsysteme einer Automobilsérie modellbasiert abgebildet und verknüpft. Zur Demonstration der Anwendung innerhalb einer SysML-basierten Implementierung der MECA-Methodik wird ein Aktuator aus dem Subsystem *Fensterheber* eines Kraftfahrzeugs betrachtet. Der Fensterheber muss zusätzlich zur Einbindung in das Software Update Management System (UN ECE 156) die Forderungen aus der UN ECE-Regularien R21 erfüllen. In der UN ECE Regularie 21 [Eur03] wird die Funktionalität „Einklemmschutz“ gefordert und als Zertifizierungsanforderung abgeleitet. Der Einklemmschutz muss dafür sorgen, dass der Aktuator des

Fensterhebers wieder öffnet, wenn beim Schließen des Fensters eine Gegenkraft von 100 N erkannt wird und den Schließvorgang blockiert. Zur Erreichung des Ziels wurden im **Metadatenmodell** sechs Artefakttypen und vier Relationstypen festgelegt. Die Artefakttypen sind Regularien, Zertifizierungsanforderungen, Kundenfunktionen, Systemfunktionen, Softwarekomponenten und Hardwarekomponente. Regularien enthalten einzelne Zertifizierungsanforderungen, die durch vorgesehene Funktionen erfüllt werden müssen. Eine für die Insassen erlebbare Kundenfunktion wird durch das Zusammenspiel von einer oder mehreren technischen Systemfunktionen ermöglicht. Systemfunktionen werden durch Software- und Hardwarekomponenten realisiert. Ausgangspunkt der in Bild 6-8 dargestellten **Wirkkettenanalyse** ist ein Software-Update auf dem *Steuergerät Tür*. Das Steuergerät realisiert die Kundenfunktion „Automatische Schließung des Seitenfensters“. Eingehende Signale kommen von den Systemfunktionen „Einklemmschutz sicherstellen“, „Zündstatus bereitstellen“ und „Fensterschließung fordern“. Die Systemfunktion „Einklemmschutz sicherstellen“ erfüllt eine Zertifizierungsanforderung, die aus der Regularie UN ECE R21 abgeleitet ist. Durch das manuelle Ablaufen der Ursache-Wirkungs-Beziehungen im SysML-Modell wird deutlich, dass bei dem Software-Update des Steuergeräts überprüft werden muss, ob der Einklemmschutz funktionsfähig bleibt.

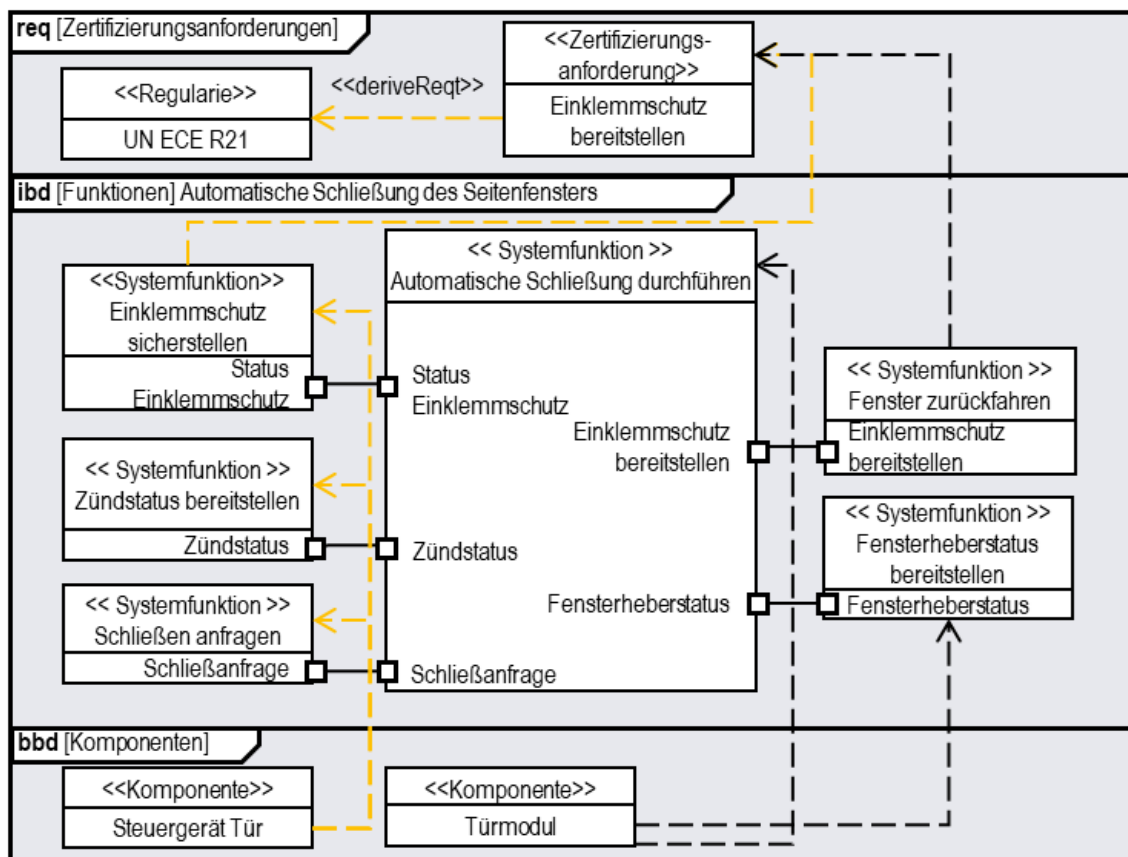


Bild 6-8: Wirkkettenanalyse für den Fensterheber (in Anlehnung an [GWK23, S. 20])

6.3 Validierung der Anforderungen

Die Validierung der Anforderungen erfolgte in drei zyklischen Validierungsworkshops während der iterativen Entwicklung der MECA-Methodik und vier abschließenden kontrollierten Experimenten:

- **Zyklische Validierungsworkshops** (WS 4, WS 8, WS 11): Digitale Validierungsworkshops zur Bewertung des aktuellen Stands der Methodik in halbjährigen Abständen. Bewertung durch vier IT-Partner, zwei akademischen Partner und vier industriellen Anwender (siehe Anhang A10.4, S. 250).
- **Kontrollierte Experimente:** Kontrollierte Experimente mit dem jeweiligen Hauptanwender der Anwendungsunternehmen CIT, EB, HFM und HP

Die kontrollierten Experimente basieren auf dem Ansatz des **Participative Designs** ([BC09, S. 301] aufbauend auf [Som07]): Die MECA-Methodik wird vom Forscher vorgestellt und anschließend durch die Probanden angewendet und bewertet. Die Rollen der separat teilnehmenden Probanden aus der Industrie sind: Entwicklungsingenieur (I1), Teamleiter IT (I2), Systemingenieur (I3), Mitarbeiter Technische Kalkulation (I4). In den Experimenten wurde die MECA-Methodik initial vom Versuchsleiter erklärt und anschließend vom Probanden eigenständig mithilfe der software-gestützten Implementierung angewendet. Zur Durchführung wurde im Heinz Nixdorf Institut ein Prüfstand in dem DFG Großgerät „Smart Innovation Lab“ eingerichtet (siehe Bild 6-9 und Anhang A10.2, S. 248) oder eine Durchführung bei den Industrieunternehmen organisiert, um die Geschäftsführung der Unternehmen und weitere potenzielle Anwender einzubeziehen. Zur Planung der empirischen Untersuchung wurden die Charakteristika empirischer Studien nach BLESSING UND CHAKRABARTI [BC09, S. 243] festgelegt (siehe Anhang A10.3, S. 249). Nach der Durchführung des Experiments wird ein **Validierungs-Fragebogen** von den Probanden ausgefüllt. Der Fragebogen umfasst sechs Abschnitte (siehe Anhang A10.5, S. 251): (1) Angaben zum betrachteten Fallbeispiel, (2) Unternehmenssituation, (3) Anwendungsaufwand der Methodik, (4) Anwendbarkeit der softwaregestützten Implementierung, (5) Genauigkeit der Ergebnisse, (6) Mehrwert der Methodik-Anwendung.

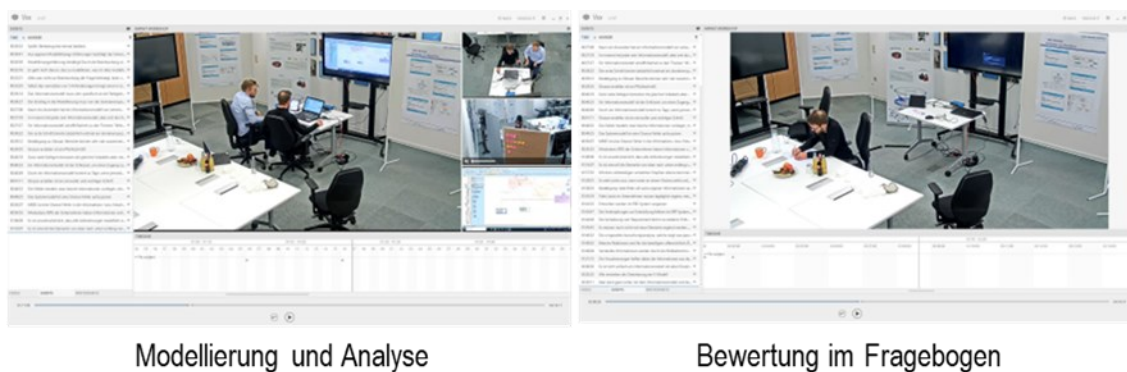


Bild 6-9: Durchführung des Validierungs-Workshops im Smart Innovation Lab

Da die Unternehmen der Fallstudien sich hinsichtlich der Unternehmensgröße, Branche, Rollen und Art der technischen Änderung unterscheiden, eignen die Probanden sich für die Bewertung der Anforderungen. Die Bewertung innerhalb des Validierungs-Fragebogens erfolgt anhand einer ungeraden Likert Skala mit 5 Merkmalsausprägungen. Die Skala der Merkmalsausprägungen umfasst [-2 stimme nicht zu], [-1 stimme eher nicht zu], [0 stimme teilweise zu], [1 stimme eher zu], [2 stimme zu] (siehe Bild 6-10). Die Validierung wird sukzessive je Kategorie der Anforderung durchgeführt.

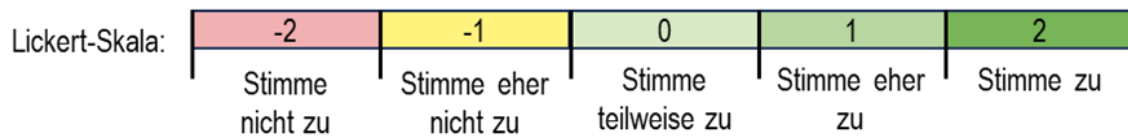


Bild 6-10: Schematische Darstellung der Likert-Skala

Zunächst werden die Anforderungen anhand der Eingangsdaten validiert (Kapitel 6.3.1). Anschließend werden die Anforderungen an die Methoden, Modelle und Hilfsmittel (Kapitel 6.3.2), die Anwendung (Kapitel 6.3.3) und die Ergebnisse (Kapitel 6.3.4) betrachtet. Die Ergebnisse aus den Fragebögen sind angehängt (siehe Anhang A10.6, S. 259).

6.3.1 Eingangsdaten

A1: Verfügbarkeit von Daten

Die Methodik muss so gestaltet sein, dass sie mit bestehenden Entwicklungsdaten durchgeführt werden kann.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn die Methodik mit allgemein relevanten Daten und Informationen aus der Entwicklung durchgeführt werden kann. Hierzu muss das Kriterium erfüllt sein, dass 75 % der für die Wirkkettenmodellierung benötigten Daten und Informationen verfügbar sind. Die Verfügbarkeit umfasst, dass die Daten und Informationen in einem angemessenen Umfang vorliegen und zugänglich sind. *„Informationen sind von angemessenem Umfang, wenn die Menge der verfügbaren Information den gestellten Anforderungen genügt.“* [HGH15, A33]. Übertragen auf die Methodik gilt, dass die Daten und Informationen in einem angemessenen Umfang vorliegen, wenn Anwender daraus Informationen der Artefakt- und Relationstypen ableiten können. *Informationen sind zugänglich, wenn sie anhand einfacher Verfahren und auf direktem Weg für den Anwender abrufbar sind.* [HGH15, S. 32]. Übertragen auf die Methodik gilt, dass die Daten und Informationen zugänglich sind, wenn Anwender direkten Zugriff auf die Daten ihres Unternehmens haben.

Bewertung: Die Verfügbarkeit der Daten wird anhand der für die Modellierungsworkshops der Fallstudien 1-3 zur Verfügung stehenden Daten bewertet. Zusätzlich wird die subjektive Verfügbarkeit typischer Partialmodelle innerhalb der Unternehmen analysiert, die in das Wirkkettenmodell eingebunden werden können.

Bewertung der Verfügbarkeit der Daten und Informationen in Modellierungsworkshops der Fallbeispiele 1-3

Zur objektiven Bewertung des Erfüllungskriteriums wird die Verfügbarkeit der Daten und Informationen anhand der Modellierungsworkshops der Fallstudien 1-3 betrachtet. Hierzu werden die verfügbaren Daten und Informationen den RFLPV²-Artefakten aus dem Metadatenmodell zugeordnet. Der Abgleich der verwendeten Modelle entlang des RFLPV²-Metadatenmodells ergibt, dass 88 % alle RFLPV²-Daten in den Fallbeispielen vertreten sind (siehe Bild 6-11).

Bewertung durch Industrieanwender im Validierungs-Fragebogen

Zur subjektiven Bewertung der Verfügbarkeit beantworten die Industrieanwender innerhalb des Validierungs-Fragebogens, welche Daten und Informationen in ihren Unternehmen grundsätzlich verfügbar sind. Die Daten und Informationen werden anhand von Kategorien von typischen Modellen aus der Entwicklung eingeordnet, die der Referenzliteratur [VWZ18, S. 15] entnommen werden und mit den Informationsbedarfen der Fallbeispiele abgeglichen wurden. Aus den Umfragen mit vier Probanden ergibt sich, dass aus subjektiver Perspektive 75 % der für die Wirkkettenanalyse nutzbaren Partialmodelle in den Beispielunternehmen vorhanden sind (siehe Anhang A6.3, S. 218). Die Differenz der objektiven und subjektiven Datenverfügbarkeit verdeutlicht, dass mehr Daten in einem Unternehmen vorliegen können, als einzelne Personen sie kennen.



Bild 6-11: Schematische Darstellung der objektiven und subjektiven Datenverfügbarkeit im Unternehmen

Validierungsergebnis: Anforderung 1 ist erfüllt.

A2: Unterschiedliche Arten der Änderung

Der Änderungsbedarf muss sich auf unterschiedliche Modellelemente beziehen können.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn alle Arten von technischen Änderungen (100 %) (aufbauend auf [BW23, S. 123]) mit der Wirkkettenanalyse adressiert werden, unabhängig von den betroffenen unterschiedlichen Modellelementen.

Bewertung: Die Bewertung wird anhand der in den Fallstudien und in den Basisalgorithmen umgesetzten Arten der Änderungspropagation durchgeführt (siehe Kapitel 6.1 und Anhang A4.9, S. 191). In der Entwicklungsphase des Systems sind folgende Bereiche der

Änderungspropagation zu unterscheiden: Auswirkungen auf Anforderungen (1), Funktionen (2), Subsysteme und Komponenten (3), Gestaltungsparameter (4), Geometrien (5), Entwicklungsrepräsentationen (Prototypen) (6) Systemverhalten und Performanz (7), Entwicklungsaufgaben (8), Verantwortliche (9) und Produktion und spätere Lebensphasen (10) (aufbauend auf [BW23, S. 123]).

Analyse der Abfragen und Basisalgorithmen von Fallstudien 1-3:

In der Fallstudie bei HFM wird die Änderung des *Zapfendurchmessers* betrachtet. Die Änderung propagiert ausgehend von einer Anforderung (1), über Funktionen (2) hin zum Subsystem Rollenlagerträger und einzelnen Systemelementen wie den Laufrollen. Das Systemverhalten wird durch die Neuberechnung des Einlagerungswinkels geprüft (7). Die Überprüfung des Einlagerungswinkels einer Entwicklungsaufgabe (8) wird dem mechanischen Konstrukteur (9) zugewiesen. Durch Auswahl eines neuen Rollenlagers ändern sich die Gestaltungsparameter Laufrollendurchmesser (4) der Komponente Laufrolle, wodurch der Einlagerungswinkel eingehalten werden kann. In der Fallstudie bei CIT und EB wurden die Auswirkungen einer geometrischen Änderung des *Achsabstandes* der physischen Komponente Pendel betrachtet (3). Die technische Änderung betrifft sowohl die Verantwortlichen aus der Entwicklung und Kalkulation von CIT als auch die technische Betreuung der EB. Durch die Anpassung von Geometriemodellen (5) werden zwei prototypische Lösungsalternativen (6) des Pendels betrachtet, die im Subsystem Midroller implementiert werden können. In der Fallstudie bei HP wurden die Auswirkungen einer *Materialänderung* bewertet. Hierbei wurden Auswirkungen auf die Toleranzeinhaltung des Artikels betrachtet, ebenso wie Auswirkungen auf die aufzubringende Zuhaltkraft der Spritzgussmaschine während des Produktionsprozesses (10). Die Basisalgorithmen verdeutlichen, dass maßgeschneiderte Wirkkettenanalysen von unterschiedlichen initial betroffenen Elementen möglich sind (siehe Anhang A4.9, S. 191).

Der Funktionsnachweis anhand der betrachteten ECRs aus den Fallbeispielen zeigt, dass sich initiale Änderungsbedarfe auf unterschiedliche Elemente beziehen können und mithilfe der MECA-Methodik alle Arten von technischen Änderungen bewertet werden können. Daher gilt: Wenn das Element Teil des Wirkkettenmodells ist, kann sich eine technische Änderung direkt auf das Element beziehen oder es indirekt beeinflussen.

Validierungsergebnis: Anforderung 2 ist erfüllt.

6.3.2 Methoden, Modelle und Hilfsmittel

A3: Tiefe, Breite und Weite des Metadatenmodells

Mit der Methodik müssen Daten auf verschiedenen Detaillierungsebenen (Tiefe), entlang von relevanten Entwicklungsartefakten (Weite) und aus verschiedenen Disziplinen (Breite) abgebildet werden.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn das Wirkkettenmodell alle (100 %) für die Wirkkettenanalyse notwendigen Informationen auf verschiedenen Detaillierungsebenen (Tiefe), entlang von relevanten Entwicklungsartefakten (Weite) und aus verschiedenen Disziplinen (Breite) abbildet (siehe Kapitel 4.2 und angelehnt an [HBF19, S. 7]). Die Detaillierungsebene (*Tiefe*) wird durch die Anzahl der Systemhierarchie bewertet. Grundlage ist die in der Arbeit verwendete Definition nach Haberfellner [HWF19, S. 8], in welcher zwischen Suprasystem (System of Systems), System, Subsystem(en) und Systemelementen unterschieden wird. Die *Breite* wird bewertet anhand der Grundstruktur eines mechatronischen Systems nach VDI/VDE 2206:2021 [VDI21, S. 13]. Mechatronische Systeme umfassen Elemente der Mechanik, Elektrik/Elektronik und Software und können durch weitere Disziplinen ergänzt werden [VDI21, S. 6]. Die *Weite* wird anhand der modellierten Knoten und Kanten der Wirkkettenmodelle nachgewiesen, die entlang des RFLPV²-Metadatenmodells abgebildet wurden.

Bewertung: Zur Bewertung des Erfüllungskriteriums werden die Wirkkettenmodelle der Fallbeispiele 1, 2, 3 und 5 betrachtet und analysiert.

Walkthrough durch Wirkkettenmodelle: Je Fallbeispiel wird die Tiefe, Breite und Weite tabellarisch dokumentiert (siehe Tabelle 6-2).

Tabelle 6-2: *Tiefe, Breite und Weite der Wirkkettenmodelle*

Fall-studie	Tiefe	Breite	Weite
FS 1	Raupenlaufwerk mit 9 Subsystemen und 44 Systemelementen	Mechanik, E/E, Hydraulik	197 Knoten 400 Kanten
FS 2	Drei Varianten einer Auswuchtmaschine mit 15 Subsystemen und bis zu 486 Systemelementen	Mechanik, E/E, Software	1218 Knoten 964 Kanten
FS 3	Spritzgusswerkzeug mit 15 Subsystemen und 76 Systemelementen	Mechanik, E/E, Hydraulik	432 Knoten, 431 Kanten
FS 5	Roboterarm innerhalb Smart Automation Lab (<i>Supra-System</i>) mit 13 Subsystemen und 134 Systemelementen	Mechanik, E/E, Software, Robotik	510 Knoten 944 Kanten

Bewertung der Industrieanwender: Zusätzlich zu dem Walkthrough werden die Anwender befragt, ob benötigte Informationen in der notwendigen Tiefe, Weite und Breite abgebildet werden können. Hierzu wird eine These beantwortet (siehe Bild 6-12):

Notwendige Informationen auf verschiedenen Granularitätslevels (Tiefe) und aus verschiedenen Disziplinen (Breite) können in dem RFLPV²-Modell abgebildet werden.



Bild 6-12: *Beantwortung der These zur Tiefe, Breite und Weite des Metadatenmodells*

Validierungsergebnis: Anforderung 3 ist erfüllt

A4: Änderungsausbreitung

Die Wirkkettenanalyse muss spezifische Propagationspfade analysieren.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn alle unterschiedlichen benötigten Propagationspfade abgebildet werden können (100% aller Propagationspfade).

Bewertung: Für die Bewertung wird eine Funktionsprüfung der praktisch auftretenden Propagationspfade der Fallstudien betrachtet.

Funktionsprüfung zur Abbildung unterschiedlicher Propagationspfade: Innerhalb eines Expertenworkshops mit vier Industrieexperten der Fallstudien 2-3 wurden 144 Ergebnisdimensionen (siehe WS-6 in Anhang A4.3, S. 181) definiert. Für die Ergebnisdimensionen wurden 26 wiederkehrende Propagationspfade innerhalb des RFLPV²-Modells abgeleitet, Queries in Cypher formuliert und als Basisalgorithmen bereitgestellt (siehe Anhang A4.9, S. 191). In den Fallstudien wurden mithilfe der MECA-Methodik 125 Queries, 11 rollenspezifische Dashboards und 28 Basisalgorithmen entwickelt. 100 % der praxisrelevanten 24 Propagationspfade (Wirkpfade) können somit in dem RFLPV²-Metadatenmodell durchlaufen und analysiert werden. Zusätzlich wird das Propagationsverhalten einzelner Elemente durch die Unterteilung in Konstanten, Absorber, Träger und Multiplikatoren [ECZ04, S. 13] qualitativ bewertet (umgesetzt in Basisalgorithmus 26).

Validierungsergebnisse: Anforderung 4 ist erfüllt.

A5: Objektive Bewertung

Die Auswirkungen der technischen Änderung müssen objektiv, qualitativ und quantitativ bewertet werden können.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn je Fallstudie objektive Bewertungskriterien definiert und angewendet wurden, um die Auswirkungen technischer Änderungen zu bewerten.

Bewertung: Die Bewertung umfasst die Prüfung der verwendeten objektiven Bewertungskriterien innerhalb der Fallstudien 1,2, 3 und 5.

Prüfung der verwendeten objektiven Bewertungskriterien

In den Fallbeispielen wurden insgesamt 57 Bewertungskriterien hergeleitet. Die qualitativen und quantitativen Bewertungskriterien adressieren unterschiedliche Ergebnisdimensionen und können in Kategorien eingeteilt werden (siehe Tabelle 6-3). Die Kategorien umfassen betroffene Artefakte, betroffene Attribute, kumulierte Werte, alleinstehende Werte, mathematische Zusammenhänge zwischen verschiedenen Parametern. Die mathematischen Zusammenhänge stammen entweder aus Fachliteratur oder aus Normen.

Tabelle 6-3: Darstellung exemplarischer verwendeter Bewertungskriterien

Kategorie	Beispiele	Fallstudie
Betroffene Artefakte	Anzahl betroffener Anforderungen, Anzahl sicherheitsrelevanter Funktionen, ...	1,2,3,5
Betroffene Attribute	Anzahl betroffener Elemente mit Bezug zu Disziplinen, Prioritäten, Personen, ...	1,2,3,5
Kumulierte Werte	Kumulierte Kosten einer betroffenen Baugruppe, variantenspezifische Kosten, ...	1,2
Mathematische Zusammenhänge aus Fachliteratur	Berechnung der Hertz'schen Pressung, Berechnung der Zuhaltkraft, ...	2,3
Normen für Qualität	Einordnung in Toleranzgruppe nach DIN ISO, ...	3

100 % aller geforderten Bewertungskriterien konnten in den Wirkkettenanalysen umgesetzt werden.

Validierungsergebnis: Anforderung 5 ist erfüllt.

A6: Prozessuale Verankerung

Die Wirkkettenanalyse muss in die unternehmensspezifischen Entwicklungsprozesse integriert werden können.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn alle Anwender bestätigen, dass die Wirkkettenanalyse in die existierenden Entwicklungsprozesse eines Unternehmens integriert werden kann (Likert-Skala ≥ 1).

Bewertung: Zur Bewertung wird eine Prüfung der Integrierbarkeit in den Unternehmen der Probanden vorgenommen und die Bewertung der Probanden im Validierungs-Fragebogen analysiert.

Prüfung der Integrierbarkeit: Die Integrierbarkeit der Wirkkettenanalysen wird in den Modellierungswshops (WS 5-7) anhand des Referenzprozesses zur Einbindung in die Produktentwicklung (siehe Kapitel 5.6) mit den Probanden diskutiert und validiert. Aktuelle Unternehmensprozesse wurden mit den Probanden aufgenommen, in BPMN modelliert und für die Integration der Wirkkettenanalysen basierend auf der MECA-Methodik angepasst. Anhand der Ergebnisse aus den Modellierungswshops wird nachvollzogen, welche Prozessschritte durch die modellbasierten Wirkkettenanalyse unterstützt werden können. Folgende Prozessschritte wurden bei den Anwendungsunternehmen identifiziert, um die Wirkkettenanalyse zu integrieren: Machbarkeitsanalyse und Mehrkostenkalkulation (FS 1), Meeting mit Fachexperten (FS 2), experten-basierte Problemanalyse (FS 3). Je Fallstudie wurden die Rahmenbedingungen zu den prototypischen Implementierungen in den Unternehmen bestimmt. Erkenntnisse wurden in den Referenzprozess zur Einbindung der Methodik in Unternehmensprozesse iterativ eingebracht (siehe Kapitel 5.6).

Bewertung der Industrieanwender: Die These zur Integration der Wirkkettenanalysen in den technischen Änderungsprozess des Unternehmens wird von den Industrieanwendern (n=4) wie folgt bewertet (siehe Bild 6-13):

Die Wirkkettenanalysen können in den Änderungsprozess des Unternehmens integriert werden.



Bild 6-13: Beantwortung der These zur objektiven Bewertung

Validierungsergebnis: Anforderung 6 ist erfüllt.

6.3.3 Anwendung

A7: Akzeptabler zeitlicher Anwendungsaufwand

Der zeitliche Modellierungs- und Anwendungsaufwand der Methodik muss in der industriellen Praxis akzeptabel sein.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn der zeitliche Anwendungsaufwand durchschnittlich als „akzeptabel“ eingestuft wird (Likert-Skala $\geq \emptyset 1$).

Bewertung: Zur Bewertung werden der zeitliche Anwendungsaufwand der MECA-Methodik quantifiziert und der Automatisierungsgrad diskutiert. Die Quantifizierung unterteilt sich in die objektive zeitliche Messung in den kontrollierten Experimenten und das subjektive zeitliche Empfinden der Probanden (Likert-Skala $\geq \emptyset 1$).

Messung des zeitlichen Aufwands: Basierend auf den Aufnahmen der Validierungs-Workshops werden die Zeiten zur Modellierung eines initialen Wirkpfads zum Export der Daten aus SysML in eine Graphdatenbank und die durchschnittlichen Laufzeiten der Queries analysiert. Hierzu wird die Software *Noldus Observer* verwendet. Die Ergebnisse der Messungen sind in Tabelle 6-4 dargestellt.

Tabelle 6-4: Messung des Anwendungsaufwands je Fallstudie

Gemessene Aktivität	Ergebnisse aus Fallstudien
Modellierung eines separaten Wirkpfads in SysML	< 300 Sekunden
Export SysML	< 3 Sekunden
Import Graphdatenbank	< 60 Sekunden
\emptyset Laufzeiten Queries	< 0.2 Sekunden

Bewertung des Automatisierungsgrads der software-gestützten Implementierung: Zur Identifikation betroffener Elemente werden Abfragen mithilfe der Abfragesprache Cypher formuliert. Die Nutzung der Abfragesprache automatisiert die Identifikation

betroffener Elemente entlang eines vordefinierten Propagationspfads. Die qualitative Bewertung der Auswirkungen erfolgt anhand von Visualisierungen der betroffenen Elemente. Hierzu werden Visualisierungsarten in den Dashboards festgelegt und bei der Durchführung der Wirkkettenanalyse automatisiert erzeugt. Die quantitative Bewertung der Auswirkungen erfolgt durch die Integration von Metriken in die Abfragen, beispielsweise durch die automatisierte Berechnung von Schwellwerten und mathematischen Gleichungen. Dadurch ist die quantitative Bewertung ebenfalls automatisiert umgesetzt.

Bewertung der Industrieanwender: Neben dem quantifizierten Zeitaufwand ist ebenfalls das subjektive zeitliche Empfinden der Anwender Teil der Bewertung. Nach der Anwendung der MECA-Methodik im kontrollierten Experiment bewerten die Anwender folgende Thesen (siehe Bild 6-14):

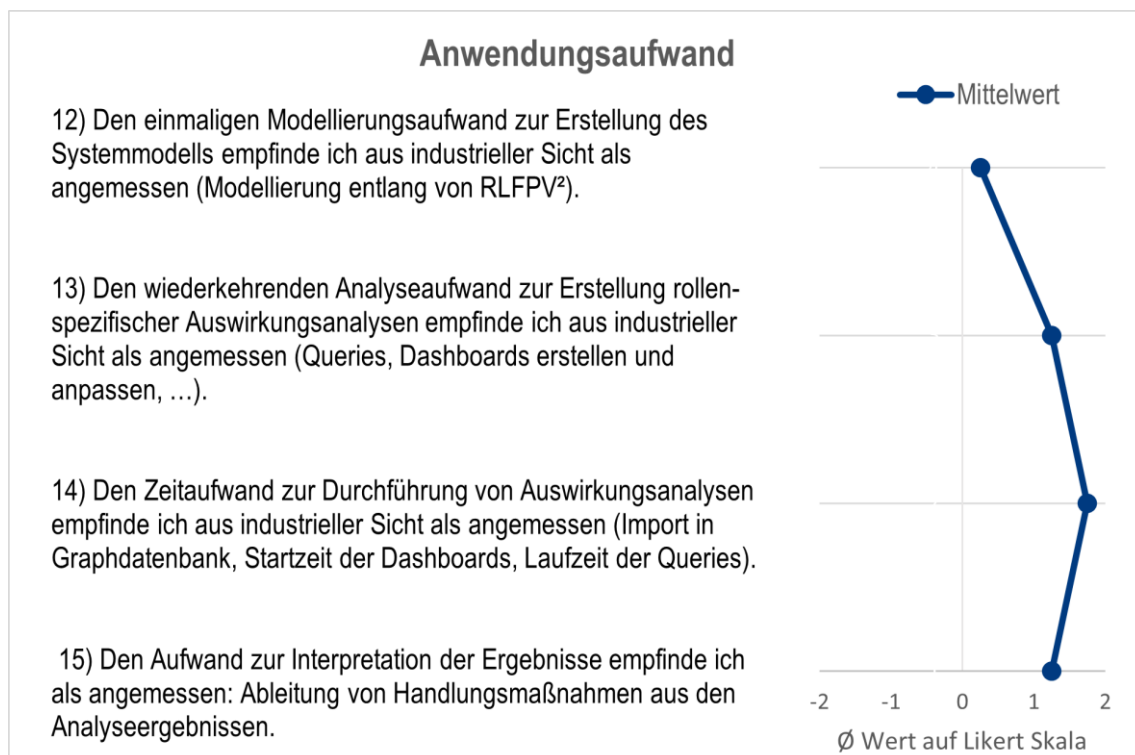


Bild 6-14: Bewertung Anwendungsaufwand

Die Thesen werden durchschnittlich als „akzeptabel“ (\emptyset Likert Skala >0) eingestuft.

Validierungsergebnis: Die Anforderung 7 ist teilweise erfüllt.

A8: Einfache Anwendbarkeit

Die Methodik muss so gestaltet sein, dass industrielle Anwender die Anwendbarkeit als einfach bewerten.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn Entwickler die Anwendung der Methodik als einfach wahrnehmen. Hierzu muss der System Usability Scale (SUS) Fragebogen einen Schwellwert von 70 von 100 Punkten überschreiten, um aus

Nutzersicht im akzeptablen Bereich eingestuft zu werden [BKM09, S. 121]. Zusätzlich müssen alle Rückfragen während des kontrollierten Experiments beantwortet werden.

Bewertung: Die Bewertung der Einfachheit umfasst die methodische Einfachheit und die Einfachheit der entwickelten IT-Lösungen.

Bewertung der software-gestützten Implementierung durch SUS-Fragebogen

Nach der anschließenden eigenständigen Anwendung der Modellierung sowie der Erstellung und Anpassung der Wirkkettenanalyse wurde der SUS-Fragebogen ausgefüllt [Bro, S. 5]. Die Ergebnisse werden in Bild 6-15 dargestellt.

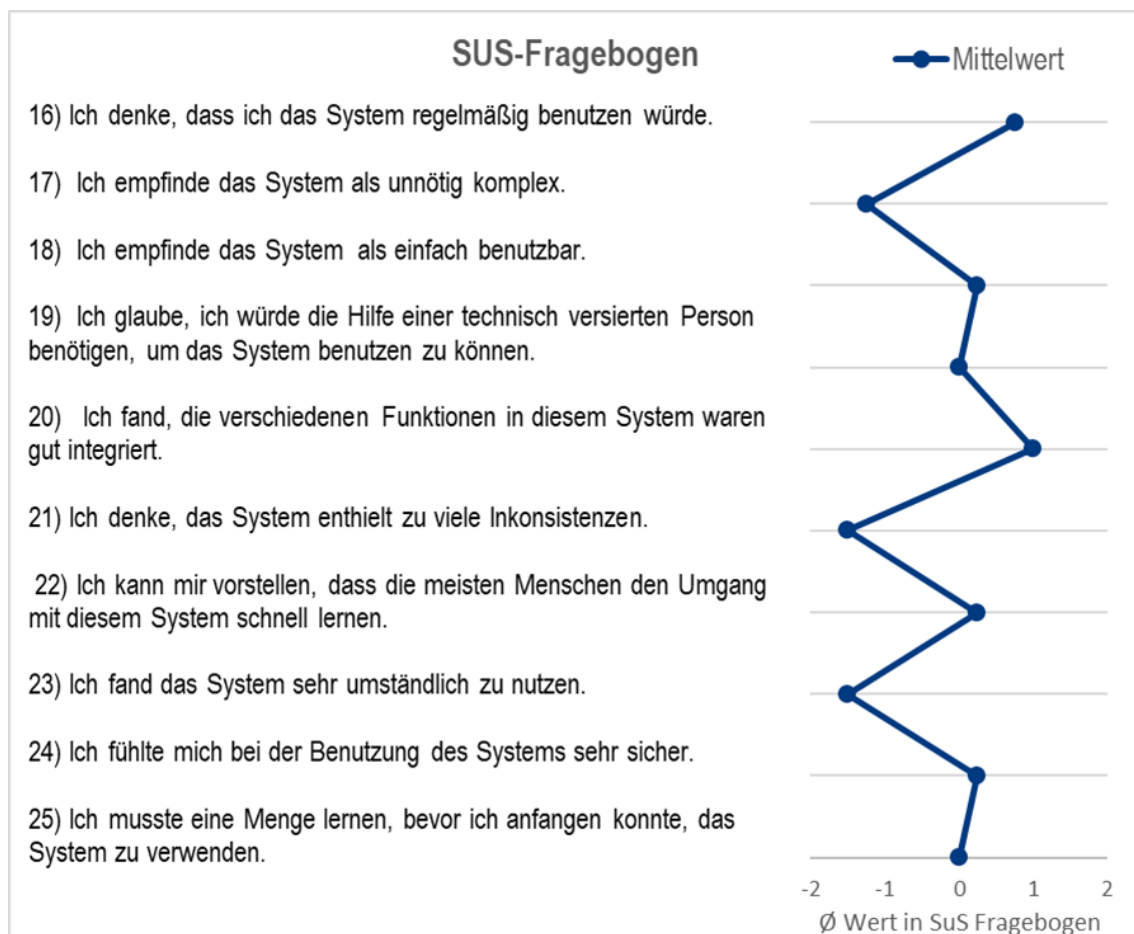


Bild 6-15: SUS-Fragebogen Ergebnis

Die ausgewerteten SUS-Scores der Fragebögen variieren zwischen 66 und 77 Punkten. Der durchschnittliche SUS-Score befindet sich bei 72 Punkten und liegt damit im akzeptablen Bereich. In den Modellierungswerkshops in vier Fallstudien wurde gezeigt, dass Anwender mit Hilfe der Methodik schnell und selbstständig eigene Wirkkettenmodelle und -analysen entwickeln können.

Analyse der Ergebnisse der kontrollierten Experimente: Die Analyse der kontrollierten Experimente ergibt, dass nur 11 Rückfragen gestellt wurden, welche zur Zufriedenheit

der Anwender beantwortet wurden. Zusätzlich wurden Aussagen identifiziert, die die einfache Anwendbarkeit der Methodik bekräftigen.

- Proband zur Anpassung des Metadatenmodells: Der Einstieg in die Modellierung muss vom Verständnis der Anwender kommen.
- Proband zur Nutzung der Ergebnisdimensionen: Es ist zwingend notwendig, den Modellierungsaufwand auf das Notwendige zu begrenzen. Dies gelingt durch die Herleitung der Frage „Welche Frage(n) soll mein Modell beantworten?“.

Die These zur einfachen Anwendbarkeit durch die frühzeitige Festlegung des Ziels wird von den Probanden wie folgt bewertet (siehe Bild 6-16):

Die frühzeitige Festlegung des Ziels der Wirkkettenanalyse hilft bei der effizienten Entwicklung der Abfragen und Metriken.



Bild 6-16: Beantwortung der These zur einfachen Anwendbarkeit

Validierungsergebnis: Anforderung 8 ist erfüllt.

A9: Kollaborative Anwendbarkeit

Die Datenbereitstellung und -auswertung muss kollaborativ möglich sein.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn die bereichs- und aktueurübergreifende Kollaboration im Anwendungszusammenhang nachgewiesen wird. Zum Nachweis müssen je Fallbeispiel mehrere Stakeholder (>2) kollaborativ Daten bereitstellen und die Änderungsauswirkungen aus ihrer Sicht bewerten können.

Bewertung: Die Bewertung der kollaborativen Datenbereitstellung und -auswertung wird anhand der Dashboards der Fallstudien 1-3 durchgeführt. Bewertungsgrundlage sind die Anzahl der durch die Dashboards verbundenen Rollen.

Analyse der rollenspezifischen Dashboards in den Fallstudien

Für die Fallstudien Raupenlaufwerk (FS 1), Auswuchtmaschine (FS 2), Spritzgussartikel (FS 3) und Roboterarm (FS 5) wurden 18 rollenspezifische Dashboards erstellt. Exemplarische unterschiedene Rollen waren Einkauf, Vertrieb, technische Kalkulation, mechanische Entwicklung sowie Messtechnik. Die Datenbereitstellung kann durch die Festlegung von Rollen und Rechten konfiguriert werden. Ein Beispiel für die kollaborative Datenbereitstellung und -auswertung wird in FS 1 durch die Zusammenarbeit des Landmaschinenherstellers CIT und EB verdeutlicht. In dem deskriptiven Szenario gibt der CIT-Entwickler Werte wie den Jahresabsatz des Raupenlaufwerks über das Dashboard vor. Die vorgegebenen Werte werden in den Dashboards des Zulieferers referenziert. Die Technischen Kalkulation von EB kann anschließend eigene Werte hinsichtlich des zu

ändernden Pendels vervollständigen. In den anderen Fallstudien werden ebenfalls kollaborative Szenarien beschrieben, indem unterschiedliche Rollen innerhalb eines Unternehmens die Änderungsauswirkungen in den rollen-spezifischen Dashboards bewerten. Die Erweiterbarkeit ist nicht limitiert. Prämisse zur Partizipation ist die Installation der Graph-Applikation auf einem Endgerät.

Validierungsergebnis: Anforderung 9 ist erfüllt.

6.3.4 Ergebnisse

A10: Maßgeschneiderte Ergebnisse

Die Ergebnisse der Methodik müssen an die rollenspezifischen Bedarfe angepasst werden können.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn die Industrieanwender bestätigen, dass alle Ergebnisse (100%) der Wirkkettenanalyse auf die unternehmensspezifischen Rollen angepasst wurden und diese hinreichend genau sind (\emptyset Likert Skala >1).

Bewertung: Zur Bewertung werden die rollenspezifischen Dashboards der Fallstudien 1-3 analysiert und die subjektiven Wahrnehmungen der Probanden ausgewertet.

Analyse der rollenspezifischen Dashboards: Die Methode ist auf die unternehmensspezifischen Anforderungen zugeschnitten, indem bereits zu Beginn die benötigten Ergebnisdimensionen der Stakeholder des Unternehmens identifiziert werden. Die Ergebnisse bestimmen die weitere Ausgestaltung des Wirkkettenmodells und der Wirkkettenanalysen. Zur Bewertung der Anpassung an die rollenspezifischen Ergebnisse wird das Participative Design ([BC09, S. 301] aufbauend auf [Som07]) in den kontrollierten Experimenten angewendet. In dem Validierungs-Workshop leiten Industrieanwender die Fragestellung, die Metrik und die Visualisierungsart von rollenspezifischen Ergebnisdimensionen eigenständig her. Anschließend werden die Ergebnisdimensionen mit Abfragen in die Dashboards (NeoDash) integriert. 100 % der Ergebnisdimensionen konnten durch die Abfragen umgesetzt werden.

Bewertung durch Industrieanwender: Nach der Durchführung des Participative Designs bewerten die Industrieanwender die Thesen (siehe Bild 6-17 und Bild 6-18):

Durch die rollenspezifischen Ergebnissichten werden die individuellen Informationsbedarfe zur Bewertung von technischen Änderungen bereitgestellt.

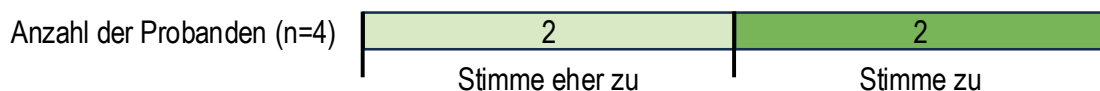


Bild 6-17: Beantwortung der These zu individuellen Informationsbedarfen

Die Ergebnisse sind hinreichend genau, um die rollenspezifischen Informationsbedarfe zu decken.

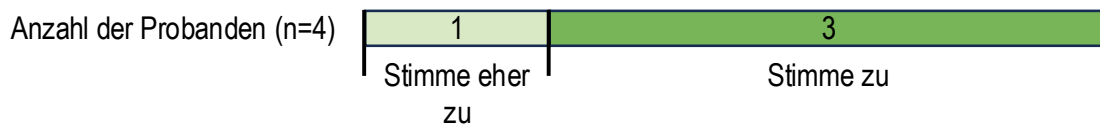


Bild 6-18: Beantwortung der These zu rollenspezifischen Informationsbedarfen

Validierungsergebnis: Die Anforderung 10 ist erfüllt.

A11: Ergebnisqualität

Die Ergebnisse der Methodik müssen die qualitativen Ansprüche der industriellen Praxis erfüllen.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn Industrieanwender die Ergebnisqualität als hinreichend genau bewerten. Hierzu müssen die Ergebnisse hinreichend genau sein, um betroffene Elemente zu identifizieren, Auswirkungen qualitativ und quantitativ zu bewerten und Handlungsmaßnahmen für den Umgang mit technischen Änderungen abzuleiten (\emptyset Likert Skala >1).

Bewertung: Zur Bewertung wird der objektive Vergleich mit den theoretisch erwarteten Ergebnissen anhand der 144 hergeleiteten Ergebnisdimensionen (siehe Anhang A4.3, S. 181) durchgeführt und durch die subjektive Bewertung der Ergebnisqualität der Probanden ergänzt.

Vergleich mit theoretischen Erwartungen: Innerhalb der Modellierungsworkshops (WS-1, WS-2, WS-3) wurden sowohl theoretisch erwartete Auswirkungen der technischen Änderungen der deskriptiven Szenarien beschrieben als auch erwartete Ergebnisdimensionen formuliert. Der Propagationspfad wurde entlang der RFLPV²-Artefakte abgebildet. Zur Bewertung der Ergebnisqualität wird der schematisch dokumentierte Propagationspfad mit dem implementierten rechner-interpretierten Propagationspfad der Wirkkettenanalyse als Cypher-Query verglichen und als Basisalgorithmus festgehalten (siehe Anhang A4.9, S. 191).

100 % der Ergebnisdimensionen konnten durch die Abfragen umgesetzt werden.

Bewertung der Industrieanwender: Nach der Entwicklung eigener Wirkkettenanalysen in dem kontrollierten Experiment bewerten die Industrieanwender die Ergebnisqualität im Validierungs-Fragebogen (siehe Bild 6-19).

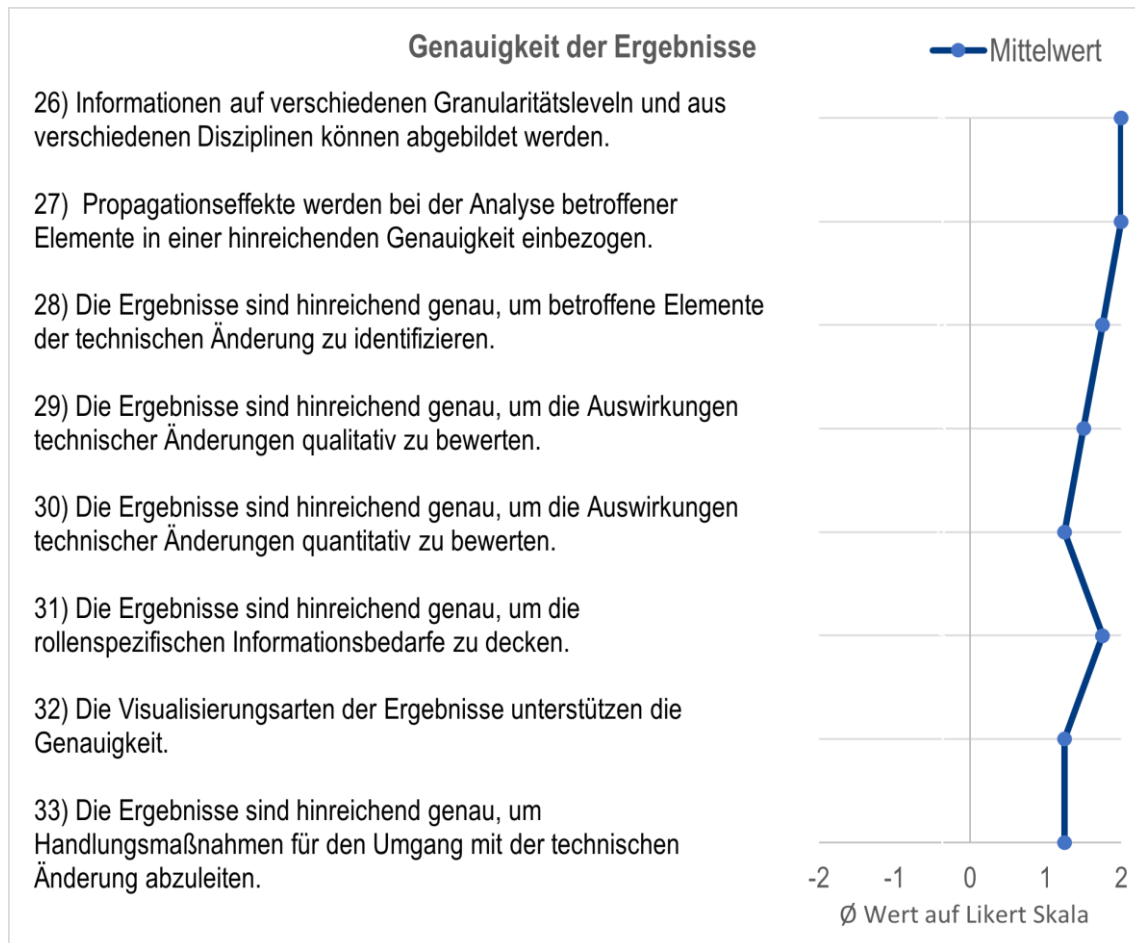


Bild 6-19: Genauigkeit der Ergebnisse

Insgesamt werden die Thesen mit einem Wert von $> \bar{x} 1$ auf der Likert Skala [Lic32, 44 ff.] bewertet. Die Qualität der Ergebnisse hängt von dem zugrunde liegenden Metadatenmodell und Datenmodell ab. Daher sind weitere Fortschritte in der Forschung erforderlich, um den Aufwand zu verringern. Ein Beispiel für die Forschung ist die automatische Identifikation von Abhängigkeiten zwischen Artefakttypen [27].

Validierungsergebnis: Anforderung 11 ist erfüllt.

A12: Nachvollziehbarkeit

Die Ergebnisse der Methodik müssen für Anwender nachvollziehbar sein.

Erfüllungskriterium: Die Anforderung gilt als „erfüllt“, wenn Industrieanwender die Ergebnisse als nachvollziehbar bewerten (\bar{x} Likert Skala > 1). Die Nachvollziehbarkeit wird durch die Rückverfolgbarkeit und Verständlichkeit sichergestellt. Rückverfolgbarkeit wird wie folgt interpretiert: *Rückverfolgbarkeit der Ursache-Wirkungs-Beziehungen zwischen den Artefakten der Systemelemente* [GWK22, S. 1887]. Verständlichkeit wird wie folgt interpretiert: *Informationen sind verständlich, wenn sie unmittelbar von den Anwendern verstanden und für deren Zwecke eingesetzt werden können.* [HGH15, S. 42]

Bewertung: Zur Bewertung werden die Diskussionen bei der Erstellung der Wirkpfade in den Experteninterviews analysiert und die subjektive Bewertung der Nachvollziehbarkeit der Probanden analysiert.

Walkthrough durch Wirkpfade in kontrollierten Experteninterviews:

Die implementierten Wirkpfade werden den Anwendern in dem Validierungs-Workshop präsentiert. Hierzu wird der Cypher-Code von exemplarischen Wirkkettenanalysen der unternehmensspezifischen Dashboards präsentiert und mit den Industrieanwendern durchgesprochen und diskutiert. Basierend auf den Rückfragen und Diskussionspunkten werden Schlussfolgerungen über die Rückverfolgbarkeit hergeleitet. Es wurden keine Rückfragen zu den Daten gestellt. (Anpassen auf Basis von Interviews)

Benchmarking der genutzten Abfrageverfahren: Die Verständlichkeit des Ergebnisrechnungsverfahrens hängt von den verwendeten Abfragesprachen ab. Im Fallbeispiel wird Cypher zur Implementierung der Algorithmen verwendet. Cypher wird im Vergleich zu anderen Abfragesprachen wie Gremlin [CN15] aufgrund der sehr gut verständlichen Syntax als einfach lesbar und wartbar bewertet, gerade für Anwender, die Structured Query Language (SQL) Kenntnisse mitbringen [Gue13, S. 200]. Laut verschiedenen Benchmarking-Studien (unter anderem [FGG18, HP13]) gilt Cypher aktuell als eine sich stetig weiter etablierende deklarative Abfragesprache, welche Anwendung in immer weitreichenderen Projekten findet [FGG18, S. 1444].

Bewertung der Industrieanwender anhand von Thesen

Neben der objektiven Bewertung der Nachvollziehbarkeit wird die subjektive Bewertung durch die Beantwortung einer These im Validierungs-Fragebogen bewertet (siehe Bild 6-20):

Die Entstehung der resultierenden Ergebnisse ist nachvollziehbar und erklärbar.

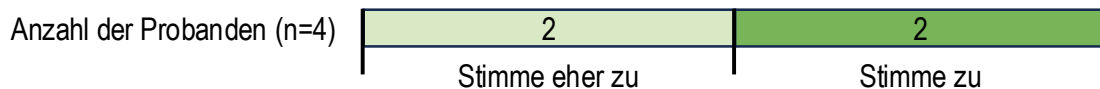


Bild 6-20: Beantwortung der These zur Nachvollziehbarkeit

Validierungsergebnis: Die Anforderung 12 ist erfüllt.

Insgesamt wurden alle zwölf Anforderungen erfüllt. Die Erfüllung der Anforderungen (Kapitel 3) verdeutlicht, dass die entwickelte MECA-Methodik die hergeleiteten Handlungsbedarfe (Kapitel 2.3) adressiert und einen Lösungsansatz zur frühzeitigen, effizienten und effektiven Handhabung technischer Änderungen darstellt. In Tabelle 6-5 wird das Ergebnis der Bewertung der Anforderungserfüllung zusammenfassend dargestellt. Lediglich der zeitliche Anwendungsaufwand (Anforderung 7) bietet Potenzial zur Verbesserung, beispielsweise durch Ansätze zur automatischen Extraktion [GÖ23, S. 169] von Artefakttypen oder die automatisierte Detektion von Relationen [GPO20, S. 200].

Tabelle 6-5: Anforderungserfüllung nach Validierung

	Anforderung 1	Anforderung 2	Anforderung 3	Anforderung 4	Anforderung 5	Anforderung 6	Anforderung 7	Anforderung 8	Anforderung 9	Anforderung 10	Anforderung 11	Anforderung 12
Erfüllungs-grad	●	●	●	●	●	●	◐	●	●	●	●	●

6.4 Methodik-Erfolg

Zusätzlich zu der Validierung der Anforderungen und der Bewertung der Anforderungserfüllung wird der Erfolg der Methodik übergreifend bewertet und diskutiert. Die Bewertung des Methodik-Erfolgs wird durch die Industrieanwender durchgeführt. Hierzu wurden sechs Thesen im Validierungs-Fragebogen (siehe Anhang A10.4, S. 195) auf einer Likert Skala [Lic32, 44 ff.] beantwortet (siehe Bild 6-21).

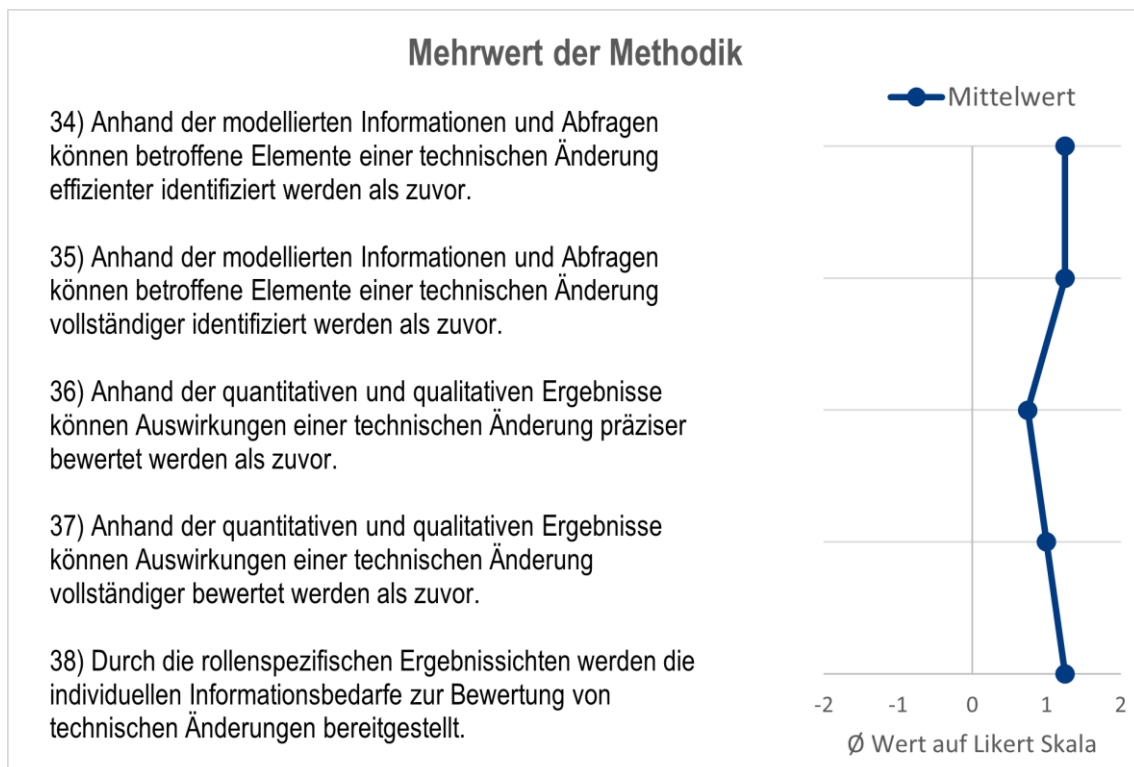


Bild 6-21: Mehrwert der Methodik

Die Auswertung der Validierungs-Fragebögen zeigt, dass die Industrieanwender allen Thesen zum Mehrwert der Methodik im Durchschnitt zustimmen oder eher zustimmen.

Zur Bewertung der Wirkkettenanalyse und der zugrunde liegenden architektonischen Konzepte wurden systematisch Ziele, relevante Fragen aus industriellen Fallbeispielen und Daten aus der Praxis gesammelt, welche als Bewertungsgrundlage des Nutzens verwendet werden können [KSF20, S. 249].

Dieses Ergebnis aus der Validierung und der Anwendung in den Fallstudien verdeutlicht, dass der Mehrwert der Methodik erkenntlich ist und der Einsatz der MECA-Methodik zu einer effektiveren und effizienteren Handhabung technischer Änderungen beiträgt. Die betroffenen Elemente werden effizient (Frage 34) und vollständig (Frage 35) identifiziert. Dadurch wird eine präzisere Bewertung der Auswirkungen möglich (Frage 36). Die Bewertung der Auswirkungen ist vollständig (Frage 37) als unterschiedlichen Ergebnissichten (Frage 38) möglich.

Durch die prototypische Implementierung der Methodik werden alle identifizierten Anwendungsfälle der modellbasierten Wirkkettenanalyse umgesetzt (siehe Anhang A3.2, S. 177.)

Die MECA-Methodik wurde in fünf Fallbeispielen gemeinsam mit industriellen Anwendern aus den Bereichen Automotive, Landtechnik, Sondermaschinenbau, Kunststoffverarbeitung und der metallverarbeitenden Industrie angewendet. Abschließend wurden in Expertenworkshops (WS 12) Lessons Learned (LL) zum Thema Wirkkettenanalysen erhoben (siehe [TIB25] und Anhang A4.11, S. 200):

- Die zu beantwortenden Fragestellungen der Wirkkettenanalyse müssen frühzeitig klar sein, damit man weiß, was modelliert werden muss. (LL15)
- Eine klare Vision für ein Ergebnis der Wirkkettenanalyse bildet die Grundlage für effiziente Modellierung. (LL17)
- Basisalgorithmen sind ein guter Startpunkt, um mit der Auswertung starten zu können. (LL24)
- Die Ergebnisse müssen klar den Sichten und Rollen zugeordnet werden können. (LL27)
- Durch die initiale Anwendung müssen sichtbare Benefits für einzelne Rollen dargestellt werden. (LL31)

Aufbauend auf den Erkenntnissen aus mehreren hundert Diskussionen und Workshops (siehe Anhang A1, S. 139) im Kontext der Wirkkettenmodellierung und -analyse wurde die MECA-Methodik stetig vom Verfasser weiterentwickelt. Zusätzlich wurde die MECA-Methodik in weiteren bilateralen Forschungsprojekten angewendet, um die Praxis zu beraten:

Bei dem Automobilzulieferer Paragon GmbH & Co. KGaA wurden Wirkketten eines Aktuatorsystems modelliert und analysiert [GWG25, S. 60]

Bei Behr-Hella Thermocontrol GmbH wurde die MECA-Methodik vorgestellt und hinsichtlich der Integration von Metriken für die Modellierungsqualität ergänzt [GWO24, S. 2545].

Auf der Fachtagung Tag des Systems Engineers im Jahr 2023 wurden die Erklärvideos vorgestellt und die MECA-Methodik mit Experten diskutiert. Zusammenfassend wurde der Methodik-Erfolg bestätigt und gezeigt, dass die MECA-Methodik zu einem frühzeitigeren, effektiveren und effizienten Umgang mit technischen Änderungen führt.

7 Zusammenfassung & Ausblick

Technische Änderungen sind während der Entwicklung komplexer technischer Systeme unvermeidbar [HCC13, S. 473]. In der Entwicklung komplexer technischer Systeme ist die Analyse und Bewertung technischer Änderungen eine zeitintensive Herausforderung für Unternehmen. Entwickler verbringen 30 %, teilweise sogar 50 %, ihrer Zeit mit der Bewertung und Umsetzung technischer Änderungen [LR98, S. 2, DKJ05, S. 4]. Gründe für den ineffizienten und ineffektiven Umgang mit technischen Änderungen sind die Bewertung der Auswirkungen technischer Änderungen auf Basis von subjektivem Expertenwissen und separierten Entwicklungsdaten, die in Insellösungen verwaltet werden.

Das vorliegende Dissertationsmanuskript umfasst eine Methodik zur effektiven und effizienten Bewertung und Analyse technischer Änderungen mithilfe von modellbasierten Wirkkettenanalysen – die **MECA (Model-based Effect Chain Analysis) Methodik**. Das wissenschaftliche Vorgehen folgt dem anwendungsorientierten Ansatz nach Ulrich [Ulr81], welcher sich durch die umfangreiche Einbindung der Praxis auszeichnet [Ulr81, S. 19]. Aktuelle Herausforderungen im Umgang mit technischen Änderungen wurden in einer initialen Feldstudie bei einem Automobilhersteller identifiziert. Zur vollständigen Erfassung und Typisierung praxisrelevanter Probleme wurden relevante Forschungsansätze aus den Bereichen der modellbasierten Systementwicklung und des technischen Änderungsmanagements identifiziert und anhand der identifizierten Herausforderungen analysiert und bewertet. Als Ausgangspunkt der Entwicklung der Methodik wurden Anforderungen an eine Methodik zur Wirkkettenanalyse literaturbasiert hergeleitet und durch industrielle Fachexperten bestätigt. Zusätzlich wurden Anwendungsfälle für modellbasierte Wirkkettenanalysen im Anwendungszusammenhang erhoben und aufgezeigt.

Aufbauend auf den Anforderungen wurden grundlegende Modelle der Methodik konzipiert. Hierzu wurden fünf Elemente für modellbasierte Wirkkettenanalysen identifiziert und in einem **konzeptionellen Architekturmodell** nach ISO 42010 [ISO11b] verankert. Zusätzlich wurde ein **Metdatenmodell** als zentraler Bestandteil des Architekturmodells konzipiert, welches Artefakt- und Relationstypen entlang der Entwicklungsartefakte RFLPV² umfasst und die interdisziplinären Ursache-Wirkungs-Beziehungen abbildet. Hinzukommend wurde ein **MBSE-Rollenmodell** konzipiert, welches Rollen zur Ausführung der modellbasierten Wirkkettenanalyse umfasst.

Aufbauend auf den konzipierten Modellen wurde die **MECA-Methodik** entwickelt, welche aus vier hauptsächlichen Schritten und einzelnen Hilfsmittel zur Durchführung der Schritte besteht. Die MECA-Methodik hat zum Ziel, technische Änderungen schnell, objektiv und rollenspezifisch zu gestalten. Im Vergleich zu den identifizierten Lösungsansätzen für die Entwicklung von konventionellen EC-Auswirkungsanalysen unterscheidet sich die MECA-Methodik durch die systematische und frühzeitige Zieldefinition der erwarteten Analyseergebnisse (1), die daraus resultierende Herleitung maßgeschneiderter durchgängiger Wirkkettenmodelle (2), die eigenständige Entwicklung rollenspezifischer Wirkkettenanalysen und Ergebnissichten (3) und die Umsetzung in einer im

Anwendungszusammenhang erprobten software-gestützten Implementierung (4). Diese vier Schritte mitsamt der Zwischenschritte sind notwendig, um Wirkkettenanalysen für die Bewertung technischer Änderungen nutzbar zu machen, wodurch Forschungsfrage 1 beantwortet wird.

In der MECA-Methodik werden die erwarteten Ergebnisse (Ergebnisdimensionen) der Wirkkettenanalyse systematisch mit den betroffenen Anwendern aufgenommen und spezifiziert. Die Durchführung von sechs Aktivitäten zur systematischen Zieldefinition verdeutlicht den Grundgedanken des **Frontloadings**. Die Anwendung der MECA-Methodik in unterschiedlichen Branchen hat gezeigt, dass die systematische Zieldefinition und die Herleitung rollenspezifischer Ergebnisdimensionen und Metriken unter Einbezug der Anwender einen kritischen Erfolgsfaktor darstellen. Ausgehend von den identifizierten Ergebnisdimensionen werden die zugrunde liegenden **Ursache-Wirkungs-Muster** analysiert und dokumentiert. Die Ursache-Wirkungs-Muster werden zu einem maßgeschneiderten Wirkkettenmodell zusammengeführt, welches lediglich die notwendigen Modellelemente enthält, um die benötigten Ergebnisse der Wirkkettenanalyse zu folgern. Durch das Maßschneidern der Wirkkettenmodelle wird im Vergleich zu etablierten Ansätzen der Modellierungsaufwand reduziert. Die **Einbindung von Partialmodellen** zur Detaillierung der Wirkkettenmodelle wird durch Modelltransformationen und die abschließende Integration der Modelle in ein durchgängiges Modell gewährleistet. Durch den modellbasierten Ansatz werden die identifizierten Ursache-Wirkungs-Muster in einem **durchgängigen Metadatenmodell** zusammengeführt. Zur Abbildung von Änderungspropagationen wurde ein graphbasiertes Metadatenmodell mit objektorientierten Modellierungsansätzen und Partialmodellen vernetzt, wodurch Forschungsfrage 2 beantwortet wird.

Aufbauend auf dem Wirkkettenmodell werden Anwender durch neu entwickelte und adaptierte **Analyseverfahren** und **Hilfsmittel** der MECA-Methodik in die Lage versetzt, Wirkkettenanalysen eigenständig zu definieren und durchzuführen. Die Wirkkettenanalysen werden mithilfe von Abfragesprachen umgesetzt und umfassen die Identifikation betroffener Elemente entlang eines spezifischen Ursache-Wirkungs-Musters und die Bewertung der Auswirkungen durch zugehörige Bewertungsmetriken. Ergebnisse der Wirkkettenanalysen werden mithilfe der **maßgeschneiderten Cypher-Abfragen** innerhalb von weniger als 0.5 Sekunden gefolgert. Die direkte Integration von Bewertungsmetriken in die Cypher-Abfragen ermöglicht die **qualitative und quantitative Bewertung der Auswirkungen**. Aufgrund der **geringeren Komplexität der Abfragesprachen** können die Queries von Ingenieuren ohne Informatikkenntnisse definiert werden ([GW23, S. 5] aufbauend auf [Kem15]). Durch die Entwicklung wiederverwendbarer Basisalgorithmen und die Rückführung der Erkenntnisse der Wirkkettenanalyse wird die Informationszirkularität (siehe [GP21, S. 13]) innerhalb des Produktentstehungsprozesses gefördert. Die gefolgerten Ergebnisse werden in **maßgeschneiderten Ergebnissichten** bereitgestellt. Auf Basis der operativen Nutzung der Ergebnissichten werden **rollen-spezifische Dashboards** entwickelt, welche von Anwendern in der operativen Anwendung zur effizienten

und effektiven Bewertung technischer Änderungen verwendet werden. Die beschriebenen Analyseverfahren eignen sich, um Auswirkungen technischer Änderungen systematisch zu bewerten, wodurch Forschungsfrage 3 beantwortet wird.

Durch die **software-gestützte Implementierung** werden zeitaufwändige, planungsintensive Besprechungen von Experten ersetzbar und die Einbindung von Lieferanten vereinfacht. Die iterative Berücksichtigung von Lessons Learned führt zur Sicherstellung der praktischen Anwendbarkeit. Zusätzlich wurde die Entwicklung der MECA-Methodik und der software-gestützten Implementierung anhand realer technischer Änderungen durchgeführt, wodurch die **domänenübergreifende Anwendbarkeit** Ergebnisse in den Bereichen Automotive, Landtechnik, Sondermaschinenbau und Kunststoffverarbeitung bestätigt wurde. Anhand der realen technischen Änderungen konnten die Anwender die **Relevanz in der Praxis** abschätzen und den Mehrwert der Methodik bestätigen. Insgesamt werden Anwender durch die MECA-Methodik bei der effizienten, effektiven und frühzeitigen Analyse von Änderungsauswirkungen unterstützt.

Prototypische Implementierungen in mehreren Unternehmen aus unterschiedlichen Domänen wurde einer vollumfänglichen Implementierung in der Praxis vorgezogen. Die im Dissertationsmanuskript beschriebene prototypische Implementierung in verschiedenen IT-Systemen wie SysML-Systemmodellen und Graphdatenbanken verdeutlicht die unterschiedlichen Arten der IT-gestützten Umsetzung. Zusätzlich wird das Potenzial deutlich, modellbasierte Wirkkettenanalysen als Funktionalität in PLM-Systeme zu integrieren (siehe Bild 7-1).

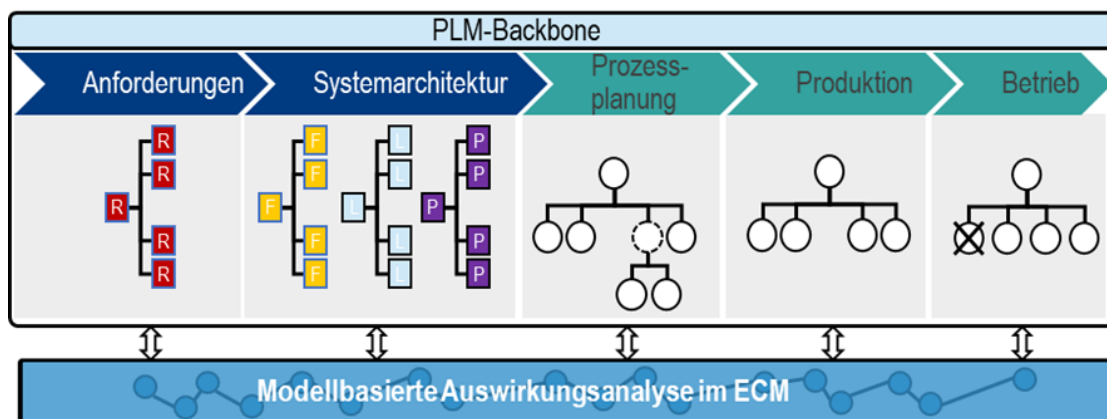


Bild 7-1: Konzeptionelle Einbindung von Wirkkettenanalysen in PLM-Systeme (angelehnt an [Eig21, S. 86])

Durch Fortschritte in weiteren Forschungsgebieten der digitalen und virtuellen Produktentwicklung besteht die Möglichkeit, modellbasierte Wirkkettenanalysen basierend auf Informationen aus vor- und nachgelagerten Phasen des Produktlebenszyklus zu erweitern. Potenzielle weitere Anwendungsfälle, in denen Wirkkettenanalysen als Hilfsmittel eingesetzt werden können, wurden durch den Verfasser in Vorpublikationen untersucht. Drei Anwendungsbeispiele werden im Folgenden dargestellt.

Die MECA-Methodik kann beispielsweise zur Wirkkettenanalyse innerhalb eines dynamischen Systems of Systems (SoS) angewendet werden, um Auswirkungen zwischen einem System und weiteren Umfeldsystemen zu analysieren [GWR24, S. 585]. Beispielhafte SoS sind autonom fahrende, kollaborierende Landmaschinen im Bereich Smart Farming oder vernetzte Produktionssysteme im Bereich Smart Manufacturing [AEF23, 15 f.]. Innerhalb des dynamischen SoS können in diesem Anwendungsfall Auswirkungen auf die Resilienz, Interoperabilität, Performanz oder die Regularieneinhaltung haben, beispielsweise zur Erbringung des CO²-Nachweises zur Nachhaltigkeitsbewertung [Sau24, Sau23]. Die grundlegende Nutzbarkeit von Wirkkettenanalysen im Bereich der Nachhaltigkeitsbewertung wird im Forschungsprojekt Decide4ECO erforscht [GPH24, S. 180].

Ein weiteres Anwendungsbeispiel der MECA-Methodik ist die Bewertung von Auswirkungen von Angriffen im Kontext von Defense in Depth [GBW23]. Die Bewertung der Auswirkungen unterstützt Entwickler bei der Gewährleistung von Sicherheit und Zuverlässigkeit ihres Systems im Anwendungskontext und bei der Entwicklung von Verteidigungselementen gegenüber potenziellen Bedrohungen und Angriffen.

Das letzte Anwendungsbeispiel umfasst die Erweiterung der Wirkketten um Informationen aus der Produktion. Modellierungssprachen wie SysML eignen sich zur Abbildung des Systemverhaltens im Kontext digitaler Zwillinge [WSS22, S. 1823]. Mithilfe von MBSE kann System auf die gesamte Lebensdauer durchgängig abgebildet werden [ZMS20, S. 54]. Wirkkettenmodelle fungieren hierbei als auswertbares Modell der Entwicklung entlang von RFLPV², welches systematisch um Informationen aus der Produktion erweitert werden kann und so zu einem Digitalen Zwilling jeder Instanz wird. Produktionsdaten und -informationen können frühzeitig eingebunden werden, beispielsweise durch die Integration von Ansätzen des Simultaneous Engineering entlang des V-Modells [GWR21a] oder durch die modellbasierte Integration menschlicher Faktoren (*Human Factors*) in der Produktion [GWR21b].

Literaturverzeichnis

- [AAB18] ANGLES, R.; ARENAS, M.; BARCELÓ, P.; HOGAN, A.; REUTTER, J.; VRGOČ, D.: Foundations of Modern Query Languages for Graph Databases. *ACM Computing Surveys*, (50)5, 2018, S. 1–40
- [Abe07] ABERDEEN GROUP: Änderungsmanagement 2.0 – Verbesserte Unternehmensentscheidungen durch intelligentes Änderungsmanagement, 2007
- [AEF23] ADLER, R.; ELBERZHAGER, F.; FALCÃO, R.; SIEBERT, J.; GROEN, E. C.; HEINRICH, J.; BALDUF, F.; LIGGESMEYER, P.: A research roadmap for trustworthy dynamic systems of systems—motivation challenges and research directions.", 2023
- [Alt12] ALT, O.: Modellbasierte Systementwicklung mit SysML. Carl Hanser Verlag, München, 2012
- [AM21] ALEKSANDRAVIČIENĖ, A.; MORKEVIČIUS, A.: MagicGrid® BOOK OF KNOWLEDGE – A Practical Guide to Systems Modeling using MagicGrid from Dassault Systèmes. 2. Auflage, 2021
- [AW17] ARNOLD, R. D.; WADE, J. P.: A complete set of systems thinking skill. *INSIGHT*, (20)3, 2017, S. 9–17
- [AWC11] AHMAD, N.; WYNN, D. C.; CLARKSON, J. P.: Information models used to manage engineering change: A Review of literature 2005-2010. In: Design Society (Hrsg.): DS 68-1: Proceedings of the 18th International Conference on Engineering Design (ICED 11) – Impacting Society through Engineering Design, Vol. 1: Design Processes, Denmark, 15.-19.08. 2011., 2011
- [BBW17] BAJAJ, M.; BACKHAUS, J.; WALDEN, T.; WAIKAR, M.; ZWEMER, D.; SCHREIBER, C.; ISSA, G.; MARTIN, L.: Graph-Based Digital Blueprint for Model Based Engineering of Complex Systems. *INCOSE International Symposium*, (27)1, 2017, S. 151–169
- [BC09] BLESSING, L. T. M.; CHAKRABARTI, A.: DRM, a design research methodology. Springer, Dordrecht, 2009
- [BC94] BASILI, V. R.; CALDIERA, GIANLUIGI, ROMBACH, DIETER H.: The Goal Question Metric Approach: Encyclopedia of software engineering, 1994, S. 528–532
- [BFK21] BAJZEK, M.; FAUSTMANN, C.; KREMS, D.; KRANABITL, P.; HICK, H.: A concept for a digital thread based on the connection of system models and specific models. *INCOSE International Symposium*, (31)1, 2021, S. 1035–1048
- [BG21a] BENDER, B.; GERICKE, K.: Entwickeln der Anforderungsbasis: Requirements Engineering. In: Bender, B.; Gericke, K. (Hrsg.): Pahl/Beitz Konstruktionslehre: Methoden und Anwendung erfolgreicher Produktentwicklung. Springer Berlin Heidelberg, Berlin, Heidelberg, 2021, S. 169–209
- [BG21b] BENDER, B.; GERICKE, K. (Hrsg.): Pahl/Beitz Konstruktionslehre: Methoden und Anwendung erfolgreicher Produktentwicklung. Springer Berlin Heidelberg, Berlin, Heidelberg, 2021
- [BGP24] BESTA, M.; GERSTENBERGER, R.; PETER, E.; FISCHER, M.; PODSTAWSKI, M.; BARTHEL, C.; ALONSO, G.; HOEFLER, T.: Demystifying Graph Databases: Analysis and Taxonomy of Data Organization, System Designs, and Graph Queries. *ACM Computing Surveys*, (56)2, 2024, S. 1–40
- [BH18] BAJAJ, M.; HEDBERG, T.: System Lifecycle Handler — Spinning a Digital Thread for Manufacturing. *INCOSE International Symposium*, (28)1, 2018, S. 1636–1650
- [BKM09] BANGOR, A.; KORTUM, P.; MILLER, J.: Determining what individual SUS scores mean: Adding an adjective rating scale: *Journal of Usability Studies*. Band 3, 2009
- [BL00] BAKER, L.; LONG, J. E.: Role of System Engineering Across The System Life Cycle, 2000

- [Ble94] BLESSING, L. T. M.: A process-based approach to computer-supported engineering design. Zugl.: Enschede, Univ. Twente, Diss., 1994. Black Bear Press, Cambridge, 1994
- [BLM09] BOGNER, A.; LITTIG, B.; MENZ, W.: Interviewing Experts. Palgrave Macmillan UK, London, 2009
- [BLO06] BRIAND, L. C.; LABICHE, Y.; O’SULLIVAN, L.; SÓWKA, M. M.: Automated impact analysis of UML models. *Journal of Systems and Software*, (79)3, 2006, S. 339–352
- [Boc96] BOCHTLER, W.: Modellbasierte Methodik für eine integrierte Konstruktion und Arbeitsplanung. Shaker, 1996
- [Bro] BROOKE, J.: SUS: A quick and dirty usability scale: Usability evaluation in industry. 194. Auflage, 1996, S. 4–7
- [BS22] BALAKRISHNAN, A. S.; SURESH, J.: Engineering changes - research findings and future directions. *International Journal of Enterprise Network Management*, (13)1, 2022, S. 66
- [BW23] BRAHMA, A.; WYNN, D. C.: Concepts of change propagation analysis in engineering design. *Research in Engineering Design*, (34)1, 2023, S. 117–151
- [CC12] CHENG, H.; CHU, X.: A network-based assessment approach for change impacts on complex product. *Journal of Intelligent Manufacturing*, (23)4, 2012, S. 1419–1431
- [CN15] CHENEY, J.; NEUMANN, T.: Proceedings of the 15th Symposium on Database Programming Languages. SPLASH '15: Conference on Systems, Programming, Languages, and Applications: Software for Humanity, 27 10 2015 27 10 2015, Pittsburgh PA USA, ACM, New York, NY, USA, 2015
- [CNF00] COHEN, T.; NAVATHE, S. B.; FULTON, R. E.: C-FAR, change favorable representation. *Computer-Aided Design*, (32)5-6, 2000, S. 321–338
- [CSE04] CLARKSON, P. J.; SIMONS, C.; ECKERT, C.: Predicting Change Propagation in Complex Design. *Journal of Mechanical Design*, (126)5, 2004, S. 788–797
- [Del14] DELLIGATTI, L.: SysML distilled – A brief guide to the systems modeling language. Addison-Wesley, Upper Saddle River, NJ, 2014
- [DIN18] Kunststoff-Formteile – Toleranzen und Abnahmebedingungen, 2018
- [DIN22] DIN 199-4:2022-09 – Technische Produktdokumentation (TPD) - Teil 4: Begriffe im Änderungswesen; Text Deutsch und Englisch, 2022
- [DKJ05] DEUBZER, F.; KREIMEYER, M.; JUNIOR, T.; ROCK, B.: Der Änderungsmanagement Report 2005. CiDaD Working Paper Series, Nr. 01/05, 2005, S. 1–12
- [DKL06] DEUBZER, F.; KREIMEYER, M.; LINDEMANN, U.: EXPLORING STRATEGIES IN CHANGE MANAGEMENT – CURRENT STATUS AND ACTIVITY BENCHMARK. DS 36: Proceedings DESIGN 2006, the 9th International Design Conference, Dubrovnik, Croatia, 2006, S. 815–822
- [Dor02] DORI, D.: Object-Process Methodology. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002
- [DSS17] DICKOPF, T.; SCHULTE, T.; SCHNEIDER, M.: Analyse existierender SysML-basierter Ansätze aus Industrie und Forschung. In: Eigner, M.; Koch, W.; Muggeo, C. (Hrsg.): Modellbasierter Entwicklungsprozess cybertronischer Systeme. Springer Berlin Heidelberg, Berlin, Heidelberg, 2017, S. 65–72
- [ECZ04] ECKERT, C.; CLARKSON, P. J.; ZANKER, W.: Change and customisation in complex engineering domains. *Research in Engineering Design*, (15)1, 2004, S. 1–21
- [Eig09] EIGNER, M.: Product Lifecycle Management – Ein Leitfaden für Product Development und Life Cycle Management. 2. Auflage, VDI-Buch, Springer, Berlin, 2009

- [Eig21] EIGNER, M.: System Lifecycle Management – Digitalisierung des Engineering. Springer Vieweg, Berlin, 2021
- [Epp12] EPPINGER, S. D.: Design structure matrix methods and applications. MIT Press, Cambridge, Mass, 2012
- [Epp91] EPPINGER, S. D.: Model-based Approaches to Managing Concurrent Engineering. *Journal of Engineering Design*, (2)4, 1991, S. 283–290
- [ES99] EVERSHEIM, W.; SCHULTEN, I.: Optimal Degree of Parallelism and Integration in Design and Process Planning. In: Kals, H.; van Houten, F. (Hrsg.): *Integration of Process Knowledge into Design Support Systems*. Springer Netherlands, Dordrecht, 1999, S. 119–128
- [Est08] ESTEFAN, J. A.: Survey of Model-Based Systems Engineering (MBSE) Methodologies, 2008
- [Eur03] Regelung Nr. 21 der Wirtschaftskommission der Vereinten Nationen für Europa (UN/ECE) – Einheitliche Vorschriften für die Genehmigung von Kraftfahrzeugen hinsichtlich ihrer Innenausstattung., 2003
- [EW23] ESTEFAN, J. A.; WEILKIENS, T.: MBSE Methodologies. In: Madni, A. M.; Augustine, N.; Sievers, M. (Hrsg.): *Handbook of Model-Based Systems Engineering*. Springer International Publishing, Cham, 2023, S. 47–85
- [FES03] FLANAGAN, T. L.; ECKERT, C. M.; SMITH, J.; EGER, T.; CLARKSON, J. P.: A functional analysis of change propagation: Proceedings of the 14th International Conference on Engineering Design (ICED'03). Stockholm, Sweden, 2003
- [FGG18] FRANCIS, N.; GREEN, A.; GUAGLIARDO, P.; LIBKIN, L.; LINDAAKER, T.; MARSAULT, V.; PLANTIKOW, S.; RYDBERG, M.; SELMER, P.; TAYLOR, A.: Cypher: An Evolving Query Language for Property Graphs. In: Das, G.; Jermaine, C.; Bernstein, P. (Hrsg.): *Proceedings of the 2018 International Conference on Management of Data. SIGMOD/PODS '18: International Conference on Management of Data*, 10 06 2018 15 06 2018, Houston TX USA, ACM, New York, NY, USA, 2018, S. 1433–1445
- [FGN00] FRICKE, E.; GEBHARD, B.; NEGELE, H.; IGENBERGS, E.: Coping with changes: Causes, findings, and strategies. *Systems Engineering*, (3)4, 2000, S. 169–179
- [FGO11] FEI, G.; GAO, J.; OWODUNNI, O.; TANG, X.: A method for engineering design change analysis using system modelling and knowledge management techniques. *International Journal of Computer Integrated Manufacturing*, (24)6, 2011, S. 535–551
- [Fre00] FREI, N.: Programm zur Auslegung mechanischer Wirkketten: 11. Symposium "Design For X", 2000
- [Fri14] FRIEDENTHAL, S.: A Practical Guide to SysML – The Systems Modeling Language. 3rd Edition, Elsevier Science, San Francisco, 2014
- [GBP21] GERICKE, K.; BENDER, B.; PAHL, G.; BEITZ, W.; FELDHOUSEN, J.; GROTE, K.-H.: Funktionen und deren Strukturen. In: Bender, B.; Gericke, K. (Hrsg.): *Pahl/Beitz Konstruktionslehre: Methoden und Anwendung erfolgreicher Produktentwicklung*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2021, 233–254
- [GBW23] GRÄBLER, I.; BODDEN, E.; WIECHEL, D.; POTTEBAUM, J.: Defense-in-Depth als neues Paradigma der sicherheitsgerechten Produktentwicklung – interdisziplinäre, bedrohungsbewusste und lösungsorientierte Security., In: VDI Verlag (Hrsg.): *Zeitschrift Konstruktion*. 11. Auflage, 2023
- [GDB19] GRÄBLER, I.; DATTNER, M.; BOTHEN, M.: Main feature list as core success criteria of organizing requirements elicitation. UB-PAD - Paderborn University Library, 2019
- [GH20] GUSENBAUER, M.; HADDAWAY, N. R.: Which academic search systems are suitable for systematic reviews or meta-analyses? Evaluating retrieval qualities of Google Scholar, PubMed, and 26 other resources. *Research synthesis methods*, (11)2, 2020, S. 181–217

- [GO22a] GRÄBLER, I.; OLEFF, C.: Voraussetzungen für die industrielle Umsetzung. In: Gräßler, I.; Oleff, C. (Hrsg.): Systems Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg, 2022, S. 29–82
- [GO22b] GRÄBLER, I.; OLEFF, C.: Einführung in Systems Engineering. In: Gräßler, I.; Oleff, C. (Hrsg.): Systems Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg, 2022, S. 1–27
- [GO22c] GRÄBLER, I.; OLEFF, C.: Kernelement Systemdenken – Ganzheitlich denken und handeln. In: Gräßler, I.; Oleff, C. (Hrsg.): Systems Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg, 2022, S. 83–108
- [GO22d] GRÄBLER, I.; OLEFF, C. (Hrsg.): Systems Engineering. Springer Berlin Heidelberg, Berlin, Heidelberg, 2022
- [GÖ23] GRÄBLER, I.; ÖZCAN, DENIZ, PREUB, DANIEL: AI-based extraction of requirements from regulations for automotive engineering: DS 125: Proceedings of the 34th Symposium Design for X (DFX2023). The 34th Symposium Design for X, 14 and 15 September 2023, The Design Society, 14 and 2023, S. 163–172
- [GOH19] GRÄBLER, I.; OLEFF, C.; HENTZE, J.: Role Model for Systems Engineering Application. Proceedings of the Design Society: International Conference on Engineering Design, (1)1, 2019, S. 1265–1274
- [GOP22] GRÄBLER, I.; OLEFF, C.; PREUB, D.: Proactive Management of Requirement Changes in the Development of Complex Technical Systems. Applied Sciences, (12)4, 2022, S. 1874
- [GOS20] GRAESSLER, I.; OLEFF, C.; SCHOLLE, P.: Method for Systematic Assessment of Requirement Change Risk in Industrial Practice. Applied Sciences, (10)23, 2020, S. 8697
- [GP21] GRÄBLER, I.; POTTEBAUM, J.: Generic Product Lifecycle Model: A Holistic and Adaptable Approach for Multi-Disciplinary Product–Service Systems, (11)10, 2021
- [GP22] GRÄBLER, I.; POTTEBAUM, J.: From Agile Strategic Foresight to Sustainable Mechatronic and Cyber-Physical Systems in Circular Economies. In: Krause, D.; Heyden, E. (Hrsg.): Design Methodology for Future Products. Springer International Publishing, Cham, 2022, S. 3–26
- [GPH24] GRAESSLER, I.; POTTEBAUM, J.; HOLLAND, M.; WIECHEL, D.; DICKOPF, T.; STJEPANDIĆ, J.: Leveraging Data Ecosystems in Model-Based Systems Engineering for Ecological, Circular Added Value. In: Cooper, A.; Trigos, F.; Stjepandić, J.; Curran, R.; Lazar, I. (Hrsg.): Engineering For Social Change. Advances in Transdisciplinary Engineering, IOS Press, 2024
- [GPO20] GRÄBLER, I.; PREUB, D.; OLEFF, C.: Automatisierte Identifikation und Charakterisierung von Anforderungsabhängigkeiten – Literaturstudie zum Vergleich von Lösungsansätzen: Proceedings of the 31st Symposium Design for X (DFX2020). Proceedings of the 31st Symposium Design for X, 16-17 December 2020, The Design Society, 2020, S. 199–208
- [Grä00] GRÄBLER, I.: Informations-und zeitbasiertes Controlling einer integrierten Konstruktion und Arbeitsplanung. Shaker, Aachen, 2000
- [Grä15] GRÄBLER, I.: Umsetzungsorientierte Synthese mechatronischer Referenzmodelle: Implementation-oriented synthesis of mechatronic reference models, 2015
- [GRW24] GRAESSLER, I.; RARBACH, S.; WIECHEL, D.: Artifact-Oriented Tailoring Approach for Model-Based Impact Analysis: 2024 IEEE International Symposium on Systems Engineering (ISSE). 2024 IEEE International Symposium on Systems Engineering (ISSE), 16.10.2024 - 19.10.2024, Perugia, Italy, IEEE, 2024, S. 1–8
- [Gue13] GUERRINI, G.: Proceedings of the Joint EDBT/ICDT 2013 Workshops. EDBT/ICDT '13: Joint 2013 EDBT/ICDT Conferences, 18 03 2013 22 03 2013, Genoa Italy, ACM, New York, NY, USA, 2013

- [GW21] GRÄBLER, I.; WIECHEL, D.: Systematische Bewertung von Auswirkungsanalysen des Engineering Change Managements: Proceedings of the 32nd Symposium Design for X (DFX2021). DFX 2021 32nd SYMPOSIUM DESIGN FOR X, 2021, pp. 1–10
- [GW23] GRÄBLER, I.; WIECHEL, D.: Customized impact analyses for technical engineering changes: 2023 18th Annual System of Systems Engineering Conference (SoSe). 2023 18th Annual System of Systems Engineering Conference (SoSe), 14.06.2023 - 16.06.2023, Lille, France, IEEE, 2023, S. 1–6
- [GWB09] GIFFIN, M.; WECK, O. DE; BOUNOVA, G.; KELLER, R.; ECKERT, C.; CLARKSON, P. J.: Change Propagation Analysis in Complex Technical Systems. Journal of Mechanical Design, (131)8, 2009
- [GWG25] GRÄBLER, I.; WIECHEL, D.; GREWE, B.; BOTH, A.: Bedarfsgerechte Wirkkettenmodellierung und -analyse komplexer akusto-mechatronischer Systeme/Effect-chain modeling of complex acoustic-mechatronic systems. Konstruktion, (77)03, 2025, S. 60–65
- [GWK22] GRÄBLER, I.; WIECHEL, D.; KOCH, A.-S.; PREUß, D.; OLEFF, C.: Model-Based Effect-Chain Analysis for Complex Systems. Proceedings of the Design Society, (2), 2022, S. 1885–1894
- [GWK23] GRÄBLER, I.; WIECHEL, D.; KOCH, A.-S.; STURM, T.; MARKFELDER, T.: Methodology for Certification-Compliant Effect-Chain Modeling. Systems, (11)3, 2023, S. 154
- [GWO22] GRÄBLER, I.; WIECHEL, D.; OLEFF, C.: Extended RFLP for complex technical systems: 2022 IEEE International Symposium on Systems Engineering (ISSE). 2022 IEEE International Symposium on Systems Engineering (ISSE), 24.10.2022 - 26.10.2022, Vienna, Austria, IEEE, 2022, S. 1–8
- [GWO24] GRAESSLER, I.; WIECHEL, D.; OEZCAN, D.; TAPLICK, P.: Tailored metrics for assessing the quality of MBSE models. Proceedings of the Design Society, (4), 2024, S. 2545–2554
- [GWP21] GRÄBLER, I.; WIECHEL, D.; POTTEBAUM, J.: Role model of model-based systems engineering application. IOP Conference Series: Materials Science and Engineering, (1097)1, 2021, S. 12003
- [GWR21a] GRÄBLER, I.; WIECHEL, D.; ROESMANN, D.; THIELE, H.: V-model based development of cyber-physical systems and cyber-physical production systems. Procedia CIRP, (100), 2021, S. 253–258
- [GWR21b] GRÄBLER, I.; WIECHEL, D.; ROESMANN, D.: Integrating human factors in the model based development of cyber-physical production systems. Procedia CIRP, (100), 2021, S. 518–523
- [GWR24] GRAESSLER, I.; WIECHEL, D.; RARBACH, S.: Model-based impact analysis in dynamic System of Systems. Procedia CIRP, (128), 2024, S. 585–590
- [GZQ22] GUO, Y.; ZHAO, H.; QI, K.; GUO, J.: Analysis of the Mainstream MBSE Methodologies from the Modeling Practice View. In: Tallón-Ballesteros, A. J. (Hrsg.): Proceedings of CECNet 2022. Frontiers in Artificial Intelligence and Applications, IOS Press, 2022
- [Ham13] HAMRAZ, B.: Engineering change modelling using a function-behaviour-structure scheme, Apollo - University of Cambridge Repository, 2013
- [HBC22] HOGAN, A.; BLOMQVIST, E.; COCHEZ, M.; D'AMATO, C.; MELO, G. DE; GUTIERREZ, C.; KIRrane, S.; GAYO, J. E. L.; NAVIGLI, R.; NEUMAIER, S.; NGOMO, A.-C. N.; POLLERES, A.; RASHID, S. M.; RULA, A.; SCHMELZEISEN, L.; SEQUEDA, J.; STAAB, S.; ZIMMERMANN, A.: Knowledge Graphs. ACM Computing Surveys, (54)4, 2022, S. 1–37
- [HBF19] HICK, H.; BAJZEK, M.; FAUSTMANN, C.: Definition of a system model for model-based development. SN Applied Sciences, (1)9, 2019

- [HCC13] HAMRAZ, B.; CALDWELL, N. H. M.; CLARKSON, P. J.: A Holistic Categorization Framework for Literature on Engineering Change Management. *Systems Engineering*, (16)4, 2013, S. 473–505
- [HCG10] HEVNER, A. R.; CHATTERJEE, S.; GRAY, P.; BALDWIN, C. Y.: *Design research in information systems – Theory and practice*. Springer, Vol. 22 New York, NY, 2010
- [Hey99] HEYN, M. (Hrsg.): *Methodik zur schnittstellenorientierten Gestaltung von Entwicklungskooperationen*. Zugl.: Aachen, Techn. Hochsch., Diss., 1998. *Berichte aus der Produktionstechnik* Bd. 99,6, Shaker, Aachen, 1999
- [HGH15] HILDEBRAND, K.; GEBAUER, M.; HINRICHS, H.; MIELKE, M. (Hrsg.): *Daten- und Informationsqualität – Auf dem Weg zur Information Excellence*. 3. Auflage, Springer Vieweg, Wiesbaden, 2015
- [HKS21] HICK, H.; KÜPPER, K.; SORGER, H. (Hrsg.): *Systems Engineering for Automotive Powertrain Development*. Powertrain, Springer International Publishing, Cham, 2021
- [HM99] HUANG, G. Q.; MAK, K. L.: Current practices of engineering change management in UK manufacturing industries. *International Journal of Operations & Production Management*, (19)1, 1999, S. 21–37
- [HME10] HÖLTTÄ, V.; MAHLAMÄKI, K.; EISTO, T.; STRÖM, M.: *Lean Information Management Model for Engineering Changes*: World Academy of Science, Engineering and Technology. Band 42, 2010
- [HMP04] HEVNER, A. R.; MARCH, S. T.; PARK, J.; RAM, S.: Design Science in Information Systems Research. *MIS Quarterly*, (28)1, 2004, S. 75
- [Hof13] HOFFMANN, H.-P.: *Systems Engineering Best Practices with the Rational Solution for Systems and Software Engineering Deskbook Release 4.1*, 2013
- [Hol21] HOLT, J.: *Systems engineering demystified – A practitioner's handbook for developing complex systems using a model-based approach*. Packt, Birmingham, 2021
- [HP13] HOLZSCHUHER, F.; PEINL, R.: Performance of graph query languages. In: Guerrini, G. (Hrsg.): *Proceedings of the Joint EDBT/ICDT 2013 Workshops*. EDBT/ICDT '13: Joint 2013 EDBT/ICDT Conferences, 18 03 2013 22 03 2013, Genoa Italy, ACM, New York, NY, USA, 2013, S. 195–204
- [HST23] HICK, H.; SANLADERER, S.; TRAUTNER, J.; RYAN, K.; PIGUET, J.; WILKING, F.; HORBER, D.; FAUSTMANN, C.; KRANABITL, P.; KOLLEGER, S.; BAJZEK, M.; SCHLEICH, B.; WARTZACK, S.: Combining System Models and CAD for Change Scenario Management. *INCOSE International Symposium*, (33)1, 2023, S. 117–132
- [HWF19] HABERFELLNER, R.; WECK, O. DE; FRICKE, E.; VÖSSNER, S.: *Systems Engineering*. Springer International Publishing, Cham, 2019
- [HYM03] HUANG, G.; YEE, W.; MAK, K.: Current practice of engineering change management in Hong Kong manufacturing industries. *Journal of Materials Processing Technology*, (139)1-3, 2003, S. 481–487
- [IEE] IEEE Standard Glossary of Software Engineering Terminology
- [INC] INCOSE TECHNICAL OPERATIONS: *Systems Engineering Vision 2020*. 2. Auflage, Seattle
- [INC15] INCOSE: *INCOSE Systems Engineering Handbook – A Guide for System Life Cycle Processes and Activities*. 4th Edition, Wiley, New York, 2015
- [IRA20] IAKYMENKO, N.; ROMSDAL, A.; ALFNES, E.; SEMINI, M.; STRANDHAGEN, J. O.: Status of engineering change management in the engineer-to-order production environment: insights from a multiple case study. *International Journal of Production Research*, (58)15, 2020, S. 4506–4528

- [IRS18] IAKYMENKO, N.; ROMSDAL, A.; SEMINI, M.; STRANDHAGEN, J. O.: Managing engineering changes in the engineer-to-order environment: challenges and research needs. IFAC-PapersOnLine, (51)11, 2018, S. 144–151
- [ISO08] INTERNATIONAL ORGANISATION FOR STANDARDIZATION: ISO/IEC 15288-2008 - System life cycle processes, 2008
- [ISO11a] INTERNATIONAL ORGANISATION FOR STANDARDIZATION: ISO/IEC 25010:2011 Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE). ISO/IEC, 2011
- [ISO11b] INTERNATIONAL ORGANISATION FOR STANDARDIZATION: ISO/IEC/IEEE 42010:2011 Systems and software engineering -- Architecture description. IEEE, Piscataway, NJ, USA, 2011
- [Jar19] JAROSCHEK, C.: Spritzgießen für Praktiker. 4. Auflage, Hanser, München, 2019
- [JCE05] JARRATT, T.; CLARKSON, J.; ECKERT, C.: Engineering change. In: Clarkson, J.; Eckert, C. (Hrsg.): Design process improvement. Springer London, London, 2005, S. 262–285
- [JEC11] JARRATT, T. A. W.; ECKERT, C. M.; CALDWELL, N. H. M.; CLARKSON, P. J.: Engineering change: an overview and perspective on the literature. Research in Engineering Design, (22)2, 2011, S. 103–124
- [JJS21] JAGLA, P.; JACOBS, G.; SIEBRECHT, J.; WISCHMANN, S.; SPREHE, J.: Using SysML to Support Impact Analysis on Structural Dynamics Simulation Models. Procedia CIRP, (100), 2021, S. 91–96
- [JL18] JAYATILLEKE, S.; LAI, R.: A systematic review of requirements change management. Information and Software Technology, (93), 2018, S. 163–185
- [Kem15] KEMPER, C.: Beginning Neo4j. Apress Springer Science+Business Media, New York, NY, 2015
- [KH13] KAIRANBAY; HAJAR: Shortest Path Alg. In: IJSTR (Hrsg.): INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH. VOLUME 2, ISSUE 6, 2013, S. 99–104
- [KJ16] KARTHIK, K.; JANARDHAN REDDY, K.: Engineering Changes in Product Design - A Review. IOP Conference Series: Materials Science and Engineering, (149), 2016, S. 12001
- [KK13] KLEINER, S.; KRAMER, C.: Model Based Design with Systems Engineering Based on RFLP Using V6. In: Abramovici, M.; Stark, R. (Hrsg.): Smart Product Engineering. Lecture Notes in Production Engineering, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, S. 93–102
- [KM10] KRAMER, T.; MÜNZENBERGER, R.: Echtzeitanalyse komplexer Wirkketten in Fahrerassistenzsystemen: Vortragsband zur 3. AutoTest. Stuttgart, 2010
- [Köh09] KÖHLER, C. M.: Technische Produktänderungen Analyse und Beurteilung von Lösungsmöglichkeiten auf Basis einer Erweiterung des CPM/PDD-Ansatzes. Universität des Saarlandes, 2009
- [KP10] KROSINICK, J. A.; PRESSER, S.: Question and Questionnaire Design. In: Marsden, P. V.; Wright, J. D. (Eds.): Handbook of survey research. Emerald Group Publ, Bingley, 2010
- [Kra20] KRAUß, L. D.: Entwicklung einer repräsentativen Unternehmensstruktur zur Identifizierung der Herausforderungen von Model-Based Systems Engineering. unveröffentlichte Studienarbeit. Paderborn, 2020
- [Krc15] KRCMAR, H.: Informationsmanagement. 6. Auflage, Springer Gabler, Wiesbaden, 2015
- [Kre10] KREIMEYER, M. F.: A structural measurement system for engineering design processes. Zugl.: München, Techn. Univ., Diss., 2009. Verl. Dr. Hut, München, 2010

- [Kru00] KRUCHTEN, P.: The Rational Unified Process--An Introduction, 2000
- [KSF20] KOSSIAKOFF, A.; SEYMOUR, S. J.; FLANIGAN, D. A.; BIEMER, S. M.: Systems engineering – Principles and practice. Wiley, Hoboken, NJ, 2020
- [KW15] KUDALE, A. E.; WANJALE, K. H.: Human Computer Interaction Model based Virtual Whiteboard: A Review: International Journal of Computer Applications. 130. Auflage, 2015, S. 44–47
- [KWS07] KORDON, M.; WALL, S.; STONE, H.; BLUME, W.; SKIPPER, J.; INGHAM, M.; NEELON, J.; CHASE, J.; BAALKE, R.; HANKS, D.; SALCEDO, J.; SOLISH, B.; POSTMA, M.; MACHUZAK, R.: Model-Based Engineering Design Pilots at JPL: 2007 IEEE Aerospace Conference. 2007 IEEE Aerospace Conference, 03.03.2007 - 10.03.2007, Big Sky, MT, USA, IEEE, 2007, S. 1–20
- [Las21] LASHIN, G.: Technisches Änderungsmanagement. In: Bender, B.; Gericke, K. (Hrsg.): Pahl/Beitz Konstruktionslehre: Methoden und Anwendung erfolgreicher Produktentwicklung. Springer Berlin Heidelberg, Berlin, Heidelberg, 2021, S. 919–942
- [Lau96] LAUFENBERG, L.: Methodik zur integrierten Projektgestaltung für die situative Umsetzung des Simultaneous Engineering. Shaker, 1996
- [Lic32] LIKERT, R.: A technique for the measurement of attitudes. New York, 1932
- [Loc20] LOCHBICHLER, M.: Systematische Wahl einer Modellierungstiefe im Entwurfsprozess mechatronischer Systeme, UB-PAD - Paderborn University Library, 2020
- [LR98] LINDEMANN, U.; REICHWALD, R. (Hrsg.): Integriertes Änderungsmanagement. Springer eBook Collection Computer Science and Engineering, Springer Berlin Heidelberg, Berlin, Heidelberg, 1998
- [LWM12] LANGER, S.; WILBERG, J.; MAIER, A.; LINDEMANN, U.: Änderungsmanagement-Report 2012 – Studienergebnisse zu Ursachen und Auswirkungen, aktuelle Praktiken, Herausforderungen und Strategien in Deutschland, 2012
- [MCP14] MHENNI, F.; CHOLEY, J.-Y.; PENAS, O.; PLATEAUX, R.; HAMMADI, M.: A SysML-based methodology for mechatronic systems architectural design. Advanced Engineering Informatics, (28)3, 2014, S. 218–231
- [MGW16] MOESER, G.; GRUNDEL, M.; WEILKIENS, T.; KÜMPEL, S.; KRAMER, C.; ALBERS, A.: Modellbasierter mechanischer Konzeptentwurf: Ergebnisse des FAS4M-Projektes. In: Schulze, S.-O.; Tschirner, C.; Kaffenberger, R.; Ackva, S. (Hrsg.): Tag des Systems Engineering. Carl Hanser Verlag GmbH & Co. KG, München, 2016, S. 417–428
- [MHH17] MILLER, A. M.; HARTMAN, N. W.; HEDBERG, T.; BARNARD FEENEY, A.; ZAHNER, J.: Towards Identifying the Elements of a Minimum Information Model for Use in a Model-Based Definition: Volume 3: Manufacturing Equipment and Systems. ASME 2017 12th International Manufacturing Science and Engineering Conference collocated with the JSME/ASME 2017 6th International Conference on Materials and Processing, 04.06.2017 - 08.06.2017, Los Angeles, California, USA, American Society of Mechanical Engineers, 2017
- [MHT23] MORDASCHEW, V.; HERRMANN, J.-P.; TACKENBERG, S.: METHODS OF CHANGE IMPACT ANALYSIS FOR PRODUCT DEVELOPMENT: A SYSTEMATIC REVIEW OF THE LITERATURE. Proceedings of the Design Society, (3), 2023, S. 2655–2664
- [Mil13] MILLER, J. J.: Graph Database Applications and Concepts with Neo4j: Proceedings of the Southern Association for Information Systems Conference. Atlanta, March 23rd–24th, 2013, S. 141–147
- [MJL17] MA, S.; JIANG, Z.; LIU, W.; HUANG, C.: Design Property Network-Based Change Propagation Prediction Approach for Mechanical Product Development. Chinese Journal of Mechanical Engineering, (30)3, 2017, S. 676–688

- [MKP22] MARTIN, A.; KASPAR, J.; PFEIFER, S.; CONSTANTIN, M.; SIMON, R.; ALBERT, A.: Advanced Engineering Change Impact Approach (AECIA) – Towards a model-based approach for a continuous Engineering Change Management: 2022 IEEE International Symposium on Systems Engineering (ISSE). 2022 IEEE International Symposium on Systems Engineering (ISSE), 24.10.2022 - 26.10.2022, Vienna, Austria, IEEE, 2022, S. 1–7
- [MM12] MACHI, L. A.; MCEVOY, B. T.: The literature review – Six steps to success. 2nd Edition, Corwin Press, Thousand Oaks, Calif., 2012
- [MMA22] MANDEL, C.; MARTIN, A.; ALBERS, A.: Addressing Factors for User Acceptance of Model-Based Systems Engineering. In: Bitetti, L. (Ed.): Innovating in a digital world – XXXIII ISPIM Innovation Conference 5-8 June 2022, Copenhagen, Denmark. LUT scientific and expertise publications, International Society for Professional Innovation Management Ltd, Worsley, Manchester, United Kingdom, 2022
- [MMR09] MAIER, M.; MAIER, M. W.; RECHTIN, E.: The art of systems architecting. 3rd Edition, CRC Press, Boca Raton, Fla., 2009
- [MS18] MADNI, A. M.; SIEVERS, M.: Model-based systems engineering: Motivation, current status, and research opportunities. Systems Engineering, (21)3, 2018, S. 172–190
- [MSS12] MORKOS, B.; SHANKAR, P.; SUMMERS, J. D.: Predicting requirement change propagation, using higher order design structure matrices: an industry case study. Journal of Engineering Design, (23)12, 2012, S. 905–926
- [NAS07] NASA: NASA Systems Engineering Handbook. 2. Auflage, 2007
- [Neo22] NEO4J INC.: The Neo4j Cypher Manual v4.1, 2022
- [Nus15] NUSSBAUMER KNAFLIC, C.: Storytelling with data – a data visualization guide for business professionals. Wiley, Hoboken, NJ, 2015
- [Obj11] Business Process Model and Notation (BPMN), Version 2.0, 2011
- [Obj17] OBJECT MANAGEMENT GROUP: Unified Modeling Language, v2.5.1, 2017
- [Ole22] OLEFF, C.: Proaktives Management von Anforderungsänderungen in der Entwicklung komplexer technischer Systeme. Universität Paderborn, 2022
- [OMG19] OMG: OMG Systems Modeling Language (OMG SysML™) – Version 1.6, 2019
- [Ope24] OPENAI: Chat GPT. Unter: <https://chat.openai.com/>, 13. Februar 2024
- [OS01] OLLINGER, G. A.; STAHOVICH, T. F.: RedesignIT: A Constraint-Based Tool for Managing Design Changes: Volume 4: 13th International Conference on Design Theory and Methodology. ASME 2001 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 09.09.2001 - 12.09.2001, Pittsburgh, Pennsylvania, USA, American Society of Mechanical Engineers, 2001, S. 197–207
- [PBF07] PAHL, G.; BEITZ, W.; FELDHUSEN, J.; GROTE, K.-H.: Konstruktionslehre. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007
- [PDH11] PARNELL, G.S.; DRISCOLL, P.J.; HENDERSON, D.L. (Eds.): Decision making in systems engineering and management. 2nd Edition, Wiley, Hoboken, NJ, 2011
- [PHA12] POHL, K.; HÖNNINGER, H.; ACHATZ, R.; BROY, M.: Model-Based Engineering of Embedded Systems. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012
- [PM98] PIKOSZ, P.; MALMQVIST, J.: A Comparative Study of Engineering Change Management in Three Swedish Engineering Companies. ASME 1998 Design Engineering Technical Conferences, 1998

- [PR21] POHL, K.; RUPP, C.: Basiswissen Requirements Engineering – Aus- und Weiterbildung nach IREB-Standard zum Certified Professional for Requirements Engineering Foundation Level. 5. Auflage, dpunkt.verlag, Heidelberg, 2021
- [PT07] PARKER, A.; TRITTER, J.: Focus group method and methodology: current practice and recent debate. *International Journal of Research & Method in Education*, 2007, S. 23–37
- [PW12] PASQUAL, M. C.; WECK, O. L. DE: Multilayer network model for analysis and management of change propagation. *Research in Engineering Design*, (23)4, 2012, S. 305–328
- [RJ01] RAMESH, B.; JARKE, M.: Toward reference models for requirements traceability. *IEEE Transactions on Software Engineering*, (27)1, 2001, S. 58–93
- [RKM15] ROHWEDER, J. P.; KASTEN, G.; MALZAHN, D.; PIRO, A.; SCHMID, J.: Informationsqualität – Definitionen, Dimensionen und Begriffe. In: Hildebrand, K.; Gebauer, M.; Hinrichs, H.; Mielke, M. (Hrsg.): *Daten- und Informationsqualität – Auf dem Weg zur Information Excellence*. 3. Auflage, Springer Vieweg, Wiesbaden, 2015
- [RM15] REMPEL, P.; MADER, P.: A quality model for the systematic assessment of requirements traceability: 2015 IEEE 23rd International Requirements Engineering Conference (RE). IEEE, 2015
- [Rup21] RUPP, C.: *Requirements-Engineering und -Management – Das Handbuch für Anforderungen in jeder Situation*. 7. Auflage, Hanser, München, 2021
- [Sau16] SAUER, B.: *Konstruktionselemente des Maschinenbaus 1*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016
- [Sau23] SAUERWALD, M.: *Methode zur Integration von Systemmodellen in die Nachhaltigkeitsbewertung*. unveröffentlichte Studienarbeit. Paderborn, 2023
- [Sau24] SAUERWALD, M.: *Modellbasierte Bewertung der Auswirkungen von Entwicklungsentscheidungen auf den CO2-Fußabdruck*. unveröffentlichte Masterarbeit. Paderborn, 2024
- [Sch14] SCHOENEBERG, K.-P. (Hrsg.): *Komplexitätsmanagement in Unternehmen – Herausforderungen im Umgang mit Dynamik, Unsicherheit und Komplexität meistern*. Springer Gabler, Wiesbaden, 2014
- [Sch15] SCHINDEL, W. D.: *Pattern-Based Systems Engineering (PBSE), Based On S*MBSE Models*, 2015
- [Sch23] SCHINDEL, W. D.: Pattern-Based Methods and MBSE. In: Madni, A. M.; Augustine, N.; Sievers, M. (Hrsg.): *Handbook of Model-Based Systems Engineering*. Springer International Publishing, Cham, 2023, S. 151–194
- [Sch89] SCHMIDT, G.: *Relationen und Graphen*. Mathematik für Informatiker, Springer, Berlin, 1989
- [She96] SHEARD, S. A.: TWELVE SYSTEMS ENGINEERING ROLES. *INCOSE International Symposium*, (6)1, 1996, S. 478–485
- [SLG15] SUBRAHMANIAN, E.; LEE, C.; GRANGER, H.: Managing and supporting product life cycle through engineering change management for a complex product. *Research in Engineering Design*, (26)3, 2015, S. 189–217
- [SMS12] SHANKAR, P.; MORKOS, B.; SUMMERS, J. D.: Reasons for change propagation: a case study in an automotive OEM. *Research in Engineering Design*, (23)4, 2012, S. 291–303
- [Som07] SOMMERVILLE, I.: *Software engineering*. 8. Auflage, International computer science series, Addison-Wesley, Harlow, 2007
- [Sta22] STARK, R.: *Virtual Product Creation in Industry*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2022

- [SZS21] SCHMIDT, M. M.; ZIMMERMANN, T. C.; STARK, R.: Systematic Literature Review of System Models for Technical System Development. *Applied Sciences*, (11)7, 2021, S. 3014
- [TD05] TAVČAR, J.; DUHOVNIK, J.: Engineering change management in individual and mass production. *Robotics and Computer-Integrated Manufacturing*, (21)3, 2005, S. 205–215
- [TIB25] Schlussbericht: mPaKT – IKT-befähigte modellbasierte Auswirkungsanalyse in der Produktentwicklung, 2025
- [Tor22] TORIELLO, M.: Creating the Digital Thread. *INSIGHT*, (25)1, 2022, S. 34–37
- [Tur10] TURAU, V.: Algorithmische Graphentheorie. 3. Auflage, Oldenbourg, München, 2010
- [Ulr81] ULRICH, H.: Die Betriebswirtschaftslehre als anwendungsorientierte Sozialwirtschaft. In: Geist, M. (Hrsg.): Die Führung des Betriebes – Curt Sandig zu seinem 80. Geburtstag gewidmet. Pöschel, Stuttgart, 1981, S. 1–27
- [UML] UML 2 SUPERSTRUCTURE FTF: United Modeling Language 2.0 Proposal
- [Uni21] UN Regulation No. 156, 2021
- [Uni24] UNIVERSITÄT PADERBORN: ImPaKT Website. Unter: <https://www.impaktprojekt.de/>, 31. Januar 2023
- [UTY15] ULLAH, I.; TANG, D.; YIN, L.: Engineering Change Implications on Product Design: A Review of the Literature: Proceedings of the 2015 International Conference on Education, Management and Computing Technology. International Conference on Education, Management and Computing Technology (ICEMCT-15), 13.06.2015 - 14.06.2015, Tianjin, China, Advances in Social Science, Education and Humanities Research, Atlantis PressParis, France, 2015
- [UTY16] ULLAH, I.; TANG, D.; YIN, L.: Engineering Product and Process Design Changes: A Literature Overview. *Procedia CIRP*, (56), 2016, S. 25–33
- [VDA09a] VDA VERBAND DER AUTOMOBILINDUSTRIE: VDA 4965 - ECM Recommendation – Part 0 (ECM), 2009
- [VDA09b] VDA VERBAND DER AUTOMOBILINDUSTRIE: VDA 4965 ECM Recommendation Part – Part 1 (ECR), 2009
- [VDI21] VDI: VDI/VDE 2206. 2021. Auflage, 2021
- [Ver19] VDI 2221: Entwicklung technischer Produkte und Systeme - Modell der Produktentwicklung – Blatt 1, 2019
- [Ver93] VDI 2221 – Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte, 1993
- [Voi23] VOIRIN, J.-L.: ARCADIA USER GUIDE – Arcadia Principles and Contents Overview, 2023
- [VWZ18] VAJNA, S.; WEBER, C.; ZEMAN, K.; HEHENBERGER, P.; GERHARD, D.; WARTZACK, S.: CAx für Ingenieure. Springer Berlin Heidelberg, Berlin, Heidelberg, 2018
- [WBC20] WORTMANN, A.; BARAIS, O.; COMBEMALE, B.; WIMMER, M.: Modeling languages in Industry 4.0: an extended systematic mapping study. *Software and Systems Modeling*, (19)1, 2020, S. 67–94
- [Wei07] WEILKIENS, T.: Systems engineering with SysML/UML – Modeling, analysis, design. Elsevier Morgan Kaufmann OMG Press, Amsterdam, 2007
- [Wei16] WEILKIENS, T.: Variant modeling with SysML. MBSE4U, Fredesdorf, 2016
- [Wic17] WICKEL, M. C.: Änderungen besser managen – Eine datenbasierte Methodik zur Analyse technischer Änderungen. Dissertation, Technische Universität München, 2017

- [WKI20] WILMS, R.; KRONSEIN, P.; INKERMANN, D.; HUTH, T.; REIK, M.; VIETOR, T.: USING A CROSS-DOMAIN PRODUCT MODEL TO SUPPORT ENGINEERING CHANGE MANAGEMENT. Proceedings of the Design Society: DESIGN Conference, (1), 2020, S. 1165–1174
- [Wri97] WRIGHT, I. C.: A review of research into engineering change management: implications for product design. Design Studies, (18)1, 1997, S. 33–42
- [WSD16] WEILKIENS, T.; SCHEITHAUER, A.; DI MAIO, M.; KLUSMANN, N.: Evaluating and comparing MBSE methodologies for practitioners: 2016 IEEE International Symposium on Systems Engineering (ISSE). 2016 IEEE International Symposium on Systems Engineering (ISSE), 03.10.2016 - 05.10.2016, Edinburgh, United Kingdom, IEEE, 2016, S. 1–8
- [WSS22] WILKING, F.; SAUER, C.; SCHLEICH, B.; WARTZACK, S.: SysML 4 Digital Twins – Utilization of System Models for the Design and Operation of Digital Twins. Proceedings of the Design Society, (2), 2022, S. 1815–1824
- [YD12] YANG, F.; DUAN, G.: Developing a parameter linkage-based method for searching change propagation paths. Research in Engineering Design, (23)4, 2012, S. 353–372
- [ZBL14] ZHENG, C.; BRICOGNE, M.; LE DUIGOU, J.; EYNARD, B.: Survey on mechatronic engineering: A focus on design methods and product models. Advanced Engineering Informatics, (28)3, 2014, S. 241–257
- [ZMS20] ZIMMERMANN, T. C.; MASUHR, C.; STARK, R.: MBSE-Entwicklungsfähigkeit für Digitale Zwillinge. Zeitschrift für wirtschaftlichen Fabrikbetrieb, (115)s1, 2020, S. 51–54.

Zitation aus studentischen Arbeiten

Die nachstehend aufgeführten studentischen Arbeiten wurden im Kontext der vorliegenden Dissertation am Lehrstuhl für Produktentstehung der Universität Paderborn angefertigt. Die Definition der Zielsetzung, die Bearbeitung sowie die Auswertung, Interpretation und Visualisierung von Ergebnissen erfolgten unter wissenschaftlicher Anleitung der Betreuenden Prof. Dr.-Ing. Iris Gräßler und Dominik Wiechel. Die erzielten Ergebnisse sind zum Teil in die Dissertation eingeflossen.

- [Kra20] Krauß, L. D. (Betreuer: Gräßler, I.; Wiechel, D.): Entwicklung einer repräsentativen Unternehmensstruktur zur Identifizierung der Herausforderungen von Model-Based Systems Engineering. Universität Paderborn, Heinz Nixdorf Institut – Lehrstuhl für Produktentstehung, unveröffentlichte Bachelorarbeit, 2020.
- [Wer21] Werner, T. (Betreuer: Gräßler, I.; Wiechel, D.): Modellbasierte Abbildung von Abhängigkeiten bei der Entwicklung von Produktvarianten. Universität Paderborn, Heinz Nixdorf Institut – Lehrstuhl für Produktentstehung, unveröffentlichte Studienarbeit, 2021.
- [Oez21] Özcan, D. (Betreuer: Gräßler, I.; Wiechel, D.): Modellbasierte Abbildung von Änderungsbedarfen und Abhängigkeiten zum Anforderungsmodell. Universität Paderborn, Heinz Nixdorf Institut – Lehrstuhl für Produktentstehung, unveröffentlichte Studienarbeit, 2021.
- [Buh22] Buhl, M. (Betreuer: Gräßler, I.; Wiechel, D.): Referenzprozess zur Integration von modellbasierten Engineering Change-Auswirkungsanalysen in Entwicklungsprozesse. Universität Paderborn, Heinz Nixdorf Institut – Lehrstuhl für Produktentstehung, unveröffentlichte Bachelorarbeit, 2022.
- [Sch22b] Schäfer, J. (Betreuer: Gräßler, I.; Wiechel, D.): Tailoring-Leitfaden zur bedarfsgerechten Einbindung von disziplinspezifischen Partialmodellen in SysML-Systemmodelle. Universität Paderborn, Heinz Nixdorf Institut – Lehrstuhl für Produktentstehung, unveröffentlichte Masterarbeit, 2022.
- [Kau23] Kaup, C. (Betreuer: Gräßler, I.; Wiechel, D.): Leitfaden zur Integration von Partialmodellen in ein SysML-Systemmodell. Universität Paderborn, Heinz Nixdorf Institut – Lehrstuhl für Produktentstehung, unveröffentlichte Studienarbeit, 2023.
- [Sau23] Sauerwald, M. (Betreuer: Gräßler, I.; Wiechel, D.): Methode zur Integration von Systemmodellen in die Nachhaltigkeitsbewertung. Universität Paderborn, Heinz Nixdorf Institut – Lehrstuhl für Produktentstehung, unveröffentlichte Studienarbeit, 2023.
- [Rar23] Rarbach, S. (Betreuer: Gräßler, I.; Wiechel, D.): Erarbeitung eines Tailoring-Leitfadens zur Implementierung einer MBSE-Referenzarchitektur. Universität Paderborn, Heinz Nixdorf Institut – Lehrstuhl für Produktentstehung, unveröffentlichte Masterarbeit, 2023.
- [Sau24] Sauerwald, M. (Betreuer: Gräßler, I.; Wiechel, D.): Modellbasierte Bewertung der Auswirkungen von Entwicklungsentscheidungen auf den CO₂-Fußabdruck. Universität Paderborn, Heinz Nixdorf Institut – Lehrstuhl für Produktentstehung, unveröffentlichte Masterarbeit, 2024.

Anhang

A1	Übersicht über Forschungsprojekte und Workshops mit Industrievertretern	139
A2	Literaturrecherchen.....	141
A2.1	Identifikation relevanter Literatur- und Fallstudien	141
A2.2	Liste an Ansätzen zur modellbasierten Auswirkungsanalyse	144
A2.3	Kriterien zur Auswahl der EC-Auswirkungsanalyse-Ansätze.....	150
A2.4	Steckbrief EC-Auswirkungsanalyse Methoden	152
A2.5	Gegenüberstellung der Ansätze anhand der Elemente	167
A2.6	Liste an Ansätzen zu Modellierungsmethoden	168
A2.7	Steckbriefe MBSE-Modellierungsmethoden	170
A2.8	Fragebogen zur Ermittlung der Herausforderungen	173
A3	Anforderungen und Anwendungsfälle.....	176
A3.1	Anwendungsfälle der Auswirkungsanalyse.....	176
A3.2	Nummerierung der Anwendungsfäll	177
A3.3	Anforderungsworkshops zur Wirkkettenanalyse.....	178
A3.4	Interviews für Anforderungen an die Wirkkettenmodellierung in Automobilindustrie.....	179
A4	Konzeption der MECA-Methodik	180
A4.1	Vorgehen zur Erstellung des Rollenmodells	180
A4.2	Modellierungsregeln	181
A4.3	Ergebnisdimensionen	181
A4.4	RFLPV Handouts.....	184
A4.5	Vollständige Ansicht der Customizations.....	186
A4.6	RFLPV ² -Profil	186
A4.7	Metamodell Graphdatenbank neo4j.....	188

A4.8	Ursache-Wirk-Muster für Wirkkettenanalysen	190
A4.9	Basisalgorithmen	191
A4.10	MECA Poster	195
A4.11	Lessons Learned	200
A5	Fallstudie kollaborativer 6-achsiger Roboterarm Nyrio Ned 2.....	202
A5.1	Übersicht der Fallstudie	202
A5.2	Modelle Roboterarm (Nyrio Ned 2)	204
A5.3	Greifvorgang bei Pick-and-Place-Bewegung	210
A5.4	Neo4j Import Roboterarm	212
A5.5	Dashboards Roboterarm	213
A6	Zusatzinformationen ImPaKT	216
A6.1	Übersichtsposter Fallbeispiele	216
A6.2	Übersicht zur Anwendung der MECA-Methodik in Fallstudien	216
A6.3	Partialmodell je Fallstudie	218
A7	Fallstudie CLAAS und Eisengiesserei Baumgarte	220
A8	Fallstudie Hofmann	226
A9	Fallstudie Hadi Plast.....	238
A10	Validierung	247
A10.1	Bewertungsmethoden nach Hevner	247
A10.2	Aufbau Prüfstand	248
A10.3	Kriterien Planung des Prüfablaufs	249
A10.4	Validierungs-Workshops (Zyklus 1-3)	250
A10.5	Validierungs-Fragebogen	251
A10.6	Auswertung der Fragebögen	259

A11	Code.....	260
A11.1	Code zum Import des Metadatenmodells	260
A11.2	Dashboards Roboterarm	267

A1 Übersicht über Forschungsprojekte und Workshops mit Industrievertretern

FP-1: ImPaKT Im BMBF-geförderten Forschungsprojekt ImPaKT („IKT-befähigte modellbasierte Auswirkungsanalyse in der Produktentwicklung“) wurde ein Verbund aus 10 Partnern gebildet, um modellbasierte Auswirkungsanalysen zu entwickeln. Von akademischer Seite waren neben dem Lehrstuhl für Produktentstehung des Heinz Nixdorf Instituts das Institut für Maschinenelemente und Systementwicklung (MSE) der RWTH Aachen beteiligt. Die akademischen Partner wurden durch die IT-Partner PROSTEP AG, Contact Software GmbH und itemis AG ergänzt. Drei industrielle Unternehmen haben die Anwenderperspektive vertreten und Fallbeispiele eingebracht: Hofmann Mess- und Auswuchttechnik GmbH & Co. KG (HFM), Hadi Plast GmbH & Co. KG Kunststoff-Verarbeitung (HP), Eisengiesserei Baumgarte GmbH (EB). Als assoziierte Partner waren CLAAS Industrietechnik GmbH (CIT), Knapheide GmbH Hydrauliksysteme und Schaeffler Technologies AG & Co. KG beteiligt.

Tabelle A-1: Nummerierung der Workshops

Nummer	Titel	Art			Anzahl Ter- mine
		E	S1	S2	
WS 1	Anwendungsfalldefinition	X			2
WS 2	Anforderungserhebung	X			2
WS 3	Ergebnisdimensionen	X			3
WS 4	Validierungs-Workshop Zyklus 1	X			1
WS 5	Modellierungsworkshop CIT / EB		X		8
WS 6	Modellierungsworkshop HFM		X		38
WS 7	Modellierungsworkshop HP		X		26
WS 8	Validierungs-Workshop Zyklus 2	X			1
WS 9	Kernteam Auswirkungsanalyse		X		4
WS 10	Workshop zu Wirkkettenanalyse	X			36
WS 11	Validierungs-Workshop Zyklus 3	X			1
WS 12	Workshop zum „Praxistransfer“			X	14
WS 13	Validierungs-Workshops mit HFM, HP, CIT, EB	X			4
Workshops ImPaKT					141

E: Einzelworkshop, S1: Serientermin einwöchentlich, S2: Serientermin zweiwöchentlich

FP-2: Automobil-OEM

In dem einjährigen Forschungsprojekt wurden Wirkkettenmodelle für eine Baureihe eines deutschen Automobilherstellers entwickelt. Durch das Eintreten der UN ECE Regularie 156 müssen die Wirkkettenmodelle entwickelt werden, um technische Änderungen von

Software-Updates bewerten zu können. Innerhalb des Forschungsprojekts wurden mehr als 300 Workshops durchgeführt, um Daten zu erheben, Wirkkettenmodelle je Fahrzeugfunktionalität zu modellieren und diese mit den Experten zu validieren. Ausgehend von Erkenntnissen aus den Herausforderungen bei der Wirkkettenmodellierung wurden die MECA-Methodik entwickelt und für die Entwicklung von Wirkkettenanalysen spezifisch angepasst.

Tabelle A-2: Workshops mit Automobil OEM

Nummer	Titel	Anzahl Termine
WS 14	Interviews zur Anforderungserhebung mit OT-Expertin, Projektmanager und Consulting Partner	3
WS 15	Datenerhebung, Modellierung und Validierung von Wirkkettenmodellen von >100 Subsystemen eines Kraftfahrzeugs	>300

FP-3 Paragon: Nach der initialen Entwicklung der MECA-Methodik wurde die Methodik bei dem Automobilzulieferer Paragon angewendet. Als Fallbeispiel wurde ein Dachantrieb ausgewählt, der in Dachsystemen verschiedener Automobilserien verbaut wird. Die Erhebung der Artefakte und Relationen wurde entlang des V-Modells der VDI/VDE 2206:2021 durchgeführt. Neben den Disziplinen Mechanik, Elektrik/Elektronik und Software wurde ebenfalls die Akustik explizit berücksichtigt. Durch die Betrachtung der Ursache-Wirkungs-Beziehungen zwischen den interdisziplinären Systemelementen können akusto-mechatronische Phänomene identifiziert und deren Auswirkungen analysiert werden.

Tabelle A-3: Workshops bei Automobilzulieferer Paragon

Nummer	Titel	Anzahl Termine
WS-16	Anwendung der MECA-Methodik für Fallbeispiel Dachantrieb	9

FP-4 BHTC: Bei dem Automobilzulieferer Behr-Hella Thermocontrol (BHTC) wurde die unternehmensspezifische Modellierungsmethodik analysiert, explizit abgebildet und durch den Vergleich mit existierenden MBSE-Modellierungsansätzen angepasst. Zusätzlich wurde eine Methode entwickelt, um die Modellqualität von Systemmodellen zu spezifischen Zeitpunkten während der modellbasierten Systementwicklung bewerten zu können. Hierzu wurden Kontrollpunkte entlang des V-Modells definiert, Fragen hergeleitet und Metriken den Fragen zugeordnet.

Tabelle A-4: Workshops bei Automobilzulieferer BHTC

Nummer	Titel	Anzahl Termine
WS-18	Interviews zur Erhebung der Modellierungsmethodik	5
WS-19	Erarbeitung der Kontrollfragen	6

A2 Literaturrecherchen

A2.1 Identifikation relevanter Literatur- und Fallstudien

Basierend auf der Analyse und Bewertung von Literaturplattformen nach GUSENBAUER und HADDAWAY [GH20, 205 ff.] wurden die Datenbanken Science Direct und Web of Knowledge gewählt.

Literaturstudien: DEUBZER ET AL. [DKJ05], JARRATT ET AL. [JEC11], LANGER ET AL. [LWM12], HAMRAZ ET AL. [HCC13], GRAESSLER und WIECHEL [GW21], BALAKRISHNAN und SURESH [BS22], JAYATILLEKE UND LAI [JL18], BRAHMA UND WYNN [BW23]; MORDASCHEW ET AL. [MHT23].

Industrielle Fallstudien: PIKOSZ UND MALMQVIST [PM98], HUANG und MAK [HM99], HUANG UND YEE [HYM03], SHANKAR ET AL. [SMS12], KARTHIK und JANARDHAN [KJ16] und IAKYMENKO ET AL. [IRS18] UND IAKYMENKO ET AL. [IRA20].

Ziel: Identifikation dokumentierter aktueller Herausforderungen in der Handhabung von Engineering Changes in der Praxis

Suchstrings:

("Engineering change management" OR "engineering changes" OR "Impact Analysis" OR "Change Propagation Analysis") AND ("Survey" OR "Case Study") AND ("product development" OR "product creation" OR "model-based Systems engineering")

Einschränkungen Google Scholar: Übersichtsarbeiten -"production management" -"production system" -"supply chain management" -"organizational change management"

(„Änderungsmanagement“ OR „Technische Änderungen“ OR "Auswirkungsanalyse" OR "propagation analyse") AND ("Studie" OR "Literaturstudie" OR "fallstudie") AND ("produktentwicklung" OR "modellbasierte systementwicklung")

Einschränkung Google Scholar: -"produktionssystem" -"nachhaltigkeit,, - "projektmanagement"

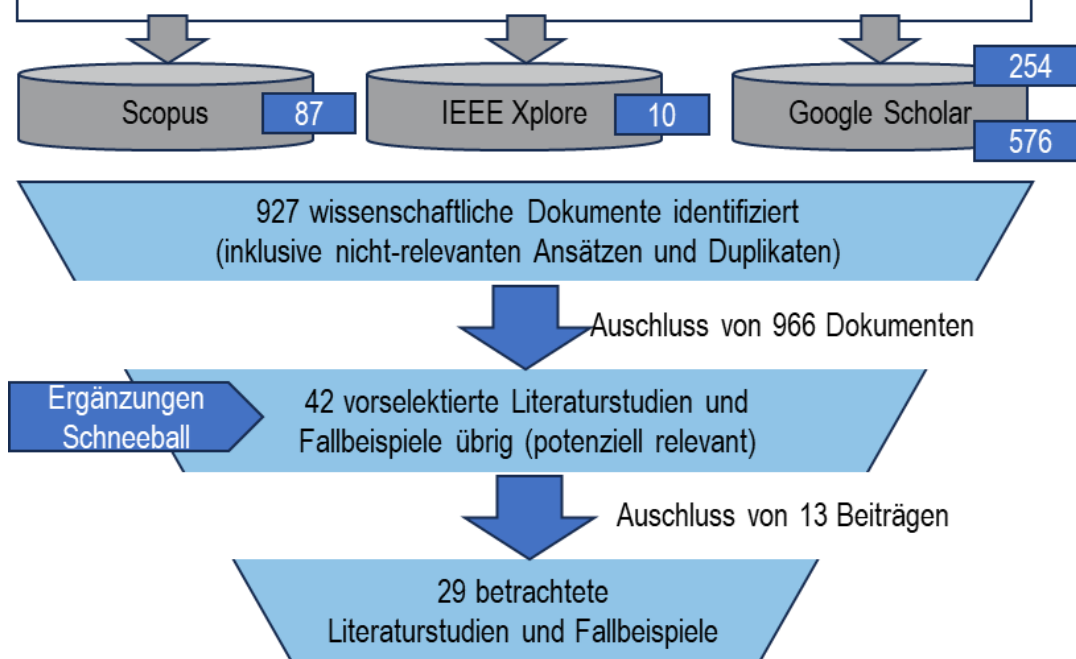


Bild A-1: Übersicht der Literaturrecherche zur Identifikation von Herausforderungen (schematische Darstellung in Anlehnung an WORTMANN ET AL. [WBC20, S. 72] und der studentischen Arbeit von KRAUß [Kra20])

Tabelle A-5: Übersicht zur Literaturstudie

Literaturstudien				
Nr.	Autor / Institution	Titel	Jahr	Quelle
1	Ahmad, N A., Wynn, D. C.; Clarkson, J. P.	Information models used to manage engineering change: a review of Literature 2005-2010	2011	[AWC11]
2	Balakrishnan, A.S.; Suresh, J.*	Engineering changes – research findings and future directions	2022	[BS22]
3	Deubzer, F.; Kreimeyer, M.; Junior, T.; Rock, B.	Der Änderungsmanagement Report 2005	2005	[DKJ05]
4	Gräßler, I.; Wiechel, D.	Systematische Bewertung von Auswirkungenanalysen des Engineering Change Managements	2021	[GW21]
5	Hamraz, B.; Caldwell, N. H. M.; Clarkson, J. P.	A Holistic Categorization Framework for Literature on Engineering Change Management	2012	[HCC13]
6	Jarratt et al.	Engineering change: an overview and perspective on the literature	2011	[JEC11]
7	Karthik, K.; Janardhan Reddy, K.	Engineering Changes in Product Design – A Review	2016	[KJ16]
8	Langer, S.; Wilberg, J.; Maier, A.; Lindemann, U.	Änderungsmanagement-Report 2012: Studienergebnisse zu Ursachen und Auswirkungen, aktuellen Praktiken, Herausforderungen und Strategien in Deutschland	2012	[LWM12]
9	Mordaschew, V.; Herrmann, J.-P.; Tackenberg, S.	METHODS OF CHANGE IMPACT ANALYSIS FOR PRODUCT DEVELOPMENT: A SYSTEMATIC REVIEW OF THE LITERATURE	2023	[MHT23]
10	Ullah, I.; Tang, D.; Yin, L.	Engineering product and process design changes: A literature overview	2016	[UTY16]
11	Wright, I. C.	A review of research into engineering change management: implications for product design	1997	[Wri97]
Fallstudien				
Nr	Autor / Institution	Titel	Jahr	Quelle
12	Deubzer, F.; Kreimeyer, M.; Lindemann, U.	Exploring strategies in change management – current status and activity benchmark	2006	[DKL06]
13	Eckert, C.; Clarkson, J. P.; Zanker, W.	Change and customisation in complex engineering domains	2004	[ECZ04]
14	Fricke, E.; Gebhard, B.; Negele, H.; Igenbergs, E.	Coping with Changes: Causes, Findings, and Strategies	2000	[FGN00]

15	Giffin	Change Propagation Analysis in Complex Technical Systems		[GWB09]
16	Hölttä, V.; Mahlamäki, K.; Eisto, T.; Ström, M.	Lean Information Management Model for Engineering Changes	2010	[HME10]
17	Huang, G. Q.; Mak, K. L.	Current practices of engineering change management in UK manufacturing industries	1999	[HM99]
18	Huang, G. Q.; Yee, W. Y.; Mak, K. L.	Current practice of engineering change management in Hong Kong manufacturing industries	2003	[HYM03]
19	Iakymenko et al.	Managing engineering changes in the engineer-to-order environment: challenges and research needs	2018	[IRS18]
20	Iakymenko et al.	Status of engineering change management in the engineer-to-order production environment: insights from a multiple case study	2020	[IRA20]
21	Morkos, B.; Shankar, P.; Summers, J. D.	Predicting requirement change propagation, using higher order design structure matrices: an industry case study	2012	[MSS12]
22	Pikosz, P.; Malmqvist	A comparative study of engineering change management in three Swedish engineering companies	1998	[PM98]
23	Subrahmanian et al.	Managing and supporting product life cycle through engineering change management for a complex product	2015	[SLG15]
24	Tavcar, J.; Duhovnik, J.	Engineering change management in individual and mass production	2005	[TD05]

A2.2 Liste an Ansätzen zur modellbasierten Auswirkungsanalyse

Literaturdatenbanken: Library der Design Society und Science Direct, Google Scholar zur Sicherstellung der Vollständigkeit [GW21, S. 6].

Google Scholar: Vervollständigung des Gesamtüberblicks, basierend auf verschiedenen Quellenarten (Grauliteratur, Standardwerke, Kapitelauszüge, Konferenz- und Journalbeiträge)

Science Direct & IEEE Explore, Design Society, Web of Science: Veröffentlichungen in Bezug auf den Forschungsrahmen, Konferenzen zum Stand der Forschung innerhalb des Forschungsfelds

Science Direct, Scopus, Web of Science: Veröffentlichung in Bezug

Scopus & Web of Science Core Collection: Durchführung einer eigenständigen Qualitätsüberprüfung

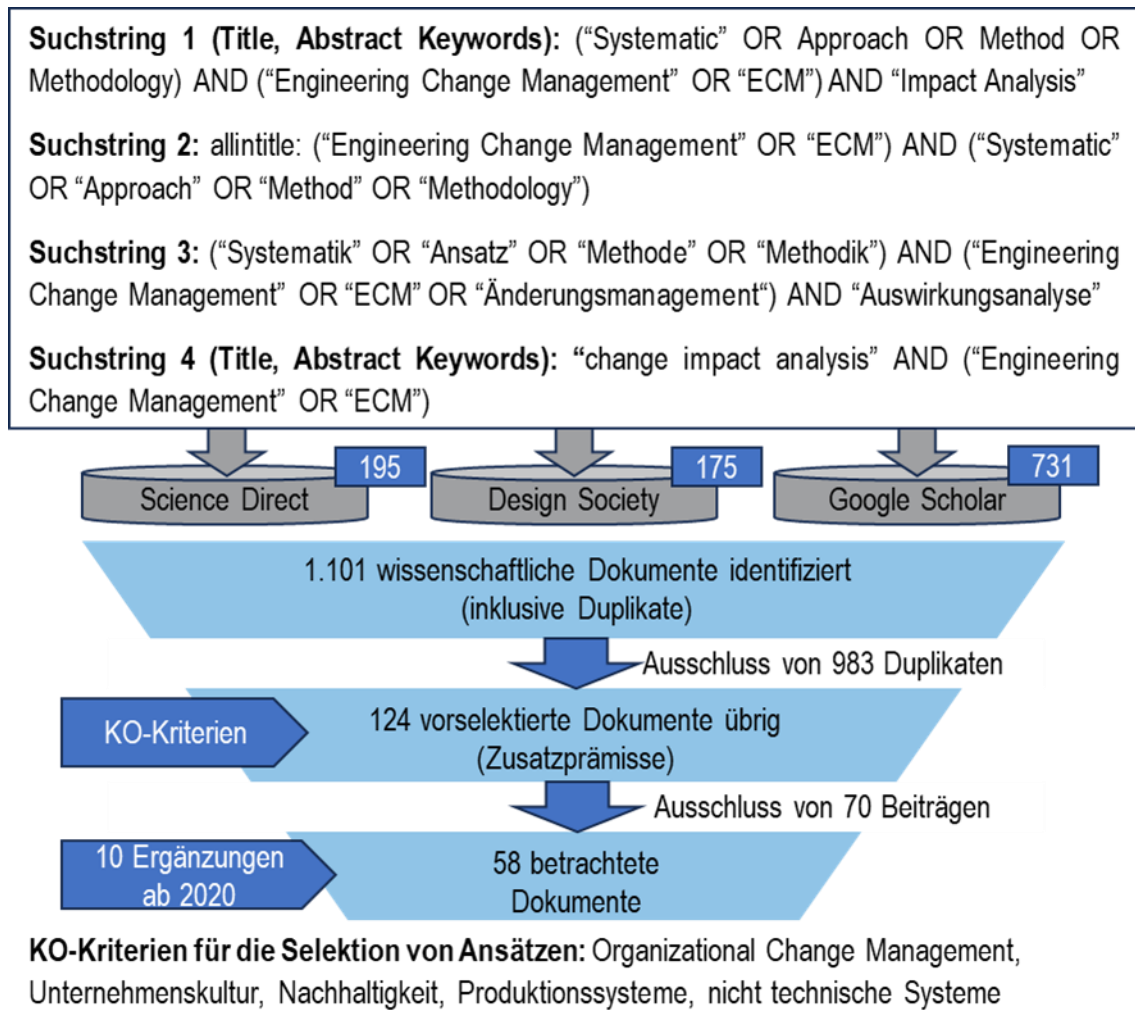


Bild A-2: Übersicht der Literaturrecherche zu modellbasierten Auswirkungsanalysen (schematische Darstellung in Anlehnung an WORTMANN ET AL. [WBC20, S. 72] und der studentischen Arbeit von KRAUß [KRA20])

Tabelle A-6: Übersicht der Ansätze zu modellbasierten Auswirkungsanalysen

Nr.	Autor	Titel	Jahr	Art	Z	Ø Z/J
1	Ahmad et al.	Development and evaluation of a tool to estimate the impact of design change	2010	M	16	1.23
2	Ariyo et al.	Predicting change propagation on different levels of granularity: An algorithmic view	2007	M	23	1.44
3	Basri et al.	A Change Impact Analysis Tool for Software Development Phase	2015	OO	7	0.88

4	Becerril L.; Sauer, M.; Lindemann, U.	Estimating the effects of Engineering Changes in early stages of product development	2016	SD	12	1.71
5	Briand et al.	Automated Impact analysis of UML Models	2006	OO	167	9.82
6	Chen H.; Chu, X.	A network-based assessment approach for change impacts on complex product	2012	G	125	11.36
7	Clarkson, P. J.; Simons C.; Eckert C.	Predicting change Propagation in complex design	2004	M	1008	53.05
8	Cohen, T.; Fulton R. E.	A data approach to tracking and evaluating engineering changes	1998	OO	24	0.96
9	Cohen, T.; Navathe S. B.; Fulton R. E.	C-FAR, change favourable representation	2000	OO	179	7.78
10	Conrad et al.	Change Impact and risk analysis (CIRA) - Combining the CPM/PDD Theory and FMEA-Methodology for an improved Engineering change Manangement	2007	G	52	3.25
11	Eger, T.; Eckert, C. M.; Clarkson, P. J.	Engineering change analysis during ongoing product development	2007	M	34	2.13
12	Fei, et al.	A method for engineering design change analysis using system modelling and knowledge management techniques	2011	G	72	6.00
13	Flanagan et al.	A functional analysis of change propagation	2003	N	55	2.75
14	Gräßler, I.; Olhoff, C.; Scholle, P.	Method for Systematic Assessment of Requirement Change Risk in Industrial Practice	2020	M	29	9.67
15	Hamraz, B.; Caldwell, N. H. M.; Clarkson, P. J.	FBS linkage model - Towards an integrated engineering change prediction and analysis method	2012	G	4	0.36
16	Hamraz, B.; Caldwell, N. H.	A matrix-calculation-based algorithm for numerical change propagation analysis	2013	M	95	9.50

	M.; Clarkson, P. J.					
17	Kattner et al.	A method for the expert-based identification of engineering change propagation	2017	M	4	0.67
18	Koh, E. C. Y.; Clarkson, P. J.	A modeling method to manage change propagation	2009	M	15	1.07
19	Köhler et al.	A matrix representation of the CPM/PDD approach as a means for change impact analysis	2008	M	58	3.87
20	Lee, J.; Hong, Y.S.	A Bayesian network approach to improve change propagation analysis	2015	G	2	0.25
21	Lemmens et al.	Methods to analyse the impact of changes in complex	2007	M	10	0.63
22	Li, J.; Zhang, D.; Li, S.	Engineering change management based on weighted complex networks	2008	G	8	0.53
23	Lough, K. G.; Stone, R. B.; Tumer, I.	Prescribing and implementing the risk in early design (RED) method.	2006	M	60	3.53
24	Ma, Y.; Chen, G.; Thimm, G.	Change propagation algorithm in a unified feature modeling scheme	2008	G	59	3.93
25	Ma et al.	Design Property Network-Based Change Propagation Prediction Approach for Mechanical Product Development	2017	G	22	3.67
26	Ma, S.; Jiang, Z. Liu, W.	A design change analysis model as a change impact analysis basis for semantic design change management	2017	G	21	3.50
27	Masmoudi et al.	Engineering Change Management: A novel approach for dependency identification and change propagation for product redesign	2017	G	12	2.00
28	Mehta, C.; Patil, L.; Dutta, D.	An Information-based Approach to compute similarity between engineering changes	2012	OO	5	0.45
29	Morkos, B.; Shankar, P.; Summers J. D.	Predicting requirement change propagation, using higher order design structure matrices: an industry case study	2012	M	147	13.36

30	Oduncuoglu, A.; Thomson, V.	Evaluating the risk of change propagation	2011	M	14	1.17
31	Ollinger, G. A.; Stahovich T. F.	ReDesignIT — a constraint-based	2001	G	57	2.59
32	Pasqual, M. C.; de Weck, O. L.	Multilayer Network Model for Analysis and Management of Change Propagation	2012	G	182	15.17
33	Raffaeli et al.	Development of a multilayer change propagation tool for modular products	2007	G	17	1.06
34	Rebentisch et al.	Assessment of changes in engineering design using change propagation cost analysis	2017	M	10	1.67
35	Reddi, K. R.; Moon, Y. B.	A framework for managing engineering change propagation	2009	OO	57	4.07
36	Wang, H.; Thomson, V.; Tang, C.	Change Propagation analysis for system modeling using Semantic Web technology	2018	OO	20	4.00
37	Wilms et al.	Using a cross-domain product model to support engineering change management	2020	OO	7	2.33
38	Yang, F.; Duan, G.	Developing a parameter linkage-based method for searching change propagation paths	2012	G	112	10.18
39	Yeasin, F. N.; Grenn, M.; Roberts, B.	A Bayesian Networks Approach to Estimate Engineering Change Propagation Risk and Duration	2020	G	22	7.33
40	Zhang, X.; Tan, Y.; Yang, Z.	Analysis of Impact of Requirements Change on Product Development Progress Based on System Dynamics	2020	G	4	1.33
Vervollständigung Google Scholar						
41	Clarkson, P. J.; Simons C.; Eickert C.	Change prediction for product redesign	2001	M	27	1.23
42	Hamraz, B.	Engineering change modelling using a function-behaviour-structure scheme (Dissertation)	2013	M	39	3.90
43	Köhler, C. M.	Technische Produktänderungen: Analyse und Beurteilung von Lösungsmöglichkeiten auf Basis einer	2009	M	31	2.21

		Erweiterung des CPM/PDD-Ansatzes				
44	Lemmens, Y. C. J.	Modelling and Analysing of Engineering Changes in Complex Systems (Dissertation)	2007	M	2	0.13
45	Oduguwa, P. A.	Cost Impact Analysis for Requirements Management (Dissertation)	2006	M	7	0.41
46	Ouertani, M. Z.; Yesilbas, L. G.; Lossent, L.	Engineering Change Process: State of the Art, A case Study and proposition of an impact analysis method	2007	OO	26	1.63
47	Reddi, K. R.	A Conceptual Framework and Simulation Modeling Of Engineering Change Management in a Collaborative Environment	2011	G	4	0.33
48	Wickel, M. C.	Änderungen besser managen (Dissertation)	2017	M	31	5.17
Ergänzungen seit 2020						
49	Ellsel, C.; Senkal, G.; Stark, R.	Using MBSE with SysML to Analyze Change Impacts, Including Efforts, in Technical Systems	2022	OO	0	0.00
50	Martin et al.	Advanced Engineering Change Impact Approach (AECIA)-Towards a model-based approach for a continuous Engineering Change Management	2022	OO	2	2.00
51	Pfeiffer et al.	Concept and implementation path for model-based impact analysis in product development	2023	OO	0	0.00
52	Gräßler, I., Wiechel, D.	Customized impact analyses for technical engineering changes	2023	G	0	0.00
53	Jagla et al.	Using SysML to Support Impact Analysis on Structural Dynamics Simulation Models	2021	OO	7	3.50
54	Gräßler, I.; Olhoff, C.; Preuß, D.	Proactive Management of Requirement Changes in the Development of Complex Technical Systems	2022	M	12	12.00
55	Li et al.	Design change propagation routing in the modular product	2022	G	2	2.00

56	Wang et al.	Product Change Propagation with Complex Network for Multiple Change Sources	2021	G	0	0.00
57	Shang et al.	Correlation Based Analysis of Parameter Change Propagation in Variant Product Design	2021	G	3	1.50
58	Gräßler et al.	Automated Requirement Dependency Analysis for Complex Technical Systems	2022	M	3	1,5

A2.3 Kriterien zur Auswahl der EC-Auswirkungsanalyse-Ansätze

Tabelle A-7: Auflistung der Kriterien zur Auswahl von Ansätzen

Kriterium	Gewichtung	1 Punkt	0,5 Punkte	0 Punkte
Relevanz (R)	40 %	> 10 Zitationen	> 2 Zitation	< 2 Zitationen
Aktualität (A)	30 %	< 5 Jahre	<15 Jahre	> 15 Jahre
Interdisziplinarität (I)	10 %	> 2 Disziplinen	2 Disziplinen	1 Disziplin
Praktische Anwendung (P)	10 %	Industrie	Forschung	Nein
Artefakte (Ar)	10 %	>3	2	1
Auswahl (W)				

Tabelle A-8: Bewertung der Ansätze anhand der Kriterien

Nr	Zitationen	Ø Zitationen	R	A	I	P	Ar	Summe	W
1	16	1.23	0	0.5	0	0.5	1	1.5	
2	23	1.44	0	0	0	1	0	0.5	
3	7	0.88	0	0.5	0	0.5	0	0.5	
4	12	1.71	0	0.5	0	0.5	1	1.5	
5	167	9.82	1	0	0	1	1	1.5	x
6	125	11.36	1	0.5	0	1	0.5	1.25	x
7	1008	53.05	1	0	0.5	1	0.5	1	x
8	24	0.96	0	0	0.5	1	0.5	1	
9	179	7.78	1	0	0.5	1	0.5	1	X
10	52	3.25	1	0	0	1	1	1.5	
11	34	2.13	0.5	0	0	1	0.5	1	
12	72	6	1	0.5	1	1	1	1.75	X
13	55	2.75	1	0	0	1	0	0.5	X
14	29	9.67	1	1	0	1	0	1	
15	4	0.36	0	0.5	0.5	1	1	1.75	
16	95	9.5	1	0.5	0	1	0	0.75	
17	4	0.67	0	0.5	0	0.5	1	1.5	
18	15	1.07	0	0.5	0	1	0	0.75	

19	58	3.87	0.5	0	0.5	0	0.5	0.5	
20	2	0.25	0	0.5	0	0	0.5	0.75	
21	10	0.63	0	0	0.5	0	0.5	0.5	
22	8	0.53	0	0	0	0	0	0	
23	60	3.53	0.5	0	0	0	0.5	0.5	
24	59	3.93	1	0	0	1	1	1.5	
25	22	3.67	1	0.5	0	1	1	1.75	
26	21	3.5	1	0.5	0	1	1	1.75	
27	12	2	1	0.5	0	0.5	0	0.5	
28	5	0.45	0	0.5	0	1	1	1.75	
29	147	13.36	1	0.5	0	0.5	0	0.5	X
30	14	1.17	0	0.5	0	0.5	0	0.5	
31	57	2.59	1	0	1	1	1	1.5	X
32	182	15.17	1	0.5	0.5	1	1	1.75	X
33	17	1.06	0	0	0.5	1	0.5	1	
34	10	1.67	0	0.5	0.5	0.5	1	1.5	
35	57	4.07	0.5	0.5	0	0.5	0	0.5	
36	20	4	0.5	0.5	0.5	0.5	1	1.5	
37	7	2.33	0.5	1	1	1	1	2	X
38	112	10.18	1	0.5	0	1	0.5	1.25	X
39	22	7.33	1	1	0	0.5	0	0.75	
40	4	1.33	0	1	0	0	0	0.5	
41	27	1.23	0	0	0	0.5	0	0.25	
42	39	3.9	0.5	0.5	0.5	1	1	1.75	
43	31	2.21	0.5	0.5	1	1	1	1.75	X
44	2	0.13	0	0	0.5	0	0.5	0.5	
45	7	0.41	0	0	0	0.5	0	0.25	
46	26	1.63	0	0	0	0	0.5	0.5	
47	4	0.33	0	0.5	0.5	0	0.5	0.75	
48	31	5.17	1	0.5	0.5	1	1	1.75	X
49	0	0	0	1	0	0.5	0.5	1.25	
50	2	2	0.5	1	1	0	1	1.5	X
51	0	0	0	1	1	0	1	1.5	
52	0	0	0	1	1	1	1	2	
53	7	3.5	0.5	1	0.5	1	0.5	1.5	X
54	12	12	1	1	0	1	0	1	
56	2	2	0.5	1	0	0.5	0.5	1.25	
57	0	0	0	1	0	0.5	0	0.75	
58	3	1.5	0	1	0	1	0.5	1.5	
59	3	1,5	0	1	0	1	0	1	

A2.4 Steckbrief EC-Auswirkungsanalyse Methoden

Steckbrief 1		
Autor: Clarkson, P. J.; Simons C.; Eckert C. [CSE04]	Jahr: 2004	Zitationen/Jahr: 53.05
Beschreibung: Clarkson et al. entwickeln einen Ansatz zur Bewertung des Änderungsrisikos einer technischen Änderung. Hierzu werden mathematische Modelle zur Berechnung der Änderungswahrscheinlichkeit und Änderungsauswirkung verwendet und auf Basis von Expertenwissen gefüllt. Im ersten Schritt der Methode werden Abhängigkeiten zwischen physischen Elementen eines Produktmodells auf Basis von Expertenwissen identifiziert. Anschließend werden Matrizen über die Wahrscheinlichkeit der Änderungspropagation sowie die vorhergesagten Auswirkungen auf Basis historischer Daten oder Expertenwissen festgelegt. Zur Quantifizierung werden separate Terme genutzt, die algorithmisch berechnet werden können. Durch die Multiplikation von Änderungswahrscheinlichkeit und Änderungsauswirkung ergibt sich das Risiko, dass eine Änderung in einem Subsystem zu einer weiteren Änderung in einem benachbarten Subsystem führt. [CSE04, S. 792]. Die Methode ist in der industriellen Praxis weit verbreitet [IRA20, S. 4509]. Die Autoren befassen sich mit der Vorhersage und dem Management von Änderungen. Als Beispiel wird die Neugestaltung eines Helikopters des Unternehmens GKN beschrieben. Die Autoren stellen heraus, dass das Zusammenspiel unterschiedlicher Rollen und das dazugehörigen Expertenwissens im Entwicklungsprozess entscheidend sind.		
Input (Art des Engineering Change)		
Änderung einer Komponente (z. B. durch eine neue Anforderung)		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
teil-automatisiert	Kollektive Betrachtung determinierter Effekte	Physische Elemente
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Physische Elemente (Komponenten)	matrizenbasiert, fördert, modelliert auf Basis von Expertenwissen	Binäre Abhängigkeiten in Komponenten-DSM
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
teil-automatisiert	hoch	Änderungsrisiko
Output (Art des Ergebnisses)		
Risk Scatter Plot, Product Risk Matrix		
Hilfsmittel		
Mathematische Modelle zur Berechnung der Änderungswahrscheinlichkeit und der Änderungsauswirkung		
Beispielhafte Anwendung		
GKN Westland Helikopter		
IKT-Unterstützung		
Software-Prototyp		

Steckbrief 2		
Autor: Köhler et al. [Köh09]	Jahr: 2009	Zitationen/Jahr: 3.87
Beschreibung: Im Characteristics-Properties Modeling / Property-Driven Development (CPM-PDD) Ansatz nach WEBER [Web05] werden Auswirkungen technischer Änderungen zwischen Charakteristika (Characteristics) und Merkmale (Properties) analysiert. Charakteristika und Eigenschaften werden miteinander verknüpft, um die Relationen zwischen diesen zu repräsentieren. Köhler et al. transformierten den CPM-PDD Ansatz anschließend in eine Matrixform, um die Relationen zwischen den Charakteristika und Eigenschaften zu repräsentieren [KCW08, S. 172]. Ausgehend vom ECR werden die betroffenen Merkmale innerhalb der Beziehungsmatrizen identifiziert. Der Ist-Zustand der Merkmale wird mit dem Soll-Zustand nach der Änderung verglichen. Sofern eine Änderung notwendig ist, werden die zu ändernden Merkmale angepasst. [KCW08, S. 172f.] Daher erhebt der Ansatz den Anspruch, sowohl die betroffenen Elemente des ECRs zu identifizieren als auch bei der ECO zu unterstützen. Die Autoren schreiben Ihren Ansatz als nutzerfreundlich, komplexitätsreduzierend und fehlervermeidend [KCW08, S. 173].		
Input (Art des Engineering Change)		
Betrachtete Änderung an einer Komponente		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
Teil-automatisiert	Kollektive Betrachtung determinierter Effekte	Physische Elemente
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Physische Elemente	Design Structure Matrix einer Produktstruktur	Numerische Werte in Matrix
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
Teil-automatisiert	hoch	Risiko von Anforderungsänderungen
Output (Art des Ergebnisses)		
Änderungsrisiko in Abhängigkeitsmatrizen		
Hilfsmittel		
Nutzung des CPM/PDD Ansatzes		
IKT-Unterstützung		
Tabellenkalkulationssoftware		

Steckbrief 3		
Autor: Wickel [Wic17]	Jahr: 2017	Zitationen/Jahr: 5.17
Beschreibung: Das sechs-schrittige Vorgehen zur Erstellung eines Änderungsstrukturmodells nach Wickel besteht aus folgenden Schritten: Ziel- und Systemdefinition, Datenakquisition und -selektion, Datenvorbereitung und -bereinigung, Datentransformation, Data Mining und Evaluation. Die Assoziationen und zugehörigen Regeln werden aus historischen Änderungsdaten extrahiert [Wic17, S. 81]. Die Maßzahlen Support, Konfidenz und Lift werden als Assoziationsregeln berechnet, um Änderungsauswirkungen entlang identifizierter Zusammenhänge zu quantifizieren [Wic17, S. 50f.]. Die posteriori Analyse wird retrospektiv auf der Änderungsstruktur durchgeführt und unterstützt Entwickler beim Identifizieren von Verbesserungspotenzialen für neue Produktgenerationen [Wic17, S. 104]. Die priori Analyse unterstützt Entwickler bei der Bestimmung der Änderungsausbreitungen und -auswirkungen, indem betroffene Komponenten entlang von Assoziationen identifiziert werden.		
Input (Art des Engineering Change)		
Änderung an Komponente		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
Hoch	Kollektive Betrachtung determinierter Effekte	a-priori und posteriori
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Produktstruktur (Komponenten)	Matrixmodell (semi-automatisiert erstellt durch Data-Mining)	Änderungsbedingte Relation
Abschätzung der Folgen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
teil-automatisiert	Mittel	Vorhersage der Änderungsausbreitung
Output (Art des Ergebnisses)		
Änderungsnetzwerk, DSM mit Inter- und Intra-Relationen zwischen Modulen		
Hilfsmittel		
Assoziationsregeln		
Beispielhafte Anwendung		
Fahrzeugtypen einer Automobilbaureihe		
IKT-Unterstützung		
Entwicklung eines IT-Tools in RapidMiner		

Steckbrief 4		
Autor: Gräßler et al. 2020 [GOS20]	Jahr: 2020	Zitationen/Jahr: 9.67
Beschreibung: Gräßler et al. entwickelten eine Methode zur systematischen Bewertung des Risikos von Anforderungsänderungen. Zur Bewertung des Risikos wird die Änderungswahrscheinlichkeit und die Änderungsauswirkung bestimmt. Die Änderungswahrscheinlichkeit unterteilt sich in die exogene und endogene Änderungswahrscheinlichkeit. Die Änderungsauswirkungen umfassen lokale Auswirkungen am initial betroffenen Element und konsekutiv folgende Auswirkungen entlang des entstehenden Propagationspfads. Mithilfe von Prioritätswerten und Abhängigkeitswerten wird das Risiko quantifiziert und in einem Risiko-Portfolio dargestellt.		
Input (Art des Engineering Change)		
Anforderungsänderung		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
hoch	Kollektive Betrachtung determinierter Effekte	Anforderungen
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Anforderungen	Matrizenbasiert, föderiert	Unidirektional Abhängigkeit zwischen Anforderungen
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
Hoch	Hoch	Änderungsrisiko
Output (Art des Ergebnisses)		
Änderungsrisiko, Risiko-Portfolio		
Hilfsmittel		
Prioritätswerte und Abhängigkeitswerte		
Beispielhafte Anwendung		
Heckflügelhalter und weitere		
IKT-Unterstützung		
eigenes IKT-Werkzeug		

Steckbrief 5		
Autor: Morkos et al. [MSS12]	Jahr: 2012	Zitationen/Jahr: 13.36
Beschreibung: Morkos et al. untersuchen, inwiefern sich Design-Struktur-Matrizen höherer Ordnung zur Bewertung von Anforderungsänderungen in Entwicklungsprojekten eignen. Die Ergebnisse der Studie verdeutlichen, dass DSM verwendet werden können, um Änderungsauswirkungen zwischen Anforderungen abzuschätzen. Bei der Modellierung haben gerade Relationen zweiter Ordnung eine hohe Relevanz, damit Entwickler die Änderungsauswirkungen adäquat abschätzen können. Zur Differenzierung können unterschiedliche Relationsarten verwendet werden oder Relationen gewichtet werden.		
Input (Art des Engineering Change)		
Anforderungsänderung		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
Teil-automatisiert	Kollektive Betrachtung determinierter Effekte	Anforderungen
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Anforderungen	Matrizenbasiert, föderiert	Bidirektionale Abhängigkeiten
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
Teil-automatisiert	Hoch	Betroffene Anforderungen
Output (Art des Ergebnisses)		
Betroffene Anforderungen		
Hilfsmittel		
DSMs höherer Ordnung		
Beispielhafte Anwendung		
Katalysator		
IKT-Unterstützung		
Tools zur Abbildung von DSM höherer Ordnung		

Steckbrief 6		
Autor: Pasqual und de Weck [PW12]	Jahr: 2012	Zitationen/Jahr: 15.17
Beschreibung: Die Ingenieure sind hierbei auf sozialer Ebene, befassen sich mit einer technischen Änderung auf der Änderungsprozessebene, welche Komponenten auf der Produktebene betrifft [PW12, S. 306]. Innerhalb einer Ebene werden Relationen zwischen den Elementen einer Ebene abgebildet, die sogenannten Intra-Layer-Relationen. Beispiele sind Flussarten zwischen den Komponenten der Produktebene, Relationen zwischen technischen Änderungen auf Änderungsebene abgebildet oder Relationen zwischen Personen oder Gruppen eines Unternehmens auf sozialer Ebene. [PW12, S. 306f.].		
Input (Art des Engineering Change)		
Änderung an Komponente		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
teil-automatisiert	Kollektive Betrachtung determinierter Effekte	Relationen auf soziale Ebene und Komponentenebene
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Produkt, ECR, Sozial	Mehrschichtiges Netzwerkmodell	Intra-Layer-Relationen, Inter-Layer-Relationen
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
automatisiert	Hoch	Propagationsindex
Output (Art des Ergebnisses)		
Propagationsbaum oder Matrix		
Hilfsmittel		
Metriken (Engineering Change Propagation Index und Propagation Directness)		
Beispielhafte Anwendung		
Anwendung anhand existierender Forschungsdaten		
IKT-Unterstützung		
Engineering-PDSM oder existierende Tools		

Steckbrief 7		
Autor: Yang, F.; Duan, G. [YD12]	Jahr: 2012	Zitationen/Jahr: 10.18
Beschreibung: Im Parameter Linkage Network Modell werden zwei Arten der Parameterverknüpfung unterschieden: die fundamentale Parameterverknüpfung aufgrund physikalischer Gesetze und die subjektiven Parameterverknüpfungen, die aus Sicht der Entwickler notwendig sind. Die Verknüpfungen haben ein unterschiedliches Propagationsverhalten, wodurch eine Analyse und Bewertung der Auswirkungen möglich ist [YD12, 353 f.]. Zusätzlich werden die Parameter klassifiziert. Zielparameter werden durch Transitionsparameter mit den direkten Parametern verknüpft. [YD12, S. 358f.]. Abhängig von der Art des initial betroffenen Parameters werden mögliche Propagationspfade innerhalb des Modells identifiziert und durch festgelegte Attribute bewertet.		
Input (Art des Engineering Change)		
Änderung an Komponente		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
gering	Kollektive Betrachtung determinierter Effekte	Komponenten und zugehörige Parameter
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Zielparameter, direkte Parameter	Netzwerkmodell, fördert	Parameterverknüpfungen
Abschätzung der Folgen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
Gering	hoch	Routing- und Diffusionspfade
Output (Art des Ergebnisses)		
Graphische Visualisierung und Metriken		
Hilfsmittel		
Methode zur Identifikation von Propagationspfaden, Prioritätsstrategien, Metriken (Number of change Parameters, Negative Change Impact, Expense of Change...)		
Beispielhafte Anwendung		
Membrankupplung		
IKT-Unterstützung		
Nicht-kommerzialisiertes IKT-Werkzeug		

Steckbrief 8		
Autor: Chen, H.; Chu, X. [CC12]	Jahr: 2012	Zitationen/Jahr: 11.36
Beschreibung: Aufbauend auf Ansätzen der Netzwerktheorie entwickelten Chen und Chu einen Ansatz zur Berechnung der Veränderlichkeit eines Systems innerhalb von Netzwerken mit gewichteten Relationen. Zur Berechnung werden zwei Netzwerke modelliert: eins für den Informationsfluss bei Entwicklungsaufgaben zur Erstellung von Komponenten und eins für räumliche Abhängigkeiten zwischen den Komponenten. Mithilfe von drei Indizes wird die Veränderlichkeit berechnet. Die Indizes unterstützen Entwickler, die Auswirkungen der technischen Änderung anhand von quantitativen Werten abzuschätzen.		
Input (Art des Engineering Change)		
Änderung an Komponente		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
Teil-automatisiert	Kollektive Betrachtung determinierter Effekte	Komponenten
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Komponenten und Entwicklungsaufgaben	Graphenbasiert (gewichtetes Netzwerk), föderiert, explizit modelliert auf Basis von Expertenwissen	Gewichtete unidirektionale Relationen für Informationsflüsse zwischen Entwicklern und physischen Abhängigkeiten
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
Teil-automatisiert	Hoch	Berechnung der Veränderbarkeit
Output (Art des Ergebnisses)		
Berechnete Indizes sowie graphische Repräsentationen in Koordinatensystemen (z.B. Average Reachability vs. Average Distance)		
Hilfsmittel		
Veränderbarkeit Indizes: Grad-Änderbarkeit für direkte Auswirkungen, Reichweiten-Änderbarkeit für indirekte Auswirkungen und Zwischen-Änderbarkeit für gegenseitige Auswirkungen		
Beispielhafte Anwendung		
Drehkolbengebläse		
IKT-Unterstützung		
-		

Steckbrief 9		
Autor: Fei, et al. [FGO11]	Jahr: 2011	Zitationen/Jahr: 6.00
Beschreibung: Fei et al. definieren eine systematische drei-schrittige Methode zur Identifikation von Propagationspfaden und zur Auflösung von Konflikten bei der Implementierung einer technischen Änderung. Im ersten Schritt werden im Systemmodell systematisch Propagationspfade durch drei Relationstypen zwischen funktionale Anforderungen und physische Komponenten in SysML und CAD abgebildet. Die funktionalen Anforderungen des Systems werden in SysML Blockdefinitionsdiagrammen modelliert, die physische Architektur durch CAD-Modelle abgebildet. Im zweiten Schritt werden in einer mehrdimensionalen Matrix, der sogenannten Composite Matrix, die Relationen repräsentiert, wodurch analysierbare Propagationspfade entstehen. Im letzten Schritt werden potenzielle Konflikte bei der Umsetzung einer technischen Änderung auf Basis des Wissens vergangener Änderungen gelöst. Als Wissensbasis werden konfliktäre Entwicklungsentscheidungen in einer vordefinierten Ontologie formalisiert. Die Anwendbarkeit des Ansatzes wird am Fallbeispiel einer Windturbine repräsentiert.		
Input (Art des Engineering Change)		
Änderungsanfrage zur funktionalen Anforderung		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
Nicht-automatisiert	Kollektive Betrachtung determinierter Effekte	Funktionale Anforderungen und Komponenten
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Funktionale Anforderungen und Komponenten	Integriert aus funktionaler Sicht und Sicht auf physischen Interaktionen (SysML) und räumlicher Sicht (CAD Modell)	Relationen zwischen funktionalen Anforderungen und zwischen physischen Komponenten
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
Teil-automatisiert	hoch	Anforderungsverifikation und Lösungskonflikte
Output (Art des Ergebnisses)		
Identifizierte und entstehende Lösungskonflikte		
Hilfsmittel		
SysML Modelle, TRIZ-Methode		
Beispielhafte Anwendung		
Windturbine		
IKT-Unterstützung		
Prototypische Implementierung mit Protégé		

Steckbrief 10		
Autor: Ollinger und Stahovich [OS01]	Jahr: 2001	Zitationen/Jahr: 6.00
Beschreibung: Die Autoren entwickeln einen Ansatz, um Entwickler bei der Durchführung von Weiterentwicklungen zu unterstützen. Ausgehend von der geplanten Neugestaltung wird ein graphenbasiertes Systemmodell verwendet, um die Effekte abzuschätzen. Durch die Nutzung von Beschränkungen können Abhängigkeiten spezifiziert werden, beispielsweise werden fixierende, maximierende und minimierende Zusammenhänge abbildbar. Durch die Abbildung der Auswirkungen in dem graphenbasierten Propagationsbaum werden Entwickler bei der Entscheidung zur Umsetzung der Neugestaltung unterstützt.		
Input (Art des Engineering Change)		
Änderung an Komponente		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
Mittel	Kollektive Betrachtung determinierter Effekte	Komponenten bis auf Parameterebene
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Parameter und Komponenten	Graphbasiert, föderiert	Unidirektionale Relationen in Abhängigkeitsgraph
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
Teil-automatisiert	Mittel	Propagationseffekte
Output (Art des Ergebnisses)		
Pläne zur Neugestaltung		
Hilfsmittel		
Propagationsbaum, Propagationsregeln, existierende Algorithmen, Ansätze des model-based Reasonings		
Beispielhafte Anwendung		
Dieselmotor		
IKT-Unterstützung		
Eigenes IKT-Werkzeug (RedesignIT)		

Steckbrief 11		
Autor: Wilms et al. [WKI20]	Jahr: 2020	Zitationen/Jahr: 2.33
Beschreibung: Wilms et al. [WKI20] entwickeln ein Konzept für ein domänenübergreifendes Produktmodell für das Engineering Change Management. Das Konzept umfasst die Zusammenführung der funktionalen Struktur der Produktstruktur und der Produktportfoliostruktur [WKI20, S. 1168f.]. Zusätzlich werden drei explizite Sichten auf das Produkt festgelegt, die Entwicklern bei der Nutzung des Systemmodells für das ECM unterstützen sollen: die funktionale Sicht, die strukturelle Sicht und die Auswirkungsanalysesicht [WKI20, S. 1169]. Die Auswirkungsanalysesicht umfasst Ausschnitte über die Änderungsvorhersage, den Propagationspfad und abstrakte Sichten auf die einzelnen Ebenen [WKI20, S. 1169]. Neben den Sichten werden ebenfalls die benötigten Artefakte und Relationen festgelegt. Funktionen werden und kategorisiert. Komponenten können ebenfalls hierarchisch untergliedert werden und miteinander durch technische Relationen (Materialfluss, Informationsfluss, Signalfluss) verknüpft werden. Das Konzept wird für das Fallbeispiel eines Kraftfahrzeugs angewendet. Innerhalb von den vier Prozessphasen Herleiten, Allokieren, Verknüpfen und Analysieren entstehen die einzelnen Sichten auf das Produktmodell. Durch die Nutzung weiterer Attribute wie Änderungskritikalität und verantwortlicher Person wird die Detaillierungstiefe der Funktionen und Komponenten erhöht [WKI20, S. 1171f.].		
Input (Art des Engineering Change)		
Änderung an Komponenten		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
Gering	Kollektive Betrachtung determinierter Effekte	Funktionen und Struktur bis auf Parameterebene
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Funktionen und Struktur	SysML-Systemmodell, fördert	Hierarchische Dekomposition, technische Relationen
Abschätzung der Folgen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
Gering	Hoch	Änderungsvorhersage & Propagationspfade
Output (Art des Ergebnisses)		
Vorgefertigte Sichten auf das SysML-Systemmodell		
Hilfsmittel		
Vorgefertigte Sichten		
Beispielhafte Anwendung		
Kraftfahrzeug		
IKT-Unterstützung		
Nutzung der Funktionalitäten von existierenden SysML-Werkzeugen		

Steckbrief 12		
Autor: Martin et al. [MKP22]	Jahr: 2022	Zitationen/Jahr: 2.00
Beschreibung: Martin et al. entwickeln den Advanced Engineering Change Impact Approach (AICEA). Durch den Ansatz können Propagationseffekte holistisch betrachtet und Auswirkungen der Änderungen kommuniziert werden. Der Ansatz besteht aus neun Aktivitäten, beispielsweise der Erstellung des Change Requests, der Analyse betroffener Modellelemente in SysML und der kriterien-basierten Bewertung von Lösungsalternativen. Grundlage des Ansatzes ist die IPEK-MBSE Methodik.		
Input (Art des Engineering Change)		
Rapid Engineering Change Request (z.B. Komponente)		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
(teil)-automatisiert	Kollektive Betrachtung determinierter Effekte	Modellelemente in SysML mit Profilerweiterung
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Anforderungen, Funktionen, Logische Elemente, Lösungsalternativen, Rollen	Objektorientiert, fördert	Unidirektionale Relationen (Allocate, derive, concern, change, replace, satisfy, ...)
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
(teil)-automatisiert	Hoch	Änderungsauswirkungen auf betroffene Elemente
Output (Art des Ergebnisses)		
Bewertete Lösungsalternativen		
Hilfsmittel		
IPEK-MBSE-Methodik		
Beispielhafte Anwendung		
Beispiel aus Automobilindustrie		
IKT-Unterstützung		
Implementierung in MBSE Tool geplant		

Steckbrief 13		
Autor: Jagla et al. [JJS21]	Jahr: 2021	Zitationen/Jahr: 3.50
Beschreibung: Jagla et al. entwickeln eine Methode zur schnellen Ermittlung des Änderungsaufwands, basierend auf Abhängigkeiten zwischen Systemkomponenten in einem SysML-Systemmodell. Bei der Modellierung werden die Abhängigkeiten zwischen den Systemkomponenten in SysML abgebildet und durch Relationen zu externen Partialmodelle ergänzt. Beispielhaft angebundene Partialmodelle sind Simulationsmodelle. Die Bewertung des Änderungsaufwands wird mithilfe von Metachain-Navigationen durchgeführt. Die Nutzung der Metachain-Funktionalität ermöglicht die Analyse der betroffenen Modellelemente, die in SysML abgebildet sind.		
Input (Art des Engineering Change)		
Änderung an einem SysML-Modellelement		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
Semi-automatisiert	Kollektive Betrachtung in Abhängigkeit der Metachain Query	Komponenten und spezifische Modelle bis auf Parameter-ebene
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Produktstruktur und Konstruktionsstruktur	Objektorientiert, föderiert	SysML-Relatione (z.B. Generalisierungen) und stereotypisierte Relationen
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
Automatisiert	Hoch	Änderungsaufwand
Output (Art des Ergebnisses)		
Änderungsaufwand und Ergebnis der Metachain		
Hilfsmittel		
Einbindung von weiteren Partialmodellen (z.B: Finite-Elemente Modelle und Mehrkörpersimulation)		
Beispielhafte Anwendung		
Aufgabe der strukturellen dynamischen Analyse		
IKT-Unterstützung		
Nutzung der Funktionalitäten von existierenden SysML-Werkzeugen		

Steckbrief 14		
Autor: Briand et al. 2006 [BLO06]	Jahr: 2006	Zitationen/Jahr: 9.82
Beschreibung: Briand et al. haben einen UML-basierten Ansatz entwickelt, um die Änderungswahrscheinlichkeit von Modellelementen innerhalb eines UML-Systemmodells abzuschätzen. Ausgehend von den modellierten Modellelementen werden Regeln mithilfe der Object Constraint Language formuliert, die das Propagationsverhalten analysieren. Mithilfe der Methode können technische Änderungen vor der Implementierung der Änderung bewertet werden. Die Autoren entwickeln ein prototypisches IKT-Werkzeug, welches bei der objektorientierten Auswirkungsanalyse unterstützt.		
Input (Art des Engineering Change)		
Änderung an UML Element		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
Teil-automatisiert	Abhängig von den definierten Regeln der Object Constraint Language	Modellierte Elemente in SysML
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
SysML-Standardprofil	Objektorientiert, fördert in SysML	SysML-Abhängigkeitstypen
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
Teil-automatisiert	Mittel	Änderungswahrscheinlichkeit
Output (Art des Ergebnisses)		
Änderungswahrscheinlichkeit für betroffene Elemente in UML		
Hilfsmittel		
Object Constraint Language		
Beispielhafte Anwendung		
-		
IKT-Unterstützung		
Eigene Softwarelösung (iACMTool)		

Steckbrief 15		
Autor: Cohen, T.; Navathe S. B.; Fulton R. E. [CNF00]		Jahr: 2000
		Zitationen/Jahr: 7.78
Beschreibung: C-FAR (Change Favorable Representation) ist eine Methodik in der existierende Produktdaten analysiert werden, um Auswirkungen auf das Produkt zu berechnen. Die Anwendung umfasst die Repräsentation, Propagation und qualitative Bewertung. Aufbauend auf dem EXPRESS Schema werden existierende Produktdaten genutzt, um Propagationspfade zu identifizieren und die Abhängigkeiten zwischen den Attributen zu analysieren.		
Input (Art des Engineering Change)		
Initial betroffene Entität		
Identifikation von Effekten		
Automatisierungsgrad	Tiefe der Analyse	Umfang der Analyse
nicht automatisiert	Kollektive Betrachtung determinierter Funktionalitäten	Produktdaten bis auf Parameterebene
Metadatenmodells		
Abgebildete Modellelemente	Art des (Meta)Datenmodells:	Abhängigkeitstypen:
Frei wählbare Entitäten	Objektorientiert (basierend auf EXPRESS), fördert, basierend auf existierenden Daten und Informationen	Gewichete Relationen (high linkage, medium linkage, low linkage)
Abschätzung der Auswirkungen		
Automatisierungsgrad	Nachvollziehbarkeit	Bezugspunkt
nicht automatisiert	Hoch	Attributwerte
Output (Art des Ergebnisses)		
Von Änderung beeinflusste Attribute		
Hilfsmittel		
EXPRESS Datenmodell		
Beispielhafte Anwendung		
Strukturanalyse bei Stoßfänger und Spritzgießen und thermische Analyse von Leiterplatten		
IKT-Unterstützung		
-		

A2.5 Gegenüberstellung der Ansätze anhand der Elemente

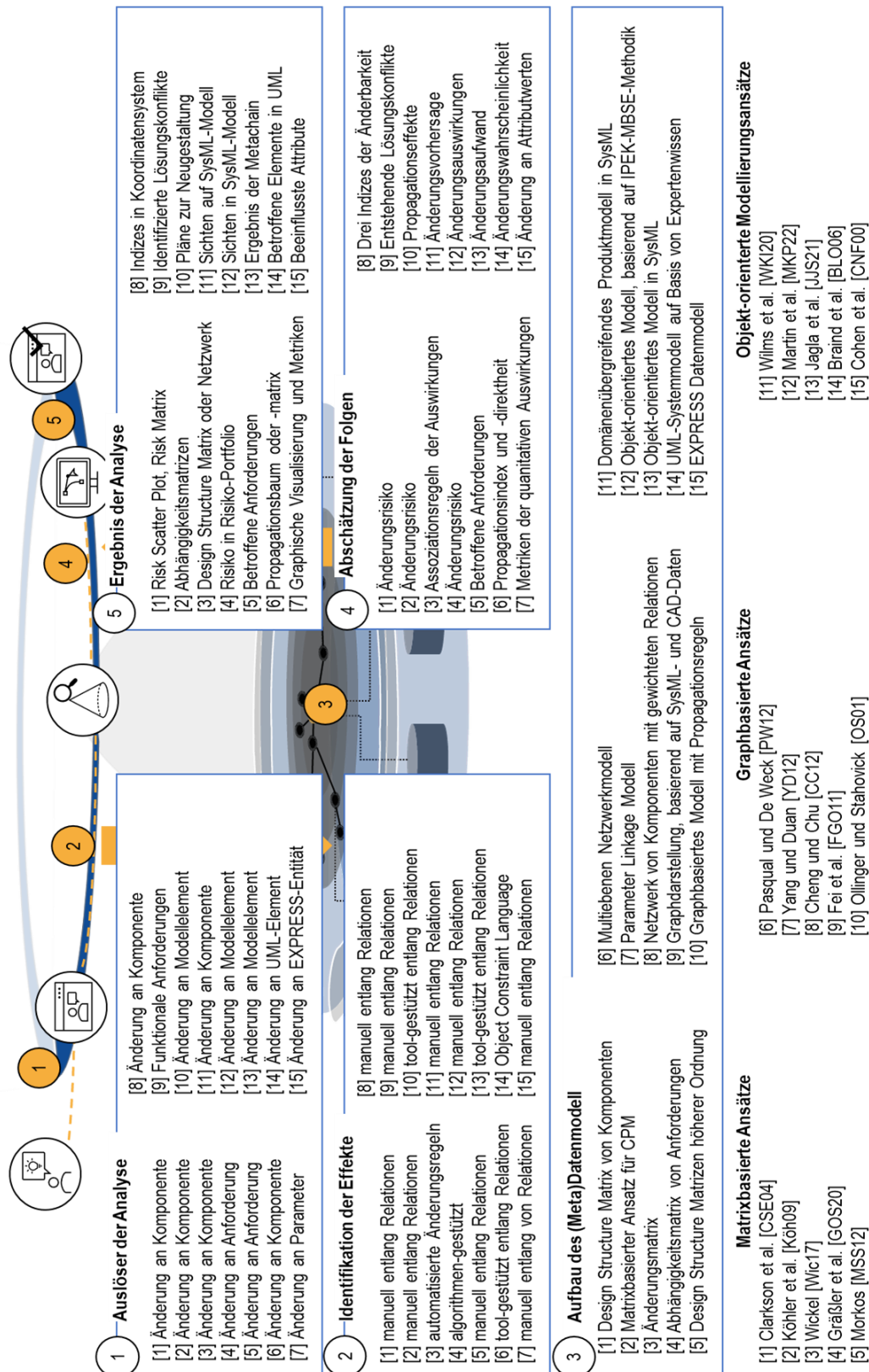


Bild A-3: Einordnung exemplarischer Ansätze anhand der identifizierten Aspekte

A2.6 Liste an Ansätzen zu Modellierungsmethoden

KO-Kriterien: Keine textuelle Dokumentation, Fokus auf SE, nicht auffindbare Publikationen, unzureichender Umfang der Beschreibung, geringe jährliche Zitationszahlen, fehlende Beschreibung anhand von Fallbeispielen

KO-Kriterien: Zugänglichkeit, eigener Ansatz, Case Study

Tabelle A-9: Übersicht identifizierter Ansätze zu Modellierungsmethoden

Titel	Quelle
IBM Harmony SE (Harmony SE)	[Kru00]
Object-Oriented Systems Engineering Method (OOSEM)	[Fri14]
Object-Process-Methodology (OPM)	[Dor02]
System Modeling Process (SysMOD)	[Wei07]
Pattern-based Systems Engineering (PBSE)	[Sch15, Sch23]
D'assault Systèmes MagicGrid (MagicGrid)	[AM21]
Architecture Analysis & Design Integrated Approach (ARCADIA)	[Voi23]
IBM Unified Process for Systems Engineering (RUP SE)	[Hof13]
Vitech Model-Based Systems Engineering (Vitech)	[BL00]
JPL State Analysis (JPL SA)	[KWS07]
SPES 2020 Methodik (SPES)	[PHA12]
VPE System Development Methodology (SE-VPE)	[Eig21]
Wirkkettenmodellierung nach Alt	[Alt12]

Tabelle A-10: Zuordnung der identifizierten Titel zu den Literaturstudien

	Mhenni [MCP14]	Estefan, Weilkiens [EW23]	Guo_Zhao_QI [GZQ22]	Estefan [Est08]	Weilkiens et al. [WSD16]	Dickopf et al. [DSS17]
Harmony SE	x		x	x		x
OOSEM	x	x		x		x
OPM		x		x		
SYSMOD		x				x
PBSE						
MagicGrid			x			
ARCADIA			x			
RUP SE				x		
Vitech				x		
JPL SA						
MARTE						x
SPES						x
CONSENS						x
SE-VPE						x
Alt						x

A2.7 Steckbriefe MBSE-Modellierungsmethoden

Steckbrief 1	
Autor: Tim Weikiens [Wei07]	Jahr: 2007
<p>Beschreibung: Die Modellierungsmethode besteht aus zwei aufeinander aufbauenden Phasen: der Analysephase und der Realisierungsphase. Die Analysephase besteht aus sechs Schritten, die Realisierungsphase aus vier Schritten. Insgesamt werden 15 weitere Zwischenschritte durchgeführt. Zu Beginn der Analysephase wird der Projektkontext beschrieben, indem relevante Stakeholder identifiziert und erste Anforderungen gesammelt werden. Bei der anschließenden Ermittlung der Anforderungen werden weitere Anforderungen erhoben und modellbasiert in einem Anforderungsdiagramm abgebildet. Hierbei werden unterschiedliche Kategorien von Anforderungen unterschieden, beispielsweise essenzielle Anforderungen und technische Anforderungen. Anschließend wird der Systemkontext abgebildet, indem partizipierende Akteure identifiziert und Schnittstellen definiert werden. Dies bildet die Ausgangslage, um Use Cases des Systems zu definieren. Ausgehend von den Use Cases können Anforderungsanalysen durchgeführt werden [Wei07, S. 63]. Zusätzlich werden in der Analysephase die Erstellung eines Glossars und die Modellierung von domänenspezifischem Wissen durchgeführt, um die eindeutige Verwendung von Begriffen und die Berücksichtigung von Expertenwissen zu gewährleisten. Die Use Cases bilden die Ausgangslage für die zweite Phase der Realisierung. Hierzu werden Interaktionen mit Systemen und Akteuren weiter spezifiziert, Schnittstellen festgelegt, die Modellstruktur abgebildet und die Systemzustände spezifiziert.</p>	
Hilfsmittel	
Profilerweiterungen für SysMOD, FAS4M und VAMOS	
Beispielhafte Anwendung	
Buchungssystem eines Mietwagens	
IKT-Unterstützung	
SysML-Modellierungswerkzeuge	

Steckbrief 2	
Autor: Sanford Friedenthal [Fri14]	Jahr: 2014 (Erstveröffentlichung 2008)
<p>Beschreibung: In der Methodik OOSEM werden sieben Modellierungsschritte zum modellbasierten Spezifizieren und Entwerfen des Systems beschrieben. Ausgehend von dem Schritt zur Aufstellung der Modellstruktur werden die Stakeholderbedürfnisse analysiert und durch Systemanforderungen spezifiziert. Innerhalb dieser ersten drei Schritte werden unter anderem Konventionen zur Modellierung festgelegt und der Zweck der Modellierung als Mission festgehalten. Basierend auf den Systemanforderungen wird die logische Architektur erstellt und physische Architekturkandidaten hergeleitet. Parallel dazu wird die Rückverfolgbarkeit zu den Anforderungen sowie die Optimierung und Bewertung von Alternativen durchgeführt. [Fri14, S. 410f.] Jeder der Schritte wird durch Aktivitäten detailliert. Die Methode wird geteilt in eine simplifizierte Methode für Beginner und eine umfangreichere ganzheitliche Methode für Experten. Die durch OOSEM modellierten Modellartefakte stellen verschiedene Sichten auf das System dar, unter anderem Sichten auf das Verhalten, die Struktur, die Eigenschaften und die Rückverfolgbarkeit zu den Anforderungen [Fri14, S. 405]. Als Hilfsmittel definiert Friedenthal Kontrollfragen, welche wesentliche Modellierungsprinzipien adressieren und dessen Beantwortung bei der Bewertung der Modellqualität unterstützen. Eine beispielhafte Frage ist: Reicht der Umfang des Modells aus, um den beabsichtigten Zweck zu erfüllen? [Fri14, S. 23f.] Die Kontrollfragen implizieren Qualitätskriterien, ohne diese explizit prüfbar zu gestalten. Ebenfalls führt Friedenthal SysML-Lite ein, indem die 9 SysML-Diagramme auf 6 essenzielle Diagramme reduziert werden. Je Diagramm wird eine reduzierte Anzahl von SysML Modellelementen bereitgestellt, wodurch die Modellierung für Anfänger vereinfacht wird [Fri14, S. 32f.].</p>	
Hilfsmittel	
Beginnerversion, Kontrollfragen	
Beispielhafte Anwendung	
Anlage der Wohnungssicherheit / Alarmanlage	
IKT-Unterstützung	
SysML-Modellierungswerkzeuge	

Steckbrief 3	
Autor: Eigner [Eig21]	Jahr: 2021
Beschreibung: Das MVPE-Modell beschreibt den Prozess über die oberste Ebene der Systemlebenszyklusphasen und orientiert sich an einem modifizierten V-Modell der VDI 2206:2004. Auf der linken Seite des V-Modells wurde RFLP hin zu KFLP (Kontext, Funktionen, Logische Elemente, Physische Elemente) transformiert. Das KSKM besteht aus vier Räumen: dem Anforderungsraum, dem Lösungsraum, dem Verifikations- und Validierungsraum und dem Administrationsraum. Innerhalb des Anforderungsraums werden Anforderungen im Kontext betrachtet. Im Lösungsraum werden Funktionen, Logische Elemente und Physische Elemente abgeleitet. Hierbei erfolgten eine stetige Verifikation und Validierung sowie eine administrative Überwachung [Eig21, S. 84].	
Hilfsmittel	
Profiliererweiterungen in SysML	
Beispielhafte Anwendung	
Autonome Baustelle	
IKT-Unterstützung	
PLM-Systeme und SysML-Modellierungswerkzeuge	

Steckbrief 4	
Autor: Alt [Alt12]	Jahr: 2012
Beschreibung: ALT beschreibt einen SysML-basierten Ansatz zur detaillierten Modellierung von Ursache-Wirkungs-Beziehungen zwischen Software und Hardware-Komponenten. Als Hilfsmittel für die Anwender werden Modellierungsregeln und Namenskonventionen spezifiziert. Durch die separate ergänzende Sicht auf die Wirkzusammenhänge wird die technisch-physikalische Architektur vervollständigt und Informationsbedarfe bezüglich der Zusammenhänge adressiert. Das entstehende Systemmodell bietet das Potenzial, detaillierte Aussagen über Wirkzusammenhänge abzuleiten. Die Nutzung des Modells für EC-Auswirkungsanalysen wird vom Autor nicht betrachtet. Ebenfalls werden keine Hilfsmittel zur Bewertung der Wirkzusammenhänge bereitzustellen, um die Systemanalysen umzusetzen. In seiner Methode werden die Einzelteile seiner Wirkkettenarchitektur näher beschrieben, beispielweise Signal- und Aktuatorketten, die Eingangs und Ausgangssignale verarbeiten können [Alt12, S. 98].	
Hilfsmittel	
SysML-Profiliererweiterung	
Beispielhafte Anwendung	
Funkuhr	
IKT-Unterstützung	
SysML-Modellierungswerkzeuge	

A2.8 Fragebogen zur Ermittlung der Herausforderungen

Fragebogen zur Ermittlung der Herausforderungen und der Ziele im ImPaKT-Projekt

Bitte beantworten Sie die folgenden Fragen zu Ihrer individuellen Ausgangssituation im ImPaKT-Projekt.

Vorgehen:

- **Priorität:** Gerne können sie auf den Vorarbeiten aufbauen, die Sie zum ImPaKT Kick-off zusammengetragen haben. Das vorgeschlagene Schema hierzu ist: A (Hohe Priorität), B (Mittlere Priorität), C (Geringe Priorität).
- **Nutzergruppen:** Ebenfalls können Sie die Nutzergruppen (Rollen im Unternehmen) nennen (z.B. Projektleitung, Entwickler*in, Einkäufer*in, Anforderungsmanager*in) auf die Ihre Antwort zutrifft.

Herausforderungen und Probleme:

I. Interne Prozesse

a. Wie greifen Sie auf benötigte Daten zu und welche Hürden gibt es dabei zu überwinden?

Antwort	Priorität	Betroffene Nutzergruppe(n)
Wir haben eine Explorer-Ordnerstruktur in der wir nach Kunden geordnet in Teile-Ordern und Unterordnern die Daten ablegen → Bei nicht standardmäßigen Prozessen macht schnell jeder seine eigenen kleinen Sammelordner oder legt Daten nicht sauber ab → Es ist nicht eindeutig erkennbar, dass die Daten auch dem aktuellen Stand entsprechen	A	Gesamte unternehmen
Wir Erhalten Daten und Tauschen diese per Mail aus	A	Gesamte unternehmen
Wenn die Datenpakete zu groß sind nutzen wir kostenlose Portale wie file2send o.ä.	C	Gesamte unternehmen

b. Wie sieht der Entscheidungsprozess aus und welche Probleme entstehen dabei?

Antwort	Priorität	Betroffene Nutzergruppe(n)
Entscheidung wird in Bilateral Gesprächen nur, wenn notwendig in Meetings getroffen → es können Perspektiven übersehen werden wenn Problemstellungen nicht erkannt werden so die notwendigen Spezialisten nicht zugezogen werden	B	Kalkulation Modellbau Qualitätsstelle Gießereileitung Messtechnik Vertrieb/GF

c. Wissen Sie, in welchem Umfang die Änderungen Auswirkungen auf Ihre Entwicklungsziele haben? (Qualität, Kosten, Zeit, ...)

Antwort	Priorität	Betroffene Nutzergruppe(n)
In dieser Tiefe werden wir nicht in den PEP mit eingebunden		

Bild A-4: Darstellung des Fragebogens (I/III)

d. Welche sonstigen internen Probleme/Herausforderungen entstehen innerhalb des Änderungsmanagement? (Widersprüchlichkeiten, späte Korrekturbedarfe, ...)

Antwort	Priorität	Betroffene Nutzergruppe(n)
Bei Änderungsbedarfen durch Probleme beim Gießprozess müssen Anfragen gestellt werden, die konkret geprüft werden müssen → iterierende Diskussionen mit dem Auftraggeber → eventuelle Terminverzögerung	A	Modellbau Qualitätsstelle Gießereileitung Messtechnik
Bei Änderungsanfragen seitens Auftraggeber nach Gießplanung und Modellerstellung müssen Machbarkeit und Risiken durch Fehlstellen im Guss neu bewertet werden → Zeitverlust und iterierende Gespräche, da die Änderungen oft nicht gussgerecht ausgestaltet sind	A	Modellbau Qualitätsstelle Gießereileitung Messtechnik Vertrieb

II. Externe Prozesse

a. Wie erfolgt die Zusammenarbeit mit Zulieferern und Kunden und welche Probleme treten dabei auf? (Kommunikation, Datenaustausch, Anzahl der Nachfragen, ...)

Antwort	Priorität	Betroffene Nutzergruppe(n)
Die Kommunikation erfolgt lediglich über Telefon/E-Mail/Meetings.	A	Vertriebler*in beim Kunden

b. Wie erfolgt die Abschätzung der Änderungsauswirkung gemeinsamen mit den Kunden und Zulieferern und welche Probleme treten dabei auf? (Preise, Machbarkeit/Realisierung, unbekannte Auswirkungen, widersprüchliche Anforderungen, ...)

Antwort	Priorität	Betroffene Nutzergruppe(n)
Kunden: Es werden Anfragen zu Änderungsbedarfen gestellt und intern bewertet. Anschließend werden eventuell entstehende Prozessänderungen in die bestehende Kalkulation eingepflegt und ein Preisvorschlag gemacht und diskutiert. → Problematisch sind die vielen Absprachen, bis man zu einem Ergebnis kommt, da oft für den Kunden schwer nachvollziehbar ist wie Mehrkosten durch unterschiedliche Änderungen entstehen → Zwei verschieden Interessengruppen: Konstruktion (QS) „VS“ Einkauf	A	Kalkulation Modellbau Qualitätsstelle Gießereileitung Messtechnik Vertrieb/GF
Zulieferer: Es werden Anfragen zu Änderungsbedarfen gestellt und intern beim Zulieferer bewertet.	B	Kalkulation Modellbau Qualitätsstelle

Bild A-5: Darstellung des Fragebogens (II/III)

Anschließend wird ein Preisvorschlag gemacht und diskutiert.		

III. Zielperspektive

a. Welche Nutzergruppe (Rolle) sollte unternehmens-intern auf welche Art von den Projektergebnissen profitieren?

Antwort	Priorität	Betroffene Nutzergruppe(n)
Die Datenverarbeitenden Abteilungen sollten selbst kontrollieren können, ob Daten aktuell sind und wenn nicht diese herunterladen können	A	Kalkulation Modellbau Qualitätsstelle Messtechnik

b. Welche Nutzergruppe (Rolle) sollte in Zusammenarbeit mit Kunden und Zulieferern auf welche Art von den Projektergebnissen profitieren

Antwort	Priorität	Betroffene Nutzergruppe(n)
Es sollte eine schnelle „Vorabkontrolle“ durch das System möglich sein, um Änderungswünsche von uns zu bewerten, dass sollten gusstechnische Probleme bestehen wir nicht mögliche Änderungsanfragen schneller verwerfen können und schneller mit realisierbaren Anfragen an den Auftraggeber herantreten können	A	Kalkulation Modellbau Qualitätsstelle Gießereileitung Messtechnik
Produkte sollten schon von Beginn des PEP an nach grundlegenden gusstechnischen Eigenschaften geprüft werden um nicht „von Null an“ alles anpassen zu müssen	A	Kalkulation Modellbau Qualitätsstelle Gießereileitung Messtechnik Vertrieb/GF
Zugriff von allen Beteiligten auf Daten, Zeichnungen etc. (Aktueller Stand?)	A	Kalkulation Modellbau Qualitätsstelle Messtechnik Vertrieb
Deutlich beschleunigte Entscheidungsfindungsprozesse Sowohl Konstruktiv, als auch Preislich	A	Kalkulation Modellbau Qualitätsstelle Gießereileitung Messtechnik Vertrieb/GF
Eine für alle ersichtliche Terminschiene / Terminplanung (Statusmeldungen z.B. „in Bearbeitung“...)	B	Kalkulation Modellbau Qualitätsstelle Gießereileitung Messtechnik Vertrieb/GF

Bild A-6: Darstellung des Fragebogens (III/III)

A3 Anforderungen und Anwendungsfälle

A3.1 Anwendungsfälle der Auswirkungsanalyse

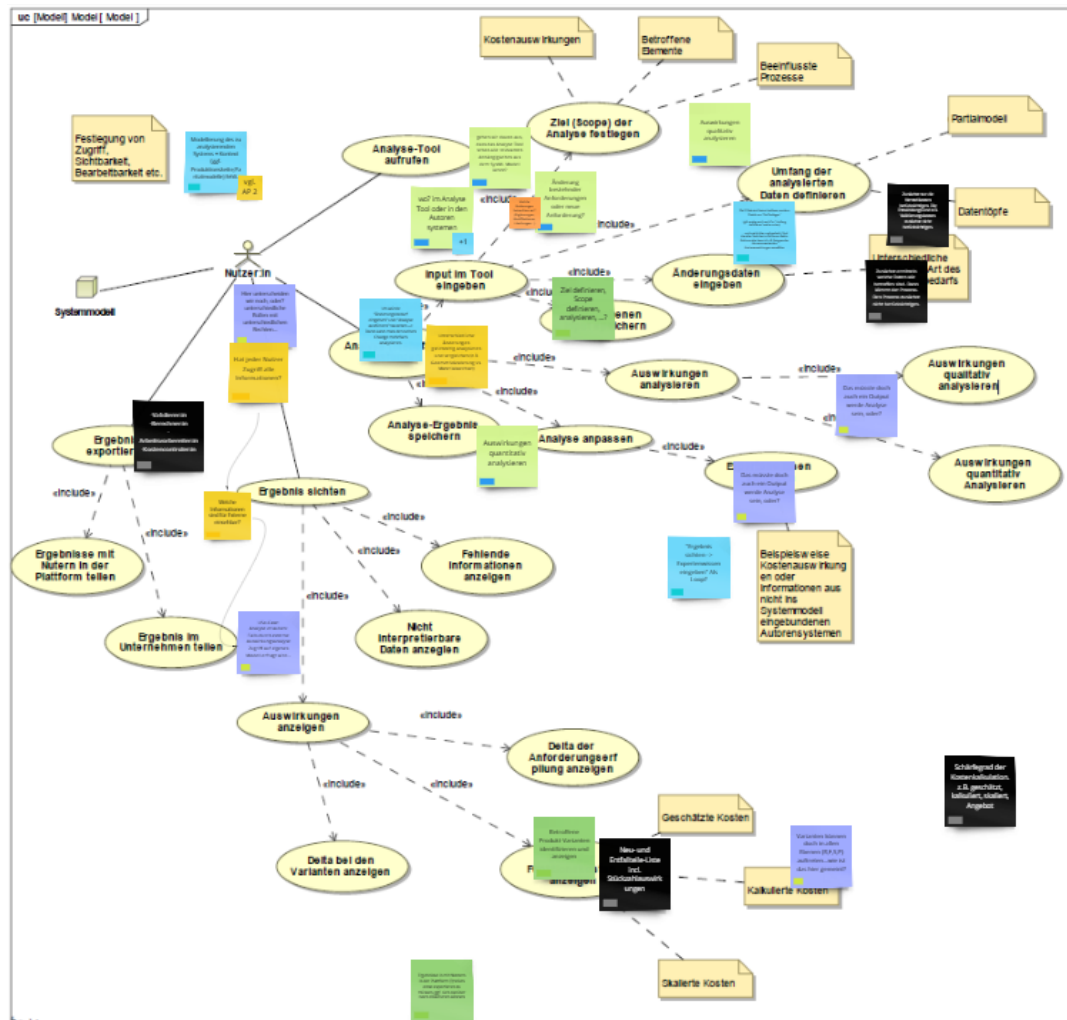


Bild A-7: Workshop zu Anwendungsfällen

A3.2 Nummerierung der Anwendungsfälle

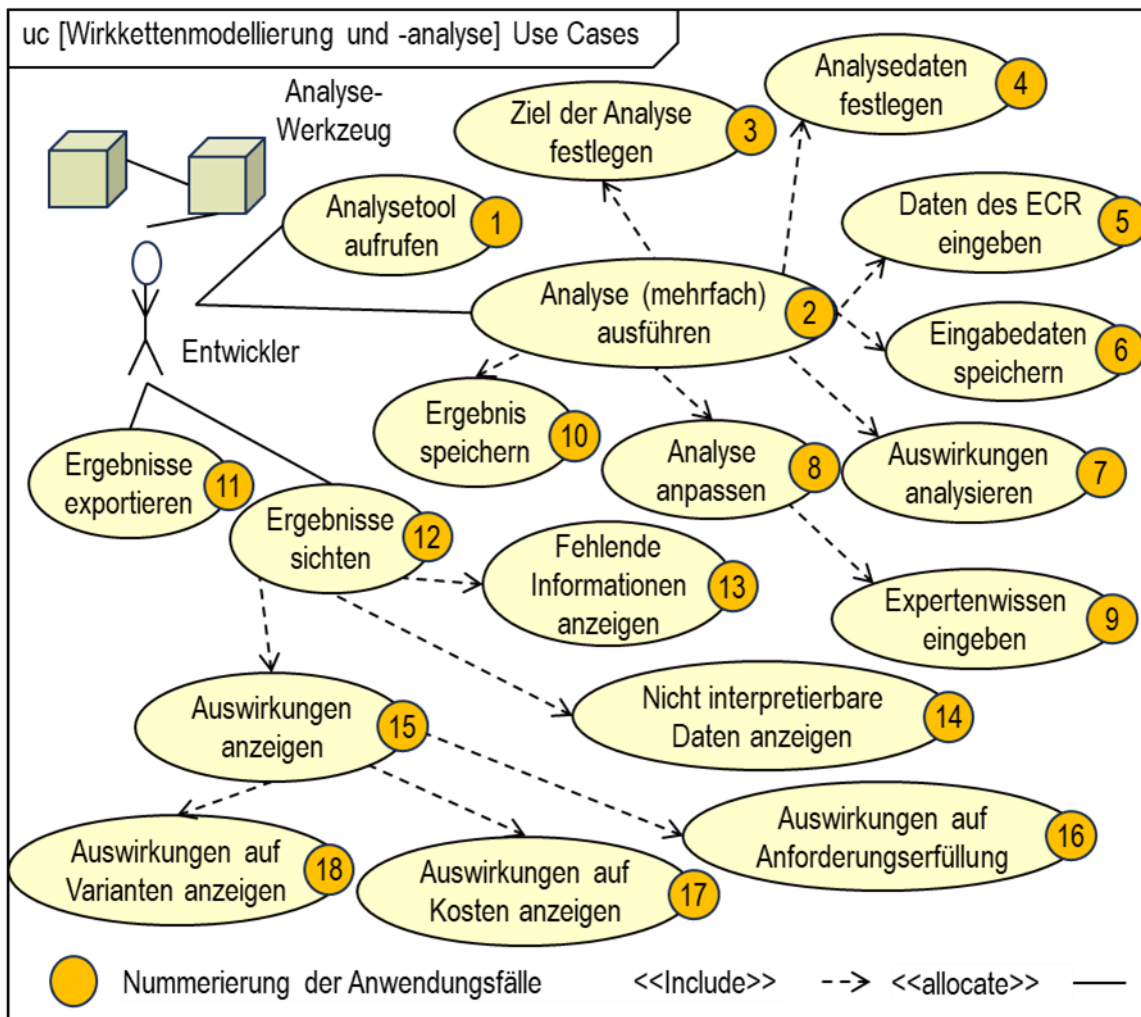


Bild A-8: Nummerierung von Anwendungsfällen

A3.3 Anforderungsworkshops zur Wirkkettenanalyse

Kategorie	Systemanforderung	Beschreibung	Input AP-Teilnehmer	Vorüberlegungen Unter-AP-Leiter
Input-orientierte Anforderungen:	Verfügbarkeit von Daten / Informationen	Die Wirkkettenanalyse muss so gestaltet sein, dass Sie mit in den Informationen und Daten aus dem integrierendes Systemmodell durchgeführt werden kann		
Input-orientierte Anforderungen:	Unterschiedliche Arten der Änderung	Der Änderungsbedarf muss sich auf unterschiedliche Elemente beziehen können (Anforderungen, Strukturelemente, Verhaltenselemente)		
Anforderungen an Modell und Modellierung:	Propagationseffekte (Änderungsausbreitungen) berücksichtigen	Die Algorithmen der Wirkkettenanalyse müssen Propagationseffekte (Ausbreitungen von Änderungen) anhand der Abhängigkeiten berücksichtigen		
Anforderungen an Modell und Modellierung:	Tiefe und Breite der Analyse	Die Algorithmen müssen Informationen auf verschiedenen Granularitätsleveln (Tiefe) und aus verschiedenen Disziplinen (Breite) berücksichtigen		
Anwendungsorientierte Anforderungen:	Zeitlicher Anwendungsaufwand	Der zeitliche Anwendungsaufwand der Wirkkettenanalyse muss in der industriellen Praxis akzeptabel sein		
Anwendungsorientierte Anforderungen:	Tailoring / Anwenderindividuelle Analyse	Die Ergebnisse der Wirkkettenanalyse müssen an die unternehmensspezifischen Bedarfe angepasst werden können (Tailoring)		
Anwendungsorientierte Anforderungen:	Akteurübergreifende Kollaboration	Die Datenauswertung und Bereitstellung muss akteurübergreifend (z.B. mit Zulieferern) möglich sein		
Ergebnisorientierte Anforderungen:	Ergebnisqualität	Die Ergebnisse der Wirkkettenanalyse müssen eine hinreichende Qualität aufweisen und die Anforderungen erfüllen		
Ergebnisorientierte Anforderungen:	Nachvollziehbarkeit	Die Ergebnisse und das Vorgehen der Wirkkettenanalyse sollen nachvollziehbar und transparent sein		
Ergebnisorientierte Anforderungen:	Bewertung der Kosten	Die entstehenden Funktionskosten der potenziellen Änderung müssen mit vorhandenen Informationen bewertet werden können		
Ergebnisorientierte Anforderungen:	Industrielle Anwendbarkeit	Die Wirkkettenanalyse muss auf unterschiedliche Unternehmensgrößen anwendbar sein, um in der industriellen Praxis zu kommen		

Bild A-9: Ergebnis aus Anforderungsworkshop

A4 Konzeption der MECA-Methodik

A4.1 Vorgehen zur Erstellung des Rollenmodells

Ergebnisse des Strings in zwei verschiedenen Forschungsplattformen

Tabelle A-11: Suchstring zur Identifikation von Rollenmodellen

		Results	Selected
Web of Science	TS=(“Role-Model” OR “Role Model” OR “Roles”) AND “Project Management”	518	11
	TS=(“Role-Model” OR “Role Model” OR “Roles”) AND (“Systems Engineering” OR “SE”)	111	6
	TS=(“Role-Model” OR “Role Model” OR “Roles”) AND (“Organizational Management”) //	153	5
	TS=(“Role Model” OR “Role-Model”) AND “Organization”		
Google Scholar	allintitle: Role Model “Project Management” //	116	10
	allintitle: Role-Model “Project Management” //		
	allintitle: Roles “Project Management”		
	allintitle: Role Model “Systems Engineering” //	41	4
	allintitle: Role-Model “Systems Engineering”		
	allintitle: “Role Model” “Organizational Management” //	183	5
	allintitle: “Role-Model” Organizational //		
	allintitle: Role Model Organization		
		1.194	41

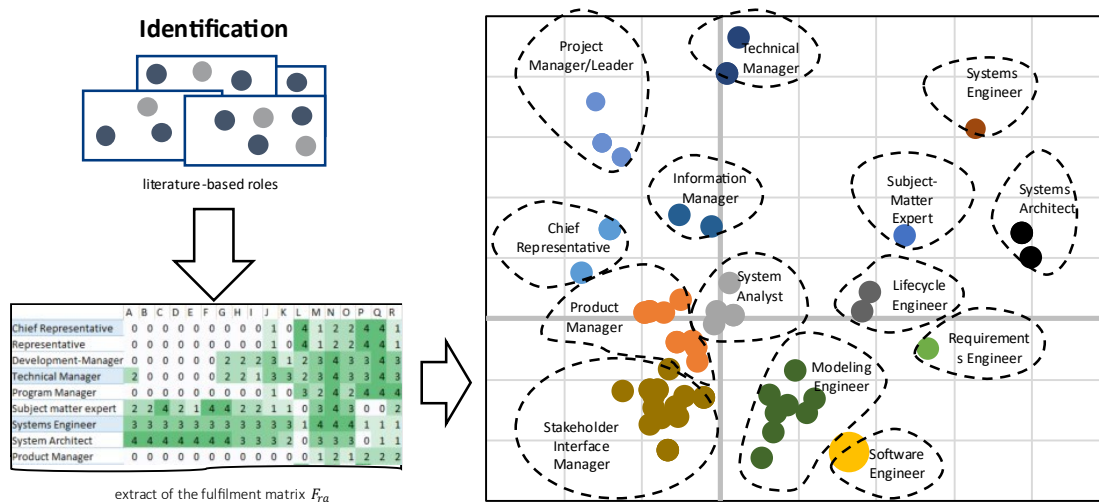


Bild A-11: Darstellung des Vorgehens zum Clustering

A4.2 Modellierungsregeln

Tabelle A-12: Auflistung von Modellierungsregeln

Nummer	Modellierungsregel
1	Die einzelnen Ebenen der RFLPV ² -Logik müssen in einheitlichen Abstraktionsebenen modelliert werden. Der höchste Detaillierungsgrad wird durch das Ziel der Modellierung und Analyse bestimmt. Ausgehend von der anfänglichen Wahl der Modellierungsintensität werden die anderen Elemente angepasst.
2	Requires-Relationen sind horizontale Relationen und können nur auf einer Hierarchieebene verwendet werden.
3	Refines-Relationen sind horizontale oder vertikale Relationen und werden als Verfeinerung bzw. Detaillierung eines Startelements durch ein Endelement verwendet.
4	Bei der Erweiterung einer Hierarchie durch eine neue Abstraktionsebene (Dekomposition) müssen alle vorhandenen Elemente auf der neuen Abstraktionsebene definiert werden.
5	Aus jeder Stakeholder-Anforderung muss mindestens eine Systemanforderung abgeleitet werden.
6	Satisfies-, Specifies- und Realizes-Relationen sind auf der detailliertesten Abstraktionsebene miteinander verbunden.
7	Verifies und Validates-Relationen werden nur verbunden, wenn die Anforderung erfüllt ist

A4.3 Ergebnisdimensionen

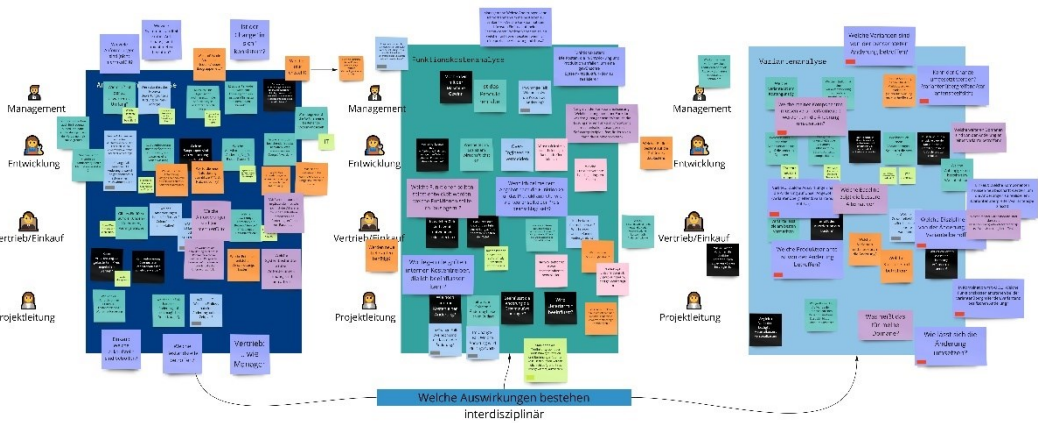


Bild A-12: Darstellung der Sammlung an Ergebnisdimensionen

Tabelle A-13: Auflistung der Ergebnisdimensionen

–	ED1	PM/RM	Wie viele Anforderungen sind (nicht mehr) erfüllt?
---	-----	-------	----------------------------------------------------

	ED2	PM/RM	Welche Anforderungen sind (nicht mehr) erfüllt?
	ED3	SME	Wie hoch ist die Differenz zwischen aktuellem Parameter und anforderungserfüllendem Parameter?
	ED4	SA	Welche Systembestandteile (über Anforderungen hinaus) sind betroffen?
	ED5	RM	Warum ist die Anforderung nicht erfüllt? (z.B. Parameter nicht im definierten Range oder keine verknüpfte Funktion modelliert, die die Anforderung erfüllt...)?
	ED6	EK	Welche Zukaufsteile sind betroffen?
	ED7	EK	Welche Bestandteile betroffen?
	ED8	SE	Welche weiteren Elemente sind betroffen?
	ED9	SE	Wie lässt sich die Anforderung kategorisieren?
	ED10	RM	Welche Anforderungen sind kritisch?
	ED11	SA/SME	Welche Baugruppen sind von Anforderung betroffen?
	ED12	PM/SA	Welche Auswirkungen hat Änderung am Element A auf Element B, C oder D?
	ED13	SME	Inwiefern ist das bestehende Produkt (System) betroffen?
	ED14	V&V	Was muss alles neu überprüft (verifiziert/validiert) werden?
	ED15	SIM	Gibt es Einflüsse auf die Kunden / Lieferanten, wenn ja welche?
	ED16	PM	Wen müssen wir einbinden?
	ED17	EK	Welche Anforderungen haben Einfluss auf Zukaufteile/-materialien?
	ED18	VT	Ist Lagerware oder ein Rahmenkontrakt betroffen?
	ED19	PM	Welche Auswirkungen entstehen für Betrieb / Service / ET-Kataloge / Handbücher etc.?
	ED20	VT	Kann die Anforderung zu gefordertem Preis angeboten werden?
	ED21	SA	Kann die Anforderung über unseren Standardbaukasten abgebildet werden?
	ED22	VT/PM	Welche Entfallteile entstehen?
	ED23	RM	Wie sind die Anforderungen priorisiert? (Festanforderung/Wunschanforderung)
	ED24	QM/RM	Welchen Einfluss hat eine Änderung die Qualität?
	ED25	PM	Welche Änderungen kann ich vornehmen, um meine Kosten zu senken?
	ED26	SA/PM	Welche Funktion hat den höchsten Einfluss auf meine Gesamtkosten?
	ED27	SA	Welche Lösungen meiner Funktion werden günstiger sein?
	ED28	VT	Wenn ich bei meinem Angebot noch die Funktion XY für das Produkt anbiete, wie viel teurer sollte der Preis beanschlagt sein?
	ED29	PM	Wo liegen die größten internen Kostentreiber, die ich beeinflussen kann?
	ED30	PM	Welche Elemente haben Einfluss auf meine Kosten?
	ED31	PM	Machen wir mit der Maschine (dem System) Gewinn?
	ED32	PM	Ist das Produkt rentabel?
	ED33	PM/VT	Wie hoch sind die Kosten der Änderung?
	ED34	SA	Können Funktionen nach Anforderungsänderung mit gleichen Teilen und gleichen Kosten noch abgebildet werden?
	ED35	PM	Welche technische Lösung ist am wirtschaftlichsten?

	ED36	PM/VT	In welcher Zeit sind notwendige Teile lieferbar?
	ED37	VT	Welche Kosten übernehmen wir? Welche werden an den Kunden weitergereicht?
	ED38	VT	Wer trägt die Änderungskosten?
	ED39	SA	Mit welcher Lösung bleibe ich im Budget?
	ED40	PM	Beeinflusst die Änderung die Gesamtaufwendungen?
	ED41	PM	Wird Liefertermin beeinflusst?
	ED42	PM/VT	Welche Preisanpassungen sind notwendig?
	ED43	PM	Welche Lösungen kosten nach Änderung wie viel?
	ED44	SA	Welcher Einfluss besteht auf die Größe des Baukastens?
	ED45	SA	Welche Variante ist am kostengünstigsten?
	ED46	SA	Welche Varianten sind von der betrachteten Änderung betroffen?
	ED47	SA	Kann die Änderung variantenübergreifend umgesetzt werden?
	ED48	SME	Welche meiner Komponenten müssen verändert/erneuert werden, um die Änderung umzusetzen?
	ED49	SME	Was heißt das für (m)eine Disziplin?
	ED50	SME	Welche weiteren Disziplinen sind von der Änderung an einer Variante betroffen?
	ED51	SME	Welche meiner Partialmodelle sind durch die variantenübergreifende/variantenspezifische Änderung betroffen?
	ED52	EK	Welche Komponenten müssen variantenübergreifend neu beschafft werden, um die Änderungen zu realisieren?
	ED53	VT	Welche Auswirkungen hat die Änderung auf unser Angebot? (variantenübergreifend/variantenspezifisch)
	ED54	SME	Welche Disziplinen sind von der Änderung an der Variante betroffen?
	ED55	SA	Welche Produktvariante ist von der Änderung betroffen?
	ED56	SA	Welchen Einfluss hat eine Variantenänderung auf andere Produktreihen / Programme
	ED57	SA	Kann ich Varianten (Baukasten) weiter standardisieren, um unterschiedliche Anforderung zu erfüllen?
	ED58	SA	Müssen neue Komponenten entwickelt werden, um Variante zu erstellen um neue Anforderung zu erfüllen?
	ED59	SA	Welche Abhängigkeiten bestehen im Variantenbaum?
	ED60	RM	Welche Variante erfüllt die Anforderung am besten?
	ED61	SA	Welche Zukaufteile/Materialien betreffen mehrere Varianten.
	ED62	SA	Welche Varianten entfallen durch die Änderung?
	ED63	VT	Welche Kunden sind betroffen?
	ED64	PM	Wie groß sind die Aufwände der unterschiedlichen Varianten und was lässt sich bis zum Zieldatum umsetzen

RM = Requirements Manager, SA = Systemarchitekt, SME = Subject Matter Expert, PM = Projektmanager, SIM = Stakeholder Interaktionsmanager, EK = Einkauf, VT = Vertrieb, V&V = V&V Ingenieur (in Anlehnung an MBSE Rollenmodell)

Hinweis: Duplikate wurden entfernt. Ebenso Ergebnisdimensionen, welche folgenden Kriterien nicht erfüllten: Betrifft die Entwicklungsphase, ist durch Entwicklungsartefakte beantwortbar, gibt Bewertungsrahmen (Metrik) vor

A4.4 RFLPV Handouts

Anforderung		
Definition: Eine Anforderung ist eine Aussage, die ein Merkmal oder eine Einschränkung eines Systems, eines Produkts oder eines Prozesses identifiziert, die bestimmte Qualitätskriterien erfüllt, überprüfbar ist und als notwendig für die Akzeptanz durch die Interessengruppen erachtet wird.		Prozessaktivitäten: Erhebung, Dokumentation, Vereinbarung und Management
Befähiger: Hauptmerkmaliste, Anforderungsschablone, Qualitätskriterien	Kontrollen: IEEE STD 830, IEEE 1233, VDI 2519	Schlüsselrolle: Requirements Engineer

Funktion		
Definition: Eine Funktion ist ein Artefakt zur Beschreibung einer beabsichtigten Beziehung zwischen Eingabe und Ausgabe eines Systems mit dem Ziel, eine Aufgabe zu erfüllen.		Prozessaktivitäten: Extraktion, Strukturierung, Spezifikation
Befähiger: Funktionskatalog, Funktionsanalyse, funktionales Whitebox-Modell	Kontrollen: ISO 26262, VDI 2803	Schlüsselrolle: Systems Architect

Logisches Element		
Definition: ein logisches Element definiert bestehende generische Elemente und ihre alternativen Formen ("was") zur Lösung eines Problems		Prozessaktivitäten: Suche nach alternativen Lösungen, Erstellung einer logischen Architektur, Vergleich logischer Architekturen
Befähiger: Lösungskataloge, standardisierte Notationen, UML-Diagramme	Kontrollen : -	Schlüsselrolle: Systems Architect

Bild A-13: Darstellung der RFLPV Handouts (I/II)

Physisches Element		
Definition: Ein physisches Element ist das ausgearbeitete Design ("wie") von logischen Elementen ("was")		Prozessaktivitäten: Spezifikation der mechanischen Struktur (Geometrie, Material, Toleranzen) und des Softwareverhaltens (Signale, Nachrichten, Rahmen), Auswahl von Hardware und Software, Zuweisung von Ressourcen, Zeitplanung
Befähiger: Expertenwerkzeuge (CaX)	Kontrollen: disziplinspezifische Richtlinien (ISO, IEEE, VDI, DIN, ANSI, SAE)	Schlüsselrolle: <i>Subject-matter Expert</i>

Verifikation		
Definition: Überprüfung des implementierten Systems und seiner Elemente im Vergleich zu den gestellten Anforderungen und Beantwortung der Frage "Haben wir richtig entwickelt?"		Prozessaktivität: Planen, Durchführen, Analysieren und Dokumentieren
Befähiger: theoretische Analyse, Inspektion, Demonstration, Test	Kontrollen: disziplinspezifische Richtlinien (ISO, IEEE, VDI, DIN, ANSI, SAE)	Schlüsselrolle: <i>V&V Engineer</i>

Validierung		
Definition: Überprüfung der Systemeigenschaften gegenüber einem vordefinierten Einsatzzweck oder Nutzen und Beantwortung der Frage "Haben wir das richtige System entwickelt?"		Prozessaktivität: Planen, Durchführen, Analysieren und Dokumentieren
Befähiger: theoretische Analyse, Inspektion, Demonstration, Test	Kontrollen: disziplinspezifische Richtlinien (ISO, IEEE, VDI, DIN, ANSI, SAE)	Schlüsselrolle: <i>V&V Engineer</i>

Bild A-14: Darstellung der RFLPV Handouts (II/II)

A4.5 Vollständige Ansicht der Customizations



Bild A-15: Ansicht der Customizations

A4.6 RFLPV²-Profil

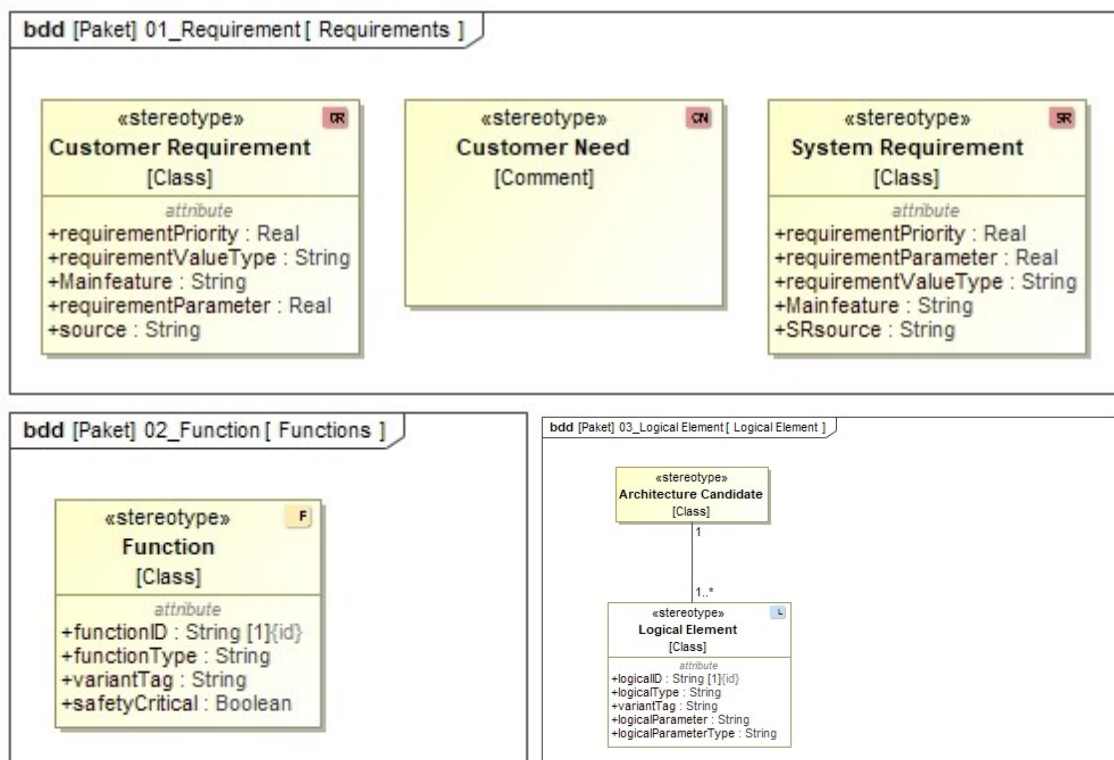


Bild A-16: Ansicht des RFLPV² Profils (I/II)

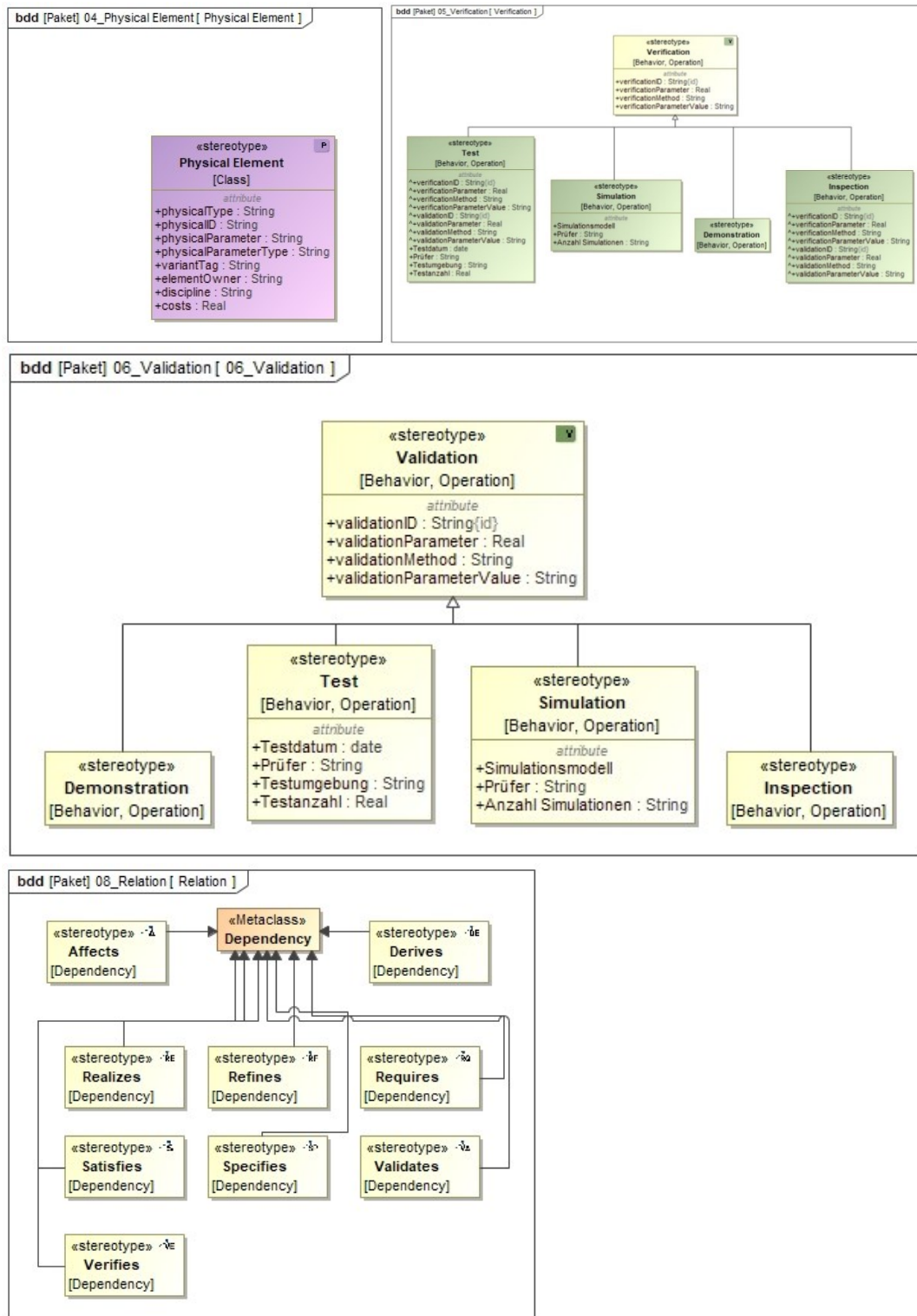


Bild A-17: Ansicht des RFPLV² Profils (II/II)

A4.7 Metamodell Graphdatenbank neo4j

$V = \{\text{Requirement, Function, LogicalElement, PhysicalElement, Verification, Validation}\}$

$E = \{\text{Requires, Refines, Dervies, Decompose, Satisfies, Specifies, Realizes, Verifies, Validates}\}$

$\rho :=$

$(\text{rel1:Requires}) = (\text{Requirement}, \text{Requirement1}),$

$(\text{rel2:Refines}) = (\text{Requirement}, \text{Requirement1}),$

$(\text{rel3:Dervies}) = (\text{Requirement}, \text{Requirement1}),$

$(\text{rel4:Satisfies}) = (\text{Function}, \text{Requirement})$

$(\text{rel5:Requires}) = (\text{Function}, \text{Function1})$

$(\text{rel6:Decompose}) = (\text{Function}, \text{Function1})$

$(\text{rel7:Specifies}) = (\text{LogicalElement}, \text{Function})$

$(\text{rel8:Requires}) = (\text{LogicalElement}, \text{LogicalElement1})$

$(\text{rel9:Decompose}) = (\text{LogicalElement}, \text{LogicalElement1})$

$(\text{rel10:Realizes}) = (\text{PhysicalElement}, \text{LogicalElement})$

$(\text{rel11:Requires}) = (\text{PhysicalElement}, \text{PhysicalElement1})$

$(\text{rel12:Decompose}) = (\text{PhysicalElement}, \text{PhysicalElement})$

$(\text{rel13:Verifies}) = (\text{Verification}, \text{Requirement})$

$(\text{rel14:Validates}) = (\text{Validation}, \text{Requirement})$

λ : Die Labels der Knoten und Kanten entsprechen den Artefakt- und Relationstypen.

σ :

Requirement { requirementID, requirementName, requirementText, requirementType, Requirement Parameter, Requirement Value Type, Requirement Priority }

Function { functionID, functionName, functionVariantTag, functionType }

LogicalElement { logicalID, logicalName, logicalParameter, logicalParameterType, logicalType, logicalVariantTag }

PhysicalElement { physicalID, physicalName, Company, physicalElementOwner, physicalParameter, physicalParameterType, Material, physicalCosts, physicalVariantTag }

Verification {verificationID, verificationName, verificatonMethod, verificationParameter}

Validation {validationID, validationName, validationMethod, validationParameter}

A4.8 Ursache-Wirk-Muster für Wirkkettenanalysen

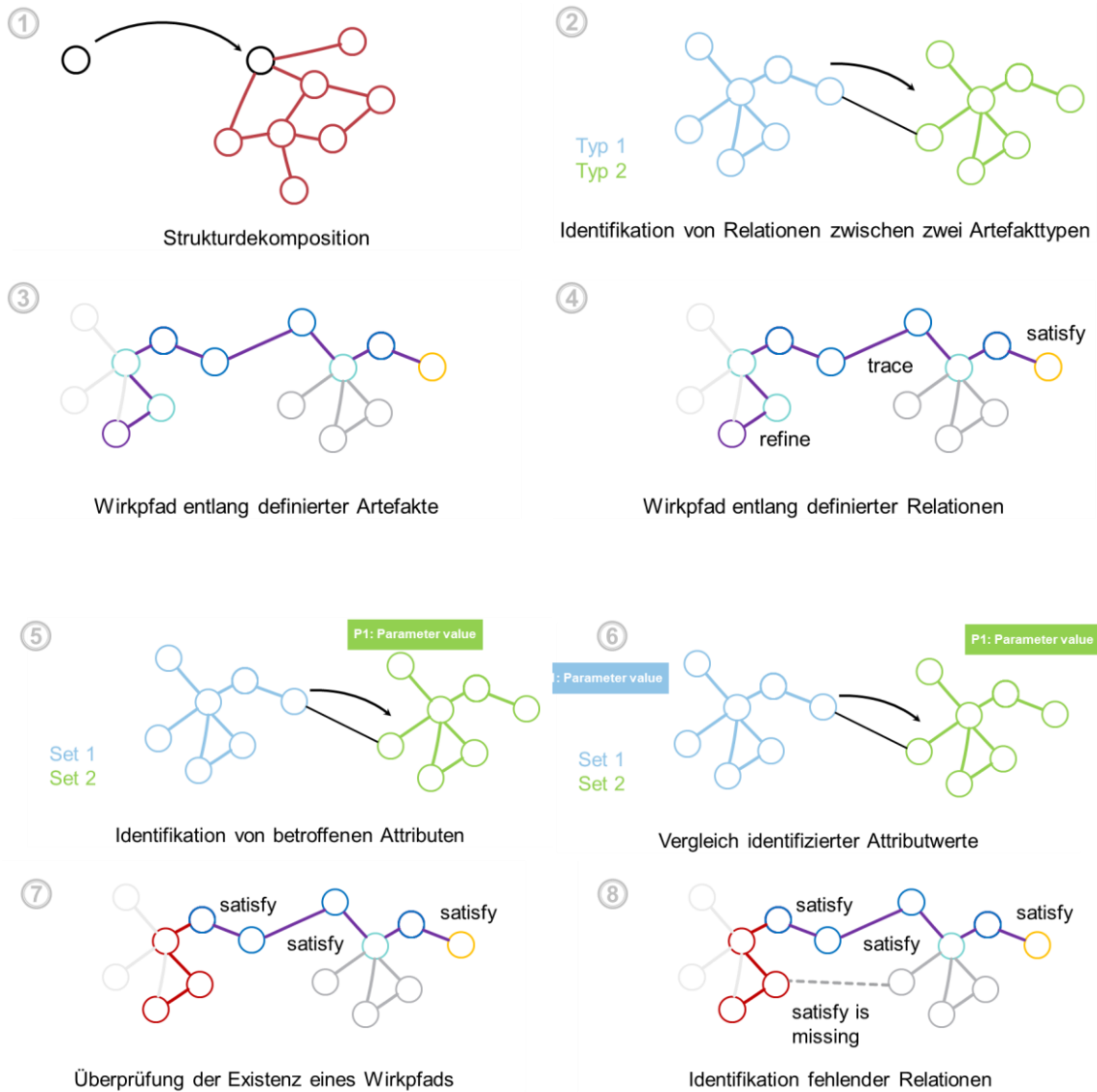


Bild A-18: Darstellung der Ursache-Wirkungs-Muster

A4.9 Basisalgorithmen

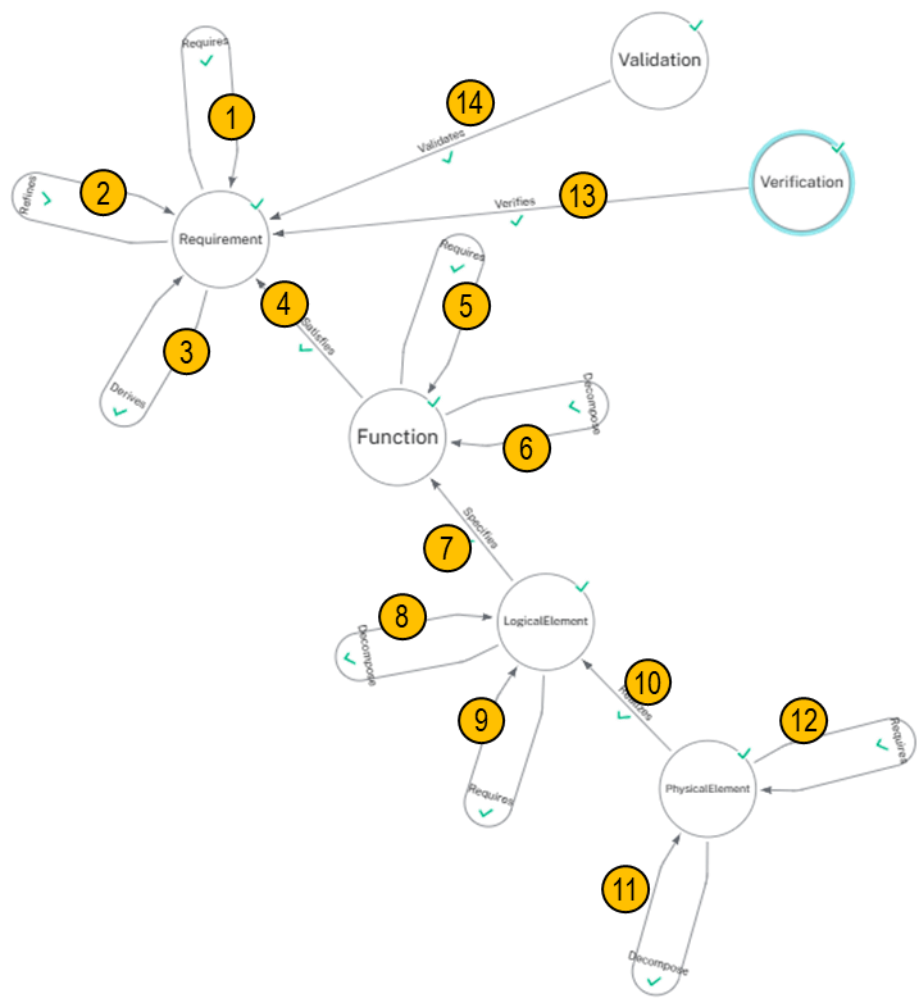


Bild A-19: Darstellung der Nummerierungen der Relationen im RFLPV² Graphen

Tabelle A-14: Basisalgorithmen auf Basis der Nummerierungen aus Bild A-19

Nr.	Ergebnisdimension	Möglichkeit 1	Visu- alisie- rung
1	Ist die betroffene Anforderung validiert?	MATCH (r:Requirement{requirement- Name:<<insertname>>>)-[rel13:Validates]- (Val:Validation)	Single Value (SV)
		RETURN CASE WHEN COUNT(DISTINCT Val) > 0 THEN '✓' ELSE '✗' END AS Status	

2	Ist die betroffene Anforderung verifiziert?	MATCH (r:Requirement{requirementName: <<insertname>>})<-[rel14:Verifies]-(Ver:Verification) RETURN CASE WHEN COUNT(DISTINCT Val) > 0 THEN '✓' ELSE '✗' END AS Status	SV
3	Welche Anforderungen sind von einer initialen Anforderungsänderung per Refines betroffen?	MATCH (r:Requirement{requirementName: <<insertname>>})-[rel3:Refines]->(r2:Requirement) RETURN r, rel3, r2	Graph (G)
4	Welche Anforderungen sind von einer initialen Anforderungsänderung per Requires betroffen?	MATCH (r:Requirement{requirementName: <<insertname>>})-[rel2:Requires]->(r2:Requirement) RETURN r, rel2, r2	G
5	Wie ist die Priorität betroffener Anforderungen?	MATCH (r:Requirement{requirementName: <<insertname>>}) RETURN r.requirementPriority	SV
6	Welche Kategorie haben betroffene Anforderungen?	MATCH (r:Requirement{requirementName: <<insertname>>}) RETURN r.requirementCategory	SV
7	Welche Anforderungen sind validiert?	MATCH (r:Requirement)<-[rel13:Validates]-(Val:Validation) RETURN DISTINCT r	G
8	Welche Anforderungen sind verifiziert?	MATCH (r:Requirement)<-[rel14:Verifies]-(Ver:Verification) RETURN DISTINCT r	G
9	Welche Anforderungsparameter müssen geprüft werden?	MATCH (r:Requirement)-[rel1_2_3:Derive Requires Refines]->(r2:Requirement) RETURN DISTINCT r2.requirementName AS Anforderungsname, r2.requirementParameter AS Anforderungsparameter	T
10	Welche Funktionen sind direkt von einer Anforderungsänderung betroffen?	MATCH (r:Requirement {requirementName: <<insertname>>})<-[rel4:Satisfies]-(f:Function) RETURN r, rel4, f	G
11	Welche Art von Funktion ist betroffen?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel1:Derive]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function) RETURN f.functionName as Funktionsname, f.functionType as Funktionsart	T
12	In welcher Variante wird die betroffene Funktion implementiert?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel1:Derive]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)	T

		RETURN f.functionName as Funktionsname, f.functionVariantTag as `Variante der Funktion`	
13	Welche logischen Elemente sind von einer Anforderungsänderung betroffen?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel1:Derive]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalElement) RETURN r, rel1, r2, rel4, f, rel7, l	G
14	Welche Art von logischem Element ist betroffen?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel1:Derive]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalElement) RETURN l.logicalName AS `logisches Element`, l.logicalType AS `Art des logischen Elements`	T
15	In welcher Variante wird das logische Element implementiert?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel1:Derive]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalElement) RETURN l.logicalName AS `logisches Element`, l.logicalVariantTag AS `Art des logischen Elements`	T
16	Welche physischen Elemente sind von einer Anforderungsänderung betroffen?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel1:Derive]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalElement)<-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement) RETURN r, rel1, r2, rel4, f, rel7, l, rel10, p	G
17	Welche Art von physischen Element ist betroffen?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel1:Derive]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalElement)<-[rel10:Realizes]-(p:PhysicalElement) RETURN p.physicalName AS `physisches Element`, p.physicalType AS `Art der technischen Änderung`	T
18	Welche Disziplinen sind von einer Änderung betroffen?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel1:Derive]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalElement)<-[rel10:Realizes]-(p:PhysicalElement) RETURN p.physicalName AS `physisches Element`, p.physicalDiscipline AS Disziplin	T
19	Welche Varianten sind von einer technischen Änderung betroffen?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel1:Derive]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalElement)<-[rel10:Realizes]-(p:PhysicalElement) RETURN p.physicalName AS `physisches Element`, p.physicalVariantTag AS Variante	T

20	Wird ein physisches Element in einer oder in mehreren Varianten verwendet?	MATCH (p:PhysicalElement {physicalName: <<insertname>>})-[rel12:Decompose]->(p2:PhysicalElement) RETURN p2.physicalVariantTag as Variante	SV
21	Welche Person ist verantwortlich für ein physisches Element?	MATCH (p:PhysicalElement {physicalName: <<insertname>>})-[rel12:Decompose]->(p2:PhysicalElement) RETURN p.physicalName AS `physisches Element`, p.physicalElementOwner AS `verantwortliche Person`	SV
22	Aus welchem Material besteht ein physisches Element?	MATCH (p:PhysicalElement {physicalName: <<insertname>>})-[rel12:Decompose]->(p2:PhysicalElement) RETURN p2.physicalMaterial AS Material	SV
23	Wie hoch sind die Kosten der betroffenen physischen Elemente?	MATCH (p:PhysicalElement {physicalName: <<insertname>>})-[rel12:Decompose]->(p2:PhysicalElement) RETURN p2.physicalName AS `physisches Element`, p2.physicalCosts AS Kosten	T
24	Welches physische Element ist der größten Kostentreiber?	MATCH (p:PhysicalElement {physicalName: <<insertname>>})-[rel12:Decompose]->(p2:PhysicalElement) WITH p2 RETURN MAX(p2.physicalCosts) AS maxCost	SV
25	Wie viele Refines-Kanten mit gehen von einem Requirement aus?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel]->() WITH COUNT(rel) AS outgoingEdgesCount RETURN outgoingEdgesCount AS `ausgehende Kanten`	SV
26	Ist die Anforderung (ID:29) in Bezug auf die Beziehung Refines ein Absorbierer, ein Träger oder ein Multiplizierer?	MATCH (r:Requirement {requirementName: <<insertname>>})-[rel]->() WITH COUNT(rel) AS outgoingEdgesCount RETURN CASE WHEN outgoingEdgesCount = 0 THEN 'Absorbierer' WHEN outgoingEdgesCount = 1 THEN 'Träger' ELSE 'Multiplizierer' END AS result	SV

A4.10 MECA Poster

Bild A-20: MECA Poster (I/V)

HEINZ NIXDORF INSTITUT
UNIVERSITÄT PADERBORN

»MECA Methodik

Zieldefinition der Auswirkungsanalyse I

«

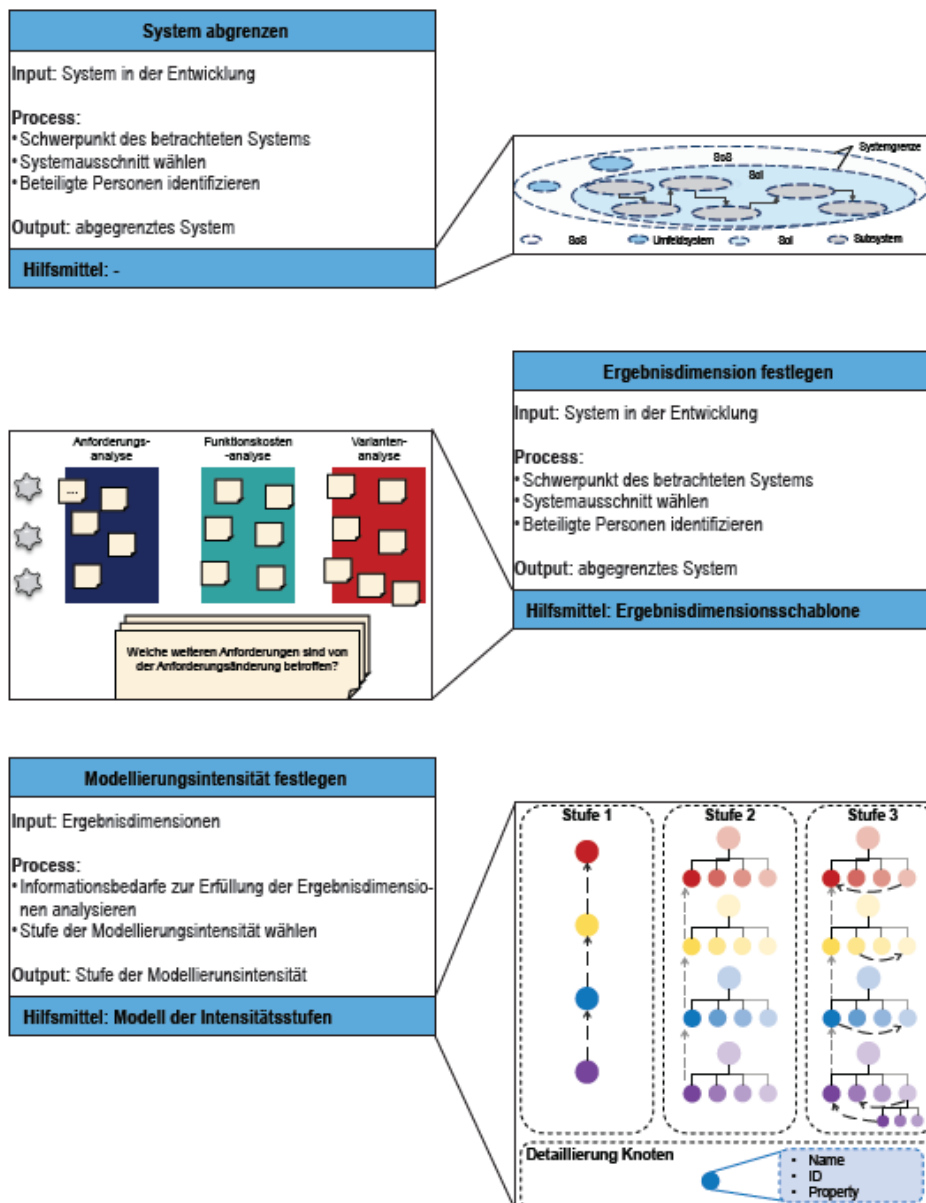


Bild A-21: MECA Poster (II/V)

» MECA Methodik

Zieldefinition der Auswirkungsanalyse II

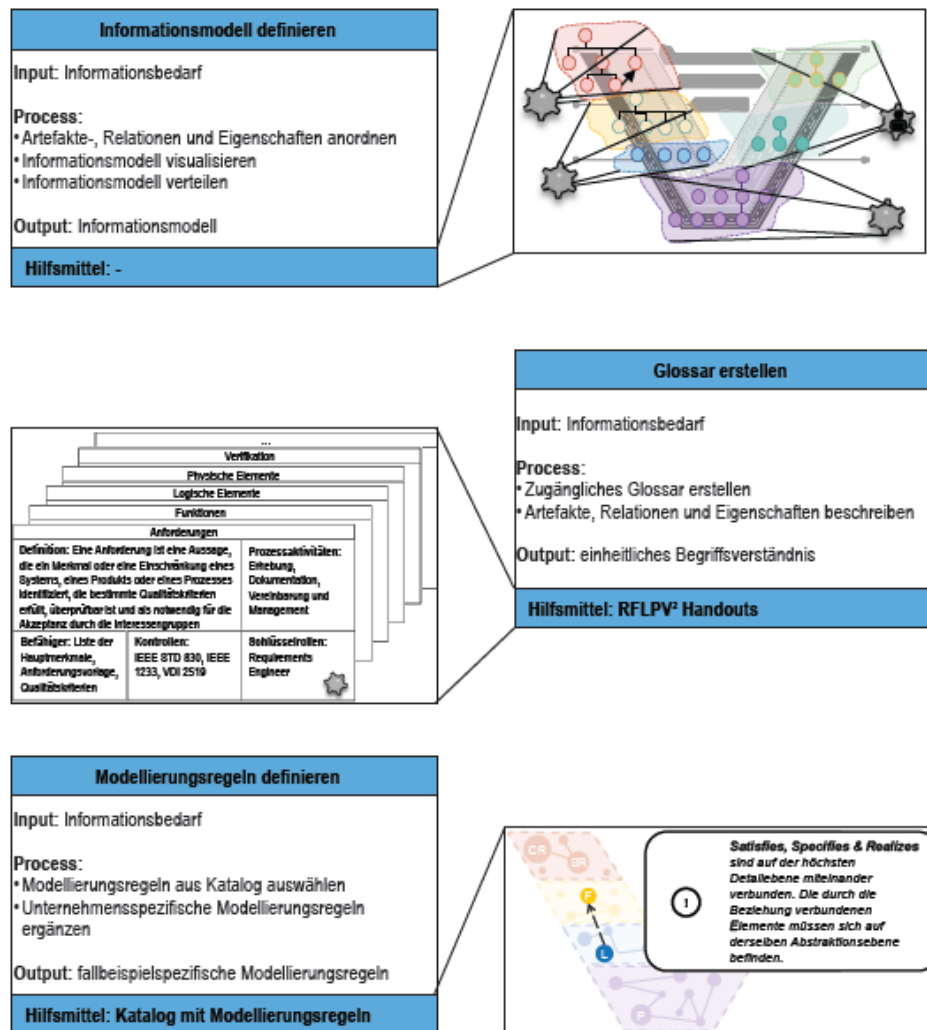


Bild A-22: MECA Poster (III/V)

» MECA Methodik

Identifikation von Informationen

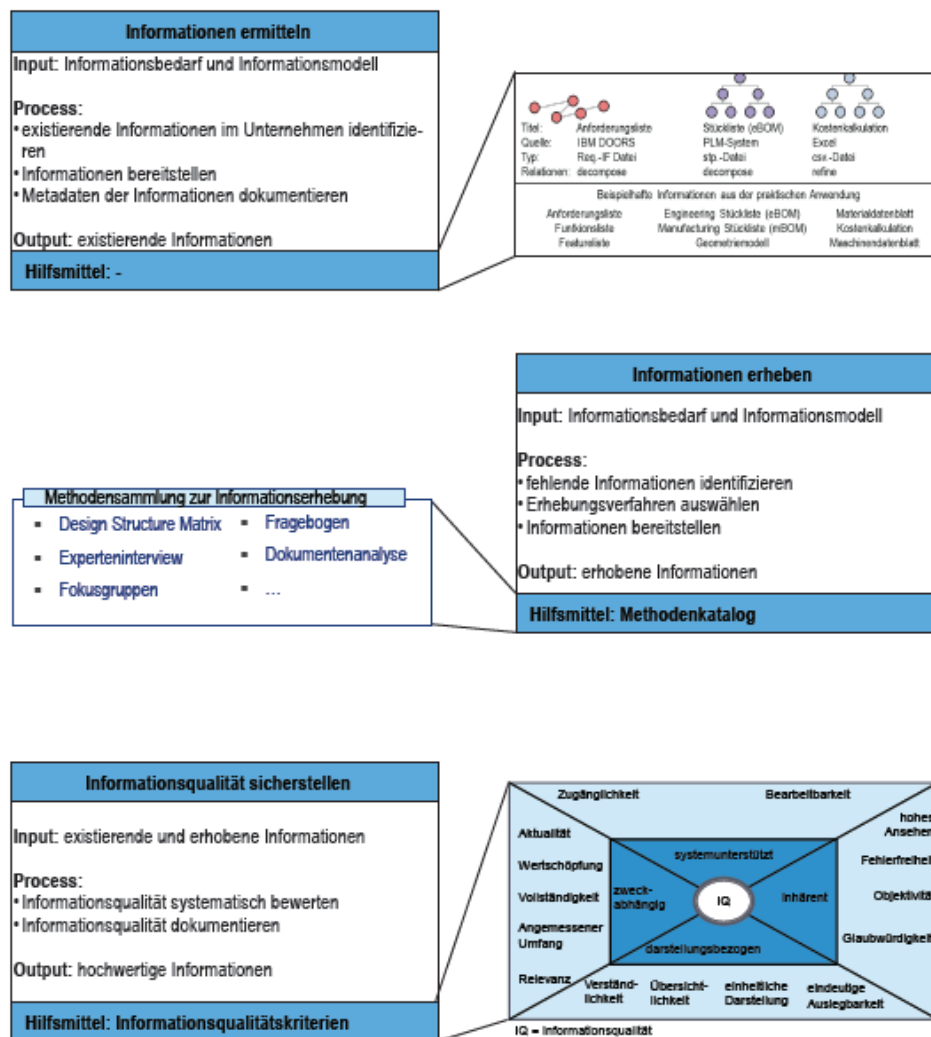


Bild A-23: MECA Poster (IV/V)

»MECA Methodik

Identifikation von Informationen

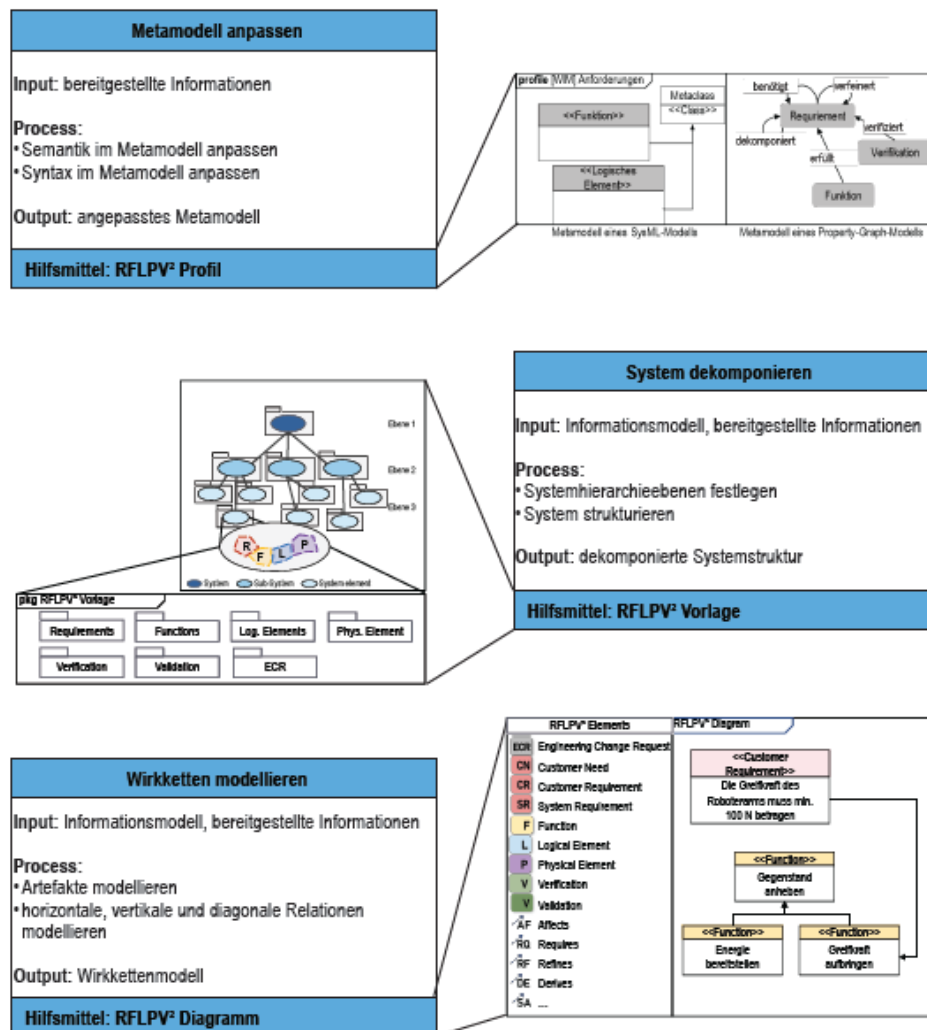
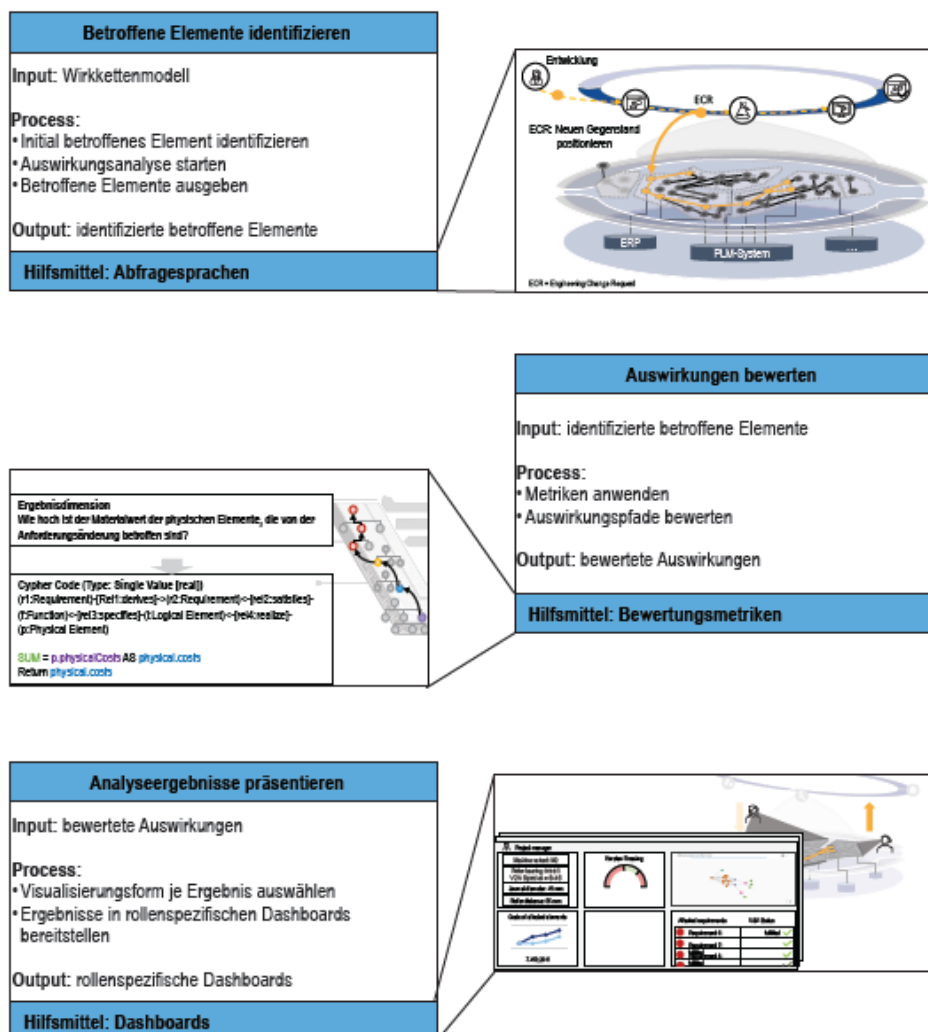


Bild A-24: MECA Poster (V/V)

» MECA Methodik

Durchführung der Wirkkettenanalyse «



A4.11 Lessons Learned

Tabelle A-15: Darstellung der Lessons Learned [TIB25]

Nr.	Lessons Learned
LL1	Die IST-Analyse der Prozesse muss durchgeführt werden, da Sie beim Aufdecken von Inkonsistenzen in vorliegenden Prozessbeschreibungen hilft.
LL2	Die Detaillierungstiefe bei der Abbildung der ECM Prozesse und des PEPs ist abhängig von aktuellem Stand des Unternehmens und vom Ziel zu wählen
LL3	Das Wissen über existierende Partialmodelle in einem Unternehmen ist Grundvoraussetzung, um Potenzial der Wirkkettenanalysen abzuschätzen.
LL4	Das Benchmarking mit anderen Unternehmen hilft beim Einschätzen des eigenen Unternehmens im deutschlandweiten Vergleich
LL5	Je besser die Prozesse etabliert sind, desto besser sind die Lösungen anzuwenden
LL6	Schlechte Prozesse (z.B. schlecht dokumentiert oder nicht gelebt) können frühzeitige Fallstricke darstellen
LL7	Die Aktivitäten, die durch die Wirkkettenanalyse unterstützt werden, müssen bereits zum Beginn klar sein
LL8	Je höher die Genauigkeit bei der Abbildung der Prozesse, desto einfach ist die Anwendung der Lösungen
LL9	Ausnahmen im Prozess müssen klar sein und berücksichtigt werden können
LL10	„Multiplier-Rollen“ (Multiplikator) sind im Unternehmen notwendig, die das Thema vertreten und verbreiten
LL11	Die Nutzung von Modellierungsregeln ist hilfreich
LL12	Externe Experten können zum Einstieg hilfreich sein
LL13	Die Qualität der eingehenden Informationen ist ein relevanter Faktor für die Aussagefähigkeit des Modells und somit auch der Auswirkungsanalysen
LL14	Der Zugriff und die Verfügbarkeit von Daten und Modelle muss sichergestellt sein
LL15	Die zu beantwortenden Fragestellungen müssen frühzeitig klar sein, damit man weiß, was modelliert werden muss
LL16	Das Mapping der Unternehmenssprache (domänenspezifischen Begriffen) auf das Informationsmodell ist die Grundvoraussetzung zum Start der Modellierung
LL17	Eine klare Vision für ein Ergebnis der Auswirkungsanalyse bildet die Grundlage für effiziente Modellierung
LL18	Die Abhängigkeit zwischen Informationsmodell und Auswirkungsanalyse muss beachtet werden, da diese iterativ wachsen können
LL19	IT-Werkzeuge und Methodiken müssen bekannt sein, um Informationsmodell und Analyse aufzubauen
LL20	Gutbeispiele (Referenzimplementierungen) dienen als Orientierungshilfe

LL21	Unternehmensübergreifende Prozesse müssen berücksichtigt werden
LL22	Rollenspezifische Auswirkungenanalyse sind notwendig, um die Informationsbedürfnisse einzelner Personen zu erfüllen
LL23	Folgende Fragen müssen beantwortet werden können: Was habe ich an Infos, welche benötige ich wirklich, wie mach ich sie für die Wirkkettenanalyse nutzbar, wie hoch ist der Aufwand?
LL24	Basisalgorithmen sind ein guter Startpunkt, um mit Auswertung starten zu können.
LL25	Gemeinsame Modellierung erhöht Verständnis über System (Struktur, Anforderung, Verhalten)
LL26	Die konkrete Vorstellung vom Ergebnis bzw. das konkrete Ziel muss frühzeitig festgelegt werden
LL27	Die Ergebnisse müssen klare den Sichten und Rollen zugeordnet werden können.
LL28	Bei der initialen Anwendung sollte man einen klaren Anwendungsfall für die Wirkkettenanalyse verfolgen
LL29	Der Rückhalt in dem Unternehmen muss frühzeitig sichergestellt werden
LL30	Die Kapazitäten im Unternehmen müssen frühzeitig sichergestellt werden
LL31	Durch die initiale Anwendung müssen sichtbare Benefits für einzelne Rollen dargestellt werden.
LL32	Die Dokumentation (z.B. als Success Story) ist wichtig und muss durchgeführt werden
LL33	Der Stand zum Thema MBSE im Unternehmen muss beachtet werden, wenn die Lösungen eingeführt werden
LL34	Das Potenzial in dem Unternehmen muss frühzeitig abgeschätzt werden (z.B. wieviel Zeit verwenden wir für technische Änderungen)

A5 Fallstudie kollaborativer 6-achsiger Roboterarm Nyiro Ned 2

A5.1 Übersicht der Fallstudie

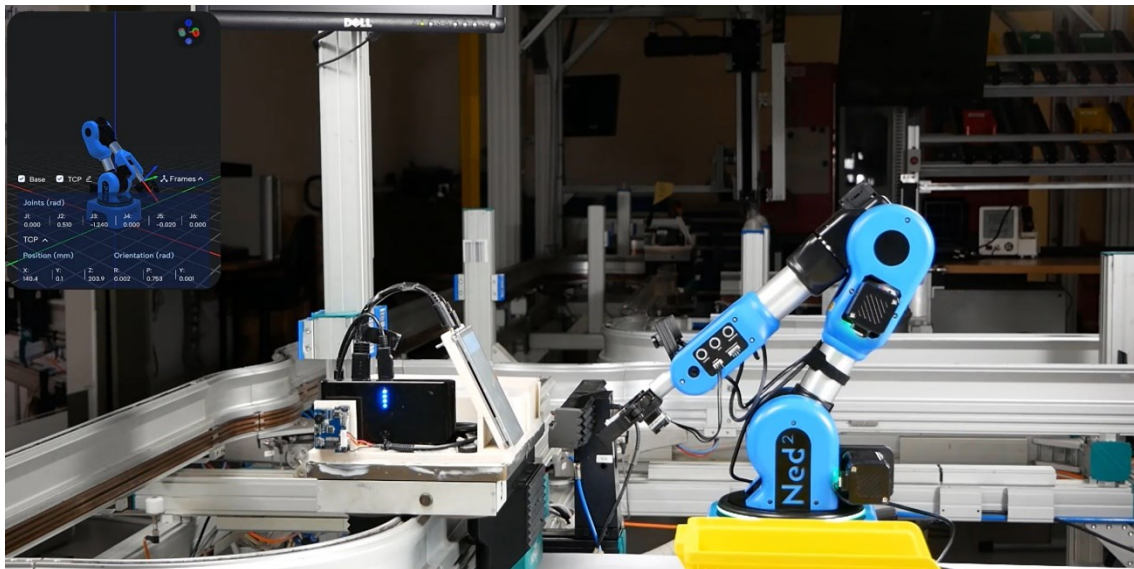


Bild A-25: Roboterarm im Smart Automation Lab

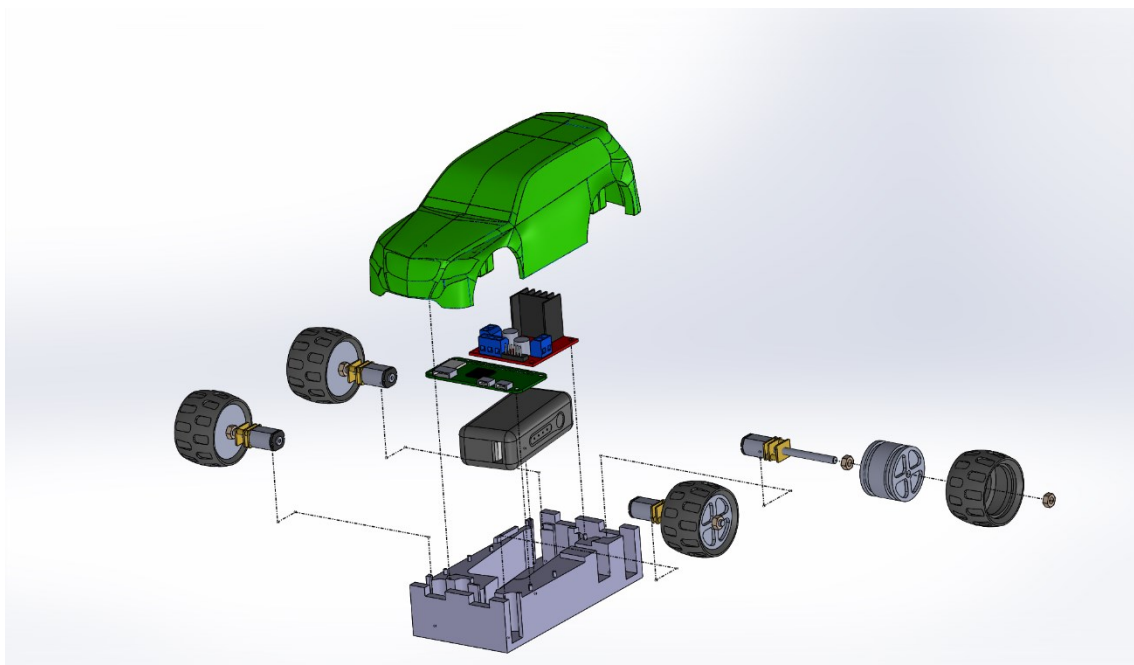


Bild A-26: Darstellung des RC-Car (Lehrstuhlhalt)

RC-Car vor Änderung mit einer Breite unter 65 mm

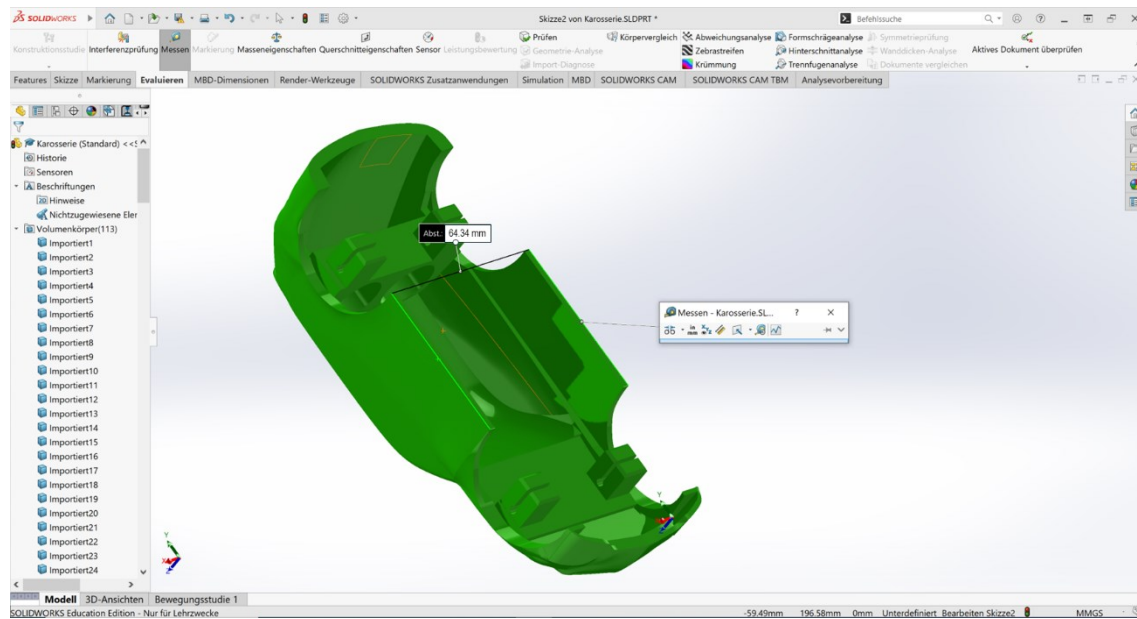


Bild A-27 : Manipulation der Karosserie in CAD-System

RC-Car nach Änderung mit einer Breite größer 70 mm

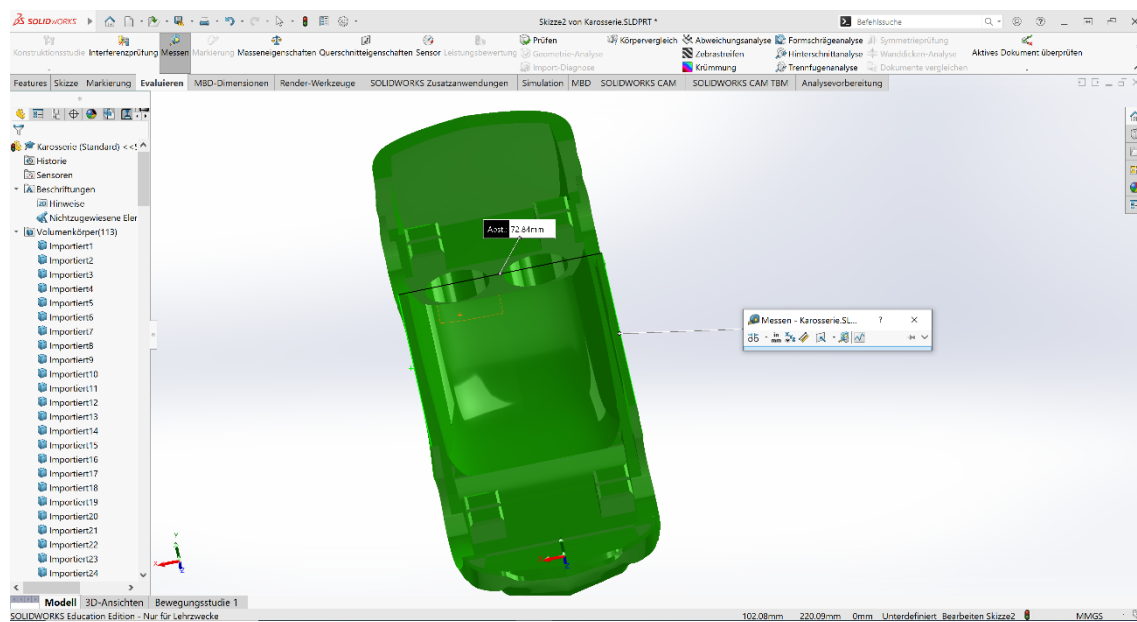


Bild A-28: Bemaßung des RC-Cars

A5.2 Modelle Roboterarm (Nyrio Ned 2)

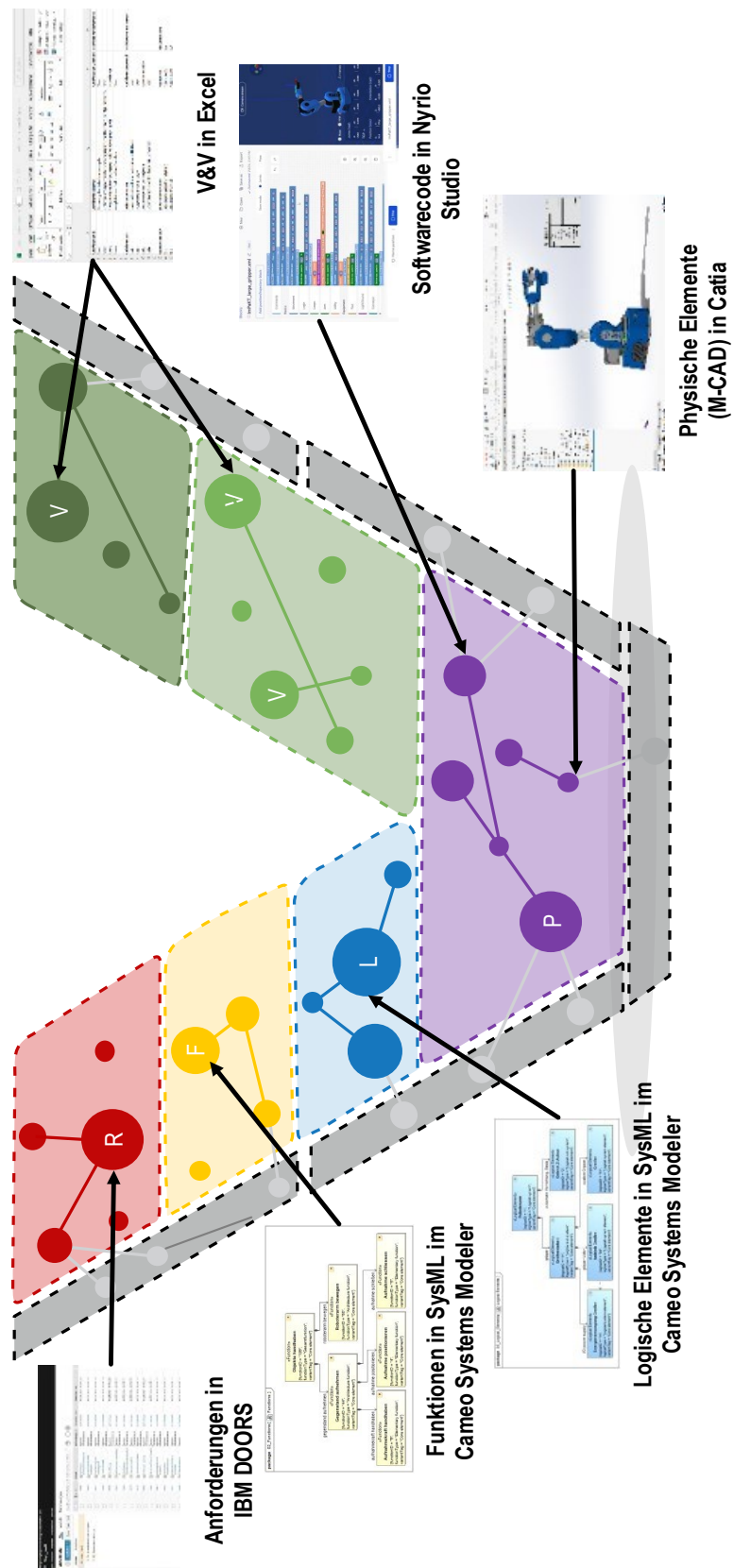


Bild A-29: Modelle entlang des V-Modells (I/II)

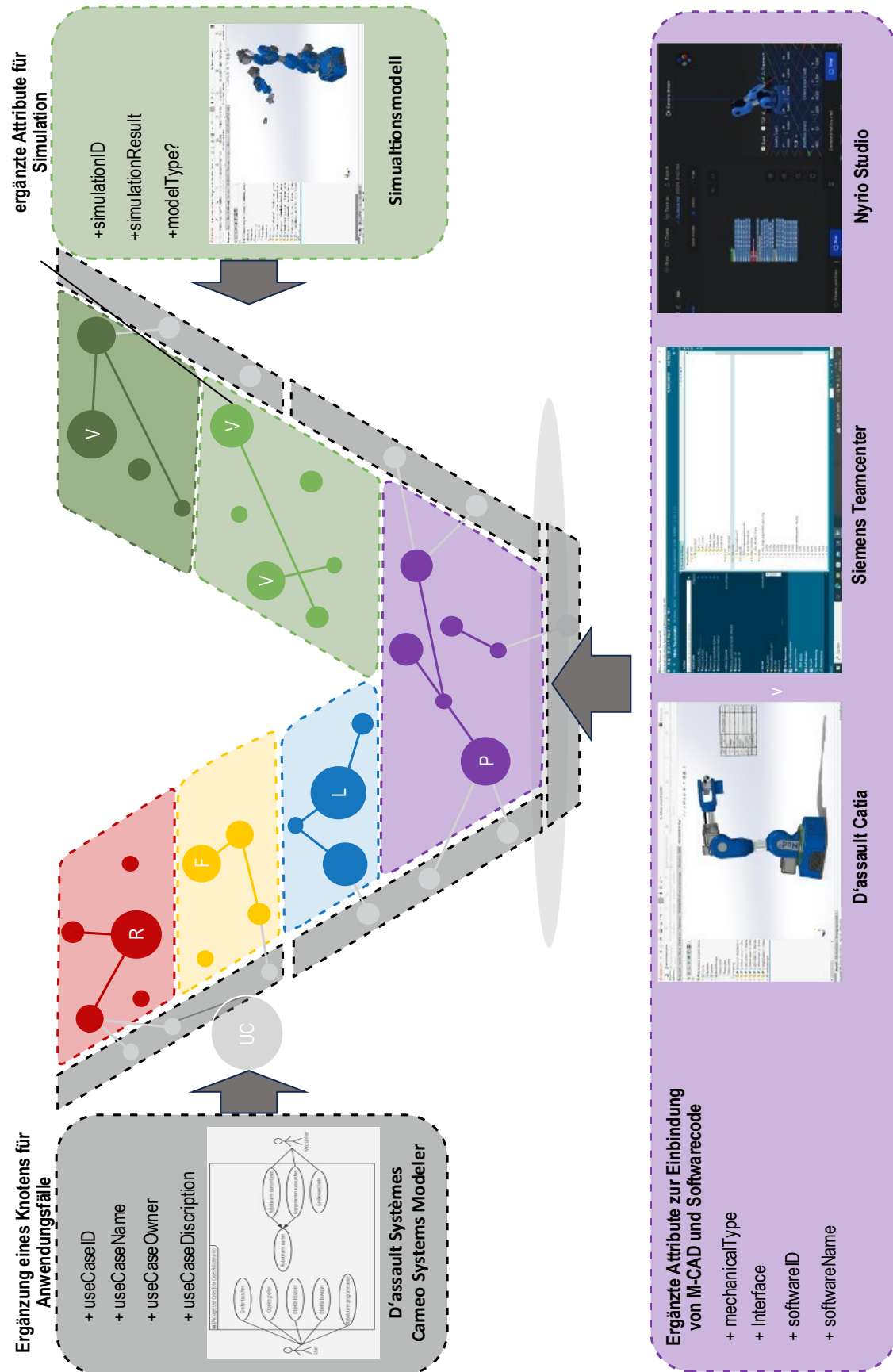


Bild A-30: Modelle entlang des V-Modells (II/II)

Anforderungen in IBM Doors

IBM Engineering Requirements Management DOORS Next (Jira)

Niryo_Ned2

Project DashboardArtefakteÜberprüfungenBerichte

Artefakte

AlleModuleSammlungen

Erstellen

Ordner Niryo_Ned2

Eingabe, um Artefakte nach Text oder ID zu filtern

OrdnerAnsichten

Niryo_Ned2

Stakeholderanforderungen

Systemanforderungen

	ID	Name	Artefakttyp	Geändert von	Geändert am
	9996	Sperrung Bedienoberfläche	System Requirement	adminhue	03.01.2024 11:17:23
	9988	Geruchsneutrale Bauteile	System Requirement	adminhue	03.01.2024 11:17:23
	9987	Validierung und Übergabe	System Requirement	adminhue	03.01.2024 11:17:23
	9986	Schichtdicke	System Requirement	adminhue	03.01.2024 11:17:23
	9985	Kühlung	System Requirement	adminhue	03.01.2024 11:17:22
	9984	Achsgeschwindigkeit	System Requirement	adminhue	03.01.2024 11:17:22
	9983	Stufenlose Rotationsbewegungen	System Requirement	adminhue	03.01.2024 11:17:22
	9979	Transformation	System Requirement	adminhue	03.01.2024 11:17:22
	9978	Systemarchitektur	System Requirement	adminhue	03.01.2024 11:17:22
	9965	Software-Schutz	System Requirement	adminhue	03.01.2024 11:17:21
	9964	Werkstoff Lagerkugeln	System Requirement	adminhue	03.01.2024 11:17:21
	9963	Bewegung des Greifers	System Requirement	adminhue	03.01.2024 11:17:21
	9958	Werkstoff Greiferfedern	System Requirement	adminhue	03.01.2024 11:17:21
	9957	Datenaustausch Ethernet	System Requirement	adminhue	03.01.2024 11:17:21
	9956	Bauteiltoleranz	System	adminhue	03.01.2024 11:17:21

Bild A-31: Anforderungen an den Roboterarm in IBM Doors

Funktionen in SysML

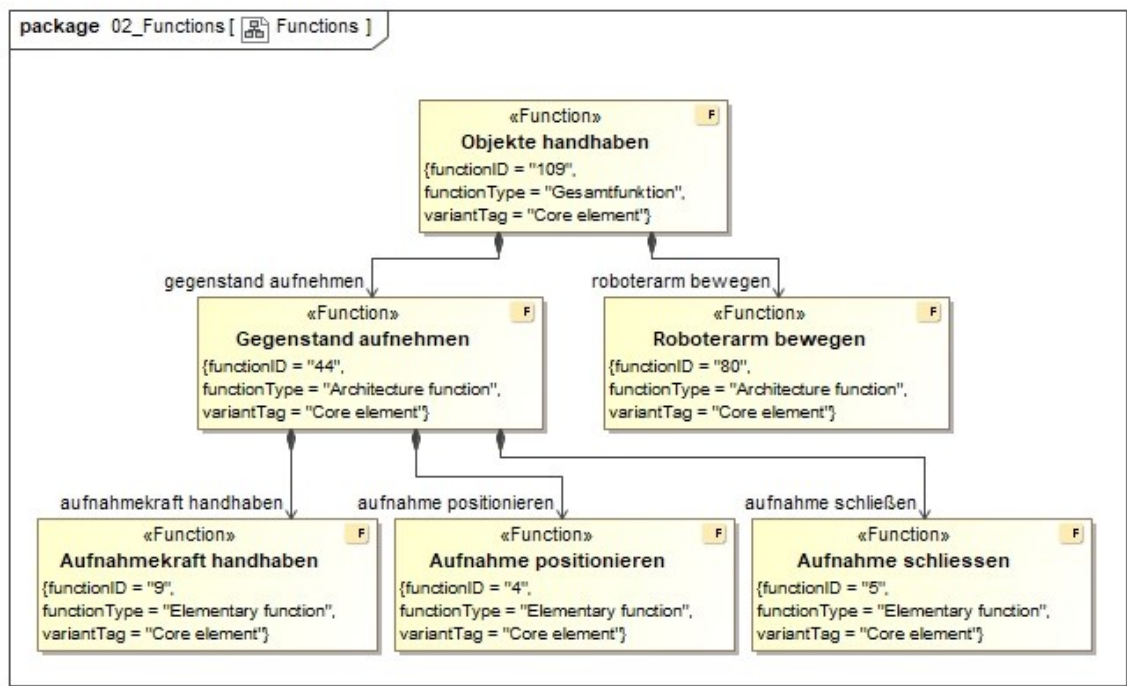


Bild A-32: Darstellung der Funktionen des Roboterarms

Simulink Modell

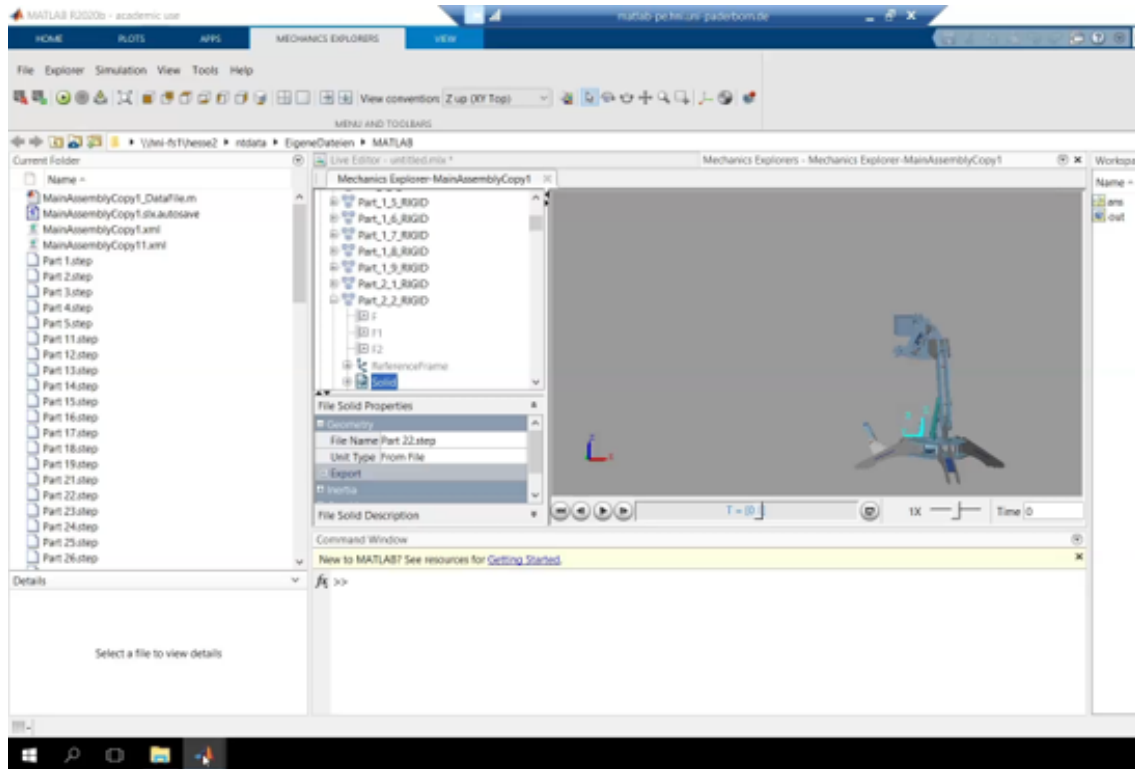


Bild A-33: Darstellung der Simulation des Roboterarms

Logische Elemente in SysML

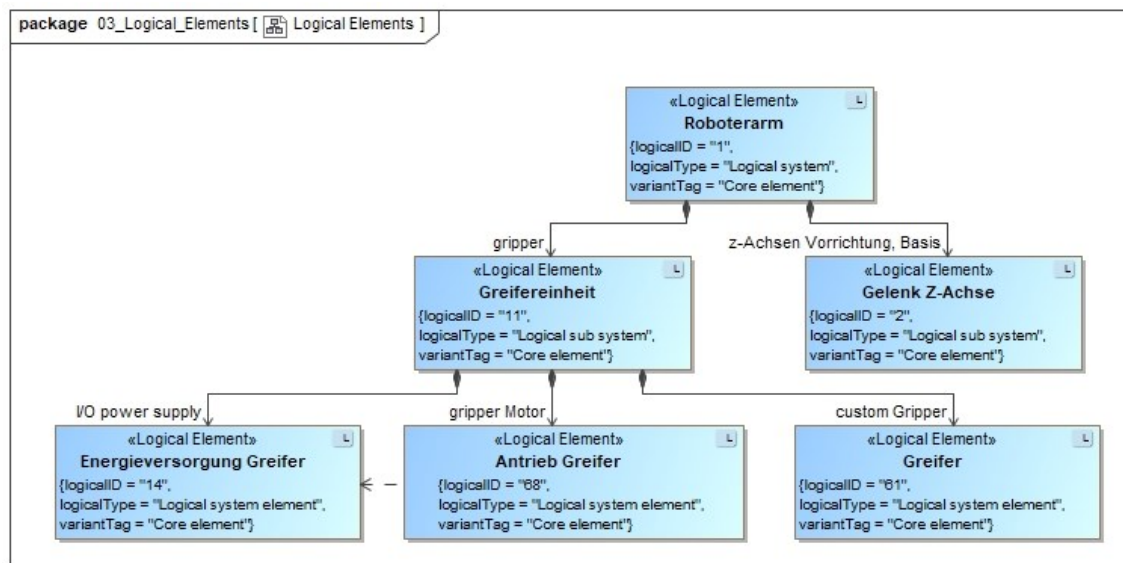


Bild A-34: Darstellung der logischen Elemente des Roboterarms

Modell in Niryo App

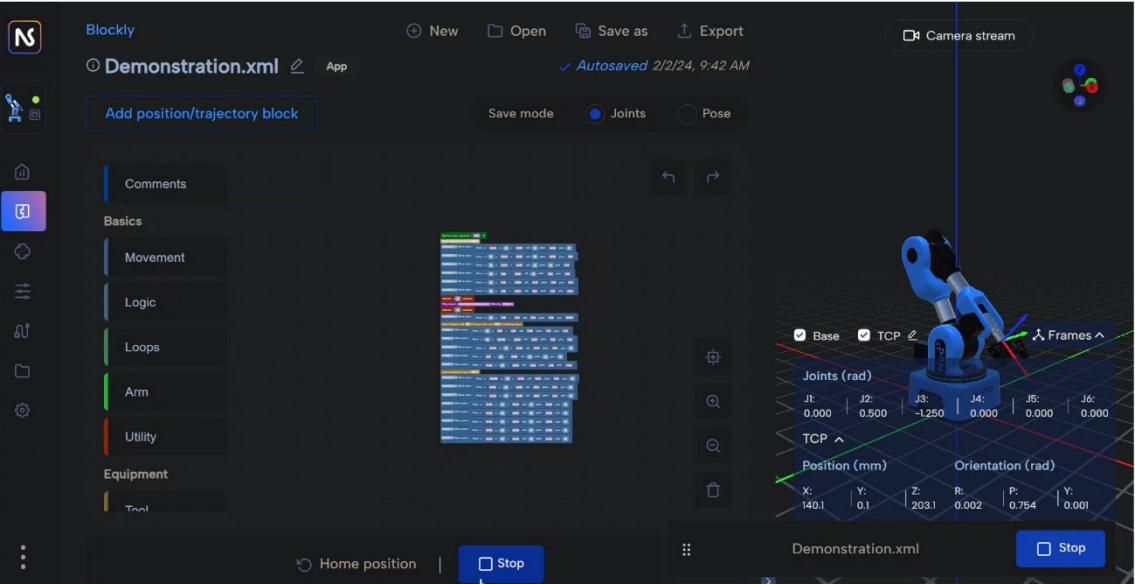


Bild A-35: Modell des Roboterarms in Niro

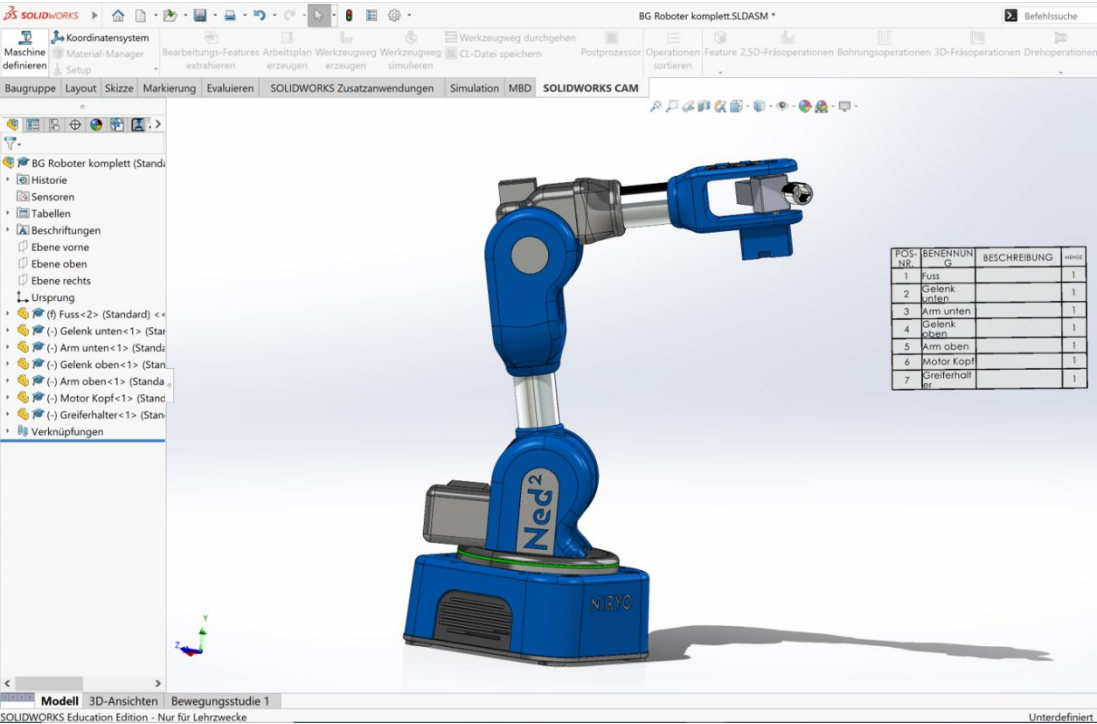


Bild A-36: Darstellung des CAD Modells

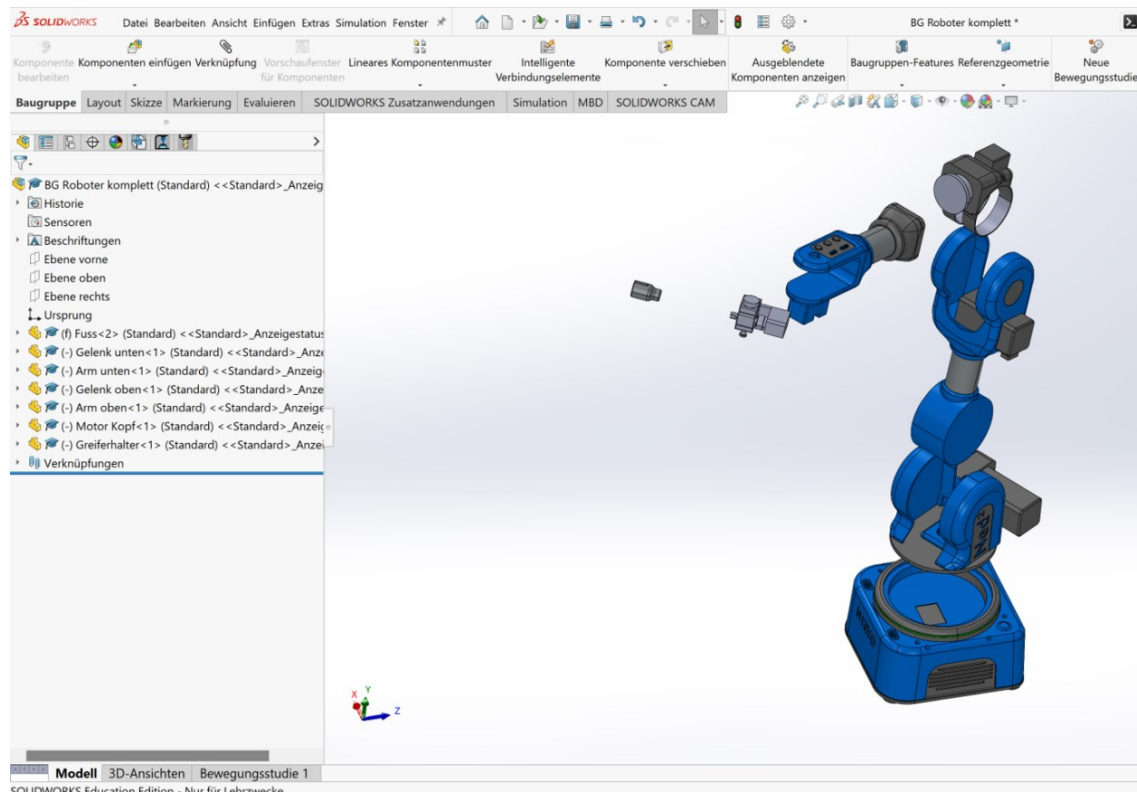


Bild A-37: Schnittstellen zur Integration und Montageplanung

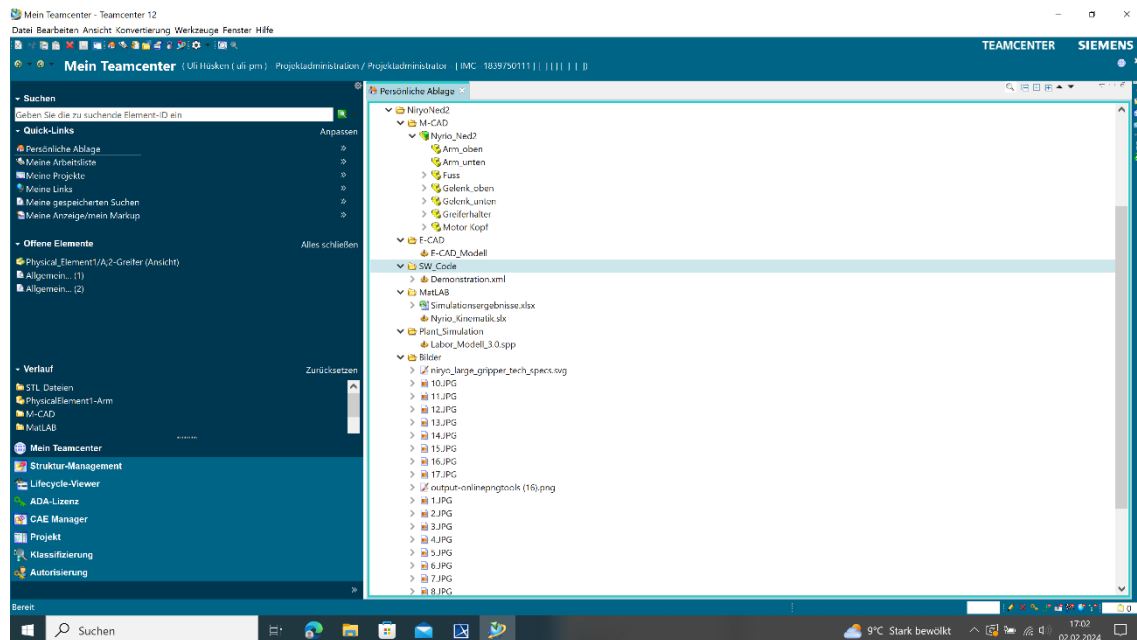


Bild A-38: Teamcenter Darstellung der M-CAD Modelle

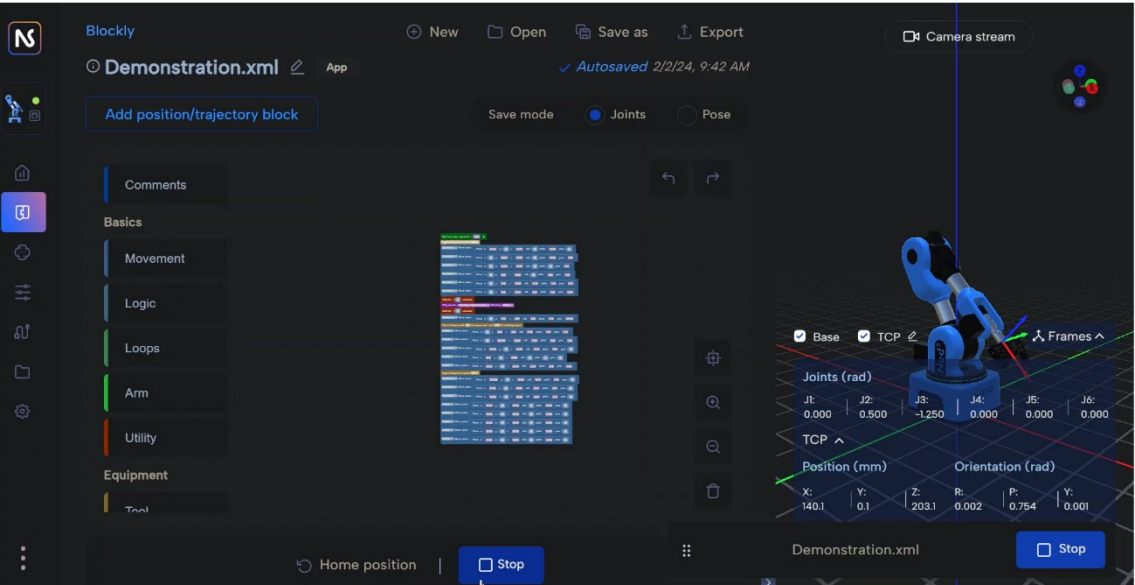


Bild A-39: Darstellung des Softwarecodes in Blockly

Automatisches Speichern

V&V • Auf "diesem PC" gesp...

Suchen

Datei

Start

Einfügen

Seitenlayout

Formeln

Daten

Überprüfen

Ansicht

Automatisieren

Entwicklertools

Hilfe

Einfügen

Zwischenablage

Calibri

11

A⁺

A⁻

F

K

U

Bild A-40: Verifikation und Validierung in Excel

A5.3 Greifvorgang bei Pick-and-Place-Bewegung



Bild A-41: Darstellung des Greifers (alt)

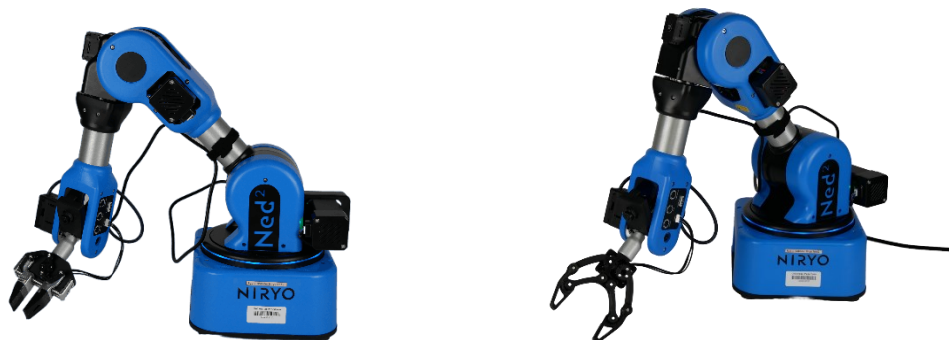


Bild A-42: Darstellung des alten Greifers (links) und neuen Greifers (rechts)



Bild A-43: Pick-and-Place-Bewegung des RC-Cars mit Greifer 2

A5.4 Neo4j Import Roboterarm

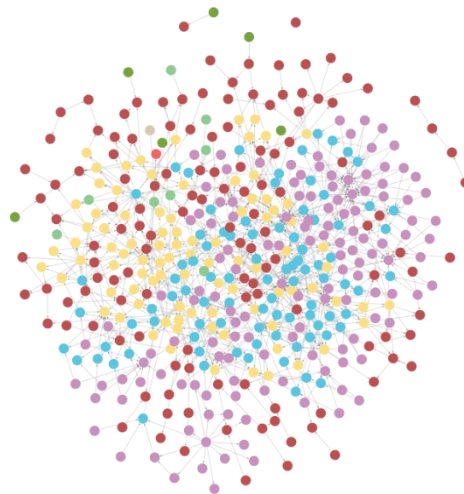


Bild A-44: Darstellung des Roboterarms im Graphen

Tabelle A-16: Inhalt des Graphen des Roboterarms

Zählung der Relationen je Relationstyp	
Derives	75
Requires	27
Refines	12
affected_by	1
Satisfies	147
Decompose	349
Specifies	193
Realizes	125
Verifies	8
Validates	6
extends	1
Summe	944
NodeLabels,Count	
Requirement	134
Function	91
LogicalElement	93
PhysicalElement	158
Verification	9
Validation	7

EngineeringChangeRequest	10
UseCase	8
Summe	

A5.5 Dashboards Roboterarm

Dashboards Roboterarm

FachexperteProjektmanagementVertriebAnforderungsentwicklerSystemarchitektmechanischer KonstrukteurBasisalgorithmen

betroffene Baugruppe

PhysicalElement physicalName
Standardgreifer

Verantwortliche Person

Max Mechaniker

alternative Baugruppe

PhysicalElement physicalName
Grosser Greifer

Verantwortliche Person

Deniz Özcan

Greifer 1

Element	Bereich	Typ	Kosten
Greiferbacken	Mechanic	System element	5
Greiferkabel	Electronic / Electric	System element	10
Greiferrumpf	Mechanic	System element	30
Backenaufnahme	Mechanic	System element	5
Fuehrungsschiene	Mechanic	System element	10

1-5 of 12 < >

Kosten Greifer 1

85.8

Kostenträger

30

Greifer 2

Element	Bereich	Typ	Kosten
XL330 Servomoto	Electronic / Electric	System element	30
Greiferkabel	Electronic / Electric	System Element	10
Gelenkarm mit Rit	Mechanic	System Element	20
Magnet	Mechanic	System Element	15
Greiferrumpf	Mechanic	System element	50

1-5 of 9 < >

Kosten Greifer 2

95.8

Kostenträger

50

Bild A-45: Dashboard Roboterarm (I/V)

Dashboards Roboterarm

FachexperteProjektmanagementVertriebAnforderungsentwicklerSystemarchitektmechanischer KonstrukteurBasisalgorithmen

betroffene Baugruppe

PhysicalElement physicalName
Standardgreifer

Verantwortliche Person

Max Mechaniker

alternative Baugruppe

PhysicalElement physicalName
Grosser Greifer

Verantwortliche Person

Deniz Özcan

Greifer 1

Element	Bereich	Typ	Kosten
Greiferbacken	Mechanic	System element	5
Greiferkabel	Electronic / Electric	System element	10
Greiferrumpf	Mechanic	System element	30
Backenaufnahme	Mechanic	System element	5
Fuehrungsschiene	Mechanic	System element	10

1-5 of 12 < >

Kosten Greifer 1

85.8

Kostenträger

30

Greifer 2

Element	Bereich	Typ	Kosten
XL330 Servomoto	Electronic / Electric	System element	30
Greiferkabel	Electronic / Electric	System Element	10
Gelenkarm mit Rit	Mechanic	System Element	20
Magnet	Mechanic	System Element	15
Greiferrumpf	Mechanic	System element	50

1-5 of 9 < >

Kosten Greifer 2

95.8

Kostenträger

50

Bild A-46: Dashboard Roboterarm (II/V)

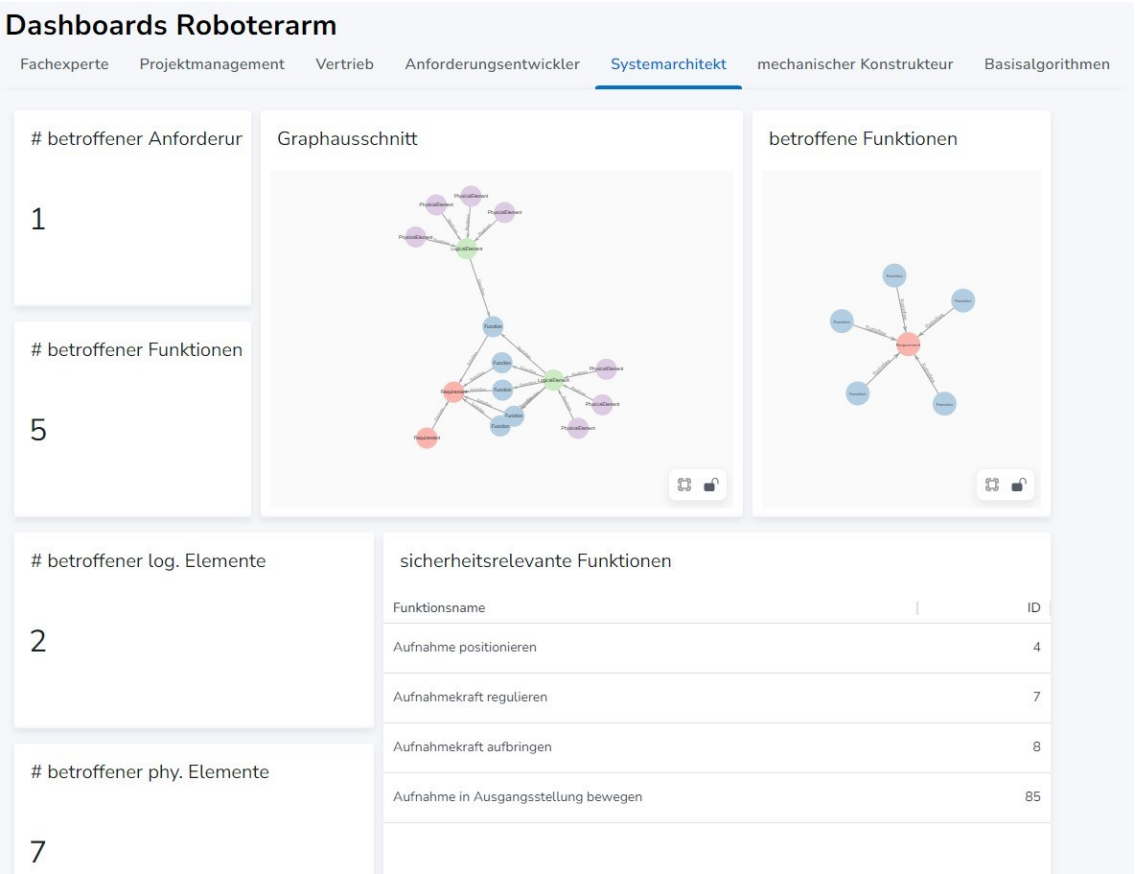


Bild A-47: Dashboard Roboterarm (III/V)

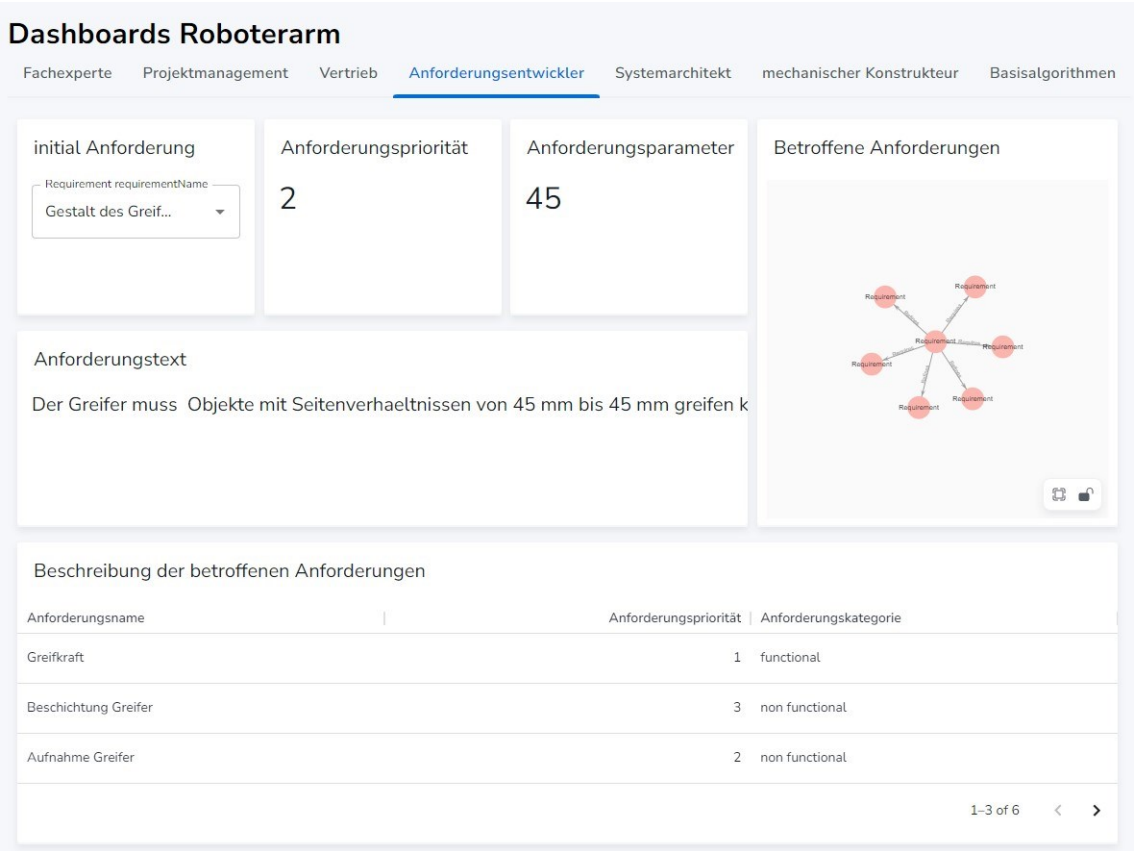


Bild A-48: Dashboard Roboterarm (IV/V)

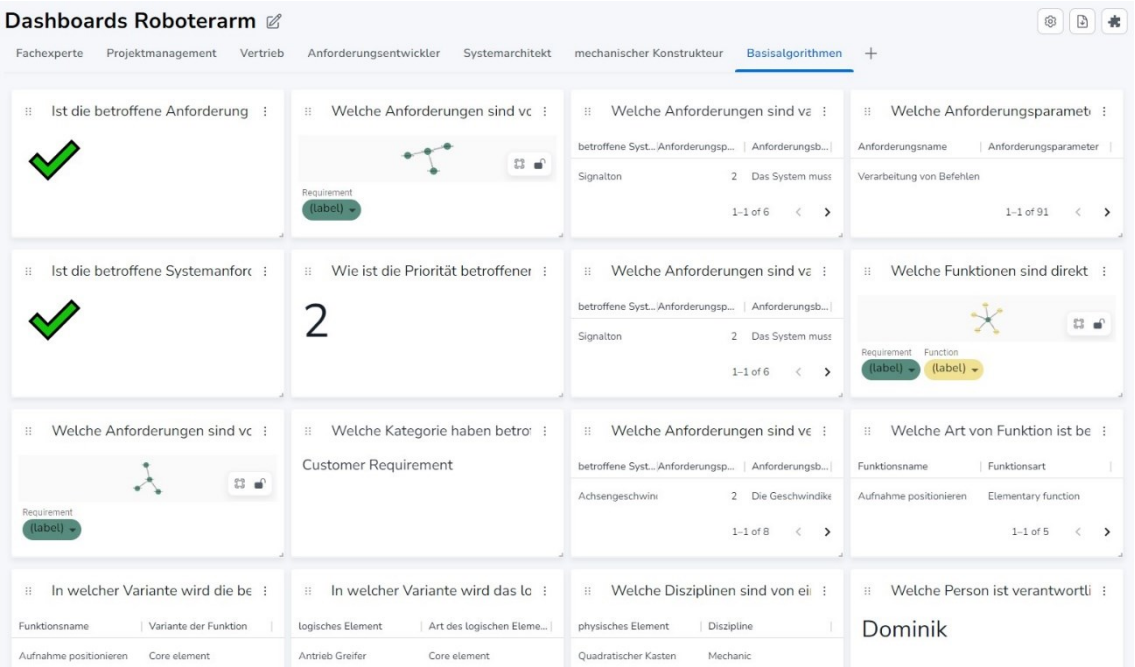


Bild A-49: Dashboard Roboterarm (V/V)

A6 Zusatzinformationen ImPaKT

A6.1 Übersichtsposter Fallbeispiele

Hofmann.

Intelligent Balancing Solutions

Im Fallbeispiel der Firma HOFMANN Mess- und Auswuchttechnik GmbH & Co. KG werden technische Änderungen in der Entwicklung von Auswuchtmaschinen betrachtet. Die Auswuchtmaschinen werden von Unternehmen unterschiedlicher Branchen benötigt und in Produktionslinien integriert. Die Mess- und Bearbeitungsfunktionen werden durch mechanische, elektronische und softwaretechnische Komponenten realisiert. Grundlagen der Entwicklung sind ein modularer Baukasten und kundenindividuelle Anpassungen. Die Anforderungen basieren unter anderem auf dem auszuwuchtenden Rotor, dem Fabriklayout und Datenschnittstellen zu Leitrechnern.

Hofmann Mess- & Auswuchttechnik



CLAAS Industrietechnik & Eisengießerei Baumgarten



CLAAS

TRADITION. PERFEKTION. INNOVATION.
EISENGIEßEREI BAUMGARTE



Im Fallbeispiel CLAAS Industrietechnik GmbH und Eisengießerei Baumgarten werden die unternehmensübergreifenden Abhängigkeiten eines mechatronischen Raupenlaufwerks analysiert. Das Raupenlaufwerk wird in verschiedenen Landmaschinen wie Mähdreschern, Traktoren und Feldhäckslern verbaut. Bei der Entwicklung muss Claas Industrietechnik zum einen Anforderungen und Restriktionen der jeweiligen Landmaschinen berücksichtigen, zum anderen aber auch Restriktionen der eigenen Zulieferer. Die Eisengießerei Baumgarten produziert unter anderem die Schwinde und das Stützradpendel des Raupenlaufwerks.

Hadi-Plast

Kunststoff-Verarbeitung

Im dritten Anwendungsbeispiel der Hadi-Plast GmbH & Co. KG liegt der Fokus auf der Belieferung verschiedener Branchen mit zertifizierten Präzisions-Spritzgussteilen aus thermoplastischen Kunststoffen. Beispiele sind Bauteile aus den Bereichen Automotive, Retail- und Automatenysteme, Armaturen sowie Medizin- und Versorgungstechnik. Im Auftrag von OEMs und Zulieferern fertigt das Unternehmen Produkte wie ein Bauteil für Ladesteckersystem zur Energieversorgung elektrisch betriebener Fahrzeuge. In Kooperation mit den Werkzeugbauern wird die Produktion des Präzisions-Spritzgussteils gewährleistet. Technische Änderungen betreffen daher meist nicht nur das Bauteil an sich, sondern auch das Spritzgusswerkzeug und die Spritzgussmaschine.

HADI-PLAST Kunststoff-Verarbeitung

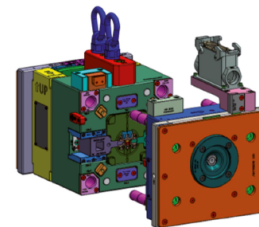


Bild A-50: Übersichtsposter der Fallbeispiele aus dem ImPaKT Projekt

A6.2 Übersicht zur Anwendung der MECA-Methodik in Fallstudien


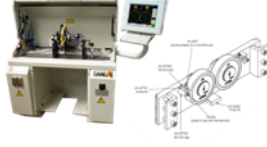
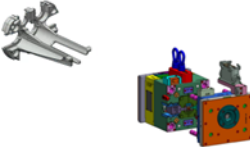
Name	Raupenlaufwerk	Auswuchtmaschine	Spritzgussartikel
Bild			
Systemgrenze (Schritt 1.1)	Betrachtung des Raupenlaufwerk mit Sub-Systemen und Systemelementen	Betrachtung der Auswuchtmaschine mit Sub-Systemen und Systemelementen	Betrachtung der Spritzgussmaschine, -werkzeug und -artikel
Ergebnis- dimensionen (Schritt 1.2)	Maßgeschneiderte Ergebnisdimensionen für die Rollen (siehe Beschreibung der Fallstudien)		
Modellierungs- intensität (Schritt 1.3)	Stufe 3		
Metadaten- modell (Schritt 1.4)	RFLPV ² +	RFLPV ² +	RFLPV ² +
Glossar (Schritt 1.5)	RFLPV ² Handouts und ergänzte unternehmensspezifische Begriffe		
Modellierungs- regeln (Schritt 1.6)	RFLPV ² Modellierungsregeln		
vorhandene Informationen (Schnitt 2.1)	Lastenheft, Stückliste, Geometriemodell, Kalkulationsmodell	Lastenhefte, Stückliste, ERP-Exporte, PLM-Exporte	Werkzeuglastenheft, Funktionslisten, Geometriemodelle, DIN-Normen, Kalkulation
erhobene Informationen (Schnitt 2.2)	Funktionen und logische Elemente		
Informations- qualität (Schnitt 2.3)	Anforderungsschablonen, Vollständigkeitscheck durch Hauptmerkmaliste, Funktionsformulierung (Sustantiv + Verb im Infinitiv)		

Bild A-51: Zusammenfassung aus den Ergebnissen aus den Fallbeispielen (I/II)

Name	Raupenlaufwerk	Auswuchtmaschine	Spritzgussartikel
Bild			
Metamodell anpassen (Schritt 3.1)	Anpassung des Metadatenmodells (siehe Anhang der Fallstudien)		
System dekomponieren (Schritt 3.2)	Bis auf parametrische Ebene der Subsysteme	Bis auf parametrische Ebene des Rollenlagers	Bis auf Komponentenebene des Werkzeugs
Wirkketten modellieren (Schritt 3.3)	197 Knoten 400 Kanten	1218 Knoten 964 Kanten	432 Knoten 431 Kanten
betreffende Elemente identifizieren (Schritt 4.1)	Darstellung der Wirkketten entlang RFLPV ² (siehe Beschreibung der Fallbeispiele)		
Ergebnisse analysieren (Schritt 4.3)	Maßgeschneiderte Dashboards für unternehmensspezifische Rollen (siehe Beschreibung der Fallstudien und Anhang der Fallstudien)		

Bild A-52: Zusammenfassung aus den Ergebnissen aus den Fallbeispielen (II/II)

A6.3 Partialmodell je Fallstudie

Tabelle A-17: Bewertung der Verfügbarkeit durch Anwender

Artefakt	Hofmann	CLAAS / EB	Hadi Plast
Anforderungen	Pflichten- und Lastenhefte (pdf.)	Teilausschnitt aus Lastenheft (pdf.)	Werkzeuglastenheft, E-Mails mit Kundenanforderungen
Funktionen	Funktionen hergeleitet in Expertenworkshops (mdzip.)	Funktionen hergeleitet in Expertenworkshops (mdzip.)	Funktionen hergeleitet in Expertenworkshops (mdzip.)

Logische Elemente	Logische Elemente hergeleitet aus Funktionen und physischen Elementen in Expertenworkshops (mdzip.)	Logische Elemente hergeleitet aus Funktionen und physischen Elementen in Expertenworkshops (mdzip.)	Logische Elemente hergeleitet aus Funktionen und physischen Elementen in Expertenworkshops (mdzip.)
Physische Elemente	<ul style="list-style-type: none"> • mBOM (m2-m4) als csv. Export • Softwarecode zur Berechnung des Einlagerungswinkels (python.) • CAD-Dateien (step.) • Standardkomponentenliste (csv.) 	<ul style="list-style-type: none"> • Stückliste (Catia) • Geometriemodelle des Pendels (step.) 	Geometriemodell Werkzeug (step.) Geometriemodell Artikel (step.)
Verifikation und Validierung			<ul style="list-style-type: none"> • Toleranzen nach ISO 20457 • Erstmusterprüfbericht
Weitere Modelle und Dokumente	<ul style="list-style-type: none"> • Mathcad-Modell zur Berechnung der Hertz'schen Pressung • Materialliste mit Streckgrenzen 	Kalkulation (Excel)	Kalkulation (Excel) Maschinenliste (Excel) Peripherieliste (Excel)

Tabelle A-18: Auflistung der Werte

	Ja	Nein
Anforderungen (Lastenheft, Pflichtenheft, Spezifikation, ...)	100 %	0 %
Funktionen (Funktionsliste, Funktionshierarchie, ...)	50 %	50 %
Logische Elemente (Vergleich von Lösungsalternativen, ...)	50 %	50 %
Physische Elemente (eBOM, mBOM, ...)	100 %	0 %
Disziplinspezifische Modelle (M-CAD, E-CAD, CASE, ...)	100 %	0 %
Kostenkalkulationen	100 %	0 %
Änderungsanfragen (Engineering Change Request) von internen oder externen Stakeholdern	75 %	25 %
Bewertungskriterien zur Abschätzung von Auswirkungen einer technischen Änderung (z.B. Schwellwerte)	25 %	75 %
Kategorie-übergreifender Durchschnitt der Antwort	75 %	25 %
Durchschnittliche Antwort	75%	25 %

A7 Fallstudie CLAAS und Eisengiesserei Baumgarte

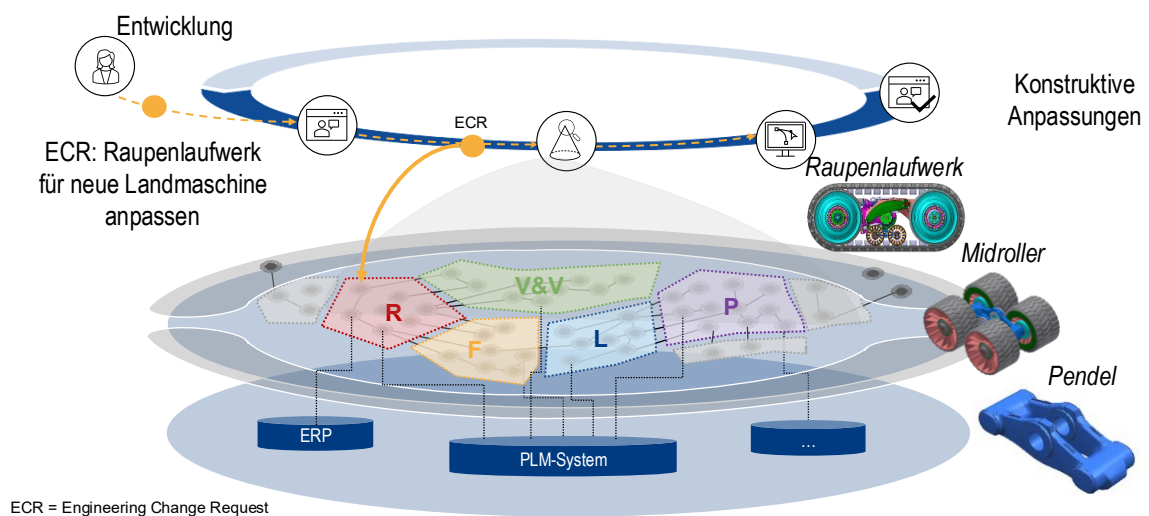


Bild A-53: Big Picture der Fallstudie CLAAS und Eisengiesserei Baumgarte

BPMN-Diagramme CLAAS Industrietechnik

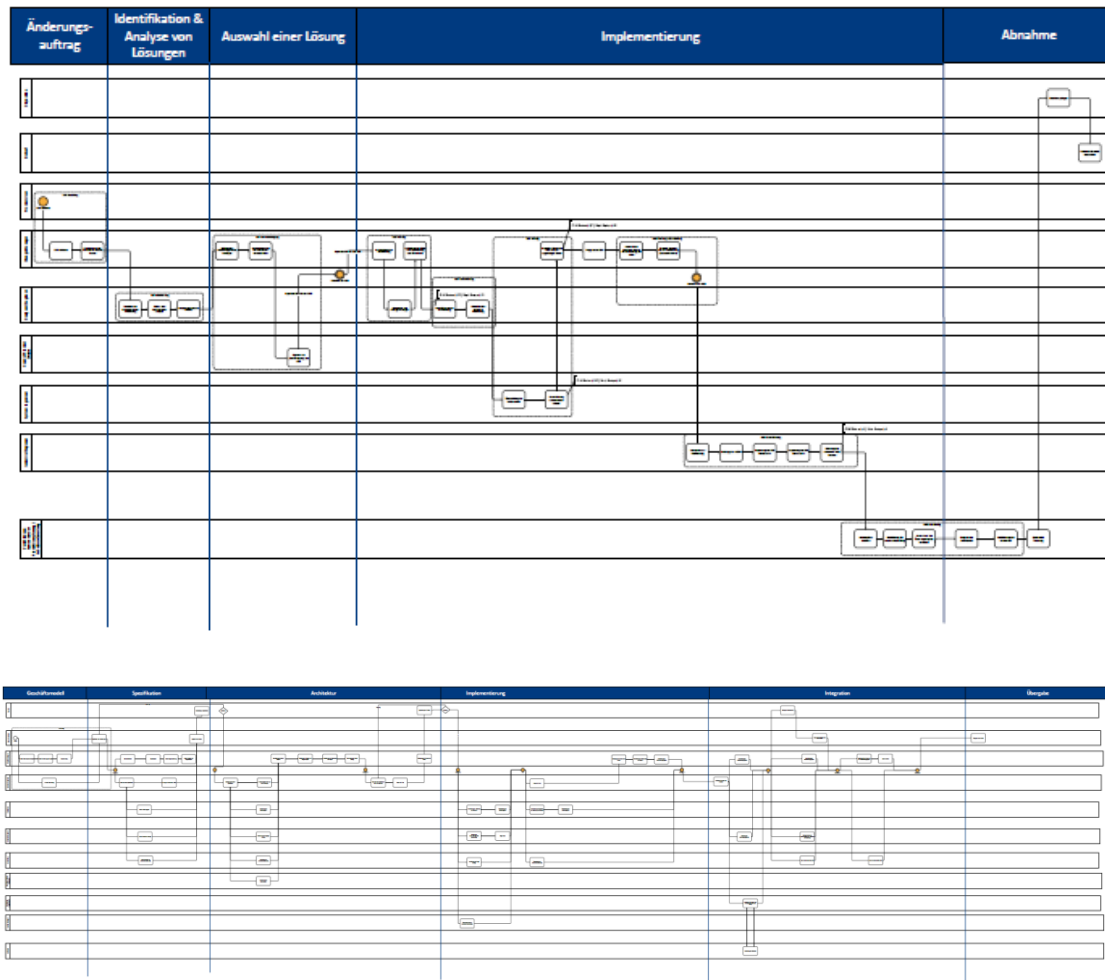


Bild A-54: BPMN Diagramme CLAAS

Historische technische Änderungen CLAAS/EB

Änderungsbedarf: „Das Gussteil erfüllt den Zielpreis nicht, wodurch eine geometrische Änderung definiert werden muss, die die Herstellkosten reduziert“

Auswirkungen der Änderung:

- In der Entwicklung müssen Lösungsalternativen monetär vergleichen, damit die eine kostengünstigere Gussteil entsteht
- Berücksichtigung von Kosten durch z.B. Prozessschritte wie *Schleifen*

Besonderheiten:

- Herstellkosten sind ausschlaggebend
- Die technische Machbarkeit muss weiterhin gewährleistet sein



Bild A-55: Historische Technische Änderung CLAAS/EB

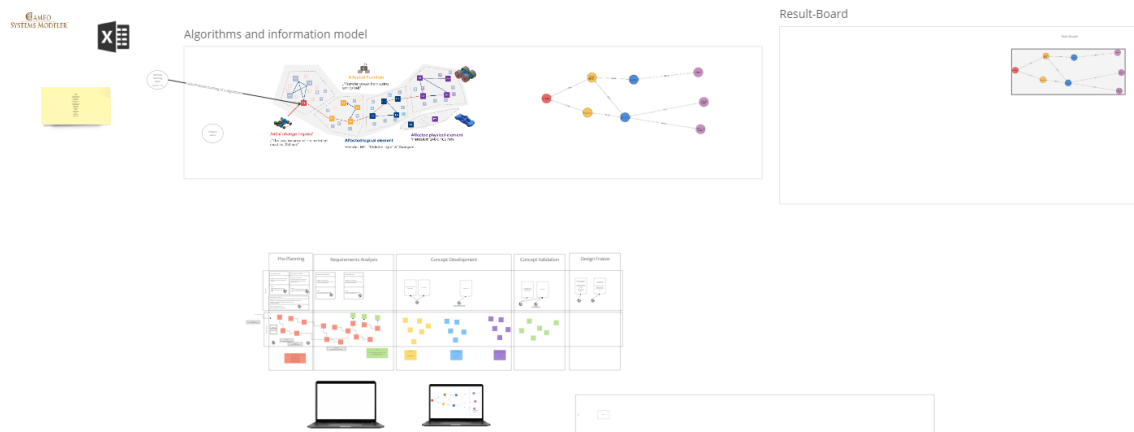


Bild A-56: Dokumentation des Modellierungsworkshops CLAAS / EB

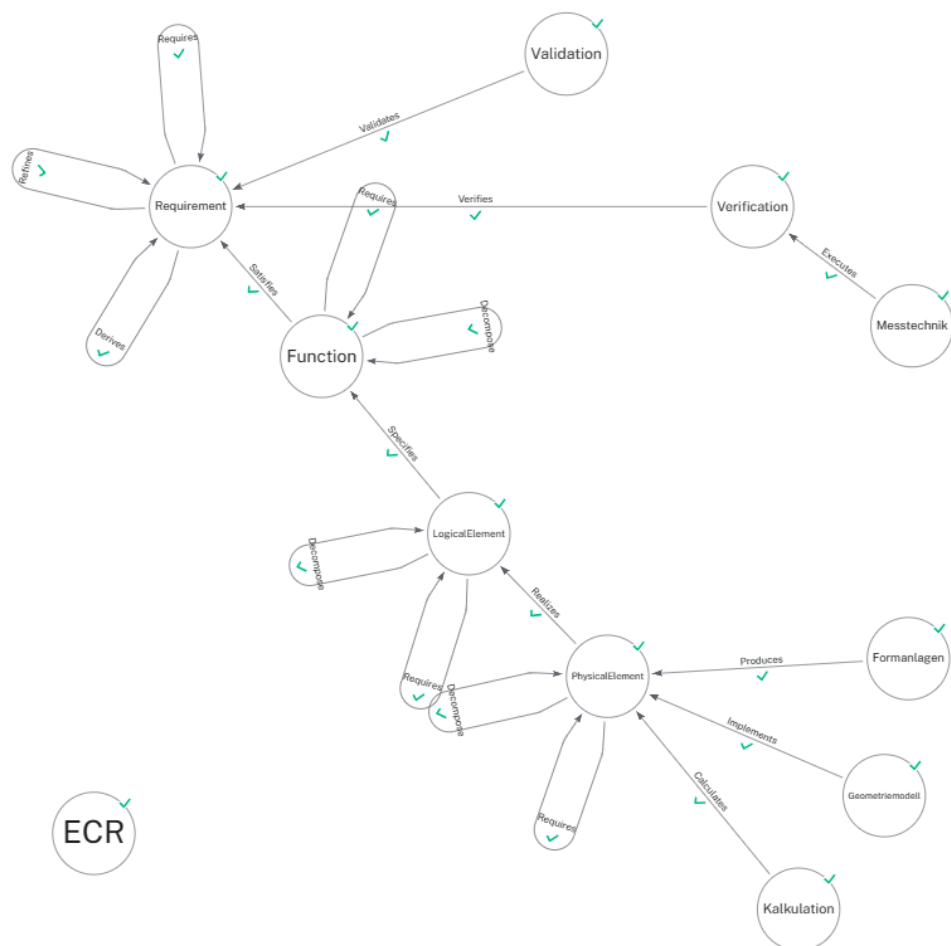


Bild A-57: Neo4j Import CLAAS/EB



Bild A-58: Darstellung im Graphen

Dashboards CLAAS/EB

Betroffene Komponente PhysicalElement physicalName Stuetzradpendel XY	Engineering Change ECR ecrID Start typing...
Jahresbedarf Variante 1 Jahresbedarf_Variante1 200	Jahresbedarf Variante 2 Jahresbedarf_Variante2 2500
Gesamtkosten Variante 92.0 €	Gesamtkosten Variante 104.55 €

Bild A-59: Dashboard CLAAS Einkauf

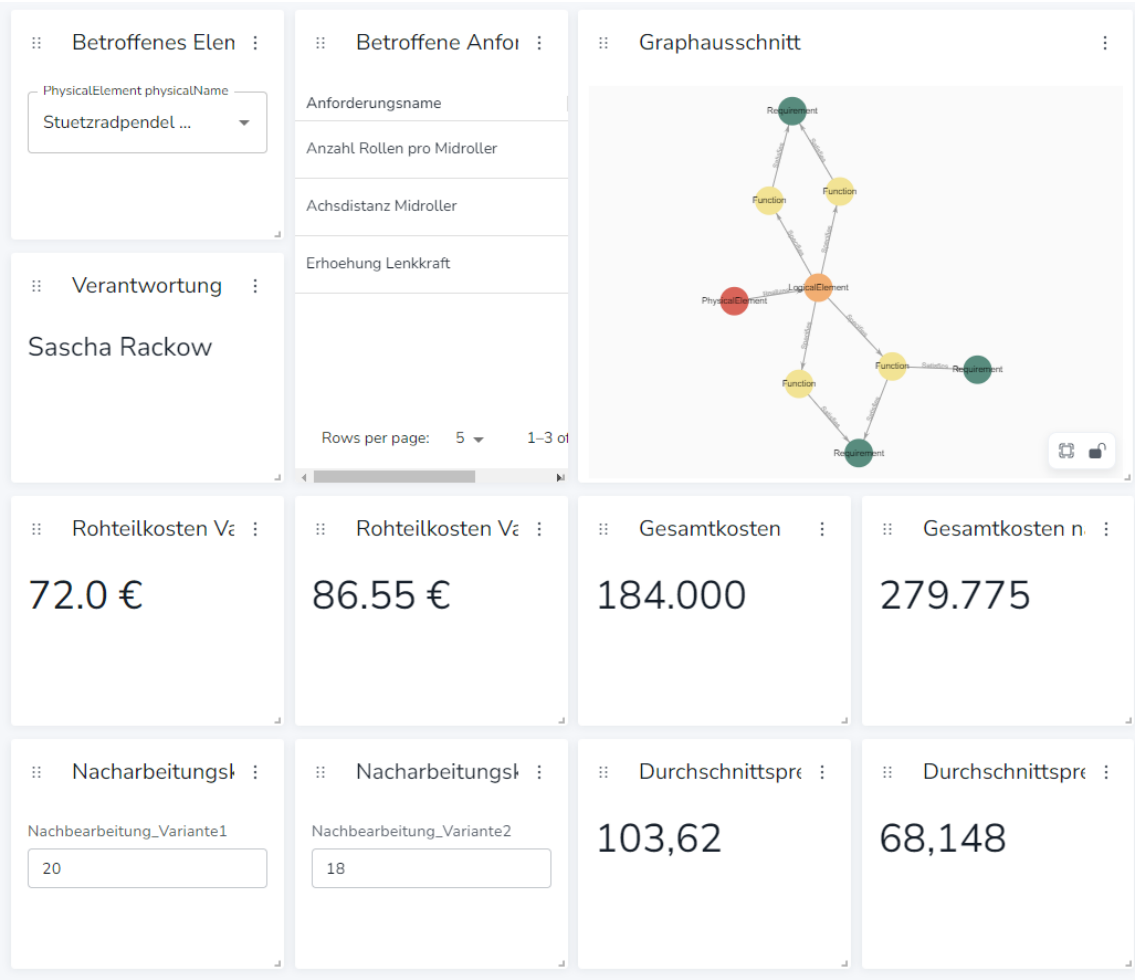


Bild A-60: Dashboard CLAAS Entwicklung

Messverfahren

Messtechnik Messverfahren

Kerbschlaguntersuchung

Geometriemodell

Geometriemodell ID

1

Betroffene Elemente

Welche Messart wurde benutzt?

Messart	Messverfahren	Zertifikationsstufe	Level	Maschine
Zerstoeerende Pruefur	Kerbschlaguntersuch	1	1	ZwickRoell ASTM E2

Wandstärke des Pendel

10 mm

Geometrieelemente

3 Kerne

Geometrische Toleranz

(+/-) 0.5 mm

Materialauswahl

GJS-600-3

Dichte

2700 kg/m³

Bruchdehnung

0.03 [%]

Zugfestigkeit

600 [N/mm²]

Streckgrenze

370 [N/mm²]

Bild A-61: Dashboard EB_Messtechnik

Unternehmen

PhysicalElement Company

Claas Industrie Technik

Betroffenes Element

PhysicalElement physicalName

Stuetzradpendel XY

Jahresbedarfe Variante

200

Jahresbedarf Variante 2

2.500

Kosten Variante 1

72.0 €

Kosten Variante 2

86.55 €

Bild A-62: Dashboard: EB_Technische Kalkulation

Unternehmen PhysicalElement Company Claas Industrie Technik	Betroffenes Element PhysicalElement physicalName Stuetzradpendel XY
Jahresbedarfe Variante 200	Jahresbedarf Variante 2 2.500
Kosten Variante 1 72.0 €	Kosten Variante 2 86.55 €

Bild A-63: Dashboard EB Vertrieb

Pendelart Kalkulation Kalkulationsname Pendel Fertigteil_2	Schmelzen 9.49 €	Hilfsstoffe 8.18 €
Kern Kern 4	Putzen Putzen 7.40	Formen 7.65 €
Metall Metall 28.5	sonstige Kosten sonstige Kosten 18	Gesamtkosten 86.54 €

Bild A-64: Dashboard zu den Kosten

A8 Fallstudie Hofmann

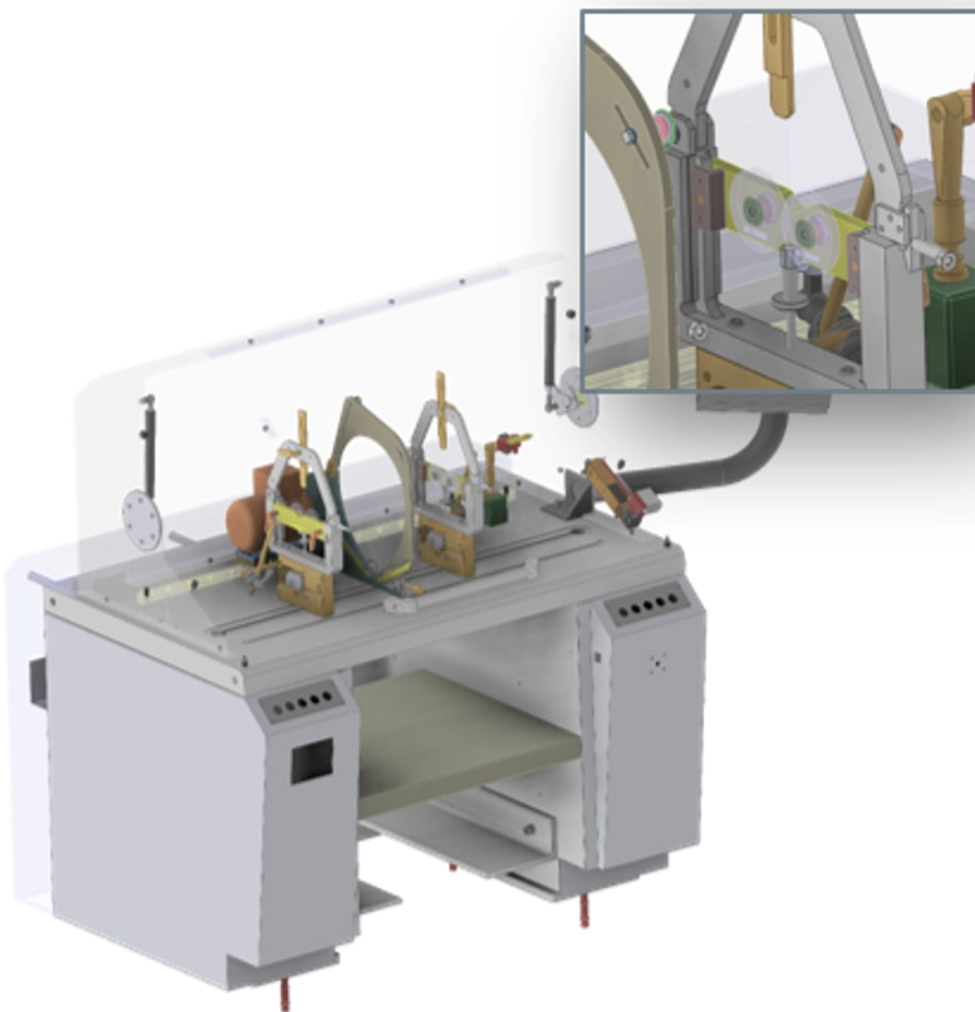


Bild A-65: CAD-Darstellung der Auswuchtmaschine

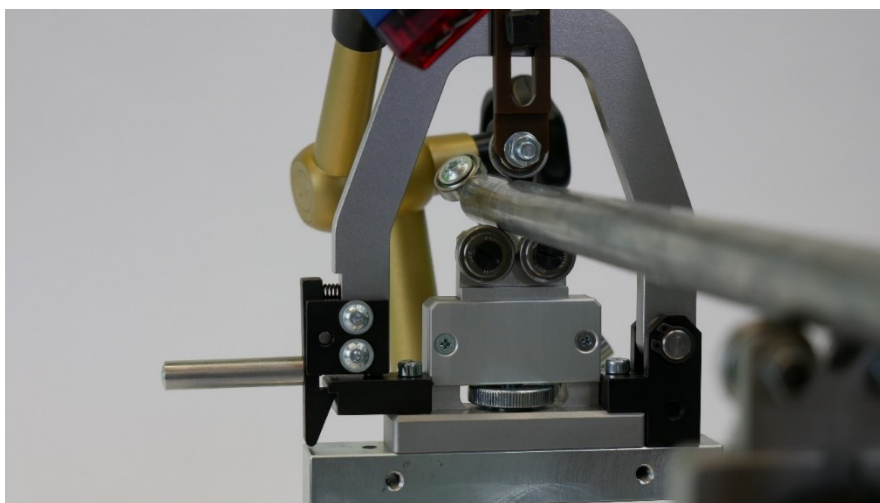


Bild A-66: Aufliegender Rotor auf einem Rollenlager

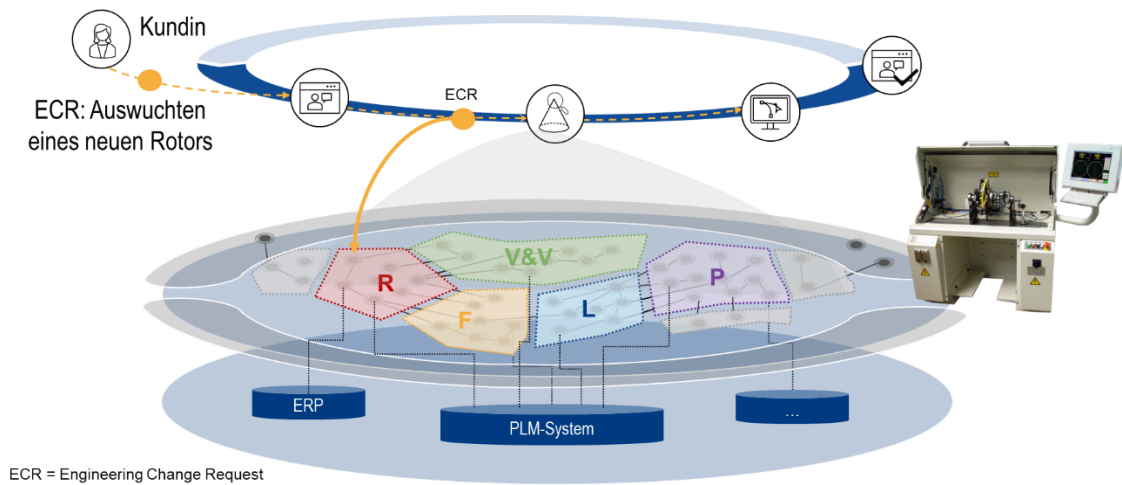


Bild A-67: Darstellung Fallbeispiel Hofmann

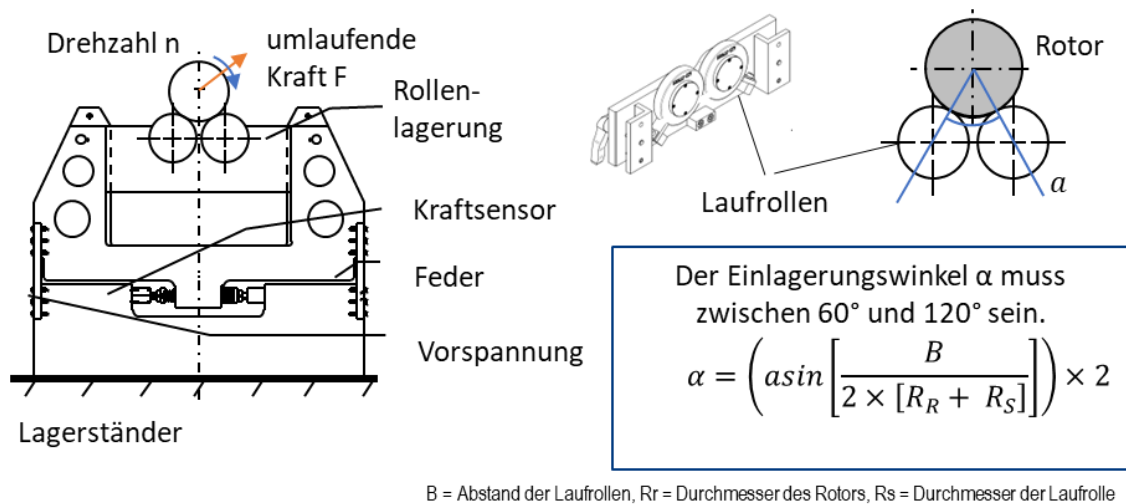


Bild A-68: Darstellung der Berechnung des Einlagerungswinkels

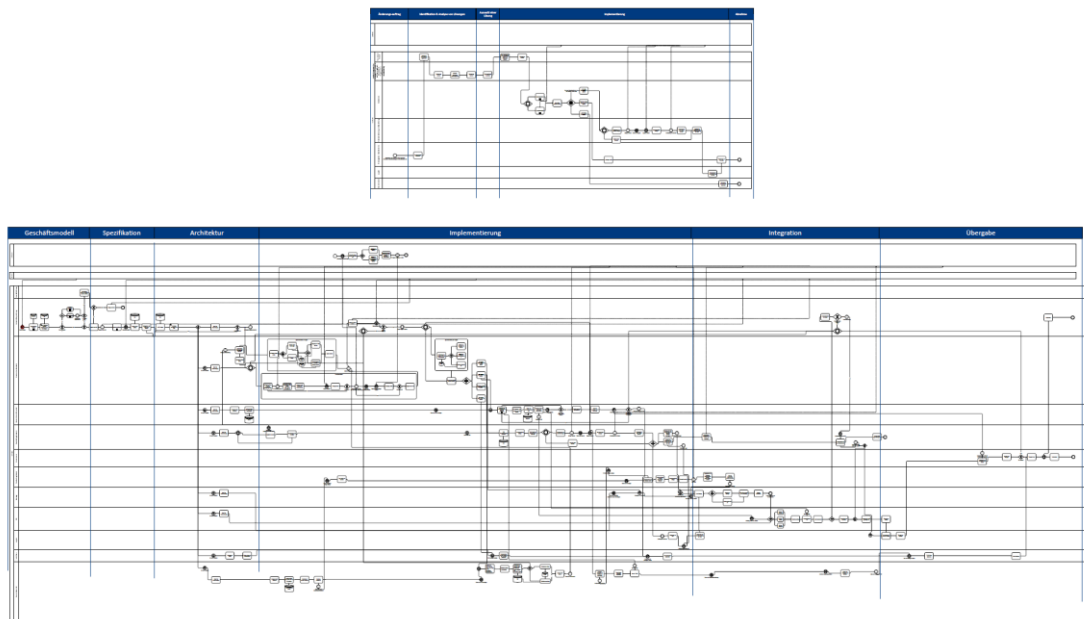


Bild A-69: BPMN Diagramme Hofmann

Historische technische Änderungen Hofmann

Änderungsbedarf: „Die Farbe der Maschine muss auf Wunsch des Kunden angepasst werden. Das Farbschema (RAL) wird vom Kunden bereitgestellt“

Auswirkungen der Änderung:

- Prüfung aller Bauteile der Stückliste auf Lackierungsänderung
- Umlackieren bestehender Vorratskomponenten oder Bezug korrekt lackierter Teile

Besonderheiten:

- Hoher Kommunikationsaufwand intern als auch mit Kunden und Lieferanten
- Abwägung: Lackierung intern durchführen oder an Lieferanten abgeben?



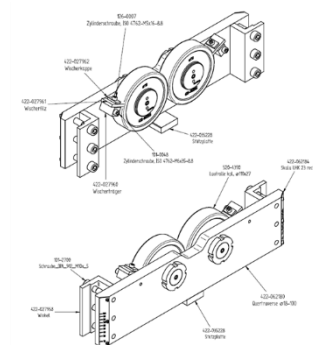
Änderungsbedarf: „Der Kunde fordert einen größeren Rotor auf der Maschine auszuwuchten. Hierzu muss die Lagerung angepasst werden.“

Auswirkungen der Änderung:

- Anpassung der Konstruktion des Rollenlagers
- Prüfung der Kompatibilität mit anderen Maschinenelementen
- Anpassung anderer Komponenten z.B. Lagerständer, Antrieb

Besonderheiten:

- Option 1: Der Rotor kann auf einer anderen Größe der Maschinen-Baureihe ausgewuchtet werden
- Option 2: Rollenlagersatz tauschen
- Option 3: Neukonstruktion des Rollenlagers falls keine passende SK vorhanden



Änderungsbedarf: „Der Kunde fordert einen magnetisierten Rotor auf der Maschine auszuwuchten. Alle rotornahen Bauteile müssen deshalb aus Edelstahl gefertigt werden“

Auswirkungen der Änderung:

- Neukonstruktion aller rotornahen Bauteile
- Prüfung der Kompatibilität mit anderen Maschinenelementen

Besonderheiten:

- Für die Entscheidung eines neuen Lösungsprinzips wird die Expertise weiterer Kolleginnen und Kollegen benötigt, um die Machbarkeit des technischen Änderungsbedarfs zu bewerten.
- Herausforderung für Lieferanten da anderer Werkstoff, eventuell auch Auswahl neuer Lieferanten



Bild A-70: Exemplarische Änderungen Hofmann

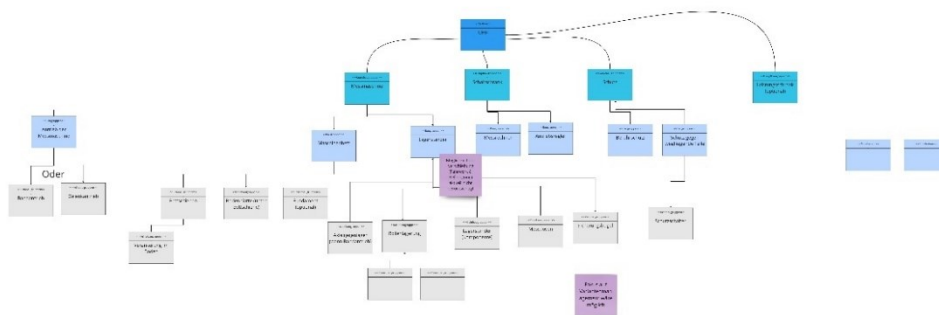


Bild A-71: Dokumentation der Modellierungsworkshop Hofmann

Parameter			
Modell	Parameter	E/A	Parameter
Projektierung Auftragskalkulation	Maschinenstruktur	E	Struktureller Aufbau der Maschinenkomponenten
	Fertigungsteilkosten	E	Geplante/ Geschätzte Anschaffungskosten für Fertigungsteile
	Kaufteilkosten	E	Geplante/ Geschätzte Anschaffungskosten für Katalogteile
	Lizenzkosten	E	Lizenzkosten für Hard-und Software
	Interne Personalkosten	E	Geplanter/geschätzter Personalaufwand
	Externe Personalkosten	E	Geplante/ geschätzte Kosten für Inbetriebnahme bei Kunden (z.B. Reisekosten)
	Transportkosten	E	Geplanter/geschätzter Aufwand für Transport zum Kunden (Speditionen, Seefracht, Luftfracht etc.)
	Zoll- und Exportkosten	E	Geplante/geschätzte Kosten der Abwicklungskosten für Ausfuhr
	Dienstleistungskosten	E	Geplante/geschätzte Kosten für externe Dienstleister (z. B. Schaltschrankbau, Übersetzung von Handbüchern, Programmierleistungen etc.)
	Angebotskosten	A	Vollständige geplante Kosten einer Maschine
	Angebotspreis	A	Angebotspreis für den Kunden
	Auftragskosten	A	Vollständige geplante Kosten einer Maschine
Projektplanung	Starttermin	E	Beginn des Produktentstehungsprozesses im Projekt
	Endtermin	E	Abschluss des Produktentstehungsprozesses im Projekt (Endabnahme)
	Beteiligte Abteilungen	E	Alle Abteilungen die am PEP beteiligt sind
	Dauer pro Produktentstehungsabschnitt	E	Zeitliche Kapazitäten die alle beteiligten Abteilungen zur Verfügung stellen müssen, aber auch Beschaffungsdauern
	Meilensteine	A	Kritische Punkte im PEP
	Großkapazitätsplanung	A	Abteilungsübergreifende Kapazitätsplanung
Messmittel	Bauteiltoleranz	E	Zulässige Restunwucht eines Bauteils, Ziel wie gut das Bauteil ausgewuchtet werden muss
	Bauteilgewicht	E	Gewicht des auszuwuchtenden Kundebauteils
	Messunsicherheit eines Referenzmesssystems	E	Messfähigkeitsuntersuchung eines in der Vergangenheit bereits erstellten Messsystems
	Cgk Wert	A	Messmittelfähigkeitsindex: Verhältnis zwischen einem bestimmten Prozentsatz der Toleranz und der durch das Messmittel verursachten Streuung unter Berücksichtigung der systematischen Messabweichung an.

Bild A-72: Hofmann Aufnahme der Parameter (I/II)

	Laufrolle	Laufrollen- Bolzen- lager	Klemm- hebel	Klemm- winkel	Stützplatte	Traverse	Zylinder- schraube	Wischer- träger	Wischerfilz	Wischer- kappe	Zylinder- schraube
Laufrolle		Wenn Geometrie der Laufrolle ändert, KANN sich Lager ändern				Wenn sich Durchmesser der Laufrolle ändert, müssen Bohrungen in der Traverse geändert werden		Wenn sich Position/ Größe der Laufrolle ändert, kann sich Position oder Größe vom Wischerträger ändern		Wenn Laufrolle breiter wird dann auch Wischerfilz	
Laufrollen- lager	Wenn sich Geometrie des Lagers ändert MUSS sich Laufrolle ändern		Wenn sich Innendurchmesser Lager ändert, muss Bolzen angepasst werden								
Bolzen						Wenn sich Bolzen ändert KANN sich Bohrung von Traverse ändern					
Klemm- hebel				Wenn sich Klemmhebelgewinde ändert, muss sich Bohrung ändern		Wenn sich Klemmhebelgewinde ändert, muss sich Bohrung ändern					
Klemm- winkel											
Stützplatte											
Traverse			Wenn Traverse dicker wird (z.B. weil VA nicht so viel Kraft aushält) muss das Gewinde des Bolzen angepasst werden (aber sehr unwahrscheinlich)								
Zylinder- schraube											
Wischer- träger						Wenn sich Position von Wischerträger ändern, muss Bohrung in Traverse geändert werden					
Wischerfilz										Wenn Laufrolle breiter wird, damit der Filz, muss auch Wischerkappe breiter werden	
Wischer- kappe											
Zylinder- schraube											

Bild A-73: Hofmann Aufnahme der Parameter (II/II)

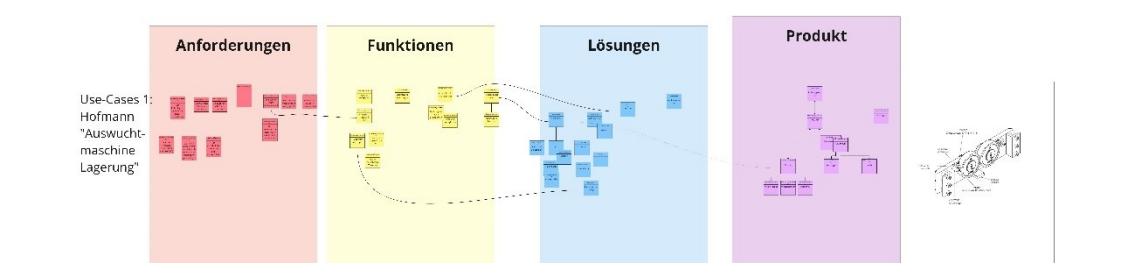
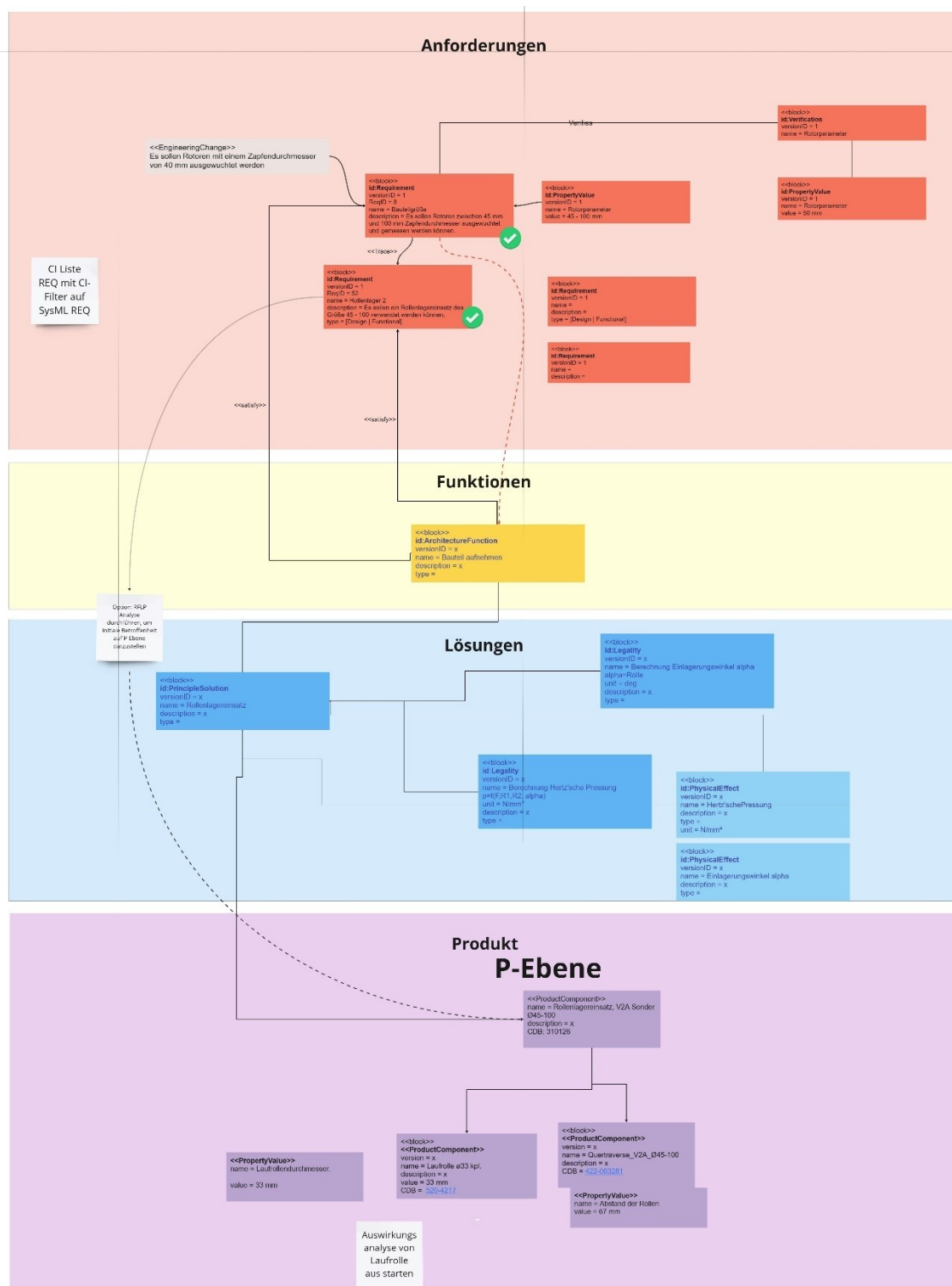


Bild A-74: Hofmann RFLPV²

Bild A-75: Hofmann RFLPV² entlang von Elementen

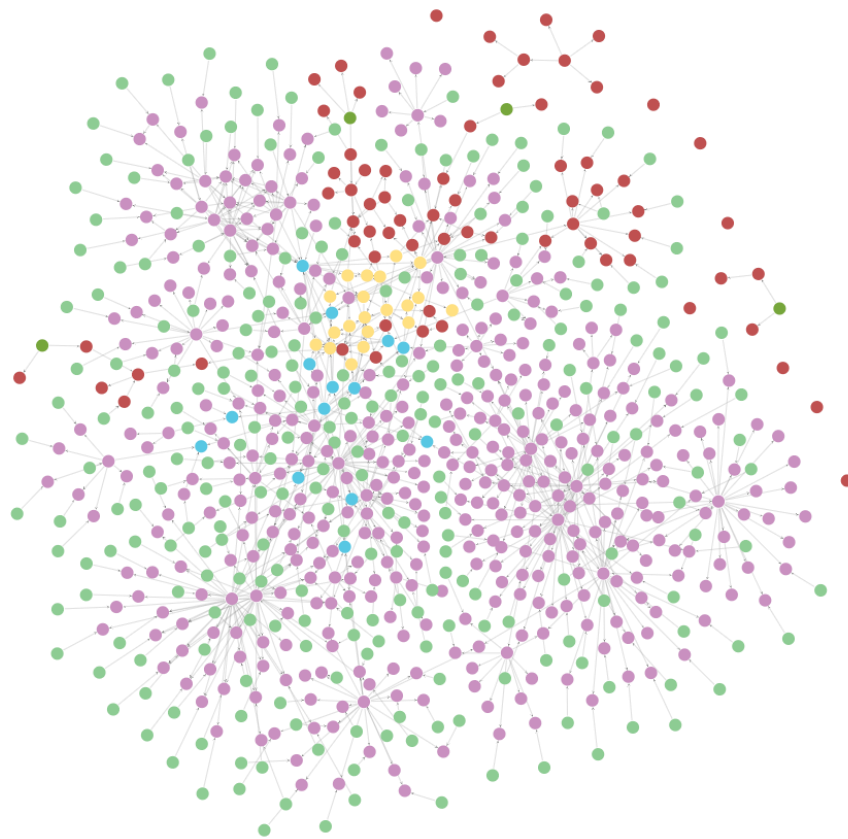


Bild A-76: Neo4j Import Hofmann

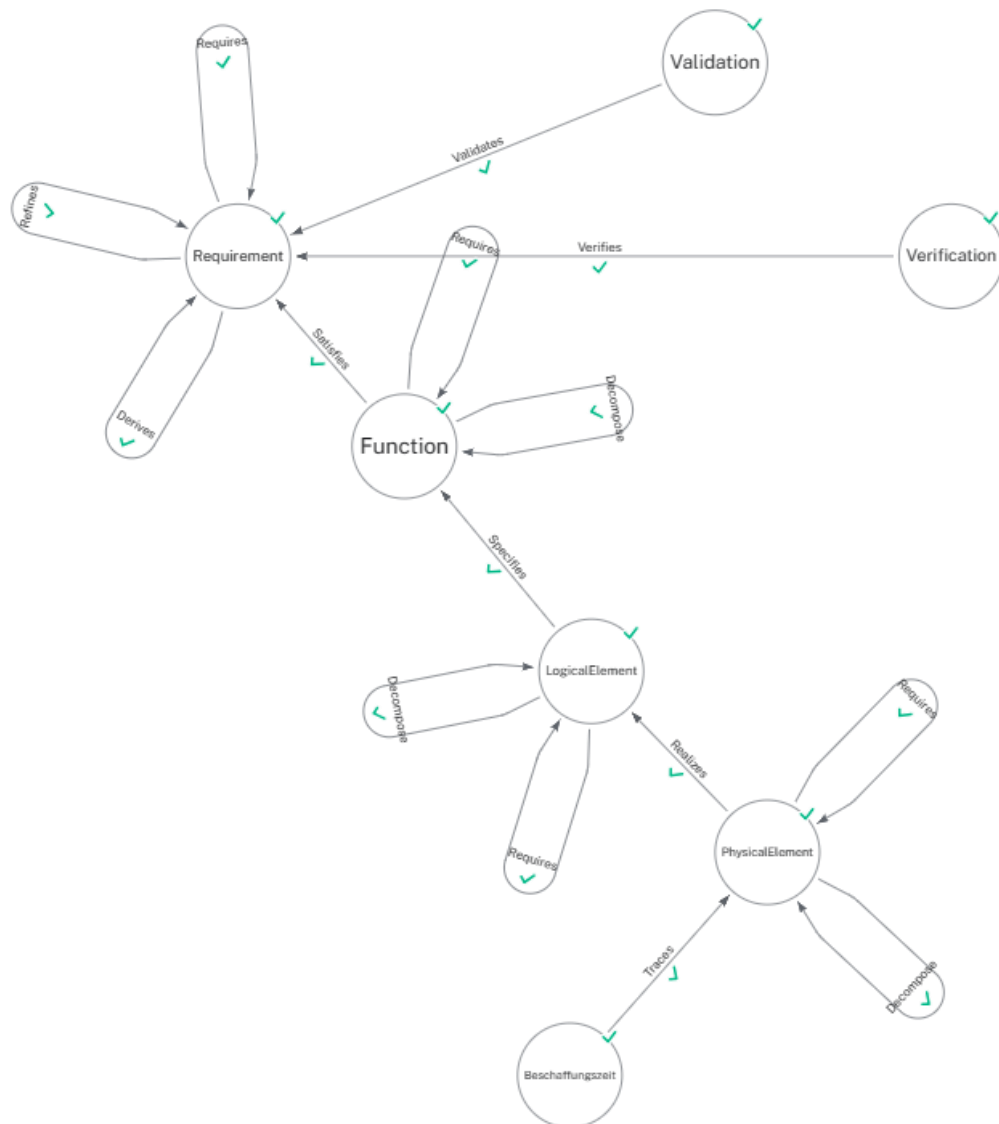


Bild A-77: Metadatenmodell des Graphen Hofmann

Tabelle A-19: Auflistung der Elemente aus dem Graphen Hofmann

NodeLabels,Count	Count	RelationshipType	Count
[Requirement]	74	Requires	16
[LogicalElement]	14	Derives	22
[PhysicalElement]	492	Refines	16
[Verification]	19	Decompose	527

[Validation]	4	Specifies	29
[Function]	20	Realizes	25
[Beschaffungszeit]	595	Verifies	21
		Validates	10
		Satisfies	23
		Traces	275
Summe	1218		964

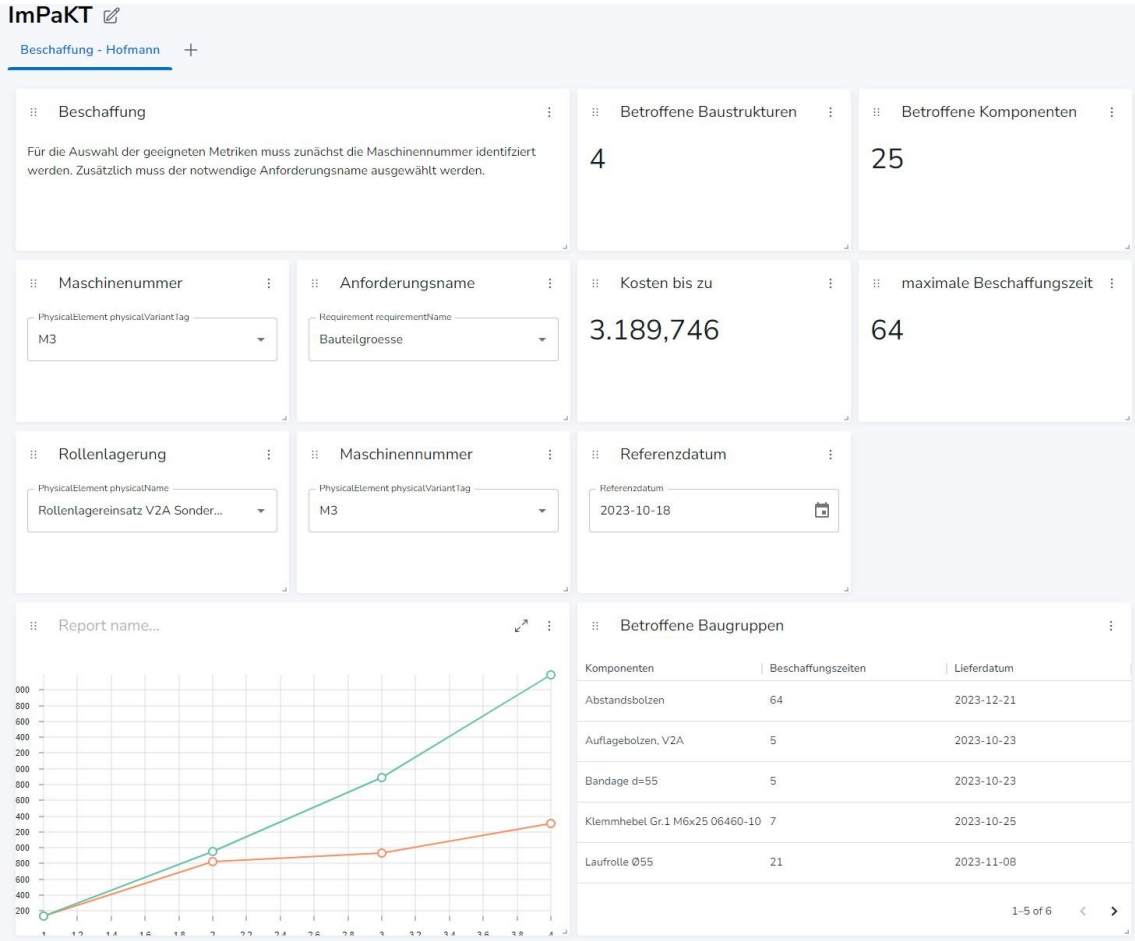


Bild A-78: Dashboard Hofmann Beschaffung

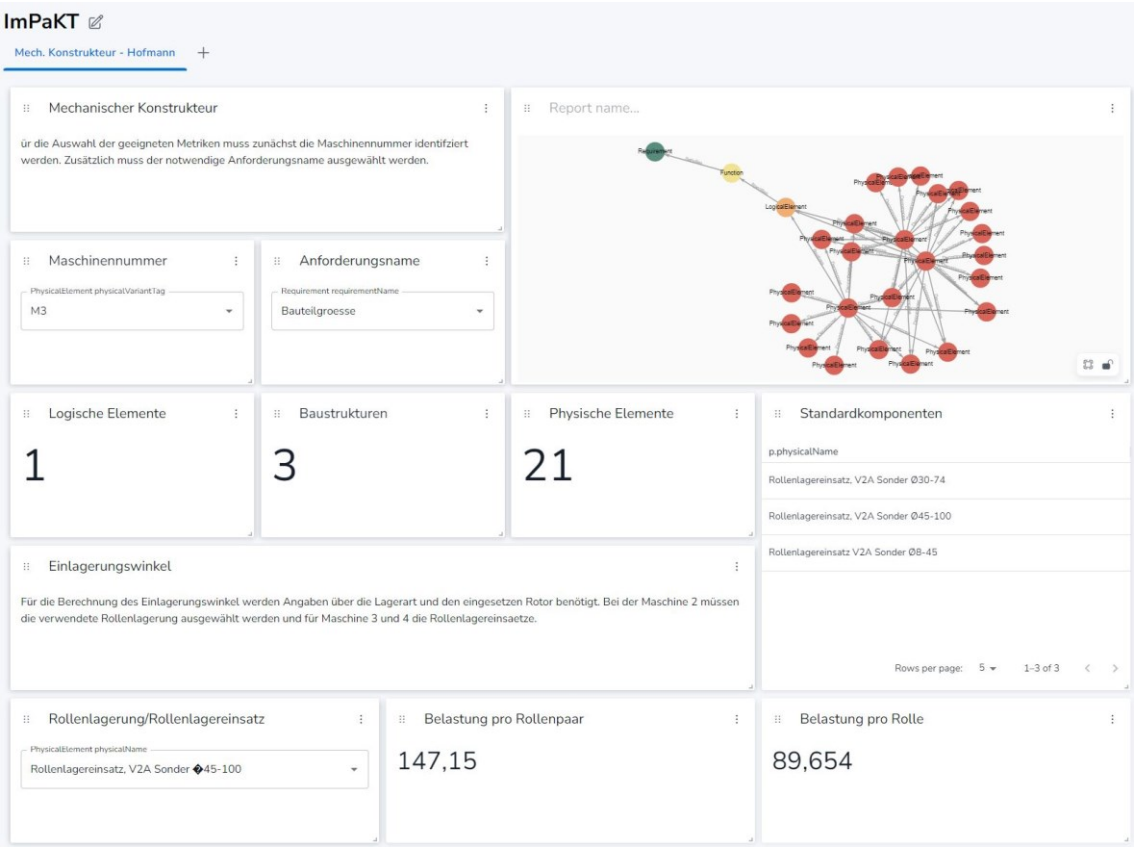


Bild A-79: Dashboard Hofmann Mechanik

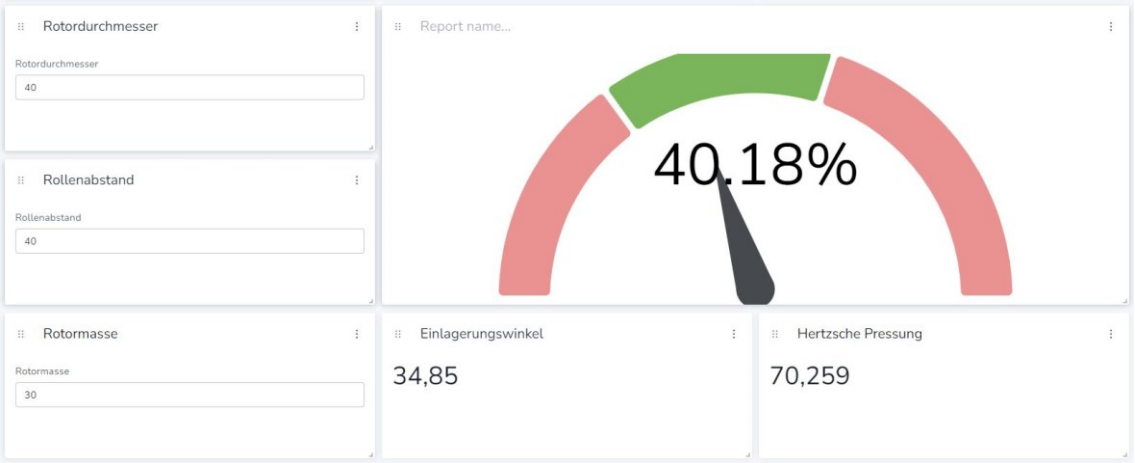


Bild A-80: Dashboard Hofmann Einlagerungswinkel

A9 Fallstudie Hadi Plast

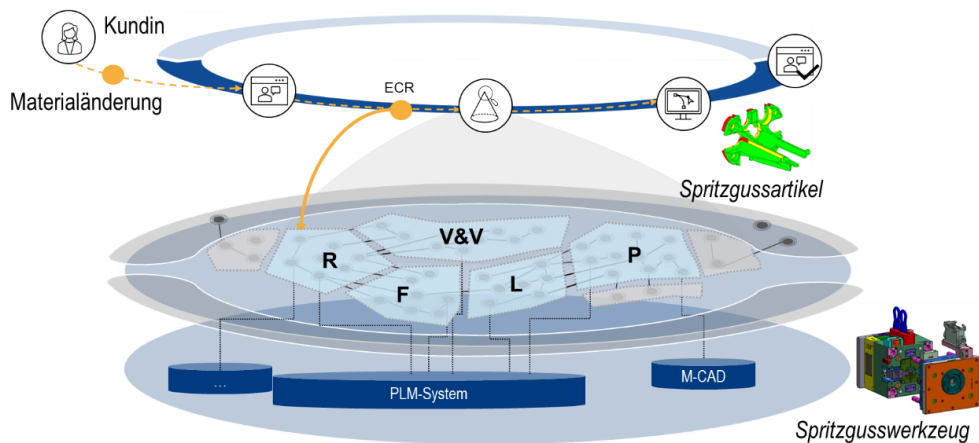


Bild A-81: Big Picture der Fallstudie Hadi Plast

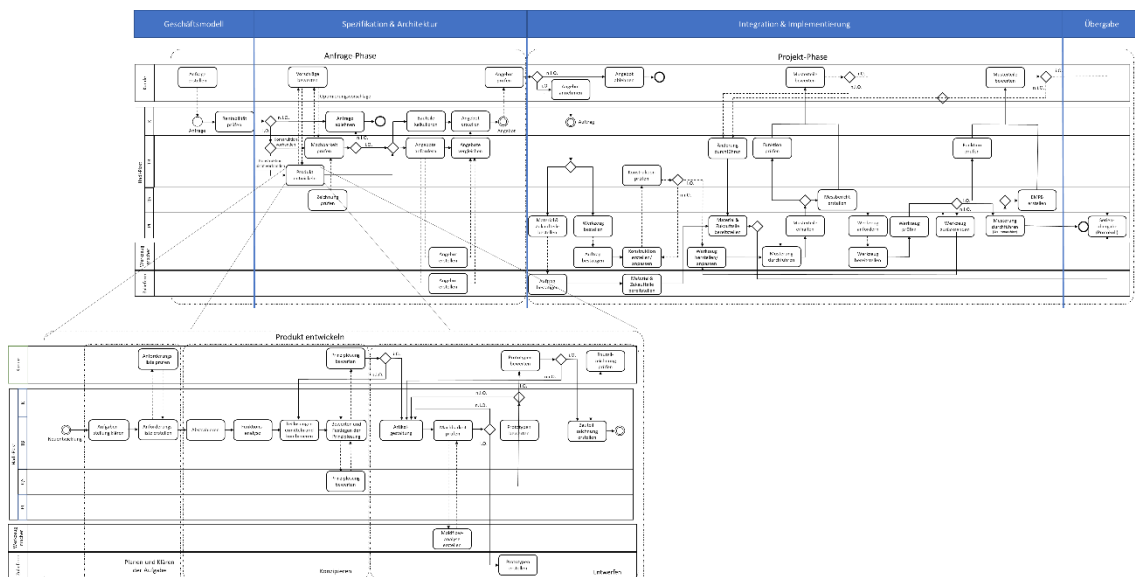


Bild A-82: BPMN Diagramme Hadi Plast

Änderungsbedarf: Die Kopf eines Schnapphakens muss vergrößert werden, um die Funktionalität des Spritzgussteils sicherzustellen.

Auswirkungen der Änderung:

- Die Konstruktion muss angepasst werden
- Die Werkzeugparameter müssen angepasst werden
- Die Kostenrechnung muss erneut durchgeführt werden

Besonderheiten:

- Intensive Kommunikation mit Kunden notwendig
- Kommunikation mit externem Werkzeugmacher notwendig

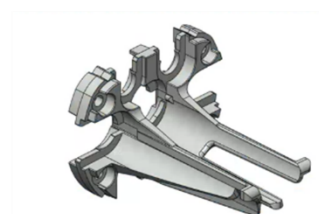


Bild A-83: Historische Änderung Hadi Plast

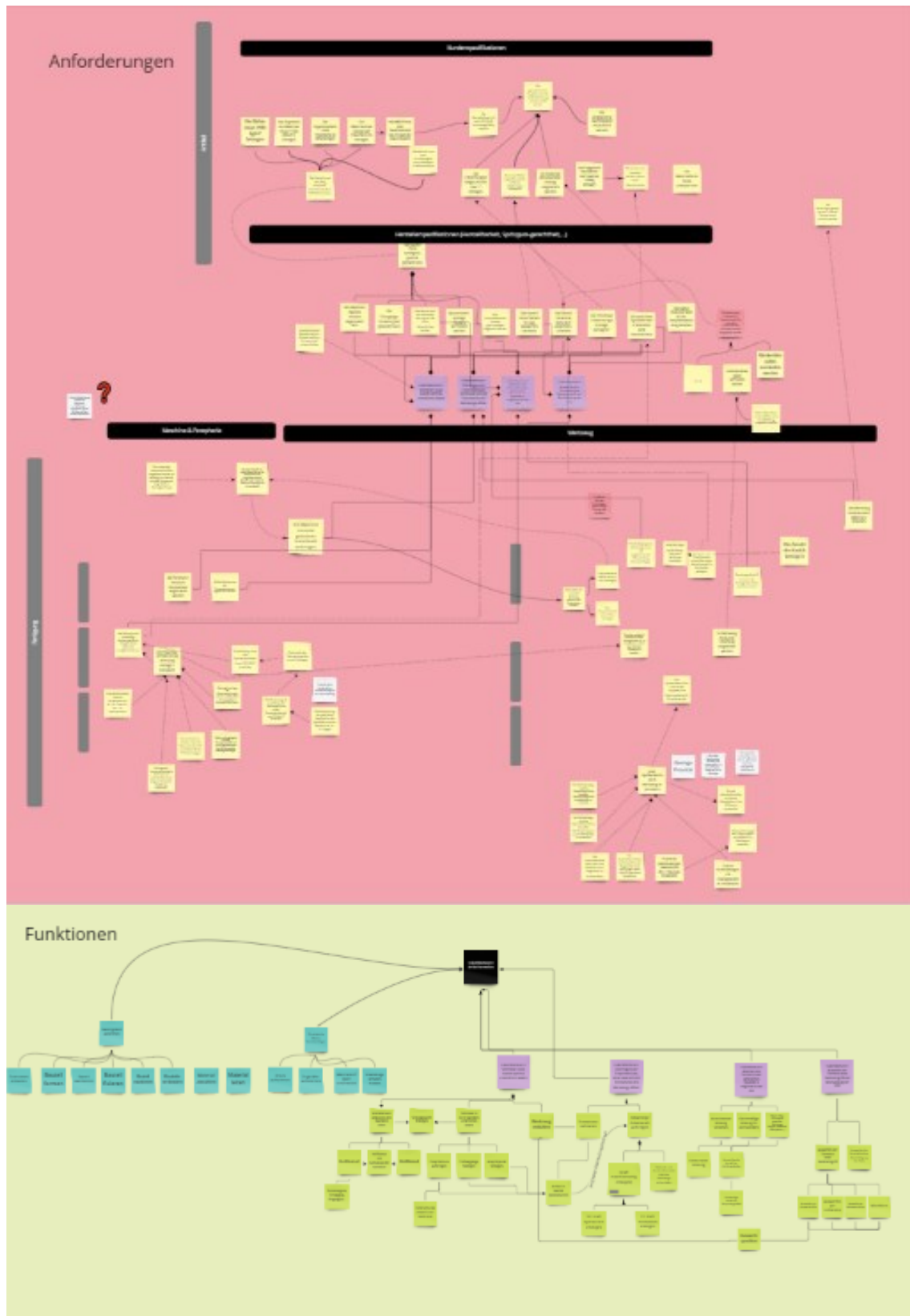


Bild A-84: Dokumentation des Modellierungswshops Hadi Plast (I/II)



Bild A-85: Dokumentation Modellierungsworkshop Hadi Plast (II/II)

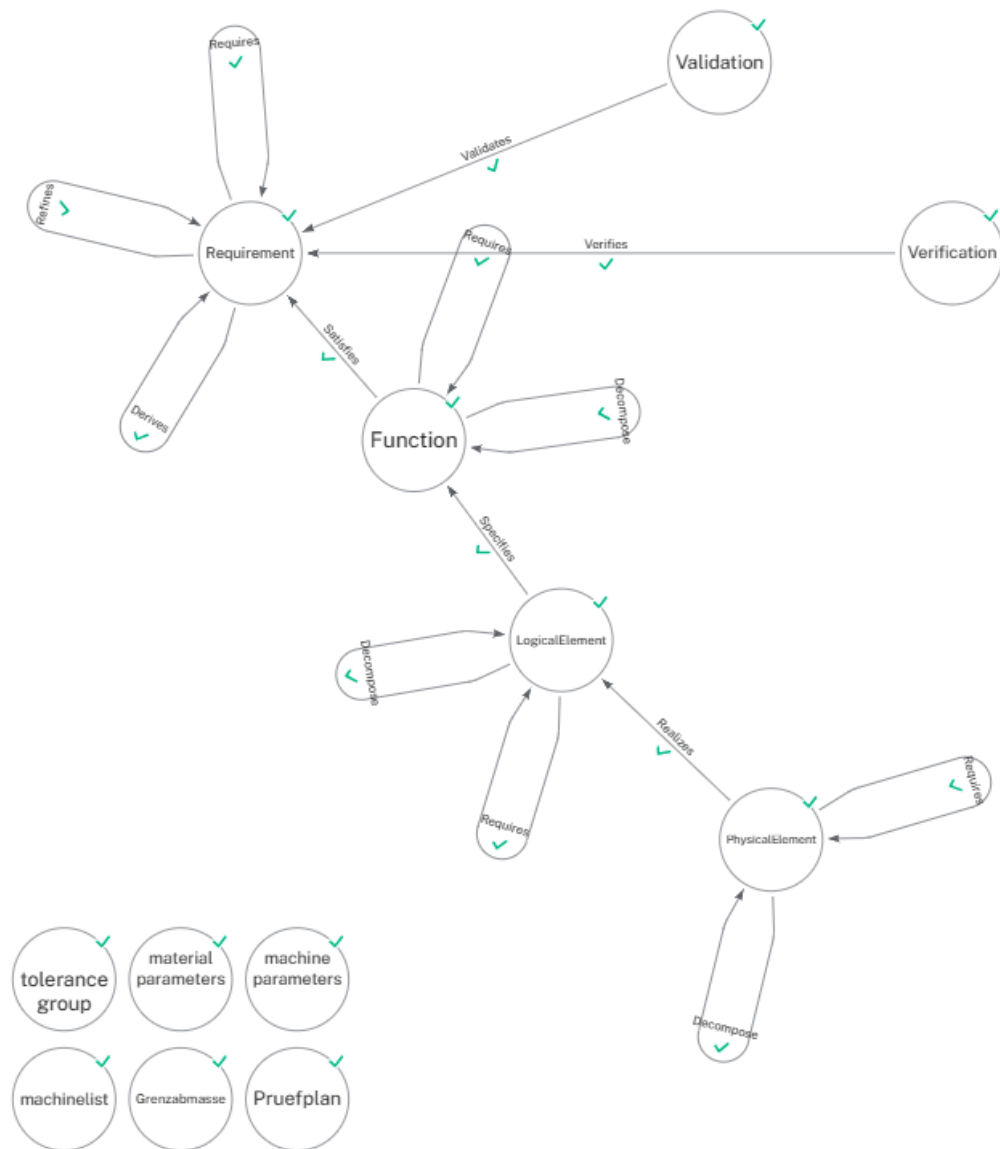


Bild A-86: Neo4j Import Hadi Plast

Tabelle A-20: Darstellung der Elemente im Beispiel Hadi Plast

RelationshipType	Count	NodeLabels	Count
Derives	42	[Requirement]	76
Requires	27	[LogicalElement]	66
Refines	12	[PhysicalElement]	109
Specifies	69	[Verification]	1
Decompose	192	[Validation]	1
Realizes	49	[Function]	58
Verifies	1	[machine parameters]	36
Validates	1	[material parameters]	3
Satisfies	39	[tolerance group]	18
		[machinelist]	38
		[Grenzabmasse]	16
		[Pruefplan]	9
Summe	432		431

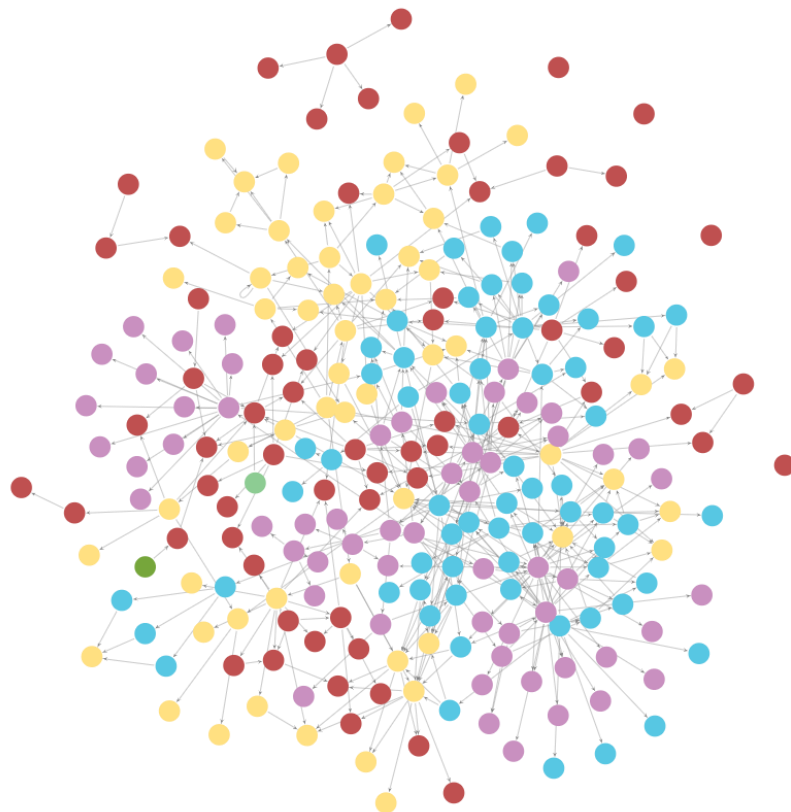


Bild A-87: Darstellung Hadi Plast im Graphen

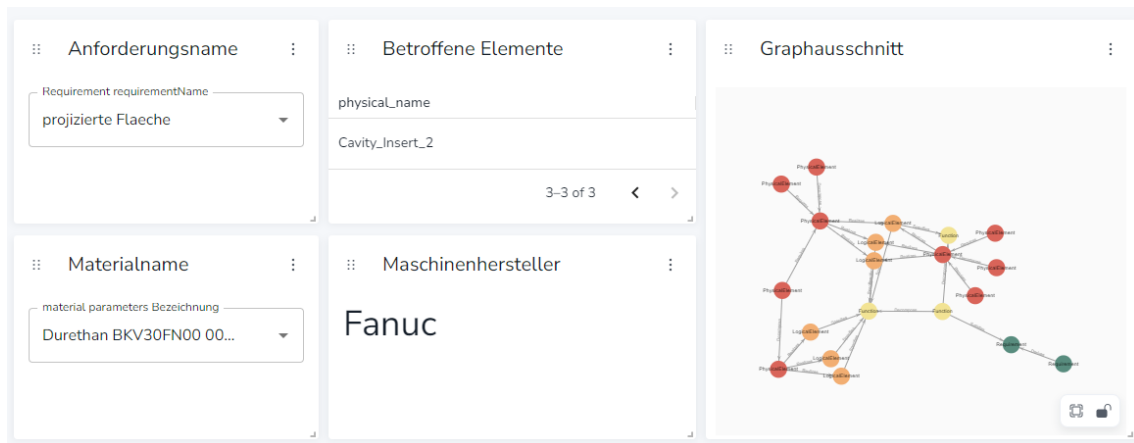


Bild A-88: Dashboards Hadi Plast Projektleiter

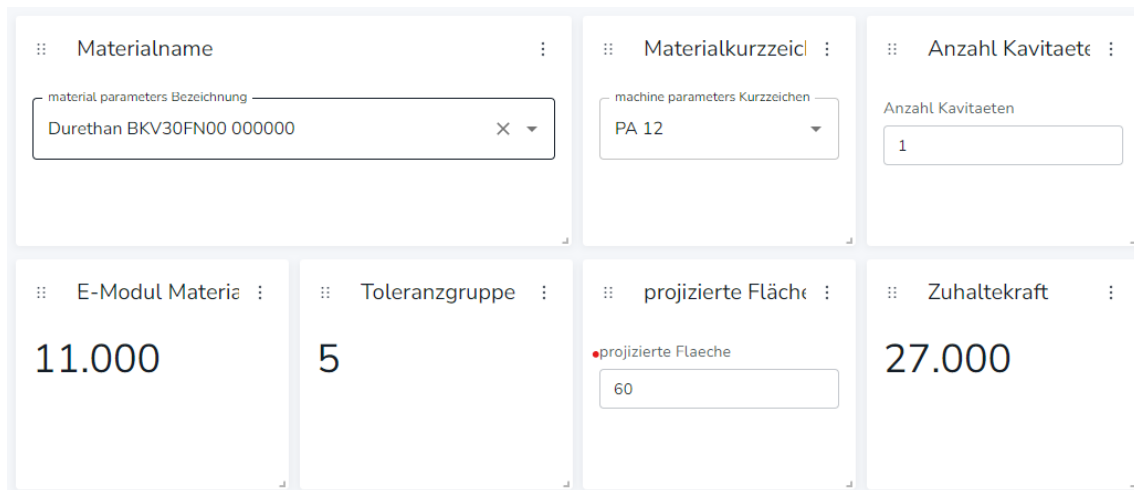


Bild A-89: Dashboard Hadi Plast Prozessingenieur

Grenzabmaße Soll Grenzabmasse <input type="text" value="37"/>	Toleranzgruppe Grenzabmasse W_NW <input type="text" value="TG4W"/>	Toleranzen Soll $\pm 0,13$
Ist-Maße <hr/>		
Max Toleranzen $0,15$	Grenzabmaße $37,1$	

Bild A-90: Dashboard Hadi Plast Qualitätssicherung (I/II)

P1. Fertigungspr tolerance group Auspraegung P1 <input type="text" value="Spritzgiessen"/>	P2. Formsteifigkeit & -härte Tolerance Group Auspraegung P2 <input type="text" value="3000"/>
P3. Verarbeitung Tolerance Group Auspraegung P3 <input type="text" value="0.001"/>	P4. Geometrie & Prozessschwi tolerance group Auspraegung P4 <input type="text" value="genau möglich +/- 10%"/>
P5. Normalfertig tolerance group Auspraegung P5 <input type="text" value="Normalfertigung"/>	Toleranzgruppe 4

Bild A-91: Dashboard Hadi Plast Qualitätssicherung (II/II)

<div>Materialname</div> <div>material parameters Bezeichnung</div> <div>Durethan BKV30FN00 000000</div>	<div>Materialkurzzeichen</div> <div>machine parameters Kurzzeichen</div> <div>PA 12</div>	<div>Anzahl Kavitaeten</div> <div>Anzahl Kavitaeten</div> <div>1</div>	
<div>E-Modul Material</div> <div>11.000</div>	<div>Toleranzgruppe</div> <div>5</div>	<div>projizierte Fläche</div> <div>projizierte Fläche</div> <div>60</div>	<div>Zuhaltekraft</div> <div>27.000</div>

Bild A-92: Dashboard Hadi Plast Technische Betreuung

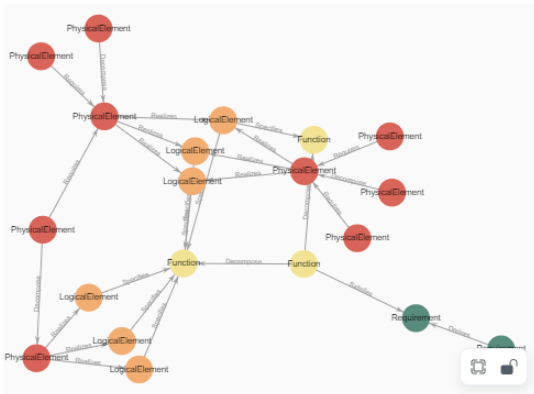
<div>Anforderungsname</div> <div>Requirement requirementName</div> <div>projizierte Fläche</div>	<div>Material Kurzzeichen</div> <div>machine parameters Kurzzeichen</div> <div>PA 12</div>	<div>Betroffene Elemente</div> 
<div>projizierte Fläche</div> <div>projizierte Fläche</div> <div>60</div>	<div>Anzahl Kavitaeten</div> <div>Anzahl Kavitaeten</div> <div>1</div>	
<div>Zuhaltekraft</div> <div>27.000</div>		

Bild A-93: Dashboard Hadiplast Zuhaltekraft

- Der Spritzgussartikel ist speziell entwickelt, um eine Platine sicher im Ladestecker für E-Autos zu halten, wodurch die Stabilität und Funktionalität gewährleistet werden.
- Der Spritzgussartikel schützt die Platine vor äußeren Einflüssen wie Feuchtigkeit und Vibrationen und ermöglicht eine einfache Montage während des Produktionsprozesses des E-Ladesteckers.

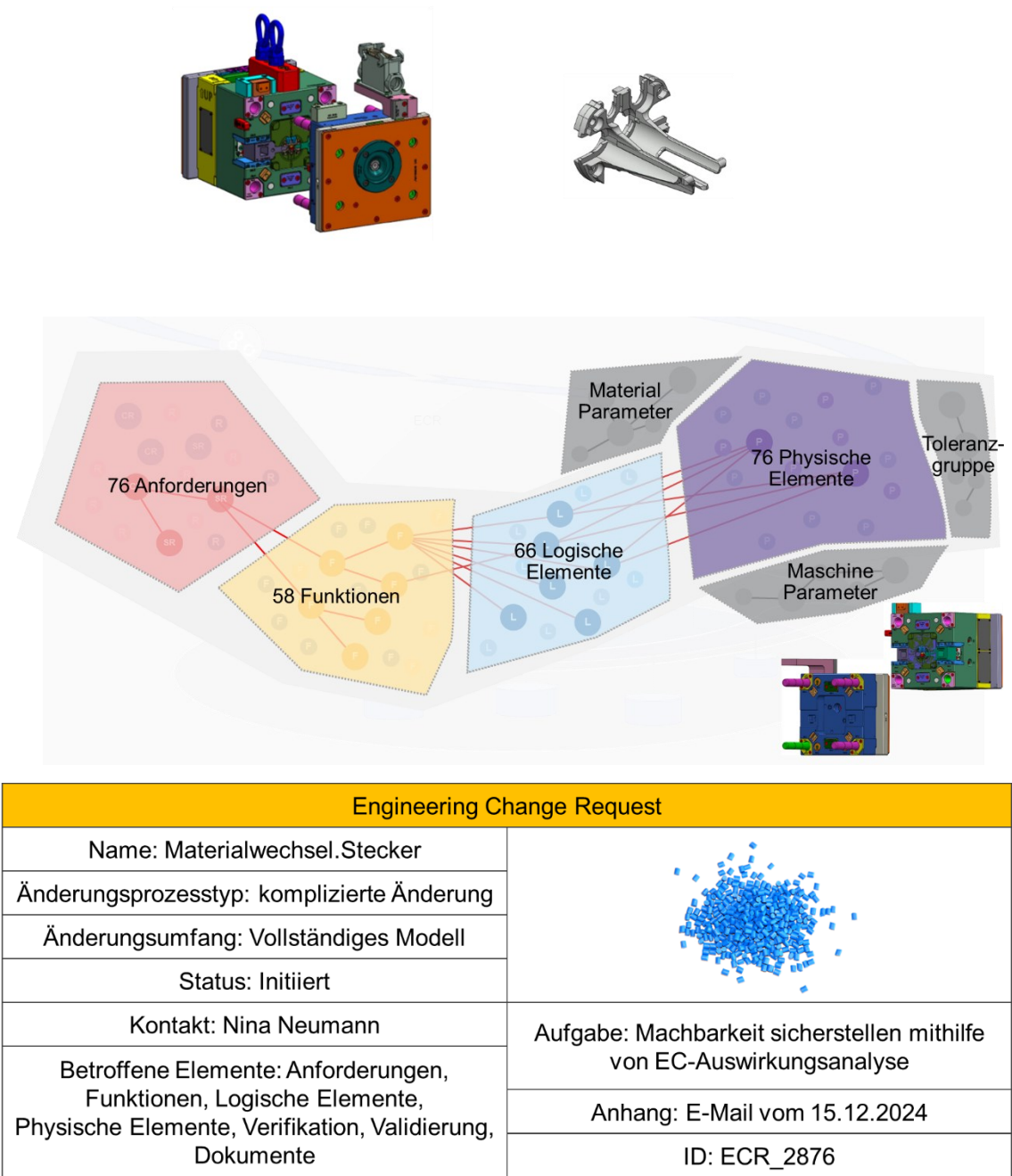


Bild A-94: Erklärvideo Hadi Plast

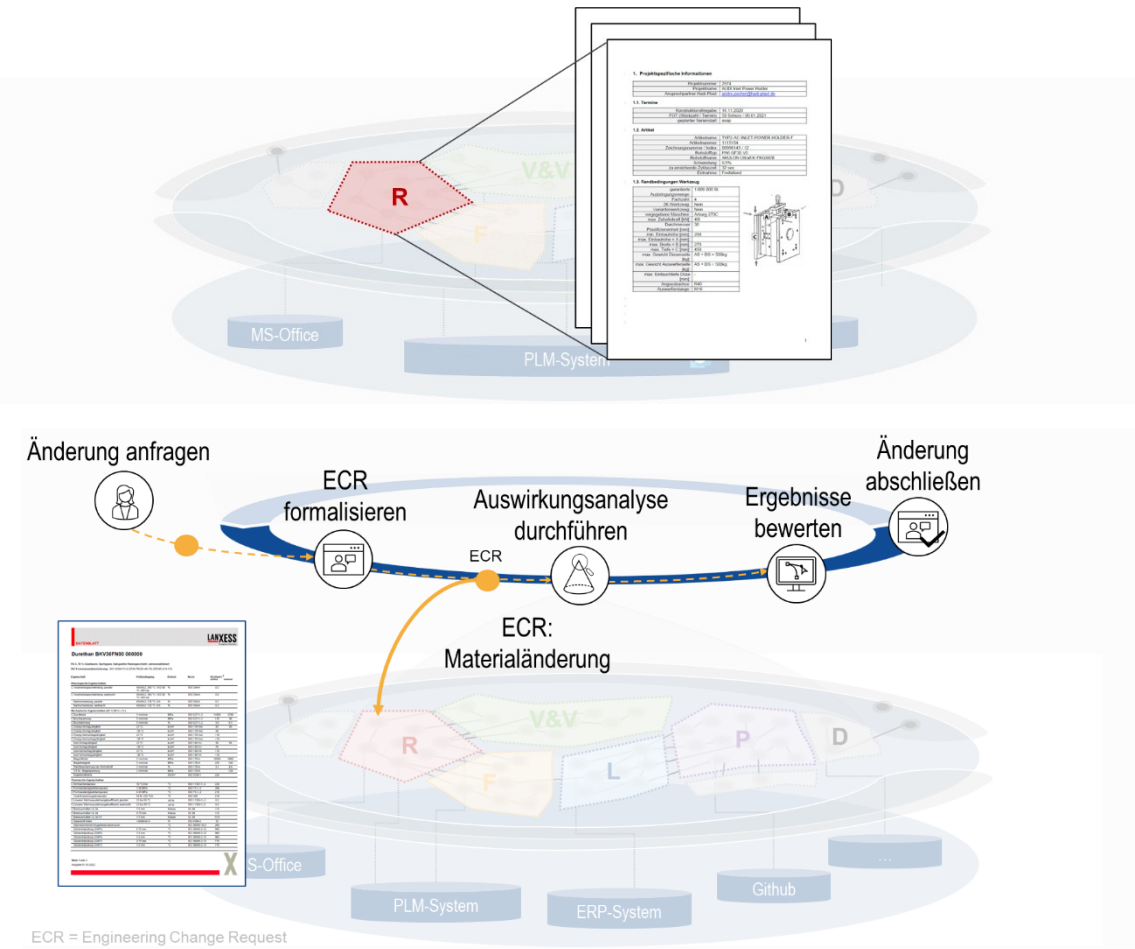


Bild A-95: Big Picture Hadi Plast

A10 Validierung

A10.1 Bewertungsmethoden nach Hevner

Tabelle A-21: Bewertungsmethoden nach Hevner [HCG10]

Beobachtende Bewertungsmethoden (Observationen)	Fallstudie: Vertieftes Studium des Artefakts im Unternehmensumfeld Feldstudie: Überwachung der Verwendung von Artefakten in verschiedenen Projekten
Analytische Bewertungsmethoden	Statische Analyse: Untersuchung der Struktur eines Artefakts auf statische Eigenschaften (z. B. Komplexität) Architektur-Analyse: Untersuchung der Einpassung des Artefakts in die technische IS-Architektur Optimierung: Demonstration inhärenter optimaler Eigenschaften des Artefakts oder Bereitstellung von Optimalitätsgrenzen für das Verhalten des Artefakts

	Dynamische Analyse: Untersuchung des Artefakts im Einsatz auf dynamische Eigenschaften (z. B. Leistung)
Experimentelle Bewertungsmethoden	Kontrolliertes Experiment: Untersuchung eines Artefakts in einer kontrollierten Umgebung auf seine Eigenschaften (z. B. Benutzerfreundlichkeit) Simulation - Ausführen des Artefakts mit künstlichen Daten
Test-Methoden	Funktionale (Black Box) Tests: Ausführen von Artefaktschnittstellen zur Entdeckung von Fehlern und zur Identifizierung von Mängeln Strukturelles (White Box) Testen: Testen der Abdeckung einiger Metriken (z.B. Ausführungspfade) in der Implementierung des Artefakts
Beschreibende (deskriptive) Bewertungsmethoden	Sachkundiges Argument: Verwendung von Informationen aus der Wissensbasis (z.B. relevante Forschung), um ein überzeugendes Argument für den Nutzen des Artefakts zu entwickeln Szenarien: Konstruieren Sie detaillierte Szenarien rund um das Artefakt, um seinen Nutzen zu demonstrieren.

A10.2 Aufbau Prüfstand

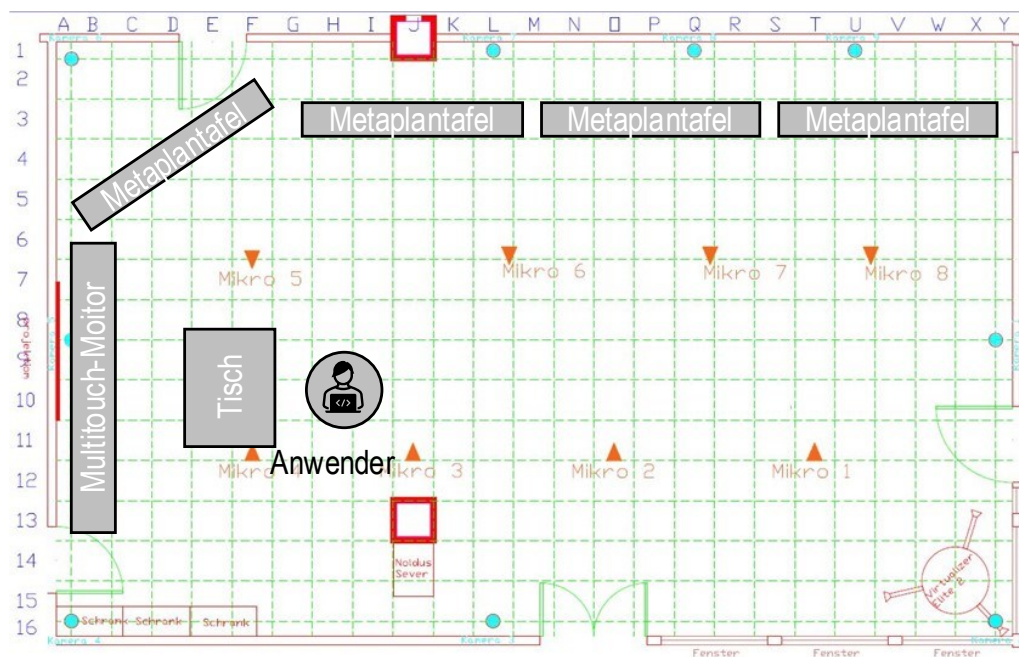


Bild A-96: Aufbau Prüfstand im Smart Automation Lab

A10.3 Kriterien Planung des Prüfablaufs

Tabelle A-22: Planung des Prüfablaufs anhand der Kriterien

Planung des Prüfablaufs auf Basis der DRM Charakteristika [BC09, S. 243]	
Dimensionen	Umsetzung im Forschungsaufbau
Zielsetzung, Forschungsfragen, Hypothesen	Bewertung des zeitlichen Aufwands je Modellierungsschritt bei der Erstellung von Wirkketten mit der MECA Methode
Theoretische Grundlage	Vorgegebene Modellierungsmethode, -tool und –sprache SysML Diagramme, Modellelemente und Relationen
Aufgabe	Industriennahe Aufgabe auf Basis eines Fallbeispiels des Lehrstuhls
Anzahl und Größe der Fälle	1 Wirkkette eines Gesamtsystems Jede Wirkkette wird separat aufgenommen
Teilnehmer	1-Personen mit technischer Hintergrund und SysML-Grundkenntnissen, Einzelarbeit
Designobjekt	Modellbasierte Wirkketten eines komplexen Gesamtsystems
Datenerhebungsmethode	Direkte Beobachtung mit Video, Coding, teilnehmende Beobachtung, Fragebogen
Rolle des Forschenden	Einbindung des Forschenden in den Forschungsprozess, nachträgliche Analyse des Videomaterials
Art der Studie	Beobachtende Studie
Zeitbeschränkung und Dauer	Fertigstellung der fünf Wirkketten
Setting	Smart Innovation Lab
Coding und Analyse	Coding mithilfe der Software des Noldus-Systems

A10.4 Validierungs-Workshops (Zyklus 1-3)



Anforderung

Anforderung		Prüfkriterium	Art des Testcases	Notizen
A1: Verfügbarkeit von Daten	Die Auswirkungsanalyse muss so gestaltet sein, dass Sie mit im den Informationen und Daten aus dem Produkt-Systemmodell (inkl. Partialmodelle, ALM, PLM) (AP2) durchgeführt werden kann	Informationen Modellelemente innerhalb der ECRS	Walkthrough durch ECR-Templates	
A4: Tiefe und Breite der Algorithmen	Die Algorithmen müssen Informationen auf verschiedenen Granularitätsleveln (Tiefe) und aus verschiedenen Disziplinen (Breite) berücksichtigen	Modellelemente innerhalb der Auswirkungsanalyse	Walkthrough durch Auswirkungsanalyse-Templates	
A7: Tailoring / Anwenderindividuelle Analyse	Die Ergebnisse der Auswirkungsanalyse müssen an die unternehmensspezifischen Bedarfe angepasst werden können (Tailoring)	Hadi Plast ECR	Bewertung der Anpassbarkeit anhand der Auswirkungsanalysen	
A11: Ergebnisqualität	Die Ergebnisse der Auswirkungsanalyse müssen eine hinreichende Genauigkeit vorweisen und die qualitativen Ansprüche hinreichend erfüllen	Alle ECRs	Bewertung der gefolgerten Ergebnisse durch die Anwender	

Bild A-97: Validierungsworkshop (I/II)

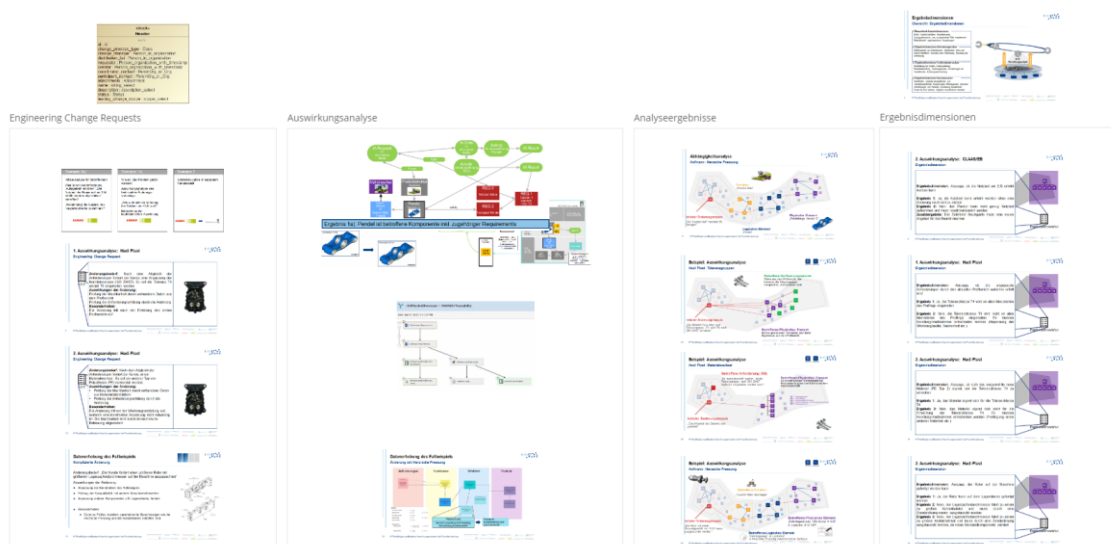


Bild A-98: Validierungsworkshop (II/II)

A10.5 Validierungs-Fragebogen

ImPaKT Fragebogen zur Validierung der Methodik zur
Auswirkungsanalyse/Wirkkettenanalyse



ImPaKT – Fragebogen zur Validierung der Methodik

Liebe(r) Teilnehmer(innen),

im Rahmen des Forschungsprojektes ImPaKT „IKT-gestützte modellbasierte Auswirkungsanalyse in der Produktentwicklung“ soll eine Validierung der erarbeiteten Methodik zur Entwicklung von Auswirkungsanalysen und der zugehörigen softwaregestützten Implementierung durchgeführt werden. Die Validierung wird in folgenden Abschnitten durchgeführt

1. Angaben zum betrachteten Fallbeispiel
2. Unternehmenssituation
3. Anwendungsaufwand der Methodik
4. Anwendbarkeit der softwaregestützten Implementierung
5. Genauigkeit der Ergebnisse
6. Mehrwert der Methodik-Anwendung

Bitte bewerten Sie die nachfolgenden Thesen in den weiteren Abschnitten nach dem folgenden Schema:

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

Bei nachträglichen Rückfragen, Unklarheiten oder Anmerkungen, können Sie sich jederzeit bei mir melden:

Dominik Wiechel

Telefon: +49 (0) 5251 / 606264

E-Mail: Dominik.Wichel@hni.upb.de

Vielen Dank für Ihre Teilnahme!

Dominik Wiechel

Bild A-99: Validierungsfragebogen (I/VIII)

ImPaKT Fragebogen zur Validierung der Methodik zur
Auswirkungsanalyse/Wirkkettenanalyse



Abschnitt 1: Angaben zum betrachteten Fallbeispiel

1) Welches Fallbeispiel wird von Ihnen betreut?

2) Welche Schwerpunkte gibt es in Ihrem Fallbeispiel zu berücksichtigen?

3) Welches Ziel wird von Ihnen durch die Auswirkungsanalyse verfolgt? (Beispiele:
Reduzierung von Entwicklungsaufwand, objektive Entscheidungen zum Umgang mit
technischen Änderungen, ...)

4) Welche Rolle(n) haben Sie in Ihrem Unternehmen?

5) Wie viel Vorerfahrungen hatten Sie mit Auswirkungsanalysen vor dem Forschungsprojekt?

6) Wie werden technische Änderungen und die dazugehörigen Aufwände aktuell in Ihrem
Unternehmen dokumentiert?

ImPaKT Fragebogen zur Validierung der Methodik zur
Auswirkungsanalyse/Wirkkettenanalyse



Bitte beurteilen Sie die folgenden Thesen zur Validierung der Ergebnisse

Abschnitt 2: Unternehmenssituation

- 7) Vor dem Forschungsprojekt wurden Auswirkungsanalysen zur Bewertung technischer Änderungen in Ihrem Unternehmen genutzt.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

- 8) Die Bewertung technischer Änderungen basierte bereits vor dem ImPaKT Projekt auf einer objektiven Informationsgrundlage.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

- 9) Vor dem Forschungsprojekt wurden Bewertungskriterien zur systematischen Abschätzung der Auswirkung einer technischen Änderung verwendet.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

- 10) Die Auswirkungen technischer Änderungen wurden qualitativ bewertet.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

- 11) Die Auswirkungen technischer Änderungen wurden quantitativ bewertet.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

Bild A-101: Validierungsfragebogen (III/VIII)

ImPaKT Fragebogen zur Validierung der Methodik zur
Auswirkungsanalyse/Wirkkettenanalyse



Abschnitt 3: Anwendungsaufwand der Methodik

12) Den einmaligen Modellierungsaufwand zur Erstellung des Systemmodells empfinde ich aus industrieller Sicht als angemessen (Modellierung entlang von RLFPV²).

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

13) Den wiederkehrenden Analyseaufwand zur Erstellung rollen-spezifischer Auswirkungsanalysen empfinde ich aus industrieller Sicht als angemessen (Queries, Dashboards erstellen und anpassen, ...).

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

14) Den Zeitaufwand zur Durchführung von Auswirkungsanalysen empfinde ich aus industrieller Sicht als angemessen (Import in Graphdatenbank, Startzeit der Dashboards, Laufzeit der Queries).

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

15) Der Aufwand zur Interpretation der Ergebnisse empfinde ich als angemessen: Ableitung von Handlungsmaßnahmen aus den Analyseergebnissen.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

Abschnitt 4: Anwendbarkeit der softwaregestützten Implementierung

16) Ich denke, dass ich das System regelmäßig benutzen würde.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

17) Ich empfinde das System als unnötig komplex.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

18) Ich empfinde das System einfach als einfach benutzbar.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

Bild A-102: Validierungsfragebogen (IV/VIII)

ImPaKT Fragebogen zur Validierung der Methodik zur
Auswirkungsanalyse/Wirkkettenanalyse



19) Ich glaube, ich würde die Hilfe einer technisch versierten Person benötigen, um das System benutzen zu können.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

20) Ich fand, die verschiedenen Funktionen in diesem System waren gut integriert.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

21) Ich denke, das System enthielt zu viele Inkonsistenzen.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

22) Ich kann mir vorstellen, dass die meisten Menschen den Umgang mit diesem System schnell lernen.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

23) Ich fand das System sehr umständlich zu nutzen.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

24) Ich fühlte mich bei der Benutzung des Systems sehr sicher.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

25) Ich musste eine Menge lernen, bevor ich anfangen konnte, das System zu verwenden.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

Folgende Daten und Informationen sind in der unternehmerischen Praxis bereits vorhanden.

	Ja	Nein
I. Anforderungen (Lastenheft, Pflichtenheft, Spezifikation, ...)		
II. Funktionen (Funktionsliste, Funktionshierarchie, ...)		
III. Logische Elemente (Vergleich von Lösungsalternativen, ...)		
IV. Physische Elemente (eBOM, mBOM, ...)		
V. Disziplinspezifische Modelle (M-CAD, E-CAD, CASE, ...)		
VI. Kostenkalkulationen		
VII. Änderungsanfragen (Engineering Change Request) von internen oder externen Stakeholdern		
VIII. Benötigte Ergebnisse zur Freigabe einer Änderung (im Sinne von Fragen der Ergebnisdimensionen)		
IX. Bewertungskriterien zur Abschätzung von Auswirkungen einer technischen Änderung (Schwellwerte, mathematische Formeln, Faustregeln, ...)		

Bild A-103: Validierungsfragebogen (V/VIII)

ImPaKT Fragebogen zur Validierung der Methodik zur
Auswirkungsanalyse/Wirkkettenanalyse



Abschnitt 5: Genauigkeit der Ergebnisse

26) Notwendige Informationen auf verschiedenen Granularitätsleveln (Tiefe) und aus verschiedenen Disziplinen (Breite) können in dem RFLPV² Modell abgebildet werden.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

27) Entstehende Propagationseffekte werden bei der Analyse betroffener Elemente in einer hinreichenden Genauigkeit einbezogen.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

28) Die Ergebnisse sind hinreichend genau, um betroffene Elemente der technischen Änderung zu identifizieren.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

29) Die Ergebnisse sind hinreichend genau, die Auswirkungen technischer Änderungen qualitativ zu bewerten.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

30) Die Ergebnisse sind hinreichend genau, die Auswirkungen technischer Änderungen quantitativ zu bewerten.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

31) Die Ergebnisse sind hinreichend genau, um die rollenspezifischen Informationsbedarfe zu decken.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

32) Die Visualisierungsarten der Ergebnisse unterstützen die Genauigkeit.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

33) Die Ergebnisse sind hinreichend genau, um Handlungsmaßnahmen für den Umgang mit der technischen Änderung abzuleiten.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

Bild A-104: Validierungsfragebogen (VI/VIII)

ImPaKT Fragebogen zur Validierung der Methodik zur
Auswirkungsanalyse/Wirkkettenanalyse



Abschnitt 6: Mehrwert der Methodik-Anwendung

Bitte vergleichen Sie den Mehrwert der Methodik gegenüber Ihrem aktuellen Vorgehen zur Bewertung technischer Änderungen. Hierzu zählt der Ablauf innerhalb Ihres Änderungsmanagement, um technische Änderungen zu bewerten und um Entscheidungen zu treffen (beispielsweise in Änderungsgremien, Expertenworkshops oder dedizierten Freigabeprozessen)

34) Anhand der modellierten Informationen und Abfragen können betroffene Elemente einer technischen Änderung effizienter identifiziert werden als zuvor.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

35) Anhand der modellierten Informationen und Abfragen können betroffene Elemente einer technischen Änderung vollständiger identifiziert werden als zuvor.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

36) Anhand der quantitativen und qualitativen Ergebnisse können Auswirkungen einer technischen Änderung präziser bewertet werden als zuvor.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

37) Anhand der quantitativen und qualitativen Ergebnisse können Auswirkungen einer technischen Änderung vollständiger bewertet werden als zuvor.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

38) Durch die rollen-spezifischen Ergebnissichten werden die individuellen Informationsbedarfe zur Bewertung von technischen Änderungen bereitgestellt.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

39) Die frühzeitige Festlegung des Ziels der Auswirkungsanalyse hilft bei der effizienten Entwicklung der Abfragen und Metriken.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

40) Der Automatisierungsgrad der Methodik erfüllt meine Erwartungen an eine Auswirkungsanalyse.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

Bild A-105: Validierungsfragebogen (VII/VIII)

ImPaKT Fragebogen zur Validierung der Methodik zur
Auswirkungsanalyse/Wirkkettenanalyse



41) Auswirkungen unterschiedliche Änderungsarten, die in der Praxis vorkommen, können mit der Methodik bewertet werden.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

42) Die Auswirkungsanalysen können in den Änderungsprozess des Unternehmens integriert werden.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

43) Die Entstehung der resultierenden Ergebnisse ist nachvollziehbar und erklärbar.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

44) Die Ergebnisse können gespeichert und bereitgestellt werden.

-2 [Stimme nicht zu]	-1 [Stimme eher nicht zu]	0 [Stimme teilweise zu]	1 [Stimme eher zu]	2 [Stimme zu]

Bild A-106: Validierungsfragebogen (VIII/VIII)

A10.6 Auswertung der Fragebögen

Tabelle A-23: Auswertung der Fragebögen

These	Proband 1	Proband 2	Proband 3	Proband 4
T7)	-2	1	-1	-2
T8)	-1	0	-1	1
T9)	-1	0	-1	0
T10)	0	1	1	1
T11)	0	1	-1	1
T12)	1	1	0	-1
T13)	1	2	1	1
T14)	2	2	1	2
T15)	2	1	1	1
T16)	1	1	1	0
T17)	-1	-1	-2	-1
T18)	0	-1	1	1
T19)	-1	2	-1	0
T20)	1	0	1	2
T21)	-2	-1	-1	-2
T22)	0	-1	1	1
T23)	-2	0	-2	-2
T24)	0	0	1	0
T25)	-1	1	0	0
T26)	2	2	2	2
T27)	2	2	2	2
T28)	1	2	2	2
T29)	0	2	2	2
T30)	0	2	1	2
T31)	1	2	2	2
T32)	0	1	2	2
T33)	1	2	1	1
T34)	1	1	2	1
T35)	2	1	2	0
T36)	1	1	2	-1
T37)	2	1	2	-1
T38)	2	1	2	0
T39)	2	2	2	0
T40)	2	1	1	0
T41)	2	1	1	0

T42)	1	1	1	1
T43)	2	2	2	2
T44)	2	2	2	1

A11 Code

A11.1 Code zum Import des Metadatenmodells

```

:param {
// Define the file path root and the individual file names required for loading.
// https://neo4j.com/docs/operations-manual/current/configuration/file-locations/
file_path_root: 'file://', // Change this to the folder your script can access the files at.
file_0: '01__Requirement.csv',
file_1: '02__Function.csv',
file_2: '03__LogicalElement.csv',
file_3: '04__PhysicalElement.csv',
file_4: '05__Verification.csv',
file_5: '06__Validation.csv',
file_6: '07__ECR.csv',
file_7: 'Geometriemodell.csv',
file_8: 'Kalkulation_Pendel.csv',
file_9: 'Messtechnik.csv',
file_10: 'Formanlagen.csv',
file_11: '06_V to CR validates_relation.csv',
file_12: '05_V to SR verifies_relation.csv',
file_13: '01_R to R requires_relation.csv',
file_14: '01_R to R refines_relation.csv',
file_15: '01_R to R derive_relation.csv',
file_16: '02_F to R satisfies_relation.csv',
file_17: '02_F to F requires_relation.csv',
file_18: '02_F to F decompose_relation.csv',
file_19: '03_L to F specifies_relation.csv',
file_20: '03_L to L decompose_relation.csv',
file_21: '03_L to L requires_relation.csv',
file_22: '04_P to L realizes_relation.csv',
file_23: '04_P to P requires_relation.csv',
file_24: '04_P to P decompose_relation.csv',
file_25: '04_M to V Executes.csv',
file_26: '04_K to P calculates.csv',
file_27: '04_G to P implements.csv',
file_28: '04_F to P Produces.csv'
};

// CONSTRAINT creation
// -----
//
// Create node uniqueness constraints, ensuring no duplicates for the given node label and ID property exist in the
// database. This also ensures no duplicates are introduced in future.
//
//
// NOTE: The following constraint creation syntax is generated based on the current connected database version
// 5.15-aura.

```



```

CREATE CONSTRAINT `imp_uniq_Requirement_requirementID` IF NOT EXISTS
FOR (n: `Requirement`)
  REQUIRE (n.`requirementID`) IS UNIQUE;
CREATE CONSTRAINT `imp_uniq_LogicalElement_logicalID` IF NOT EXISTS
FOR (n: `LogicalElement`)
  REQUIRE (n.`logicalID`) IS UNIQUE;
CREATE CONSTRAINT `imp_uniq_PhysicalElement_physicalID` IF NOT EXISTS
FOR (n: `PhysicalElement`)
  REQUIRE (n.`physicalID`) IS UNIQUE;
CREATE CONSTRAINT `imp_uniq_Verification_verificationID` IF NOT EXISTS
FOR (n: `Verification`)
  REQUIRE (n.`verificationID`) IS UNIQUE;
CREATE CONSTRAINT `imp_uniq_Validation_validationID` IF NOT EXISTS
FOR (n: `Validation`)
  REQUIRE (n.`validationID`) IS UNIQUE;
CREATE CONSTRAINT `imp_uniq_Function_functionID` IF NOT EXISTS
FOR (n: `Function`)
  REQUIRE (n.`functionID`) IS UNIQUE;
CREATE CONSTRAINT `imp_uniq_ECR_ecrID` IF NOT EXISTS
FOR (n: `ECR`)
  REQUIRE (n.`ecrID`) IS UNIQUE;
CREATE CONSTRAINT `imp_uniq_Geometriemodell_ID` IF NOT EXISTS
FOR (n: `Geometriemodell`)
  REQUIRE (n.`ID`) IS UNIQUE;
CREATE CONSTRAINT `imp_uniq_Kalkulation_ID_Kalkulation` IF NOT EXISTS
FOR (n: `Kalkulation`)
  REQUIRE (n.`ID_Kalkulation`) IS UNIQUE;
CREATE CONSTRAINT `imp_uniq_Messtechnik_ID_Messen` IF NOT EXISTS
FOR (n: `Messtechnik`)
  REQUIRE (n.`ID_Messen`) IS UNIQUE;
CREATE CONSTRAINT `imp_uniq_Formanlagen_ID_Anlage` IF NOT EXISTS
FOR (n: `Formanlagen`)
  REQUIRE (n.`ID_Anlage`) IS UNIQUE;

:param {
  idsToSkip: []
};

// NODE load
// -----
//
// Load nodes in batches, one node label at a time. Nodes will be created using a MERGE statement to ensure a
// node with the same label and ID property remains unique. Pre-existing nodes found by a MERGE statement will
// have their other properties set to the latest values encountered in a load file.
//
// NOTE: Any nodes with IDs in the 'idsToSkip' list parameter will not be loaded.
LOAD CSV WITH HEADERS FROM ($file_path_root + $file_0) AS row
WITH row
WHERE NOT row.`requirementID` IN $idsToSkip AND NOT toInteger(trim(row.`requirementID`)) IS NULL
CALL {
  WITH row
  MERGE (n: `Requirement` { `requirementID`: toInteger(trim(row.`requirementID`)) })
  SET n.`requirementID` = toInteger(trim(row.`requirementID`))
  SET n.`requirementName` = row.`requirementName`
  SET n.`requirementText` = row.`requirementText`
  SET n.`requirementType` = row.`requirementType`
  SET n.`Requirement Parameter` = row.`Requirement Parameter`
  SET n.`Requirement Value Type` = row.`Requirement Value Type`
  SET n.`Requirement Priority` = row.`Requirement Priority`

```

```
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_2) AS row
WITH row
WHERE NOT row.`logicalID` IN $idsToSkip AND NOT toInteger(trim(row.`logicalID`)) IS NULL
CALL {
WITH row
MERGE (n: `LogicalElement` { `logicalID`: toInteger(trim(row.`logicalID`)) })
SET n.`logicalID` = toInteger(trim(row.`logicalID`))
SET n.`logicalName` = row.`logicalName`
SET n.`logicalParameter` = row.`logicalParameter`
SET n.`logicalParameterType` = row.`logicalParameterType`
SET n.`logicalType` = row.`logicalType`
SET n.`logicalVariantTag` = row.`logicalVariantTag`
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_3) AS row
WITH row
WHERE NOT row.`physicalID` IN $idsToSkip AND NOT toInteger(trim(row.`physicalID`)) IS NULL
CALL {
WITH row
MERGE (n: `PhysicalElement` { `physicalID`: toInteger(trim(row.`physicalID`)) })
SET n.`physicalID` = toInteger(trim(row.`physicalID`))
SET n.`physicalName` = row.`physicalName`
SET n.`Company` = row.`Company`
SET n.`physicalElementOwner` = row.`physicalElementOwner`
SET n.`physicalParameter` = row.`physicalParameter`
SET n.`physicalParameterType` = row.`physicalParameterType`
SET n.`Material` = row.`Material`
SET n.`physicalCosts` = toInteger(trim(row.`physicalCosts`))
SET n.`physicalVariantTag` = row.`physicalVariantTag`
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_4) AS row
WITH row
WHERE NOT row.`verificationID` IN $idsToSkip AND NOT toInteger(trim(row.`verificationID`)) IS NULL
CALL {
WITH row
MERGE (n: `Verification` { `verificationID`: toInteger(trim(row.`verificationID`)) })
SET n.`verificationID` = toInteger(trim(row.`verificationID`))
SET n.`verificationName` = row.`verificationName`
SET n.`verificatonMethod` = row.`verificatonMethod`
SET n.`verificationParameter` = toFloat(trim(row.`verificationParameter`))
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_5) AS row
WITH row
WHERE NOT row.`validationID` IN $idsToSkip
CALL {
WITH row
MERGE (n: `Validation` { `validationID`: toLower(trim(row.`validationID`)) IN ['1','true','yes'] })
SET n.`validationID` = toLower(trim(row.`validationID`)) IN ['1','true','yes']
SET n.`validationName` = row.`validationName`
SET n.`validationMethod` = row.`validationMethod`
SET n.`validationParameter` = row.`validationParameter`
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_1) AS row
WITH row
```

```

WHERE NOT row.`functionID` IN $idsToSkip AND NOT toInteger(trim(row.`functionID`)) IS NULL
CALL {
WITH row
MERGE (n: `Function` { `functionID`: toInteger(trim(row.`functionID`)) })
SET n.`functionID` = toInteger(trim(row.`functionID`))
SET n.`functionName` = row.`functionName`
SET n.`functionVariantTag` = row.`functionVariantTag`
SET n.`functionType` = row.`functionType`
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_6) AS row
WITH row
WHERE NOT row.`ecrID` IN $idsToSkip
CALL {
WITH row
MERGE (n: `ECR` { `ecrID`: toLower(trim(row.`ecrID`)) IN ['1','true','yes'] })
SET n.`ecrID` = toLower(trim(row.`ecrID`)) IN ['1','true','yes']
SET n.`ercName` = row.`ercName`
SET n.`ecrGrund` = row.`ecrGrund`
SET n.`ecrArtderAenderung` = row.`ecrArtderAenderung`
SET n.`ecrAnsprechpartner` = row.`ecrAnsprechpartner`
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_7) AS row
WITH row
WHERE NOT row.`ID` IN $idsToSkip AND NOT toInteger(trim(row.`ID`)) IS NULL
CALL {
WITH row
MERGE (n: `Geometriemodell` { `ID`: toInteger(trim(row.`ID`)) })
SET n.`ID` = toInteger(trim(row.`ID`))
SET n.`GeometrischeAbmessungen` = row.`GeometrischeAbmessungen`
SET n.`Wandstaerken` = row.`Wandstaerken`
SET n.`GeometrischeToleranzen` = row.`GeometrischeToleranzen`
SET n.`Gussteilgeometrie` = row.`Gussteilgeometrie`
SET n.`Materialauswahl` = row.`Materialauswahl`
SET n.`Dichte` = row.`Dichte`
SET n.`Zugfestigkeit` = row.`Zugfestigkeit`
SET n.`Streckgrenze` = row.`Streckgrenze`
SET n.`Bruchdehung` = row.`Bruchdehung`
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_8) AS row
WITH row
WHERE NOT row.`ID_Kalkulation` IN $idsToSkip AND NOT toInteger(trim(row.`ID_Kalkulation`)) IS NULL
CALL {
WITH row
MERGE (n: `Kalkulation` { `ID_Kalkulation`: toInteger(trim(row.`ID_Kalkulation`)) })
SET n.`ID_Kalkulation` = toInteger(trim(row.`ID_Kalkulation`))
SET n.`Kalkulationsname` = row.`Kalkulationsname`
SET n.`Werkstoff` = row.`Werkstoff`
SET n.`Stueckgewicht` = toInteger(trim(row.`Stueckgewicht`))
SET n.`Kreislauf` = toInteger(trim(row.`Kreislauf`))
SET n.`Kastengewicht` = toInteger(trim(row.`Kastengewicht`))
SET n.`Ausbringung` = toFloat(trim(row.`Ausbringung`))
SET n.`Kosten_Schmelze` = toFloat(trim(row.`Kosten_Schmelze`))
SET n.`FA 1` = toInteger(trim(row.`FA 1`))
SET n.`Ruesten` = toInteger(trim(row.`Ruesten`))
SET n.`Kasten_L` = toInteger(trim(row.`Kasten_L`))
SET n.`Kasten_B` = toInteger(trim(row.`Kasten_B`))

```

```

SET n.`Kasten_H` = toInteger(trim(row.`Kasten_H`))
SET n.`Volumen` = toInteger(trim(row.`Volumen`))
SET n.`Kosten_Formen` = toFloat(trim(row.`Kosten_Formen`))
SET n.`Kerne` = toInteger(trim(row.`Kerne`))
SET n.`Kosten Kerne` = toInteger(trim(row.`Kosten Kerne`))
SET n.`Speiser` = toInteger(trim(row.`Speiser`))
SET n.`Impfmittel` = toInteger(trim(row.`Impfmittel`))
SET n.`Kokillen` = toInteger(trim(row.`Kokillen`))
SET n.`Filter` = toInteger(trim(row.`Filter`))
SET n.`Kosten_Putzen` = toFloat(trim(row.`Kosten_Putzen`))
SET n.`Kosten_Metall` = toFloat(trim(row.`Kosten_Metall`))
SET n.`sonstige_Kosten` = toInteger(trim(row.`sonstige_Kosten`))
SET n.`Gesamtkosten` = toFloat(trim(row.`Gesamtkosten`))
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_9) AS row
WITH row
WHERE NOT row.`ID_Messen` IN $idsToSkip AND NOT toInteger(trim(row.`ID_Messen`)) IS NULL
CALL {
WITH row
MERGE (n: `Messtechnik` { `ID_Messen`: toInteger(trim(row.`ID_Messen`)) })
SET n.`ID_Messen` = toInteger(trim(row.`ID_Messen`))
SET n.`Messart` = row.`Messart`
SET n.`Messverfahren` = row.`Messverfahren`
SET n.`Zertifikatsstufe` = toInteger(trim(row.`Zertifikatsstufe`))
SET n.`Level` = toInteger(trim(row.`Level`))
SET n.`Maschine` = row.`Maschine`
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_10) AS row
WITH row
WHERE NOT row.`ID_Anlage` IN $idsToSkip AND NOT toInteger(trim(row.`ID_Anlage`)) IS NULL
CALL {
WITH row
MERGE (n: `Formanlagen` { `ID_Anlage`: toInteger(trim(row.`ID_Anlage`)) })
SET n.`ID_Anlage` = toInteger(trim(row.`ID_Anlage`))
SET n.`Anlagenname` = row.`Anlagenname`
SET n.`Hoehe` = toInteger(trim(row.`Hoehe`))
SET n.`Breite` = toInteger(trim(row.`Breite`))
SET n.`Laenge1` = row.`Laenge1`
SET n.`Laenge2` = row.`Laenge2`
} IN TRANSACTIONS OF 10000 ROWS;

// RELATIONSHIP load
// -----
//
// Load relationships in batches, one relationship type at a time. Relationships are created using a MERGE state-
// ment, meaning only one relationship of a given type will ever be created between a pair of nodes.
LOAD CSV WITH HEADERS FROM ($file_path_root + $file_22) AS row
WITH row
CALL {
WITH row
MATCH (source: `PhysicalElement` { `physicalID`: toInteger(trim(row.`physicalID`)) })
MATCH (target: `LogicalElement` { `logicalID`: toInteger(trim(row.`logicalID_realizes`)) })
MERGE (source)-[: `Realizes`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_15) AS row

```

```
WITH row
CALL {
WITH row
MATCH (source: `Requirement` { `requirementID`: toInteger(trim(row.`requirementID`)) })
MATCH (target: `Requirement` { `requirementID`: toInteger(trim(row.`requirementID_derives`)) })
MERGE (source)-[r: `Derives`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_12) AS row
WITH row
CALL {
WITH row
MATCH (source: `Verification` { `verificationID`: toInteger(trim(row.`verificationID`)) })
MATCH (target: `Requirement` { `requirementID`: toInteger(trim(row.`systemrequirementID_verifies`)) })
MERGE (source)-[r: `Verifies`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_21) AS row
WITH row
CALL {
WITH row
MATCH (source: `LogicalElement` { `logicalID`: toInteger(trim(row.`logicalID`)) })
MATCH (target: `LogicalElement` { `logicalID`: toInteger(trim(row.`logicalID_requires`)) })
MERGE (source)-[r: `Requires`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_23) AS row
WITH row
CALL {
WITH row
MATCH (source: `PhysicalElement` { `physicalID`: toInteger(trim(row.`physicalID`)) })
MATCH (target: `PhysicalElement` { `physicalID`: toInteger(trim(row.`physicalID_requires`)) })
MERGE (source)-[r: `Requires`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_14) AS row
WITH row
CALL {
WITH row
MATCH (source: `Requirement` { `requirementID`: toInteger(trim(row.`requirementID`)) })
MATCH (target: `Requirement` { `requirementID`: toInteger(trim(row.`requirementID_refines`)) })
MERGE (source)-[r: `Refines`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_13) AS row
WITH row
CALL {
WITH row
MATCH (source: `Requirement` { `requirementID`: toInteger(trim(row.`requirementID`)) })
MATCH (target: `Requirement` { `requirementID`: toInteger(trim(row.`requirementID_requires`)) })
MERGE (source)-[r: `Requires`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_11) AS row
WITH row
CALL {
WITH row
MATCH (source: `Validation` { `validationID`: toLower(trim(row.`validationID`)) IN ['1','true','yes'] })
MATCH (target: `Requirement` { `requirementID`: toInteger(trim(row.`customerrequirementID_validates`)) })
```

```
MERGE (source)-[r: `Validates`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_24) AS row
WITH row
CALL {
WITH row
MATCH (source: `PhysicalElement` { `physicalID`: toInteger(trim(row.`physicalID`)) })
MATCH (target: `PhysicalElement` { `physicalID`: toInteger(trim(row.`physicalID_decompose`)) })
MERGE (source)-[r: `Decompose`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_20) AS row
WITH row
CALL {
WITH row
MATCH (source: `LogicalElement` { `logicalID`: toInteger(trim(row.`logicalID`)) })
MATCH (target: `LogicalElement` { `logicalID`: toInteger(trim(row.`logicalID_decompose`)) })
MERGE (source)-[r: `Decompose`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_16) AS row
WITH row
CALL {
WITH row
MATCH (source: `Function` { `functionID`: toInteger(trim(row.`functionID`)) })
MATCH (target: `Requirement` { `requirementID`: toInteger(trim(row.`requirementID_satisfies`)) })
MERGE (source)-[r: `Satisfies`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_19) AS row
WITH row
CALL {
WITH row
MATCH (source: `LogicalElement` { `logicalID`: toInteger(trim(row.`logicalID`)) })
MATCH (target: `Function` { `functionID`: toInteger(trim(row.`functionID_relation`)) })
MERGE (source)-[r: `Specifies`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_17) AS row
WITH row
CALL {
WITH row
MATCH (source: `Function` { `functionID`: toInteger(trim(row.`functionID`)) })
MATCH (target: `Function` { `functionID`: toInteger(trim(row.`functionID_requires`)) })
MERGE (source)-[r: `Requires`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_18) AS row
WITH row
CALL {
WITH row
MATCH (source: `Function` { `functionID`: toInteger(trim(row.`functionID`)) })
MATCH (target: `Function` { `functionID`: toInteger(trim(row.`functionID_decompose`)) })
MERGE (source)-[r: `Decompose`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_25) AS row
WITH row
```

```

CALL {
  WITH row
  MATCH (source: `Messtechnik` { `ID_Messen`: toInteger(trim(row.`ID_Messen`)) })
  MATCH (target: `Verification` { `verificationID`: toInteger(trim(row.`VerificationID`)) })
  MERGE (source)-[r: `Executes`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_26) AS row
WITH row
CALL {
  WITH row
  MATCH (source: `Kalkulation` { `ID_Kalkulation`: toInteger(trim(row.`KalkulationsID`)) })
  MATCH (target: `PhysicalElement` { `physicalID`: toInteger(trim(row.`physicalID`)) })
  MERGE (source)-[r: `Calculates`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_27) AS row
WITH row
CALL {
  WITH row
  MATCH (source: `Geometriemodell` { `ID`: toInteger(trim(row.`ID`)) })
  MATCH (target: `PhysicalElement` { `physicalID`: toInteger(trim(row.`physicalID`)) })
  MERGE (source)-[r: `Implements`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

LOAD CSV WITH HEADERS FROM ($file_path_root + $file_28) AS row
WITH row
CALL {
  WITH row
  MATCH (source: `Formanlagen` { `ID_Anlage`: toInteger(trim(row.`ID`)) })
  MATCH (target: `PhysicalElement` { `physicalID`: toInteger(trim(row.`physicalID`)) })
  MERGE (source)-[r: `Produces`]->(target)
} IN TRANSACTIONS OF 10000 ROWS;

```

A11.2 Dashboards Roboterarm

Dashboard

```

{
  "title": "RFLPV2",
  "version": "2.4",
  "settings": {
    "pagenumber": 6,
    "editable": true,
    "fullscreenEnabled": false,
    "parameters": {
      "neodash_requirement_requirementname": "Gestalt des Greifobjekts",
      "neodash_requirement_requirementname_display": "Gestalt des Greifobjekts",

```

```

"neodash_physicalelement_physicalname": "Standardgreifer",
"neodash_physicalelement_physicalname_display": "Standardgreifer",
"neodash_physicalelement_physicalname_2": "Grosser Greifer",
"neodash_physicalelement_physicalname_2_display": "Grosser Greifer"
},
"theme": "light",
"disableRowLimiting": true,
"downloadImageEnabled": false
},
"pages": [
{
"title": "Konstruktion",
"reports": [
{
"id": "001d2e1d-9d0c-4454-a09f-11efb50dac35",
"title": "initiale Anforderung",
"query": "MATCH (n:`Requirement`) \nWHERE toLower(toString(n.`requirementName`))
CONTAINS toLower($input) \nRETURN DISTINCT n.`requirementName` as value, n.`requirement-
Name` as display ORDER BY size(toString(value)) ASC LIMIT 100",
"width": 5,
"height": 2,
"x": 0,
"y": 0,
"type": "select",
"selection": {},
"settings": {
"type": "Node Property",
"entityType": "Requirement",
"propertyType": "requirementName",
"propertyTypeDisplay": "requirementName",
"parameterName": "neodash_requirement_requirementname",
"suggestionLimit": 100
},
"schema": []
},
{
"id": "2553382b-75e7-4809-8f22-6a9db1e9f95b",
"title": "Graphausschnitt",
"query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derive]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalEle-
ment)<-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN r,
rel1, r2, rel4, f, rel7, l, rel10, p",

```



```
"width": 8,
"height": 6,
"x": 5,
"y": 0,
"type": "graph",
"selection": {
  "Requirement": "(label)",
  "Derive": "(label)",
  "Satisfies": "(label)",
  "Function": "(label)",
  "Specifies": "(label)",
  "LogicalElement": "(label)",
  "Realizes": "(label)",
  "PhysicalElement": "(label)"
},
"settings": {
  "enableEditing": false,
  "showPropertiesOnHover": true,
  "hideSelections": true,
  "nodeLabelFontSize": 3,
  "defaultNodeSize": 3,
  "nodeColorScheme": "pastel1",
  "relLabelFontSize": 0
},
"schema": [
  [
    "Requirement",
    "requirementName",
    "requirementText",
    "RequirementPriority",
    "requirementID",
    "requirementType",
    "RequirementValueType",
    "RequirementParameter"
  ],
  [
    "Derive"
  ],
  [
    "Satisfies"
  ],
  [
```

```
"Function",
"functionID",
"functionName",
"functionVariantTag",
"functionSafetyCritical",
"functionType"
],
[
  "Specifies"
],
[
  "LogicalElement",
  "logicalType",
  "logicalID",
  "logicalName",
  "logicalVariantTag",
  "Variant Tag"
],
[
  "Realizes"
],
[
  "PhysicalElement",
  "physicalName",
  "physicalDiscipline",
  "physicalID",
  "physicalType",
  "physicalElementOwner",
  "physicalParameter_2",
  "physicalVariantTag",
  "physicalParameterType_2",
  "physicalCosts",
  "physicalParameterType",
  "physicalParameter",
  "physicalQuantity"
]
]
},
{
  "id": "e78d4c72-9ec1-40d8-9b3f-8df5913f0fdd",
  "title": "betroffene mechanische Bauteile",
```

```

    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derive]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEle-
ment)-[rel10:Realizes]-(p:PhysicalElement)-[rel12_Decompose]->(p2:PhysicalElement)\nWHERE
p.physicalDiscipline = 'Mechanic' AND (p:PhysicalElement OR p2:PhysicalElement)\nWITH
COUNT(DISTINCT p) AS affectedPhysicalElements_p, COUNT(DISTINCT p2) AS affectedPhysical-
Elements_p2\nRETURN affectedPhysicalElements_p + affectedPhysicalElements_p2 AS totalAffec-
tedPhysicalElements\n",
    "width": 5,
    "height": 2,
    "x": 0,
    "y": 4,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 40,
      "textAlign": "left",
      "verticalAlign": "middle"
    },
    "schema": []
  },
  {
    "id": "f8aaf9d1-e054-4128-b866-357b59fbbf54",
    "title": "betroffene logische Elemente",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derive]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEle-
ment)-[rel10:Realizes]-(p:PhysicalElement)-[rel12_Decompose]->(p2:PhysicalElement)\nRETURN
COUNT(DISTINCT l) AS betroffenelogischeElemente\n",
    "width": 5,
    "height": 2,
    "x": 0,
    "y": 2,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 40,
      "verticalAlign": "middle"
    },
    "schema": []
  },
  {
    "id": "298d689c-921c-4b86-becf-60a462776006",
    "title": "betroffene physische Elemente",

```

```

    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[:Derive]->(:Requirement)<-[:Satisfies]-(:Function)<-[:Specifies]-(:LogicalElement)<-[:Real-
izes]-(p:PhysicalElement)\nRETURN p.physicalID AS ID,p.physicalDiscipline AS Bereich, p.physical-
Name AS Name, p.physicalType AS Typ\n\nUNION\n\nMATCH (r:Requirement {requirementName:
$neodash_requirement_requirementname})-[:Derive]->(:Requirement)<-[:Satisfies]-(:Function)<-
[:Specifies]-(:LogicalElement)<-[:Realizes]-(p:PhysicalElement)-[:Decompose]->(p2:PhysicalEle-
ment)\nRETURN p2.physicalID AS ID, p2.physicalDiscipline AS Bereich, p2.physicalName AS
Name, p2.physicalType AS Typ\n",
    "width": 8,
    "height": 3,
    "x": 13,
    "y": 0,
    "type": "table",
    "selection": {},
    "settings": {
      "compact": false,
      "columnWidthsType": "Relative (%)",
      "columnWidths": "[10,30,40,20]"
    }
  },
  {
    "id": "737641bc-03cd-4347-8120-b9f901d948be",
    "title": "V&V Planung",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[:Derive]->(r2:Requirement)<-[:Satisfies]-(f:Function)<-[:Specifies]-(l:LogicalElement)<-
[:Realizes]-(p:PhysicalElement)-[:Decompose]->(p2:PhysicalElement)\nMATCH (ver:Verification)-
[:Verifies]->(r2)\nRETURN DISTINCT ver.verificationName AS Name, r2.requirementName AS `be-
troffene Anforderung`, 'Verifikation' AS Methode\n\nUNION \n\nMATCH (val:Validation)-[:Vali-
dates]->(r)\nRETURN DISTINCT val.validationName AS Name, r.requirementName AS `betroffene
Anforderung`, 'Validierung' AS Methode\n",
    "width": 8,
    "height": 3,
    "x": 13,
    "y": 3,
    "type": "table",
    "selection": {},
    "settings": {
      "columnWidths": "[40,40,20]"
    }
  }
]
},

```

```

{
  "title": "Projektmanagement",
  "reports": [
    {
      "id": "e462e5d8-a80f-4eee-96d7-0b5ef7b35b40",
      "title": "Graphausschnitt",
      "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derive]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEle-
ment)-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN r,
rel1, r2, rel4, f, rel7, l, rel10, p",
      "width": 8,
      "height": 6,
      "x": 4,
      "y": 0,
      "type": "graph",
      "selection": {
        "Requirement": "(label)",
        "Derive": "(label)",
        "Satisfies": "(label)",
        "Function": "(label)",
        "Specifies": "(label)",
        "LogicalElement": "(label)",
        "Realizes": "(label)",
        "PhysicalElement": "(label)"
      },
      "settings": {
        "hideSelections": true,
        "nodeLabelFontSize": 3,
        "defaultNodeSize": 3,
        "nodeColorScheme": "pastel1",
        "relLabelFontSize": 0,
        "showPropertiesOnHover": true
      },
      "schema": [
        "Requirement",
        "requirementName",
        "requirementText",
        "RequirementPriority",
        "requirementID",
        "requirementType",
        "RequirementValueType",

```

```
"RequirementParameter"
],
[
  "Derive"
],
[
  "Satisfies"
],
[
  "Function",
  "functionID",
  "functionName",
  "functionVariantTag",
  "functionSafetyCritical",
  "functionType"
],
[
  "Specifies"
],
[
  "LogicalElement",
  "logicalType",
  "logicalID",
  "logicalName",
  "logicalVariantTag",
  "Variant Tag"
],
[
  "Realizes"
],
[
  "PhysicalElement",
  "physicalName",
  "physicalDiscipline",
  "physicalID",
  "physicalType",
  "physicalElementOwner",
  "physicalParameter_2",
  "physicalVariantTag",
  "physicalParameterType_2",
  "physicalCosts",
  "physicalParameterType",
```

```

        "physicalParameter",
        "physicalQuantity"
    ]
]
},
{
    "id": "eae684be-b403-4c97-ba4d-e8d42144152e",
    "title": "betroffene logische Elemente ",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derive]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEle-
ment)-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN
COUNT(DISTINCT l) AS betroffenelogischeElemente\n\n\n",
    "width": 4,
    "height": 2,
    "x": 0,
    "y": 2,
    "type": "value",
    "selection": {},
    "settings": {
        "fontSize": 40
    },
    "schema": []
},
{
    "id": "7b4dff59-9300-4e78-986b-bc7c4215a939",
    "title": "betroffene physische Elemente",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derive]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEle-
ment)-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement)\nWITH
COUNT(DISTINCT p) AS affectedPhysicalElements_p, COUNT(DISTINCT p2) AS affectedPhysical-
Elements_p2\nRETURN affectedPhysicalElements_p + affectedPhysicalElements_p2 AS totalAffec-
tedPhysicalElements\n\n\n",
    "width": 4,
    "height": 2,
    "x": 0,
    "y": 4,
    "type": "value",
    "selection": {},
    "settings": {
        "fontSize": 40
    },
    "schema": []
}

```

```

    },
    {
      "id": "d795611c-fbc0-4b1d-886f-74e39a40f754",
      "title": "initiale Anforderung",
      "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derive]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEle-
ment)-[rel10:Realizes]-(p:PhysicalElement)-[rel12_Decompose]->(p2:PhysicalElement)\nRETURN
DISTINCT r.requirementName\n\n\n",
      "width": 4,
      "height": 2,
      "x": 0,
      "y": 0,
      "type": "value",
      "selection": {},
      "settings": {
        "fontSize": 32
      },
    },
    "schema": []
  },
  {
    "id": "63a64a10-0ce9-4859-9745-a7d7210b4891",
    "title": "betroffene Systemanforderungen",
    "query": "MATCH (r:Requirement {requirementName:$neodash_requirement_requirement-
name})-[rel1:Derive]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEle-
ment)-[rel10:Realizes]-(p:PhysicalElement)-[rel12_Decompose]->(p2:PhysicalElement)\n\n\n\nRE-
TURN DISTINCT r2.`Requirement Priority` as Anforderungspriorität, (r2.requirementName) as `be-
troffene Systemanforderungen`\n\n\n",
    "width": 7,
    "height": 3,
    "x": 12,
    "y": 0,
    "type": "table",
    "selection": {},
    "settings": {
      "fontSize": 40,
      "columnWidths": "[30,70]",
      "compact": true
    },
    "schema": []
  },
  {
    "id": "4eb80b6f-3dac-4c50-bdb5-aa31df6fce1f",

```



```

    "title": "betroffene physische Elemente",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[:Derive]->(:Requirement)<-[:Satisfies]-(:Function)<-[:Specifies]-(:LogicalElement)<-[:Real-
izes]-(p:PhysicalElement)\nRETURN p.physicalID AS ID,p.physicalDiscipline AS Bereich, p.physical-
Name AS Name, p.physicalType AS Type\n\nUNION\n\nMATCH (r:Requirement {requirementName:
$neodash_requirement_requirementname})-[:Derives]->(:Requirement)<-[:Satisfies]-(:Function)<-
[:Specifies]-(:LogicalElement)<-[:Realizes]-(p:PhysicalElement)-[:Decompose]->(p2:PhysicalEle-
ment)\nRETURN p2.physicalID AS ID, p2.physicalDiscipline AS Bereich, p2.physicalName AS
Name, p2.physicalType AS Type\n\n\n",
    "width": 7,
    "height": 3,
    "x": 12,
    "y": 3,
    "type": "table",
    "selection": {},
    "settings": {
      "columnWidths": "[1,33,33,33]",
      "compact": true
    }
  }
]
},
{
  "title": "Vertrieb",
  "reports": [
    {
      "id": "2830ebfb-27ad-412a-a1a5-0ab61ad77a7a",
      "title": "initiale Anforderung",
      "query": "MATCH (n:`Requirement`) \nWHERE toLower(toString(n.`requirementName`))
CONTAINS toLower($input) \nRETURN DISTINCT n.`requirementName` as value, n.`requirement-
Name` as display ORDER BY size(toString(value)) ASC LIMIT 100",
      "width": 4,
      "height": 2,
      "x": 0,
      "y": 0,
      "type": "select",
      "selection": {},
      "settings": {
        "type": "Node Property",
        "entityType": "Requirement",
        "propertyType": "requirementName",
        "propertyTypeDisplay": "requirementName",

```

```

    "parameterName": "neodash_requirement_requirementname",
    "suggestionLimit": 100
  },
  "schema": []
},
{
  "id": "d08b9287-4eb5-42fa-a9a4-ef58f77f9729",
  "title": "Anforderungstext",
  "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[:Derive]->(:Requirement)\nRETURN r.requirementText\n\n",
  "width": 8,
  "height": 2,
  "x": 0,
  "y": 2,
  "type": "value",
  "selection": {},
  "settings": {
    "fontSize": 20,
    "verticalAlign": "middle",
    "format": "auto"
  },
  "schema": []
},
{
  "id": "692a1107-9eef-4fb4-9060-56c8125d4afc",
  "title": "Anforderungspriorität",
  "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derive]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEle-
ment)-[rel10:Realizes]-(p:PhysicalElement)-[rel12_Decompose]->(p2:PhysicalElement)\nRETURN
MAX(r2.`RequirementPriority`) AS maxPriority\n",
  "width": 4,
  "height": 2,
  "x": 4,
  "y": 0,
  "type": "value",
  "selection": {},
  "settings": {
    "verticalAlign": "middle",
    "fontSize": 40
  },
  "schema": []
},

```

```

{
  "id": "e198e7cb-9236-464a-af15-076e4aebcf31",
  "title": "Sind Elektronikkomponenten betroffen?",
  "query": "MATCH (:Requirement {requirementName: $neodash_requirement_requirement-
name})-[:Derive]->(:Requirement)<-[:Satisfies]-(:Function)<-[:Specifies]-(:LogicalElement)<-[:Real-
izes]-(:PhysicalElement)-[:Decompose]->(p:PhysicalElement)\nWHERE    p.physicalDiscipline    =
'Electronic / Electric'\nRETURN CASE WHEN COUNT(DISTINCT p) > 0 THEN '✓' ELSE '✗'
END AS Status\n\n\n",
  "width": 5,
  "height": 2,
  "x": 8,
  "y": 0,
  "type": "value",
  "selection": {},
  "settings": {
    "textAlign": "center",
    "verticalAlign": "middle"
  },
  "schema": []
},
{
  "id": "ce51fd6b-9ddc-4e66-b3bf-5cfeff0b3c75",
  "title": "Sind mechanische Komponenten betroffen?",
  "query": "MATCH (:Requirement {requirementName: $neodash_requirement_requirement-
name})-[:Derive]->(:Requirement)<-[:Satisfies]-(:Function)<-[:Specifies]-(:LogicalElement)<-[:Real-
izes]-(:PhysicalElement)-[:Decompose]->(p2:PhysicalElement)\nWHERE p.physicalDiscipline = 'Me-
chanic'\nRETURN CASE WHEN COUNT(DISTINCT p) > 0 THEN '✓' ELSE '✗' END AS Sta-
tus\n",
  "width": 5,
  "height": 2,
  "x": 8,
  "y": 6,
  "type": "value",
  "selection": {},
  "settings": {
    "textAlign": "center",
    "verticalAlign": "middle"
  },
  "schema": []
}
]
},

```

```
{
  "title": "Anforderungsentwickler",
  "reports": [
    {
      "id": "3795185e-0fc5-411b-8b7f-9460cc8975fb",
      "title": "Betroffene Anforderungen ",
      "query": "MATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})-[rel1_2_3:Derive|Requires|Refines]->(r2:Requirement)\nRETURN r, rel1_2_3, r2\n\n",
      "width": 6,
      "height": 4,
      "x": 0,
      "y": 6,
      "type": "graph",
      "selection": {
        "Requirement": "(label)",
        "Requires": "(label)",
        "Refines": "(label)"
      },
      "settings": {
        "hideSelections": true,
        "showPropertiesOnHover": true
      },
      "schema": [
        [
          "Requirement",
          "requirementName",
          "requirementText",
          "requirementCategory",
          "requirementID",
          "requirementType",
          "requirementPriority"
        ],
        [
          "Requires"
        ],
        [
          "Refines"
        ]
      ]
    },
    {
      "id": "a76e0800-0470-4fbf-8e7e-18c3df8a03c1",
```

```

    "title": "initial Anforderung",
    "query": "MATCH (n:`Requirement`) \nWHERE toLower(toString(n.`requirementName`))
CONTAINS toLower($input) \nRETURN DISTINCT n.`requirementName` as value, n.`requirement-
Name` as display ORDER BY size(toString(value)) ASC LIMIT 100",
    "width": 4,
    "height": 2,
    "x": 0,
    "y": 0,
    "type": "select",
    "selection": {},
    "settings": {
        "type": "Node Property",
        "entityType": "Requirement",
        "propertyType": "requirementName",
        "propertyTypeDisplay": "requirementName",
        "parameterName": "neodash_requirement_requirementname",
        "suggestionLimit": 100
    },
    "schema": []
},
{
    "id": "c15027a6-7136-40f6-a2a5-51c33bcdeb33",
    "title": "Anforderungstext",
    "query": "MATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)\nRETURN r2.requirementText\n\n\n",
    "width": 12,
    "height": 2,
    "x": 0,
    "y": 2,
    "type": "value",
    "selection": {},
    "settings": {
        "fontSize": 20,
        "monospace": false
    },
    "schema": []
},
{
    "id": "6496c0c5-7d03-46eb-81b9-6b17dce1135a",
    "title": "Anforderungspriorität",
    "query": "MATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})\nRETURN r.requirementPriority\n",

```

```

    "width": 4,
    "height": 2,
    "x": 4,
    "y": 0,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 32
    },
    "schema": []
  },
  {
    "id": "594834df-d2d2-44d0-a2e7-6442a5bb484c",
    "title": "Anforderungsparameter",
    "query": "MATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})-[:Derives]-(r2:Requirement)\nRETURN r2.requirementParameter\n\n\n",
    "width": 4,
    "height": 2,
    "x": 8,
    "y": 0,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 32
    },
    "schema": []
  },
  {
    "id": "73bb4dc2-8b3a-494b-aaa8-0d6987d3485f",
    "title": "Beschreibung der betroffenen Anforderungen",
    "query": "MATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})-[:rel1_2_3:Derive|Requires|Refines]->(r2:Requirement)\nRETURN r2.requirementName AS
Anforderungsname, r2.requirementPriority AS Anforderungspriorität, r2.requirementCategory AS An-
forderungskategorie\n\n\n",
    "width": 6,
    "height": 4,
    "x": 6,
    "y": 6,
    "type": "table",
    "selection": {},
    "settings": {}
  },

```

```

{
  "id": "31e7642e-b5a8-4dc5-ae31-903c6ca51b15",
  "title": "Betroffene Anforderungen",
  "query": "-----\n\n\n",
  "width": 12,
  "height": 2,
  "x": 0,
  "y": 4,
  "type": "text",
  "selection": {},
  "settings": {},
  "schema": []
}
],
{
  "title": "Systemarchitekt",
  "reports": [
    {
      "id": "e5d81d3b-2187-45ac-9aa9-3a3d7af93661",
      "title": "Graphausschnitt",
      "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEl-
ement)-[rel10:Realizes]-(p:PhysicalElement)\nRETURN r, rel1, r2, rel4, f, rel7, l, rel10, p\n\n\n",
      "width": 4,
      "height": 4,
      "x": 4,
      "y": 0,
      "type": "graph",
      "selection": {
        "Requirement": "(label)",
        "Derives": "(label)",
        "Satisfies": "(label)",
        "Function": "(label)",
        "Specifies": "(label)",
        "LogicalElement": "(label)",
        "Realizes": "(label)",
        "PhysicalElement": "(label)"
      },
      "settings": {
        "hideSelections": true,

```

```
"nodeLabelFontSize": 3,
"defaultNodeSize": 3,
"showPropertiesOnHover": true,
"nodeColorScheme": "pastel1"
},
"schema": [
  [
    "Requirement",
    "requirementName",
    "requirementText",
    "requirementCategory",
    "requirementID",
    "requirementType",
    "requirementPriority",
    "requirementValueType",
    "requirementParameter"
  ],
  [
    "Derives"
  ],
  [
    "Satisfies"
  ],
  [
    "Function",
    "functionID",
    "functionName",
    "functionVariantTag",
    "functionSafetyCritical",
    "functionType"
  ],
  [
    "Specifies"
  ],
  [
    "LogicalElement",
    "logicalType",
    "logicalID",
    "logicalName",
    "logicalVariantTag",
    "Variant Tag"
  ],
]
```



```

[
  "Realizes"
],
[
  "PhysicalElement",
  "physicalName",
  "physicalDiscipline",
  "physicalID",
  "physicalType",
  "physicalElementOwner",
  "physicalParameter_2",
  "physicalVariantTag",
  "physicalQuantity",
  "physicalParameterType_2",
  "physicalCosts",
  "physicalParameterType",
  "physicalParameter"
]
],
},
{
  "id": "0e08fbcc-2e3a-4e90-b092-616b54bc3727",
  "title": "# betroffener Anforderungen",
  "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEl-
ement)-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN
COUNT(DISTINCT r2) AS betroffeneSystemanforderungen\n\n\n",
  "width": 4,
  "height": 2,
  "x": 0,
  "y": 0,
  "type": "value",
  "selection": {},
  "settings": {
    "fontSize": 32,
    "textAlign": "left",
    "verticalAlign": "middle"
  },
  "schema": []
},
{
  "id": "4090ce5c-f523-49e8-a5eb-21e8c1764710",

```

```

    "title": "# betroffener Funktionen",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEl-
ement)-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN
COUNT(DISTINCT f) AS betroffeneFunktionen\n\n\n",
    "width": 4,
    "height": 2,
    "x": 0,
    "y": 2,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 32,
      "verticalAlign": "middle"
    },
  },
  "schema": []
},
{
  "id": "faeddaf9-41c4-4bdb-8bb6-cd5db93e84c7",
  "title": "# betroffener log. Elemente",
  "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEl-
ement)-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN
COUNT(DISTINCT l) AS betroffenelogischeElemente\n\n\n",
  "width": 4,
  "height": 2,
  "x": 8,
  "y": 0,
  "type": "value",
  "selection": {},
  "settings": {
    "fontSize": 32,
    "verticalAlign": "middle"
  },
  "schema": []
},
{
  "id": "43383a58-7cce-4474-a563-f5acd616fa75",
  "title": "# betroffener phy. Elemente ",
  "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-

```

```

(l:LogicalElement)<-[rel10:Realizes]-(p:PhysicalElement)\nRETURN COUNT(DISTINCT p) AS be-
troffenephysischeElemente\n\n\n",
    "width": 4,
    "height": 2,
    "x": 8,
    "y": 2,
    "type": "value",
    "selection": {},
    "settings": {
        "fontSize": 32,
        "verticalAlign": "middle"
    },
    "schema": []
},
{
    "id": "454e8821-f546-4456-b3e0-4eed76816b27",
    "title": "Betroffene Funktionen",
    "query": "-----\n\n\n",
    "width": 12,
    "height": 2,
    "x": 0,
    "y": 4,
    "type": "text",
    "selection": {},
    "settings": {},
    "schema": []
},
{
    "id": "8c7cb0c1-55a4-4593-9098-fc1e84e30817",
    "title": "sicherheitsrelevante Funktionen",
    "query": "\nMATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalEl-
ement)<-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement)\nWHERE
f.functionSafetyCritical = true\nRETURN DISTINCT f.functionName as Funktionsname, f.functionID
AS ID\n\n\n",
    "width": 5,
    "height": 3,
    "x": 7,
    "y": 6,
    "type": "table",
    "selection": {}
}

```

```

"settings": {
  "columnWidths": "[0.8,0.2]",
  "columnWidthsType": "Relative (%)"
}
},
{
  "id": "00696e8f-0349-426f-825e-81aa5c8d02df",
  "title": "betroffene Funktionen",
  "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEl-
ement)-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN
r2, rel4, f\n\n",
  "width": 7,
  "height": 3,
  "x": 0,
  "y": 6,
  "type": "graph",
  "selection": {
    "Requirement": "(label)",
    "Satisfies": "(label)",
    "Function": "(label)"
  },
  "settings": {
    "nodeColorScheme": "pastel1",
    "nodeLabelFontSize": 2,
    "defaultNodeSize": 2,
    "hideSelections": true,
    "showPropertiesOnHover": true
  },
  "schema": [
    [
      "Requirement",
      "requirementName",
      "requirementText",
      "requirementCategory",
      "requirementID",
      "requirementValueType",
      "requirementType",
      "requirementParameter",
      "requirementPriority"
    ],
    [

```

```

    "Satisfies"
  ],
  [
    "Function",
    "functionID",
    "functionName",
    "functionVariantTag",
    "functionSafetyCritical",
    "functionType"
  ]
]
},
{
  "id": "257a64cc-3bf8-4629-abb7-fb3dccfcff6b",
  "title": "Betroffene logische Elemente ",
  "query": "-----\n\n\n",
  "width": 12,
  "height": 2,
  "x": 0,
  "y": 9,
  "type": "text",
  "selection": {},
  "settings": {},
  "schema": []
}
],
{
  "title": "mechanischer Konstrukteur",
  "reports": [
    {
      "id": "a630e8d9-bced-41db-ba46-9e651fd6509a",
      "title": "betroffene Baugruppe",
      "query": "MATCH (n:`PhysicalElement`) \nWHERE toLower(toString(n.`physicalName`))
CONTAINS toLower($input) \nRETURN DISTINCT n.`physicalName` as value, n.`physicalName` as
display ORDER BY size(toString(value)) ASC LIMIT 5",
      "width": 6,
      "height": 2,
      "x": 0,
      "y": 0,
      "type": "select",

```

```

"selection": {},
"settings": {
  "type": "Node Property",
  "entityType": "PhysicalElement",
  "propertyType": "physicalName",
  "propertyTypeDisplay": "physicalName",
  "parameterName": "neodash_physicalelement_physicalname"
},
"schema": []
},
{
  "id": "babdf314-65ac-4d0f-8a54-1af025383545",
  "title": "",
  "query": "MATCH (p:PhysicalElement {physicalName:$neodash_physicalelement_physical-
name})-[:Decompose]->(p2:PhysicalElement)\nRETURN p2.physicalName AS Element, p2.physical-
Discipline AS Bereich, p2.physicalType AS Typ, p2.physicalCosts AS Kosten\n\n\n",
  "width": 8,
  "height": 4,
  "x": 0,
  "y": 2,
  "type": "table",
  "selection": {},
  "settings": {}
},
{
  "id": "f53569b2-271d-4d62-8575-c9d2cef9417c",
  "title": "Verantwortliche Person",
  "query": "MATCH (p:PhysicalElement{physicalName: $neodash_physicalelement_physical-
name})-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN p.physicalElementOwner \n\n\n",
  "width": 6,
  "height": 2,
  "x": 6,
  "y": 0,
  "type": "value",
  "selection": {},
  "settings": {
    "fontSize": 32
  },
  "schema": []
},
{
  "id": "d2317416-b8cd-4fd4-a69f-d4d19b3fb238",

```

```

    "title": "Gesamtkosten mech. Bauteile",
    "query": "MATCH (p:PhysicalElement {physicalName: $neodash_physicalelement_physical-
name, physicalDiscipline: \"Mechanic\"})-[rel12:Decompose]->(p2:PhysicalElement {physicalDiscip-
line: \"Mechanic\"})\nRETURN SUM(tofloat(p2.physicalCosts))AS totalPhysicalCost\n",
    "width": 4,
    "height": 2,
    "x": 8,
    "y": 2,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 32
    },
    "schema": [],
  },
  {
    "id": "b10ac568-77ea-4355-b7e3-fd40f4f677c5",
    "title": "größter Kostenträger",
    "query": "MATCH (p:PhysicalElement {physicalName: $neodash_physicalelement_physical-
name, physicalDiscipline: \"Mechanic\"})-[rel12:Decompose]->(p2:PhysicalElement)\nWITH p2\nRE-
TURN MAX(p2.physicalCosts) AS maxCost\n\n\n",
    "width": 4,
    "height": 2,
    "x": 8,
    "y": 4,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 32
    },
    "schema": [],
  },
  {
    "id": "62b3daac-e57b-48c8-8cba-a25c79015ed5",
    "title": "alternative Baugruppe",
    "query": "MATCH (n:`PhysicalElement`) \nWHERE toLower(toString(n.`physicalName`))
CONTAINS toLower($input) \nRETURN DISTINCT n.`physicalName` as value, n.`physicalName` as
display ORDER BY size(toString(value)) ASC LIMIT 5",
    "width": 6,
    "height": 2,
    "x": 12,
    "y": 0,

```

```

    "type": "select",
    "selection": {},
    "settings": {
      "type": "Node Property",
      "entityType": "PhysicalElement",
      "propertyType": "physicalName",
      "propertyTypeDisplay": "physicalName",
      "parameterName": "neodash_physicalelement_physicalname_2",
      "id": "2"
    },
    "schema": []
  },
  {
    "id": "c75733dd-40e8-4f83-851f-61a5f27b0b33",
    "title": "Verantwortliche Person",
    "query": "MATCH (p:PhysicalElement{physicalName: $neodash_physicalelement_physical-
name_2})-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN          p.physicalElementOwner
\n\n\n\n\n\n",
    "width": 6,
    "height": 2,
    "x": 18,
    "y": 0,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 32
    },
    "schema": []
  },
  {
    "id": "48e30dc9-8076-4ab2-91e1-5a037091d7d7",
    "title": "",
    "query": "MATCH (p:PhysicalElement {physicalName:$neodash_physicalelement_physical-
name_2})-[:Decompose]->(p2:PhysicalElement)\nRETURN  p2.physicalName AS Element, p2.physi-
calDiscipline AS Bereich,  p2.physicalType AS Typ, p2.physicalCosts AS Kosten\n\n\n",
    "width": 8,
    "height": 4,
    "x": 12,
    "y": 2,
    "type": "table",
    "selection": {},
    "settings": {}
  }

```



```

    },
    {
      "id": "f494c595-a23d-4590-ac37-d194283e74b2",
      "title": "Gesamtkosten mech. Bauteile",
      "query": "MATCH (p:PhysicalElement {physicalName: $neodash_physicalelement_physical-
name_2, physicalDiscipline: \"Mechanic\"})-[rel12:Decompose]->(p2:PhysicalElement {physical-
Discipline: \"Mechanic\"})\nRETURN SUM(tofloat(p2.physicalCosts))AS totalPhysicalCost\n",
      "width": 4,
      "height": 2,
      "x": 20,
      "y": 2,
      "type": "value",
      "selection": {},
      "settings": {
        "type": "Node Property",
        "fontSize": 32
      },
    },
    "schema": []
  },
  {
    "id": "12f505bb-df7e-434c-b7ee-30624d8efbf0",
    "title": "größter Kostenträger",
    "query": "MATCH (p:PhysicalElement {physicalName: $neodash_physicalelement_physical-
name_2, physicalDiscipline: \"Mechanic\"})-[rel12:Decompose]->(p2:PhysicalElement)\nWITH
p2\nRETURN MAX(p2.physicalCosts) AS maxCost\n\n\n",
    "width": 4,
    "height": 2,
    "x": 20,
    "y": 5,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 32
    },
    "schema": []
  }
]
},
{
  "title": "Basisalgorithmen",
  "reports": [
    {

```

```

    "id": "5c0d7cfd-7d1b-42ad-81d7-de99d0dd2eb6",
    "title": "Ist die betroffene Anforderung validiert?",
    "query": "MATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})<-[rel13:Validates]-(Val:Validation)\n\nRETURN CASE WHEN COUNT(DISTINCT Val) > 0
THEN '✔' ELSE '✘' END AS Status\n\n\n",
    "width": 6,
    "height": 2,
    "x": 0,
    "y": 0,
    "type": "value",
    "selection": {},
    "settings": {},
    "schema": []
  },
  {
    "id": "8c26efa5-8f88-41dd-b910-069ec4fe7a65",
    "title": "Ist die betroffene Systemanforderung verifiziert?",
    "query": "MATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)<-[rel14:Verifies]-(Ver:Verification)\nRETURN CASE
WHEN COUNT(DISTINCT Ver) > 0 THEN '✔' ELSE '✘' END AS Status\n\n\n",
    "width": 6,
    "height": 2,
    "x": 0,
    "y": 2,
    "type": "value",
    "selection": {},
    "settings": {},
    "schema": []
  },
  {
    "id": "c108b271-b5e5-478d-aaeb-74954e81d2da",
    "title": "Welche Anforderungen sind von einer initialen Anforderungsänderung per Refines be-
troffen?",
    "query": "\nMATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})-[rel3:Refines]->(r2:Requirement)\nRETURN r, rel3, r2\n\n\n",
    "width": 6,
    "height": 2,
    "x": 0,
    "y": 4,
    "type": "graph",
    "selection": {
      "Requirement": "(label)",

```

```

    "Refines": "(label)"
  },
  "settings": {},
  "schema": [
    [
      "Requirement",
      "requirementName",
      "requirementText",
      "requirementCategory",
      "requirementID",
      "requirementType",
      "requirementPriority"
    ],
    [
      "Refines"
    ]
  ]
},
{
  "id": "3bea8afa-d3af-4333-b409-95eb42b46786",
  "title": "Welche Anforderungen sind von einer initialen Anforderungsänderung per Requires betroffen?",
  "query": "MATCH (r:Requirement{requirementName: $neodash_requirement_requirement-name})-[rel2:Requires]->(r2:Requirement)\nRETURN r, rel2, r2\n\n\n",
  "width": 6,
  "height": 2,
  "x": 6,
  "y": 0,
  "type": "graph",
  "selection": {
    "Requirement": "(label)",
    "Requires": "(label)"
  },
  "settings": {},
  "schema": [
    [
      "Requirement",
      "requirementName",
      "requirementText",
      "requirementCategory",
      "requirementID",
      "requirementType",

```

```

        "requirementPriority"
    ],
    [
        "Requires"
    ]
]
},
{
    "id": "687e59c0-15df-4c6c-a44d-b9af845ec8fb",
    "title": "Wie ist die Priorität betroffener Anforderungen?",
    "query": "MATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})\nRETURN r.requirementPriority\n\n\n",
    "width": 6,
    "height": 2,
    "x": 6,
    "y": 2,
    "type": "value",
    "selection": {},
    "settings": {},
    "schema": []
},
{
    "id": "fd40d11d-f5c5-4360-92a9-47a3c403fa5a",
    "title": "Welche Kategorie haben betroffene Anforderungen?",
    "query": "MATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})\nRETURN r.requirementType\n\n\n",
    "width": 6,
    "height": 2,
    "x": 6,
    "y": 4,
    "type": "value",
    "selection": {},
    "settings": {},
    "schema": []
},
{
    "id": "6f2c4494-a844-4357-aa9b-e6a0623f1ae1",
    "title": "Welche Anforderungen sind validiert?",
    "query": "MATCH (r:Requirement)-[rel13:Validates]-(Val:Validation)\nRETURN DISTINCT
r.requirementName as `betroffene Systemanforderungen`, r.requirementPriority as Anforderungspriori-
tät, r.requirementText as Anforderungsbeschreibung\n",
    "width": 6,

```

```

    "height": 2,
    "x": 12,
    "y": 2,
    "type": "table",
    "selection": {},
    "settings": {}
  },
  {
    "id": "12595a1a-8482-4ca0-9673-e6cdbff1525f",
    "title": "Welche Anforderungen sind validiert?",
    "query": "MATCH (r:Requirement)-[rel13:Validates]-(Val:Validation)\nRETURN DISTINCT
r.requirementName as `betroffene Systemanforderungen`, r.requirementPriority as Anforderungspriori-
tät, r.requirementText as Anforderungsbeschreibung \n\n\n",
    "width": 6,
    "height": 2,
    "x": 12,
    "y": 0,
    "type": "table",
    "selection": {},
    "settings": {},
    "schema": []
  },
  {
    "id": "4c0e98c0-1fc7-4f2f-8df2-cf615bd78a35",
    "title": "Welche Anforderungsparameter müssen geprüft werden?",
    "query": "MATCH (r:Requirement)-[rel1_2_3:Derives|Requires|Refines]->(r2:Require-
ment)\nRETURN DISTINCT r2.requirementName AS Anforderungsname, r2.requirementParameter
AS Anforderungsparameter \n",
    "width": 6,
    "height": 2,
    "x": 18,
    "y": 0,
    "type": "table",
    "selection": {},
    "settings": {}
  },
  {
    "id": "8d6fd5-030d-4dde-903b-8aeddda46562",
    "title": "Welche Anforderungen sind verifiziert?",
    "query": "MATCH (r:Requirement)-[rel14:Verifies]-(Ver:Verification)\nRETURN DISTINCT
r.requirementName as `betroffene Systemanforderungen`, r.requirementPriority as Anforderungspriori-
tät, r.requirementText as Anforderungsbeschreibung \n\n\n",

```

```

    "width": 6,
    "height": 2,
    "x": 12,
    "y": 4,
    "type": "table",
    "selection": {},
    "settings": {}
  },
  {
    "id": "839ce56f-11ef-4ce1-acf2-daaa566ea540",
    "title": "Welche Funktionen sind direkt von einer Anforderungsänderung betroffen?",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)\nRETURN r2, rel4, f\n\n",
    "width": 6,
    "height": 2,
    "x": 18,
    "y": 2,
    "type": "graph",
    "selection": {
      "Requirement": "(label)",
      "Satisfies": "(label)",
      "Function": "(label)"
    },
    "settings": {},
    "schema": [
      [
        "Requirement",
        "requirementName",
        "requirementText",
        "requirementCategory",
        "requirementID",
        "requirementValueType",
        "requirementType",
        "requirementParameter",
        "requirementPriority"
      ],
      [
        "Satisfies"
      ],
      [
        "Function",
        "functionID",

```

```

        "functionName",
        "functionVariantTag",
        "functionSafetyCritical",
        "functionType"
    ]
}
{
    "id": "d5532f8b-3a6f-41d8-b0b1-029a2702e95a",
    "title": "Welche Art von Funktion ist betroffen?",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)\nRETURN f.functionName as
Funktionsname, f.functionType as Funktionsart\n\n\n",
    "width": 6,
    "height": 2,
    "x": 18,
    "y": 4,
    "type": "table",
    "selection": {},
    "settings": {},
    "schema": []
},
{
    "id": "d8575ebe-b90c-48a1-a3a4-1d1429568ba2",
    "title": "In welcher Variante wird die betroffene Funktion implementiert?",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)\nRETURN f.functionName as
Funktionsname, f.functionVariantTag as `Variante der Funktion`\n\n\n\n",
    "width": 6,
    "height": 2,
    "x": 0,
    "y": 6,
    "type": "table",
    "selection": {},
    "settings": {}
},
{
    "id": "201498dd-857d-47c5-806c-a07044aa91b5",
    "title": "Welche logischen Elemente sind von einer Anforderungsänderung betroffen?",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEl-
ement)\nRETURN r, rel1, r2, rel4, f, rel7, l\n\n\n",

```

```
"width": 6,
"height": 2,
"x": 0,
"y": 8,
"type": "graph",
"selection": {
  "Requirement": "(label)",
  "Derives": "(label)",
  "Satisfies": "(label)",
  "Function": "(label)",
  "Specifies": "(label)",
  "LogicalElement": "(label)"
},
"settings": {},
"schema": [
  [
    "Requirement",
    "requirementName",
    "requirementText",
    "requirementCategory",
    "requirementID",
    "requirementType",
    "requirementPriority",
    "requirementValueType",
    "requirementParameter"
  ],
  [
    "Derives"
  ],
  [
    "Satisfies"
  ],
  [
    "Function",
    "functionID",
    "functionName",
    "functionVariantTag",
    "functionSafetyCritical",
    "functionType"
  ],
  [
    "Specifies"
```



```

    ],
    [
        "LogicalElement",
        "logicalType",
        "logicalID",
        "logicalName",
        "logicalVariantTag",
        "Variant Tag"
    ]
]
},
{
    "id": "a89f8760-5819-4e05-a028-fb65b1781d25",
    "title": "Welche Art von logischem Element ist betroffen?",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalEl-
ement)\nRETURN l.logicalName AS `logisches Element`, f.functionVariantTag as `Art des logischen
Elements`\n\n\n",
    "width": 6,
    "height": 2,
    "x": 0,
    "y": 10,
    "type": "table",
    "selection": {},
    "settings": {}
},
{
    "id": "2a675b21-0ed9-438c-a8c9-b19e817b1227",
    "title": "In welcher Variante wird das logische Element implementiert?",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalEl-
ement)\nRETURN l.logicalName AS `logisches Element`,l.logicalVariantTag AS `Art des logischen
Elements`\n\n\n",
    "width": 6,
    "height": 2,
    "x": 6,
    "y": 6,
    "type": "table",
    "selection": {},
    "settings": {}
},
{

```

```

    "id": "d4378323-0f10-4334-9c71-3555bd14dc87",
    "title": "Welche physischen Elemente sind von einer Anforderungsänderung betroffen?",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEl-
ement)-[rel10:Realizes]-(p:PhysicalElement)-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN
r, rel1, r2, rel4, f, rel7, l, rel10, p\n\n\n",
    "width": 6,
    "height": 2,
    "x": 6,
    "y": 8,
    "type": "graph",
    "selection": {
      "Requirement": "(label)",
      "Derives": "(label)",
      "Satisfies": "(label)",
      "Function": "(label)",
      "Specifies": "(label)",
      "LogicalElement": "(label)",
      "Realizes": "(label)",
      "PhysicalElement": "(label)"
    },
    "settings": {},
    "schema": [
      [
        "Requirement",
        "requirementName",
        "requirementText",
        "requirementCategory",
        "requirementID",
        "requirementType",
        "requirementPriority",
        "requirementValueType",
        "requirementParameter"
      ],
      [
        "Derives"
      ],
      [
        "Satisfies"
      ],
      [
        "Function",

```

```

        "functionID",
        "functionName",
        "functionVariantTag",
        "functionSafetyCritical",
        "functionType"
    ],
    [
        "Specifies"
    ],
    [
        "LogicalElement",
        "logicalType",
        "logicalID",
        "logicalName",
        "logicalVariantTag",
        "Variant Tag"
    ],
    [
        "Realizes"
    ],
    [
        "PhysicalElement",
        "physicalName",
        "physicalDiscipline",
        "physicalID",
        "physicalType",
        "physicalElementOwner",
        "physicalVariantTag"
    ]
]
},
{
    "id": "80940784-df28-4641-9d16-221c39d4a1e6",
    "title": "Welche Art von physischen Element ist betroffen?",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)<-[rel4:Satisfies]-(f:Function)<-[rel7:Specifies]-(l:LogicalEl-
ement)<-[rel10:Realizes]-(p:PhysicalElement)\nRETURN  p.physicalName AS`physisches Element`,
p.physicalType AS `Art der technischen Änderung`\n\n\n",
    "width": 6,
    "height": 2,
    "x": 6,
    "y": 10,

```

```

    "type": "table",
    "selection": {},
    "settings": {}
  },
  {
    "id": "75d0e787-191b-4dad-bf1d-d98274d8965c",
    "title": "Welche Disziplinen sind von einer Änderung betroffen?",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEl-
ement)-[rel10:Realizes]-(p:PhysicalElement)\nRETURN p.physicalName AS`physisches Element`,
p.physicalDiscipline AS Disziplin\n\n\n",
    "width": 6,
    "height": 2,
    "x": 12,
    "y": 6,
    "type": "table",
    "selection": {},
    "settings": {}
  },
  {
    "id": "c09e6572-4997-4dac-8965-34b67f365785",
    "title": "Welche Varianten sind von einer technischen Änderung betroffen?",
    "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-
name})-[rel1:Derives]->(r2:Requirement)-[rel4:Satisfies]-(f:Function)-[rel7:Specifies]-(l:LogicalEl-
ement)-[rel10:Realizes]-(p:PhysicalElement)\nRETURN p.physicalName AS`physisches Element`,
p.physicalVariantTag AS Variante\n\n\n",
    "width": 6,
    "height": 2,
    "x": 12,
    "y": 8,
    "type": "table",
    "selection": {},
    "settings": {}
  },
  {
    "id": "7ef1264b-4ca6-4467-868d-69cfef3ac64d",
    "title": "Wird ein physisches Element in einer oder in mehreren Varianten verwendet?",
    "query": "\nMATCH (p:PhysicalElement {physicalName: $neodash_physicalelement_physical-
name})-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN p2.physicalVariantTag as Vari-
ante\n\n\n",
    "width": 6,
    "height": 2,

```

```

    "x": 12,
    "y": 10,
    "type": "value",
    "selection": {},
    "settings": {},
    "schema": []
  },
  {
    "id": "5c511ea9-145b-4d68-8f7a-edecd4eebd18",
    "title": "Welche Person ist verantwortlich für ein physisches Element?",
    "query": "MATCH (p:PhysicalElement {physicalName: $neodash_physicalelement_physical-
name})-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN p.physicalElementOwner AS `verant-
wortliche Person`\n",
    "width": 6,
    "height": 2,
    "x": 18,
    "y": 6,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 32
    },
    "schema": []
  },
  {
    "id": "52fb2be2-d050-4655-acf3-2ae317a33b75",
    "title": "Aus welchem Material besteht ein physisches Element?",
    "query": "MATCH (p:PhysicalElement {physicalName: $neodash_physicalelement_physical-
name})-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN p2.physicalMaterial AS Mate-
rial\n\n\n",
    "width": 6,
    "height": 2,
    "x": 18,
    "y": 8,
    "type": "value",
    "selection": {},
    "settings": {},
    "schema": []
  },
  {
    "id": "b790b84d-700b-4ae2-8fbb-de63a58cbe92",
    "title": "Wie hoch sind die Kosten der betroffenen physischen Elemente?",

```

```

    "query": "MATCH (p:PhysicalElement {physicalName: $neodash_physicalelement_physical-
name})-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN p2.physicalName AS `physisches Ele-
ment`, p.physicalCosts AS Kosten\nUNION\nMATCH (p:PhysicalElement {physicalName: $neo-
dash_physicalelement_physicalname})-[rel12:Decompose]->(p2:PhysicalElement)\nRETURN p2.phy-
sicalName AS `physisches Element`, p2.physicalCosts AS Kosten\n",
    "width": 6,
    "height": 2,
    "x": 18,
    "y": 10,
    "type": "table",
    "selection": {},
    "settings": {}
  },
  {
    "id": "c8f035bf-cdcb-4d1f-a07e-95ae59faeb3f",
    "title": "Welches physische Element ist der größten Kostentreiber?",
    "query": "MATCH (p:PhysicalElement {physicalName: $neodash_physicalelement_physical-
name})-[rel12:Decompose]->(p2:PhysicalElement)\nWITH p2\nRETURN MAX(p2.physicalCosts)
AS maxCost\nLIMIT 1\n",
    "width": 6,
    "height": 2,
    "x": 0,
    "y": 12,
    "type": "value",
    "selection": {},
    "settings": {
      "fontSize": 32
    },
    "schema": []
  },
  {
    "id": "330b3a2a-8977-4059-9719-a500152ab2f2",
    "title": "Wie viele Refines-Kanten mit gehen von einem Requirement aus?",
    "query": "\nMATCH (r:Requirement{requirementName: $neodash_requirement_requirement-
name})-[rel]->()\nWITH COUNT(rel) AS outgoingEdgesCount\nRETURN outgoingEdgesCount AS
`ausgehende Kanten`\n\n",
    "width": 6,
    "height": 2,
    "x": 0,
    "y": 14,
    "type": "value",
    "selection": {}
  }

```

```
"settings": {
  "fontSize": 32
},
"schema": []
},
{
  "id": "f8c886e9-90b3-441e-85cf-f66231dcbcb3",
  "title": "Ist die betrachte Anforderung in Bezug auf die Beziehung Refines ein Absorbierer, ein  
Träger oder ein Multiplizierer?",
  "query": "MATCH (r:Requirement {requirementName: $neodash_requirement_requirement-  
name})-[rel]->()\nWITH COUNT(rel) AS outgoingEdgesCount\nRETURN\nCASE\nWHEN outgoingEdgesCount = 0 THEN 'Absorbierer'\nWHEN outgoingEdgesCount = 1 THEN 'Träger'\nELSE 'Mul-  
tiplizierer'\nEND AS result\n\n\n",
  "width": 6,
  "height": 2,
  "x": 0,
  "y": 16,
  "type": "value",
  "selection": {},
  "settings": {
    "fontSize": 32
  },
  "schema": []
}
]
}
],
"parameters": {},
"extensions": {
  "active": true,
  "activeReducers": []
},
"uuid": "72c52c3b-2ce4-42c2-a69c-08f067db235f"
}
```


Das Heinz Nixdorf Institut – Interdisziplinäres Forschungszentrum für Informatik und Technik

Das Heinz Nixdorf Institut ist ein Forschungszentrum der Universität Paderborn. Es entstand 1987 aus der Initiative und mit Förderung von Heinz Nixdorf. Damit wollte er Ingenieurwissenschaften und Informatik zusammenführen, um wesentliche Impulse für neue Produkte und Dienstleistungen zu erzeugen. Dies schließt auch die Wechselwirkungen mit dem gesellschaftlichen Umfeld ein.

Die Forschungsarbeit orientiert sich an dem Programm „Dynamik, Vernetzung, Autonomie: Neue Methoden und Technologien für die intelligenten technischen Systeme von morgen“. In der Lehre engagiert sich das Heinz Nixdorf Institut in Studiengängen der Informatik, der Ingenieurwissenschaften und der Wirtschaftswissenschaften.

Heute wirken am Heinz Nixdorf Institut acht Professoren/in mit insgesamt 120 Mitarbeiterinnen und Mitarbeitern. Pro Jahr promovieren hier etwa 15 Nachwuchswissenschaftlerinnen und Nachwuchswissenschaftler.

Heinz Nixdorf Institute – Interdisciplinary Research Centre for Computer Science and Technology

The Heinz Nixdorf Institute is a research centre within the University of Paderborn. It was founded in 1987 initiated and supported by Heinz Nixdorf. By doing so he wanted to create a symbiosis of computer science and engineering in order to provide critical impetus for new products and services. This includes interactions with the social environment.

Our research is aligned with the program “Dynamics, Networking, Autonomy: New methods and technologies for intelligent technical systems of tomorrow”. In training and education the Heinz Nixdorf Institute is involved in many programs of study at the University of Paderborn. The superior goal in education and training is to communicate competencies that are critical in tomorrows economy.

Today eight Professors and 120 researchers work at the Heinz Nixdorf Institute. Per year approximately 15 young researchers receive a doctorate.

Erklärung zur Zitation von Inhalten aus studentischen Arbeiten

In Ergänzung zu meinem Antrag auf Zulassung zur Promotion in der Fakultät für Maschinenbau der Universität Paderborn erkläre ich gemäß §11 der Promotionsordnung und unter Beachtung der Regelung zur Zitation studentischer Arbeiten:

Die von mir vorgelegte Dissertation habe ich selbstständig verfasst, und ich habe keine anderen als die dort angegebenen Quellen und Hilfsmittel benutzt. Es sind Inhalte studentischen Ursprungs (studentische Arbeiten) in dieser Dissertation enthalten.

Ich habe die verwendeten Arbeiten entsprechend der Regelung „Zitation aus studentischen Arbeiten in Dissertationen“ zitiert.


Paderborn, 28.04.2025:

Zuletzt erschienene Bände der Verlagsschriftenreihe des Heinz Nixdorf Instituts

- Bd. 407 JAVED, A. R.: Mixed-Signal Baseband Circuit Design for High Data Rate Wireless Communication in Bulk CMOS and SiGe BiCMOS Technologies. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 407, Paderborn, 2022 – ISBN 978-3-947647-26-2
- Bd. 408 DUMITRESCU, R., KOLDEWEY, C.: Daten-gestützte Projektplanung. Fachbuch. Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 408, Paderborn, 2022 – ISBN 978-3-947647-27-9
- Bd. 409 PÖHLER, A.: Automatisierte dezentrale Produktionssteuerung für cyber-physische Produktionssysteme mit digitaler Repräsentation der Beschäftigten. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 409, Paderborn, 2022 – ISBN 978-3-947647-28-6
- Bd. 410 RÜDDENKLAU, N.: Hardware-in-the-Loop-Simulation von HD-Scheinwerfer-Steuergeräten zur Entwicklung von Lichtfunktionen in virtuellen Nachtfahrten. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 410, Paderborn, 2023 – ISBN 978-3-947647-29-3
- Bd. 411 BIEMELT, P.: Entwurf und Analyse modell-prädiktiver Regelungsansätze zur Steigerung des Immersionsempfindens in interaktiven Fahrsimulationen. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 411, Paderborn, 2023 – ISBN 978-3-947647-30-9
- Bd. 412 HAAKE, C.-J., MEYER AUF DER HEIDE, F., PLATZNER, M., WACHSMUTH, H., WEHRHEIM, H. (Eds.): On-The-Fly Computing - Individualized IT-Services in dynamic markets, Collaborative Research Centre 901 (2011 - 2023), Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 412, Paderborn, 2023 – ISBN 978-3-947647-31-6
- Bd. 413 DUMITRESCU, R.; HÖLZLE, K. (Hrsg.): Vorausschau und Technologieplanung. 17. Symposium für Vorausschau und Technologieplanung, Heinz Nixdorf Institut, 14./15. September 2023, Berlin-Brandenburgische Akademie der Wissenschaften, Berlin, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 413, Paderborn, 2023 – ISBN 978-3-947647-32-3
- Bd. 414 ABUGHANNAM, S.: Low-power Direct-detection Wake-up Receiver at 2.44 GHz for Wireless Sensor Networks. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 414, Paderborn, 2024 – ISBN 978-3-947647-33-0
- Bd. 415 REINHOLD, J.: Systematik zur muster-basierten Transformation von Wertschöpfungssystemen für Smart Services. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 415, Paderborn, 2024 – ISBN 978-3-947647-34-7
- Bd. 416 YANG, X.: Eine Methode zur Unterstützung von Entscheidungen bei der Entwicklung modularer Leichtbauprodukte. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 416, Paderborn, 2024 – ISBN 978-3-947647-35-4
- Bd. 417 GRÄLER, M.: Entwicklung adaptiver Einrichtungssysteme für Produktionsprozesse. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 417, Paderborn, 2024 – ISBN 978-3-947647-36-1
- Bd. 418 RÖSMANN, D.: Menschenzentrierte Montageplanung und -steuerung durch fähigkeitsorientierte Aufgabenzuordnung. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 418, Paderborn, 2024 – ISBN 978-3-947647-37-8
- Bd. 419 BAHMANIAN, M.: Optoelectronic Phase-Locked Loop, Theory and Implementation. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 419, Paderborn, 2024 – ISBN 978-3-947647-38-5

Zuletzt erschienene Bände der Verlagsschriftenreihe des Heinz Nixdorf Instituts

- Bd. 420 HEIHOFF-SCHWEDE, J.: Spezifikations-technik zur Analyse, Gestaltung und Bewertung von Engineering-IT-Architekturen. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 420, Paderborn, 2024 – ISBN 978-3-947647-39-2
- Bd. 421 MEYER, M.: Systematik zur Planung und Verwertung von Betriebsdaten-Analysen in der strategischen Produktplanung. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 421, Paderborn, 2024 – ISBN 978-3-947647-40-8
- Bd. 422 MALENA, K.: Konzipierung, Analyse und Realumsetzung eines mehrstufigen modellprädiktiven Lichtsignalanlagenregelungssystems. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 422, Paderborn, 2024 – ISBN 978-3-947647-41-5
- Bd. 423 GÖTTE, R.-S.: Online-Schätzung von Modellgenauigkeiten zur automatischen Modelladaptation unter Beibehaltung einer physikalisch-technischen Interpretierbarkeit. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 423, Paderborn, 2024 – ISBN 978-3-947647-42-2
- Bd. 424 PIERENKEMPER, C.: Systematik zur Entwicklung Leistungsstufen-basierter Industrie 4.0-Strategien. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 424, Paderborn, 2024 – ISBN 978-3-947647-43-9
- Bd. 425 DUMITRESCU, R.; HÖLZLE, K. (Hrsg.): Vorausschau und Technologieplanung. 18. Symposium für Vorausschau und Technologieplanung, Heinz Nixdorf Institut, 5./6. Dezember 2024, Berlin-Brandenburgische Akademie der Wissenschaften, Berlin, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 425, Paderborn, 2024 – ISBN 978-3-947647-44-6
- Bd. 426 HESSE, M.: Interaktive Inbetriebnahme von Steuerungen und Regelungen für partiell bekannte dynamische Systeme mittels Gauß-Prozess-Regression. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 426, Paderborn, 2024 – ISBN 978-3-947647-45-3
- Bd. 427 PREUß, D.: Methodik zur automatisierten Extrahierung, Formalisierung und Handhabung von Änderungen technischer Anforderungen. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 427, Paderborn, 2024 – ISBN 978-3-947647-46-0
- Bd. 428 JUNKER, A.: Datengetriebene Modellbildung für nicht lineare mechatronische Systeme in regelungstechnisch verwertbarer Form. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 428, Paderborn, 2024 – ISBN 978-3-947647-47-7
- Bd. 429 THIELE, H.: Technologievorausschau auf Basis mathematischer Modellierung. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 429, Paderborn, 2025 – ISBN 978-3-947647-48-4
- Bd. 430 TIMMERMANN, J., JUNKER, A., GÖTTE, R. S., HESSE, M., SCHWARZER, L.: DART - Datengetriebene Methoden in der Regelungstechnik. Fakultät für Maschinenbau, Universität Paderborn, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 432, Paderborn, 2025 – ISBN 978-3-947647-49-1
- Bd. 431 DUMITRESCU, R.; HÖLZLE, K. (Hrsg.): Vorausschau und Technologieplanung. 19. Symposium für Vorausschau und Technologieplanung, Heinz Nixdorf Institut, 23./24. Oktober 2025, Berlin-Brandenburgische Akademie der Wissenschaften, Berlin, Verlagsschriftenreihe des Heinz Nixdorf Instituts, Band 432, Paderborn, 2025 – ISBN 978-3-947647-50-7



Der unsystematische und ineffiziente Umgang mit technischen Änderungen führt zu hohen Personalkosten durch Mehraufwand, einer reduzierten Entscheidungsqualität und kann in Projektfehlschlägen resultieren. Herausforderungen beim Umgang mit technischen Änderungen liegen in der Berücksichtigung relevanter Ursache-Wirkungs-Zusammenhänge, der quantitativen und qualitativen Bewertung der Auswirkungen und der effizienten prozessualen Einbindung. Mithilfe eines modellbasierten Lösungsansatzes lassen sich die Auswirkungen technischer Änderungen schneller bewerten und fundierte Entwicklungsentscheidungen treffen. In der vorliegenden Dissertation wird eine Methodik für die Entwicklung modell-basierter Wirkkettenanalysen zur Bewertung technischer Änderungen beschrieben. Durch die modellbasierte Wirkkettenanalyse können direkt oder indirekt von der Änderung betroffene Modellelemente identifiziert werden. Das Datenmodell wird mithilfe von Abfragesprachen analysiert und die Auswirkungen werden mithilfe von integrierten Metriken quantifiziert. Die Ergebnisse der Wirkkettenanalyse werden in rollenspezifischen Sichten bereitgestellt, wodurch eine bedarfsgerechte Entscheidungsunterstützung gewährleistet wird. Die Validierung anhand von vier industriellen Fallstudien belegt die Praxistauglichkeit der Methodik. Die Methodik verbindet das Änderungsmanagement mit den Vorteilen einer modellbasierten Systementwicklung und bildet somit einen synergetischen neuen Forschungsbereich.