

Jürgen Gausemeier
Roman Dumitrescu
Franz Rammig
Wilhelm Schäfer
Ansgar Trächtler (Hrsg.)

Entwurf mechatronischer Systeme

- Grundlagen, Methoden und Werkzeuge
- Adaption, Selbstoptimierung und Verlässlichkeit
- Integration Mechanik und Elektronik,
Miniaturisierung

10. Paderborner Workshop
Entwurf mechatronischer Systeme
23. und 24. April 2015
Heinz Nixdorf MuseumsForum

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

©Heinz Nixdorf Institut, Universität Paderborn – Paderborn – 2015

ISSN 2195-5239

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: Polina Decheva und Franziska Reichelt

Hersteller: Hans Gieselmann Druck und Medienhaus GmbH & Co. KG
Druck · Buch · Verlag
Bielefeld

Printed in Germany

Vorwort

Die Symbiose von Informatik und Ingenieurwissenschaften ist das herausragende Merkmal der Universität Paderborn und insbesondere des Heinz Nixdorf Instituts. Die Aktivitäten auf den Gebieten Mechatronik und Intelligente Technische Systeme sind ein Ausdruck dieser Positionierung.

Wir haben ein vitales Interesse an einem intensiven Dialog mit der Fachwelt; aus diesem Grund veranstalten wir zweijährlich unter dem Dach „Wissenschafts- und Industrieforum – Intelligente Technische Systeme“ den Workshop „Entwurf mechatronischer Systeme“. Wir richten uns an Fachleute aus Industrie und Forschungsinstituten, die sich maßgeblich mit der Entwicklung der technischen Systeme von morgen befassen. Die Veranstaltung bietet ihnen eine Plattform zur Diskussion und zum Erfahrungsaustausch.

Für die Begutachtung und Auswahl der Beiträge danken wir den Mitgliedern des Programmkomitees herzlich. Ohne deren Expertise wäre diese Tagung in dieser Form nicht möglich.

Prof. Dr.-Ing. Dr. h.c. A. Albers,
Universität Karlsruhe

Dr.-Ing. T. Koch,
Porsche AG

Prof. Dr.-Ing. R. Anderl,
TU Darmstadt

Prof. Dr.-Ing. F.-L. Krause,
TU Berlin

Prof. Dr.-Ing. Prof. h.c. T. Bertram,
TU Dortmund

Prof. Dr. rer. nat. H. Kück,
HSG-IMAT

Prof. Dr. Dr. h.c. M. Broy,
TU München

Prof. Dr.-Ing. U. Lindemann,
TU München

Prof. Dr.-Ing. R. Dudziak,
FH Bochum

Prof. Dr.-Ing. U. Rückert,
Universität Bielefeld

Prof. Dr.-Ing. J. Franke,
Universität Erlangen-Nürnberg

Dr. E. Sailer,
Miele & Cie. KG

Dr.-Ing. M. Gebauer,
Karlsruher Institut für Technologie

Dr.-Ing. E. Schäfers,
Siemens AG

Prof. Dr. Dr.-Ing. h.c. A. Gilg,
Siemens CT

Prof. Dr. R. Scheidl,
Universität Linz

Prof. Dr.-Ing. I. Gräßler,
Universität Paderborn

Dr.-Ing. H. Schütte,
dSPACE GmbH

Dr.-Ing. M. Hahn,
iXtronics GmbH

Prof. Dr.-Ing. habil. W. Sextro,
Universität Paderborn

Prof. Dr.-Ing. R. Kasper,
Universität Magdeburg

Prof. Dr.-Ing. C. Weber,
TU Ilmenau

Prof. Dr.-Ing. A. Kecskeméthy,
Universität Duisburg-Essen

Prof. Dr.-Ing. M. Zäh,
TU München

Paderborn, im April 2015

Prof. Dr.-Ing. J. Gausemeier, Dr.-Ing. R. Dumitrescu, Prof. Dr. rer. nat. F. Rammig,
Prof. Dr. rer. nat. W. Schäfer und Prof. Dr.-Ing. A. Trächtler

Inhaltsverzeichnis

Einführung

J. Gausemeier, A. Czaja, C. Dülme Innovationspotentiale auf dem Weg zu Industrie 4.0.....	11
--	----

Model-Based Systems Engineering in der Praxis

S. Kleiner Model Based Systems Engineering: Einführung und Anwendungs- erfahrung der modellbasierten Entwicklung im Maschinenbau	53
N. Schmitt, R. Dumitrescu Ein durchgängiges Vorgehen zur Anforderungserfassung und -nachverfolgung.....	67
M. Schäfer, G. Stollt, A. Nyßen, R. Dorociak Erweiterung der Entwurfsmethodik CONSENS um absichernde Maßnahmen zur Risikominimierung in Produkten	81

Intelligente Regelung mechatronischer Systeme

T. Kaul, T. Meyer, W. Sextro Integrierte Modellierung der Dynamik und der Verlässlichkeit komplexer mechatronischer Systeme	101
A. Rütting, C. Henke, A. Warkentin, A. Trächtler Entwurf eines Tripod-basierten Inspektionssystems – Vom virtuellen Prototypen zum Vorseriensystem	113
S. Berger, D. Simon, P. Stich, G. Reinhart Mechatronisches Roboter-System zum Rückbau von Reaktordruckbehältereinbauten	127

Modellierung eingebetteter Systeme

L. Krawczyk, R. Höttger, C. Wolff, C. Brink, D. Fruhner AMALTHEA – Eine durchgängige Entwicklungsplattform für die modellgetriebene Entwicklung automobiler eingebetteter Systeme	143
---	-----

J. Meyer, J. Holtmann, T. Koch, M. Meyer
Generierung von AUTOSAR-Modellen aus UML-Spezifikationen 159

S. Windmann, O. Niggemann
Selbstdiagnose und Selbstoptimierung technischer Systeme auf Basis
datenbasierter Prozessmodelle 173

Vernetzte mechatronische Systeme

M. Petersen, G. Rehage, J. Gausemeier, F. Bauer
Wissensaufbereitung und -bereitstellung durch Ontologien im
Lebenszyklus von Produktionssystemen..... 189

D. Störkle, A. Barthelmey, J. Deuse, B. Kuhlenkötter, S. Magerstedt
Technische Dokumentation in der Smart Factory..... 211

J. Ponn
Portfolio-Management für Elektroantriebe in Powertools bei Hilti:
Herausforderungen und Lösungsansätze 225

Intelligente Elektronikentwicklung

J. Zeitler, C. Fischer, J. Franke
Integrativer Ansatz zur rechnergestützten Entwicklung räumlicher
optomechatronischer Baugruppen 239

T. Schierbaum, J. Gausemeier
Systematik zur Kostenbewertung von mechatronischen Systemen in der
Technologie Molded Interconnect Devices..... 251

S. Herbrechtsmeier, T. Jungeblut, M. Pormann
Datenflussmodellierung als Methode zur Optimierung von
Entwicklungsprozessen am Beispiel der Leiterplattenentwicklung 265

Modellierung und Simulation mechatronischer

J. Roßmann, M. Schluse, M. Rast, M. Hoppen, R. Dumitrescu, C. Bremer,
M. Hillebrand, O. Stern, F. Blümel, C. Averdung
Integrierte Entwicklung komplexer Systeme mit modellbasierter
Systemspezifikation und -simulation 279

L. Weingartner, P. Hehenberger, S. Boschert, R. Rosen Simulationsgetriebene Systemmodellierung zur Analyse und Optimierung von Stückgut-Förderanlagen.....	291
S. Krottil, G. Reinhart Online-Simulation fluiddynamischer Prozesse in der Konzeptionsphase	305
A. Albers, C. Bremer, T. Bruns, R. Dumitrescu, M. Krüger Modellbasierte Analyse und Simulation industrieller Großwäschereien	319

Einführung

Innovationspotentiale auf dem Weg zu Industrie 4.0

Jürgen Gausemeier, Anja Czaja, Christian Dülme

Heinz Nixdorf Institut, Universität Paderborn

Fürstenallee 11, 33102 Paderborn

Tel. +49 (0) 5251 / 60 62 67, Fax. +49 (0) 5251 / 60 62 68

E-Mail: {Juergen.Gausemeier|Anja.Czaja|Christian.Dülme}@hni.upb.de

Zusammenfassung

Mehr denn je wird die rasante Entwicklung der Informations- und Kommunikationstechnik den Maschinenbau und verwandte Branchen prägen. Es zeichnen sich Erzeugnisse und Geschäftsprozesse mit einer inhärenten Teilintelligenz ab, die über das Internet kommunizieren und ggf. kooperieren. Dafür steht der Begriff Cyber-Physical Systems (CPS). Die Ausprägung von CPS auf die industrielle Wertschöpfung wird als Industrie 4.0 bezeichnet. Offenbar eröffnen sich aus Industrie 4.0 faszinierende Möglichkeiten. Dabei darf aber nicht übersehen werden, dass der Einsatz geballter Informations- und Kommunikationstechnik am Ende einer wohlstrukturierten Handlungskette stehen muss, die in erster Linie durch die Notwendigkeit der strategischen Unternehmensführung bestimmt sein muss. In diesem Sinne zeigt der Beitrag, wie die Erfolgspotentiale der Zukunft frühzeitig erkannt werden und wie erfolgversprechende Geschäftsmodelle entstehen können. Einen weiteren Schwerpunkt bildet die Systematik der Produktentstehung. Diese verdeutlicht, dass Produkt-, Dienstleistungs- und Produktionssystementwicklung integrativ zu erfolgen haben und diese wiederum im engen Wechselspiel mit der strategischen Planung vorzunehmen sind. Für diese neue Art der Entwicklung von Marktleistungen, muss ein neuer Ansatz der Entwicklungsmethodik gewählt werden, weil die heute etablierten Entwicklungsmethodiken fachgebietszentriert sind und die Integration der beteiligten Fachgebiete wie Mechanik, Elektrotechnik/Elektronik und Softwaretechnik nicht unterstützen. Systems Engineering ist der von uns propagierte Ansatz. Wir zeigen, wie auf der Basis von digitalen Modellen der zu entwickelnden Objekte im Zeitalter von Cyber-Physical Systems intelligente Produkte und Produktionssysteme erfolgreich entwickelt werden können.

Schlüsselworte

Intelligente Technische Systeme, Cyber-Physical Systems, Industrie 4.0, Produktentstehung, Systems Engineering, Virtualisierung, Szenarien, Unternehmensführung, Geschäftsstrategien, Geschäftsmodelle

Innovation Potential on the Way to Industrie 4.0

Abstract

The fast development of information and communication technology will influence machines construction and manufacturing engineering, as well as the related industries more than ever. There will be products and business processes with inherent intelligence, which communicate and cooperate through the internet. The term Cyber-Physical Systems (CPS) stands for this direction. The particular aspect of Cyber-Physical Systems (CPS), which focuses on the added industrial value, is called Industrie 4.0. Obviously, Industrie 4.0 allows fascinating opportunities. In this context, it should be borne in mind that for the application of information and communication technology a well-structured chain of actions is necessary, which first and foremost has to be based on the strategic business management. In this regard, this paper presents how to early detect success potential and to deduce promising business models. Another focus lies on the systematic of product development. This illustrates that product, service, and production systems have to be developed interactively and closely related to the strategic planning. Therefore, a new development approach is necessary; the established development approaches only focus on one discipline and do not support the integration of mechanics, electronics, and software fields. Systems engineering (SE) is an approach that has a potential to fulfill these requirements. We show how to successfully develop intelligent products and production systems based on virtual system models.

Keywords

intelligent technical systems, cyber-physical systems, Industrie 4.0, product engineering, systems engineering, virtualization, scenarios, business management, business strategy, business models

1 Von der Mechatronik zu Industrie 4.0

Die von uns primär betrachteten Branchen Maschinen- und Anlagenbau, Elektrotechnik/Elektronik, Automobilindustrie und Medizintechnik vollziehen mit ihren Erzeugnissen und Produktionssystemen einen Wandel zu Systemen mit einer inhärenten Teilintelligenz. Dafür stehen die populären Begriffe Intelligente Technische Systeme, Cyber-Physical Systems und Industrie 4.0. Wesentlicher Treiber dieser Entwicklung ist die Informations- und Kommunikationstechnik (IKT), die in enger Beziehung zur Kognitionswissenschaften Systeme ermöglicht, die adaptiv, robust, vorausschauend und benutzungsfreundlich sind (siehe Bild 1).

Intelligente Technische Systeme...



adaptiv

... interagieren mit dem Umfeld und passen sich diesem autonom an



robust

... bewältigen auch unerwartete und vom Entwickler nicht berücksichtigte Situationen in einem dynamischen Umfeld



vorausschauend

... antizipieren auf Basis von Erfahrungswissen die künftigen Wirkungen von Einflüssen und mögliche Zustände



benutzungsfreundlich

... berücksichtigen das spezifische Benutzerverhalten

Bild 1: Eigenschaften von intelligenten technischen Systemen

Im Rahmen der Technologiekonzeption des BMBF-Spitzenclusters „Intelligente Technische Systeme Ostwestfalen-Lippe (it's OWL)“ strukturieren wir entsprechend Bild 2 ein intelligentes technisches System in vier Einheiten: Grundsystem, Sensorik, Aktorik und Informationsverarbeitung. Der Informationsverarbeitung kommt dabei eine zentrale Rolle zu. Sie interveniert via Kommunikationssystem zwischen der Sensorik, durch die die notwendigen Informationen über die Betriebssituation wahrgenommen werden, und der Aktorik, die im Zusammenspiel mit dem Grundsystem die letztliche physische Systemaktion ausführt. Beim Grundsystem handelt es sich in der Regel um mechanische Strukturen.

Wir bezeichnen eine derart elementare Konfiguration aus den genannten vier Einheiten als **Teilsystem**. Beispiele für Teilsysteme sind Antriebe, Automatisierungskomponenten, intelligente Energiespeicher etc. **Systeme**, wie eine Werkzeugmaschine, bestehen in der Regel aus mehreren Teilsystemen, die als interagierender Verbund zu betrachten sind.

In erster Linie prägt die Art der Informationsverarbeitung den beabsichtigten Wandel von mechatronischen zu intelligenten technischen Systemen. So verfügen Erstere nur über eine reaktive und relativ starre Informationsverarbeitung zwischen Sensorik und Aktorik. Intelligente Technische Systeme hingegen können diese gezielt modifizieren. Reaktive Wirkungsabläufe werden dabei nicht vollständig ersetzt, da die meisten existentiellen Systemmechanismen schon aus Gründen der Sicherheit reaktiv und reflexartig ablaufen müssen.

Das aus der Kognitionswissenschaft stammende **Dreischichtenmodell** für die Verhaltenssteuerung [Str98] veranschaulicht diese abstrakte Sichtweise auf die Informationsverarbeitung intelligenter Systeme: Die nicht-kognitive Regulierung enthält die kontinuierliche Steuerung und Regelung (motorischer Regelkreis) sowie Reflexe. Übertragen auf ein mechatronisches System wäre das beispielsweise die Sicherstellung des kontrollierten Bewegungsverhaltens eines Mehrkörpersystems, z.B. aktives Fahrwerk eines PKWs. Die assoziative Regulierung enthält u.a. Reiz-Reaktions-Mechanismen und Konditionierung. In einem technischen System würde die Reglerumschaltung – z.B. von einer Geschwindigkeits- auf eine Abstandsregelung – auf dieser Schicht veranlasst.

Die kognitive Regulierung weist typische Funktionen der künstlichen Intelligenz wie Zielmanagement, Planung und Handlungssteuerung auf. Eine technische Realisierung auf dieser Schicht wäre die Selbstoptimierung, wonach ein System aufgrund geänderter Betriebsbedingungen automatisch seine Ziele modifiziert und dann sein Verhalten selbstständig an die veränderten Ziele anpasst [GRS14].

Ein weiterer zentraler Punkt des Technologiekonzepts ist, dass intelligente technische Systeme – die häufig geographisch verteilt sind – kommunizieren und kooperieren. Die Funktionalität des entstehenden **vernetzten Systems** erschließt sich erst durch das Zusammenspiel der Einzelsysteme. Weder die Vernetzung noch die Rolle der Einzelsysteme ist statisch; vielmehr kann sich beides im Sinne der geforderten Gesamtfunktionalität verändern.

Die Vernetzung erfolgt zunehmend in globaler Dimension. Dabei werden Ansätze im Sinne von Cyber-Physical Systems integriert, die in der Vergangenheit völlig separat betrachtet wurden, wie beispielsweise Cloud Computing einerseits und eingebettete Systeme andererseits. Das vernetzte System wird nicht mehr ausschließlich durch eine globale Steuerung beherrschbar sein, vielmehr muss auch durch lokale Strategien ein global gutes Verhalten erreicht werden.

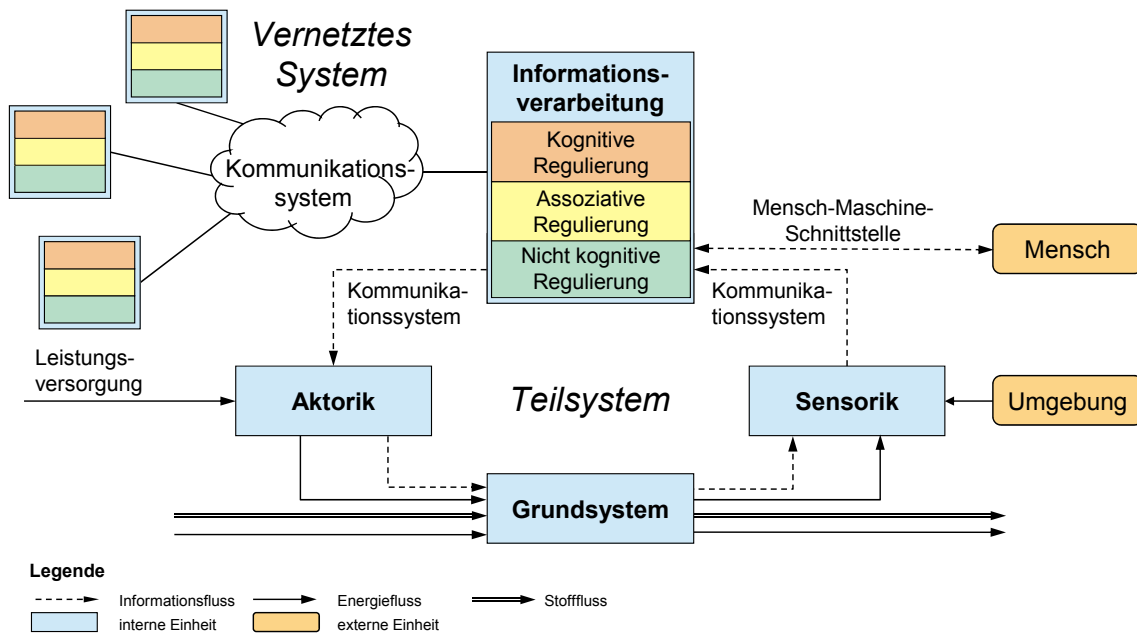


Bild 2: Intelligente mechatronische Systeme

Vor diesem Hintergrund erweist sich die Verbindung der Konzepte mit der Bezeichnung Internet der Dinge und Internet der Daten und Dienste als ein entscheidender Treiber der skizzierten Entwicklung zu cyber-physischen Systemen. Wie in Bild 3 dargestellt, handelt es sich um zwei konvergierende Entwicklungsstränge, die neue Perspektiven in vielen Lebens- und Wirtschaftsbereichen eröffnen. Diese neuen Anwendungsfelder sind in Bild 4 beispielhaft wiedergegeben.

Das **Internet der Dinge** ist letztlich das Resultat aus der oben beschriebenen Technologiekonzeption. In diesem kommunizieren physische Objekte, z.B. Werkstücke, Maschinen, Betriebsmittel, Lager- und Transportsysteme und Fertigungsleitstand, die inhärente Teilintelligenz aufweisen, über Internet oder andere Netze.

Der Wandel zu einer virtuellen Geschäftswelt basiert hingegen auf einem zunehmenden Angebot von internetbasierten Dienstleistungen und der Verfügbarkeit von großen Datenmengen, die immer schneller verarbeitet werden können.

Globale Datennetze, basierend auf Technologien wie Big Data, Cloud Computing und Smart Devices, ermöglichen ein **Internet der Daten und Dienste**, aus dem sich faszinierende Möglichkeiten für innovative Dienstleistungen – oft in geschickter Kombination von Sachleistungen – und attraktive Geschäftsmodelle ergeben [ASSW14].

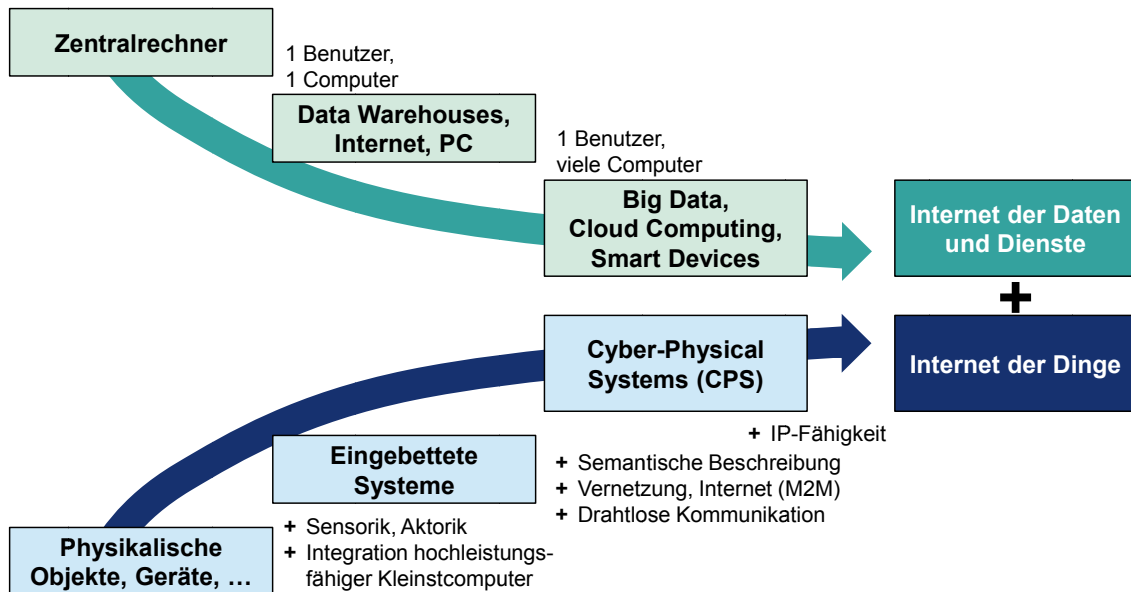


Bild 3: Zwei konvergierende Entwicklungsstränge als Innovationstreiber [FA13]

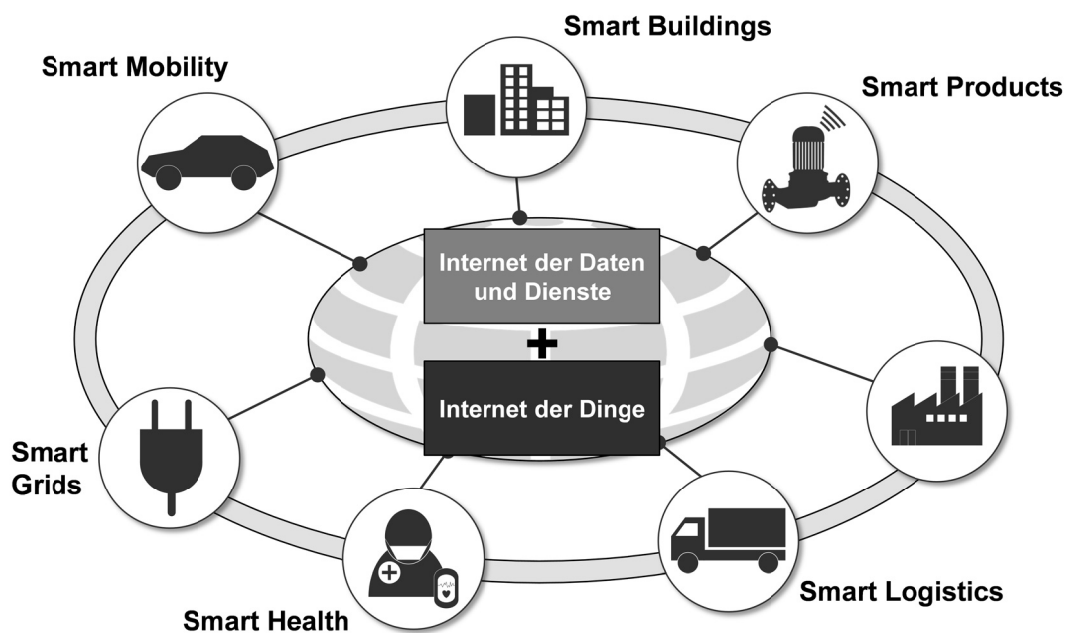


Bild 4: Anwendungsfelder von intelligenten vernetzten Systemen – Cyber-Physical Systems nach [FA13]

Die Ausprägung der beschriebenen Konzepte Internet der Dinge und Internet der Daten und Dienste auf die industrielle Wertschöpfung führt zu dem Begriff Industrie 4.0, der eine Entwicklung zum Ausdruck bringt, die in der Rückschau möglicherweise als die 4. Industrielle Revolution gesehen wird. Wir verstehen unter Industrie 4.0 die Fähigkeit von Produktionsmitteln, sich ad hoc zu leistungsfähigen Wertschöpfungsnetzwerken über Unternehmensgrenzen hinweg zu konfigurieren (siehe Bild 5). Dies setzt voraus, dass die Produktionsmittel, wie Maschinen, Betriebsmittel, Lager- und Transportsysteme eine inhärente Teilintelligenz mit den Eigenschaften adaptiv, robust und voraus-

schauend aufweisen und sich über innovative Benutzungsoberflächen jeder Zeit erklären und dem Menschen kontextkonforme Möglichkeiten der Interaktion mit dem System anbieten. In Ergänzung zu den Produktionsmitteln tragen die zu produzierenden Objekte wie Werkstücke, Baugruppen und Erzeugnisse Informationen über die Resultate bereits durchlaufener und bevorstehender Wertschöpfungsstufen mit sich, sodass das die „smarten“ Produkte sich quasi ihren Weg durch ein Wertschöpfungsnetzwerk bahnen. Die klassische strenge Strukturierung der Informationsverarbeitung in einem automatisierten Produktionssystem in eine horizontale Integration (auf gleicher Aggregationsstufe – also Maschine/Maschine oder Fertigungsbetrieb/Fertigungsbetrieb) und in eine vertikale Integration (über die klassischen Leitebenen: Sensor/Aktor-Maschinen-, Prozess-, Fertigungs- und Betriebsleitebene) ist nicht mehr zwingend, da jedes IP-fähige physische Objekt mit jedem anderen via Internet kommunizieren kann.

Die Protagonisten von Industrie 4.0 versprechen sich eine Fülle von Vorteilen, wie beispielsweise die oft genannte Fähigkeit komplexe Erzeugnisse kundenspezifisch zu den Kosten eines Großserienerzeugnisses herzustellen. Es zeichnet sich ab, dass das Paradigma der flexiblen Automation einen erheblichen Leistungsschub erfährt, der auch über den einzelnen Fertigungsbetrieb hinaus ein gesamtes unternehmensübergreifendes Wertschöpfungsnetzwerk erfasst. Die erwartete Produktivitätssteigerung dürfte entscheidend dazu beitragen, den sich in den nächsten Jahrzehnten abzeichnenden Fachkräftemangel zu kompensieren und die Wettbewerbsfähigkeit einer Hochlohnwirtschaft zu erhalten.

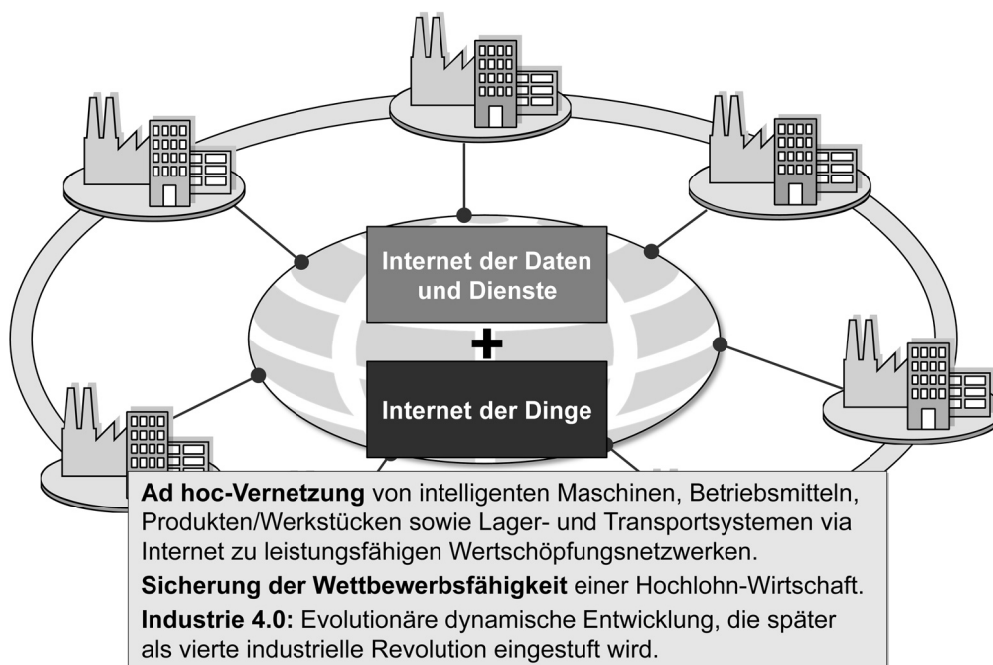


Bild 5: Industrie 4.0 – Ein neues Paradigma für industrielle Wertschöpfung

Bei aller Euphorie für Industrie 4.0 darf nicht übersehen werden, dass die Einführung und Nutzung von IT-Systemen am Ende einer gut überlegten Handlungskette steht und

nicht am Anfang; „das Pferd darf nicht von hinten aufgezäumt werden“. Bild 6 verdeutlicht die vier Stufen einer idealtypischen Handlungskette [GP14].

- 1) Vorausschau: Hier geht es um das Antizipieren der Entwicklungen von Märkten, Technologien und Geschäftsumfeldern, um die Chancen von morgen, aber auch die Bedrohungen für das etablierte Geschäft von heute frühzeitig zu erkennen.
- 2) Strategien: Hier sind Geschäfts-, Produkt- und Technologiestrategien zu entwickeln, um die erkannten Chancen von morgen rechtzeitig zu nutzen.
- 3) Prozesse: Getreu dem Motto „structure follows strategy“ sind auf dieser Ebene die Geschäftsprozesse so zu gestalten, dass sie die gewählten Strategien optimal unterstützen.
- 4) Systeme: Auf dieser Ebene sind die wohlstrukturierten Geschäftsprozesse durch IT-Systeme zu unterstützen.

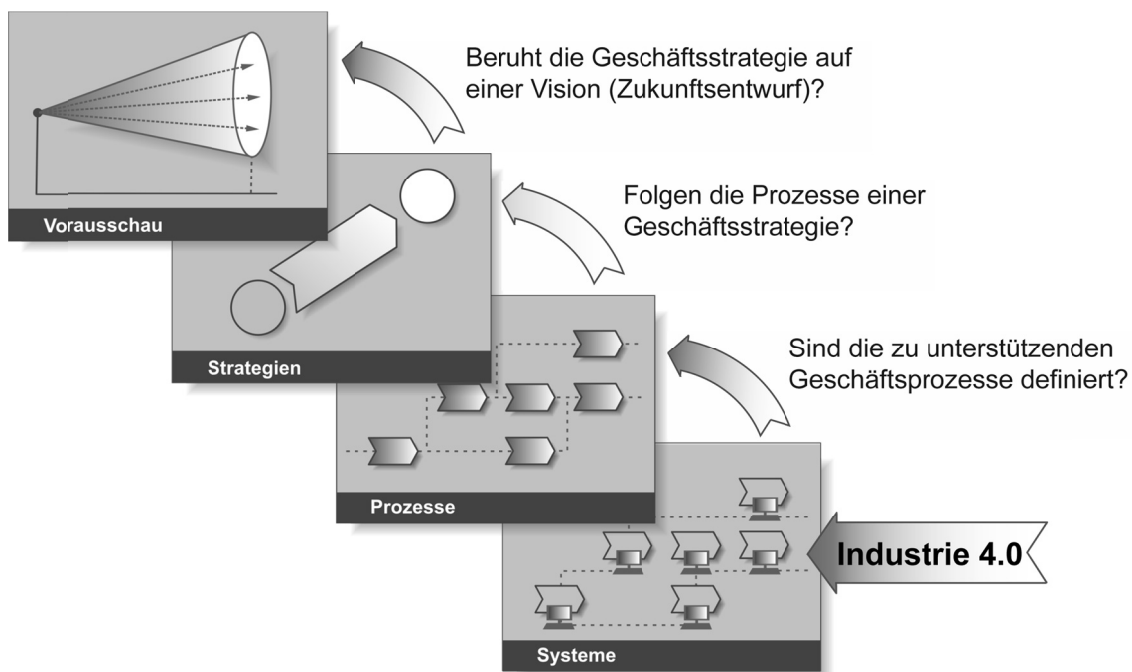


Bild 6: 4-Ebenen-Modell zur zukunftsorientierten Unternehmensgestaltung

Im Prinzip zielen die beiden oberen Ebenen auf die unternehmerische Effektivität – also das „Richtige“ tun – und die beiden unteren Ebenen auf die strategisch begründete Effizienz – also es „richtig“ tun – ab. Wie in Bild 6 angedeutet adressiert Industrie 4.0 in erster Linie die Systemebene. Bevor es hier zu Aktivitäten und Investitionen kommt, sind mit Bezug auf die drei übergeordneten Ebenen die im Bild wiedergegebenen Fragen zu stellen. Erst wenn diese beantwortet sind, haben Industrie 4.0-Lösungen eine Aussicht auf Erfolg.

Eine exportorientierte Industrienation mit der Branchenstruktur Deutschlands kann es sich nicht leisten, sich auf die Anwendung von Industrie 4.0 zu beschränken. Deutschland muss auch ein Anbieter von Industrie 4.0-Ausrüstung und -Lösungen auf den globalen Märkten werden. Der Begriff „Duale Strategie“ bringt das zum Ausdruck, was

heißt, dass Deutschland sowohl Leitmarkt ist als auch als Leitanbieter für Industrie 4.0 auftritt. Vor diesem Hintergrund stellen sich die folgenden Schlüsselfragen:

- Welche Rahmenbedingen müssen gegeben sein, damit Deutschland ein **Leitmarkt** werden kann?
- Wo liegen die Märkte für die **Leitanbieterindustrie** und was fordern diese?
- Auf welche Mitbewerber wird die deutsche Leitanbieterindustrie stoßen?
- Wie muss sich die Leitanbieterindustrie entwickeln, um auf den Märkten von morgen erfolgreich zu sein?

Im folgenden Kapitel versuchen wir derartige Fragen zu beantworten.

2 Erfolgspotentiale der Zukunft erkennen

Industrie 4.0 eröffnet eine Vielzahl von Marktleistungsinnovationen. Um diese zu erkennen, sind die Grenzen des gewohnten Denkens zu überwinden. Die Szenario-Technik ist das dafür geeignete Denkzeug. Ein Szenario ist eine allgemeinverständliche und nachvollziehbare Beschreibung einer möglichen Situation in der Zukunft, die auf einem komplexen Netz von Ausprägungen (Projektionen) von Einflussfaktoren beruht [GP14]. Bei der Vorausschau ist zwischen dem Szenariofeld und dem Gestaltungsfeld zu unterscheiden (siehe Bild 7). Das Szenariofeld umfasst das zukünftige Umfeld für Industrie 4.0 in Deutschland. Es enthält ausschließlich externe, von einzelnen Unternehmen nicht lenkbare Umfeldgrößen. Das Gestaltungsfeld dient der Vorausschau alternativer Gestaltungsmöglichkeiten von Industrie 4.0; Basis sind von Politik, Verbänden und Unternehmen beeinflussbare Lenkungsgrößen.

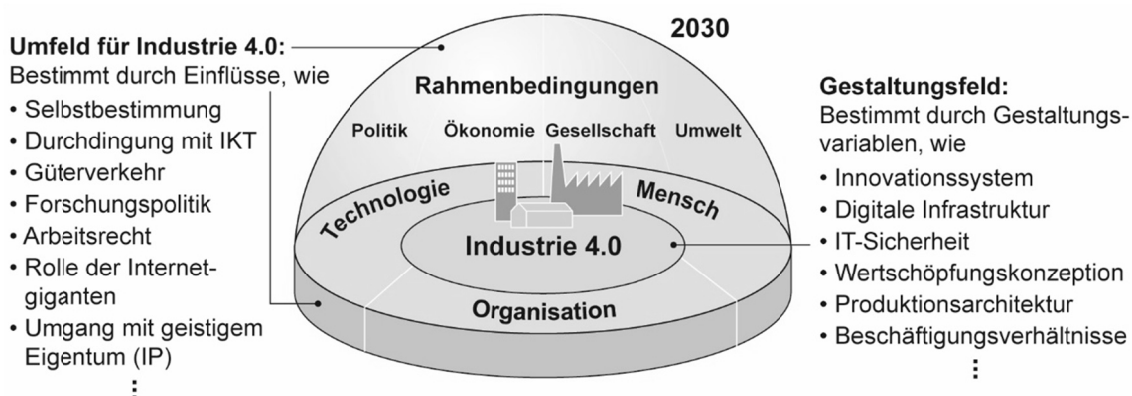


Bild 7: Umfeld Deutschland und Gestaltungsfeld Industrie 4.0

Das Vorgehen bei der Szenario-Erstellung umfasst fünf Schritte, die im Folgenden anhand des Beispiels Industrie 4.0 erläutert werden.

Einflussfaktoren identifizieren (Schritt 1): Hierzu wird das Szenariofeld in die Einflussbereiche Rahmenbedingungen, Technologie, Mensch und Organisation unterteilt. Die Einflussbereiche dienen als Suchfelder zur Identifikation von Einflussfaktoren. Dies

führt zu 61 Einflussfaktoren. Beispiele für Einflussfaktoren sind Selbstbestimmung, Struktur der Arbeit oder Durchdringung mit IKT:

- **Selbstbestimmung:** Selbstbestimmung bedeutet nach freiem Willen über sein Leben entscheiden zu können und setzt daher voraus, dass der Mensch sich über seine eigenen Ziele im Klaren ist. Es gibt zwei psychologische Ausprägungen: Der Mensch ist mit festen Vorgaben und damit einhergehender Entbindung von Entscheidungen zufrieden (z.B. aufgrund von Überforderung). Oder er will selbst entscheiden wann, wo und wie er arbeiten möchte. Im zweiten Fall erhöht selbstbestimmtes Arbeiten die Motivation und somit auch die Leistung.
- **Struktur der Arbeit:** Hierunter wird die Verteilung von hoch und gering qualifizierten Tätigkeiten verstanden. Nach HIRSCH-KREINSEN kann Industrie 4.0 zu einer ausgeprägten Polarisierung der Arbeit führen. Viele Tätigkeiten würden demnach anspruchsvoller, zugleich würde es dort, wo sich Automatisierung nicht lohnt, einen hohen Anteil manueller Arbeit geben [Hir14].
- **Durchdringung mit IKT:** Die Durchdringung der Arbeits- und Lebenswelt mit Informations- und Kommunikationstechnik wird zum einen durch die technische Entwicklung in diesem Bereich beeinflusst. Diese Entwicklungen stoßen oft weitere Innovationen in anderen Industriezweigen an und wirken daher als Befähiger (Enabler). Zum anderen spielen die Bereitschaft bzw. die Möglichkeit der Menschen, neue Informations- und Kommunikationstechnologien in ihren Alltag zu integrieren, eine große Rolle für deren Verbreitung. So kann die zunehmende Durchdringung mit Informations- und Kommunikationstechnik „als Segen“ oder „als Fluch“ empfunden werden.

Schlüsselfaktoren ermitteln (Schritt 2): Aus der relativ großen Anzahl von Einflussfaktoren sind diejenigen zu ermitteln, die das Szenariofeld besonders prägen und einen besonders hohen Einfluss auf den Untersuchungsgegenstand ausüben. Dazu wird auf eine *Einflussanalyse* zur Betrachtung der direkten und indirekten Beziehung der Einflussfaktoren und eine *Relevanzanalyse* zur Ermittlung der Bedeutung der Einflussfaktoren für das Gestaltungsfeld zurückgegriffen. Hierdurch ergeben sich 20 Schlüsselfaktoren.

Entwicklungsmöglichkeiten beschreiben (Schritt 3): Je Schlüsselfaktor werden mit einem vorgegebenen Zeithorizont (wie 2030) mehrere Entwicklungsmöglichkeiten (Projektionen) beschrieben. Die Erarbeitung von solchen alternativen Zukunftsbildern je Schlüsselfaktor ist der entscheidende Schritt der Szenario-Technik, weil damit die Bausteine für die späteren Szenarien geschaffen werden. Es gilt sowohl aus heutiger Sicht plausible, als auch extreme, aber vorstellbare Entwicklungen in Betracht zu ziehen. Bezogen auf den Schlüsselfaktor „Selbstbestimmung“ können drei Projektionen ermittelt werden (siehe Bild 8). Neben der eher wahrscheinlichen Entwicklung „Autonomie zählt“ sind auch die Projektionen „Stress vermeiden“ und „Trügerisches Bild der

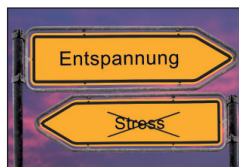
Selbstbestimmung“ denkbar. Der Schlüsselfaktor „Durchdringung mit IKT“ weist drei Projektionen auf (siehe Bild 9).

1A Autonomie zählt



Die Menschen in Deutschland haben nahezu in allen Bereichen des Erwerbslebens so viele Möglichkeiten ihr Leben zu gestalten wie nie zuvor. Bildung, Wohnort, Beruf u.v.m. sind variable Größen, die nicht länger durch die Herkunft determiniert sind. Die neuen Freiheiten haben gravierende Auswirkungen auf die Lebensentwürfe der Menschen. Arbeitnehmer verlangen Verantwortung und Entscheidungsspielräume. Industrie 4.0 hat hier neue Perspektiven eröffnet. Der Umfang der Entfaltungsmöglichkeiten bestimmt die Auswahl des Arbeitsverhältnisses.

1B Stress vermeiden



Die psychische Belastung der Beschäftigten steigt kontinuierlich. Neue Entscheidungsfreiräume sind Fluch und Segen zugleich. Sie ermöglichen den Menschen sich selbst zu verwirklichen, führen oftmals aber auch zur Überlastung. Die Zahl der Burn-out-Erkrankten in Deutschland steigt seit Jahren. Arbeitsverhältnisse, die mit weniger Engagement und Verantwortung verbunden sind, finden hohen Zuspruch. Viele trennen strikt Arbeit und Freizeit. Es herrscht die Haltung vor, andere unternehmen zu lassen und sich darauf zu beschränken, zu definieren, wie das für einen selbst eingerichtet sein muss, um angenehm zu sein.

1C Trügerisches Bild der Selbstbestimmung



Leistungsfähige Assistenzsysteme unterstützen die Beschäftigten in nahezu allen Bereichen der Wirtschaft. Sie liefern vielfältige Informationen, nehmen Diagnosen vor und unterbreiten Vorschläge für Handlungen. Auf dem ersten Blick sind die Benutzer der Assistenzsysteme Entscheider. Bei näherem Hinsehen sind es Getriebene; gut organisierte Leistungserstellungsprozesse und die damit verbundenen Assistenzsysteme geben den Takt vor. Die Beschäftigten sind häufig Marionetten der Algorithmen.

Bild 8: Projektionen für den Schlüsselfaktor „Selbstbestimmung“

8A Maximal vernetzte Welt



Die rapide Durchdringung der Arbeits- und Freizeitwelt mit Informations- und Kommunikationstechnik hat in den vergangenen Jahren weiter zugenommen. Jeder hat immer und überall Zugriff auf Informationen und Dienste. Der Einsatz semantischer Technologien ermöglicht eine effiziente Bewältigung der Fülle von verfügbaren Informationen. Durchgesetzt haben sich vor allem intuitiv zu bedienende Systeme. Die Faszination der neuen Möglichkeiten blendet die Menschen: Sicherheitsrisiken, wie die Weitergabe sensibler Daten, werden verdrängt. Es herrscht ein Widerspruch zwischen der theoretischen Wertschätzung und der gelebten Achtlosigkeit im Umgang mit der eigenen Privatsphäre.

8B Informationseliten



Die täglich produzierte Datenmenge erlangt neue Ausmaße. Viele sehen darin ungeahnte Möglichkeiten. Die ungeheure Fülle an Daten erweist sich wie von vielen vorhergesagt als das Gold des 21. Jahrhunderts. Doch die freie Verfügbarkeit von Daten ist trügerisch. Nur wenigen gelingt es, die technologischen Möglichkeiten zu nutzen und aus Daten Informationen zu generieren: Es bilden sich Informationseliten. Diese beherrschen die Algorithmen und ziehen konsequent Nutzen aus den Daten. Offensichtlich herrscht kein Mangel an Informationen; aber die Menschen fühlen sich nicht informiert.

8C IT-Frustration



Der IKT-Hype der Freizeitwelt hat keinen Einzug in die Produktion gefunden. Die notwendigen hohen Anforderungen an die Verlässlichkeit der IT-Systeme werden nicht erfüllt. Die erwarteten Effizienzsteigerungen sind ausgeblieben. Auch im Privatleben ändert sich der Umgang mit IKT. Die Skepsis gegenüber IKT nimmt stark zu, weil die Folgen des „lockeren“ Umgangs nun allmählich sichtbar werden. Menschen sehen sich mit der totalen Transparenz konfrontiert und streben nach Privatheit. Privatheit bedeutet die Fähigkeit, seine Sichtbarkeit im Netz selbst definieren und regulieren zu können.

Bild 9: Projektionen für den Schlüsselfaktor „Durchdringung mit IKT“

Konsistente Zukunftsbilder (Szenarien) ermitteln (Schritt 4): Entscheidend für die Glaubwürdigkeit von Zukunftsbildern ist die Konsistenz, d.h. die Widerspruchsfreiheit der einzelnen Projektionen zueinander. Basis dafür ist eine paarweise Konsistenzbewertung. Der Bewertungsbereich erstreckt sich von einer *starken gegenseitigen Unterstützung* (5) bis zur *totalen Inkonsistenz* (1). Bild 10 zeigt das Beispiel einer Konsistenzbewertung der Faktoren „Durchdringung mit IKT“ und „Selbstbestimmung“. Die Projektionen „Informationseliten“ und „Autonomie zählt“ sind inkonsistent und können daher nicht gemeinsam in einem Szenario vorkommen. Dagegen begünstigen sich die Projektionen „Informationseliten“ und „Trügerisches Bild der Selbstbestimmung“ stark; sie können gut in einem in sich schlüssigen Szenario vorkommen.




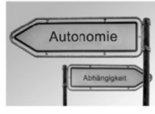
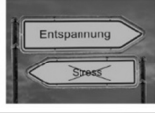

Konsistenzmatrix Fragestellung: „Wie verträgt sich Zukunftsprojektion i (Zeile) mit Zukunftsprojektion j (Spalte)?“ Bewertungskala: 1 = totale Inkonsistenz 2 = partielle Inkonsistenz 3 = neutral oder voneinander unabhängig 4 = gegenseitiges Begünstigen 5 = starke gegenseitige Unterstützung		SF	Durchdringung mit IKT			
		Zukunftsprojektion	Maximal vernetzte Welt 	Informationseliten 	IT-Frustration 	
SF	Zukunftsprojektion	Nr.	8A	8B	8C	
Selbstbestimmung	Autonomie zählt 	1A	4	1	4	Dass in einer Welt der Informationeliten die Autonomie aller zählt, ist inkonsistent.
	Stress vermeiden 	1B	2	4	2	Ein trügerisches Bild der Selbstbestimmung geht einher mit Informationeliten.
	Trügerisches Bild der Selbstbestimmung 	1C	4	5	2	

Bild 10: .Paarweise Bewertung der Konsistenz von Zukunftsprojektionen

Auf Basis der ausgefüllten Konsistenzmatrix werden mithilfe der Scenario-Software konsistente Zukunftsbilder ermittelt. Zunächst werden konsistente Projektionsbündel bestimmt; das sind Kombinationen von Projektionen, die besonders gut zueinander passen. Sie bestehen je Schlüsselfaktor aus genau einer Projektion. Im Rahmen einer Clusteranalyse werden die konsistentesten Projektionsbündel zu Szenarien zusammengefasst. Die Ergebnisse der Clusteranalyse werden mit dem sog. Zukunftsraum-Mapping visualisiert. Hier werden die verschiedenen Projektionsbündel in einer Hilfebene dargestellt. Es entsteht eine „Landkarte der Zukunft“. Bevorzugtes Instrument des Zukunftsraum-Mappings ist die Multidimensionale Skalierung (MDS). Sie liefert je Projektionsbündel zwei Koordinatenwerte, so dass die Projektionsbündel auf einer Ebene positioniert werden können. Dabei werden die Projektionsbündel so positioniert, dass ähnliche Bündel möglichst dicht beieinander und unähnliche Bündel möglichst weit voneinander entfernt liegen. In einer derartigen Graphik zeigen sich Szenarien als

„Bündel-Gruppen“. Für das betrachtete Beispiel führt das zu vier **Umfeldszenarien** (siehe Bild 11).

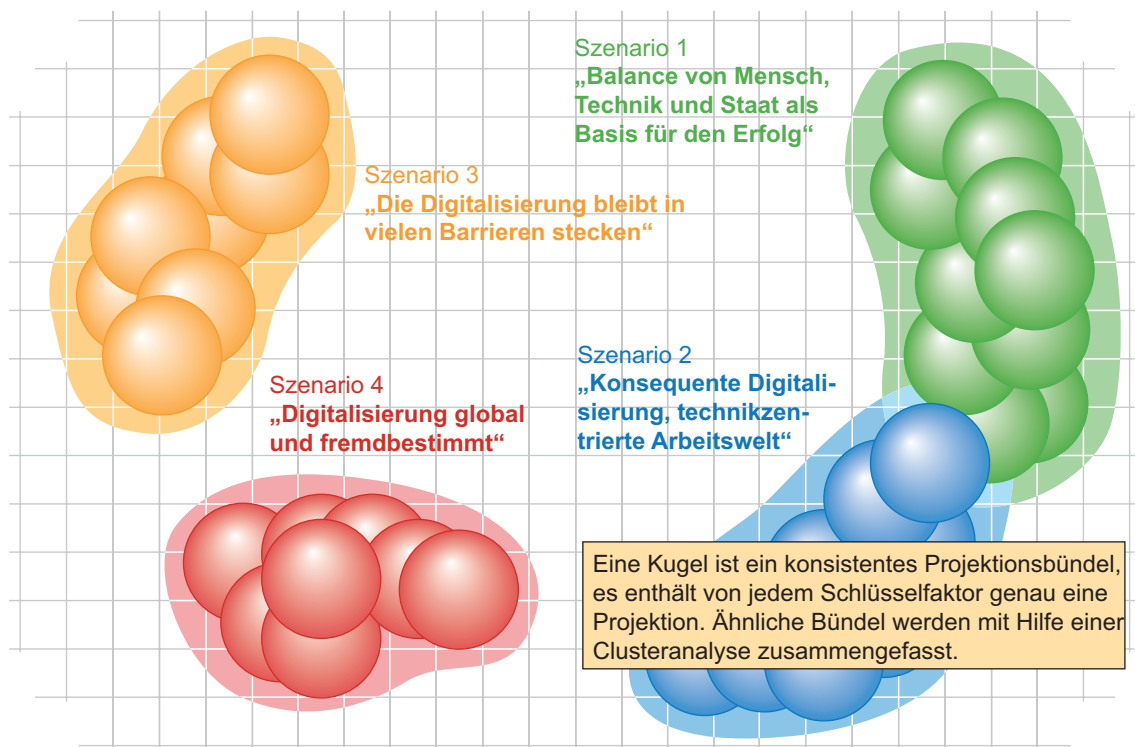


Bild 11: Visualisierung der Szenarien mittels multidimensionaler Skalierung

Szenarien „in Prosa“ beschreiben (Schritt 5): Entscheidend für den Erfolg der Szenarien ist, dass sie nachvollziehbar, verständlich und leicht kommunizierbar sind. Bei der Beschreibung der Szenarien „in Prosa“ wird auf die Textbausteine zurückgegriffen, die im Zuge der Bildung der Zukunftsprojektionen formuliert worden sind. Nachfolgend sind beispielhaft die Management-Summaries von Szenario 1 und Szenario 4 angeführt.

Szenario 1: „Balance von Mensch, Technik und Staat als Basis für den Erfolg“

- Die Menschen leben bewusst und gesund. Die Technik entlastet und unterstützt die Menschen.
- Die Digitalisierung hat Aus- und Weiterbildung stark verändert; Lehre und Lernen erfolgt überwiegend orts- und zeitungebunden; Massive Open Online Courses setzen sich durch. Die Zahl der Lehrenden ist erheblich gesunken.
- Der Staat engagiert sich wirksam, wenngleich es Tendenzen gibt, das Engagement des Staates zurückzufahren. Die Forschungsförderung ist sehr erfolgreich. Die seit Jahren erfolgreich praktizierte Sozialpartnerschaft steht unter Druck; die Liberalisierung des Arbeitsrechts schreitet voran.
- Die Möglichkeiten der Informations- und Kommunikationstechnik werden genutzt, um innovative Konzepte einer „Teaching & Learning“ Factory zu realisieren. Im Fokus steht die Unterstützung der Tätigkeiten durch Assistenzsysteme.

Es ist eine De-Polarisierung der Struktur der Arbeit zu verzeichnen. Die Löhne gleichen sich global tendenziell an. Die soziale Ungleichheit hat abgenommen.

- Es haben sich länderunabhängige Wirtschaftszonen etabliert. Der Einfluss der Internetgiganten auf Wirtschaft und Gesellschaft liegt deutlich unter den Befürchtungen. Die Bürger und Bürgerinnen legen größten Wert darauf, über ihre Daten zu bestimmen. Vielfalt hat sich durchgesetzt.
- Die digitale Vernetzung der Welt ist hoch. Agile Allianzen prägen die Wertschöpfungsnetzwerke.
- Die Innovationsdynamik ist hoch: Open Source ist sehr verbreitet; Patente verlieren an Bedeutung. Offene Standards setzen sich durch.
- Die Migrationseffizienz ist hoch: Ältere Maschinen und Anlagen können durch innovative Upgrade-Technologien ohne großen Aufwand in das Digitalisierungszeitalter überführt werden. Die Benutzungsfreundlichkeit ist hoch und die IT-Sicherheit gewährleistet.
- Das Umweltbewusstsein ist stark ausgeprägt. Die Kreislaufwirtschaft hat sich durchgesetzt; Güter werden lokal produziert.

Szenario 4: „Digitalisierung global und fremdbestimmt“

- Die Technikzentrierung prägt die Arbeitswelt; die Maschinen geben den Takt vor; die Menschen sind teils Marionetten der Algorithmen.
- Die Digitalisierung hat Aus- und Weiterbildung verändert; Lehre und Lernen erfolgt vermehrt orts- und zeitungebunden. In vielen Bereichen wird weiterhin auf die bewährten Konzepte zurückgegriffen; der menschliche Kontakt und das Mentoring zählen nach wie vor.
- Der Staat hat sich aus vielen Bereichen zurückgezogen. Die Liberalisierung des Arbeitsrechts schreitet voran. Auch die Forschungsförderung wurde reduziert, was zur Zweitklassigkeit der Forschungslandschaft geführt hat; Spitzenforscher sind in die Wirtschaft abgewandert.
- Hochflexible Automatisierung ist gang und gäbe. Nur ein relativ kleiner Teil an Aufgaben ist anspruchsvoller geworden; hier wurde der Bediener zum Entscheider. Einfache Tätigkeiten in der Produktion sind teilweise entfallen. Die Löhne gleichen sich auf eher niedrigem Niveau global tendenziell an.
- Die vergangenen Wirtschaftskrisen haben die Währungs- und Wirtschaftsräume auf die Probe gestellt. Dadurch sind die jeweiligen Länder enger zusammengewachsen. Getrieben durch die fortschreitende Digitalisierung und die Verbreitung von virtuellen Währungen (Bitcoins) entstehen auch unabhängig von den Grenzen der Nationalstaaten Wirtschaftsräume. Der Einfluss der Internetgiganten hat ungeahnte Ausmaße erlangt.

- Es haben sich zwei Klassen von Kooperationspartnern entwickelt: Relativ wenige kreative Gestalter und viele austauschbare Ausführende. Diese Entwicklung wird durch die Bildung von Informationseliten forciert. Diesen gelingt es, die technologischen Möglichkeiten zu nutzen und aus Daten Informationen und Wissen zu generieren.
- Die Standardisierung kommt nicht voran, weil die führenden Industrienationen sich primär durch Eigeninteresse leiten lassen. Nur selten setzen sich Standards durch. Angesichts von vereinzelt Wildwuchs und hohen Engineeringaufwänden in der Automatisierung bevorzugt ein großer Teil der Anwender Produkte der dominierenden Anbieter; selbst wenn diese nicht innovativ und teuer sind; aber man weiß, was man erhält. Vor diesem Hintergrund gibt es Bestrebungen, das Patentwesen zu rationalisieren und global zu organisieren.
- In ausgewählten Bereichen ermöglicht die dynamische Entwicklung der IKT technische Systeme mit einer inhärenten Teilintelligenz. Oft wird die Komplexität der Technik nicht beherrscht. Digitalisierung und Automatisierung sind mit viel Aufwand verbunden. In der Bevölkerung und der Wirtschaft fehlt ein Sicherheitsbewusstsein.
- Das Umweltbewusstsein ist gering; die Kreislaufwirtschaft bleibt unter „ferner liefen“. Der Güterverkehr führt zum Dauerstau auf der Straße; nur zaghafte Investitionen werden in den Ausbau des intelligenten Güterverkehrs getätigt.

Darüber hinaus empfiehlt sich die bildliche Darstellung der Szenarien in Form von so genannten Pictures of the Future oder Collagen. Für eine Collage wird jede Projektion mit einem treffenden Bild versehen. Anschließend werden die Bilder entsprechend der in einem Szenario auftretenden Projektionen zusammengeführt. Bild 12 zeigt beispielhaft die Collage des Szenarios 1.

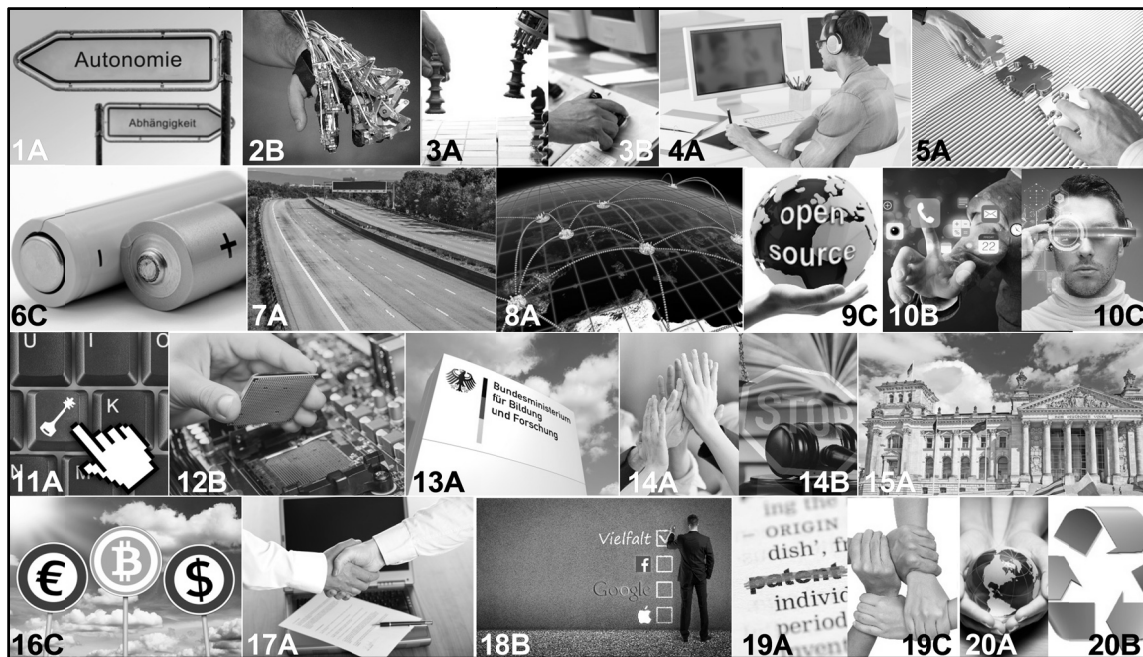


Bild 12: Szenario 1 „Balance von Mensch, Technik und Staat als Basis für den Erfolg“

Zur **Auswahl eines Referenzszenarios** als Ausgangspunkt für die Entwicklung einer Marktleistung werden die Szenarien anhand ihrer Eintrittswahrscheinlichkeit und ihrer Auswirkungen auf den Produktions- und Wirtschaftsstandort Deutschland bewertet. Da das Szenario 1 eine starke Veränderung des Produktions- und Wirtschaftsstandorts Deutschland mit sich bringt und eine hohe Eintrittswahrscheinlichkeit aufweist, wird es als Referenzszenario ausgewählt.

Bei der Erstellung der **Gestaltungsfeld-Szenarien** wird analog verfahren. Für Industrie 4.0 führt dies zu 18 Gestaltungsfaktoren. Beispiele sind das Innovationssystem, die Wertschöpfungskonzeption und die Produktionsarchitektur:

- **Innovationssystem:** Unter einem Innovationssystem wird die Gesamtheit an Organisationen und Institutionen – vor allem Unternehmen, Universitäten und Forschungseinrichtungen – verstanden, die einzeln oder in Kooperation miteinander an der Erschaffung, Verbreitung und Anwendung wissenschaftlichen oder technologischen Wissens beteiligt sind [DIW14-01].
- **Wertschöpfungskonzeption:** Das Wertschöpfungskonzept definiert, wie das Wertversprechen gegenüber dem Kunden erfüllt wird, das heißt, wie durch die Kombination von unternehmensinternen und externen Ressourcen und Fähigkeiten in einem Wertschöpfungsnetzwerk Wert geschaffen wird [BR11].
- **Produktionsarchitektur:** In der Produktionsarchitektur werden die Struktur und die Schnittstellen, der Standardisierungs- und Flexibilisierungsgrad sowie Regeln zur Gestaltung der Produktion festgelegt [TS13].

Für die Gestaltungsfaktoren wurden mögliche Ausprägungen ermittelt. Bild 13 zeigt beispielhaft die Ausprägungen für den Gestaltungsfaktor „Wertschöpfungskonzeption“.

11A Systemkopf



Viele deutsche Unternehmen verfolgen die Wertschöpfungskonzeption „Systemkopf“: Sie konzentrieren sich auf Funktionen mit dem höchsten Differenzierungsgrad und damit Wertschöpfungspotential wie z.B. Forschung und Entwicklung, Design, Marketing und hochwertige Produktion. Sie organisieren von Deutschland aus global verteilte Wertschöpfungsnetze und lagern insbesondere Einfacharbeit aus.

11B Intelligenter Produktionsdienstleister



Industrie 4.0 führt zu einer Renaissance von Deutschland als Produktionsstandort. Deutschland ist weltweit führend im Aufbau von vernetzten, intelligenten Fertigungssystemen. Dies ermöglicht Fertigungskosten auf bzw. sogar teilweise unter dem Niveau der Niedriglohnländer.

11C Ganzheitlicher Wertschöpfungsansatz



Die deutsche Industrie ist kreativer Gestalter und Umsetzer zugleich. Die klassische Produktorientierung wurde durch eine Nutzenorientierung abgelöst. Geschäftsmodellideen made in Germany sind Realität. Die Unternehmen verfügen über das notwendige Kapital sowie ein exzellentes Marketing. In Kombination mit den traditionellen Stärken den Bereichen FuE, Produktion etc. nehmen viele deutsche Unternehmen zentrale Rollen in den globalen Wertschöpfungsnetzen ein.

Bild 13: Ausprägungen des Gestaltungsfaktors „Wertschöpfungskonzeption“

Die Ausprägungen der Gestaltungsfaktoren wurden paarweise auf ihre Verträglichkeit bewertet. Basierend auf der Konsistenzmatrix ergeben sich drei **Gestaltungsfeldszenarien**, die folgende Titel erhalten: 1) „Ausgeprägte Interoperabilität“, 2) „Verlässlichkeit im Fokus“ und 3) „Fortsetzung der Offline-Produktion“.

Die entsprechenden Konzepte zur Gestaltung von Industrie 4.0 sind in Relation zu den ermittelten Entwicklungen des Umfelds zu setzen. Die zugrundeliegende Fragestellung lautet: Welche Kombinationen von Gestaltungs- und Umfeldszenarien passen gut zueinander? Die Bewertungsskala reicht von ++ (sehr hohe Konsistenz) bis -- (sehr hohe Inkonsistenz). Sehr hohe Konsistenz bedeutet, dass das Gestaltungsfeldszenario sehr gut in einem Umfeldszenario vorstellbar ist. Das Gegenteil ist der Fall, wenn die Gegenüberstellung mit einer sehr hohen Inkonsistenz bewertet wird. Hier ist das entsprechende Gestaltungsfeldszenario bei Eintreten des Umfeldszenarios nicht sinnvoll. Bild 14 zeigt die Gegenüberstellung der drei entwickelten Gestaltungsfeldszenarien in den Zeilen mit den vier Umfeldszenarien in den Spalten. Aus der Gegenüberstellung von Umfeld- und Gestaltungsfeldszenarien eröffnen sich für Unternehmen neue Perspektiven für ein attraktives Geschäft im Kontext von Industrie 4.0. Für ein Unternehmen, das auf der Suche nach einer neuen Marktleistung ist, ist das Gestaltungsfeldszenario von besonderer Relevanz, das eine hohe Konsistenz zum ausgewählten Referenzszenario (Szenario 1) aufweist, da dies der voraussichtlichen Handlung seines Kunden im Kontext von Industrie 4.0 entspricht. Im vorliegenden Fall trifft dies auf Gestaltungsfeldszenario 2 zu: „Verlässlichkeit im Fokus“ stellt für den Kunden eine konsistente Handlungsoption dar, falls das Referenzszenario eintritt.

Nachfolgend wird das Vorgehen zur Ableitung einer strategischen Stoßrichtung zur Gestaltung des Geschäfts von morgen exemplarisch an der Marktleistung Schaltschrankbau erläutert. Wir versetzen uns in die Perspektive eines Komponentenlieferanten, der das Gestaltungsfeldszenario mit der Brille seines Kunden (Schaltschrankbauer) betrachtet. Die zugrundeliegende Fragestellung lautet demnach: Wie gestaltet der Schaltschrankbauer sein Geschäft im Lichte des Gestaltungsfeldszenarios 2?

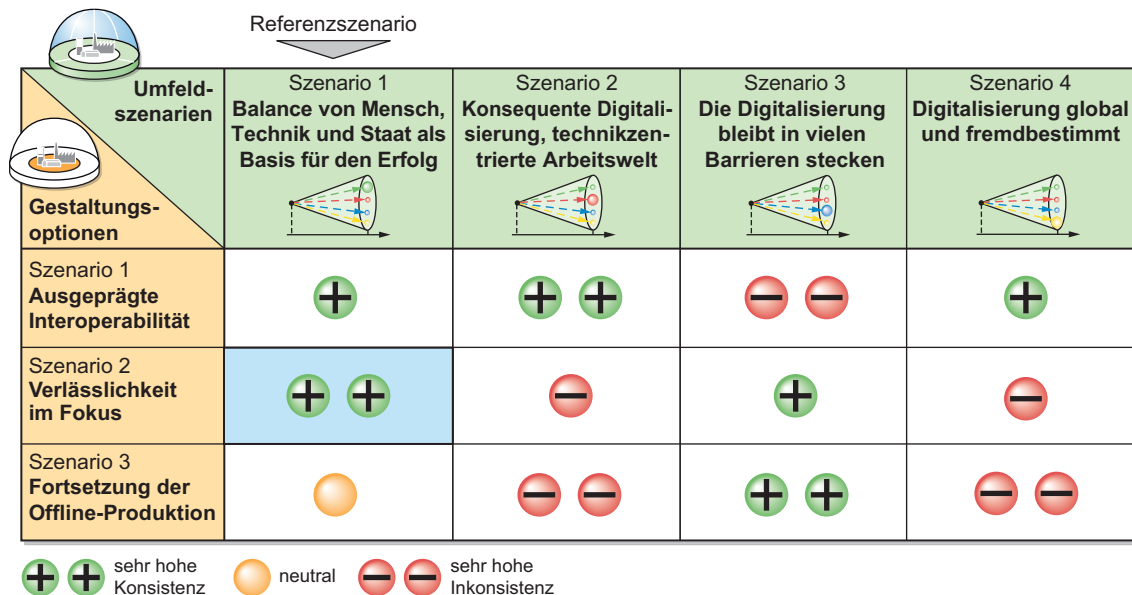


Bild 14: Ableitung von strategischen Stoßrichtungen

Derzeit entfällt der mit Abstand größte Anteil an der Durchlaufzeit eines Schaltschranks auf die Montage, insbesondere auf die Verdrahtung und die Prüfung. Diese Aufgaben werden manuell durchgeführt und sind aufgrund ihrer hohen Komplexität sehr fehleranfällig. Eine Automatisierung ist in der Regel jedoch nicht wirtschaftlich. Die Zusammenarbeit der Wertschöpfungspartner wird derzeit durch Medienbrüche¹ erschwert. Dies könnte sich im Lichte von Industrie 4.0 ändern (siehe Bild 15). Das Gestaltungsfeldszenario 2 „Verlässlichkeit im Fokus“ ermöglicht einem Komponentenlieferanten die Stoßrichtung Vorwärtsintegration, d.h. die Ergänzung seiner Marktleistung durch Dienstleistungen, die der Kunde bislang im Prinzip selbst durchgeführt hat. Eine solche Dienstleistung wäre eine AR²-basierte Montageunterstützung. Die entsprechende Wertschöpfungskette sähe wie folgt aus.

1 Unter einem Medienbruch wird der Wechsel zwischen verschiedenen Medien verstanden. Hierunter fällt insbesondere der Wechsel zwischen Papier und computerunterstützten Informationssystemen im Verlauf eines Geschäftsprozesses [All05].

2 AR: Augmented Reality, computergenerierte Informationen (z.B. Darstellung des zu montierenden Bauteils) werden in das Sichtfeld des Monteurs eingeblendet.

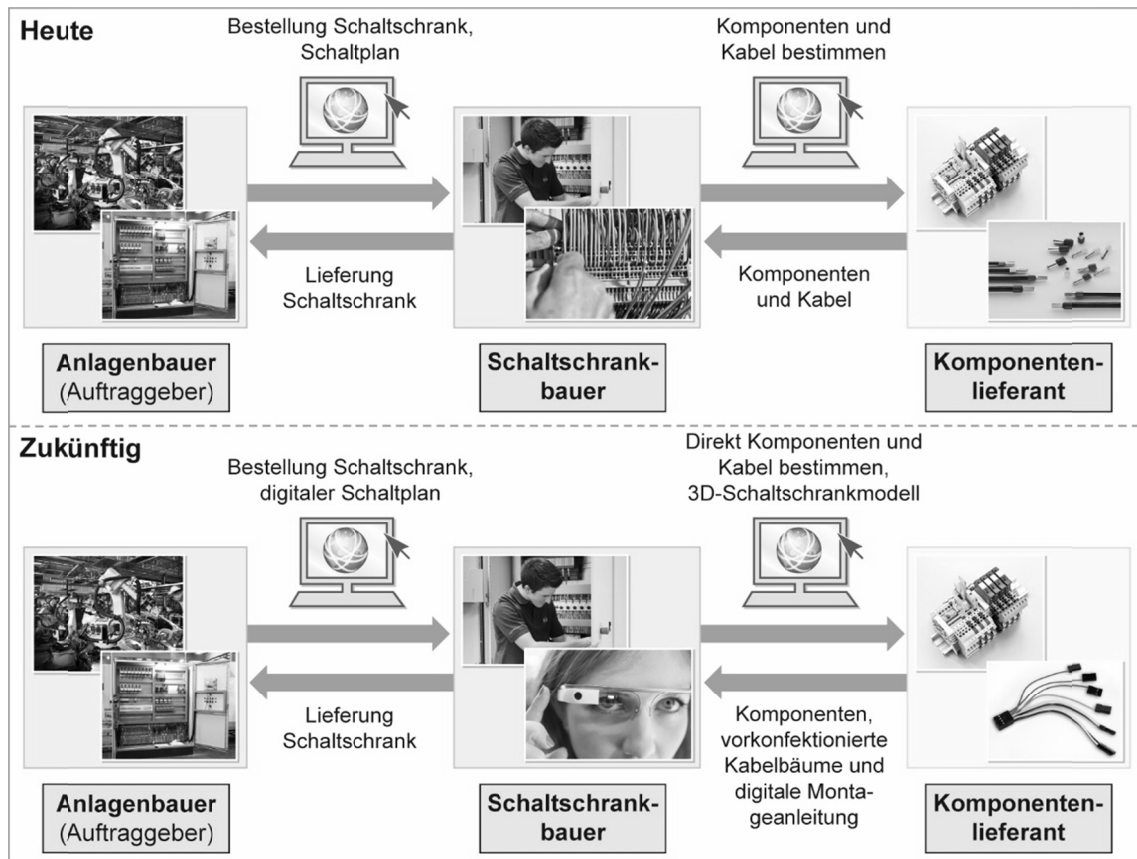


Bild 15: Wertschöpfungskette Schaltschrankbau

Der Anlagenbauer (Hersteller von maschinenbaulichen Anlagen) vergibt den Auftrag über einen Schaltschrank an den Schaltschrankbauer. Wesentlicher Teil des Auftrags ist der digitale Schaltplan. Dieser wird zusammen mit dem CAD-Modell des Schaltschranks vom Schaltschrankbauer an den Komponentenlieferant weitergegeben. Auf Basis beider Dateien erstellt der Komponentenlieferant passgenaue, vorkonfektionierte Kabelbäume. Zusätzlich wird eine digitale Montageanleitung bereitgestellt. Diese unterstützt den Schaltschrankbauer mittels einer AR-Brille bei der Bestückung des Schaltschranks mit Komponenten und vorkonfektionierten Kabelbäumen. Auf diese Weise resultiert aus der Erweiterung der Leistungen des Komponentenlieferanten und dem Einsatz neuer Techniken zur Unterstützung der Montage und Prüfung beim Schaltschrankbau ein gesteigener Kundennutzen für den Anlagenbauer in Form eines zuverlässigeren und schneller gelieferten Schaltschranks.

Die Geschäftsidee „Angebot vorkonfektioniierter Kabelbäume und Montageunterstützung durch AR“ ist anschließend zu einem Wert versprechenden Geschäftsmodell auszugestalten. Geschäftsmodelle leisten einen signifikanten Beitrag zur Differenzierung im Wettbewerb und lassen sich nicht ohne weiteres imitieren [Sch13]. Aufgabe der Geschäftsmodellentwicklung ist es, die einzelnen Geschäftsmodellelemente zu spezifizieren, sodass sich ein schlüssiges, d.h. in sich konsistentes Geschäftsmodell ergibt. Am Heinz Nixdorf Institut wurde hierfür eine Systematik entwickelt, welche es ermöglicht,

konsistente Geschäftsmodellalternativen zu entwickeln [GRK13], [Kös14]. Grundlage ist eine Geschäftsmodellstruktur, die aus den vier Partialmodellen Angebots-, Kunden-, Wertschöpfungs- und Finanzmodell besteht. Dabei können jedem Partialmodell einzelne Geschäftsmodellelemente zugeordnet werden (Kundensegmente, Nutzenversprechen etc.) (vgl. Bild 16).

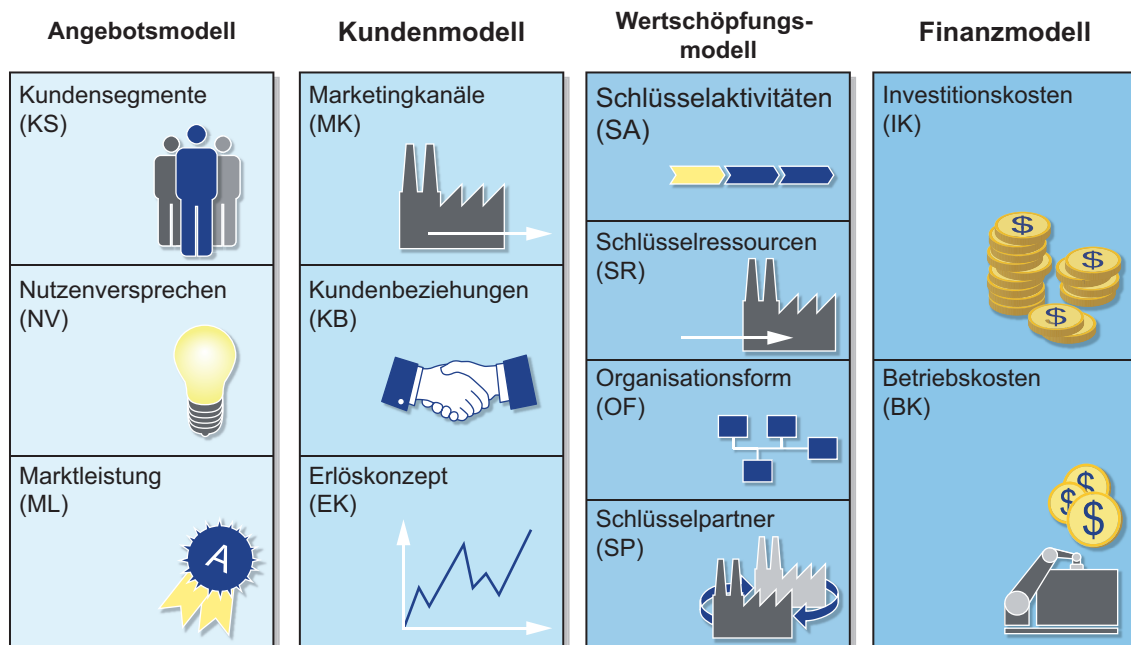


Bild 16: Struktur eines Geschäftsmodells [GA14], [Kös14]

Die Geschäftsmodellelemente dienen als Suchfelder für Geschäftsmodellvariablen und zugehörige Gestaltungsoptionen. Geschäftsmodellvariablen sind Einflussgrößen (Hebel), die zur Gestaltung eines Geschäftsmodells verfügbar sind. Gestaltungsoptionen sind die je Variable möglichen Handlungsalternativen. Das Geschäftsmodellelement Kundensegment enthält z.B. die Geschäftsmodellvariable „Flexibilitätsanspruch Kabelbaumlösung“. Hierbei existieren die drei Gestaltungsoptionen „Hohe Flexibilität erwünscht“, „Wenig Flexibilität gewünscht“ und „Standardisierte Kabelbäume“.

Die Systematik folgt der Basisidee, dass schlüssige Geschäftsmodelle auf der konsistenten Kombination von Gestaltungsoptionen beruhen. Aus der Sammlung der relevanten Geschäftsmodellvariablen und Gestaltungsoptionen sind daher diejenigen miteinander zu kombinieren, die in einem Geschäftsmodell gut zusammenpassen. Hierbei wird analog zur Szenario-Technik vorgegangen. Auf Grundlage einer paarweisen Verträglichkeitsbewertung (siehe Bild 10) und anschließenden Clusteranalyse werden alternative Geschäftsmodelle entwickelt. Für den Komponentenhersteller der elektrischen Verbindungstechnik ergeben sich auf diese Weise acht konsistente Geschäftsmodelle. Bild 17 zeigt einen Auszug der entsprechenden Ausprägungsliste [GAD+14].

			Geschäftsmodellalternative					
GME	GMV	Gestaltungsoptionen	Nr.	GMA1	GMA2	GMA3	GMA4	...
Kundensegmente	Geschäftsgegenstand	Engineeringdienstleister	1A			100		
		Produktionsdienstleister	1B		100			
		Schaltschrankbauer	1C	100				100
	Flexibilitätsanspruch Kabelbaumlösung	Hohe Flexibilität gewünscht	2A	20		85	100	
		Wenig Flexibilität benötigt	2B		40	15		
		Standardisierte Kabelbäume	2C	80	60			
	Softwareunterstützung von Entwicklung und	Durchgängig ohne Medienbrüche	3A	85	100	25	10	
		Medienbrüche zur Fertigung	3B	10				85
		Keine Softwareunterstützung	3C	5		75	5	
...	...							
Nutzenversprechen	Innovation	Reduktion Einbauzeit	11A	80		100	40	
		Funktionsintegration	11B				40	
		Reduktion Montagerisiko	11C	20		20	20	
		Reduktion Prüfaufwand	11D					
		Plug & Work	11E		100			
						
...								

GME Geschäftselement Eindeutige Ausprägung Alternative Ausprägung
GMA Geschäftsmodellalternative Dominante Ausprägung Ausprägung tritt nicht auf

Bild 17: Ausprägungsliste der ermittelten Geschäftsmodellalternativen [GAD+14]

Aus den acht Geschäftsmodellalternativen ist das Erfolg versprechendste auszuwählen. Für die Bewertung der entwickelten Geschäftsmodellalternativen werden die Dimensionen Attraktivität und Erreichbarkeit im Rahmen einer Nutzwertanalyse bewertet. Die Attraktivität weist die Kriterien Strategiekonformität, Wettbewerbsfähigkeit und Zukunfts-Fit auf [PG05]. Die Erreichbarkeit bewertet den finanziellen und zeitlichen Aufwand, der mit dem Wechsel zu einer Geschäftsmodelloption verbunden ist. Die damit verbundenen Kriterien sind Investitionsvolumen und mögliche Markteintrittsbarrieren. Im vorliegenden Beispiel wird die Geschäftsmodellalternative 1 „Vorkonfektionierte Kabelbäume mit AR-Betreibermodell“ ausgewählt. In diesem Geschäftsmodell verkauft der Komponentenhersteller vorkonfektionierte Kabelbäume, welche die Einbauzeit für den Schaltschrankbauer reduzieren. Zugleich bietet er an, dem Schaltschrankbauer einen AR-basierten Dienst inkl. Hardware (AR-Brille etc.) und Software sowie den 3D-Modellen und 3D-Daten zur Unterstützung der Montage im Rahmen eines Betreibermodells bereitzustellen.

3 Systematik der Produktentstehung

Die Informations- und Kommunikationstechnik ist der wesentliche Treiber für die Erzeugnisse der hier betrachteten Branchen. Daraus ergeben sich hohe Anforderungen an den sogenannten Produktentstehungsprozess. Dieser Prozess erstreckt sich von der Pro-

dukt- bzw. Geschäftsidee bis zum Serienanlauf (Start of Production – SOP) und weist die Aufgabenbereiche Strategische Produktplanung, Produktentwicklung, Dienstleistungsentwicklung und Produktionssystementwicklung auf. Die Produktionssystementwicklung beinhaltet im Prinzip die Fertigungsplanung bzw. Arbeitsplanung ergänzt um die Materialflussplanung. Unserer Erfahrung nach kann der Produktentstehungsprozess nicht als stringente Folge von Phasen und Meilensteinen verstanden werden. Vielmehr handelt es sich um ein Wechselspiel von Aufgaben, die sich in vier Zyklen gliedern lassen; Bild 18 und Bild 19 sollen dies verdeutlichen.

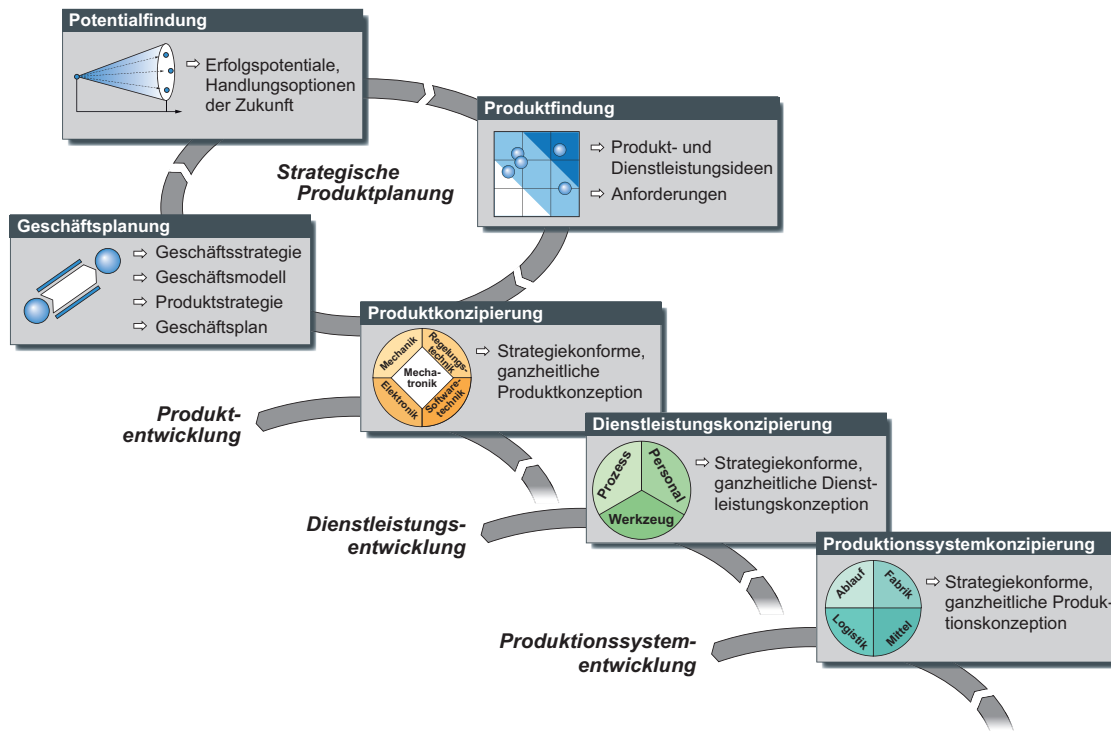


Bild 18: Referenzmodell der strategischen Planung und integrativen Entwicklung von Marktleistungen (die 3 Entwicklungszyklen sind nur angedeutet)

Erster Zyklus: Strategische Produktplanung

Dieser Zyklus charakterisiert das Vorgehen vom Finden der Erfolgspotentiale der Zukunft bis zur erfolgversprechenden Produktkonzeption – der sog. prinzipiellen Lösung. Er umfasst die Aufgabenbereiche Potentialfindung, Produktfindung, Geschäftsplanung und Produktkonzipierung. Das Ziel der Potentialfindung ist das Erkennen der Erfolgspotentiale der Zukunft sowie die Ermittlung entsprechender Handlungsoptionen. Es werden Methoden wie die Szenario-Technik, Delphi-Studien oder Trendanalysen eingesetzt. Basierend auf den erkannten Erfolgspotentialen befasst sich die Produktfindung mit der Suche und der Auswahl neuer Produkt- und Dienstleistungsideen zu deren Erschließung. In der Geschäftsplanung geht es um die Erstellung einer Geschäftsstrategie und damit verbunden um die Entwicklung eines Geschäftsmodells sowie um die Produktstrategie. Letztere enthält Aussagen zur Gestaltung des Produktprogramms, zur wirtschaftlichen Bewältigung der vom Markt geforderten Variantenvielfalt, zu einge-

setzten Technologien, zur Programmpflege über den Produktlebenszyklus etc. Die Produktstrategie mündet in einen Geschäftsplan, der den Nachweis erbringt, ob mit dem neuen Produkt bzw. mit einer neuen Produktoption ein attraktiver Return on Investment zu erzielen ist.

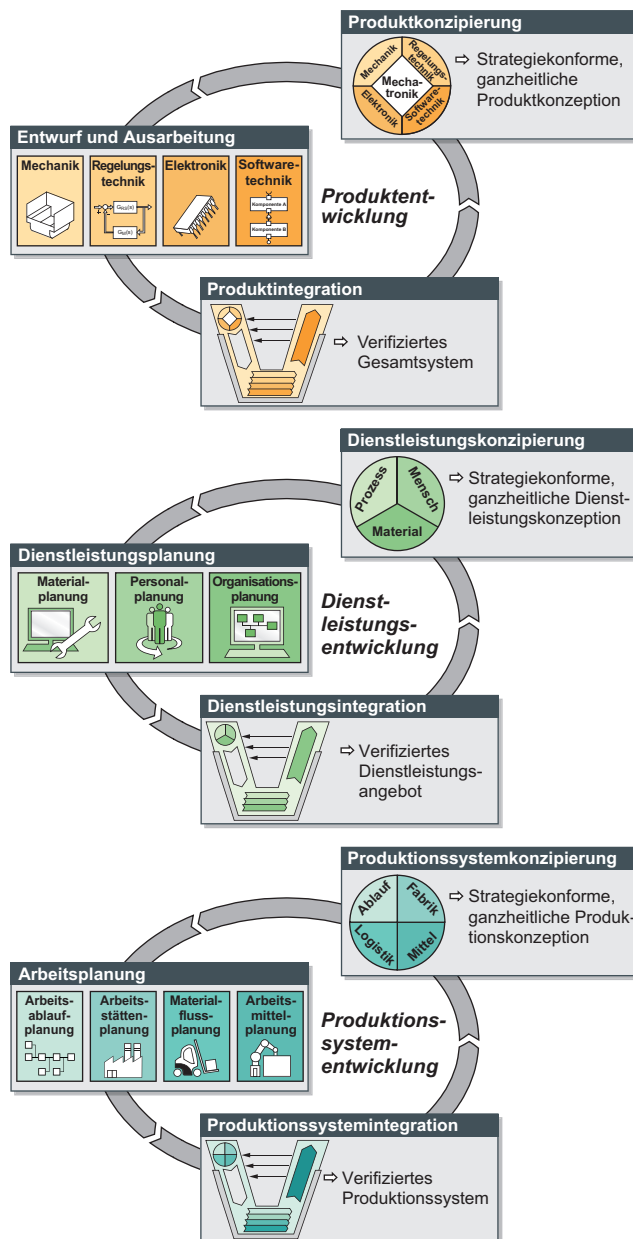


Bild 19: Die drei Entwicklungszyklen

Zweiter Zyklus: Produktentwicklung/Virtuelles Produkt

Dieser Zyklus umfasst die fachgebietsübergreifende Produktkonzipierung, den fachgebietspezifischen Entwurf und die entsprechende Ausarbeitung sowie die Integration der Ergebnisse der einzelnen Fachgebiete zu einer Gesamtlösung. Da in diesem Zusammenhang die Bildung und Analyse von rechnerinternen Modellen eine wichtige Rolle

spielt, hat sich der Begriff Virtuelles Produkt bzw. Virtual Prototyping verbreitet [SK97].

Dritter Zyklus: Dienstleistungsentwicklung

Ziel dieses Zyklus ist die Umsetzung einer Dienstleistungsidee in eine Marktleistung. Empirische Feldstudien haben gezeigt, dass auch die Dienstleistungsentwicklung nicht als stringente Folge von Phasen und Meilensteinen verstanden werden kann. Auch hier handelt es sich um ein Wechselspiel von Aufgaben, und zwar der Dienstleistungskonzipierung, der Dienstleistungsplanung und der Dienstleistungsintegration.

Vierter Zyklus: Produktionssystementwicklung/Digitale Fabrik

Den Ausgangspunkt bildet die Konzipierung des Produktionssystems. Dabei sind die vier Aspekte Arbeitsablaufplanung, Arbeitsmittelplanung, Arbeitsstättenplanung und Produktionslogistik (Schwerpunkt: Materialflussplanung) integrativ zu betrachten. Diese vier Aspekte sind im Verlauf dieses vierten Zyklus weiter zu konkretisieren. Die Begriffe Virtuelle Produktion bzw. Digitale Fabrik drücken aus, dass in diesem Zyklus ebenfalls rechnerinterne Modelle gebildet und analysiert werden – Modelle von den geplanten Produktionssystemen bzw. von Subsystemen wie Fertigungslinien und Arbeitsplätze.

Produkt-, Dienstleistungs- und Produktionssystementwicklung sind parallel und eng aufeinander abgestimmt voranzutreiben. Nur so wird sichergestellt, dass auch alle Möglichkeiten der Gestaltung einer leistungsfähigen und innovativen Marktleistung ausgeschöpft werden. In allen vier Zyklen als auch Zyklen-übergreifend sind mehrere Aspekte zu integrieren. Die etablierten, jeweils auf die spezifischen Fachgebiete fokussierten Entwicklungsmethoden, wie die Konstruktionslehre nach PAHL/BEITZ [PBF+07] im klassischen Maschinenbau und, wie die VDI-Richtlinie 2206 „Entwicklungsmethodik für mechatronische Systeme“ [VDI2206] in der Mechatronik, können dies nicht leisten. **Systems Engineering (SE)** hat das Potential, Disziplinen und vielfältige Aspekte zu integrieren, und kann somit als Grundlage für eine Weiterentwicklung der Entwicklungsmethodik dienen [HFW+12], [GDS+13]. Besondere Herausforderungen bestehen in den sogenannten frühen Phasen „Planen und Klären der Aufgabe“ und „Konzipierung“, deren wesentliches Ergebnis der fachdisziplinübergreifende ganzheitliche Systementwurf ist. Im Folgenden stellen wir die von uns entwickelte Spezifikationstechnik für die fachdisziplinübergreifende Systemkonzipierung vor.

Integrative Konzipierung von Produkt und Produktionssystem

Im Zentrum des Handlungsbedarfs auf dem Weg zu einer von Systems Engineering geprägten Entwicklungsmethodik für die hier behandelten intelligenten technischen Systeme steht eine ganzheitliche integrative Spezifikation des Systemkonzepts. Wie in Bild 20 angedeutet soll damit die Lücke zwischen dem Anforderungskatalog, der eine erste grobe Spezifikation des Gesamtsystems darstellt und naturgemäß interpretierbar ist, und den etablierten Spezifikationstechniken der einzelnen Fachgebiete geschlossen

werden. Die Systemspezifikation führt zu einem Systemmodell. Es bildet alle relevanten Aspekte ab und ermöglicht frühzeitig erste Analysen (z.B. der Zuverlässigkeit oder des dynamischen Verhaltens). Das Systemmodell bildet zudem den Ausgangspunkt für die fachdisziplinspezifische Entwurfs- und Ausarbeitungsphase. Es ist die Grundlage für die Kommunikation und Kooperation der involvierten Fachexperten über den gesamten Produktentwicklungsprozess und dient als Plattform für den Erhalt der Konsistenz aller im Zuge des Produktentwicklungsprozesses entstehenden Partialmodelle bzw. Dokumente.

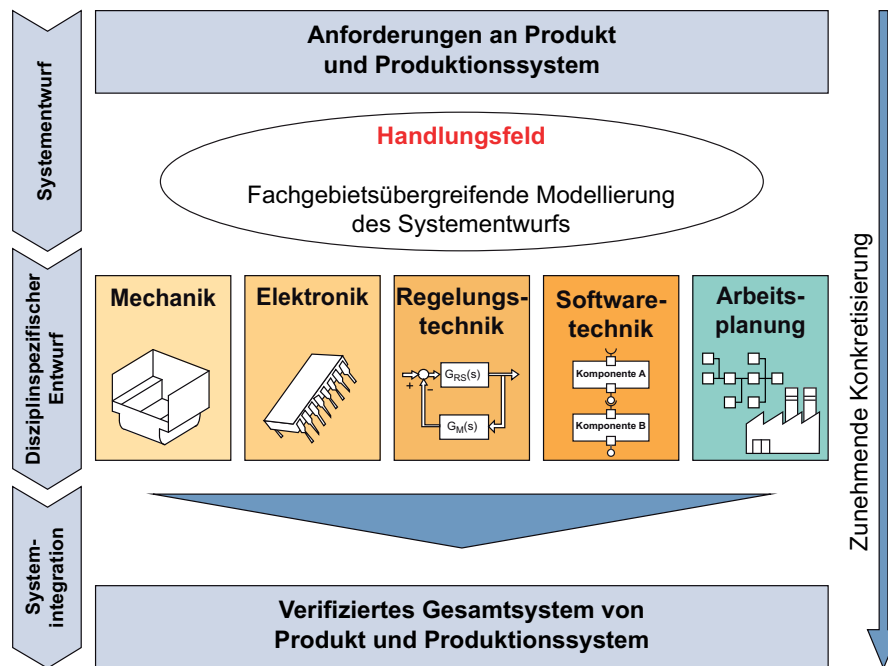


Bild 20: Zentrale Herausforderung: Eine neue Spezifikationstechnik zur frühzeitigen fachgebietsübergreifenden Modellierung des Systementwurfs eines komplexen multidisziplinären Systems

Die Spezifikationstechnik CONSENS – „CONceptual design Specification technique for the ENgineering of complex Systems“ gliedert sich in zehn Aspekte, die jeweils rechnerintern als Partialmodelle repräsentiert werden und zusammen ein System kohärenter Partialmodelle bilden (Bild 21) [GRS14], [GLL12]. Mit Hilfe dieser Aspekte werden sämtliche Gesichtspunkte des Systems berücksichtigt und allgemeinverständlich beschrieben. Die Aspekte Umfeld, Anwendungsszenarien, Anforderungen, Funktionen, Wirkstruktur, Gestalt und Verhalten beschreiben die Konzeption des Produkts; die Aspekte Prozesse, Ressourcen und Gestalt (PS) beschreiben die Konzeption des zugehörigen Produktionssystems. Die Aspekte der Spezifikationstechnik werden im Folgenden näher erläutert [GFD+09].

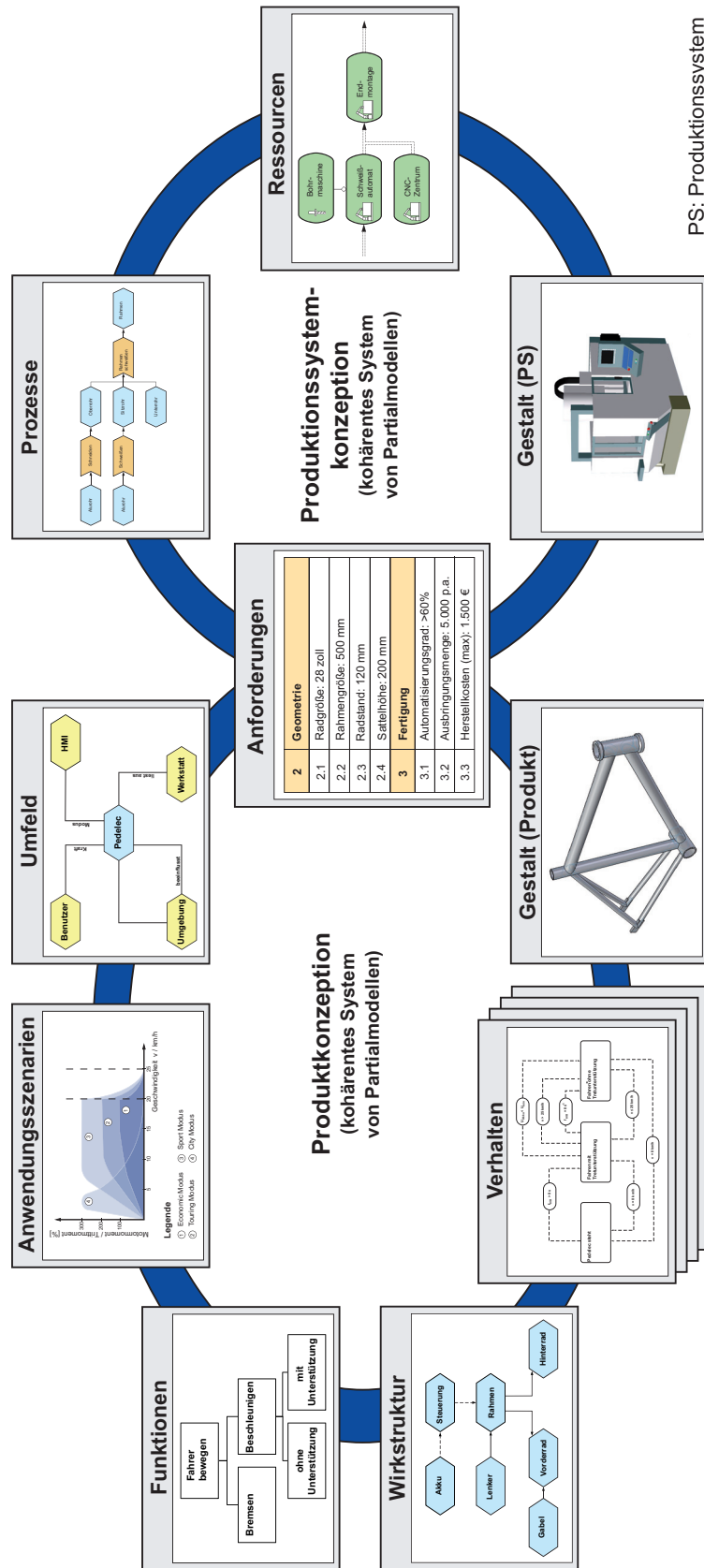


Bild 21: Aspekte zur fachgebietsübergreifenden Beschreibung der Konzeption eines multidisziplinären Produkts und des entsprechenden Produktionssystems

Umfeld: Das System wird zu Beginn als „Black Box“ in seinem Umfeld abgebildet. Alle Einflüsse, die auf das System wirken sowie Systemelemente des Umfelds, die in Wechselwirkung mit dem System stehen, werden modelliert.

Anwendungsszenarien: Sie beschreiben eine situationsspezifische Sicht auf das in der Systemkonzipierung beschriebene System und das Systemverhalten. Sie bestehen aus einem Steckbrief und verweisen auf alle für das Szenario relevanten Elemente der Systemkonzeption.

Anforderungen: Das Partialmodell umfasst eine strukturierte Sammlung aller Anforderungen an das zu entwickelnde Produkt. In den Entwicklungsphasen Konzipierung und Konkretisierung sind diese Anforderungen umzusetzen. Jede Anforderung wird textuell beschrieben. Quantifizierbare Anforderungen werden durch Attribute und deren Ausprägungen konkretisiert. Hilfestellung beim Aufstellen von Anforderungslisten geben Checklisten [PBF+07], [Rot01].

Funktionen: Dieser Aspekt beinhaltet eine hierarchische Aufgliederung der Funktionalität des Systems. Eine Funktion ist der allgemeine und gewollte Zusammenhang zwischen Eingangs- und Ausgangsgrößen mit dem Ziel, eine Aufgabe zu erfüllen. Funktionen werden durch Lösungsmuster bzw. deren Konkretisierungen realisiert. Eine Untergliederung in Subfunktionen erfolgt so lange, bis zu den Funktionen sinnvolle Lösungsmuster gefunden werden.

Wirkstruktur: In der Wirkstruktur werden die Systemelemente, deren Merkmale sowie die Beziehungen der Systemelemente zueinander beschrieben. Ziel ist die Abbildung des grundsätzlichen Aufbaus und der prinzipiellen Wirkungsweise des Systems. Systemelemente repräsentieren Systeme, Module, Bauteile oder Software-Komponenten. Stoff-, Energie- und Informationsflüsse sowie logische Beziehungen beschreiben die Wechselwirkungen zwischen den Systemelementen.

Gestalt (Produkt): Bereits in der Konzipierung sind erste Festlegungen der Gestalt des Systems vorzunehmen. Sie sind ebenfalls Teil der Prinziplösung. Der Aspekt umfasst Angaben über Anzahl, Form, Lage, Anordnung und Art der Wirkflächen und Wirkorte des Systems. Des Weiteren können Hüllflächen und Stützstrukturen beschrieben werden. Die rechnerunterstützte Modellierung erfolgt mit Hilfe gängiger 3D-CAD-Systeme.

Verhalten: Bei der Spezifikation von mechatronischen Systemen spielen die Modellierung von Aktivitäten, Zuständen und Zustandsübergängen sowie die Auswirkung auf die Wirkstruktur eine wesentliche Rolle. Diese Art der Modellierung erfolgt im Partialmodell Verhalten.

Prozesse: Die Wirkstruktur und erste Gestaltinformationen dienen dazu, das Produktkonzept in eine gestaltorientierte Struktur zu überführen. Hierdurch werden die zu fertigenden Bestandteile der Wirkstruktur identifiziert. Dies sind die gestaltbehafteten Systemelemente wie Module, Bauteile und Baugruppen. Anschließend werden im Partial-

modell Prozesse eine erste Montage- und Fertigungsreihenfolge ermittelt. Diese werden mit Hilfe von Prozessen und Materialelementen spezifiziert. Die Prozesse beschreiben den Produktionsablauf als eine Folge von Arbeitsvorgängen. Diese beinhaltet sowohl Herstellungs-, Montage- als auch Transportvorgänge.

Ressourcen: Die Prozesse beschreiben die durchzuführenden Arbeitsschritte betriebsmittelunabhängig. Die Durchführung der einzelnen Prozesse erfolgt durch Ressourcen. Als Ressourcen werden alle benötigten Sachmittel sowie das Personal bezeichnet [DIN69901]. Die Ressourcen werden den Prozessen zugeordnet, wobei es möglich ist, dass eine Ressource mehrere Prozesse ausführt. Ressourcen sind durch Materialflüsse miteinander verbunden. Die einzelnen Ressourcen werden durch Parameter und Gestaltinformationen konkretisiert.

Gestalt (Produktionssystem): Analog zur Produktentwicklung werden bereits während der Konzipierung des Produktionssystems erste Festlegungen zu seiner Gestalt getroffen. Unter Gestalt verstehen wir Arbeitsräume und Platzbedarf von Maschinen oder Wirkflächen von Handhabungseinrichtungen. Diese Informationen werden in Form von Flächenbedarfen, Skizzen oder 3D-CAD-Modellen gespeichert und den Ressourcen zugewiesen.

Virtualisierung in der Produktentwicklung

Neben der integrativen und interdisziplinären Zusammenarbeit der beteiligten Fachexperten spielt die durchgängige Virtualisierung eine wichtige Rolle im Entwicklungsprozess. Durchgängige Virtualisierung bedeutet die Bildung und Analyse von rechnerinternen Modellen verschiedener Detaillierungs- und Konkretisierungsstufen über den gesamten Prozess der Produkt- und Produktionssystementwicklung. Bild 22 verortet verschiedene Aspekte der virtuellen modellbasierten Entwicklung im Entwicklungsprozess anhand des V-Modells der VDI-Richtlinie 2206. Ausgangspunkt ist die **fachgebietsübergreifende Systemspezifikation**. Zur rechnerunterstützten Modellierung der Systemkonzeption mit CONSENS existiert das Software-Werkzeug *Mechatronic Modeller* [GLL12]. Außerdem kann eine rechnerinterne Abbildung des Systemmodells mit CONSENS durch verschiedene UML /SySML -Werkzeuge erfolgen. Hierfür wurde exemplarisch ein SysML4CONSENS-Profil für den Sparx Enterprise Architect (EA) erarbeitet, welches eine CONSENS-konforme Modellierung ermöglicht. Das methodische Vorgehen von CONSENS wurde so mit der Modellierungssprache SysML und dem Softwarewerkzeug Enterprise Architect kombiniert [IKD13].

Auf Basis der modellbasierten Systemspezifikation können erste Simulationsmodelle zur frühzeitigen Analyse auf Systemebene abgeleitet und entsprechende Simulationen durchgeführt werden; z.B. die integrierte Dynamikanalyse zur frühzeitigen Analyse des dynamischen Systemverhaltens auf Basis von Modelica-Modellen [GTS14]. Die Ergebnisse können dann wiederum in die Systemspezifikation zurückgespiegelt werden.

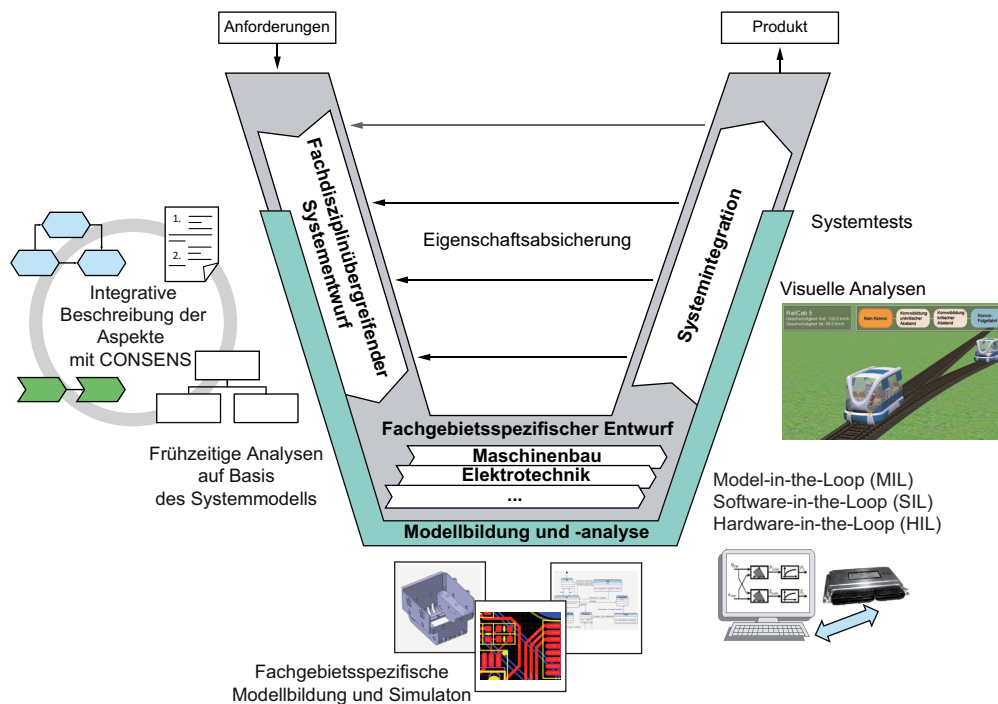


Bild 22: Das V-Modell als Basis für Modellierung und Analyse

Ausgehend von der fachgebietsübergreifenden Systemspezifikation erfolgt die fachgebietspezifische Konkretisierung des Systems. Dafür werden die etablierten Werkzeuge wie 3D-CAD und E-CAD-Systeme eingesetzt.

Visuelle Analysen: Im Zuge der Konkretisierung kann die Technologie Virtual Reality (VR) zur Analyse und Bewertung virtueller Prototypen eingesetzt werden. Virtual Reality trägt dazu bei, virtuelle Prototypen erlebbar zu machen. Dies gilt insbesondere für die Gestalt und das Verhalten, wie die Kinematik eines Systems. Weitere Visualisierungstechniken ermöglichen es, VR-Modelle mit Simulationsdaten zu überlagern, so dass das visuell wahrgenommene Verhalten des Systems leicht nachvollziehbar wird [Was13].

Des Weiteren gibt es X-in-the-Loop Ansätze zum rechnerunterstützten Testen komplexer mechatronischer Systeme. Zur frühzeitigen Systemverifikation und -test der Softwareentwicklung können beispielsweise Model-in-the-Loop (MiL) und Software-in-the-Loop (SiL) Simulationen eingesetzt werden. Bei MiL Simulationen erfolgt eine Simulation des Gesamtsystems ausschließlich auf Basis von Modellen. Bei SiL Simulationen werden realisierte Softwarekomponenten (Regler ist im C-Code implementiert) in simulierten Umgebungen ausgeführt. Auf diese Weise können ihre Eigenschaften rein virtuell getestet werden. Zum Test weiterer Funktionen oder Hardwarekomponenten können mit Hilfe von Prüfständen auch Hardware-in-the-Loop (HiL) Simulationen durchgeführt werden, ohne einen vollständigen realen Prototyp realisieren zu müssen [GTS14].

Im Folgenden geben wir zwei Beispiele für das Leistungsvermögen der Virtualisierung der Produktentstehung. Es handelt sich zum einen um einen rekonfigurierbaren Fahrsi-

mulator, der es ermöglicht, Test- und Trainingsumgebungen für fortgeschrittene Fahrerassistenzsysteme rationell zu erzeugen, und zum anderen um die virtuelle Inbetriebnahme von Produktionssystemen.

Entwicklung eines rekonfigurierbaren Fahrsimulators

Fahrsimulatoren werden seit Jahrzehnten erfolgreich in verschiedenen Anwendungsbereichen eingesetzt. Sie unterscheiden sich weitgehend in ihrer Struktur, Genauigkeit, Komplexität und in ihren Kosten. In der Regel werden Fahrsimulatoren individuell für eine spezielle Aufgabe entwickelt und haben typischerweise eine festgelegte Struktur. In speziellen Anwendungsbereichen werden jedoch unterschiedliche Varianten eines Fahrsimulators benötigt. Es besteht daher Handlungsbedarf für die Entwicklung eines rekonfigurierbaren Fahrsimulators, der es dem Betreiber ermöglicht, ohne umfassende Fachkenntnisse einfach und schnell anwendungsspezifische Varianten eines Fahrsimulators zu erstellen.

Die Entwicklung von rekonfigurierbaren mechatronischen Systemen, die überwiegend aus modularen Komponenten bestehen, beruht auf dem Prinzip des morphologischen Kastens. Zur Rekonfiguration eines bestehenden Fahrsimulators, entsteht die Notwendigkeit, eine Phase zwischen der Modellierungs- und Simulationsphase einzufügen (Bild 23). In dieser Phase kann der Fahrsimulator-Betreiber die gewünschten Komponenten in Form von Hardware-Komponenten und Softwaremodulen auswählen, um eine anwendungsspezifische Variante des Fahrsimulators zu konfigurieren. Beispiele für Hardware-Komponenten sind die Bewegungsplattform, das Visualisierungssystem oder Eingabegeräte. Beispiele für Softwaremodule sind das Fahrzeugmodell oder weitere Verkehrsmodelle. Die Hardware-Komponenten und Softwaremodule liegen in unterschiedlichen Ausprägungen in Form von Lösungselementen vor. Unterschiedliche Ausprägungen des Visualisierungssystems sind z.B. die Lösungselemente Rundprojektion, stereoskopischer Bildschirm oder ein tragbares Sichtsystem (engl. head mounted display). Beispiele für verschiedene Ausprägungen des Softwaremoduls zum Fahrzeugmodell sind unterschiedliche Modellierungstiefen der Lösungselemente, wie z.B. ein komplexes echtzeitfähiges Modell zur Fahrdynamik oder ein reduziertes 1-Spurmodell. Verschiedene Kombinationen der Lösungselemente führen zu den entsprechenden Fahrsimulatorvarianten [HBA+13]. Die Herausforderung dabei ist, dass die ausgewählten Lösungselemente, also die jeweiligen Ausprägungen der Hardware-Komponenten und der Softwaremodule, miteinander kombinierbar sein müssen. Dies erfordert kompatible Hard- und Softwareschnittstellen und einen zuverlässigen Kontrollmechanismus, um konsistente Konfigurationen des Systems zu erreichen. Hierfür wurde ein Konzept entwickelt, das die einzelnen Komponenten in eine gültige und funktionierende Simulator-konfiguration überführt [Has14].

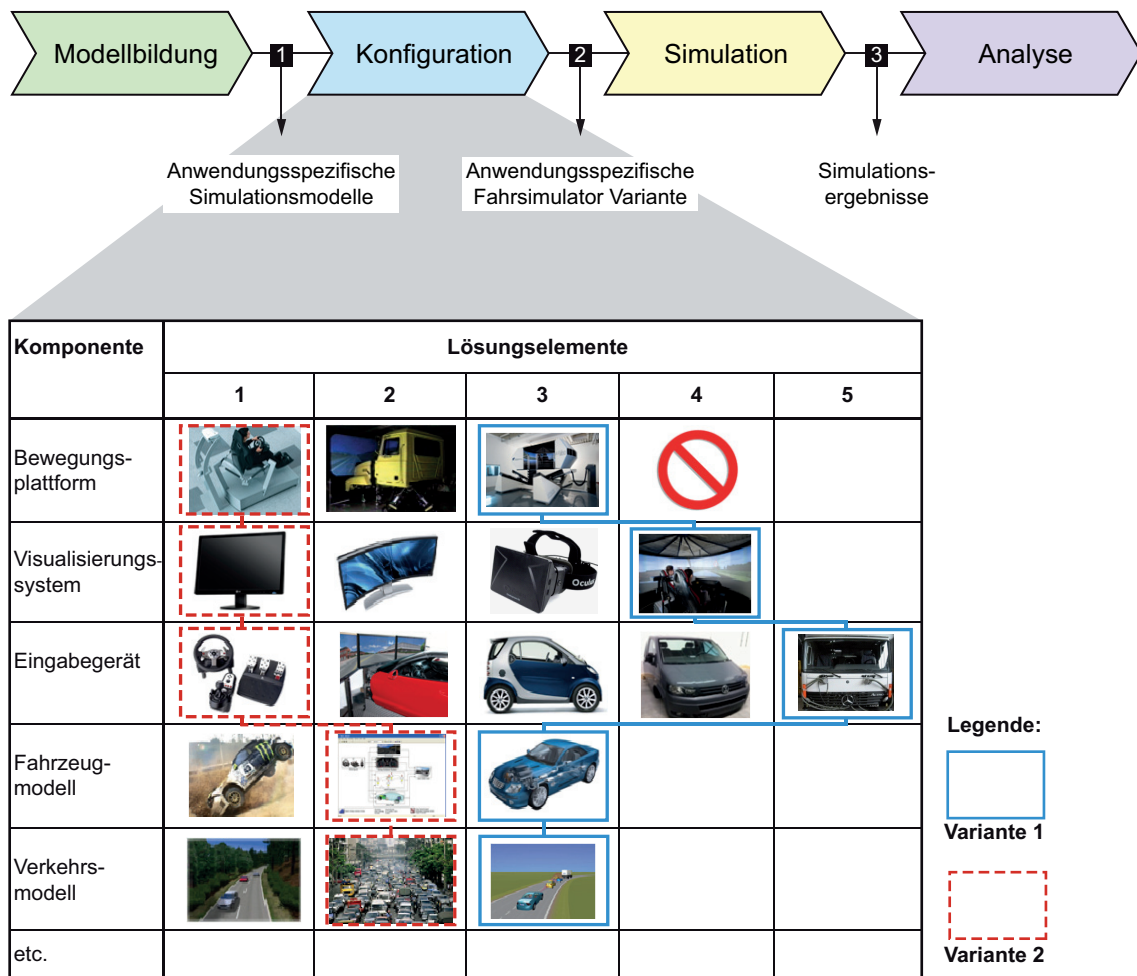


Bild 23: Morphologischer Kasten als Basis für das Konfigurieren von Fahrsimulatoren

Das Konzept wurde im Rahmen des Forschungsprojektes TRAFFIS erfolgreich angewendet. Dazu wurden drei Simulatorvarianten definiert:

1) TRAFFIS Komplettversion: Diese Fahrsimulatorvariante besitzt die komplexeste Struktur mit einer voll steuerbaren Bewegungsplattform, einer Rundprojektion und einer realen Fahrerkabine. Ziel ist die detaillierte Darstellung der Fahrdynamik durch die voll steuerbare Bewegungsplattform, die vollständige Darstellung der Fahrumgebung auf der Rundprojektion sowie die realistische Bedienung in der Fahrerkabine (Bild 24). Diese Fahrsimulatorvariante bietet damit ideale Voraussetzungen für den HIL-Test von Fahrerassistenzsystemen (FAS) in einer simulierten Fahrzeugumgebung.

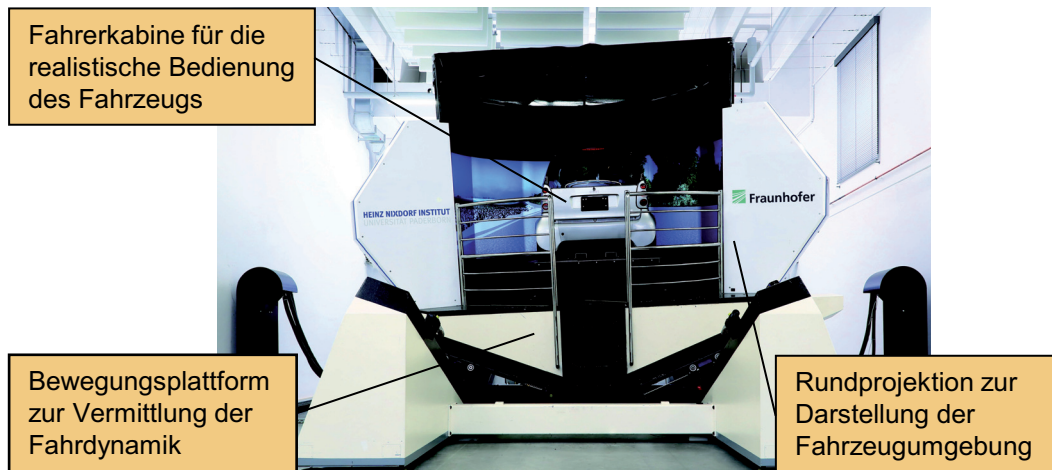


Bild 24: TRAFFIS Komplettversion als mögliche Varianten eines Fahrsimulators mit der höchsten Modellierungstiefe und komplexesten Ausprägung der Lösungselemente nach [HBA+13].

2) TRAFFIS portable Version: Diese Fahrsimulatorvariante umfasst eine einfache Bewegungsplattform mit eingeschränkten Freiheitsgraden zur prinzipiellen Darstellung der Fahrdynamik, einen stereoskopischen Bildschirm zur Darstellung der Fahrersicht nach vorn und einem einfachen Force-Feedback Lenkrad zur rudimentären Bedienung des Fahrzeugs. Ein mögliches Anwendungsgebiet dieser Fahrsimulatorvariante ist die Schulung von Kraftfahrern für ausgewählte Fahrerassistenzsysteme (Bild 25).

3) TRAFFIS Light Version: Diese Fahrsimulatorvariante stellt eine minimale Basisversion dar. Ein tragbares Sichtsystem (engl. Head mounted display) unterstützt eine einfache Darstellung der Fahrzeugumgebung. Die Fahrzeugbedienung erfolgt über ein einfaches Lenkrad. Ein Bewegungssystem zur Darstellung der Fahrdynamik fehlt hier vollständig. Ein mögliches Anwendungsgebiet dieser Fahrsimulatorvariante ist der Test der FAS-Software nach dem Software-in-the-Loop (SiL)-Prinzip. Diese Simulatorvariante dient FAS-Entwicklern, in frühen Entwicklungsphasen, zum täglichen Testen der Kontrollalgorithmen (siehe Bild 26).



Bild 25: TRAFFIS portable Version mit reduzierter Modellierungstiefe und eingeschränkter Ausprägung der Lösungselemente nach [HBA+13].

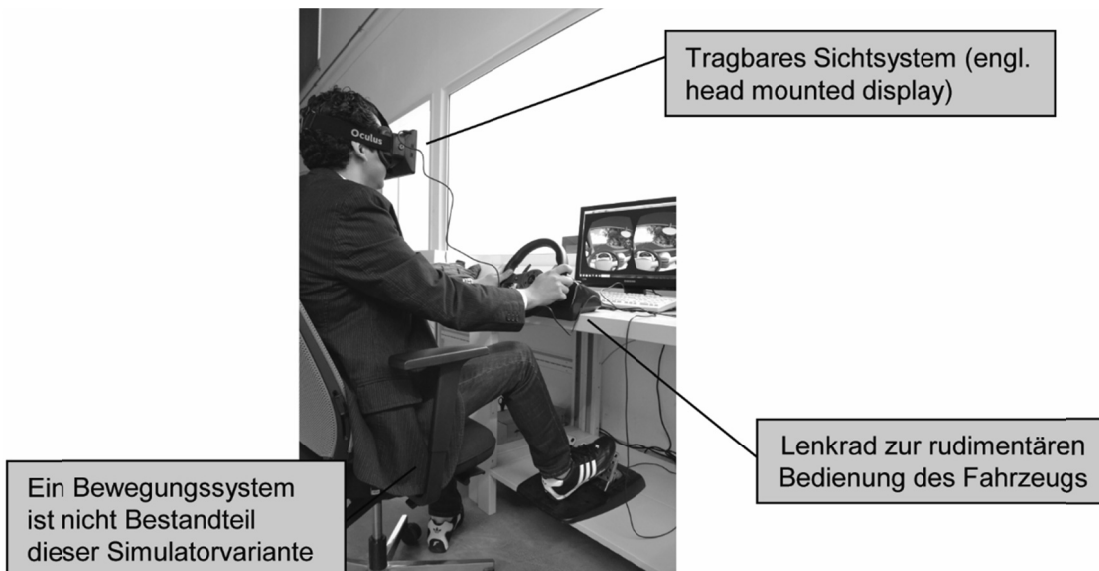


Bild 26: TRAFFIS Light Version mit geringer Modellierungstiefe und minimaler Ausprägung der Lösungselemente nach [HBA+13].

Virtuelle Inbetriebnahme

Die virtuelle Inbetriebnahme (VIBN) ist ein Lösungsansatz, um zeitraubende und kostenintensive Iterationsschleifen bei der Planung und Inbetriebnahme maschinenbaulicher Anlagen zu vermeiden. Bei der virtuellen Inbetriebnahme wird das Verhalten einer Anlage in einem Verhaltensmodell abgebildet. Die Steuerungsprogramme können dadurch einem frühzeitigen Test am Modell unterzogen werden [Wün08], [Sau11]. Die Vorwegnahme des Programmtests führt zu einer deutlichen Verkürzung der eigentlichen Inbetriebnahmezeit. Gleichzeitig steigt durch die systematischen Tests die Qualität der Steuerungsprogramme, da Fehler frühzeitig identifiziert und behoben werden können.

nen [Zäh06]. Zudem kann das Maschinenverhalten gezielt auf solche Störungen getestet werden, die bei der realen Anlage eine Gefahr für Mensch und Umwelt darstellen. Der Nachteil der virtuellen Inbetriebnahme liegt in der aufwändigen Modellerstellung. Häufig kompensiert diese den Zeitvorteil bei der Inbetriebnahme, und der kumulierte Zeitaufwand bis zum Produktionsanlauf der Anlage bleibt gleich [Wün08].

Im Rahmen eines von der Deutschen Forschungsgemeinschaft (DFG) geförderten Projekts wurde am Heinz Nixdorf Institut eine neue Systematik zur virtuellen Inbetriebnahme von maschinenbaulichen Anlagen auf Basis von objektorientierten Verhaltensmodellen erarbeitet. Diese beruht auf der Strukturierung einer maschinenbaulichen Anlage in Module, für die Verhaltensmodelle mit wählbaren Modellierungstiefen und adaptiver Umschaltung zwischen diesen Konkretisierungsstufen erstellt werden. Ziel der Systematik ist die Reduzierung des Modellierungsaufwands.

Der Lebenszyklus von maschinenbaulichen Anlagen gliedert sich in die sechs Hauptphasen: Anlagenkonzipierung, Anlagenausarbeitung, Anlagenherstellung, Anlageninbetriebnahme, Anlagenbetrieb und Redistribution. Die virtuelle Inbetriebnahme bildet eine weitere, parallele Phase in diesem Prozess (Bild 27). Sie gliedert sich in die fünf Teilphasen: Testspezifikationen definieren, Anlagenmodule analysieren, Umschaltbedingungen festlegen, Verhaltensmodelle erstellen und VIBN durchführen [SLB13]. Zwischen dem Prozess der Anlagenentstehung und der virtuellen Inbetriebnahme erfolgt eine durchgehende Synchronisation. Ausgetauscht werden z.B. die Funktionshierarchie und Wirkstruktur als Eingangsinformationen zur Definition der Testspezifikationen, wohingegen Anforderungen aus den Testspezifikationen abgeleitet und synchronisiert werden.

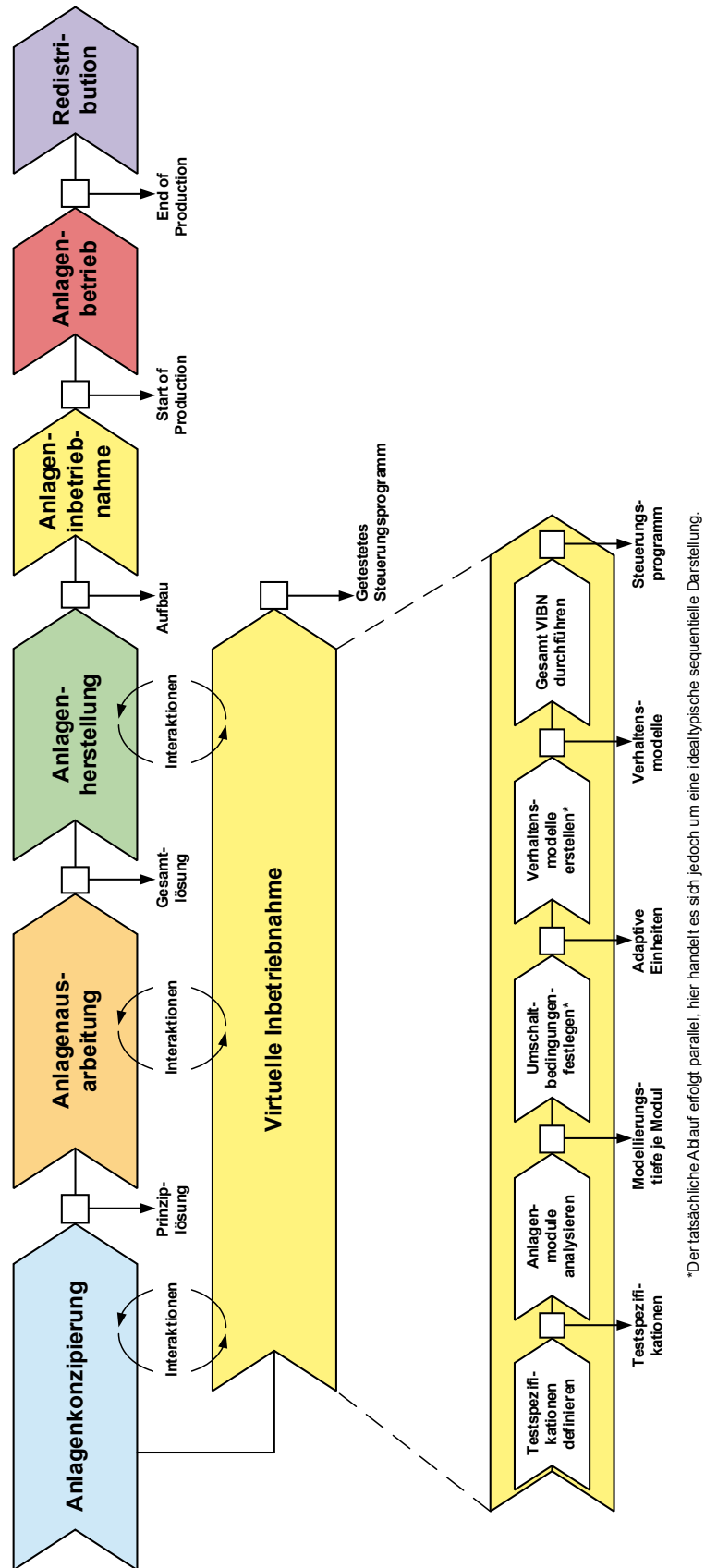


Bild 27: Anlagenentstehungsprozess mit der virtuellen Inbetriebnahme

Die Testspezifikationen bestehen aus Anwendungsszenarien sowie Störszenarien. Anwendungsszenarien umfassen sowohl das Betriebsverhalten der Anlage/Anlagenmodule als auch gewünschte Sonderbetriebsarten (z.B. Wartung oder Handbetrieb). Störszenarien hingegen spezifizieren das Verhalten der Anlagen im Störfall und stellen in diesem Fall den gewünschten Zustand sicher. Nach der Definition der Testspezifikationen erfolgt die Analyse der Anlagenmodule. In dieser Phase wird den Modulen jeweils die notwendige Modellierungstiefe zugewiesen. Es sind drei allgemeine Modellierungstiefen für die virtuelle Inbetriebnahme definiert worden: *idealisierte Funktion*, *prinzipielle Machbarkeit* und *system-spezifisches Verhalten* [SLB13]. Grundidee der adaptiven Umschaltung ist, ein Modell mit niedriger Modellierungstiefe gegen das äquivalente Modell mit höherer Modellierungstiefe auszutauschen, sofern das aktuelle Modell nicht die geforderten Informationen liefert. Für den Fall, dass mehr Informationen vorliegen als benötigt, wird von einem Modell mit höherer Detaillierung zu einem mit geringer Modellierungstiefe umgeschaltet. Durch die adaptive Umschaltung kann bspw. zur Laufzeit auf geänderte Randbedingungen reagiert werden und lediglich die minimal erforderliche Modellierungstiefe verwendet werden. Weiterhin kann die Steuerung einer Anlage auf ihr korrektes Verhalten insbesondere bei unvorhergesehenen Ereignissen und Sonderbetriebsarten wie z.B. Störungen oder Notaus untersucht werden. Die Schritte der Systematik von der Definition der Testspezifikationen bis zur Erstellung der Verhaltensmodelle wurden am Beispiel eines flexibel automatisierten Fertigungssystems erfolgreich validiert (Bild 28).

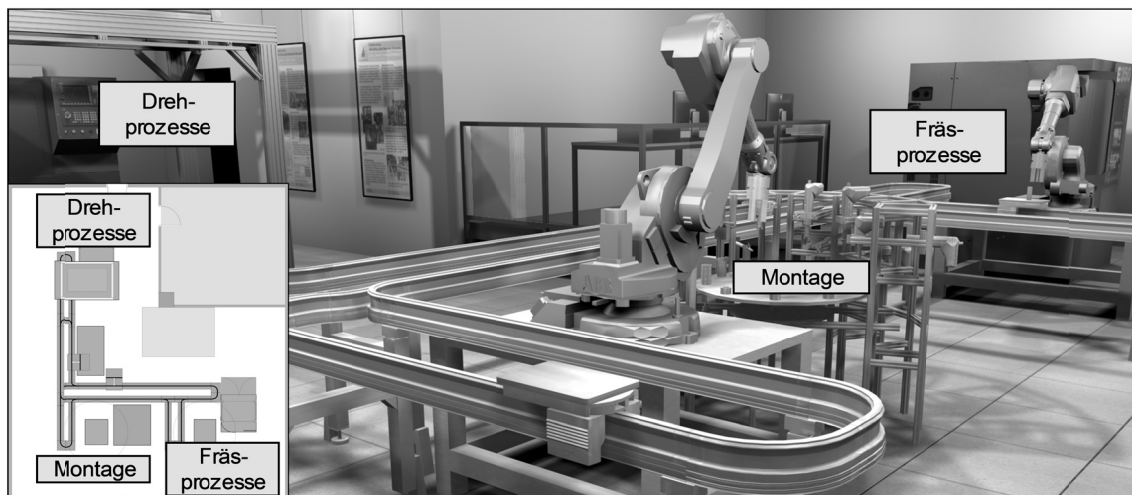


Bild 28: Überblick inkl. Layout des Labors zur flexiblen Industrieautomatisierung

Literatur

- [All05] ALLWEYER, T.: Geschäftsprozessmanagement – Strategie, Entwurf, Implementierung, Controlling. W3L-Verlag, Herdecke, 2005
- [ASSW14] ARBEITSKREIS SMART SERVICE WELT: Smart Service Welt – Umsetzungsempfehlungen für das Zukunftsprojekt Internet-basierte Dienste für die Wirtschaft, 2014
- [BR11] BIEGER, T.; REINHOLD, S.: Das wertbasierte Geschäftsmodell – Ein aktualisierter Strukturierungsansatz. In Bieger, T.; zu Knyphausen-Aufseß, D.; Krys, C. (Hrsg.): Innovative Geschäftsmodelle. Springer Verlag, Berlin, 2011
- [DIW14-ol] DEUTSCHES INSTITUT FÜR WIRTSCHAFTSFORSCHUNG (Hrsg.): DIW Glossar – Innovationssystem. Unter: http://www.diw.de/de/diw_01.c.439330.de/presse/diw_glossar/inovations-system.html, am 30. Oktober 2014
- [FA13] FORSCHUNGSUNION WIRTSCHAFT-WISSENSCHAFT; ACATECH – DEUTSCHE AKADEMIE DER TECHNIKWISSENSCHAFTEN (Hrsg.): Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0, 2013
- [GA14] GAUSEMEIER, J.; AMSHOFF, B.: Diskursive Geschäftsmodellentwicklung – Erfolgreiche Positionierung in der Wettbewerbsarena durch integrative Entwicklung von Marktleistung und Geschäftsmodell. In: ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb, 109(6), 2014, S. 428-434
- [GAD+14] GAUSEMEIER, J.; AMSHOFF, B.; DÜLME, C.; KAGE, M.: Strategische Planung von Marktleistungen im Kontext Industrie 4.0. In: Gausemeier, J. (Hrsg.): Vorausschau und Technologieplanung – 10. Symposium für Vorausschau und Technologieplanung. HNI-Verlagsschriftenreihe, Band 334, Paderborn, 2014
- [GDS+13] GAUSEMEIER, J.; DUMITRESCU, R.; STEFFEN, D.; CZAJA, A.; WIEDERKEHR, O.; TSCHIRNER, C.: Systems Engineering in der industriellen Praxis. Heinz Nixdorf Institut; Fraunhofer-Institut für Produktionstechnologie IPT, Projektgruppe Entwurfstechnik Mechatronik; UNITY AG, Paderborn, 2013
- [GFD+08] GAUSEMEIER, J.; FRANK, U.; DONOTH, J.; KAHL, S.: Spezifikationstechnik zur Beschreibung der Prinziplösung selbstoptimierender Systeme des Maschinenbaus. In Konstruktion: Teil 1 in 7/8 – 2008, Teil 2 in 9 – 2008, VDI Verlag, Berlin, 2008
- [GLL12] GAUSEMEIER, J.; LANZA, G.; LINDEMANN, U.: Produkte und Produktionssysteme integrativ konzipieren – Modellbildung und Analyse in der frühen Phase der Produktentstehung. Carl Hanser Verlag, München, 2012
- [GP14] GAUSEMEIER, J.; PLASS, C.: Zukunftsorientierte Unternehmensgestaltung – Strategien, Geschäftsprozesse und IT-Systeme für die Produktion von morgen. Carl Hanser Verlag, München, Wien, 2014
- [GRK13] GAUSEMEIER, J.; RÜBBELKE, R.; KÖSTER, O.: Systematik zur Entwicklung von Geschäftsmodellen in der Produktentstehung. In: Gausemeier, J. (Hrsg.): Vorausschau und Technologieplanung – 9. Symposium für Vorausschau und Technologieplanung. HNI-Verlagsschriftenreihe, Band 318, Paderborn, 2013
- [GRS14] GAUSEMEIER, J.; RAMMIG, F.-J.; SCHÄFER, W. (Eds.): Design Methodology for Intelligent Technical Systems – Develop Intelligent Technical Systems of the Future. Springer-Verlag, Berlin Heidelberg, 2014
- [GTS14] GAUSEMEIER, J.; TRÄCHTLER, A.; SCHÄFER, W.: Semantische Technologien im Entwurf mechatronischer Systeme: Effektiver Austausch von Lösungswissen in Branchenwerteschöpfungsketten. Carl Hanser Verlag, München, 2014
- [Has14] HASSAN, B.: A Design Framework for Developing a Reconfigurable Driving Simulator. Dissertation, Fakultät Maschinenbau. Universität Paderborn, HNI-Verlagsschriftenreihe, Band 333, Paderborn, 2014

- [HBA+13] HASSAN, B.; BERSSENBRÜGGE, J.; ALQAISI, I.; STÖCKLEIN, J.: Reconfigurable Driving Simulator for Testing and Training of Advanced Driver Assistance Systems. In: Proceedings of 2013 IEEE International Symposium on Assembly and Task Planning (ISAM), July 30 - August 2 2013, Xi'an, China, 2013
- [HFW+12] HABERFELLNER, R.; FRICKE, E.; WECK, O. DE; VÖSSNER, S.: Systems Engineering – Grundlagen und Anwendung. Orell Füssli Verlag, Zürich, 12. Auflage, 2012
- [Hir14] HIRSCH-KREINSEN, H.: Eine rein technologische Sicht führt in die Sackgasse. In: VDI Nachrichten, Heft Nr. 5, Düsseldorf, 31. Januar 2014
- [IKD13] IWANEK, P.; KAISER, L.; DUMITRESCU, R.; NYBEN, A.: Fachdisziplinübergreifende Systemmodellierung mechatronischer Systeme mit SysML und CONSENS. Maurer, M.; Schulze, S. (Hrsg.) Tag des Systems Engineerings, Carl Hanser Verlag, 2013.
- [Kös14] KÖSTER, O.: Systematik zur Entwicklung von Geschäftsmodellen in der Produktentstehung. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 326, Paderborn, 2014
- [PBF+07] PAHL, G.; BEITZ, W.; FELDHUSEN, J.; GROTE, K.-H.: Konstruktionslehre – Grundlagen erfolgreicher Produktentwicklung – Methoden und Anwendung. Springer Verlag, Berlin, 7. Auflage, 2007
- [PG05] PATELI, A.; GIAGLIS, G.: Technology Innovation-induced Business Model Change – A Contingency Approach. Journal of Organizational Change Management 18 (2005)
- [Rot01] ROTH, K.: Konstruieren mit Konstruktionskatalogen: Band 2 – Kataloge. Springer Verlag, Berlin, 3. Auflage, 2001
- [Sau11] SAUER, O.: Informationstechnik in der Fabrik der Zukunft: Fabrik 4.0 – Aktuelle Rahmenbedingungen, Stand der Technik und Forschungsbedarf. ZWF, Ausgabe 12/2011, S. 955-962
- [Sch13] SCHALLMO, D.: Geschäftsmodell-Innovation – Grundlagen, bestehende Ansätze, methodisches Vorgehen und B2B-Geschäftsmodelle. Springer Verlag, Ulm, 2013
- [SK97] SPUR, G.; KRAUSE, F.-L.: Das virtuelle Produkt. Carl Hanser Verlag, München, Wien, 1997
- [SLB13] SCHMÜDDERRICH, T.; TRÄCHTLER, A.; BRÖKELMANN, J.; GAUSEMEIER, J.: Procedural Model for the Virtual Commissioning on the Basis of Model-based Design. 23rd CIRP Design Conference, 11.-13. März 2013, Ruhr-Universität Bochum, Bochum, 2013
- [Str98] STRUBE, G.: Modellierung Motivation and Action Control in Cognitive Systems. In: Schmid, U.; Krems, J. F.; Wysocki, F. (Eds.). Mind Modelling. Pabst, Berlin, 1998
- [TS13] TÜCKS, G.; STOFFEL, M.: Die Klaviatur der Produktion – Von der Strategie bis zur Umsetzung. In: Complexity Management, Nr. 02/2013, S. 4-9
- [VDI2206] VEREIN DEUTSCHER INGENIEURE (VDI): Entwicklungsmethodik für mechatronische Systeme. VDI-Richtlinie 2206, Beuth-Verlag, Berlin, 2004
- [Was13] WASSMANN, H.: Systematik zur Entwicklung von Visualisierungstechniken für die visuelle Analyse fortgeschrittener mechatronischer Systeme in VR-Anwendungen. Dissertation, Fakultät Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 309, Paderborn 2013
- [Wün08] WÜNSCH, G.: Methoden für die virtuelle Inbetriebnahme automatisierter Produktionssysteme. Herbert Utz, München, 2008
- [Zäh06] ZÄH, M.; WÜNSCH, G.; HENSEL, T.; LINDWORSKY, A.: Feldstudie – Virtuelle Inbetriebnahme. Wt Werkstatttechnik online, 96 Jg., Heft 10, 2006

Fotolia: Beboy, bluedesign, N. Chan, Cybrain, djama, S. Duda, fotodesign-jegg.de, fovito, kanvag, mojolo, M. Nivelet, ra2 studio, Reimer, N. Sorokin, stockWERK, TechnikNeuheiten, D. Titov, WavebreakmediaMicro

Autoren

Prof. Dr.-Ing. Jürgen Gausemeier ist Seniorprofessor am Heinz Nixdorf Institut der Universität Paderborn. Seine Arbeitsschwerpunkte sind Strategische Produktplanung und Systems Engineering. Er war Sprecher des Sonderforschungsbereiches 614 „Selbstoptimierende Systeme des Maschinenbaus“ und von 2009 bis 2015 Mitglied des Wissenschaftsrats. Jürgen Gausemeier ist Initiator und Aufsichtsratsvorsitzender des Beratungsunternehmens UNITY AG. Seit 2003 ist er Mitglied von acatech – Deutsche Akademie der Technikwissenschaften und seit 2012 Vizepräsident. Ferner ist Jürgen Gausemeier Vorsitzender des Clusterboards des BMBF-Spitzenclusters „Intelligente Technische Systeme Ostwestfalen-Lippe (it's OWL)“.

Dipl.-Wirt.-Ing. Anja Czaja ist wissenschaftliche Mitarbeiterin bei Prof. Gausemeier am Heinz Nixdorf Institut der Universität Paderborn. Im Rahmen ihrer Tätigkeit arbeitet sie an Methoden und Werkzeugen für den Entwurf intelligenter mechatronischer Systeme.

M.Sc. Christian Dülme studierte im Rahmen eines dualen Studiums Wirtschaftsingenieurwesen mit der Fachrichtung Maschinenbau an der Universität Paderborn. Seit 2013 ist er wissenschaftlicher Mitarbeiter am Heinz Nixdorf Institut bei Prof. Gausemeier im Team Strategische Planung und Innovationsmanagement. Seine Tätigkeitsschwerpunkte liegen in der Potentialfindung und in der Geschäftsplanung.

Model-Based Systems Engineering in der Praxis

Model Based Systems Engineering: Einführung und Anwendungserfahrung der modellbasierten Entwicklung im Maschinenbau

Dr. Sven Kleiner

em engineering methods AG

Rheinstr. 97, 64295 Darmstadt

Tel. +49 (0) 6151 / 95 05 424, Fax. +49 (0) 6151 / 95 05 421

E-Mail: sven.kleiner@em.ag

Zusammenfassung

Die Einführung von Systems Engineering (SE) für die mechatronische Produktentwicklung gewinnt derzeit auch in den klassischen Maschinenbauunternehmen an Bedeutung und wird diskutiert. Unternehmen stellen sich dabei die Frage, was Ihnen der Einsatz von SE bezogen auf ihre konventionellen und etablierten Entwicklungsprozesse an Mehrwert bietet und welcher Nutzen hinsichtlich Verkürzung der Entwicklungszeit, Kosteneinsparung sowie Steigerung von Produktivität, Qualität und Innovation daraus entsteht. Zu Recht muss auch die Frage gestellt werden, wie sich die Investitionen in Systems Engineering Prozesse, Methoden und IT-Werkzeuge für ein Unternehmen rechnen und welche konkreten und messbaren Verbesserungen damit für die Produktentwicklung und Produktion erzielt werden können.

Im vorliegenden Beitrag werden Herausforderungen und Nutzen beim Einsatz von (Model Based) Systems Engineering, kurz (MB)SE dargestellt, die sich auf Analysen und Erfahrungen aus der Praxis stützen. Die gezielte Einführung von „Model Based Systems Engineering“ (MBSE) kann dabei helfen von der dokumentenzentrierten Vorgehensweise zu einer modellbasierten Entwicklungsmethodik zu gelangen, um die gewünschten Nutzenpotentiale auszuschöpfen. Der vorliegende Beitrag zeigt die Erfahrung aus der Einführung und Erstanwendung von MBSE am Beispiel der Maschinenentwicklung auf.

Schlüsselworte

(Model Based) Systems Engineering, Einführung, Anwendungserfahrung, Maschinenbau, Modellierung und Auslegung mechatronischer Systeme, interdisziplinäre Zusammenarbeit

Model Based Systems Engineering: Introduction, Deployment and Practical Experiences from Industry

Abstract

The introduction of Systems Engineering (SE) for the mechatronic product development is important even in classic engineering companies and shall be discussed. Therefore, companies are asking the question, what added value is offered by the integration of SE with respect to their conventional and established development process as well as what benefits will arise with respect to a shortening of the development time, cost savings such as an increase in productivity, quality and innovation. Rightly so, the question must also be asked, how high the investments in Systems Engineering processes, methods and IT-Tools are to be expected for a company. In the following contribution the challenges and benefits of SE will be presented. The deliberate introduction of “Model Based Systems Engineering” (MBSE) can aid in the conversion from a document-centered approach to a model-based development methodology in order to exploit the desired potential benefits. This paper presents the experience of the introduction and adaptation of MBSE based on the example of machine development.

Keywords

(Model Based) Systems Engineering, Rollout, Deployment, Practical Experience, Industry, Modeling and Design of Mechatronic Products, Interdisciplinary Collaboration

1 Ausgangssituation

Die Gesellschaft für Systems Engineering (GfSE e.V.) beschreibt Systems Engineering (SE) als eine umfassende Ingenieur Tätigkeit, die zur Entwicklung komplexer Produkte notwendig ist [GfSE14]. Die Aufgaben im SE sind vielfältig und beginnen bei der Anforderungsermittlung und Systemanalyse und reichen über die Systementwicklung bis hin zur Absicherung und zum Test. Aufgrund der Vielzahl an beteiligten Disziplinen, wie Mechanik, Elektrik/Elektronik und Software in heutigen innovativen Produktentwicklungsprozessen, stehen Unternehmen von verschiedenen Herausforderungen und die Beherrschung des (MB)SE erscheint zunehmend zu einem echten Wettbewerbsvorteil für Unternehmen zu werden. Beispielsweise sollen an dieser Stelle folgende Herausforderungen erwähnt werden, die im Maschinenbau mehrfach identifiziert wurden und durch den Einsatz von MBSE begegnet werden können.

- Projektmanagement: Projektmanagement ist noch nicht vollkommen umgesetzt, Projekte laufen unterschiedlich ab, Prozesse sind nicht standardisiert, fehlender interdisziplinärer Produktentwicklungsprozess
- Entwicklung Mechanik: Potential „Mechanik“ ausgeschöpft – „Mechatronik“ ist noch nicht umgesetzt, Mechanische Konstruktion/Entwicklung bieten nur Mechanik Lösung, Elektronische Bauteile werden durch Mechanik ausgewählt
- Elektrotechnik: unzureichender Informationsfluss zwischen Elektrotechnik und Mechanik führt zu aufwändigen Abstimmungen und Suche, Elektrotechnische Bauteile werden oftmals zugekauft, Kernkompetenzen sind am Markt verfügbar
- Softwareentwicklung: fehlende Integration der Softwareentwicklung in „mechatronische“ Systementwicklung, unzureichende Prozess-/ IT-Systemunterstützung

Die gezielte Einführung von (MB)SE kann dabei helfen von der dokumentenzentrierten Vorgehensweise zu einer modellbasierten Entwicklungsmethodik zu gelangen, um die gewünschten Nutzenpotentiale z.B. wie verbesserte Kommunikation zwischen Systemingenieuren und den Vertretern aus den Fachdisziplinen, verbindliche Anforderungen durch gezieltes Requirements Engineering und Management, oder Virtualisierung der Produktabsicherung auszuschöpfen und Föderation der Disziplinen, Prozesse, Methoden und Tools abzusichern (siehe Bild 1).

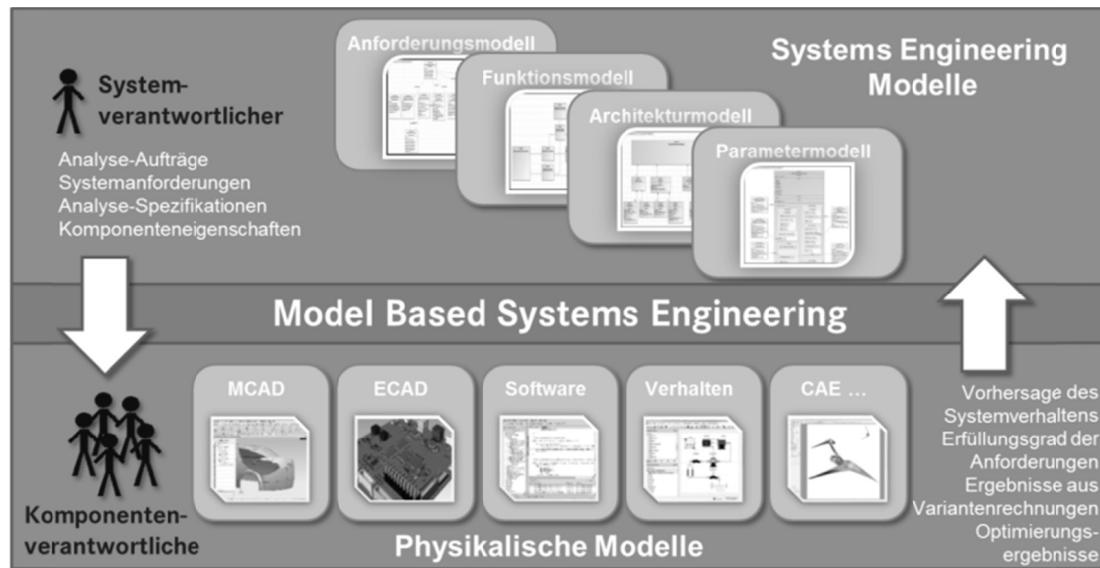


Bild 1: Model Based Systems Engineering

2 Nutzen von Systems Engineering

Unternehmen stellen sich bei der Betrachtung und Einführung berechtigt die Frage, was Ihnen SE bezogen auf ihre konventionellen und etablierten Entwicklungsprozesse an Mehrwert bietet. Zu Recht muss auch die Frage gestellt werden, ob und wann sich die Investitionen in SE-Methoden und IT-Werkzeuge für ein Unternehmen rechnen.

In der Arbeit „Systems Engineering Return on Investment“ von Eric C. Honour aus dem Jahr 2013 wurde in einer breit angelegten Studie der Return on Investment (ROI) für Systems Engineering ermittelt [Hon13]. Dabei wurden 51 Entwicklungsprogramme aus 16 Unternehmen der Luft- und Raumfahrtindustrie sowie der Verteidigungsindustrie analysiert. Ein wesentliches Ergebnis war, dass der optimale Aufwand für SE bei 14,4% der gesamten Entwicklungskosten für ein mittleres Entwicklungsprogramm (ca. 14 Mio. EUR Entwicklungsbudget) liegt. Bei kleineren oder größeren Programmen und in Abhängigkeit von weiteren Charakteristiken (Tiefe der Systemintegration, Größe der Systems, Technologisches Risiko etc.) kann der Wert zwischen 8% und 19% variieren. Bei einer mittleren Programmgröße lag der ermittelte SE-ROI bei 3,5:1.

Bild 2 zeigt an welchen Stellen der Nutzen von SE generiert werden kann. Mit Hilfe des modellbasierten Systems Engineering (MBSE) wird der Übergang von der heutigen häufig versuchs- und prototypgetriebenen funktionalen Absicherung zur virtuellen Systemabsicherung verlagert (1). Dabei können sowohl die Entwicklungskosten (2) als auch die Entwicklungszeit (3) reduziert werden.

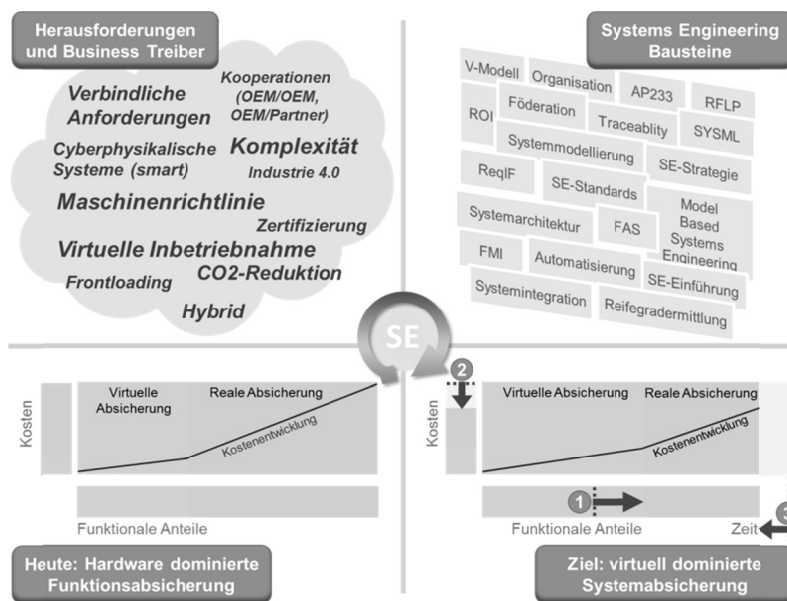


Bild 2: Nutzeneffekte durch den Einsatz von Systems Engineering Bausteinen

Der Nutzen von MBSE stellt sich dabei durch mehrere Effekte ein [KK14]:

MBSE hilft die Komplexität zu beherrschen: Die Vernetzung der Systeme nimmt durch heutige und zukünftige Innovation massiv zu und die Beherrschung dieser domänenübergreifenden Abhängigkeiten und Zusammenhänge kann nur mit SE-Methoden bewältigt werden. Die Transparenz und die Nachvollziehbarkeit der Abhängigkeiten sind ein wesentlicher Bestandteil zur Steigerung der Effizienz im Entwicklungsprozess. Durch den Einsatz von SE/MBSE-Methoden ist der interdisziplinäre Änderungsprozess einführbar und umsetzbar.

MBSE ermöglicht interdisziplinäres Qualitätsmanagement und Absicherung: Die Konformität der Systemeigenschaften zu den Anforderungen (z.B. Kundenanforderungen, Normen, Zertifizierungen oder Länderrichtlinien) bezüglich einzelner Komponenten und bezüglich des Gesamtsystems muss während des Entwicklungsprozesses sichergestellt werden. Hier setzt die SE-Methode an, indem jederzeit während der Entwicklung eine Beurteilung des Erfüllungsgrades von Anforderungen bezogen auf das Gesamtsystem erfolgt.

MBSE reduziert Hardware und Prototypen: Insbesondere in den etablierten Versuchs- und Berechnungsdisziplinen hat man die Anzahl der realen Hardwaretests durch Einsatz der Simulation erheblich reduziert und damit Entwicklungszeit und -kosten eingespart. Diese Entwicklung erfährt Ihre Fortführung in einer zunehmenden Virtualisierung der Absicherung - also in einem Übergang von Hardware-(HiL) über Software-(SiL) bis hin zu Model-in-the-Loop(MiL)-Tests. In gleichem Maße kann durch SE das multidisziplinäre Systemverhalten in frühen Entwicklungsphasen durch die Vernetzung von Verhaltenssimulationen aus unterschiedlichen Disziplinen erfolgen. Das Vorhandensein entsprechender Toolschnittstellen erlaubt die Weiterverwendung der Verhaltensmodelle für virtuelle Tests in allen Modellgranularitäten und somit in allen Phasen des Entwick-

lungsprozesses. Damit kann die Anzahl von Hardwareprototypen durch die frühe virtuelle Absicherung reduziert werden.

MBSE ermöglicht kooperative Entwicklungsprozesse: Die Zusammenarbeit mit Lieferanten und Entwicklungspartnern sowie die OEM/OEM-Kooperation ist im Konstruktionsbereich und insbesondere beim Austausch von CAD-Daten seit langem gelebte Praxis. Mit SE-Methoden kann auch in den frühen Phasen der Produktentwicklung die Einbindung von Lieferanten zur Anforderungsdefinition, Systemgestaltung und Lösungsoptimierung ermöglicht werden. Unterstützt wird dies durch die zunehmende Standardisierung von Schnittstellen zur Werkzeugkopplung. Exemplarisch sei hier auf das Functional Mock-up Interface verwiesen, einen werkzeugunabhängigen Standard für den Austausch funktionaler Verhaltensmodelle [FMI14-ol].

Als übergeordnetes Ziel von MBSE kann die Steigerung der Effizienz bei der Entwicklung mechatronischer Systeme durch die Bereitstellung von modellbasierten Methoden und die durchgängige Vernetzung heterogener Entwicklungswerkzeuge formuliert werden.

3 Systemmodellierung und Einführung von Model Based Systems Engineering

Der Entwicklungsprozess ist geprägt durch das ständige Entscheiden zwischen verschiedenen Konzept- und Lösungsalternativen, um die beste Lösung für das Produkt zu erzielen. In der Regel besteht dabei ein Zielkonflikt zwischen der Anzahl der möglichen Lösungen und der im Entwicklungsprojekt verfügbaren Zeit, um alle Lösungsalternativen zu bewerten und frühzeitig Entscheidungen zu treffen.

Einen Ansatz zur systematischen Entwicklung von Lösungskonzepten, Systemarchitekturen und Komponenten auf Grundlage des modellbasierten Systems Engineering (MBSE) bietet beispielsweise die sogenannte FAS-Methode [FAS14-ol]. Die Methode „Functional Architecture for Systems“ (FAS) wird als ein Kommunikationsinstrument verstanden und kann zur Herleitung von Architekturentscheidungen herangezogen werden. Die FAS-Methode ist unabhängig von Modellierungssprachen und von Modellierungswerkzeugen. Modellierungswerkzeuge auf Basis der Sprache SysML unterstützen die modellbasierte Analyse und Synthese von Systemen aus der Sicht des Systemingenieurs [Wei14]. Dabei werden unter anderem Anforderungs-, Funktions- und Architekturmodelle entwickelt, die um erste Parameter- und Verhaltensmodelle in der Konzeptphase auf Systemebene ergänzt werden. Dabei entsteht das sogenannte Systemmodell (siehe Bild 3).

Gerade im SE ist es entscheidend, die Festlegungen unterschiedlicher Disziplinen in einem Systemmodell zu beschreiben und integriert zu behandeln. Die reine Integration der Daten, die das Systemmodell beschreiben, in eine Datenbank ist dabei nicht hinreichend und zielführend. Das gilt insbesondere wenn bereits etablierte und existierende

IT-Werkzeuge im Entwicklungsprozess weiterhin verwendet werden sollen. Vielmehr ist der föderative Integrationsansatz die geeignete Lösung, wenn Daten zwischen heterogenen Entwicklungswerkzeugen verknüpft werden müssen [Kle03].

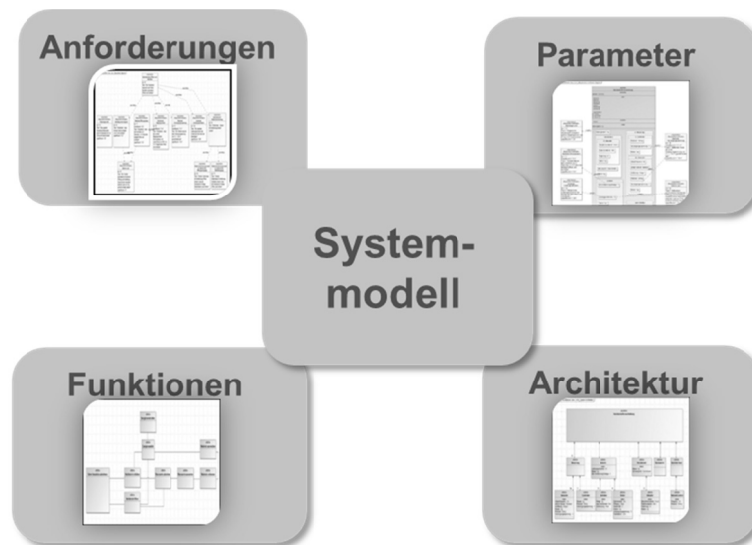


Bild 3: Systemmodell

Die Verwendung des MBSE-Ansatzes ermöglicht Systemingenieuren oder Systemarchitekten alle Aspekte eines multidisziplinären Systems zu berücksichtigen und zu optimieren. Sie stehen dabei stetig vor der Herausforderung während des Entwicklungsprozesses zu prüfen, inwieweit sich der Systementwurf und das Verhalten des Gesamtsystems innerhalb der definierten Anforderungen befinden. Gleichzeitig widerspricht die detaillierte Beherrschung aller dafür notwendigen Autorenwerkzeuge durch den Systemarchitekten dem Ziel einer Effizienzsteigerung sowohl im Hinblick auf das Handling der anspruchsvollen Umgebungen als auch bezüglich der damit verbundenen Lizenzkosten.

Der modellbasierte SE-Ansatz (Model Based Systems Engineering - MBSE) sieht darüber hinaus die Verknüpfung der physikalischen Modelle der einzelnen Disziplinen zu einem übergeordneten Systemmodell vor. MBSE-Entwicklungswerkzeuge, die die grafische Modellierung mit der Sprache SysML unterstützen, werden vermehrt zur Systemmodellierung eingesetzt. Sie sind, wenn es um die Möglichkeiten geht, das Systemmodell mit den konkreten physikalischen Verhaltens- und Simulationsmodellen zu verknüpfen, jedoch limitiert. Diese Verknüpfung ist unbedingt notwendig, um Vorhersagen zur Einhaltung von Anforderungen, zu Gestalt und Verhalten des Gesamtsystems oder zu den Kosten zu treffen. Die transparente Darstellung der Lösungsalternativen und die Unterstützung bei der Entscheidungsfindung sollten dabei auf Knopfdruck möglich sein [F4M14-01].

CAD-Systeme unterstützen die Beschreibung der mechanischen oder elektrischen Systemstruktur und Produktgestalt. CAE-Werkzeuge helfen bei der Vorhersage physikalischer Eigenschaften oder bei der Simulation des Verhaltens von Teilsystemen auf Basis

von Modellen im Rahmen der virtuellen Produktentwicklung (siehe Bild 2). Jedoch bleibt die Optimierung des Gesamtsystems häufig ein zeitintensiver und auf Erfahrungen basierender Prozess, der zusätzlich mit Hilfe von Prüfständen oder Prototypen für Tests unter realen Bedingungen ergänzt wird. Um die Optimierung eines mechatronischen Gesamtsystems durchführen zu können ist die interdisziplinäre und automatisierte Kopplung verschiedenster CAX-Werkzeuge und Modelle notwendig. Dabei müssen Parameter, Eigenschaften, Eingangs- und Ergebnisinformationen, Varianten und Berechnungsalternativen durchgängig im interdisziplinären Entwicklungsprozess als Bindeglied zwischen den Modellen verfügbar sein.

Zahlreiche Herausforderungen sind dabei zu bewältigen, wie z.B.

- jederzeit prüfen zu können, ob die Anforderungen an das Gesamtsystem über alle Disziplinen hinweg erfüllt sind,
- verschiedene im Berechnungsprozess existierende CAX-Werkzeuge zu integrieren und Parameter und Ergebnisse zwischen den Werkzeugen auszutauschen,
- einen reproduzierbaren Berechnungsdurchlauf sicherzustellen und auch nach Jahren nochmals nachvollziehen zu können,
- Informationen aus verschiedenen Disziplinen, wie Mechanik, Elektrik/Elektronik oder Software in die Simulation des Gesamtsystems einzubinden,
- Auswirkungen von Änderungen an einzelnen Modellen auf das Gesamtsystem nachvollziehen und beurteilen zu können oder
- trotz Heterogenität in der Werkzeuglandschaft und der mangelnden Verschaltung der existierenden, hochoptimierten Berechnungs- und Simulationswerkzeuge, belastbare und nachvollziehbare Ergebnisse für das betrachtete Gesamtsystem zu liefern.

Für die Einführung von (MB)SE gilt es zunächst eine geeignete Strategie für den Einsatz zu definieren und die Einführung zu planen und erfolgreich umzusetzen. Dabei hat sich in verschiedenen Einführungsprojekten im Maschinenbau folgende Vorgehensweise bewährt:

- Stufenweise Einführung von Model Based Systems Engineering (MBSE) in kleinen Schritten mit Hilfe von Pilotprojekten (sogenannte „MBSE-Labore“)
- Umstellung auf parallele Systementwicklung entlang des V-Modells statt sequentielle Entwicklung
- Unterstützung mit Hilfe von externer Expertise und Wissensmanagement

Die Einführung von MBSE betrifft mehrere Bereiche (ME, EE, SW) und erfordert Berücksichtigung des gesamten Produktlebenszyklus. Dabei ist die Akzeptanz und Transparenz in den verschiedenen Unternehmensbereichen sicherzustellen. Da Prozesse, Me-

thoden und Werkzeuge für modellbasierte Entwicklung noch nicht hinreichend bekannt und akzeptiert sind, sind Sensibilisierung, Erfahrungsaufbau, Eignungsprüfung und Anpassungen in Pilotanwendungen für den Einsatz in Serienentwicklung notwendig.

Der vorliegende Beitrag zeigt die Erfahrungen aus der Einführung und Erstanwendung von MBSE an einem konkreten Beispiel der Maschinenentwicklung auf.

4 Praxiserfahrungen für einen föderativen MBSE-Ansatz in der Maschinenentwicklung auf Basis von ModelCenter

Exemplarisch soll anhand des folgenden Anwendungsbeispiels das modellbasierte Systems Engineering beschrieben werden. Es handelt sich um den Entwurf eines mechatronischen Funktionsträgers. Der Funktionsträger ist ein Sub-System einer Textilmaschine, das für die Erstellung von Mustern bei industriell hergestellten Stoffen verantwortlich ist. Das Sub-System besteht aus einem pneumatischen Zylinder (1), dem eigentlichen Funktionsträger (2), einem Servomotor (3) sowie der Steuerung (4). Die Steuerung legt die Verfahrkurve des Funktionsträgers fest (siehe Bild 3).

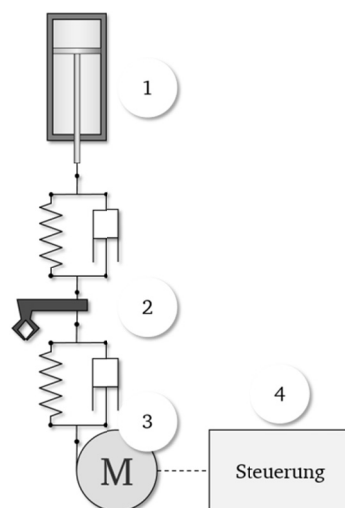


Bild 4: Konzeption des Funktionsträgers als Sub-System einer Textilmaschine

Ziel der modellbasierten Entwicklung ist die Konzeption und Absicherung alternativer Antriebskonzepte, um die Leistungsanforderung an eine erhöhte Drehzahl und damit die Erhöhung der Stoffmenge pro Zeit zu erfüllen. Im Folgenden sollen die angewandte Vorgehensweise, Methoden und Werkzeuge beim modellbasierten Systems Engineering zusammenfassend beschrieben werden.

Zur Modellierung der funktionalen Architektur wurde wie folgt vorgegangen: System Kontext identifizieren, Anforderungsanalyse, Anwendungsfälle und deren Verfeinerung durch Aktivitäten, funktionale Blockdiagramme erstellen, Ausarbeitung der funktionalen Modelle bis zur logischen Architektur sowie Identifikation der Wirkprinzipien des Systems, Gewinnung von CAD-Modellen anhand der erkannten Wirkprinzipien, Ge-

winnung eines multidisziplinären Simulationsmodells und Durchführung von Untersuchungen in der Simulation. Dabei wurden auf Basis der beschriebenen Anforderungen sowohl Systemkontext, Anforderungsmodelle, Anwendungsfälle als auch verschiedene Systemmodelle (Systemarchitektur, Funktionsstruktur, etc.) mit der Sprache SysML modelliert.

Der Systemkontext gem. Bild 4 beschreibt zunächst die von den Projektbeteiligten identifizierten Systeme mit Hilfe der Beschreibung in SysML. Zudem werden auf Grundlage des Systemkontextes Anforderungen und Anwendungsfälle des Systems in SysML beschrieben. Unter Anwendung der Methode Functional Architecture for Systems (FAS) entsteht eine lösungsneutrale funktionsorientierte Systemstruktur. Diese wird abschließend in Zielsystemarchitekturen für Lösungsalternativen überführt und ebenfalls in SysML mit Hilfe des Autorensystems CAMEO Systems Modeler von NoMagic dokumentiert. Exemplarisch wird in Bild 5 der Systemkontext für den Funktionsträger in einem SysML-Diagramm dargestellt.

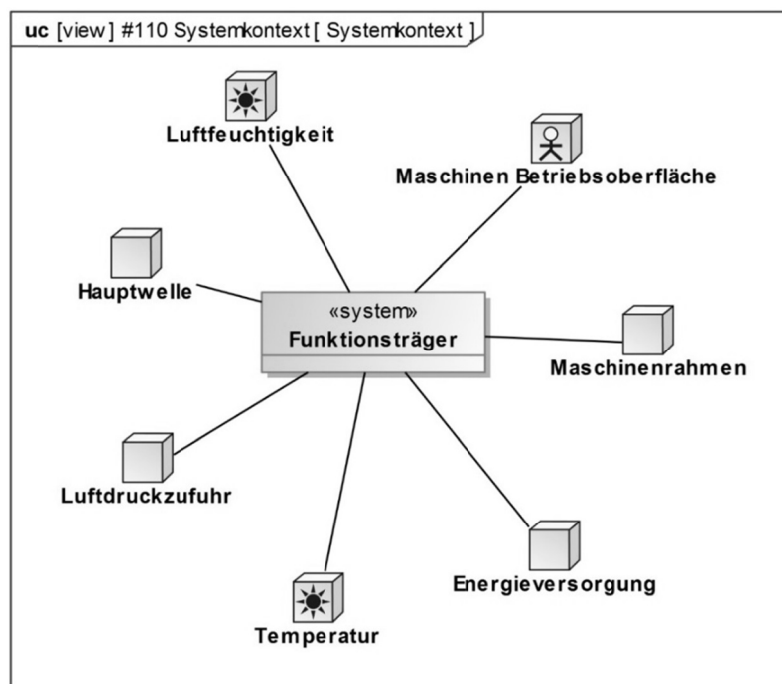


Bild 5: Beschreibung des Systemkontexts für den mechatronischen Funktionsträger

Basierend auf den Anforderungen konnten bereits in der Konzeptphase die Lösungsalternativen auf Grundlage der ausgewählten Komponenten für die Zielarchitektur und das Antriebskonzept mit Hilfe der Modellierungssprache Modelica in einem Verhaltensmodell beschrieben und die multiphysikalischen Eigenschaften im multidisziplinären Simulationswerkzeug SimulationX simuliert werden.

Dazu ist es erforderlich, dass die fachdisziplinspezifischen Eigenschaften von Systemmodell und Partialmodellen aus verschiedenen Werkzeugen miteinander verknüpft werden. Bei der Verwendung von vorhandenen Werkzeugen in einer heterogenen IT-

Systemumgebung für die Produktentwicklung eignet sich die Verknüpfung auf Basis des föderativen Integrationsansatzes.

Die Verknüpfung der Daten aus den Partialmodellen und die Föderation der Werkzeuge Excel zur Vorauslegung, SimulationX zur Simulation, Matlab zur Auswertung der Ergebnisse und CAMEO Systems Modeler zur Gegenüberstellung von Ergebnissen und Anforderungen wird dabei mit Hilfe der Entwicklungsplattform ModelCenter MBSE-Pak für das modellbasierte Systems Engineering realisiert. Im sogenannten Dashboard auf Basis von ModelCenter werden die Simulationsergebnisse angezeigt und diese mit den Anforderungen aus dem SysML-Modell verglichen (siehe Bild 6).

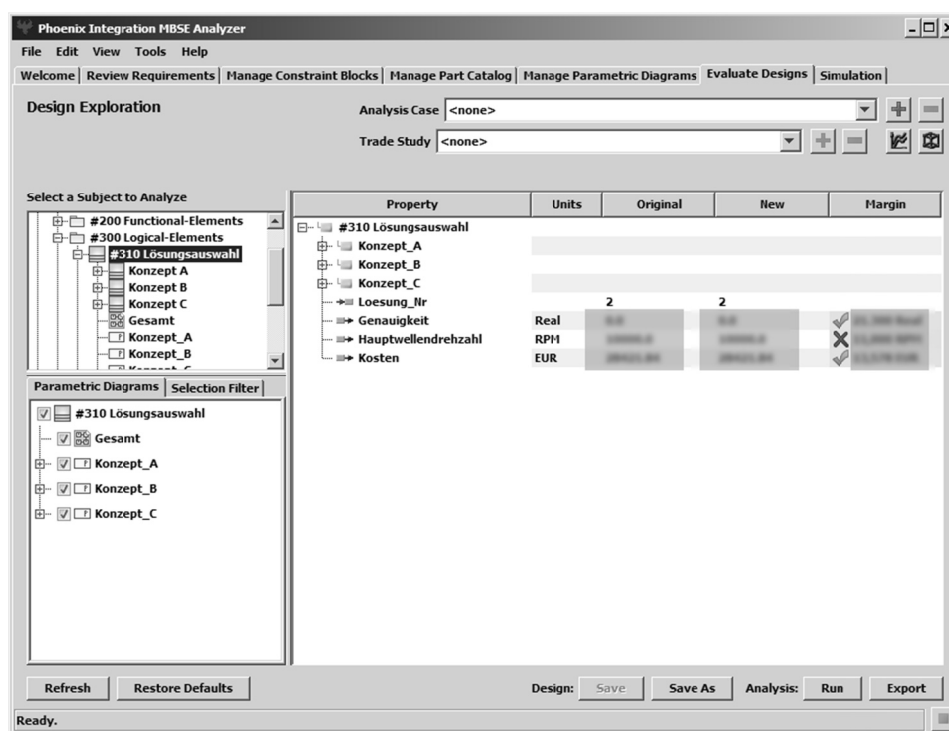


Bild 6: Dashboard in ModelCenter MBSE Analyzer zur Darstellung der Ergebnisse

Da wesentliche Parameter des Systems und insbesondere auch kritische geometrische Größen in den Anforderungen formuliert sind, wurde bereits in der frühen Konzeptphase die interdisziplinäre Kommunikation mit Hilfe der Simulation unterstützt. Die erzielten Simulationsergebnisse hatten dabei eine hohe Genauigkeit und haben so nicht nur frühzeitige Entscheidungsfindung ermöglicht, sondern auch Akzeptanz für das Vorgehen geschaffen. Als Ergebnis des Pilotprojekts soll die modellbasierte Systementwicklung unternehmensweit eingeführt werden.

ModelCenter ist eine Plattform mit der sowohl die Integration von Entwicklungsprozessen als auch die Design Optimierung von Gesamtsystemen durchgeführt werden kann. Dabei ist es von entscheidender Bedeutung, dass die existierenden Design-, Simulations- und Berechnungswerkzeuge im Unternehmen weiter verwendet werden können.

Mit Hilfe von ModelCenter können schnell und einfach CAx-Workflows definiert werden, die automatisierte und wiederholbare Simulationen sicherstellen. Durch die Ausführung der Workflows können Ergebnisse aus verschiedenen Werkzeugen erfasst, Designalternativen evaluiert, gegen definierte Anforderungen geprüft und das Verhalten visualisiert werden. Somit können Entscheidungen, die ein lokales Optimum für die Entwicklung bedeuten, hinsichtlich der für das Gesamtsystem besten Lösung überprüft, verifiziert und ggf. verworfen werden. In den wenigsten Fällen ist die Optimierung nur einer Stellgröße im System aus Gesamtsystemsicht.

Bild 7 zeigt beispielhaft die beteiligten Werkzeuge bei der modellbasierten Systementwicklung für den mechatronischen Funktionsträger einer Textilmaschine. ModelCenter dient dabei als Informationsdrehzscheibe für Parameter und Ergebnisse zur Prozessintegration und Systemoptimierung für das Zusammenspiel verschiedenster Entwicklungswerkzeuge sowie zur Reifegradbeurteilung mit Hilfe eines Dashboards im Rahmen des modellbasierten Systems Engineering.



Bild 7: ModelCenter als zentrale Informationsdrehzscheibe für das modellbasierte Systems Engineering

5 Resümee und Ausblick

Die Einführung von Systems Engineering für die mechatronische Produktentwicklung gewinnt auch in den klassischen Maschinenbauunternehmen an Bedeutung und wird derzeit intensiv diskutiert. Unternehmen stellen sich dabei die Frage, was der Einsatz von MBSE bezogen auf ihre konventionellen und etablierten Entwicklungsprozesse an Mehrwert bietet und welcher Nutzen hinsichtlich Verkürzung der Entwicklungszeit, Kosteneinsparung sowie Steigerung von Produktivität, Qualität und Innovation daraus entsteht.

Der Nutzen von MBSE stellt sich erst durch die ganzheitliche Betrachtung des Gesamtsystems über alle Disziplinen und Entwicklungsphasen ein. Dazu wird empfohlen zunächst eine Evaluierung der Vorgehensweise, Methoden und Werkzeuge zum modellba-

sierten Systems Engineering in einem Pilotprojekt durchzuführen. Denn das Thema MBSE klingt für die Beteiligten in der Maschinenentwicklung zunächst theoretisch und abstrakt und wird erst durch die praktische Anwendung begreifbar und nachvollziehbar. Zudem können im Rahmen der Evaluierung Methoden und Werkzeuge ausprobiert, bewertet und für den späteren Einsatz ausgewählt werden und auf Basis der ersten Erfahrungen notwendige Anpassungen an die Rahmenbedingungen für die Einführung vorgenommen werden.

Die Unterstützung der Anwender bei der Einführung durch MBSE-Experten ist erforderlich, um Wissen im Sinne von Know-that, Know-why, Know-how zu transferieren und damit grundlegend Entwickler zu befähigen sowie direkten Nutzen durch den Einsatz von MBSE zu schaffen. Eine individuelle, modulare Einführungsstrategie (z.B. projektbezogener Ersteinsatz) empfiehlt sich nach Sensibilisierung und Qualifizierung der beteiligten Mitarbeiter. Sind die Anfangshürden erst einmal überwunden, können bisher nicht erreichbare Effizienzpotentiale im Entwicklungsprozess aufgedeckt und realisiert werden.

Die Systemkomplexität im Maschinenbau wird zukünftig noch weiter steigen - die Variantenvielfalt und die Individualisierung der Produkte beschleunigen diesen Prozess zusätzlich. Daher ist die rechtzeitige Planung und Einführung von SE-Methoden entscheidend für die Wettbewerbsfähigkeit zukünftiger Maschinen und damit indirekt auch für den mittel- und langfristigen Unternehmenserfolg.

Literatur

- [FAS14-ol] www.fas-method.org, aufgerufen am 30.11.2014
- [F4M14-ol] www.fasform.org, aufgerufen am 30.11.2014
- [FMI14-ol] www.fmi-standard.org, aufgerufen am 30.11.2014
- [GfSE14-ol] www.gfse.de, aufgerufen am 30.11.2014
- [HON13] HONOUR, E. C.: Systems engineering return on investment. Dissertation, University of South Australia, 2013
- [KK14] KLEINER, S.; KRASTEL, M.: Modellbasiertes Systems Engineering: Einführung und Einsatz von MBSE für die Fahrzeugentwicklung und im Maschinenbau, Deutschsprachige NAF-EMS Konferenz 2014, Bamberg
- [KLE03] KLEINER, S.: Föderatives Informationsmodell zur Systemintegration für die Entwicklung mechatronischer Produkte, Shaker, 2003
- [MCP14-ol] www.mecpro.de, aufgerufen am 30.11.2014
- [WEI14] WEILKIENS, T.: Systems Engineering mit SysML / UML. dpunkt.verlag, 2014

Autor

Dr. Sven Kleiner verantwortet den Geschäftsbereich CAD|CAE. Außerdem ist Sven Kleiner als COO (Chief Operating Officer) verantwortlich für das operative Geschäft. Dr. Sven Kleiner begann seine berufliche Laufbahn 1995 bei der ABB Flexible Automation GmbH in Friedberg im Bereich Kundendienst/Service und war ab 1999 als wissenschaftlicher Mitarbeiter an der TU Darmstadt im Fachbereich Maschinenbau am Fachgebiet DiK (Datenverarbeitung in der Konstruktion) tätig. Er promovierte 2003 zu dem Thema "Föderatives Informationsmodell zur Systemintegration für die Entwicklung mechatronischer Produkte". Im Jahr 2009 erhielt Sven Kleiner eine Berufung als Lehrbeauftragter für „CAD“ an die Hochschule Darmstadt und bietet dort Lehrveranstaltungen im Fachbereich Maschinenbau und Kunststofftechnik an. Ferner ist Dr. Sven Kleiner Mitglied im VDI Fachausschuss „Wissensmanagement im Engineering“ und „Wissensbasierte Konstruktion“

Ein durchgängiges Vorgehen zur Anforderungserfassung und -nachverfolgung

Nicholas Schmitt

AUDI AG

85045 Ingolstadt

Tel. +49 (0) 841 / 89 57 03 14

E-Mail: nicholas.schmitt@audi.de

Dr.-Ing. Roman Dumitrescu

Fraunhofer-Institut für Produktionstechnologie IPT

Projektgruppe Entwurfstechnik Mechatronik

Zukunftsmühle 1 , 33102 Paderborn

Tel. +49 (0) 5251 / 54 65 124

E-Mail: Roman.Dumitrescu@ipt.fraunhofer.de

Zusammenfassung

Die Entwicklung moderner Systeme im Automobil wird von mehreren Komplexitätstreibern beeinflusst. Durch steigende Kundenansprüche im Bereich Sicherheit, Umweltfreundlichkeit, Qualität und Komfort finden sich in modernen Fahrzeugen zunehmend komplexe mechatronische Systeme. Diese Systeme sind im Betrieb vollständig vernetzt. Eine kundenerlebbare Gesamtfunktionalität kann nur durch ein nahtloses und abgestimmtes Zusammenspiel aller an der Entwicklung beteiligten Disziplinen erreicht werden. Die verteilte Entwicklung der mechatronischen Systeme über mehrere Lieferantenebenen hinweg stellt eine zusätzliche Herausforderung dar. Die frühzeitige Erfassung aller relevanten Anforderungen an ein solches mechatronisches System stellt einen der zentralen Erfolgsfaktoren für eine zielgerichtete Entwicklung dar. Das vorgestellte Vorgehen stellt einen in der Industrie validierten Ansatz dar, um Anforderungen aller Stakeholder frühzeitig zu erfassen. Zudem bindet das Vorgehen sämtliche Lieferantenebenen ein und erlaubt ein anforderungsbasiertes Reifegradmanagement.

Schlüsselworte

Mechatronik, Anforderungserfassung, MBSE, Automotive

Consistent approach for requirements elicition and traceability

Abstract

The development of modern automotive systems is influenced by several complexity drivers. By increasing customer expectations in safety, environmental performance, quality and comfort an increasingly amount of complex mechatronic systems can be found in modern vehicles. These systems are interconnected. A new functionality can only be achieved through a seamless and coordinated interaction of all disciplines involved in the development. The distributed development of mechatronic systems across multiple suppliers provides an additional challenge. The early identification of all relevant requirements of such a mechatronic system is one of the key success factors for a focused development. The proposed approach is used to capture requirements of all stakeholders at an early stage in development and is validated in industrial application. In addition, the approach integrates all levels of suppliers and allows a requirements-based maturity stage control.

Keywords

Mechatronics, requirements engineering, MBSE, automotive

1 Einleitung

In modernen Automobilen steigt der Anteil an mechatronischen Systemen zunehmend. In der Entwicklung dieser mechatronischen Systeme stellt die hohe Komplexität eine große Herausforderung dar. Die etablierten Entwicklungsmethoden des klassischen Maschinenbaus, wie die Konstruktionslehre nach Pahl/Beitz [PB07] und der Mechatronik, sowie die VDI-Richtlinie 2206 „Entwicklungsmethodik für mechatronische Systeme“ [VDI2206], können dieser Herausforderung nicht gerecht werden. Dies trifft insbesondere auf die frühen Phasen der Entwicklung zu. Nur mittels eines einheitlichen Systemverständnisses aller Fachleute und Entwicklungsdisziplinen können die geforderten Systemanforderungen identifiziert werden. Systemanforderungen werden benötigt, um neben den Komponenten- und Bauteilanforderungen die Gesamtheit des zu entwickelnden mechatronischen Systems zu beschreiben. Systems-Engineering bietet hierfür geeignete Lösungsmöglichkeiten. Durch das Fokussieren auf ein ganzheitliches Systemdenken und das Einbeziehen von Systemmodellen als zentrales Element im Bereich des Model-Based Systems-Engineering wird dies erreicht. Zur Abbildung des Wissens von interdisziplinären Entwicklungsteams und zur Gewährleistung einer zielgerichteten Systementwicklung besteht daher die Notwendigkeit eines durchgängigen, disziplinübergreifenden Vorgehens im Sinne des Systems-Engineering [INC07]. Hierbei stehen vor allem die geforderten Systemeigenschaften und -funktionen im Fokus. Das Ergebnis dieses Vorgehens sind alle relevanten Anforderungen an ein mechatronisches System.

In der industriellen Praxis ist ein solches Systems-Engineering-Vorgehen jedoch nicht flächendeckend etabliert. Vor dem Hintergrund der verteilten Entwicklung steigt der Bedarf nach einem flächendeckenden Einsatz eines solchen Vorgehens jedoch zunehmend an. Obwohl gerade der überwiegend fachdisziplinspezifische Komponentenentwurf eine übergeordnete systemische Sicht und ein durchgängiges Anforderungsmanagement erfordern, wird dies aufgrund teils fehlendem Abstraktionsvermögen (insbesondere einer funktionalen Betrachtung), Kapazitäts- und Zeitmangel nicht durchgeführt. Eine weitere Herausforderung ist, dass das Wissen über die Methoden und Modelle des Systems-Engineering bei den Entwicklern nur mangelhaft ausgeprägt ist. Ebenfalls sind zentrale Rollen des Systems Engineering, wie von SHEARD vorgeschlagen der Systems-Engineer oder der Systemarchitekt nicht durchgängig etabliert [She96]. Zudem müssen Stakeholder integriert werden, die nicht über ein tiefgreifendes technisches Wissen verfügen [ALL07a]. Das in diesem Beitrag vorgestellte Vorgehen bietet einen zielgerichteten Ansatz zur Erfassung der relevanten Systemanforderungen, unterstützt die Ableitung von Komponentenanforderungen und sorgt mittels einer anforderungsbasierten System-Reifegraderfassung für eine frühzeitige Risikoidentifizierung im Entwicklungsprozess. Das Gesamtverfahren wird erfolgreich bei der AUDI AG eingesetzt.

2 Motivation

Das Thema Anforderungserfassung und -management stellt eines der zentralen Handlungsfelder eines durchgängigen Systems Engineering Vorgehens dar [All08, WW02]. Daher wird das Thema in der Wissenschaft und industriellen Praxis kontinuierlich erforscht und weiterentwickelt. In der Literatur finden sich zahlreiche Vorgehensmodelle und Methoden. Bei den meisten dieser Ansätze werden die Anforderungen auf unterschiedliche Weise erarbeitet. Die Hauptansätze sind Zielorientierung [All08, Lam04], Anwendungsfallbetrachtungen (Use Cases) [All08, Jar99] und Sichtenbildung (Stakeholder) [All08, Jar99]. Zudem bildet die Traceability (Verfolgbarkeit) der Anforderungen einen seit Jahren weiteren Hauptforschungsschwerpunkt [Wie95].

All die oben genannten Vorgehensmodelle und Methoden haben gemeinsam, dass dem Sachverhalt der verteilten Entwicklung unzureichend Rechnung getragen wird. Verteilte Entwicklung zeigt sich in verschiedenen Ausprägungen und Ebenen in der heutigen Automobilindustrie. Mechatronische Systeme werden vermehrt für sogenannte modulare Fahrzeugbaukästen entwickelt. Dies bedeutet, dass alle Marken eines Automobilkonzerns die Systeme in ihre Fahrzeuge integrieren können müssen. Daher wird die Möglichkeit einer dezentralen Erarbeitung der Modelle und der Anforderungen in verteilten Entwicklungsteams, gestützt durch ein Vorgehensmodell und Gütekriterien, benötigt (Horizontale Anforderungserfassung) (siehe Bild 1). Hierbei liegt der Fokus auf der Erarbeitung von Anforderungen der richtigen Granularität. Eine fehlerhafte Systemumsetzung resultiert zumeist aus fehlerhaften Anforderungen, die Annahmen über ein System und dessen Komponenten zulassen [Alm06].

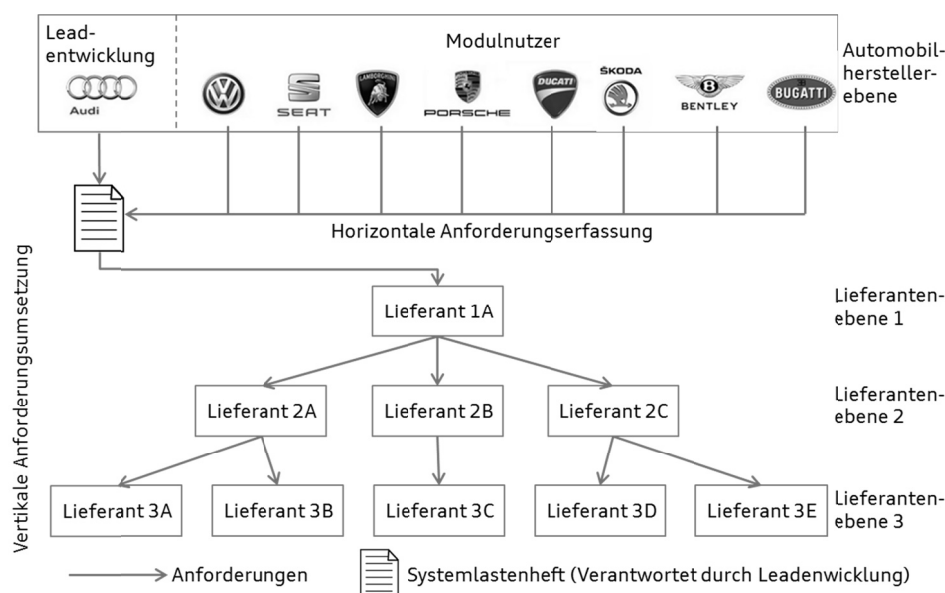


Bild 1: Beispielhafte Darstellung der Ebenen der verteilten Entwicklung in der Automobilindustrie und Darstellung des Anforderungsflusses

Sind alle Anforderungen an ein mechatronisches System erfasst, werden diese durch die Lieferantenkette umgesetzt. Dies stellt die zweite Herausforderung im Bereich der ver-

teilten Entwicklung dar. Im Bereich der vertikalen Anforderungsumsetzung und Anforderungsdekomposition wird ein Verfahren zur Verfolgung von Anforderungen und deren Umsetzungsreife benötigt (siehe Bild 1).

Für die Entwicklung und die Ableitung von Anforderungen eines mechatronischen Systems wird ein zentrales Arbeitsmedium und Informationsträger benötigt, der sowohl von allen beteiligten Entwicklern, als auch von allen beteiligten Unternehmen auf vertikaler und horizontaler Ebene gelesen, verstanden und weiter detailliert werden kann (siehe Bild 1). Hierzu eignet sich ein Systemmodell, das sämtliche benötigten Sichtweisen und sämtliche Stakeholderbedürfnisse beinhaltet [INC07]. Zudem muss dieses Systemmodell das Ausleiten von Anforderungen in der richtigen Granularität ermöglichen. Ein ganzheitliches Vorgehen muss das Erstellen des Systemmodells in verteilten Entwicklungsteams mit Mitgliedern ohne tiefgreifende Systems Engineering Kenntnissen ermöglichen.

3 Vorgehen

Das Gesamtverfahren zur Erarbeitung von Anforderungen an ein mechatronisches System mittels eines durchgängigen Systems Engineering Vorgehens bei Audi ist in Bild 2 dargestellt. Es umfasst mehrere Einzelmethode, die verschiedene Sichten auf ein System generieren. Jede der vorgestellten Einzelmethode beinhaltet eine klare Vorgangsbeschreibung sowie formale Gütekriterien.

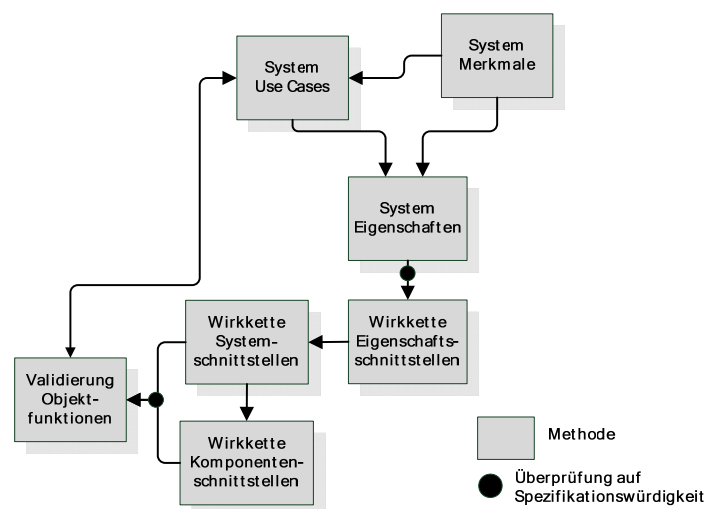


Bild 2: Gesamtverfahren zur Anforderungserarbeitung für mechatronische Systeme

Die Methode „Validierung Objektfunktionen“ dient der Bewertung des Gesamtverfahrens. Hierdurch ist es möglich festzustellen, ob das Gesamtverfahren und dessen Einzelmethode richtig bearbeitet wurden und alle benötigten Informationen im Systemmodell abgebildet sind. Als Systemmodell dient hierbei eine sogenannte Wirkkette. Mittels der Methode „Überprüfung auf Spezifikationswürdigkeit“ werden zum einen

Eigenschaften aber auch Schnittstellen und Wechselwirkungen der Wirkkette einer inhaltlichen Prüfung unterzogen.

Das Gesamtverfahren wird anhand eines durchgängigen Beispiels vorgestellt. Dabei handelt es sich um ein Ladegerät für ein Elektrofahrzeug der AUDI AG (siehe Bild 3). Mit Hilfe dieses Ladegerätes wird ein Elektrofahrzeug an einer externen Stromversorgung an einer externen Stromversorgung geladen. Das Laden geschieht hierbei Kabelgebunden.



Bild 3: Ladegerät eines Audi A3 e-tron

3.1 Systemmerkmale

Die Systemmerkmale werden mittels der sogenannten Elementklassifikation erarbeitet. Diese Methode besteht im Kern aus einem Fragenkatalog zur Erfassung der Merkmale und einer Beziehungsmatrix zur Gewichtung der einzelnen Merkmale. Ein Merkmal hat einen allgemeingültigen, objektunabhängigen Charakter. Erst durch die Spezifikation eines Merkmals wird dies objektindividuell, -abhängig und wird als Eigenschaft bezeichnet.

Der für die AUDI AG erarbeitete Fragenkatalog enthält 107 Merkmale in 18 Kategorien. Diese Kategorien sind unter anderem: Kraft, Energie, Signal, Sicherheit, Geometrie, Stoff, Kinematik. Grundlage hierfür ist die Hauptmerkmalsliste nach Pahl/Beitz [PB07]. Die Inhalte der Liste wurden angepasst und um fahrzeugspezifische Aspekte erweitert, um den Anforderungen im Automotivumfeld gerecht zu werden. Die 107 Merkmale decken hierbei die Bedürfnisse aller relevanten Stakeholder ab. Zu diesen gehören unter anderem: der Kunde, die Produktion, der Kundendienst sowie der Gesetzgeber.

In einer Beziehungsmatrix werden die 107 Merkmale durch 4140 gewichtete Beziehungen miteinander verknüpft. Diese sind ein Ergebnis aus der Analyse bestehender interner Datenquellen. Hierzu zählen Lessons-Learned-Datenbanken, existierende Systemarchitekturen, Lastenhefte, Ergebnisse aus Reverse Engineering Prozessen, Testspezifikationen und Experteninterviews. Die gewichteten Beziehungen sind obligatorisch und auf den Automotivbereich ausgelegt. Durch eine Anforderungserhebung für bereits existierende Fahrzeugsysteme konnten die Merkmale und die gewichteten Beziehungen validiert und bestätigt werden.

Wurde der Fragenkatalog befüllt und die Beziehungsmatrix angewandt, so ergeben sich die hauptrelevanten Merkmale für ein zu entwickelndes mechatronisches System. Hierzu wird nach Bild 4 die Summe aus der gewichteten Beeinflussung eines Merkmals berechnet. Die Befüllung des Fragenkataloges dauert hierbei im Schnitt nur 15 Minuten und wird im Wesentlichen durch den systemverantwortlichen Entwickler durchgeführt.

Beziehungsmatrix		Fragestellung: „Wie stark beeinflusst das Merkmal A (Zeile) das Merkmal B (Spalte)?“					Projektspezifische Einschätzung	
0 = nicht relevant 3 = gering 6 = mittel 9 = hoch		Merkmal	Prüfvorschriften (TÜV, Gesetze)	Gesamtfahrzeugstörung	elektrische Leistung	...		Dichtigkeit
Merkmal	Nr.	1	2	3	...			
Prüfvorschriften (TÜV, Gesetze)	1	0	0	0	3	9	0	
Gesamtfahrzeugstörung	2	3	0	6	0	0	3	
elektrische Leistung	3	6	M_i	0	6	3	P_i	
...	...	0	0	0	0	0	9	
Dichtigkeit	n	0	3	0	6	0	6	
		$\Sigma (M_i \cdot P_i)$						
Gewichtung des Merkmals		1212	2061	1089	972	1872		

Bild 4: Beziehungsmatrix zur Ausleitung der hauptrelevanten Merkmale

Die hauptrelevanten Merkmale des Ladegerätes sind Prüfvorschriften (TÜV, Gesetze,...), Gesamtfahrzeugstörung, elektrische Leistung, Dichtigkeit und Misuse durch einen Benutzer. Diese ergeben sich aus dem vom Entwickler befüllten Fragenkatalog und durch die gegenseitige Beeinflussung der Merkmale der Elementklassifikation.

3.2 System Use Cases

Zur Erfassung und Ableitung der Anforderungen, die ein System bereitstellen muss, ist es unerlässlich die Funktionalität des Systems darzustellen. Die Use Cases leiten sich immer aus der Benutzung eines Systems durch einen Kunden ab. Da ein Benutzer/Fahrer meist nie direkt mit einem System im Fahrzeug interagiert, leiten sich die Funktionalitäten des Systems aus einer Interaktion mit dem Gesamtfahrzeug ab.

Ausgehend von der Funktionalität auf Gesamtfahrzeugebene werden die hauptrelevanten Systemfunktionen identifiziert. Diese werden anschließend weiter dekomponiert. Die Erarbeitung der Funktionalität geschieht mittels einer wiederkehrenden Fragestellung. Die Fragestellung wurde so gestaltet, dass bauteilunabhängige Funktionen erarbei-

tet werden. Basis der Beschreibung der Funktionen ist das Vorgehen definiert in der [VDI2222]. Dieses wurde abgewandelt und erweitert um die spezifischen Anforderungen einer Automobilentwicklung zu unterstützen. Zudem ist dies nötig, um bei der Erstellung der Funktionen und deren Unterfunktionen auf eine einheitliche und eindeutige Fragestellung zurückgreifen zu können. Die Fragestellung zur Erarbeitung der Funktionalitäten lautet: Welche Unterfunktion, die eine „physikalische Größe“, eine „Elektromagnetische Welle“, einen „Stoff/Betriebsstoff“, eine „Information“ oder ein „systemexternes Element“ beschreibt wird benötigt, um eine Funktion zu ermöglichen? Eine Funktion wird immer durch ein Substantiv und Verb beschrieben.

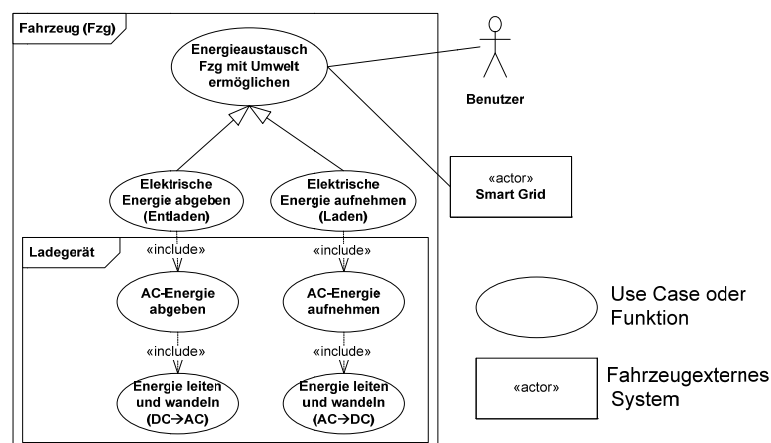


Bild 5: System Use Cases des Ladegerätes

Das Ladegerät muss beispielsweise die Funktionen „AC-Energie abgeben“ und „AC-Energie aufnehmen“ realisieren, um den Kunden Use Case „Energieaustausch Fahrzeug mit der Umwelt“ zu ermöglichen (siehe Bild 5).

3.3 Systemeigenschaften

Systemeigenschaften sind eine systemspezifische Ausprägung einer Funktionalität zur Erreichung einer kundenerlebbaren Systemgesamtfunktion. Der Einsatz von Eigenschaften in der Entwicklung ist unerlässlich. Einerseits lassen sich die Bedürfnisse von Stakeholdern mit nicht ausgeprägtem technischem Verständnis erfassen, zum anderen wird hierbei dem Sachverhalt der Entwicklung modularer Fahrzeugbaukästen Rechnung getragen. Die Ausprägung einer Funktion kann durch Markenwerte oder –image beeinflusst werden. Beispielsweise sind die Markenwerte der AUDI AG Sportlichkeit, Progressivität und Hochwertigkeit. Da die entwickelten Systeme markenübergreifend zum Einsatz kommen, ist es unerlässlich die markenspezifischen Systemeigenschaften zu kennen und in der Entwicklung mit zu berücksichtigen.

Die Systemeigenschaften werden mittels sogenannter Satzmuster erarbeitet. Die Satzmuster bestehen aus einem statischen und variablen Anteilen [FHM14]. Der variable Anteil besteht aus den System Use Cases und den Systemmerkmalen. Mittels der Satzmuster werden diese verknüpft und es ist gewährleistet, dass eine Rückverfolgbarkeit

von Eigenschaften hin zu Funktionalitäten und Merkmalen möglich ist. Die Satzmuster sind als Fragen aufgebaut und wiederkehrend. Die Beantwortung dieser Fragesatzmuster ergibt sämtliche relevanten Systemeigenschaften. Eines dieser Satzmuster lautet: Unter dem Aspekt <Merkmal> und <System Use Case> welche Eigenschaften muss das System besitzen?

Die Systemeigenschaften eines Ladegerätes bestehen beispielsweise aus einer „toleranzfreien Energieabgabe, gemäß externen Anforderungen“. Diese Eigenschaft wurde über die Fragestellung „Unter dem Aspekt elektrische Leistung und DC-Energie abgeben welche Eigenschaften muss das System besitzen?“ erarbeitet.

4 Wirkkette

Als Systemmodell kommt eine sogenannte Wirkkette zum Einsatz. Diese beinhaltet neben der Systemarchitektur bestehend aus der Systemabgrenzung, Komponenten, Bauelementen, Schnittstellen und Wechselwirkungen auch die System Use Cases, die Merkmale und die Systemeigenschaften. Hierdurch sind alle relevanten Elemente des Gesamtverfahrens in einem Modell vereint und es lässt sich eine Rückverfolgung über alle Methodenartefakte, Komponenten, Bauelemente und Schnittstellen/Wechselwirkungen erreichen.

In der Wirkkette lassen sich sowohl gewollte wie ungewollte als auch Misuse Wechselwirkungen dokumentieren. Diese Wechselwirkungen werden mit weiteren Attributen versehen, die zu einer optischen Differenzierbarkeit führen. Diese Attribute sind im Einzelnen: Mechanik, thermische Energie, Information, Stoff/Material, Elektrik, HV-Energie/-leistung, Optik/visuelle Wirkung und Akustik/Luftschall. Anschließend werden die Wechselwirkungen mit einem Text versehen, der diese eindeutig beschreibt. Beispielsweise führt die gewollte Wechselwirkung „Befestigung“ eines Objektes bei mechanischer Anregung an dem befestigten Objekt zu der ungewollten Wechselwirkung „Vibrationsübertragung“. Das so entstandene Systemmodell kann in der verteilten Entwicklung sowohl auf horizontaler als auch auf vertikaler Ebene durch die Entwicklungspartner weiter ausdetailliert werden.

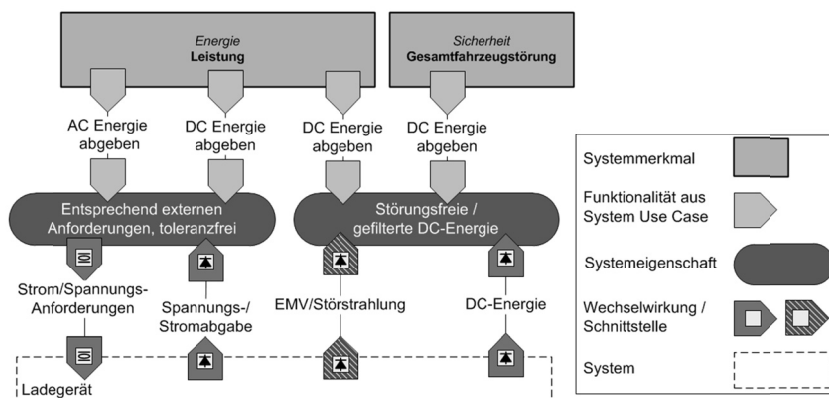


Bild 6: Ausschnitt aus dem Systemmodell/Wirkkette des Ladegerätes

Der Aufbau und die Dekomposition der Schnittstellen werden durch eine eindeutige Vorgehensbeschreibung unterstützt. Nach Erarbeitung des Systemmodells stehen formale Gütekriterien zur Überprüfung des Systemmodells zur Verfügung. Das Systemmodell wurde in einer VW-Norm genormt. Diese VW-Norm steht nicht nur dem gesamten VW-Konzern zur Verfügung, sondern auch sämtlichen Lieferanten. Hierdurch wird ein firmenübergreifender Einsatz ermöglicht.

4.1 Validierung Objektfunktionen

Die Methode Validierung Objektfunktionen dient der Überprüfung, ob sämtliche Funktionalitäten, die in den Use Cases erarbeitet wurden, sich im Systemmodell wiederfinden. Da die System Use Cases am Anfang des Gesamtvorgehens stehen, ist hierdurch ebenfalls eine Überprüfung der Durchführung der Gesamtmethode möglich.

Die Methode basiert auf der Arbeit von KAISER [Kai13]. Durch Identifizierung der Hauptschnittstellen eines Systemelements, kann die Funktion dieses Elementes eindeutig definiert werden. Die Funktionen der Elemente werden aus den Wechselwirkungsattributen der Hauptschnittstellen abgeleitet. Hierzu steht ein Katalog mit allen möglichen Kombinationen der Ein- und Ausgangsschnittstellen zur Verfügung. In diesem Katalog ist zu jeder Kombination die realisierte Funktion hinterlegt. Beispielsweise ist die abgeleitete Funktion eines Elementes mit der Eingangsschnittstelle „Elektrik / Elektronik“ und der Ausgangsschnittstelle „Optik/ visuelle Wirkung“ eine „Optische Rückmeldung“ an einen Benutzer. Nach Ausleiten der Funktionen der Elemente in der Wirkkette findet ein Abgleich mit den System Use Cases statt. Gibt es zu jeder Funktion eines Elementes genau eine Funktion in den Use Cases, so ist sichergestellt, dass sowohl die Use Cases, als auch die Wirkkette vollständig sind.

Im Systemmodell/Wirkkette des Ladegerätes befindet sich beispielsweise eine Leuchtdiode zur Ladestatusanzeige. Diese besitzt als Eingang eine elektrische Schnittstelle und als Ausgang eine visuelle Wirkung. Die Abgeleitete Funktion „Optische Rückmeldung“ entspricht der Funktion „Ladevorgang optisch signalisieren“ in den Use Cases.

4.2 Spezifikationswürdigkeit

Mittels des Gütekriteriums „Spezifikationswürdigkeit“ wird überprüft, ob aus einer Wechselwirkung/Schnittstelle oder Eigenschaft eine Anforderung abgeleitet werden kann. Es wird folglich überprüft, ob eine Wechselwirkung/Schnittstelle oder Eigenschaft die benötigte Voraussetzung erfüllt, um eine qualitativ hochwertige Anforderung abzuleiten

Die Spezifikationswürdigkeit von Eigenschaften ist davon Abhängig, ob die richtige Granularität der Eigenschaft gewählt wurde. Grobgranulare Eigenschaften lassen sich aufgrund ihres unspezifischen Charakters schwer oder gar nicht in Form von physikalischen Wechselwirkungen und Schnittstellen darstellen. Mittels eines eindeutigen Vor-

gehens wird hinterfragt, ob die Eigenschaft aufgrund von externen Anforderungen in Form von Normen oder Gesetzen besteht. Ist dies der Fall, so müssen Schnittstelle und Wechselwirkungen ausgeleitet werden. Desweiteren werden die Zustände und die Randbedingungen der Eigenschaft hinterfragt. Sind diese nicht eindeutig definierbar oder gibt es mehr als drei mögliche Zustände und Randbedingungen, so muss die Eigenschaft durch Funktionalitäten auf einer tieferen Ebene der Use Cases gebildet werden.

Die Systemeigenschaften des Ladegerätes „toleranzfreie Energieabgabe, gemäß externen Anforderungen“ entspricht dem Gütekriterium der Spezifikationswürdigkeit. Im Falle der Energieabgabe eines Fahrzeuges an eine Infrastruktur sind verschiedenste Gesetze und Normen zu beachten. Das Ladegerät erfasst diese und regelt die Energieabgabe dahingehend ein, dass diese toleranzfrei den externen Anforderungen erfolgt.

Die Spezifikationswürdigkeit von Schnittstellen und Wechselwirkungen richtet sich neben der Semantik auch an die Darstellung innerhalb des Systemmodells. Im Sinne der Übersichtlichkeit werden Schnittstellen der gleichen Art häufig zu einer Schnittstelle zusammengefasst. Hierdurch können wesentliche System- oder Komponentenschnittstellen aus dem Fokus der einzelnen Entwickler geraten. Dies wird durch eine Evaluierung der Schnittstellen vermieden.

5 Ausleiten von Anforderungen

Die Anforderungen an ein System lassen sich direkt aus den Schnittstellen und Wechselwirkungen der Wirkkette ausleiten. Die Schnittstellen und Wechselwirkungen erfüllen sämtliche Vorgaben an Anforderungen. Diese sind unter anderem: Eineindeutigkeit, Konsistenz, Vollständigkeit, Nachverfolgbarkeit und Testbarkeit [IEEE1233]. Da durch Anwendung des Gesamtverfahrens garantiert ist, dass sämtliche Bedarfe der Stakeholder, sämtliche benötigten Funktionalitäten und alle gewünschten Eigenschaften eines mechatronischen Systems in den Schnittstellen und Wechselwirkungen repräsentiert sind, sind die Anforderungen vollständig beschrieben. Die Anforderungen können anschließend in einem Lastenheft, welches an die Lieferanten übergeben wird, dokumentiert werden.

6 Reifegradmanagement

Wurden alle Anforderungen an ein mechatronisches System erarbeitet, werden diese in ein Reifegradmanagement überführt. Hierdurch ist es möglich, frühzeitig in der Systementwicklung Risiken zu identifizieren. Aufgrund der verteilten Entwicklung mit mehreren Lieferanten und Standorten ist ein zentrales Reifegradmanagement bezüglich der Anforderungen und deren Umsetzung unabdingbar.

Das Reifegradmanagement wird mittels sogenannter Reifegradmeilensteine (RM) durchgeführt. Diese Reifegradmeilensteine wurden so gewählt, dass sie alle relevanten Schritte in der Entwicklung mechatronischer Systeme umfassen. Mittels des Reifegradmeilensteins 1 wird ermittelt, ob alle Schnittstellen im Systemmodell eine Verantwortlichkeit besitzen und spezifiziert sind. Die Dokumentation in einem Lastenheft und die Bestätigung in einem Pflichtenheft eines Zulieferers wird mittels des RM 2 bewertet. RM 3 bewertet die Dekomposition des Systemmodells und der Anforderungen. Die Validierung der Spezifikation einer Anforderung oder Schnittstelle wird zum RM 4 abgeprüft. Reifegradmeilenstein 5 und 6 bewerten die Integration und die Absicherung von Schnittstellen in einer Komponente und anschließend im Gesamtsystem. Der Gesamtsystemreifegrad setzt sich hierbei aus den Reifegraden aller Schnittstellen zusammen.

7 Resümee und Ausblick

Das hier vorgestellte Gesamtverfahren erlaubt das strukturierte Erarbeiten und Erfassen von Anforderungen an ein komplexes mechatronisches System im praktischen Einsatz bei einem Automobilhersteller. Es wird der Herausforderung der verteilten Entwicklung gerecht, da die Einzelmethoden in verteilten Entwicklungsteams erarbeitet und anschließend die Ergebnisse zusammengefügt werden können. Die Einzelmethoden eignen sich durch ihre klare Vorgehensbeschreibung und der Gütebewertung auch für Entwickler ohne ausgeprägte Systems-Engineering-Erfahrung. Durch Anwendung des Gesamtverfahrens werden sämtliche Sichtweisen auf ein System und die Anforderungen aller relevanten Stakeholder mit erfasst. Das Gesamtverfahren ist hierbei an den Bedarfen der Entwicklung orientiert und sehr effizient. Durch das Reifegradmanagement wird in den folgenden Entwicklungsschritten sichergestellt, dass alle Anforderungen über alle Entwicklungspartner hinweg richtig kommuniziert und umgesetzt wurden. Es entsteht die Möglichkeit Risiken frühzeitig zu identifizieren und die Reife des Systems jederzeit zu erfassen.

Das Vorgehen wurde erfolgreich bei der AUDI AG in mehreren Projekten eingesetzt und bestätigt. Es eignet sich ebenfalls für andere Industriezweige, wie den Maschinen- und Anlagenbau. Lediglich die Merkmale und deren Verknüpfung in der Methode Elementklassifikation müssen für einen anderen Industriezweig angepasst werden.

Literatur

- [All07a] ALLMANN, C.: Anforderungen auf Kundenfunktionsebene in der Automobilindustrie. In: Workshop: Wie viel Requirements Engineering steckt im Software Engineering? Im Rahmen der SE2007, 2007
- [All08] ALLMANN, C.: Situations- und szenariobasiertes Anforderungsmanagement in der automatisierten Elektronikentwicklung. Universität Hannover. 2008

- [Alm06] ALMEFELT, LARS; BERGLUND, FREDRIK; NILSSON, PATRIK; MALMQVIST, JOHAN: Requirements management in practice: findings from an empirical study in the automotive industry, Research in Engineering Design 17/2006, 2006
- [FHM14] FOCKEL, MARKUS; HOLTSMANN, JÖRG; MEYER, MATTHIAS: Mit Satzmustern hochwertige Anforderungsdokumente effizient erstellen. In: OBJEKTSpektrum RE/2014 (Online Themenspecial Requirements Engineering), 2014
- [IEEE1233] IEEE Guide for Developing System Requirements Specifications, 1998
- [INC07] INCOSE: Systems Engineering Vision 2020, INCOSE-TP-2004-004-02, Version 2.03, September 2007
- [Jar99] JARKE, MATTHIAS: CREWS: Towards Systematic Usage of Scenarios, Use Cases and Scenes. In: 4. Internationale Tagung Wirtschaftsinformatik (WI'99), Seiten 469-486, 1999.
- [Kai13] KAISER, L.: Rahmenwerk zur Modellierung einer plausiblen Systemstruktur mechatronischer Systeme. Universität Paderborn. 2013
- [Lam04] LAMSWEERDE, AXEL VAN: Goal-Oriented Requirements Engineering: A Roundtrip from Research to Practice. In: Proceedings of 12th IEEE Joint International Requirements Engineering Conference (RE04), S. 4-8, 2004.
- [PB07] PAHL, G.; BEITZ, W.: Konstruktionslehre – Grundlagen erfolgreicher Produktentwicklung; Methoden und Anwendung. Springer, Berlin, 7. Auflage, 2007.
- [She96] SHEARED, S. A.: Twelve Systems Engineering Roles. In: INCOSE (Hrsg.): Proceedings of the 6th INCOSE Annual International Symposium, Boston, Massachusetts, USA, 1996
- [VDI2206] Entwicklungsmethodik für mechatronische Systeme, 2004
- [VDI2222] Konstruktionsmethodik – Methodisches Entwickeln von Lösungsprinzipien, 1997
- [Wie95] WIERINGA, ROEL: An Introduction to Requirements Traceability. Technischer Bericht, Faculty of Mathematics and Computer Science, University of Vrije, Amsterdam, 1995.
- [WW02] WEBER, M. ; WEISBROD, J.: Requirements Engineering in automotive development: Experience and challenges. In: Proceedings of the IEEE Joint International Conference on Requirements Engineering (RE'02), 2002

Autoren

Dipl.-Ing. Nicholas Schmitt studierte Mechatronik an der Friedrich-Alexander-Universität Erlangen-Nürnberg. Seit 2012 promoviert er bei der AUDI AG in Kooperation mit dem Fraunhofer IPT in Paderborn und dem Heinz Nixdorf Institut der Universität Paderborn.

Dr.-Ing. Roman Dumitrescu studierte Mechatronik an der Friedrich-Alexander-Universität Erlangen-Nürnberg. Im Anschluss war er wissenschaftlicher Mitarbeiter am Lehrstuhl für Produktentstehung am Heinz Nixdorf Institut der Universität Paderborn. Unter der Leitung von Professor Jürgen Gausemeier promovierte er 2010 im Bereich »Entwicklungsmethodik für fortgeschrittene mechatronische Systeme«. Seit dem Start der Fraunhofer-Projektgruppe Entwurfstechnik Mechatronik in Paderborn im März 2011 leitet er in dieser die Abteilung »Produktentstehung«.

Erweiterung der Entwurfsmethodik CONSENS um absichernde Maßnahmen zur Risikominimierung in Produkten

Dr. Martin Schäfer, Dr. Guido Stollt

Smart Mechatronics GmbH

Kronenburgallee 2 , 44141 Dortmund

Tel. +49 (0) 231 / 84 16 85 100, Fax. +49 (0) 231 / 84 16 85 299

E-Mail: martin.schaefer@smartmechatronics.de

Dr. Alexander Nyßen

itemis AG

Am Brambusch 15-24, 44536 Lünen

Tel. +49 (0) 231 / 98 60 606, +49 (0) 231/ 98 60 211

E-Mail: alexander.nyssen@itemis.de

Dr. Rafał Dorociak

HELLA KGaA Hueck & Co.

Beckumer Str. 130, 59552 Lippstadt

Tel. +49 (0) 2941 / 380

E-Mail: Rafal.Dorociak@hella.com

Zusammenfassung

Miniaturisierung, Bedienerfreundlichkeit, Energieeffizienz und Vernetzbarkeit sind nur einige Beispiele an Anforderungen, die an aktuelle Produkte gestellt werden.

Dies erfordert zunehmend die Einbeziehung von konstruktiven, elektronischen und softwaretechnischen Komponenten in ein Produktkonzept. Gleichzeitig steigen die Ansprüche an die Zuverlässigkeit und Sicherheit von Produkten. Dies spiegelt sich unter anderem in der Zunahme von Richtlinien, die bei der Produktgestaltung zu berücksichtigen sind. Die Maschinenrichtlinie [EG06] und verschiedene, branchenspezifische Normen zur funktionalen Sicherheit [ISO11], [IEC10] sind Beispiele hierfür.

Die zunehmenden Anforderungen an die funktionale Sicherheit aber auch allgemeine Qualitätsanforderungen erfordern eine frühzeitige Einbeziehung von potentiellen Risiken in die Produktgestaltung.

Mit CONSENS ist eine bewährte Entwurfsmethodik für die Produktgestaltung in der Konzeptphase gegeben [GLL12], [Fra06]. Ihre systematische Vorgehensweise ermöglicht über die Festlegung von Produktfunktionen und Wirkstrukturen die Erstellung eines Produktkonzeptes mit einem hohen Reifegrad zu einer frühen Entwicklungsphase.

Aus diesem Konzept geht durch Ausarbeitung der konstruktiven, elektronischen und softwaretechnischen Komponenten das Gesamtprodukt hervor.

Eine explizite Risikobetrachtung ist in CONSENS zur Zeit nicht enthalten. Die systematische Vorgehensweise ermöglicht jedoch die Erweiterung der Entwurfsmethodik um etablierte Qualitäts- und Sicherheitsmethoden. Eine allgemeine Gefahren- und Risikoanalyse, aber auch spezialisierte Methoden wie Fehlermöglichkeits- und Einflussanalyse (FMEA) oder Fehlerbaumanalyse (FTA) lassen sich zielgerichtet in die Erstellung eines Produktkonzeptes integrieren. Verborgene Produktrisiken werden damit früh identifiziert und können durch Erweiterung der Anforderungen an die Produktkomponenten minimiert werden. Weiterhin ist gewährleistet, dass im Sinne der Nachverfolgbarkeit eine durchgängige Dokumentation der Behandlung von Produktrisiken ab einem frühen Entwicklungsstand besteht.

Schlüsselworte

Entwurfsmethodik, funktionale Sicherheit, Datensicherheit, FMEA, FTA, Absicherungsmethodik, Produktrisiken

Applying standardized Safety Engineering Measures in Combination with the Design Methodology CONSENS

Abstract

Miniaturization, User Friendliness, Energy Efficiency and Interconnectedness are only some of many example of requirements on the products of today.

In order to meet this requirements, an increasing integration of mechanical, electronic and software components in the product concept is essential. This leads also to increasing requirements with regards to reliability and safety of such products. An indicator for that is the increasing number of guidelines and standards that need to be considered within the product development. The machinery directive [EG06] and different, industry specific norms for functional safety are examples for this [ISO11], [IEC10]

The increasing requirements on the functional safety as well as general quality requirements require an early consideration of potential risk into the conception and development of the product.

CONSENS is an established design methodology for the conception and development of products in the phase conceptual design [GLL12], [Fra06]. Its systematic procedure supports the definition of product functions and structure and thus the creation of product concepts with a high maturity to an early stage of the product development. Based on this product concept, a concretization into mechanical, electronic and software components takes place which together form the overall product.

An explicit risk evaluation is currently not a constituent part of CONSENS. The systematic procedure of CONSENS can, however, be extended with established quality and safety methods. A general hazard and risk analysis as well as specialized methods such as the Failure Mode and Effects Analysis (FMEA) and Fault Tree Analysis (FTA) can be integrated and used to create the product concept. As a consequence, hidden product risk are identified from early on and can be mitigated by incorporation of additional requirements on the product components. In addition, the overall documentation of handling of product risks at early development stages in sense of traceability is supported.

Keywords

design methodology, functional safety, data safety, FMEA, FTA, assurance methodology, product risks

1 Sicherheit in der Produktentwicklung

Das V-Modell illustriert den Verlauf einer Produktentwicklung über die verschiedenen Abstraktionsebenen eines Produktes. In der Konzeptphase liegt der Schwerpunkt der Betrachtung beim Produktumfeld und bei den Produktfunktionen. Über den linken Zweig verschiebt sich die Betrachtung auf Produkt-Komponenten und deren Schnittstellen mit zunehmender Granularität, bis in die Implementierungsphase, welche die Codierung von Software-Modulen oder die Auslegung elektronischer oder mechanischer Komponenten beschreibt. Der rechte Ast des V-Modells beinhaltet die schrittweise Integration und den Test der einzelnen Module oder Komponenten bis zum Gesamtprodukt.

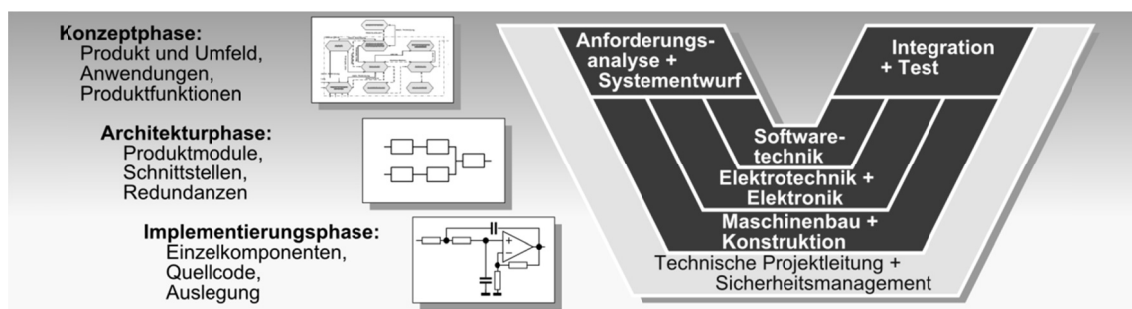


Bild 1: V-Modell der Produktentwicklung

Jeder dieser Phasen lassen sich auch sicherheitsbezogene Tätigkeiten zuordnen. Dabei spiegelt sich der über das V-Modell variierende Abstraktionsgrad wieder. In der Konzeptphase lassen sich Risiken aus dem Produkt in seinem Umfeld und seiner Benutzung identifizieren. Dazu sind neben dem erwünschten Produktverhalten auch Folgen möglicher Produktfehlfunktionen durch Fehlbedienung, Missbrauch oder Ausfälle einzelner Komponenten [Bir10] zu berücksichtigen. Aus den identifizierten Gefährdungspotentialen leiten sich Anforderungen für die folgenden Phasen ab. Dies können Anforderungen an die Produktstruktur sein, z.B. die Forderung nach Integration von Sicherheitskomponenten oder auch an den Entwicklungsprozess, z.B. Art und Tiefe von Dokumentation und Verifikation. Mit den folgenden Entwicklungsphasen verschiebt sich auch bei den sicherheitsbezogenen Tätigkeiten der Fokus vom Gesamtprodukt in seinem Umfeld auf die nachweislich richtige Ausarbeitung der Produktkomponenten. Dies führt bis zu Implementierungsdetails, wie die besondere Vermeidung von Puffer-Überläufen in der Softwareentwicklung oder die besondere Auslegung von Bauteilen in der Elektronikentwicklung.

Ein wichtiger Punkt ist die Nachverfolgbarkeit (traceability): Die Kette von Anforderungen, die sich aus Gründen der Sicherheit ergeben und den daraus folgenden Arbeitsprodukten muss über den gesamten Entwicklungsprozess nachvollzogen werden können.

1.1 Forderung nach Sicherheitsbetrachtungen in der Produkt-Konzeptphase

Grundlegende Sicherheitsaspekte sind in der frühen Konzeptphase der Produktentwicklung zu berücksichtigen [DG12], [DIK10]. Sicherheitsrisiken, die sich aus einem Produktumfeld oder einer Produktverwendung ergeben und nicht im Produktkonzept berücksichtigt wurden, lassen sich in späteren Entwicklungsphasen nur mit sehr hohem Aufwand korrigieren.

Analysierbare Modelle der Produktkonzeption sind eine wesentliche Voraussetzung für den frühzeitigen Einsatz von Methoden zur Absicherung der Sicherheit eines mechatronischen Produkts. Es bedarf einer modellbasierten Spezifikation der Produktkonzeption, welche den grundsätzlichen Aufbau, die Wirkungsweise und das Verhalten des Produkts unter besonderer Berücksichtigung der Sicherheit disziplinübergreifend beschreibt. Eine derartige Spezifikation der Produktkonzeption ermöglicht erste grundlegende Aussagen zur Sicherheit des Produkts. Insbesondere wird mit der Spezifikation der Produktkonzeption eine Basis für die wirksame Kooperation und Kommunikation von Experten aus den involvierten Fachdisziplinen geschaffen. Hierbei gilt es auf etablierten Beschreibungsmitteln und Analysemethoden aufzubauen und diese zu erweitern.

Analysierbare Modelle der Produktkonzeption, die in elektronischer Form vorliegen, erlauben ferner die eindeutige, bidirektionale Verknüpfung von einem Modell zu nachfolgenden Arbeitsprodukten wie Anforderungen, Modulentwürfen, Implementierungen oder Testspezifikationen. Damit ist die durchgehende Verfolgbarkeit („traceability“) von Sicherheitsaspekten von der Analyse über die Implementierung bis zur Validierung sichergestellt.

2 Identifikation von Risiken

Die erste Aufgabe bei der Risikobetrachtung ist die Identifikation von Gefährdungsszenarien, die vom Produkt ausgehen. Hierbei lassen sich zwei Sichtweisen unterscheiden.

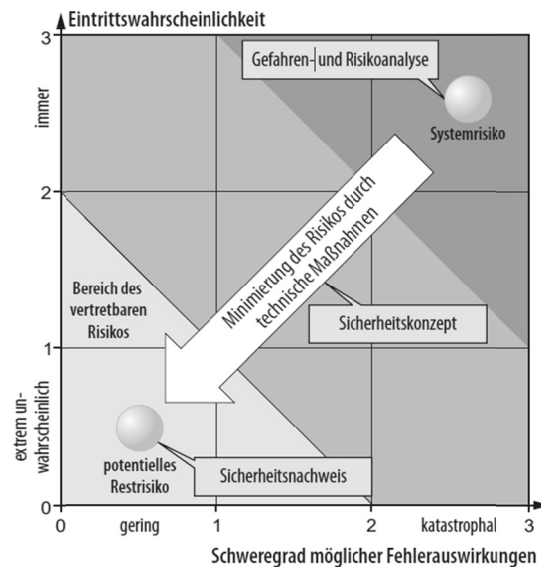


Bild 2: Risikominimierung [PH13]

Einerseits können die Anwendungsszenarien oder Umfeldbetrachtungen bereits offensichtliche Gefährdungssituationen aufzeigen. In dem Fall ist das Systemdesign des Produktes so auszulegen, dass das Auftreten dieser Situationen hinreichend unwahrscheinlich ist, d.h. dass die jeweiligen Risiken minimiert sind. Ausgehend von jedem identifiziertem Gefahrenfall ist der Beitrag der einzelnen Komponenten des Systementwurfs zu ermitteln, um daraus weitere Anforderungen an diese Komponenten abzuleiten.

Andererseits können vom Systementwurf bisher nicht identifizierte Gefährdungssituationen ausgehen. Um diese aufzudecken, ist eine Analyse des Systementwurfs notwendig.

Aus identifizierten Risiken leiten sich Anforderungen an das Produktkonzept, die Implementierung und den Entwicklungsprozess ab. In jedem Fall Risiken zu auf ein tolerierbares Maß zu minimieren, indem entweder die Eintrittswahrscheinlichkeiten und/oder die Auswirkungen verringert werden.

2.1 Klassifikation von Risiken

Sicherheitsrisiken in der Produktentwicklung sind entsprechend ihrer Schwere zu klassifizieren. Die ISO 26262 [ISO11] ordnet Risiken in die Klassen ASIL A bis ASIL D ein, in der IEC 61508 [IEC10] sind es die Klassen SIL 1 bis SIL 4, andere branchenbezogene Normen haben weitere Bezeichnungen. Abhängig von Produktbranche und geltenden Normen unterscheiden sich die Nomenklaturen und Vorgaben zur Einordnung von Sicherheitsrisiken.

Vergleichbar ist jedoch die generelle Vorgehensweise zur Klassifikation. Ein Sicherheitsrisiko wiegt umso schwerer,

- je höher die Auswirkungen im Sinne von Verletzungen oder Todesfällen ist;
- je wahrscheinlicher die Risikosituation ist;

- je geringer die Möglichkeit auf eine Gefahr zu reagieren ist.

Zur Beschreibung von identifizierten Risiken gehören daher die Informationen, „unter welchen Umständen“ diese Risiken eintreten können, „welche Auswirkungen“ sie haben, und „welche Abwehrmöglichkeiten“ für den Anwender im Risikofall vorhanden sind¹.

2.2 Etablierte Analysemethoden

Zur Identifikation von Risiken in einem Produktkonzept sind verschiedene aus der Qualitätssicherung stammende Methoden anwendbar [Eri05], [Bir10]. Alle Methoden basieren auf strukturierten Vorgehensweisen, um die Nachvollziehbarkeit und Vollständigkeit der Analysen zu gewährleisten. Weiterhin sehen die meisten Methoden eine Bearbeitung im Team vor, um fachdisziplinübergreifende Inhalte zu berücksichtigen.

Eine verbreitete, induktive Methode ist die FMEA (Fehlermöglichkeits- und Einflussanalyse, engl. failure mode and effect analysis). Mit der FMEA werden für jedes Systemelement (ggfs. auch Prozesse oder Funktionen) mögliche Ausfallarten sowie deren potentielle Ursachen und Auswirkungen in einer Tabelle beschrieben. Zudem wird eine Risikobewertung mithilfe der Risikoprioritätszahl (RPZ) oder Risikomatrizen durchgeführt und es werden Gegenmaßnahmen definiert. Die Analyse innerhalb einer FMEA geht von einer Systemstruktur und deren Funktionen bzw. Prozessen aus und untersucht die möglichen Folgen auf der Gesamtsystemebene.

Eine klassische deduktive Methode zur Analyse bereits identifizierter Risiken ist die FTA (Fehlerbaumanalyse, engl. fault tree analysis). Mit einer FTA wird ausgehend von einem zu untersuchenden Hauptereignis unter Berücksichtigung von UND- und ODER-Verknüpfungen systematisch auf deren potentiellen Ursachen geschlossen. Die Ergebnisse werden in einem Fehlerbaum festgehalten. Qualitatives Analyseergebnis einer FTA sind die notwendigen Kombinationen von Ursachen zum Erreichen des Hauptereignisses (Minimalschnitte). Wenn für die Ursachen Ausfallwahrscheinlichkeiten angegeben werden können, ist als quantitatives Analyseergebnis die Berechnung der Eintrittswahrscheinlichkeit des Hauptereignisses möglich.

Weitere Methoden wie PHA (engl.: preliminary hazard analysis), SHA (engl.: system hazard analysis), HAZOP (engl.: hazard and operability study), ETA (engl.: event tree analysis), Markov-Ketten und andere können je nach Produktart, Risikoumfang, und Vorgaben der anzuwendenden Normen zum Einsatz kommen.

¹ Die Begriffe sind an dieser Stelle absichtlich umgangssprachlich gehalten, um einen Bezug zur Nomenklatur einer von mehreren möglichen anzuwendenden Norm zu vermeiden.

3 Integration der Sicherheitsbetrachtung bei der Produktkonzeption

3.1 Ganzheitliche fachdisziplinübergreifende Beschreibung der Produktkonzeption mit CONSENS

CONSENS (CONceptual design Specification technique for the Engineering of mechatronic Systems) ist eine am Heinz Nixdorf Institut erarbeitete, fachdisziplinübergreifende MBSE-Spezifikationstechnik [GLL12]. Neben einer Modellierungssprache (proprietär, sowie über ein Profil als Erweiterung der SysML verfügbar) legt CONSENS auch eine Methodik fest, die sich im Wesentlichen an etablierten Methoden des Maschinenbaus orientiert.

Ein Systemmodell wird nach CONSENS durch unterschiedliche Partialmodelle beschrieben: Umfeld, Anwendungsszenarien, Anforderungen, Funktionen, Wirkstruktur (mit Verhalten) und Baustruktur/Gestalt.

Zu Beginn wird im Umfeld die Systemgrenze festgelegt (siehe Bild 3). Hierbei werden die Elemente innerhalb des Umfelds identifiziert, die mit dem zu entwickelnden System in Beziehung stehen. Dabei werden insbesondere Wechselwirkungen (Flüsse) zwischen diesen Elementen und dem System analysiert und in Stoff-, Energie-, sowie Materialflüsse klassifiziert. Dabei fließen in die Betrachtung auch ungewollte Wechselwirkungen ein, die als Störflüsse modelliert werden.

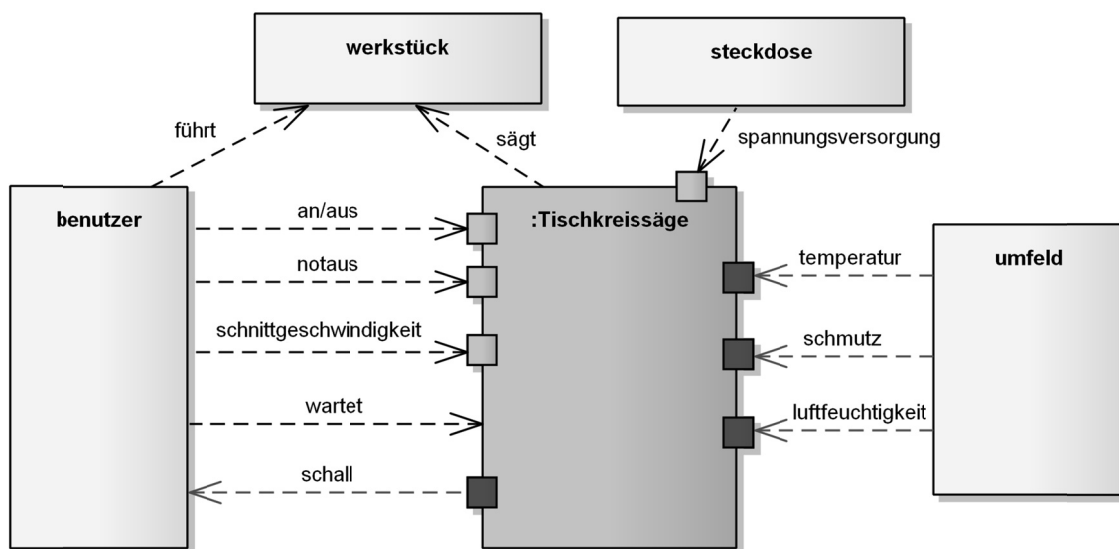


Bild 3: Umfeldmodell nach CONSENS (SysML-Profil)

Anschließend erfolgt, wie in Bild 4 dargestellt, eine Analyse der verschiedenen Situationen, in denen das System zum Einsatz kommt. Dabei werden in Form sogenannter Anwendungsszenarien die Situation sowie das erwartete Systemverhalten beschrieben.

Hieraus ergeben sich unter Umständen neue Anforderungen, die die Menge der initial vorhanden Anforderungen ergänzen.

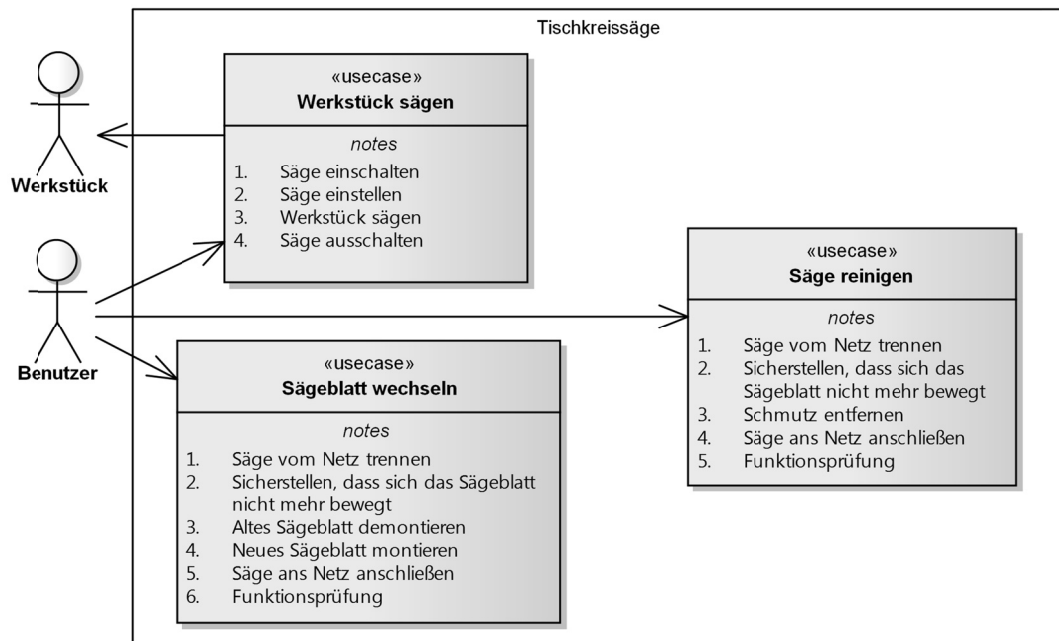


Bild 4: Anwendungsszenarien nach CONSENS

Die im Rahmen der Umfeld- und Anwendungsszenario-Modellierung ergänzten und verfeinerten Anforderungen werden in funktionale und nicht-funktionale Anforderungen klassifiziert und in Funktionen überführt. Eine Funktion ist dabei ein „allgemeiner und gewollter Zusammenhang zwischen Eingang und Ausgang eines Systems mit dem Ziel, eine Aufgabe zu erfüllen“ [PBF+06].

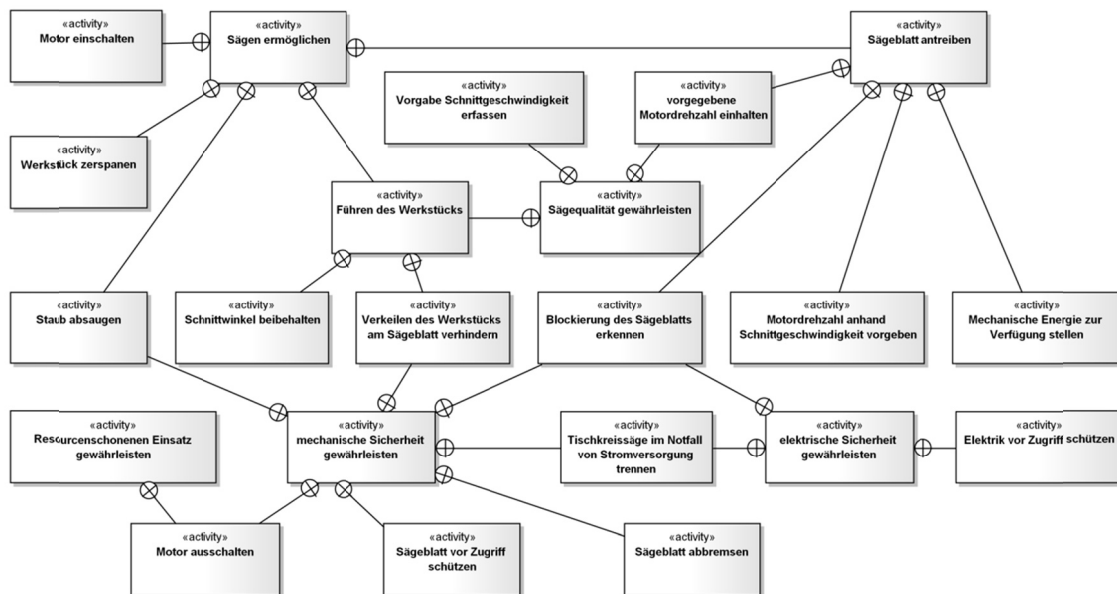


Bild 5: Funktionsnetz nach CONSENS

Um Zusammenhänge zwischen Funktionen zu erfassen, werden die Funktionen in einer Funktionshierarchie bzw. einem Funktionsnetz² organisiert (siehe Bild 5). Dabei wird die zu leistende Funktionalität so lange heruntergebrochen, bis Wirkprinzipien bzw. Lösungsmuster zur Umsetzung der jeweiligen Funktionen identifiziert werden können.

Basierend darauf wird eine Wirkstruktur erarbeitet, in welcher der grundsätzliche Aufbau und die Wirkungsweise des Systems spezifiziert werden. Dabei wird das System in Systemelemente dekomponiert, welche die zuvor definierten Funktionen umsetzen.

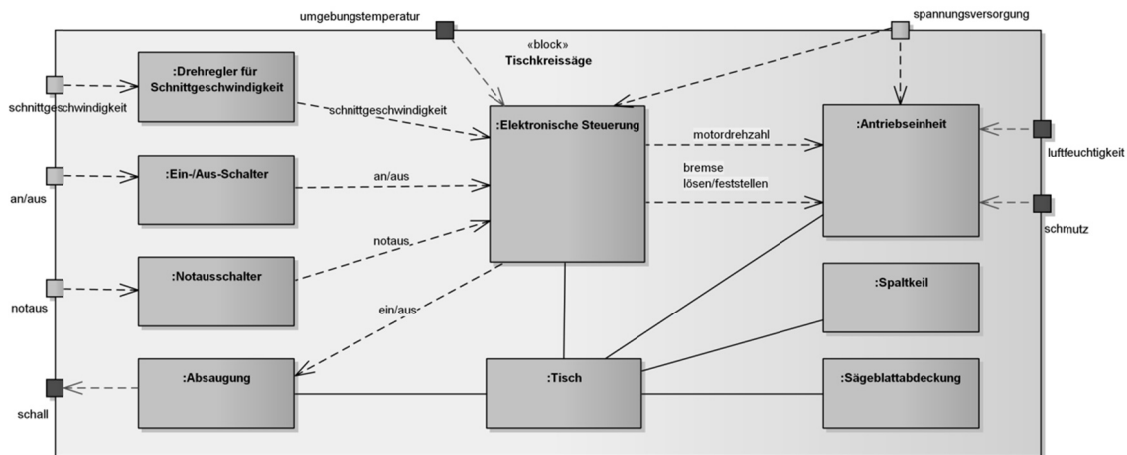


Bild 6: Wirkstruktur nach CONSENS

Ähnlich wie zwischen dem System und Umfeldelementen, werden hierbei – wie Bild 6 in ersichtlich – neben der Hierarchiebeziehung auch Wechselwirkungen (Stoff-, Energie-, und Materialflüsse) zwischen den Systemelementen identifiziert.

3.2 Erweiterung von CONSENS um ein Partialmodell „Gefahren“

Durch die Systematik und Durchgängigkeit der CONSENS-Methode, lässt sich eine Betrachtung von Gefahren und Risiken einfach und konsistent integrieren. Dabei kann bei der Modellierung des Umfelds auch die Betrachtung möglicher Gefahrenauswirkungen (engl. hazard effects) erfolgen, wenn offensichtliche Gefahren bereits bekannt sind.

Sind die Elemente des Umfelds und ihre Wechselwirkungen mit dem System bekannt, so lassen sich insbesondere die ungewünschten Wechselwirkungen dahingehend untersuchen, in wieweit Umfeldelemente durch Gefahrenauswirkungen Schaden nehmen könnten. Diese können im Umfeld ergänzt und die Schwere der Auswirkung (engl. severity) kann jeweils bewertet werden.

² CONSENS definiert streng genommen das Modellieren einer Funktionshierarchie. Ein Funktionsnetz scheint den Autoren aber im Allgemeinen angemessener.

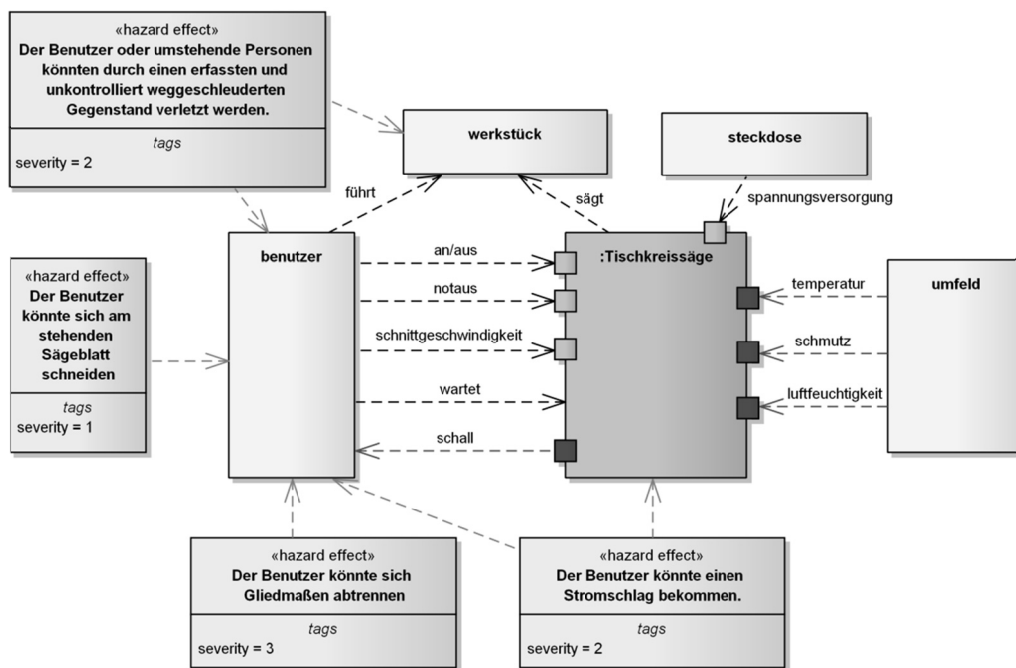


Bild 7: Umfeldmodell, ergänzt um Gefahrene Auswirkungen

In den Anwendungsszenarien lassen sich auf gleicher Basis Gefährdungssituationen (engl. hazardous situation) ergänzen, in denen mögliche Gefahren eintreten können. Wie die Gefahrene Auswirkungen werden auch die Gefährdungssituationen bewertet, hier aber anhand der Wahrscheinlichkeit ihres Eintretens (engl. probability), die sich aus den Anwendungsszenarien ableitet.

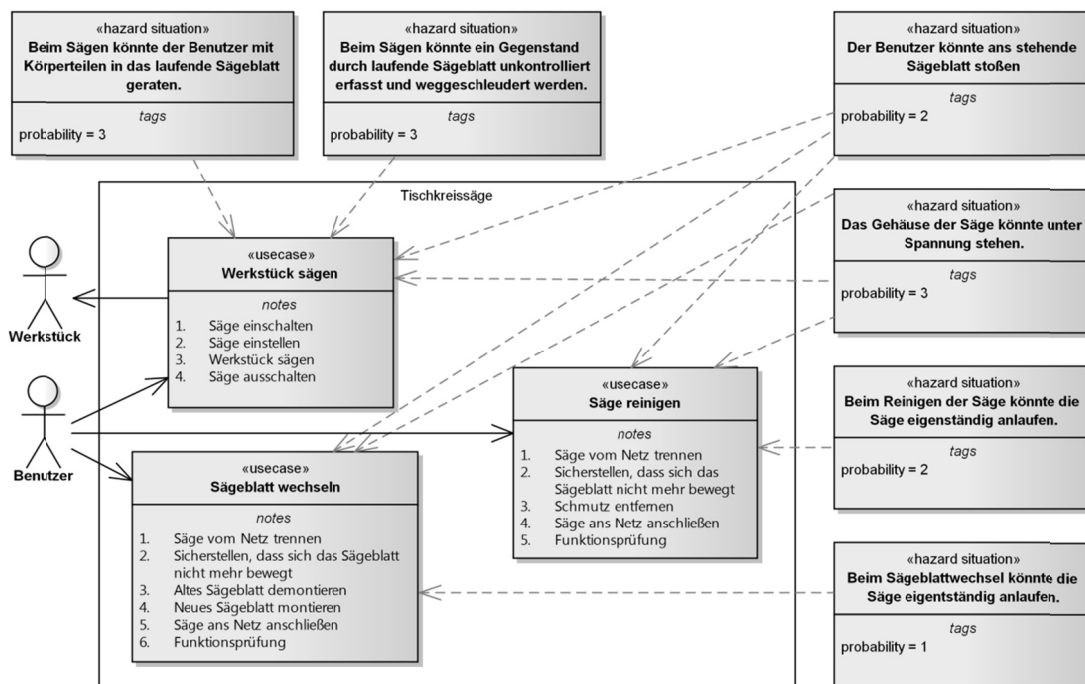


Bild 8: Anwendungsszenarien, ergänzt um Gefährdungssituationen

Das systematische Erfassen von Gefahren erfolgt schließlich, indem alle Kombinationen aus Gefährdungssituationen und möglicher Gefahreauswirkungen betrachtet werden. Dabei verknüpft eine Gefahr genau eine Gefährdungssituation mit einer Gefahreauswirkung.

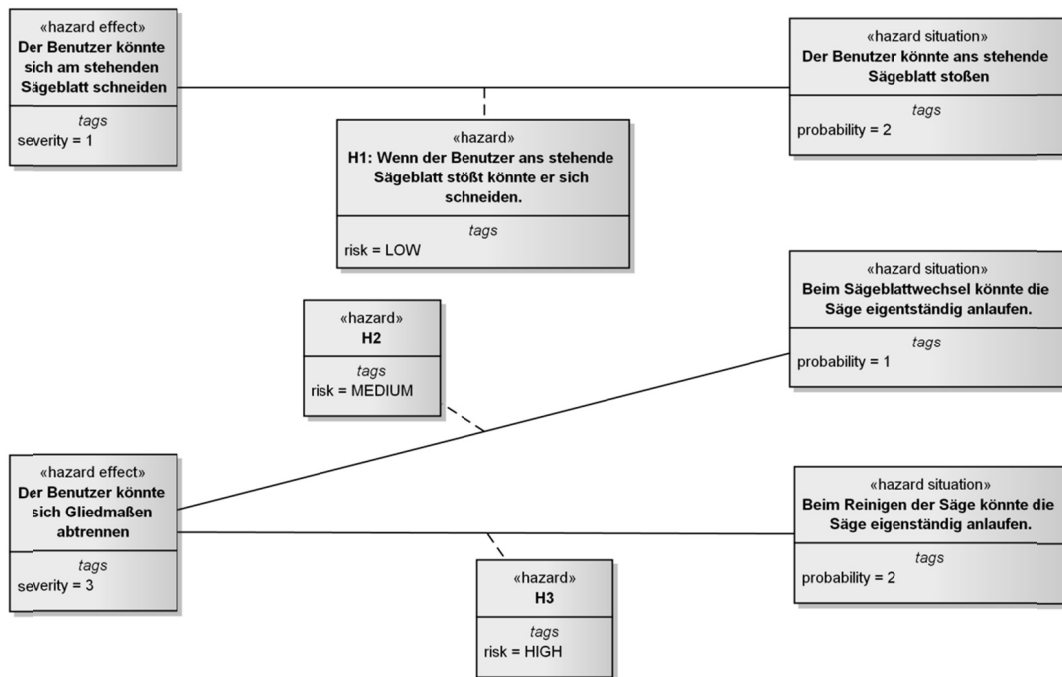


Bild 9: Gefahrenmodell als Ergänzung zu CONSENS

Das mit der Gefahr verbundene Risiko (engl. risk) lässt sich anhand der Wahrscheinlichkeit des Eintretens der Situation, vorhandener Abwehrmöglichkeiten, sowie der Schwere der Auswirkung bewerten. Basierend darauf kann eine Klassifikation abhängig von der zu verwendenden Normen erfolgen (beispielsweise SIL-Level bei IEC61508).

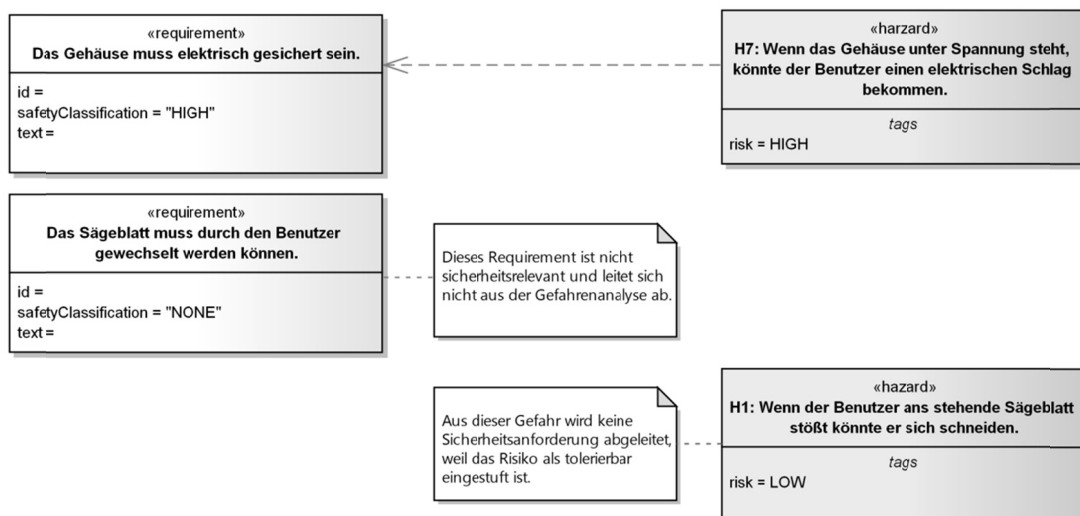


Bild 10: Sicherheitsanforderungen, abgeleitet aus Gefahren

Schließlich können aus identifizierten Gefahren Sicherheitsanforderungen abgeleitet werden. Bild 10 stellt dies dar.

Dabei muss nicht jede Gefahr automatisch einer Sicherheitsanforderung zugeordnet werden, sofern das Risiko als tolerierbar eingestuft werden kann. Ist eine Anforderung aber sicherheitskritisch, und aus der Gefahrenanalyse unmittelbar abgeleitet, so kann die Klassifikation der Anforderung übereinstimmend mit der Klassifikation des zugehörigen Risikos erfolgen.

Zusammenfassend betrachtet, lassen sich die zur systematischen Gefahren und Risikoanalyse notwendigen Ergänzungen in einem zusätzlichen, kohärenten Partialmodell „Gefahren“ vereinen.

Wie in Bild 11 dargestellt, fasst dieses Gefahrenauswirkungen, Gefährdungssituationen und Gefahren zu einem einheitlichen Partialmodell zusammen und ist eng mit den übrigen Partialmodellen von CONSENS verknüpft.

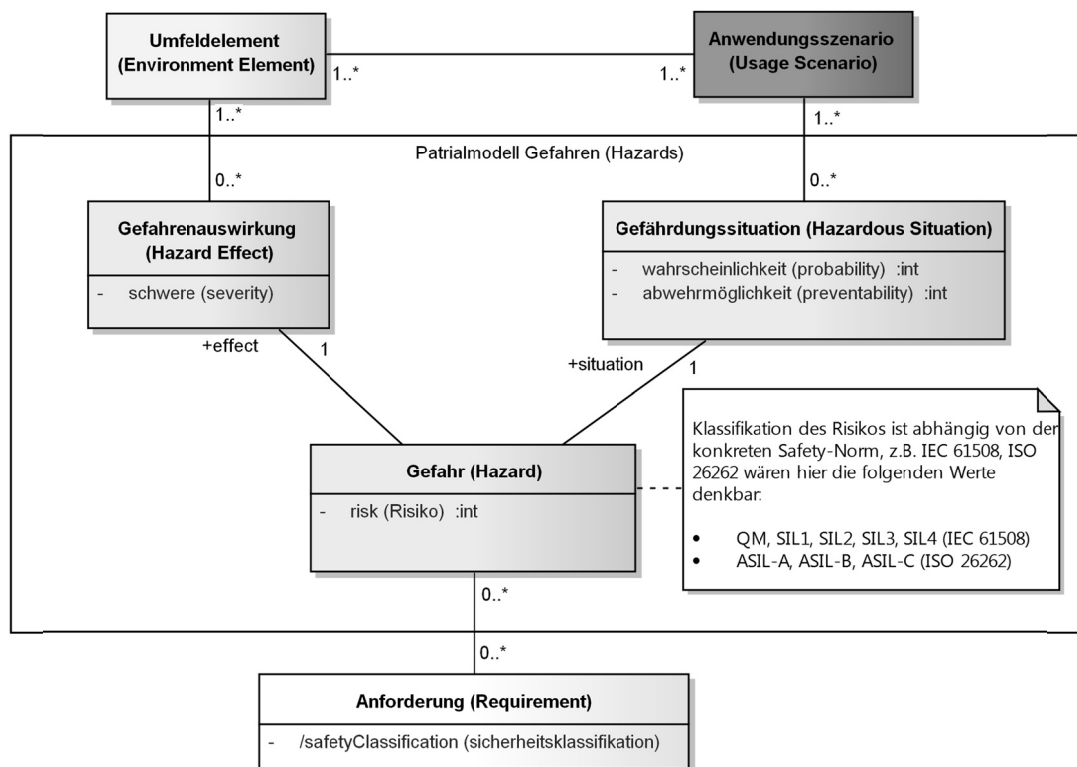


Bild 11: Neues Partialmodell "Gefahren"

3.3 Anwendung etablierter Analysemethoden in Verbindung mit CONSENS

Mit den vorstehend beschriebenen Erweiterungen (Partialmodell „Gefahren“) lassen sich etablierte Methoden zur Risikoanalyse mit CONSENS kombinieren. Anhand der Methoden FMEA und FTA wird das im Folgenden kurz skizziert.

3.3.1 Kombination von FMEA und CONSENS

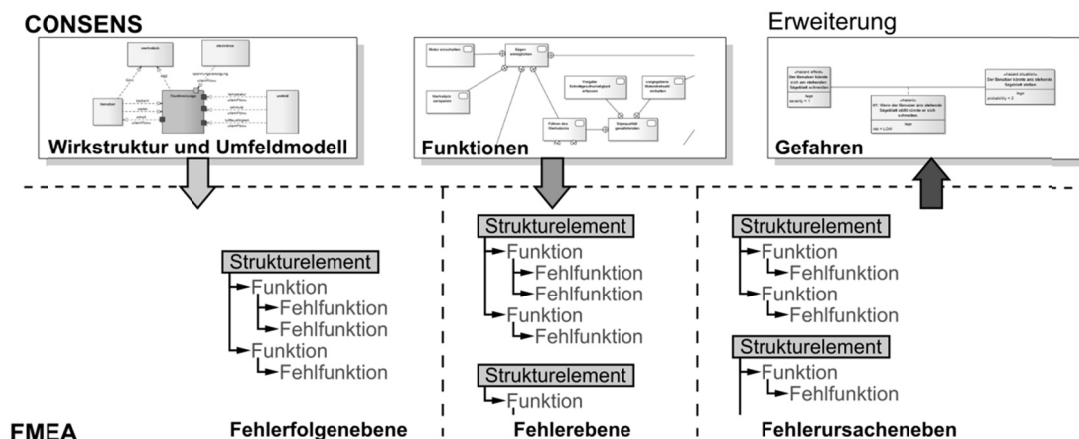


Bild 12: Anwendung einer FMEA in Verbindung mit CONSENS

VDA beschreibt das methodische Vorgehen bei der Erstellung einer FMEA (siehe Bild 12) [VDA10]. Als erstes sind die Strukturelemente des zu untersuchenden Produktes zu erstellen. Diese lassen sich aus der Wirkstruktur und dem Umfeldmodell aus CONSENS ableiten. Zu den Strukturelementen sind in der FMEA Funktionen festzulegen. Diese sind durch das Funktionsnetz in CONSENS ebenfalls gegeben. Im Rahmen der FMEA lassen sich nun zu den Funktionen jeweils Fehlfunktionen, sowie deren Ursachen und Auswirkungen identifizieren. Diese gehen als Inhalte in das neu ergänzte Partialmodell „Gefahren“ in CONSENS ein. Über die Anwendungsszenarien aus CONSENS lassen sich die Risikosituationen und mögliche Abwehrmaßnahmen ableiten. Aus allem zusammen ergeben sich klassifizierbare Risiken.

3.3.2 Kombination von FTA und CONSENS

Offensichtliche Gefahren können, basierend auf einem Systemmodell nach CONSENS, mittels eines Fehlerbaums analysiert werden [Dor15], [DG12] (siehe Bild 13). Dabei geht eine im Partialmodell „Gefahren“ identifizierte Gefahr als Top-Ereignis in einen Fehlerbaum ein. Die Wirkstruktur und die Anwendungsszenarien induzieren dabei die Struktur des Fehlerbaums. Diese wird basierend auf der Systemmodellspezifikation bis hin zu Basis-Ereignissen heruntergebrochen. Die Basisereignisse sind Ereignisse, welche nicht weiter untersucht werden. Sie beschreiben, unter welchen Bedingungen die bereits identifizierte Gefahr tatsächlich eintreten kann. Ausgehend von diesen Bedingungen lassen sich Abstellmaßnahmen in Form von Sicherheitsanforderungen formulieren, welche das mit der Gefahr verbundene Risiko minimieren.

Wenn sich in der Konzeptphase bereits Werte zu Ausfall- oder Reparaturraten einzelner Systemkomponenten angeben lassen, dann ist eine quantitative Analyse mit der FTA auf Grundlage einer gegebenen maximalen Eintrittswahrscheinlichkeit der Gefahr möglich.

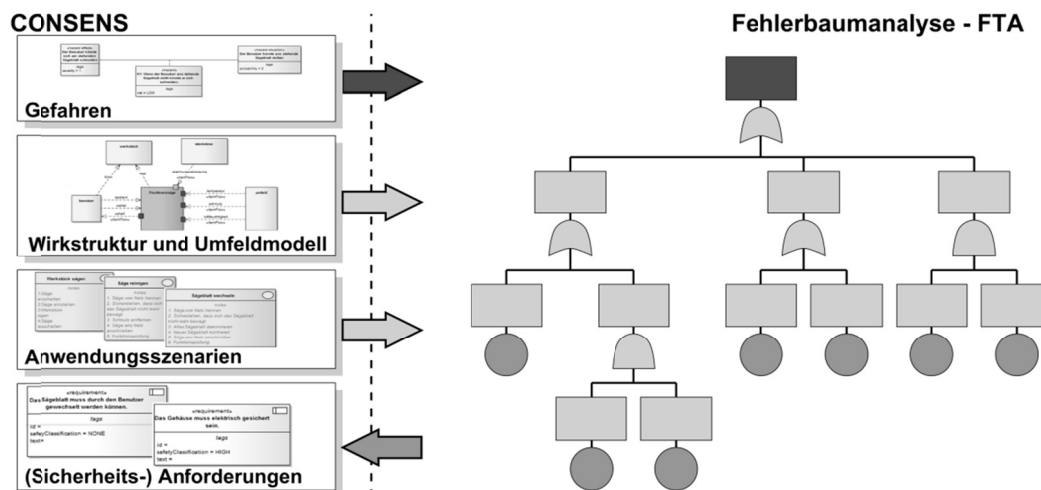


Bild 13: Anwendung einer FTA in Verbindung mit CONSENS

4 Resümee und Ausblick

Mit der Erweiterung von CONSENS um ein zusätzliches Partialmodell „Gefahren“, ist es möglich, Gefahren zu erfassen. Dabei werden Gefährdungssituationen systematisch mit möglichen Gefahrenauswirkungen kombiniert. Die Einbindung von Kenngrößen für Eintrittswahrscheinlichkeit, Abwehrmöglichkeiten und Schwere erlaubt eine Klassifikation des Gefahrenrisikos. Dies ist konform zu den gängigen Normen zur funktionalen Sicherheit.

Eine systematische und vollständige Risikoanalyse wird in der funktionalen Sicherheit normativ gefordert. Sie kann auf Basis etablierter Methoden wie FMEA und FTA erfolgen, welche sich auf Basis der vorgestellten Erweiterungen mit CONSENS nahtlos kombinieren lassen.

Durch die Verknüpfung des Partialmodells „Gefahren“ mit den anderen Partialmodellen ist zudem die Durchgängigkeit und Nachverfolgbarkeit der Sicherheitsbetrachtung gewährleistet.

Neben den in diesem Beitrag aufgeführten Methoden gibt es weitere etablierte Methoden der Sicherheitstechnik wie PHA, SHA, ETA, HAZOP, Markov-Ketten etc., die prinzipiell frühzeitig bereits in der Konzipierung eingesetzt werden können [Eri05]. In diesem Zusammenhang ergibt sich eine Reihe von potentiellen weiterführenden Themen [Dor15]. Zum einen gilt es die Frage zu klären, welche der vielen etablierten Methoden zur Absicherung der Sicherheit eines technischen Systems sich bereits in der Konzipierung auf Basis der Spezifikation der Produktkonzeption einsetzen lassen und wie diese zu gestalten sind, um diesen frühzeitigen Einsatz zu ermöglichen. Zum anderen sind Hilfsmittel zur Auswahl von Methoden zur Absicherung der Sicherheit in der Konzipierung notwendig, welche den Entwickler bei einer effektiven Suche, Auswahl und Kombination der für seine Entwicklungsaufgabe adäquaten Methoden unterstützen.

Ein weiteres Handlungsfeld besteht in der integrativen Betrachtung der funktionalen Sicherheit (Safety) im Zusammenhang mit weiteren Verlässlichkeitsaspekten wie z.B. der Zugriffssicherheit (Security).

Literatur

- [Bir10] BIROLINI, A.: Reliability Engineering . Springer Berlin 2010
- [Dor15] DOROCIAC, R.: Systematik zur frühzeitigen Absicherung der Sicherheit und Zuverlässigkeit fortschrittlicher mechatronischer Systeme. Dissertation, Fakultät für Maschinenbau, Universität Paderborn. HNI-Verlagsschriftenreihe, Paderborn 2015 (to be published)
- [DG12] DOROCIAC, R.; GAUSEMEIER, J.: Modeling of the Failure Propagation of an Advanced Mechatronic System within the Specification of its Principle Solution. In: Proceedings of the International Design Conference – DESIGN, May 21-24, Dubrovnik, Croatia, 2012
- [DIK10] DAVID, P.; IDASIAK, V.; KRATZ, F.: Reliability study of complex physical systems using SysML. In: Reliability Engineering and System Safety, Volume 95, Issue 4, Elsevier, Oxford, 2010, S. 431-450
- [EG06] Richtlinie 2006/42/EG des europäischen Parlaments und des Rates vom 17. Mai 2006 über Maschinen und zur Änderung der Richtlinie 95/16/EG (Neufassung)
- [Eri05] ERICSON, C. A.: Hazard Analysis Techniques for System Safety. John Wiley & Sons, Inc., Hoboken, New Jersey, 2005
- [Fra06] FRANK, U.: Spezifikationstechnik zur Beschreibung der Prinziplösung selbstoptimierender Systeme. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 175, Paderborn, 2006
- [GLL12] GAUSEMEIER, J.; LANZA, G.; LINDEMANN, U. (Hrsg.): Produkte und Produktionssysteme integrativ konzipieren: Modellbildung und Analyse in der frühen Phase der Produktentstehung. Carl Hanser Verlag, München, 2012
- [IEC10] INTERNATIONAL ELECTROTECHNICAL COMMISSION: IEC 61508/2010 Functional safety of electrical/electronic/programmable electronic safety-related systems; Parts 1-7
- [ISO11] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION: ISO 26262/2011: Road vehicles - - Functional safety; Parts 1-10
- [PBF+06] PAHL, G.; BEITZ, W.; FELDHUSEN, J.; GROTE, K.-H.: Konstruktionslehre – Grundlagen. Springer Verlag, Berlin, 7. Auflage, 2006
- [PH13] PFEFFER, P.; HARRER, M. (HRSG.): Lenkungsbandbuch. 2. Auflage, ATZ/MTZ-Fachbuch, Springer Fachmedien, Wiesbaden, 2013
- [VDA10] VERBAND DER AUTOMOBILINDUSTRIE: Quality Management in the Automobile Industry - Quality Assurance in the Process Landscape – general, risk analyses, methods, process models, VDA 2010

Autoren

Dr.-Ing. Guido Stollt ist Geschäftsführer und Gesellschafter der Smart Mechatronics GmbH. Smart Mechatronics ist der Entwicklungspartner für intelligente mechatronische Systeme. Die Entwicklungsbereiche Requirement und Systems-Engineering, Hardware-Technik, Software-Technik sowie Modellbildung, Simulation und Regelungstechnik arbeiten eng zusammen bei der Entwicklung der Produkte für die Märkte von morgen!

Herr Dr. Stollt studierte an der Universität Paderborn Wirtschaftsingenieurwesen und war im Anschluss daran wissenschaftlicher Mitarbeiter und Teamleiter in der Fachgruppe Produktentstehung von Prof. Dr.-Ing. Jürgen Gausemeier am Heinz Nixdorf Institut der Universität Paderborn.

Dr. Martin Schäfer ist seit 2008 Mitarbeiter der Smart Mechatronics GmbH und unterstützt verschiedene Kunden in den Bereichen Elektronikentwicklung, Reglerentwicklung und funktionale Sicherheit. Er promovierte 2001 an der Universität-Gesamthochschule Paderborn im Fachbereich Elektrotechnik. In seiner siebenjährigen Tätigkeit bei der HELLA KGaA Hueck & Co baute er ein Team zur modellbasierten Entwicklung von SW-Systemen zur Steuerung mechatronischer Systeme in Kraftfahrzeugen unter besonderer Berücksichtigung von Sicherheits- und Qualitätsaspekten. Weiterhin leitete er eine Abteilung zur Standardisierung der Elektronikentwicklung.

Dr. Alexander Nyßen arbeitet seit 2009 bei der itemis AG in Lünen. Dort unterstützt er als Consultant und Coach vor allem Kunden im Automotive-Umfeld bei der Konzeption und Einführung modellbasierter Methoden des Software- und Systems-Engineering, unter besonderer Berücksichtigung der durchgängigen Betrachtung funktionaler Sicherheit. Als Projektleiter und Architekt betreut er zudem die Entwicklung unterstützender, durchgängiger Werkzeugketten, vor allem auf der Basis von Eclipse. Vor seiner nun sechsjährigen Beschäftigung bei der itemis AG promovierte er an der RWTH Aachen University zur modellbasierten Entwicklung kleiner eingebetteter Software-Systeme.

Dr.-Ing. Rafal Dorociak³ leitet die Gruppe „Safety Platform“ in der Zentralabteilung Funktionale Sicherheit bei der HELLA KGaA Hueck & Co. in Lippstadt. Er ist verantwortlich für die Funktionale Sicherheit der nach dem Plattform-Ansatz entwickelten Produkte. Vor der Übernahme der Gruppenleitung war er als Safety Manager in den Produktbereichen Fahrerassistenzsysteme und Body Electronics bei HELLA tätig. Davor war er wissenschaftlicher Mitarbeiter am Heinz Nixdorf Institut der Universität Paderborn bei Prof. Gausemeier und erstellt in diesem Zusammenhang seine Doktorarbeit zur frühzeitigen Absicherung der Zuverlässigkeit und Sicherheit komplexer mechatronischer Systeme auf Basis einer fachgebietsübergreifenden Produktspezifikation.

³ Herr Dr. Dorociak war in dem durch die Deutsche Forschungsgemeinschaft (DFG) geförderten Sonderforschungsbereich (SFB) 614 „Selbstoptimierende Systeme des Maschinenbaus“ der Universität Paderborn als wissenschaftlicher Mitarbeiter tätig. Teile dieser Veröffentlichung sind im Zusammenhang mit seiner Tätigkeit darin entstanden.

Intelligente Regelung mechatronischer Systeme

Integrierte Modellierung der Dynamik und der Verlässlichkeit komplexer mechatronischer Systeme

Thorben Kaul, Tobias Meyer, Walter Sextro

Universität Paderborn

Pohlweg 47-49, 33102 Paderborn

Tel. +49 (0) 5251 / 60 18 11, Fax. +49 (0) 5251 / 60 18 03

E-Mail: {thorben.kaul|tobias.meyer|walter.sextro}@uni-paderborn.de

Zusammenfassung

Die starke Integration von Sensorik, Aktorik, Hard- und Software stellt Herausforderungen an die Verlässlichkeit intelligenter mechatronischer Systeme dar. Diese Systeme verfügen aber auch über großes Potential zur Verbesserung ihrer Verlässlichkeit durch eine Anpassung des Systemverhaltens an den aktuellen Zustand. Um den Umfang der Systemmodelle zu reduzieren und die Anpassung des Systemverhaltens zu ermöglichen, sind fortschrittliche Modellierungsmethoden notwendig, mit denen die Verlässlichkeit in frühen Phasen des Entwicklungsprozesses sichergestellt und evaluiert werden kann.

Von den Attributen der Verlässlichkeit ist insbesondere die Zuverlässigkeit in hohem Maße von den auftretenden Belastungen an den Komponenten und damit vom dynamischen Systemverhalten abhängig. Bisherige Modellierungsansätze bilden diese Abhängigkeit nur unzureichend ab. Es wird daher ein Ansatz zur integrierten Modellierung mechatronischer Systeme vorgestellt. Dieser ist in der Lage, sowohl die Dynamik als auch die Zuverlässigkeit des Systems abzubilden. Die Transformation eines Modells des dynamischen Systemverhaltens generiert dabei ein Zuverlässigkeitsmodell. Für typischerweise konkurrierende Ziele können mit Hilfe von Mehrzieloptimierungsverfahren Betriebspunkte eines Systems bestimmt werden. Das integrierte Modell kann zur Erzeugung von Zielfunktionen für die Dynamik als auch für die Zuverlässigkeit genutzt werden. Die Ergebnisse ermöglichen eine Verhaltensanpassung durch Wahl eines paretooptimalen Betriebspunkts während des Betriebs.

Das vorgeschlagene Konzept zur integrierten Modellierung mechatronischer Systeme bietet aufgrund des modellbasierten Entwicklungsansatzes und der automatisierten Transformation eines Verlässlichkeitsmodells eine Reduktion der Benutzereingaben und eine Entlastung des Benutzers. Dadurch wird die Wahrscheinlichkeit von Benutzerfehlern gesenkt und die Verlässlichkeit bereits während der Entwicklung erhöht. Somit können Iterationsschleifen vermieden und die Entwicklungskosten gesenkt werden.

Schlüsselworte

Verlässlichkeit, Zuverlässigkeit, Dynamik, integrierte Modellierung

Integrated model for dynamics and dependability of complex mechatronic systems

Abstract

Intelligent mechatronic systems feature complex sensor and actuator systems as well as more sophisticated information processing than in a standard mechatronic system, making system dependability a challenging effort. However, the risk of increased system complexity also provides the chance to adapt system behavior to increase dependability this way. To enable for these opportunities, reduced complexity of the system model as well as advanced modelling technics are required to evaluate system dependability in early design phase.

System reliability is highly influenced by load on individual system components during operation as the prevailing load on a component is crucial for its degradation and therefore for its lifetime. The introduced integrated model is more convenient than prior approaches in modelling the dependency between system dynamics and reliability. Therefore an integrated model for dynamics and dependability of mechatronic systems is introduced. Here a transformation of a model of system dynamics generates a model of system reliability. However, for competitive objectives of the system at hand multi-objective optimization can be used to compute operating points. The integrated model can be used to formulate performance- as well as reliability-related objective functions. To achieve an adaption of system behavior to current environmental conditions or systems states, pre-computed pareto-optimal operating points are chosen.

The proposed concept of integrated modelling of mechatronic systems reduces the user input and is less susceptible to user errors due to the model-based approach and the automated transformation of a reliability model. Applied in early design phase, integrated modelling is capable of increasing system dependability and supporting an efficient design process.

Keywords

Reliability, Dependability, System Dynamics, Integrated Modelling

1 Einleitung

An moderne mechatronische Produkte des Maschinenbaus werden steigende Funktionsanforderungen gestellt, die mit erhöhter Komplexität einhergehen. Die Kombination von Mechanik, Elektro- und Regelungstechnik, Informatik und Mathematik erlaubt darüber hinaus die Entwicklung intelligenter mechatronischer Systeme und ermöglicht die Realisierung von selbstoptimierenden Systemen als eine Klasse intelligenter mechatronischer Systeme. Sie sind in der Lage, autonom auf veränderte Umwelteinflüsse oder Systemzustände zu reagieren und ihr Betriebsverhalten daran anzupassen. Die Integration dieser Fähigkeiten steigert die ohnehin hohe Komplexität mechatronischer Systeme nochmals. Neben dieser stellen die kurzen Entwicklungszyklen und steigender Kostendruck in Entwicklung und Produktion eine Herausforderung an die Verlässlichkeit der Produkte dar [GRS+14]. Dazu trägt bei, dass aus der Reaktion selbstoptimierender Systeme auf äußere Einflüsse ein schwer vorhersagbares Betriebsverhalten resultiert, das über die Lebensdauer zu stark ändernden dynamischen Lasten auf das System selbst führt [MSK+13]. In diesem Umfeld kann die Systemzuverlässigkeit nur durch zusätzliche Maßnahmen wie fortgeschrittene Modellierungsmethoden sichergestellt werden.

Als Grundlage selbstoptimierender Systeme dient die modellbasierte Mehrzieloptimierung, mit der mögliche Betriebspunkte bestimmt werden. Soll das System in der Lage sein, sein Verhalten an die aktuelle Verlässlichkeit anzupassen, muss diese bereits in der Mehrzieloptimierung zugrundeliegenden Modell berücksichtigt werden. Dabei muss die Verlässlichkeit des Systems in einer Zielfunktion quantifiziert werden. Zur Quantifizierung der Verlässlichkeit wird die Zuverlässigkeit verwendet, da diese eine starke Abhängigkeit von dem dynamischen Systemverhalten aufweist. Aufgrund der komplexen Wirkzusammenhänge ist dabei ein integrierter Modellierungsansatz von Dynamik und Zuverlässigkeit des Systems notwendig, wie es auch im zweiten Abschnitt dieses Artikels noch näher erläutert werden wird.

Das aktuelle Vorgehen zur Ableitung verlässlichkeitsrelevanter Zielfunktionen wird vorgestellt und ein Vorgehen zur integrierten Modellierung von Dynamik und Zuverlässigkeit skizziert. Das aus dem Modellierungsansatz hervorgehende Zuverlässigkeitsmodell kann zur Aufstellung von Zielfunktionen für Mehrzieloptimierungsverfahren genutzt werden. Der dritte Abschnitt beleuchtet den Stand der Technik zur integrierten Modellierung von Dynamik und Verlässlichkeit. Im vierten Abschnitt wird ein Systemmodell für die geschlossene Modellierung von Dynamik und Zuverlässigkeit im Rahmen der Mehrzieloptimierung vorgestellt und beschrieben. Im fünften Abschnitt wird ein Fazit gezogen.

2 Verlässlichkeit komplexer mechatronischer Systeme

Bei der Entwicklung technischer Produkte müssen verschiedene Eigenschaften des Systemverhaltens, wie beispielsweise Dynamik, Effizienz und Zuverlässigkeit, als Zielgrö-

Ben gegeneinander abgewogen werden. Selbstoptimierende Systeme sind in der Lage, die Auswahl eines geeigneten Kompromisses von der Entwicklungs- in die Betriebsphase zu verlegen, indem mittels modellbasierter Mehrzieloptimierung zahlreiche mögliche Betriebspunkte bestimmt werden [GRS+14]. Dazu ist es allerdings notwendig, mathematische Modelle der Zielgrößen als Zielfunktionen zu formulieren, wobei der Entwickler mit der vollen Komplexität des Produktes konfrontiert wird.

Anpassungen des Systemverhaltens können konzeptioneller Art sein, sich auf die Auswahl der Komponenten beziehen, konstruktiv umgesetzt werden oder auch softwaretechnische Größen wie Reglerparameter und -strategien einschließen. Alle diese Änderungen, ob konzeptioneller, konstruktiver oder softwaretechnischer Art, haben Einfluss auf die Systemdynamik [GRS+14], die wiederum die auf einzelne Komponenten wirkenden Lasten und darüber die Zuverlässigkeit des Systems beeinflusst. Soll die Zuverlässigkeit, oder einzelne Aspekte wie etwa die Zuverlässigkeit, innerhalb der Mehrzieloptimierung als Zielgröße berücksichtigt werden, um eine Verhaltensanpassung zu ermöglichen, wird aktuell ein Modell der Systemzuverlässigkeit aus einem Dynamikmodell manuell abgeleitet wie in Bild 1 links dargestellt. Die Systemzuverlässigkeit wird in Abhängigkeit der Lasten der Komponenten, die sich durch die Optimierungsparameter und die zugehörige Systemdynamik ergibt, berechnet. Das Aufstellen einer Zielfunktion für die Systemzuverlässigkeit erfolgt ebenfalls auf Grundlage der Einschätzungen des Entwicklers.

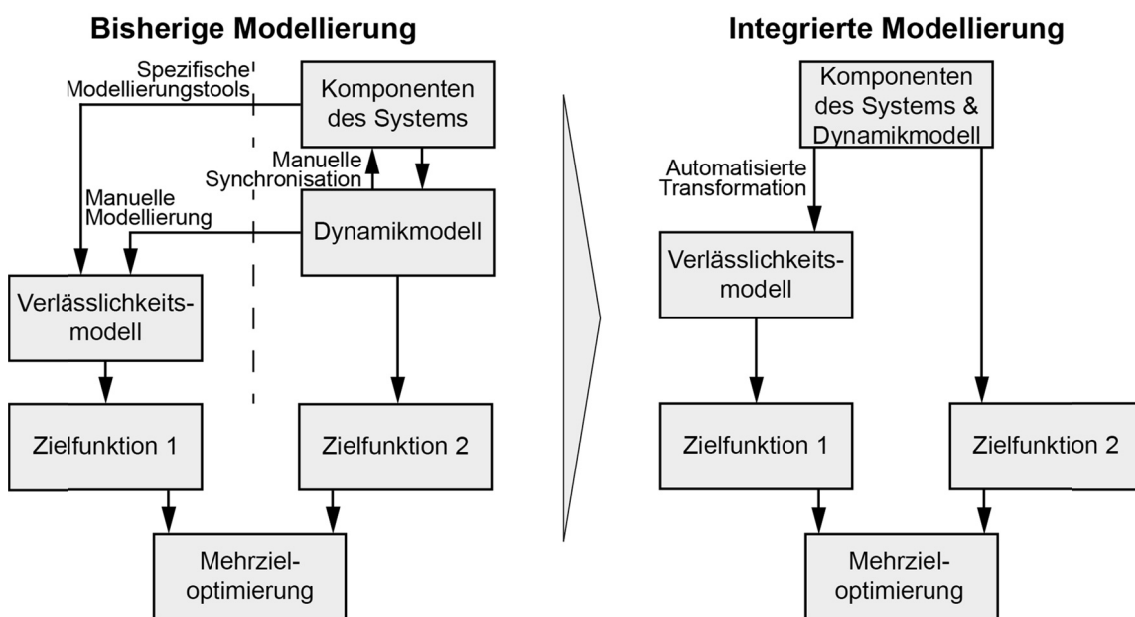


Bild 1: Bisheriges und integriertes Vorgehen zur geschlossenen Modellierung der Dynamik und Zuverlässigkeit in der Mehrzieloptimierung

Aus der manuellen Ableitung eines Modells der Systemzuverlässigkeit ergeben sich einige schwerwiegende Nachteile. Aufgrund der Komplexität mechatronischer Systeme und der starken Interaktion der Komponenten können nicht alle Aspekte berücksichtigt werden, wodurch das reale Ausfall- und Verschleißverhalten nicht vollständig abgebildet

det wird. Mit jeder Anpassung des Mehrzieloptimierungsproblems tritt nicht nur eine Veränderung der Systemdynamik oder -struktur auf, sondern es muss auch manuell das Verlässlichkeitsmodell aktualisiert werden. Dieser Prozess ist sowohl fehleranfällig als auch ineffizient. Eine automatisierte Ableitung eines Modells der Systemzuverlässigkeit (Bild 1 rechts) ist in der Lage den Entwickler zu unterstützen und sowohl die Vollständigkeit als auch die Effizienz bei der Modellbildung zu erhöhen.

3 Stand der Technik

[GFD+09] liefert einen umfangreichen Überblick über bestehende Ansätze, die neben dem Verhalten des Systems weitere Aspekte wie etwa die Ziele, die Struktur und die Gestalt abbilden. Dabei wird allerdings kein automatisch synchronisiertes Vorgehen verfolgt, sondern es werden einzelne, nicht direkt gekoppelte Modelle für die unterschiedlichen Aspekte aufgebaut.

Stärker fokussiert auf eine integrierte Modellierung von Dynamik und Verlässlichkeit komplexer Systeme sind zahlreiche Verfahren, die sich in zwei Modellierungsansätze unterteilen lassen. Der erste Ansatz versucht ein System interdisziplinär und domänenübergreifend zu beschreiben. Hier finden sowohl Sprachen zur umfassenden Systembeschreibung wie SysML, MODELICA und VHDL-AMS Anwendung ([DIK10], [Sch11], [BML07]) als auch auf die Modellierung von Dynamik und Verlässlichkeit ausgerichtete spezialisierte Sprachen wie LARES (Language for REconfigurable dependable Systems) oder AltaRica. Dem gegenüber steht der Ansatz, bestehende multidomänen-Modellierungswerkzeuge wie Matlab/Simulink mittels zusätzlichen Programmpaketen zu erweitern ([PM01], [HM08]).

Beide genannten Modellierungsansätze wurden in zahlreichen Arbeiten für eine integrierte Modellierung verwendet. Ziel der Verlässlichkeitsmodelle ist eine Quantifizierung der Systemzuverlässigkeit, qualitative Modelle werden daher nur der Vollständigkeit halber erwähnt.

Die automatisierte Erzeugung einer FMEA (Fehlermöglichkeits- und -einflussanalyse) aus SysML-Modellen wird in [DIK10] vorgestellt und ein Vergleich zu AltaRica gezogen. Das methodische Vorgehen zur Modellbildung mit SysML wird in den Entwicklungsprozess eingebunden. Qualitative Methoden wie die FMEA können jedoch nicht als Basis einer Zielfunktion zur numerischen Mehrzieloptimierung genutzt werden.

Ein Ansatz zur integrierten Modellierung von elektrischen Systemen in Flugzeugen wird in [Sch11] vorgestellt. Es werden MODELICA-Bibliotheken zur Simulation der Energieversorgung von Flugzeugen erstellt und ein Algorithmus zur automatisierten Erstellung von Zuverlässigkeitsblockdiagrammen und Fehlzustandsbäumen vorgestellt. Das Vorgehen ist auf elektrischen Komponenten beschränkt, deren Ausfallverhalten ausschließlich durch Zufallsausfälle beschrieben wird. Lebensdauerinformationen wie Ausfallraten müssen vom Benutzer eingegeben werden.

In [BML07] wird eine Methode zur Berücksichtigung der Systemzuverlässigkeit im Entwicklungsprozess elektronischer Systeme diskutiert. Ausgehend von einer Systembeschreibung in VHDL-AMS wird eine Monte-Carlo-Simulation zur Abbildung des Ausfallverhaltens des Systems durchgeführt und evaluiert. Die Methode berücksichtigt den Einfluss der Degradation auf das Betriebsverhalten elektronischer Komponenten. Eine domänenübergreifende Modellierung ist mit VHDL-AMS nicht möglich – sie dient ausschließlich der Beschreibung elektronischer Schaltung.

Zur Bewertung der Zuverlässigkeit eines selbstoptimierenden mechatronischen Systems wird in [MSS+13] das LARES-Framework verwendet. Es werden die Hardware-Komponenten des Systems in Abhängigkeit der aus der aktuellen Situation resultierenden Belastungen und die nicht-deterministische Verhaltensadaption modelliert, um die Verlässlichkeit des betrachteten Systems zu evaluieren. Dabei wird allerdings der Zusammenhang zwischen Systemkonfiguration und -verhalten zur aktuell gültigen Ausfallrate nicht im Modell selbst bestimmt, sondern muss extern vorgegeben werden.

In [PM01] wird ein Algorithmus zur automatisierten Ableitung von Fehlzustandsbäumen aus Matlab/Simulink-Modellen vorgestellt. Der Algorithmus basiert auf einer Gefahrenanalyse mittels HAZOP (HAZard and OPerability study) für jede Komponente und ist damit auf umfangreiches Benutzerwissen und -eingaben angewiesen.

Ein teilautomatisierter Algorithmus zur Durchführung einer FMEA aus Matlab/Simulink-Modellen wird in [HM08] beschrieben. Das Vorgehen ist auf umfangreiche Benutzereingaben angewiesen und eignet sich aufgrund seines qualitativen Modells nicht zur Ableitung einer Zielfunktion.

Allen Ansätzen ist gemein, dass die Synchronisation zwischen Dynamik- und Verlässlichkeitsmodell, beispielsweise über veränderte Ausfallraten oder Lasten, nicht vollautomatisch durchgeführt wird sondern noch immer eines Benutzereingriffs bedarf, was den Nutzen stark einschränkt und der Einbettung in ein Mehrzieloptimierungsproblem im Wege steht.

4 Integrierte Modellierung

Die interdisziplinäre Betrachtung von Dynamik und Verlässlichkeit eines mechatronischen Systems im Rahmen einer integrierten Modellierung erfordert umfassende Systemkenntnis. Soll das integrierte Systemmodell im Rahmen der Mehrzieloptimierung genutzt werden, müssen alle für die Zielfunktionen relevanten Modelle integriert werden. Es wird mit allen notwendigen Informationen zur Abbildung der Zielgrößen angereichert und unter Berücksichtigung aller auftretenden Wechselwirkungen zwischen den einzelnen Domänen ausgewertet.

Das Systemmodell und das Modell der Systemzuverlässigkeit können unabhängig von der Mehrzieloptimierung verwendet werden. So werden kritische Komponenten durch Evaluation der Systemzuverlässigkeit ohne Formulierung eines Mehrzieloptimierungs-

problems identifiziert und Maßnahmen können ergriffen werden. Dies stellt einen ressourcenschonenden Ansatz zur Sicherstellung einer gewünschten Systemzuverlässigkeit dar.

Im Entwicklungsprozess mechatronischer Systeme besitzt die Modellierung der Dynamik einen großen Stellenwert. Das Systemmodell basiert daher üblicherweise auf einem Modell des dynamischen Systemverhaltens, das um Informationen zu Verlässlichkeitsaspekten, z.B. Lebensdauerinformationen oder relevante geometrische Größen der Komponenten im System, erweitert wird, ohne den zugrundeliegenden Charakter eines Dynamikmodells zu verlieren. Weitere Lebensdauerinformationen der Komponenten, wie Ausfallrate oder Zuverlässigkeit, werden dann aus diesem Modell erzeugt.

Die Abbildung der Zuverlässigkeit des betrachteten Systems erfolgt automatisiert mit einer Transformation des Systemmodells in ein Modell der Systemzuverlässigkeit. Zentrales Ziel ist, zusätzliche Benutzereingaben auf ein Minimum zu reduzieren und einen hohen Automatisierungsgrad in der Modellsynthese zu gewährleisten.

4.1 Systemmodell

Der integrierten Modellierung von Dynamik und Zuverlässigkeit und der automatisierten Ableitung eines Modells der Systemzuverlässigkeit aus einem Systemmodell liegt ein topologieorientierter Modellierungsansatz zugrunde (siehe Bild 2). Das Modell des dynamischen Systemverhaltens kann unabhängig vom Modell der Systemzuverlässigkeit simuliert werden, sodass eine gezielte Auslegung der Dynamik ohne erhöhten Berechnungsaufwand durchgeführt werden kann.

Die Zuverlässigkeit der in einem System enthaltenen Komponenten bestimmt in hohem Maße die Verlässlichkeit des Gesamtsystems. Das dynamische Systemverhalten beeinflusst die an den Komponenten auftretenden Lasten. Der Einfluss dieser Lasten auf die Komponenten führt zu Degradation und einer Herabsetzung der geplanten Lebensdauer. Um die Degradation der Komponenten abbilden zu können, werden daher Lebensdauermodelle genutzt. Dazu ist für jede Komponente im Systemmodell ein Lebensdauermodell notwendig, da die auftretenden Belastungen komponentenabhängig sind und nicht auf baugleiche Komponenten verallgemeinert werden können. Das klassische Modell des dynamischen Systemverhaltens muss im Systemmodell um Modelle zur Abbildung der Lasten auf die einzelnen Komponenten erweitert werden. Die Lasten hängen von der Simulationszeit und den Optimierungsparametern ab.

Im Zuge einer Hierarchisierung und Strukturierung komplexer mechatronischer Systeme erfolgt die Bildung von Baugruppen im topologieorientierten Systemmodell entsprechend dem Aufbau des realen Systems. Den Baugruppen können eigene Lebensdauermodelle zugewiesen werden, wobei dann Lebensdauermodelle von Komponenten innerhalb der Baugruppe überschrieben werden. Die Lebensdauermodelle der Komponenten oder Baugruppen eines Systems werden hierarchisch betrachtet – das Lebens-

dauermodell auf der höchsten Hierarchieebene geht in die Berechnung der Systemzuverlässigkeit ein, darunter liegende Lebensdauermodelle werden vernachlässigt. Die im Systemmodell vorhandene Hierarchisierung der Komponenten in Baugruppen wird für eine konsistente Abbildung im Zuverlässigkeitsmodell übernommen.

Um die Abhängigkeit zwischen dynamischem Systemverhalten und Degradation der Komponenten mittels Lebensdauermodellen abbilden zu können, ist eine Berechnung des Systemmodells auf zwei unterschiedlichen Zeitskalen notwendig. Die dynamischen Lasten auf die Komponenten werden mittels einer Simulation des Dynamikmodells bestimmt. Es wird das Dynamikmodell für ein *charakteristisches Manöver* mit hinreichend kleiner Integrationsschrittweite simuliert. Das charakteristische Manöver stellt einen Bewegungsablauf des betrachteten Systems dar, der die tatsächliche Nutzung möglichst gut abbildet, und ist identisch mit dem Betriebszyklus für den das Systemverhalten im Rahmen der Mehrzieloptimierung optimiert wird.

Es wird eine konsistente Behandlung des Systemmodells während der Simulation des dynamischen Systemverhaltens und der Optimierung gewährleistet und der Simulationsaufwand begrenzt. Aus dem charakteristischen Manöver muss eine aussagekräftige Quantifizierung des Systemverhaltens mittels der Zielfunktionen der Systemdynamik und der Systemzuverlässigkeit möglich sein.

Die Berechnung der Systemzuverlässigkeit aus dem abgeleiteten Zuverlässigkeitsmodell wird für die Betriebsdauer des betrachteten Systems auf der zweiten Zeitskala durchgeführt und liegt im Bereich von Stunden bis Jahren. Die Simulation des Dynamikmodells über die gesamte Betriebsdauer des Systems zur vollständigen Abbildung der auftretenden Lasten ist nicht sinnvoll und mit steigender Komplexität des betrachteten Systems nicht mehr durchführbar.

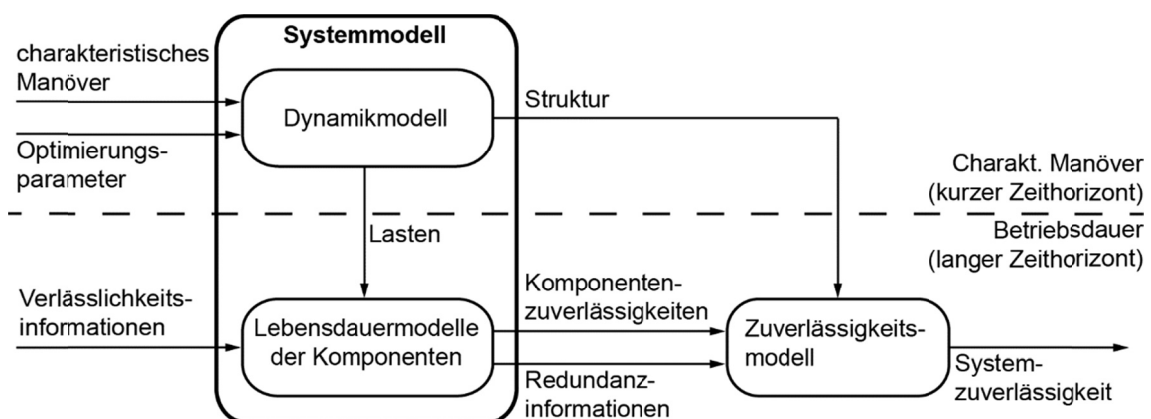


Bild 2: Systemmodell und abgeleitetes Zuverlässigkeitsmodell

Die im charakteristischen Manöver berechneten Lasten werden in den Lebensdauermodellen der Komponenten kumuliert und eine äquivalente Last für das charakteristische Manöver berechnet. Für die äquivalente Last kann mittels einer Energiebetrachtung die Degradation der Komponente bestimmt und die Lebensdauer ermittelt werden. Die Le-

bensdauer wird unter der Annahme berechnet, dass die gewählten Modellparameter Mittelwerte einer Menge an baugleichen Komponenten abbilden. Demnach kann die berechnete Lebensdauer als Mittelwert der ausfallfreien Zeit, MTTF (Mean Time To Failure), der Komponente betrachtet werden.

Das Ausfallverhalten maschinenbaulicher Produkte kann in guter Näherung mit der Weibull-Verteilung beschrieben werden. Für viele Maschinenelemente, wie Wälzlager, sind die Parameter der Verteilung bekannt [Bir07]. Elektronische Bauteile sind zumeist Zufallsausfällen unterworfen. Die unbekannt Parameter der Verteilungsfunktionen können mittels der MTTF berechnet werden. Unter Annahme einer Verteilungsfunktion ist es möglich, die Zuverlässigkeit der Komponenten über der Zeit in Abhängigkeit der Optimierungsparameter zu ermitteln.

Die Zuverlässigkeitswerte der Komponenten gehen in die Parametrierung des Zuverlässigkeitsmodells ein. Es stellt die Abhängigkeiten und die Auswirkungen des Ausfallverhaltens einzelner Komponenten auf das Gesamtsystem dar. Eine Auswertung des Zuverlässigkeitsmodells liefert die Systemzuverlässigkeit über der Betriebsdauer, aus der dann eine Zielfunktion für die Mehrzieloptimierung synthetisiert wird.

4.2 Transformation eines Zuverlässigkeitsmodells

Zentral für die integrierte Modellierung von Dynamik und Zuverlässigkeit mechatronischer Systeme ist ein Algorithmus zur vollständig automatisierten Transformation eines Zuverlässigkeitsmodells aus dem Systemmodell.

Das Zuverlässigkeitsmodell umfasst sowohl strukturelle als auch quantitative Anteile. Die Modellstruktur kann aus der Analyse der Interaktion der Komponenten abgeleitet werden. Je nach vorliegendem Dynamikmodell können beispielsweise Signalpfade zwischen den Komponenten analysiert werden, um Rückschlüsse auf Abhängigkeiten im kausalen Ausfallverhalten der Komponenten ziehen zu können. Die quantitativen Anteile des Zuverlässigkeitsmodells werden in der Parametrierung berücksichtigt und umfassen unter anderen Komponentenzuverlässigkeiten oder Redundanzinformationen.

Der Transformationsalgorithmus muss unterschiedliche Definitionen des Systemausfalls in der Modellbildung berücksichtigen. Die Benutzervorgabe, wann das betrachtete System als ausgefallen gilt, wird vom Algorithmus in geeigneter Form in das Zuverlässigkeitsmodell übertragen. Die Parametrierung des Zuverlässigkeitsmodells findet auf Ebene des Systemmodells statt. Der Benutzer hat aufgrund der automatisierten Transformation keine direkte Interaktionsmöglichkeit mit dem Zuverlässigkeitsmodell, was eine konsistente Modellbildung im Rahmen der Optimierungsiterationen gewährleistet.

4.3 Parametrierung des Zuverlässigkeitsmodells

Das Systemmodell wird vorzugsweise über eine GUI (Graphical User Interface) parametrieren. Diese Parametrierung ist in zwei Abschnitte unterteilt – auf die Modelle der Dynamik und der Zuverlässigkeit.

Die Parametrierung des Zuverlässigkeitsmodells umfasst die Definition des charakteristischen Manövers, eines kritischen Ereignisses und allgemeiner verlässlichkeitsrelevanter Kenngrößen. Das charakteristische Manöver ist bereits für die Mehrzieloptimierung definiert und kann hier erneut verwendet werden. Im Modell wird es durch vom Benutzer identifizierte Eingänge des Systemmodells und geeignet definierte Eingangsgrößen abgebildet.

Der Ausfall eines beliebigen Systems ist an die zu erfüllende Funktion gebunden — kann die gewünschte Funktion des Systems nicht erfüllt werden, gilt es als ausgefallen. Komplexe mechatronische Systeme besitzen neben einer Hauptfunktion typischerweise eine große Anzahl an Nebenfunktionen, die keinen unmittelbaren Beitrag zur Hauptfunktion leisten, wie etwa Komfortfunktionen im Automobil. Der Benutzer muss anhand der Funktionen eines mechatronischen Systems eine Definition des zu betrachtenden Systemausfalls vornehmen. Dies kann in Abhängigkeit des gewählten Zuverlässigkeitsmodells über die Definition eines unerwünschten Ereignisses durchgeführt werden. Das unerwünschte Ereignis kann beispielsweise das Ausbleiben oder die Überschreitung eines Toleranzbandes für ein beliebiges Signal innerhalb des Systemmodells sein.

Die Eingabe von verlässlichkeitsrelevanten Kenngrößen und Komponentenparametern findet über die GUI der Komponenten im Systemmodell statt. Die Eingabe von Kenngrößen des Zuverlässigkeitsmodells hängt von der Modellierungstiefe des dynamischen Systemverhaltens ab. So kann es für Redundanzen ausreichend sein, die Redundanzordnung einer Komponente in der GUI anzugeben. Dieses Vorgehen führt zu einem Minimalmodell des dynamischen Systemverhaltens, in dem redundante Komponenten nur einmal eingebunden sind. Eine Rekonfiguration bei der Nutzung redundanter Komponenten ist so nicht abbildbar. Hier muss das Dynamikmodell des Systems alle redundanten Komponenten und ein Logikelement zur Rekonfiguration enthalten. Daneben ist die Eingabe von Ausfallraten oder Komponentenzuverlässigkeiten möglich, wodurch die Auswertung eines Lebensdauermodells nicht mehr notwendig ist. Die Eingabe von Komponentenparametern, wie etwa Geometrie oder Werkstoffkenngrößen, sind von den Parametern des Modells des dynamischen Systemverhaltens zu unterscheiden und dienen ausschließlich der Parametrierung der Lebensdauermodelle der Komponenten.

4.4 Einbettung in Mehrzieloptimierung

Das beschriebene Systemmodell und die automatisierte Transformation eines Zuverlässigkeitsmodells können im Rahmen der Optimierung komplexer mechatronischer Systeme im Entwicklungsprozess angewandt werden.

Numerische Mehrzieloptimierungsverfahren arbeiten iterativ; Parameterwerte werden im Optimierungsalgorithmus mit jeder Iteration variiert, das dynamische Systemverhalten beeinflusst, und das resultierende Verhalten über die Zielfunktionen quantifiziert und bewertet. Unter der Annahme, dass keine Rekonfiguration redundanter Komponenten stattfindet, wird das Zuverlässigkeitsmodell zu Beginn der Optimierung abgeleitet. In den Iterationsschritten der Optimierung findet nur noch eine Anpassung der Parameter aufgrund der veränderten dynamischen Lasten auf die Komponenten und eine Berechnung der aktuellen Systemzuverlässigkeit statt. Die dynamischen Lasten auf die Komponenten werden aus der Simulation des charakteristischen Manövers innerhalb der Optimierung gewonnen. Der zusätzliche Berechnungsaufwand zur Berücksichtigung der Zuverlässigkeit in einer Zielfunktion wird so stark reduziert.

5 Resümee und Ausblick

Die zentralen Aspekte dieser Arbeit sind einerseits die Synthese eines Zuverlässigkeitsmodells aus einem Systemdynamikmodell und andererseits die algorithmische Zusammenführung von Lebensdauermodellen einzelner Komponenten oder Baugruppen mit Parametrierung aus dem Systemdynamikmodell. Zentraler Bestandteil ist die Modellierung des betrachteten Systems auf zwei Zeitskalen zur Berechnung der Systemzuverlässigkeit über die gesamte Betriebsdauer.

Ein Systemmodell, das eine automatisierte Transformation eines Zuverlässigkeitsmodells ermöglicht, entlastet den Benutzer erheblich und ist in der Lage, den Entwicklungsprozess bereits in frühen Phasen signifikant zu unterstützen.

Die vorgestellte integrierte Modellierung der Dynamik und der Verlässlichkeit komplexer mechatronischer Systeme ermöglicht die Abbildung der Abhängigkeit der Zuverlässigkeit vom dynamischen Systemverhalten. Unberücksichtigt bleibt jedoch der Einfluss der Degradation der Komponenten auf die Dynamik des Systems. Ein Ansatz weiterer Arbeiten kann die Untersuchung des Einflusses von Degradation und daraus resultierender Geometrie- und Parameteränderungen auf das dynamische Systemverhalten sein, was zu einer vollständigeren Abbildung mechatronischer Systeme im Sinne der integrierten Modellierung führen würde.

Literatur

- [Bir07] BIROLINI, A.: Reliability engineering. Bd. 5, Springer, 2007.
- [BML07] BESTORY, C.; MARC, F.; LEVI, H.: Statistical analysis during the reliability simulation, In: Microelectronics Reliability 47.9, S. 1353-1357, 2007.
- [DIK10] DAVID, P.; IDASIAK, V.; KRATZ, F.: Reliability study of complex physical systems using SysML, In: Reliability Engineering & System Safety, S. 431-450, 2010.
- [GFD+09] GAUSEMEIER, J.; FRANK, U.; DONOTH, J.; KAHL, S.: Specification technique for the description of self-optimizing mechatronic systems, Research in Engineering Design 20.4, S. 201-223, 2009.

- [GRS+14] GAUSEMEIER, J.; RAMMIG, F. J.; SCHÄFER, W.; SEXTRO, W.: Dependability of Self-Optimizing Mechatronic Systems, Lecture Notes in Mechanical Engineering. Springer Berlin Heidelberg, 2014.
- [HM08] HECHT, H.; MENES, R.: Software FMEA Automated and as a Design Tool, In: Proc of the Society of Automotive Engineers, 2008.
- [MSS+13] MEYER, T.; SONDERMANN-WÖLKE, C.; SEXTRO, W.; MARTIN RIEDL; GOUBERMAN, A.; SIEGLE, M.: Bewertung der Zuverlässigkeit selbstoptimierender Systeme mit dem LARES-Framework, In: 9. Paderborner Workshop Entwurf mechatronischer Systeme, S. 161-174, 2013.
- [MSK+13] MEYER, T.; SONDERMANN-WÖLKE, C.; KIMOTHO, J. K.; SEXTRO, W.: Controlling the Remaining Useful Lifetime using Self-Optimization, In: Chemical Engineering Transactions 33, S. 625-630, 2013.
- [PM01] PAPADOPOULOS, Y.; MARUHN, M.: Model-based synthesis of fault trees from matlab-simulink models, In: International Conference on Dependable Systems and Networks, DSN 2001. S. 77-82, 2001.
- [Sch11] SCHALLERT, C.: Inclusion of reliability and safety analysis methods in modelica, In: 8th International Modelica Conference, 2011.

Autoren

Thorben Kaul hat Maschinenbau an der Universität Paderborn studiert. Seit 2014 ist er wissenschaftlicher Mitarbeiter am Lehrstuhl für Mechatronik und Dynamik der Universität Paderborn. Im Rahmen seiner Tätigkeit am Lehrstuhl befasst er sich mit der integrierten Modellierung von Dynamik und Verlässlichkeit mechatronischer Systeme.

Tobias Meyer hat Maschinenbau an der Universität Paderborn studiert. Seit 2011 ist er wissenschaftlicher Mitarbeiter am Lehrstuhl für Mechatronik und Dynamik der Universität Paderborn. Seine Forschung befasst sich mit der Nutzung von Selbstoptimierung zur Anpassung des Systemverhaltens zur Erhöhung der Verlässlichkeit.

Walter Sextro hat Maschinenbau an der Leibniz Universität Hannover und am Imperial College in London studiert. Nach seiner Industrietätigkeit in Deutschland und den USA promovierte er 1997 an der Universität Hannover. Seine Habilitation hat er im Bereich dynamischer Kontaktprobleme mit Reibung verfasst. In den Jahren 2004-2009 hatte er eine Professur für Mechanik und Getriebelehre an der TU Graz inne. Seit 2009 leitet er den Lehrstuhl für Mechatronik und Dynamik der Universität Paderborn.

Entwurf eines Tripod-basierten Inspektionssystems – Vom virtuellen Prototypen zum Vorseriensystem

***M. Sc. Arne Rüting, Dr.-Ing Christian Henke, Dr.-Ing. Andreas Warkentin,
Prof. Dr.-Ing. habil. Ansgar Trächtler***

Fraunhofer-Institut für Produktionstechnologie IPT

Projektgruppe Entwurfstechnik Mechatronik

Zukunftsmeile 1 , 33102 Paderborn

Tel. +49 (0) 5251 / 54 65 433 , Fax. +49 (0) 5251 / 54 65 102

E-Mail: arne.rueting@ipt.fraunhofer.de

Zusammenfassung

Die exakte und automatisierte Positionierung von Sensoren in beengten Arbeitsbereichen mit einer Vielzahl von Hindernissen spielt heute gerade im Rahmen von Inspektionssystemen eine große Rolle. Nur dadurch können die immer weiter steigenden Anforderungen an die Qualität, die Sicherheit und die Zuverlässigkeit von Produkten erfüllt und gleichzeitig der Ressourcenaufwand sowie die dabei anfallenden Kosten reduziert werden. Bei der Entwicklung solcher Positioniersysteme ermöglicht der Einsatz von Mehrkörpersimulationen eine detaillierte Untersuchung des Systems anhand eines virtuellen Prototyps, sowie die exakte Absicherung der geforderten Eigenschaften, bevor ein reales Vorseriensystem aufgebaut wird.

Der vorliegende Beitrag beschäftigt sich mit dem Entwurfsprozess einer hybriden Kinematik zur Positionierung eines sensorbasierten Inspektionssystems innerhalb eines schlanken, konischen Arbeitsraumes mit innenliegenden konstruktions- und prozessbedingten Hindernissen. Es wird zunächst eine modellbasierte Analyse bestehender Systeme durchgeführt, bevor im Anschluss eine modellbasierte Eigenentwicklung näher betrachtet wird. Dabei wird auch der anhand eines virtuellen Prototypen durchgeführte Entwurf der zugehörigen Steuerungs- und Antikollisionsalgorithmen dargestellt.

Schlüsselworte

Modellbasierter Entwurf, Mehrkörpersimulation, hybride Kinematik, Co-Simulation, virtueller Prototyp

Design of a tripod-based inspection system – From a virtual prototype to a pre-series system

Abstract

The precise and automated positioning of sensors in cramped working areas with several obstacles is very important in the field of inspection systems. This is imperative in order to both fulfill the requirements for the quality, the safety and the reliability of products as well as to reduce the amount of resources and costs. During the development of such a positioning system, a multi-body simulation enables a detailed study of the system by using a virtual prototype. Also, the required properties can be accurately tested, before a real pre-series system is build up.

The present article deals with the design process of a hybrid kinematic for positioning a sensor based inspection system. The system will be used in a slim-line working area with construction and process related obstacles. At first, a model-based analysis of existing systems is realised. Subsequently, a model-based in-house development of a positioning system is presented. Additionally, the development of the control and anti-collision algorithm, based on the virtual prototype, is shown.

Keywords

model-based design, multi-body simulation, hybrid kinematics, co-simulation, virtual prototype

1 Einleitung

Um die heutzutage immer weiter steigenden Anforderungen an die Qualität und Sicherheit von Produkten und Anlagen zu erfüllen und gleichzeitig die dabei anfallenden Kosten zu reduzieren, spielt die exakte und automatisierte Positionierung von Sensoren im Rahmen von Inspektionssystemen eine immer größere Rolle. Die sich dabei ergebenden Arbeitsräume sind oftmals beengt durch eine Vielzahl von innenliegenden konstruktions- und prozessbedingten Hindernissen. Zudem weisen sie nicht selten anspruchsvolle Umgebungsbedingungen auf, wie z.B. eine explosionsfähige Atmosphäre, was einen direkten Einfluss auf die Auswahl der einzusetzenden Komponenten hat.

Um solche Positioniersysteme möglichst zeit- und kosteneffizient zu entwickeln, werden in der Entwurfsphase vermehrt Mehrkörpersimulationen (MKS) eingesetzt. Dabei können die einzelnen Komponenten des Systems direkt aus einem CAD-Modell (engl. Computer-Aided Design) importiert oder in der Simulation aus Grundbausteinen zusammengesetzt werden. Verbunden werden die Körper über passive Koppellemente und semiaktive oder aktive Stellglieder [RS10]. Auch wenn die Mehrkörpermodelle gegenüber dem realen System vereinfacht sind, so können doch die Eigenschaften des Systems bereits vor der Erstellung eines ersten Prototypens überprüft und etwaige Schwachstellen somit ohne Kosten für Material und Fertigungszeit aufgedeckt und behoben werden. Die Integration von Regelungs- und Softwaretechnik ermöglicht es zudem, die direkt von der Wahl der Komponenten und ihrer Anordnung abhängigen Steuerungs- und Antikollisionsalgorithmen im Vorfeld an dem vorhandenen Mehrkörpermodell zu verifizieren. Dadurch werden die Zuverlässigkeit und die Sicherheit der Systeme erhöht, sowie die spätere Inbetriebnahme vereinfacht.

Im Folgenden wird zunächst die modellbasierte systematische Analyse verschiedener am Markt erhältlicher kinematischer Systeme unter Berücksichtigung der konstruktiven Beschaffenheit des Arbeitsraumes vorgestellt. Dabei werden die Systeme bezüglich ihrer Einsetzbarkeit zur Positionierung der Sensoren bewertet. Im Anschluss wird die ebenfalls modellbasiert durchgeführte Neuentwicklung eines Positioniersystems beschrieben, bevor näher auf die erforderlichen Steuerungs- und Antikollisionsalgorithmen eingegangen wird.

2 Modellbasierte Analyse bestehender Systeme

Die erste Wahl bei der Suche nach Systemen zur Positionierung von Sensoren in beengten Arbeitsräumen fällt auf bereits am Markt verfügbare Systeme. Dadurch, dass diese vollständig entwickelt und getestet sind, sinkt das Entwicklungsrisiko des sensorbasierten Inspektionssystems stark. Zudem bieten die Hersteller im Fehlerfall einen entsprechenden Support an. Jedoch sind die existierenden Systeme zum Großteil nicht für eine Positionierung von Sensoren in beengten Arbeitsräumen entwickelt worden. Deswegen ist eine Analyse durchzuführen, ob und wenn ja welches System geeignet ist.

Obwohl es sich bei diesem Arbeitsschritt um eine Analyse und nicht um einen Entwurf eines mechatronischen Systems handelt, so können doch einige der dort etablierten Ansätze adaptiert werden [TOS11], [GSA+11]. Das zugehörige Vorgehensmodell ist in Bild dargestellt. Zunächst ist es erforderlich, eine fachgebietsübergreifende Zielbestimmung durchzuführen und die möglichen Anwendungsszenarien zu klären. Zudem muss das Umfeld, in diesem Fall der Arbeitsraum, und die daraus resultierenden Einflüsse auf das Positioniersystem analysiert werden. Aus diesen beiden Punkten ergeben sich letztendlich die Anforderungen an das Positioniersystem. Dies kann etwa die durch die Größe des Arbeitsraumes gegebene erforderliche Reichweite des Systems, aber auch der bei dem eingesetzten Sensor einzuhaltender Messabstand zu einem Objekt sein. Es ergibt sich im vorliegenden Fall, dass das Positioniersystem einen zunächst zylindrischen und dann konusförmig zulaufenden Raum von maximal 1,3 m Durchmesser und 1,7 m Tiefe abdecken muss, wobei der Abstand zwischen Sensor und zu untersuchender Oberfläche maximal 100 mm betragen darf. Es ist jedoch lediglich eine Positioniergenauigkeit im Millimeterbereich mit einer geringen Bewegungsdynamik erforderlich, da der Sensor eine geringe maximale Verfahrensgeschwindigkeit aufweist. Zudem sind für alle eingesetzten Komponenten und Systeme Explosionsschutzzulassungen nach der ATEX¹-Richtlinie erforderlich. Die Herstellungskosten sind für eine erfolgreiche Vermarktung möglichst gering zu halten.

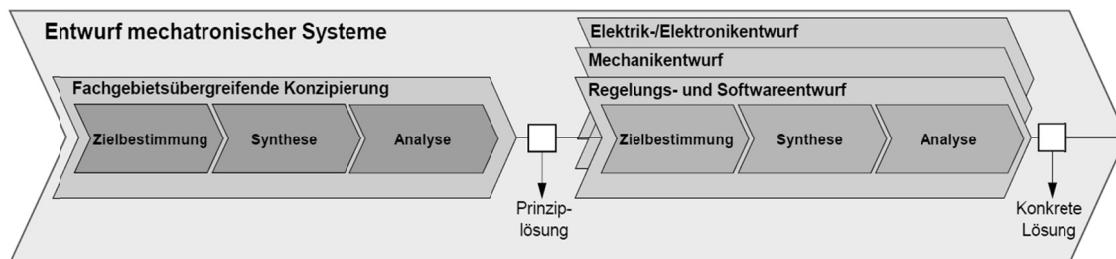


Bild 1: Vorgehensmodell für den Entwurf mechatronischer Systeme [GSA+11]

Gerade durch die Komplexität des Arbeitsraumes mit den innenliegenden konstruktions- und prozessbedingten Hindernissen lassen sich jedoch nicht alle Anforderungen an das Positioniersystem im Voraus konkret formulieren. Zudem werden dynamische Effekte, die bei Bewegungen des Systems auftreten, zum Großteil vernachlässigt. Deswegen wird im Folgenden auf eine modellbasierte Analyse von bereits am Markt erhältlichen Positioniersystemen zurückgegriffen, um die verschiedenen Kinematiken bezüglich ihrer Einsetzbarkeit für die Positionierung eines Sensors in einem definierten Arbeitsraum zu untersuchen. Die bereits ermittelten Rahmenbedingungen ermöglichen es, eine Vorauswahl zu treffen und somit den Aufwand zu senken. Es werden sowohl serielle und parallele als auch hybride Kinematiken betrachtet. Dabei wird auf die vom jeweiligen Hersteller bereitgestellten CAD-Daten zurückgegriffen, um die Systeme möglichst realitätsnah abbilden zu können. In der verwendeten Mehrkörpersimulationssoft-

¹ ATEX (fr. ATmosphère EXplosible): bezeichnet die EU-Explosionsschutzrichtlinie 2014/34/EU

ware RecurDyn² werden diese dann um passive Koppelglieder und aktive Stellglieder mit hinterlegten Bewegungsfunktionen ergänzt, um auch dynamische Effekte zu berücksichtigen. Auf eine detaillierte Nachbildung der Antriebssysteme etc. wird verzichtet. Jedoch wird zusätzlich der gesamte Arbeitsraum in der Mehrkörpersimulation inklusive beweglicher Komponenten abgebildet, um die Erreichbarkeit aller mit dem Sensor abzudeckenden Flächen so genau wie möglich analysieren zu können. Mittels einer Co-Simulation mit der Software CoLink werden zusätzlich Vorgaben für die Bewegung und Testtrajektorien eingebunden (siehe Bild 2).

Die Ergebnisse der modellbasierten Analyse werden in den folgenden Abschnitten beschrieben, Tabelle 1 enthält eine abschließende Zusammenfassung.

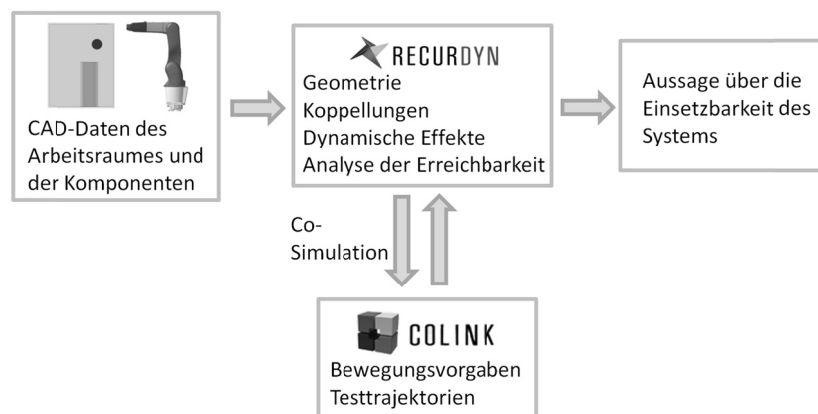


Bild 2: Vorgehen bei der modellbasierten Analyse

2.1 Serielle Kinematiken

Bei den seriellen Kinematiken sind aufgrund der bereits definierten Anforderungen lediglich Leichtbau-Knickarmroboter sowie Schlangenroboter zu betrachten, da es sich hier um verhältnismäßig schlanke Systeme mit großer Reichweite handelt.

Bei den Knickarmrobotern werden Systeme unterschiedlicher Hersteller, u.a. ein UR10 von Universal Robots [UR14-01] und ein VS087 von Denso [DR14-01], analysiert (siehe Bild 3). Für die Modellierung in RecurDyn wird auf die vom Hersteller bereitgestellten CAD-Daten zurückgegriffen. Bei den durch aktive Stellglieder nachgebildeten Gelenken werden direkt die möglichen Winkel sowie die maximalen Drehmomente als feste Grenzen berücksichtigt, zusätzliche Komponenten wie die Robotersteuerung werden vernachlässigt. Es stellt sich heraus, dass die Systeme für den geplanten Anwendungsfall nicht geeignet sind. Zum einen sind die Gelenke der Roboter verhältnismäßig stabil ausgeführt, um eine hohe Steifigkeit und damit hohe Positioniergenauigkeiten zu erzielen. Dadurch können die am TCP (engl. Tool Center Point) befestigten Sensoren nicht

² Mehrkörpersimulationsumgebung der Firma FunctionBay, Inc., die Schnittstellen zu verschiedenen CAD-Dateiformaten bietet und eine Co-Simulation etwa mit MATLAB Simulink ermöglicht

in enge Bereiche des Arbeitsraumes verfahren werden. Dies ist in Bild 4 dargestellt (linke Seite). Der Roboter (Denso VS087) soll den TCP in der Spitze des konusförmigen Arbeitsraumes platzieren, aufgrund eines vertikalen Hindernisses ist dies jedoch nicht möglich. Zum anderen werden hohe Reichweiten nur durch lange starre Elemente zwischen den Gelenken erreicht. Dadurch ist gerade die Beweglichkeit um Hindernisse herum stark eingeschränkt und somit keine vollständige Erreichbarkeit gegeben.

Die Schlangenroboter ermöglichen bei der Bewegung, je nach Anzahl der Gelenke und der Länge der dazwischen befindlichen Verbindungselemente, engere Radien als die Knickarmroboter [OCR14-ol] (siehe Bild 3). Jedoch zeigt die in RecurDyn durchgeführte Analyse, dass die Masse des Sensors bei der geforderten Reichweite schnell die Traglast des Systems überschreiten kann. Zudem haben dynamische Effekte einen großen Einfluss auf die in den Gelenken auftretenden Kräfte. Somit sind auch Schlangenroboter für den Anwendungsfall nicht einsetzbar.

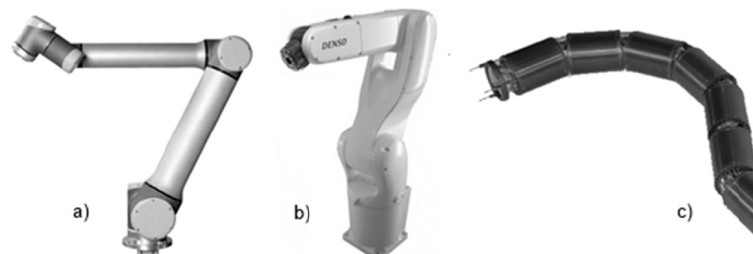


Bild 3: Übersicht der seriellen Kinematiken: a) Knickarmroboter UR10 [UR14-ol], b) Knickarmroboter VS087 [DR14-ol], c) Schlangenroboter Explorer [OCR14-ol]

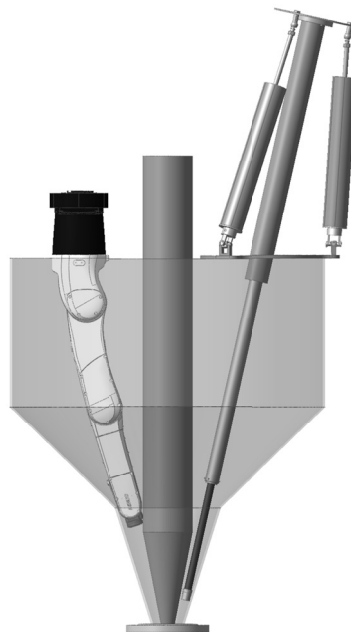


Bild 4: Eingeschränkte Beweglichkeit des Roboters und Erreichbarkeit mit hybrider Kinematik

2.2 Parallele Kinematiken

Im Bereich der parallelen Kinematiken werden insbesondere sogenannte Tripode betrachtet (siehe Bild 5). Die meisten der am Markt erhältlichen Systeme sind für Montage- und Handlingaufgaben entwickelt, bei denen schnelle und präzise Bewegungen mit geringen Lasten gefordert werden und mit hoher Taktrate definierte Positionen anzufahren sind. Die Systeme sind schlank aufgebaut, um die benötigte hohe Dynamik zu erreichen. Jedoch resultiert daraus eine geringe Nutzlast. Zum anderen gibt es Ansätze zur Nutzung in Bearbeitungsmaschinen, um etwa ein Fräs Werkzeug zu führen. Der mögliche Arbeitsbereich hat in beiden Anwendungsfällen zwar einen verhältnismäßig großen Radius, um z.B. größere Montagestationen zu bedienen, aber nur eine geringe Tiefe (z.B. Fanuc M-2iA/3SL: Radius 565 mm bei maximal 400 mm vertikaler Bewegung in der Mitte des Arbeitsbereichs [FRI13-ol]). Dadurch fallen diese Systeme schon aufgrund der zu Beginn ermittelten Anforderungen an die Reichweite aus, da auch eine Kombination mit zusätzlichen Portalachsen nicht sinnvoll erscheint. Es ist somit keine zusätzliche Analyse mittels Mehrkörpersimulation erforderlich.

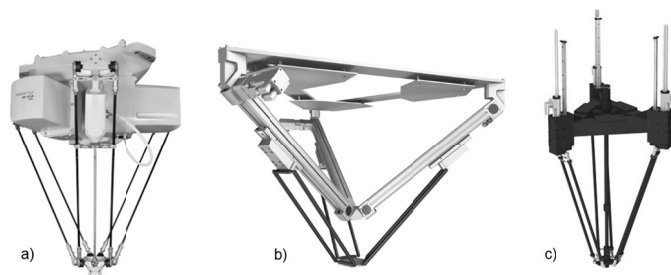


Bild 5: Übersicht der parallelen Kinematiken: a) Fanuc M-2iA/3SL [SAS14-ol] b) Tripod EXPT [FE14-ol], c) ParaPicker [IAP14-ol]

2.3 Hybride Kinematiken

Gerade im Bereich der Fertigungsmaschinen wird versucht, die Vorteile von seriellen und parallelen Kinematiken zu kombinieren. Dadurch soll die Fertigung großer und komplexer Bauteile ermöglicht werden. Als Basissystem dient oftmals ein Tripod, an dessen TCP eine zusätzliche serielle Kinematik, etwa ein Linearantrieb mit einem daran befestigten Schwenkantrieb, angebracht wird (siehe Bild 6). Dadurch wird der Arbeitsbereich in vertikale Richtung erweitert. Das resultierende System wird als hybride Kinematik bezeichnet [WEC02]. Allerdings sind diese Systeme für eine bestimmte Maschine, wie z.B. ein Fräszentrum, optimiert und fest mit dieser verknüpft. Um die dort gestellten Anforderungen an Präzision und Steifigkeit zu erfüllen, sind die einzelnen Komponenten massiv ausgeführt, woraus ein großes Volumen und ein hohes Gewicht resultieren. So hat der Tricept IRB 940 von ABB einen Arbeitsbereich mit einem Radius von 800 mm bei einer Reichweite von 700 mm in vertikaler Richtung und einer Wiederholgenauigkeit von $\pm 0,002$ mm, wiegt jedoch 580 kg [ABB03]. Trotzdem wird ein solches System als Mehrkörpermodell aufgebaut, um zu untersuchen, ob das grundle-

gende Prinzip für die angestrebte Positionierung eines Sensors geeignet erscheint. Bei der Simulation stellt sich heraus, dass hybride Kinematiken weiter betrachtet werden müssen, da diese durch die serielle Komponente eine gute Skalierbarkeit der Reichweite bieten und schlank konstruiert werden können. Diese Erkenntnis bildet einen wichtigen Ausgangspunkt für die im folgenden Kapitel beschriebene modellbasierte Eigenentwicklung.

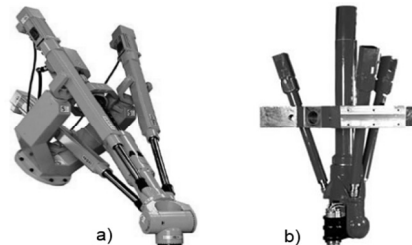


Bild 6: Übersicht der hybriden Kinematiken: a) IRB 940 [ABB03] b) T805 [PK142-ol]

Tabelle 1: Übersicht der Ergebnisse der modellbasierten Analyse

Anforderung System	Kinematik	Horizontale Reichweite \geq 2m	Vertikale Reichweite \geq 2m	Beweglichkeit in beengter Umgebung	Nutzlast \geq 5kg	ATEX
Universal Robots UR10	Seriell	-	-	-	+	-
Denso VS087	Seriell	-	-	-	+	-
ORC Explorer	Seriell	o	o	o	-	-
FESTO EXPT	Parallel	-	-	-	-	-
Para-Picker	Parallel	-	-	-	o	-
ABB IRB940	Hybrid	-	-	o	+	o
Tricept T805	Hybrid	-	-	o	+	o

+ Anforderung vollständig erfüllt
o Anforderung teilweise erfüllt
- Anforderung nicht erfüllt

3 Modellbasierte Eigenentwicklung

Die Vorgehensweise bei der modellbasierten Eigenentwicklung orientiert sich wieder stark an den Entwurfstechniken mechatronischer Systeme. Die Anwendungsszenarien bleiben dabei wie bei der modellbasierten Analyse ermittelt (vgl. Kapitel 2), die Anforderungen werden jedoch mittels der dort gewonnen Erkenntnisse ergänzt und präzisiert. So werden ausschließlich hybride Kinematiken betrachtet. Als Ergebnis der fachgebietsübergreifenden Konzipierung werden Prinziplösungen ermittelt. Dies geschieht wieder modellbasiert, um anhand eines Mehrkörpermodells eine erste Aussage über das theoretische Verhalten des Systems bzgl. Erreichbarkeit im Arbeitsraum und dynamischem Verhalten treffen zu können. Für die als theoretisch einsetzbar eingestuften Prinziplösungen wird eine fachgebietsspezifische Konkretisierung durchgeführt. Dabei werden die Erkenntnisse aus der Mehrkörpersimulation genutzt, um charakteristische Größen einzelner Komponenten, wie etwa den benötigten Winkel eines Gelenks oder den ungefähr benötigten Drehmomentbereich eines Antriebs, festzulegen [TOS11]. Im Bereich des Mechanikentwurfs werden dann die einsetzbaren Komponenten (Lager etc.) ermittelt, CAD-Modelle erstellt und Möglichkeiten der Fertigung berücksichtigt. Beim Elektronikentwurf werden die Aktoren und Sensoren ausgewählt [GSA+11]. Es gilt direkt Komponenten zu wählen, die aufgrund der Umgebungsbedingungen erforderlichen Schutzklassen, hier die ATEX-Zulassung für explosionsgefährdete Bereiche, aufweisen. Dabei müssen eine stetige Abstimmung und ein kontinuierlicher Informationsaustausch zwischen den einzelnen Fachdisziplinen erfolgen, um letztendlich ein funktionsfähiges Gesamtsystem zu erhalten. Es ist zudem erforderlich, das Mehrkörpermodell der Prinziplösung fortwährend zu aktualisieren und die Detaillierungstiefe unter Berücksichtigung von Effekten wie Reibung zu erhöhen [TOS11]. In der Modellierung sehr aufwendige Effekte, wie etwa das Spiel von Gelenken, werden jedoch vernachlässigt, da das System wie in den Anforderungen definiert keine erhöhten Genauigkeitsanforderungen hat. Es entsteht ein virtueller Prototyp des Positioniersystems, der das Verhalten des realen Systems so genau wie erforderlich abbildet. Als Konsequenz muss lediglich ein Vorseriensystem aufgebaut werden, da der Großteil der Problemstellen bereits anhand des Modells identifiziert wird und entsprechende Optimierungsmöglichkeiten untersucht werden können. In Bild 4 (rechte Seite) ist dargestellt, dass der TCP der hybriden Kinematik im Gegensatz zum Knickarmroboter bis in die Spitze des konusförmigen Arbeitsraumes gelangt und somit die geforderte Erreichbarkeit liefert.

Entgegen den Entwurfstechniken mechatronischer Systeme findet der Entwurf der benötigten Steuerungs- und Antikollisionsalgorithmen nicht parallel zu dem Mechanik- und Elektronikentwurf statt, sondern gesondert im Anschluss. Grund dafür ist die starke Abhängigkeit, insbesondere der benötigten Koordinatentransformationen, von der Geometrie des Positioniersystems. So haben etwa Änderungen der Gelenkpositionen oder der Winkel der Antriebe des parallelkinematischen Systems zueinander einen direkten Einfluss auf die Transformation. Da die Steuerungsalgorithmen wiederum auf Koordinatentransformationen basieren, können diese ebenfalls erst im Anschluss an den Sys-

tementwurf entwickelt werden. Sie müssen eine manuelle Bedienung des Systems mittels eines Joysticks ebenso ermöglichen wie einen vollautomatisierten Bewegungsablauf. Zudem wird für eine sichere manuelle Bedienung ein Antikollisionsalgorithmus benötigt. Dabei wird ein möglichst pragmatischer Ansatz angestrebt, um einen echtzeitfähigen Betrieb auf einer Industriesteuerung zu gewährleisten. Zudem wird keine Neuentwicklung von Algorithmen durchgeführt, sondern auf bereits bestehende Verfahren zurückgegriffen. Ausgangspunkt bildet dabei die Annäherung der Hindernisse durch geometrische Grundkörper, um die Kollisionserkennung auf einfache geometrische Berechnungen zu reduzieren (siehe Bild 7). Für das im Bild angegebene Beispiel sind die möglichen Schnittpunkte zwischen dem mit einem Zylinder und einem Quader angenäherten Hindernis und dem Positioniersystem, dessen lineare Komponente als einfach Gerade angenähert wird, zu berechnen. Für jeden dieser Punkte ist dann eine Plausibilitätsprüfung durchzuführen, indem der Abstand d_{us} zwischen dem Ursprungspunkt des Positioniersystems und der Sollposition des Werkzeugs mit dem Abstand d_{uk} zwischen demselben Ursprungspunkt und dem möglichen Kollisionspunkt verglichen wird (siehe Bild 7). Da es sich um eine lineare Bewegung zwischen dem Ursprung und der Sollposition handelt gilt dabei, dass bei $d_{uk} > d_{us}$ keine Kollision vorliegt.

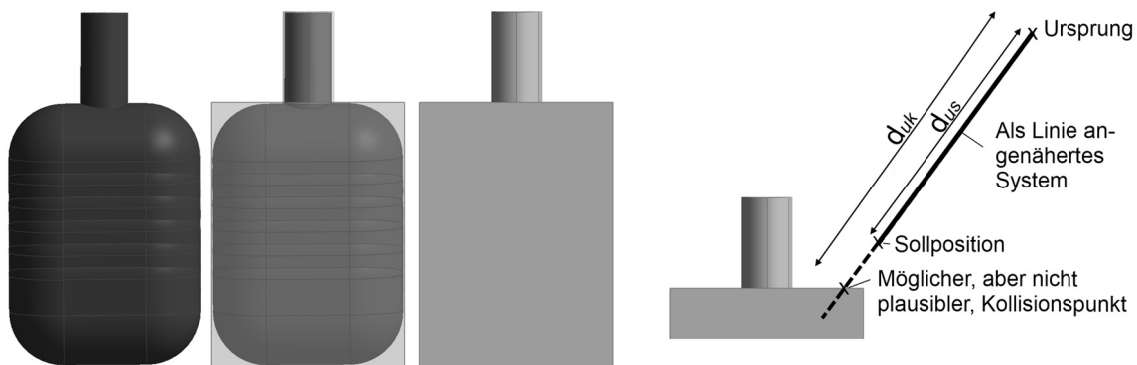


Bild 7: Annäherung von Hindernissen durch geometrische Grundkörper zur Kollisionserkennung

Die Implementierung erfolgt in MATLAB Simulink[®], um zum einen die Algorithmen sowie die Koordinatentransformationen im Rahmen einer Co-Simulation mit dem aus der fachgebietsspezifischen Konkretisierung resultierenden Mehrkörpermodell zu verifizieren. Zum anderen hat Simulink den Vorteil, dass dort erstellter Code mittels der Software Automation Studio Target[®] direkt auf einer Industriesteuerung des Herstellers B&R exportiert werden kann und so für eine Inbetriebnahme des realen Vorseriensystems genutzt werden kann. Dadurch entsteht eine Durchgängigkeit in der Werkzeugkette, die zu einer Reduzierung von Fehlerquellen führt (siehe Bild 8).

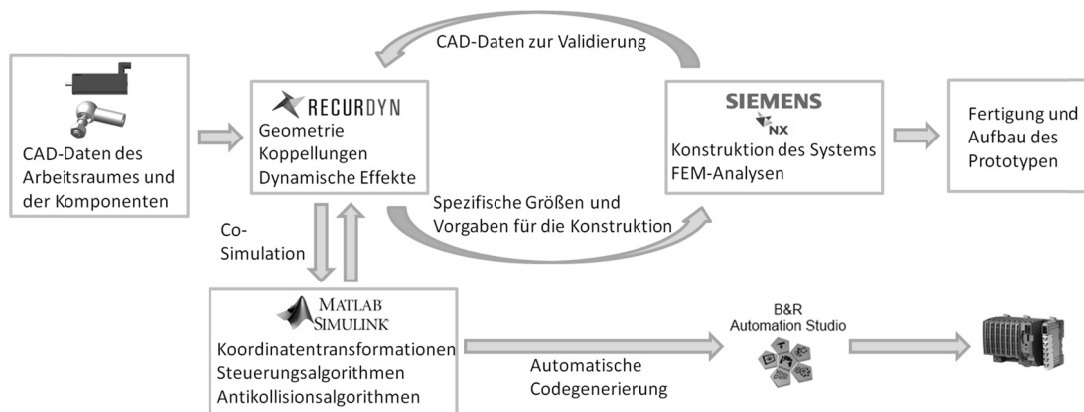


Bild 8: Vorgehen bei der modellbasierten Eigenentwicklung

In Bild 9 ist das Ergebnis einer Co-Simulation dargestellt. Dabei wird in Simulink eine Solltrajektorie vorgegeben, aus der mittels der ermittelten Transformation die Sollpositionen der Antriebe berechnet werden. Diese Größen werden dann über einen entsprechenden RecurDyn-Client an die Mehrkörpersimulation übergeben und wirken dort auf die aktiven Koppellemente des Modells ein. Es resultiert die im Bild dargestellte Trajektorie des TCP, wobei keine Abweichung zur Solltrajektorie auftritt. Zusätzlich wird der Antikollisionsalgorithmus simulativ getestet, wobei das bereits beschriebene Hindernis verwendet und in die geplante Bewegungsbahn des TCP hineingesetzt wird. Die Simulation zeigt, dass der Algorithmus wie gefordert die Kollisionsstellen erkennt und das Positioniersystem rechtzeitig stoppt. Durch den direkten Export der Algorithmen auf die beim Vorseriensystem eingesetzte Steuerung kann im Anschluss an die erfolgreich durchgeführte Co-Simulation die Inbetriebnahme beginnen.

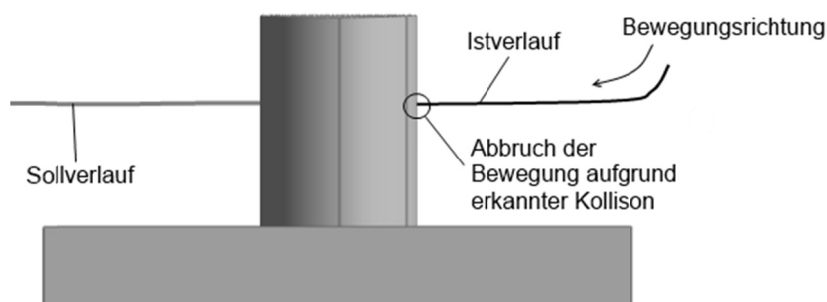


Bild 9: Überprüfung des Antikollisionsalgorithmus

4 Resümee und Ausblick

Ziel des Beitrags war es die Entwicklung eines Inspektionssystems von der Idee bis zum ersten Vorseriensystem darzustellen. Dabei sollte vor allem gezeigt werden, wie wichtig eine strukturierte Vorgehensweise unter Einsatz von Mehrkörpersimulationsmodellen ist. Dieses ermöglicht zum einen die Analyse bereits bestehender Systeme bezüglich ihrer Einsetzbarkeit unter Berücksichtigung des Arbeitsraumes. Zum anderen dient es der Verifizierung einer Eigenentwicklung sowohl von Seiten der Kinematik als auch

bezüglich des dynamischen Verhaltens. Dies führt dazu, dass nur noch ein Vorserien-system tatsächlich aufgebaut werden muss. Zudem können die benötigten Steuerungs- und Antikollisionsalgorithmen bereits am Mehrkörpermodell mittels Co-Simulation verifiziert werden, wodurch die Inbetriebnahme des realen Systems schneller durchzuführen ist und die Gefahr von Schäden aufgrund fehlerhafter Software stark sinkt.

Das entwickelte Positioniersystem basiert auf dem Prinzip hybrider Kinematiken. Ein Tripod wurde dabei um eine zusätzliche lineare Achse ergänzt, um den geforderten Arbeitsbereich abzudecken. Bei der Konstruktion wurden ausschließlich Kaufteile mit Explosionsschutzzulassung (Antriebe) und selbst gefertigte Komponenten verwendet. Anhand des aus den CAD-Daten der Konstruktion erstellten Mehrkörpermodells wurden echtzeitfähige Steuerung- und Antikollisionsalgorithmen simulativ verifiziert. Abschließend wurden diese dann direkt aus Simulink auf eine Steuerung des Herstellers B&R adaptiert und das reale Vorseriensystem mit manueller Bedienung in Betrieb genommen. In den folgenden Arbeitsschritten gilt es nun, den automatisierten Betrieb zu integrieren und am Prototyp zu optimieren.

Literatur

- [ABB03] ABB AUTOMATION GMBH: Datenblatt IRB 940 Tricept – Spezialroboter, Mai 2003
- [DR14-ol] DENSO ROBOTICS: New VS-087 – Specifications unter: <http://www.densorobot.co.uk/products/vs087.html>, 29.11.2014
- [FE14-ol] FESTO AG & CO. KG: Tripod EXPT Produktbeschreibung unter: http://www.festo.com/cms/de_de/13886.htm, 13.11.2014
- [FRI13-ol] FANUC ROBOTICS INDUSTRIEROBOTER: M-2iA Produktbroschüre, unter: <http://www.fanurobotics.de/~media/FRDE/Files/M-2iA%20Brosch%C3%BCre.ashx>, 2013, 29. November 2014
- [GSA+11] GAUSEMEIER, J., SCHÄFER, W., ANACKER, H., BAUER, F., DZIWOK, S.: Einsatz semantischer Technologien im Entwurf mechatronischer Systeme, HNI-Verlagsschriftenreihe, Band 294, Paderborn, 2011
- [IAP14-ol] INOTEC AP GMBH: ParaPicker® Roboter Produktbeschreibung unter: <http://www.ja2-gmbh.de/parapicker.html>, 13.11.2014
- [MSM12] MILUTINOVIC, M., SLAVKOVIC, N., MILUTINOVIC, D.: Kinematic Modeling of Tricept Based 5-Axis Machine Tools. Journal of Production Engineering, Band 2, Novi Sad, Serbien, 2012
- [OCR14-ol] OCROBOTICS: Explorer Range unter: http://www.ocrobotics.com/downloads/explorer_catalogue.pdf, 29.11.2014
- [PK14-ol] PKMTRICEPT SL: Tricept T805 Produktbeschreibung unter: <http://www.loxin2002.com/general-description>, 13.11.2014
- [RS10] RILL, G., SCHAEFFER, T.: Grundlagen und Methodik der Mehrkörpersimulation, Vieweg+Teubner Verlag, Auflage 1, Wiesbaden, 2010
- [SAS14-ol] SAS DIRECTINDUSTRY: Deltaförmig-Roboter / 4-Achsen / für Montage / Verpackung 3 kg, 1 130 mm | M-2iA/3SL unter: <http://www.directindustry.de/prod/fanuc-robotics/deltaformig-roboter-4-achsen-montage-verpackung-14532-1224559.html>, 13.11.2014

- [TOS11] TICHY, M., OESTERSÖTEBIER, F., SCHIERBAUM, T.: Entwurfstechnik intelligente Mechatronik, Internationales Forum Mechatronik 7, 2011
- [UR14-ol] UNIVERSAL ROBOTS A/S: Technische Spezifikationen UR10 Artikel Nr. 110110 unter: http://media1.limitless.dk/UR_Tech_Spec/UR10_DE.pdf, 29. November 2014
- [VDI2206] VDI-Richtlinie 2206: Entwicklungsmethodik für mechatronische Systeme, VDI-Verlag, Düsseldorf, 2004
- [WEC02] WECK, M., STAIMER, D.: Parallel Kinematik Machine Tools – Current State and Future Potential, CIRP Annals – Manufacturing Technology 51, 2002

Danksagung

Für die finanzielle Unterstützung des ZIM-Projekts KF3204701WZ3 sei dem Bundesministerium für Wirtschaft und Energie gedankt.

Autoren

M. Sc. Arne Rüting hat sein duales Bachelorstudium Elektrotechnik 2011 an der privaten Fachhochschule für Wirtschaft und Technik (FHWT) in Oldenburg abgeschlossen. Parallel absolvierte er seine Ausbildung zum Elektroniker für Betriebstechnik bei der IMA Klessmann GmbH in Lübbecke. Im Anschluss studierte er bis 2013 im Masterstudiengang Elektrotechnik mit der Spezialisierung Automatisierungstechnik an der Universität Paderborn. Seit 2014 ist er wissenschaftlicher Mitarbeiter der Projektgruppe Entwurfstechnik Mechatronik des Fraunhofer IPT in Paderborn. Sein Arbeitsgebiet liegt im Bereich der Regelungstechnik.

Dr.-Ing. Christian Henke studierte Elektrotechnik mit dem Schwerpunkt Automatisierungstechnik an der Universität Paderborn. Von 2004 bis 2009 war er wissenschaftlicher Mitarbeiter im Fachgebiet Leistungselektronik und Elektrische Antriebstechnik der Universität Paderborn. Dort promovierte er über das Thema »Betriebs- und Regelstrategien für den autonomen Fahrbetrieb von Schienenfahrzeugen mit Linearantrieb«. Von 2009 bis 2011 war er zudem wissenschaftlicher Mitarbeiter am Lehrstuhl Regelungstechnik und Mechatronik der Universität Paderborn. Dort koordinierte er unter anderem Auftragsforschungs- und Verbundprojekte mit Unternehmen, die anspruchsvolle mechatronische Produkte entwickeln. Seine Arbeitsschwerpunkte liegen im Bereich der modellbasierten Entwicklung mechatronischer Systeme, des Regelungsentwurfs und der industriellen Automatisierungstechnik. Seit April 2011 leitet er die Abteilung »Regelungstechnik« der Projektgruppe Entwurfstechnik Mechatronik des Fraunhofer IPT.

Dr.-Ing. Andreas Warkentin studierte an der Altäischen Staatlichen Technischen Universität Barnaul und promovierte 1985 am Lehrstuhl „Theorie der Mechanismen und Maschinen“ der Staatlichen Polytechnischen Universität Sankt Petersburg. Von 1986 bis 1997 war er Dozent am Lehrstuhl „Theorie der Mechanismen und Maschinen“ der Altäischen Staatlichen Technischen Universität. Im Anschluss arbeitete er bis 2001 als wissenschaftlicher Mitarbeiter am Mechatronik Laboratorium Paderborn (MLaP) der

Universität Paderborn. In den Jahren 2001 bis 2010 folgte eine Tätigkeit als Projekt- und Entwicklungsingenieur bei der iXtronics GmbH, einer Ausgründung der Universität Paderborn. Seit Juni 2011 ist er wissenschaftlicher Mitarbeiter der Projektgruppe Entwurfstechnik Mechatronik des Fraunhofer IPT in Paderborn. Seine Arbeitsschwerpunkte sind Mehrkörpersystemmodellierung und Modellbildung komplexer mechatronischer Systeme.

Prof. Dr.-Ing. habil. Ansgar Trächtler, geb. 1964, hat an der Universität Karlsruhe (TH) Elektrotechnik studiert und 1991 am Institut für Regelungs- und Steuerungssysteme bei Prof. Föllinger promoviert. Von 1992 bis 1998 war er als wissenschaftlicher Assistent am Institut für Meß- und Regelungstechnik (Prof. Mesch) der Universität Karlsruhe tätig, habilitierte sich 2000 und erhielt die Venia Legendi für das Fach „Meß- und Regelungstechnik“. Für seine wissenschaftlichen Arbeiten erhielt er 1998 den Eugen-Hartmann-Preis der GMA und den Messtechnik-Preis des AHMT (Arbeitskreis der Hochschullehrer für Messtechnik e.V). Von 1998 bis 2004 war er bei der Robert Bosch GmbH tätig und leitete zuletzt den Bereich Vorausentwicklung Fahrwerksysteme. Seit November 2004 ist er Professor und Leiter des Lehrstuhls für Regelungstechnik und Mechatronik der Universität Paderborn. Außerhalb der Universität engagiert sich Prof. Trächtler in der VDI/VDE-Gesellschaft für Meß- und Automatisierungstechnik (GMA) und in der International Federation of Automatic Control (IFAC).

Mechatronisches Roboter-System zum Rückbau von Reaktordruckbehältereinbauten

Simon Berger, Dominik Simon, Peter Stich, Prof. Dr.-Ing. Gunther Reinhart
Fraunhofer Institut für Werkzeugmaschinen und Umformtechnik, IWU
Beim Glaspalast 5, 86153 Augsburg
Tel. +49 (0) 821 / 56 88 396, Fax. +49 (0) 821 / 56 88 350
E-Mail: simon.berger@iwu.fraunhofer.de

Zusammenfassung

In den kommenden Jahren wird eine Vielzahl von kerntechnischen Anlagen zum Rückbau anstehen. Besonders herausfordernd ist dabei die Zerlegung des Reaktordruckbehälters mit dessen radioaktiv aktivierten Einbauten. Diese werden bisher manuell unter Wasser gehandhabt und zerlegt, was kosten- und zeitintensiv ist. Daher wird der Einsatz von Robotern zur Automatisierung dieser Tätigkeiten erforscht. Zentrale Herausforderungen sind dabei der Aufbau und die Qualifizierung des Robotersystems zur Arbeit unter Wasser, die Integration eines Kamerasystems mit Bilderkennung und das Entwickeln einer Ansteuerung mit einem haptischen Bediengerät.

Schlüsselworte

Unterwasser Robotik, Bildverarbeitungssysteme, Teleoperation, Haptisches Kraftrückkopplungssystem

Mechatronic Roboter-Systems for Dismantling of Reactor Pressure Vessel Internals

Abstract

Germany faces the dismantling of many nuclear power plants in the near future. One of the major challenges is the disassembly of the nuclear pressure vessel and its internals. For the time being, this step is done manually, using different cranes and grippers. Since this method is very time and money consuming new measures have to be taken into account. Thus, an underwater robot cell is planned in order to automate the manipulation of the internals. For this, the first step is to get the robot ready for the under-water application. Furthermore a machine vision system and a force feedback master arm has to be developed.

Keywords

Underwater robotics, Camera Vision System, Teleoperation, Haptic Force Feedback System

Dieser Beitrag ist in der Online-Version
leider nicht verfügbar.

Modellierung eingebetteter Systeme

AMALTHEA¹ – Eine durchgängige Entwicklungsplattform für die modellgetriebene Entwicklung automobiler eingebetteter Systeme

M.Sc. Lukas Krawczyk, M.Eng. Robert Höttger, Prof. Dr. Carsten Wolff

Fachhochschule Dortmund

Emil-Figge-Straße 42, 44227 Dortmund

Tel. +49 (0) 231 / 91 12 549, Fax. +49 (0) 231 / 91 12 548

E-Mail: lukas.krawczyk@fh-dortmund.de

Dipl.-Inform. (FH) Christopher Brink

Fachgebiet Softwaretechnik, Heinz Nixdorf Institut, Universität Paderborn

Zukunftsmeile 1, 33102 Paderborn

Tel. +49 (0) 5251 / 54 65 156, Fax. +49 (0) 5251 / 60 36 15

E-Mail: Christopher.Brink@upb.de

M.Sc. Daniel Fruhner

Behr-Hella Thermocontrol GmbH

Hansastraße 40, 59557 Lippstadt

Tel. +49 (0) 2941 / 66 64 05, Fax. +49 (0) 2941 / 66 47 64 05

E-Mail: daniel.fruhner@bhtc.com

Zusammenfassung

Ein großer Teil der Innovation in automobilen Systemen wird durch Software erzielt, die aufgrund ihrer steigenden Komplexität neue Herausforderungen in der Entwicklung mit sich bringen. Um diese zu bewältigen sind in aller Regel zahlreiche, oft proprietäre, Werkzeuge involviert, die für eine gemeinsame Verwendung erst aufwändig zu einer Werkzeugkette integriert werden müssen. Die AMALTHEA Tool Plattform unterstützt die Integration solcher Werkzeuge in eine Werkzeugkette und ermöglicht über gemeinsame Datenmodelle den automatischen Datenaustausch zwischen diesen. In diesem Beitrag werden die Konzepte anhand eines industriellen Anwendungsfalls überprüft. Hierfür wird der Anwendungsfall mithilfe einer AMALTHEA Werkzeugkette realisiert.

Schlüsselworte

Modellbasierte Entwicklung, eingebettete Systeme, durchgängige Werkzeugkette, Automotive, Austauschformat, Multicore

¹ Diese Arbeit ist im ITEA2 EU-Projekt AMATHEA entstanden, an dem u.a. die Universität Paderborn, die Fachhochschule Dortmund und BHTC beteiligt waren und wurde durch das BMBF gefördert (FKZ: 01IS11020J).

AMALTHEA² – A Continuous Model-Based Development Platform for Automotive Embedded Systems

Abstract

A major part of innovation in automotive systems is achieved by software. Accordingly, software becomes more and more complex, which leads to new challenges during the development process. Usually, these challenges are addressed by many different and often proprietary tools, which require a manual and inconvenient integration into a tool chain. The AMALTHEA Tool Platform supports the integration of such tools as well as automatic data exchange using specific data models. Based on an industrial use case, we review the concepts of AMALTHEA within this paper. For this purpose the use case has been realized using the AMALTHEA Tool Platform.

Keywords

Model-based development, embedded systems, continuous tool-chain, automotive, exchange format, multicore

² The work leading to this results has been funded by the BMBF (Grant 01IS11020J) within the ITEA2 EU-Project AMALTHEA and involved, among others, the University Paderborn, Dortmund University of Applied Sciences and Arts and BHTC.

1 Einführung

Durch immer fortschrittlichere Fahrerassistenzsysteme und andere Systeme in heutigen Fahrzeugen, müssen auch die zugrundeliegenden Steuergeräte stetig leistungsfähiger werden. Hierbei stoßen gängige Single-Core Systeme mehr und mehr an ihre Grenzen, sodass in der Industrie vermehrt Multicore-Systeme eingesetzt werden. Dieser Umstieg bringt jedoch neue Herausforderungen [ZD14] in der Entwicklung mit sich. So erfordert die Ausführung von Software, bedingt durch die steigende Anzahl von Kernen, Erweiterungen an bestehenden Techniken und Werkzeugen, wie beispielsweise die Anpassung der Parallelisierung der vorwiegend sequentiellen Aufrufe und die Verteilung der ausführbaren Elemente auf die Kerne (Massive Parallelität).

Weiterhin wird im Entwicklungsprozess von automobilen Steuergeräten eine Vielzahl an Werkzeugen eingesetzt, wobei jedes eine spezifische Sicht auf das System hat oder in einem bestimmten Entwicklungsschritt eingesetzt wird. Dabei verwendet jedes Werkzeug jeweils bestimmte Daten, die verarbeitet und ggf. verändert werden. Diese veränderten Daten bilden anschließend die Basis für weitere Entwicklungsschritte, sodass eine Werkzeugkette entsteht. Hierbei müssen die Daten zwischen Werkzeugen oftmals aufwändig, aufgrund unterschiedlicher Datenformate, manuell übertragen werden.

Um den Herausforderungen zu begegnen, wurde im Projekt AMALTHEA [BJ14] die auf Eclipse-basierte AMALTHEA Tool Plattform entwickelt. Diese wurde unter einer Open Source Lizenz veröffentlicht und unterstützt den Entwicklungsprozess von der Anforderungsanalyse bis hin zur Code Generierung. Die Plattform umfasst dabei u.a. gemeinsame Modelle für die Beschreibung der Hardware, der Software, von Varianten sowie *Constraints* (Abhängigkeiten). Darüber hinaus stehen für die Plattform verschiedene AMALTHEA Werkzeuge zur Verfügung, die speziell auf die Modellierung von Multicore-Systemen ausgelegt wurden. Ziel der Plattform war es, durch gemeinsame Modelle den Aufwand des Datenaustauschs zwischen Entwicklungswerkzeugen zu reduzieren oder vollständig zu automatisieren. Weiterhin können beliebige Werkzeuge, sowohl freie als auch kommerzielle, in die Plattform integriert werden, sodass hierdurch eine individuelle Werkzeugkette zusammengestellt werden kann. Mittlerweile wird die Plattform von verschiedenen Zulieferern und OEMs für den Austausch von Entwicklungsartefakten eingesetzt.

In diesem Papier werden die entwickelten Konzepte innerhalb der AMALTHEA Tool Plattform anhand eines industriellen Anwendungsfalls präsentiert. Als industrieller Anwendungsfall dient ein HVAC (*Heating, Ventilation and Air Conditioning*) System des Unternehmens Behr-Hella Thermocontrol (BHTC), das mit der AMALTHEA Tool Plattform sowie den darin entwickelten Werkzeugen prototypisch umgesetzt wird. Hierbei werden Schwerpunkte auf die Bereiche Variantenhandling, Partitionierung, Mapping und Codegenerierung gelegt, die mithilfe von AMALTHEA-eigenen Werkzeugen realisiert werden.

2 AMALTHEA Tool Plattform

Die *AMALTHEA Tool Plattform*³ ist eine auf Eclipse basierende Plattform zur Erstellung von Werkzeugketten für die automobiler Softwareentwicklung. Die Plattform unterstützt die Entwicklung eingebetteter automobiler Multicore-Systeme. Hierfür wird ein Entwicklungsprozess definiert, der die Aktivitäten sowie die Ergebnisartefakte beschreibt. Weiterhin wurden für den nahtlosen Datenaustausch zwischen den Werkzeugen und Entwicklungsschritten gemeinsame Datenmodelle innerhalb der Plattform spezifiziert, die u.a. die Beschreibung der Hardware, Software, Abhängigkeiten und Varianten ermöglichen. Obwohl die Plattform rudimentäre Werkzeuge für die Entwicklung in Form von Open Source Plugins enthält, lassen sich einzelne oder alle Werkzeuge sowohl durch kommerzielle als auch Open-Source-Drittsoftware austauschen. Diese werden über definierte Schnittstellen mit der Plattform verbunden, sodass der Aufbau einer individuellen Werkzeugkette durch einen Anwender (z.B. Zulieferer oder OEM) ermöglicht wird, wobei der Datenaustausch zwischen den Werkzeugen (semi-) automatisch durchgeführt wird.

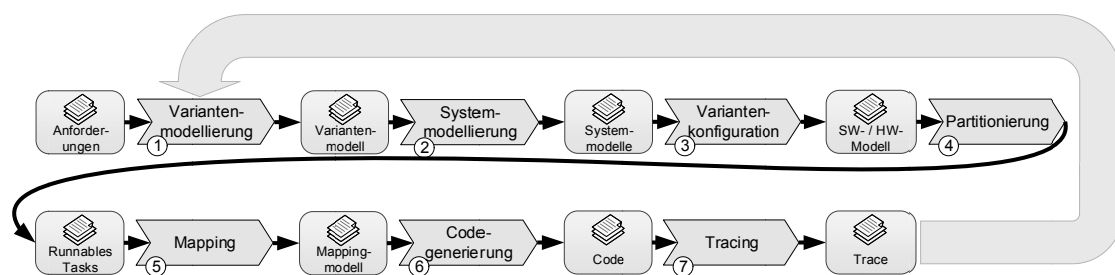


Bild 1: Entwicklungsprozess innerhalb der AMALTHEA Tool Plattform

Bild 1 zeigt den Entwicklungsprozess einer durch die *AMALTHEA Tool Plattform* erzeugten Werkzeugkette. Darin werden Artefakte durch Rechtecke mit abgerundeten Kanten repräsentieren, Aktivitäten durch blaue Pfeile und Iterationen durch gelbe Pfeile. In dieser Werkzeugkette wurde auf Aktivitäten zur Simulation verzichtet, welche parallel oder alternativ zum Tracing durchgeführt werden können.

Der Prozess startet mit der Modellierung von Varianten (Schritt 1) auf Basis eines Anforderungsdokuments, welches entweder durch einen Auftraggeber definiert oder mithilfe eines eigenen Werkzeuges erstellt werden kann. Durch den Fokus auf automobiler eingebettete Systeme werden sowohl Hardware- wie auch Softwarevarianten im Modell definiert sowie deren Abhängigkeiten berücksichtigt [BKP+14]. Softwarevarianten werden durch spätere Prozessschritte um realisierende Artefakte wie beispielsweise Softwarekomponenten sowie notwendige Parameter erweitert. Das Modell zur Beschreibung von Hardwarevarianten kann verschiedene Hardwareplattformen sowie deren Varianten enthalten, die während der Systemmodellierung um spezifische Beschreibungen angereichert werden.

³ www.amalthea-project.org/

Nach der Modellierung der Varianten folgt im zweiten Schritt die Systemmodellierung durch den Entwickler, bei der sowohl die Software als auch Hardware für alle Varianten modelliert werden. Dabei wird das System in Komponenten aufgeteilt, z.B. Sensoren und Aktuatoren, und die Struktur sowie das Verhalten der Softwarekomponenten modelliert. Hierfür können die vorhandenen AMALTHEA-Werkzeuge sowie externe Werkzeuge, wie beispielsweise Matlab Simulink, für die Modellierung des Verhaltens der Softwarekomponenten verwendet werden. Weiterhin ist es möglich, direkt C-Code zu verwenden. Die Komponenten werden nach ihrer Fertigstellung mit dem zuvor definierten Variantenmodellen verbunden, womit die Grundlage für die Konfiguration geschaffen wird.

Während der Variantenkonfiguration (Schritt 3) werden sowohl notwendige Software- als auch Hardwarevarianten durch den Entwickler ausgewählt, wobei die Abhängigkeiten Werkzeugseitig aufgelöst werden. Sofern Anforderungen der Software an die Hardware durch die gewählte Hardwareplattform bzw. Variante nicht erfüllt werden, wird hierauf durch das Werkzeug hingewiesen. Nach der erfolgreichen Produktkonfiguration wird aus den im Variantenmodell hinterlegten Artefakten ein Softwarekomponentenmodell sowie eine Hardwareplattformbeschreibung für das spezifische Produkt u.a. in Form eines initialen *AMALTHEA Software Modells* generiert. Nach diesem Schritt wird, unter Berücksichtigung der Hardwareplattform, eine initiale Bestimmung der Laufzeiten durchgeführt, die zur weiteren Verwendung innerhalb der Werkzeugkette in das *AMALTHEA Software Modell* übertragen wird.

Aus der Verhaltensbeschreibung der Softwarekomponenten werden während der Partitionierung Abhängigkeitsgraphen erzeugt, die aus atomaren Aufgaben (*Runnables*) und ihren Lese-/ Schreib Abhängigkeiten bestehen. Um eine parallele Verarbeitung dieser Aufgaben zu ermöglichen, fasst die Partitionierung (4) diese Aufgaben anhand verschiedener automatisierter Analysen in *Tasks* zusammen. Sie stellt somit den ersten Schritt zur Parallelisierung der modellierten Applikationen dar. Insbesondere werden für diese Schritte die Abhängigkeiten der *Runnables* untereinander analysiert, zyklische Abhängigkeiten aufgelöst und Parallelität anhand gerichteter azyklischer Graphen untersucht. Ziel der Analysen ist eine automatisierte und möglichst effiziente Verteilung der *Runnables* auf *Tasks* unter Berücksichtigung von Aktivierungsraten, Abhängigkeiten (Label-Zugriffe) und Gewichten (Instruktionen). Eine möglichst effiziente Verteilung wird durch eine gleichmäßige Lastenverteilung sowie eine möglichst hohe Reduktion der Ausführungszeit gebildet.

Die Verteilung der Software auf die gewählte Hardwareplattform wird im fünften Schritt (Mapping) bestimmt. Hierbei liegt das Ziel darin, eine Abbildung der einzelnen Elemente der Software (Aufgaben, Daten, Kommunikationen, ...) auf die korrespondierenden Elemente der Hardware (Prozessoren, Speicher, Netzwerke, ...) zu finden. Dabei soll nicht nur eine gültige sondern auch eine optimalitätsnahe Lösung gefunden werden, für die beispielsweise die Ausführungszeit oder der Energieverbrauch minimiert werden. Das Ergebnis dieses Arbeitsschrittes wird in Form eines Mapping Modells zusam-

mengefasst, dass sämtliche ermittelten Allokationen seitens der Software auf die Hardware beinhaltet.

Mit Bestimmung der Verteilung wird die Grundlage für den sechsten und siebten Schritt geschaffen. Bei der Code-Generierung werden die Code-Generatoren der Modellierungswerkzeuge für den Verhaltenscode mit einem integrierten Generator für betriebs-systemspezifischen Code kombiniert. Letzterer erlaubt beispielsweise die Umsetzung der zuvor ermittelten Allokationen. Durch Einbindung plattformspezifischer Code-Templates lässt sich der finale Code für die Zielplattform erzeugen, der über Compiler und Flasher auf die Hardwareplattform übertragen und auf dieser ausgeführt werden kann. Über das in der Werkzeugkette integrierte Tracing Werkzeug lässt sich anschließend die Ausführung der Software auf der Hardware in Form von Traces aufzeichnen und nachverfolgen. Diese können gleichermaßen für die Verifikation des Softwareverhaltens, als auch zur Softwareoptimierung genutzt werden, in dem z.B. andere Verteilungsstrategien beim Mapping oder leistungsstärkere Plattformen bei der Variantenkombi-figuration gewählt werden.

3 Demonstrationsaufbau

Zur Überprüfung der Projektergebnisse wurde die AMALTHEA Tool Plattform verwendet, um eine Werkzeugkette für die Entwicklung eines industriellen HVAC Systems (*Heating, Ventilation, and Air Conditioning*) [Ama14-ol] des Projektpartners Behr-Hella Thermocontrol (BHTC) prototypisch umzusetzen. Das HVAC beinhaltet einen Lüfter und einer Reihe von Klappen zur Steuerung des Luftstroms und verfügt über verschiedene Bedienelemente zur Ansteuerung des Systems, z.B. Regler zur Luftstromverteilung, der Gebläsestärke und des Frischlufteinlaßes. Darüber hinaus ermöglicht es die Temperaturen für die Fahrer- und Beifahrer-Zonen unabhängig voneinander einzustellen, die bei Bedarf durch das HVAC Steuergerät automatisch geregelt werden. Weiterhin verfügt das System über verschiedene Varianten, sodass das System entweder über ein Tablet oder über Bedienelemente, wie sie sich standardmäßig in einem Fahrzeug finden, gesteuert werden kann. Zudem kann die Ansteuerung der Mechanik entweder direkt, über das verwendete Steuergerät, oder über ein weiteres externes Steuergerät erfolgen. Durch die Vereinigung von Regelungs- und Kontrollflussberechnungen sowie die Ansteuerungsaufgaben und verschiedene Varianten verfügt das System somit über verschiedene Aspekte die während der Umsetzung mit der erstellten Werkzeugkette realisiert werden müssen.

Der konzeptuelle Aufbau einer Variante des HVAC Systems sowie seine Sensoren und Aktuatoren sind in Bild 2 aufgeführt. Das HVAC (u.l.) verfügt über die gesamte Mechanik (Lüfter und Klappen) zur Steuerung des Luftmassenstroms und verzichtet dabei auf die Heiz- und Kühlelemente. Als Steuergerät wird ein Freescale Board mit einem MPC5668G SoC verwendet. Es verfügt über zwei heterogene Kerne und dient sowohl für die Regelung des Luftstroms als auch für die Ansteuerung der Klappenmotoren. Die

Benutzereingaben finden in dieser Variante über ein Tablett statt, das die Signale mittels W-Lan Gateway über eine SPI Schnittstelle an das Freescale Board weiterreicht.

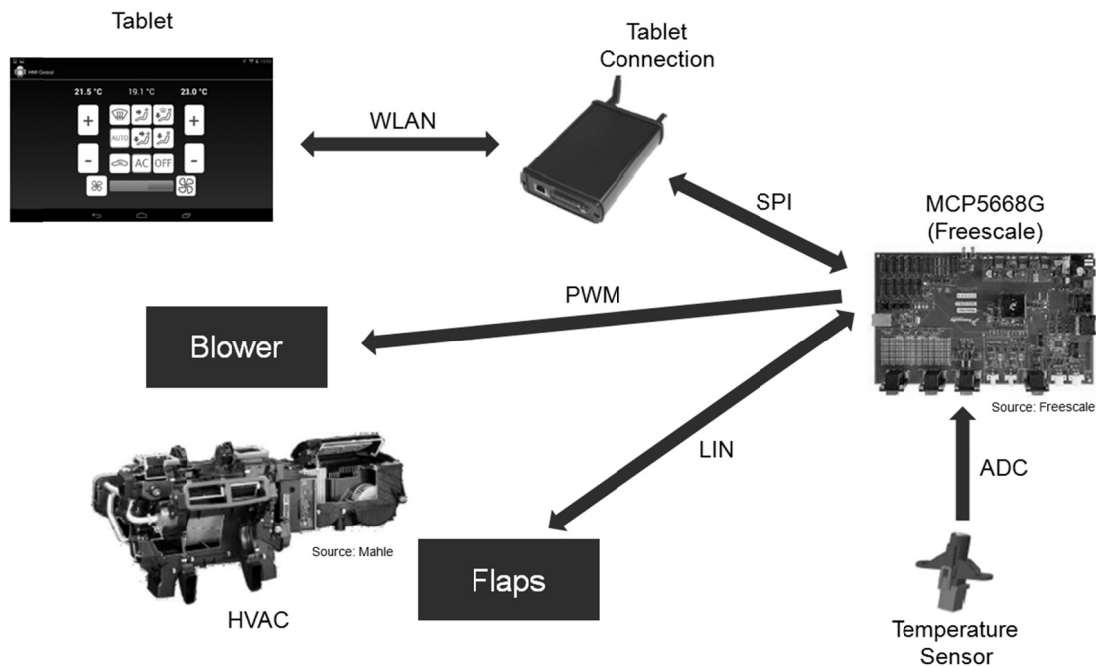


Bild 2: Konzeptioneller Aufbau einer Variante des HVAC Systems (Quelle: BHTC)

4 Entwicklung mit einer AMALTHEA Werkzeugkette

Die Entwicklung der Software für das HVAC Steuergerät wurde mit einer Werkzeugkette durchgeführt, die mit Hilfe der AMALTHEA Tool Plattform erzeugt wurde. Hierbei wurden sowohl in die Plattform integrierte Werkzeuge wie auch Werkzeuge von Drittanbietern eingesetzt, die über definierte Schnittstellen angebunden wurden. Zur Spezifikation von Anforderungen wurde das Werkzeug ProR⁴ eingesetzt, das die Beschreibung von textuellen Anforderungen ermöglicht. Für die Modellierung der Softwarearchitektur wurde Yakindu CoMo und für das Softwareverhalten Yakindu Statecharts⁵ sowie Matlab Simulink⁶ eingesetzt.

Im Folgenden werden die Modellierungsschritte des HVAC Systems vorgestellt, wobei dabei ein Fokus auf die Werkzeuge der AMALTHEA Plattform gelegt wird.

⁴ Vgl. <http://eclipse.org/rmf/pror/>

⁵ Vgl. <http://statecharts.org/>

⁶ Vgl. <http://de.mathworks.com/products/simulink/>

4.1 Variantenmodellierung

Auf Basis der definierten Anforderungen werden gemeinsame und variable Systemteile für die Software in Form eines sog. Feature-Modells [KCJ+90] (Bild 3a) und für die Hardwareplattformen in Form einer AMALTHEA-eigenen Hardwarebeschreibung definiert, die auch die Modellierung von Hardwarevarianten [BKP+14] erlaubt (Bild 3b).

Das HVAC System besteht auf der Softwareseite aus verschiedenen notwendigen Merkmalen (*Mandatory*), die alle HVAC-Varianten gemeinsam haben. Diese beinhalten u.a. die Steuerung für die Lüftung. Weiterhin verfügt das HVAC über die beiden in Kapitel 3 beschriebenen Alternativen (*Alternative*). Jedes Element im Modell enthält die notwendigen Softwarekomponenten, die während der Systemmodellierung spezifiziert werden, sowie Verbindungen untereinander, die über Ports spezifiziert werden (siehe Bild 3c).

Die Hardware des HVAC besteht aus verschiedenen möglichen Plattformen (MPC5668G sowie Freescale i.MX6) sowie deren Varianten. Diese werden in einem hierarchischen Modell spezifiziert, das einen Fokus auf die einzelnen Hardwareelemente sowie deren Eigenschaften legt. Varianten der Plattformen (*Variationpoint*) geben an, welche Elemente bzw. Eigenschaft hinzukommen oder entfernt werden müssen. Beim MPC5668G wird über eine Variante ein weiterer Bus vom Typ *SPI* hinzugefügt, der für die Verbindung mit dem Tablet notwendig ist.

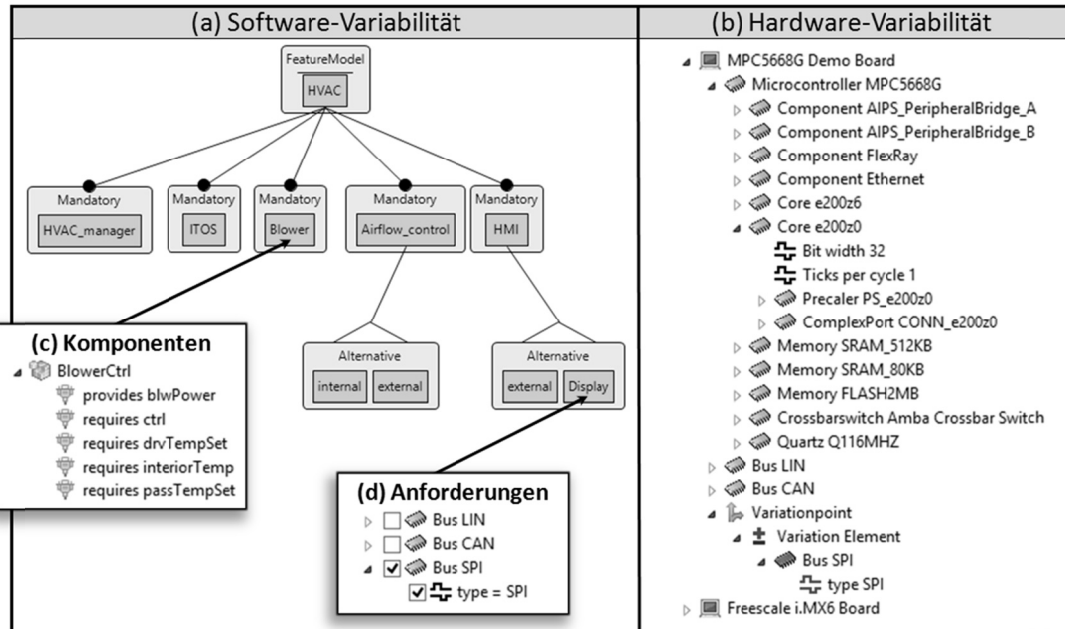


Bild 3: Teilbereiche der Variantenmodellierung des HVACs

Um Anforderungen der Software an die Hardware zu berücksichtigen, werden diese ebenfalls in einer hierarchisch aufgebauten Struktur zu jedem Merkmal spezifiziert. Dabei werden sowohl die notwendige Hardwareelementstruktur sowie die Hardwareeigenschaft berücksichtigt. Für das HVAC wurde auf diese Weise spezifiziert, dass für die

Variante *Display* eine Hardwareplattform benötigt wird, die über eine Schnittstelle (BUS) zur Ansteuerung verfügt (siehe Bild 3d). Diese Modellierungstechnik ermöglicht es, während der Konfiguration eines HVACs die Abhängigkeiten der gewählten Varianten zu berücksichtigen und die Kompatibilität auf Basis der definierten Anforderung zu prüfen.

4.2 Systemmodellierung

Zur Modellierung des Systems sowie aller Varianten wurden für das HVAC neun Softwarekomponenten mithilfe von Yakindu CoMo spezifiziert. Diese wurden zudem um das Komponentenverhalten erweitert. Hierzu wurde der Kontrollfluss, also die Verarbeitung der Benutzereingaben wie Temperatureinstellungen oder Luftstromereinstellungen und die Aktualisierung des Betriebszustandes und der Sensoren, über Yakindu Statecharts spezifiziert. Die automatische Regelung des Luftstroms wurde als Datenflussmodell in Matlab Simulink realisiert. Zudem wurden die Ein- und Ausgabeknoten nach der Spezifikation auf die jeweiligen Komponentenports abgebildet. Die Softwarekomponenten wurden im Anschluss den jeweiligen Varianten im Softwarevariantenmodell zugeordnet, um diese für die Konfiguration zu nutzen. Die Hardwareplattformen wurden in Form mehrerer Hardwaremodelle spezifiziert, die ebenfalls dem Variantenmodell zugeordnet wurden.

4.3 Variantenkonfiguration

Im Anschluss an die Systemmodellierung wird durch die Konfiguration (Auswahl) der Varianten ein spezifisches HVAC System erzeugt. Hierzu werden zunächst die Softwarevarianten ausgewählt, wobei das Werkzeug den Benutzer dahingehend unterstützt, dass gegenseitige Ausschlüsse sowie notwendige Abhängigkeiten angezeigt werden. Anschließend wird ebenfalls die gewünschte Hardwareplattform gewählt, woraufhin das Werkzeug die Anforderungen der Software an die Hardware prüft.

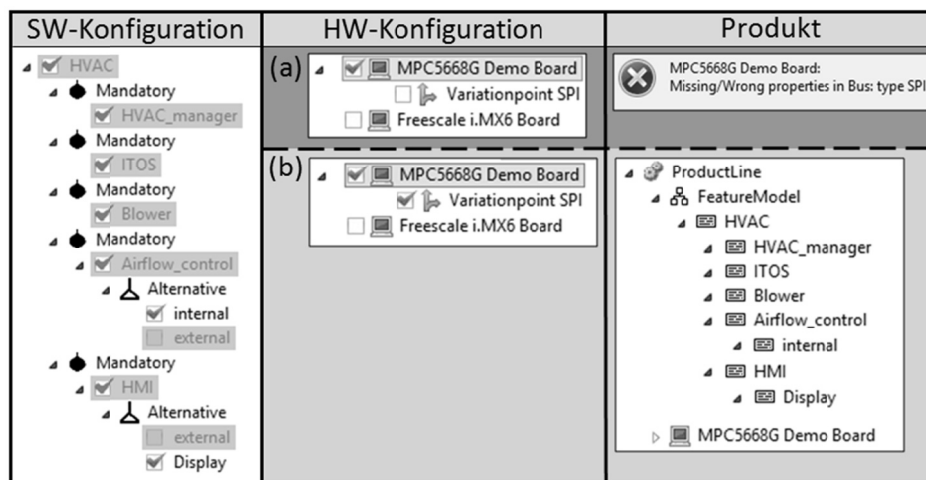


Bild 4: Schritte der Hardware/Software-Variantenkonfiguration

Sollte die gewählte Plattform diesen Anforderungen nicht entsprechen, wird eine Fehlermeldung angezeigt (siehe Bild 4a). Anderenfalls wird das konfigurierte System erzeugt. Hierzu werden alle notwendigen Softwarekomponenten in Form eines Komponentenmodells mit allen Komponentenverbindungen generiert. Die Komponenten enthalten zudem das zuvor spezifizierte Verhalten. Weiterhin wird das Hardwaremodell für die gewählte Plattform erzeugt.

4.4 Partitionierung

Sobald die Softwarekomponenten generiert wurden, können *Runnables* abgeleitet werden, die innerhalb des Softwaremodells atomare Prozesse repräsentieren. Bevor die *Runnables* allerdings auf *Tasks* verteilt werden können, müssen ihre Laufzeiten mittels *Profiling* bestimmt werden. Anschließend werden sie bezüglich ihrer Aktivierungsreferenzen gruppiert, ihre Abhängigkeiten durch gemeinsame Speicherzugriffe identifiziert und Zyklen innerhalb der Abhängigkeitsgraphen aufgelöst. Dabei entsteht ein zusätzliches *Constraints-Modell*, welches die *Task-Abhängigkeiten* beinhaltet und Graphen repräsentiert. Anschließend stehen zwei unterschiedliche Partitionierungsverfahren zur Verfügung die unterschiedliche Methoden zur Verteilung bieten.

Das entwickelte LGP (*Local Graph Partitioning*) Verfahren identifiziert die kritischen Pfade der Graphen, bildet daraus die ersten Partitionen und teilt die restlichen *Runnables* anhand der Abhängigkeiten weiteren Partitionen zu. Ein kritischer Pfad beschreibt hierbei nach [KA96] eine Menge an Knoten und Kanten die einen Pfad von einem Eingangsknoten zu einem Ausgangsknoten konstituieren, bei dem die Summe aller Berechnungs- und Kommunikationskosten maximal ist. Nach [CSR+01] beschreibt der kritische Pfad eines Graphen die untere Grenze in Bezug auf die Gesamtausführungszeit des Graphen bzw. in Bezug auf die Berechnung aller Knoten innerhalb des Graphen. Die Methodik ermöglicht also eine minimale Gesamtlaufzeit des Systems, allerdings können sehr viele Partitionen entstehen da das LGP Verfahren die Anzahl der Partitionen nicht limitiert.

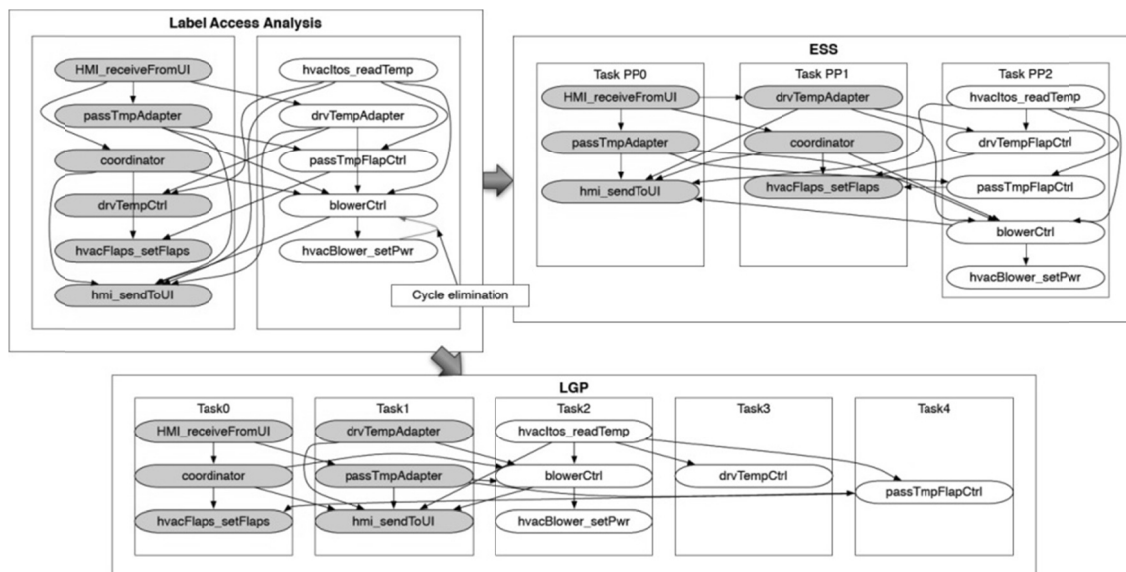


Bild 5: HVAC Partitionierung LGP und ESS Verfahren

Um die Anzahl der Partitionen zu limitieren, bietet das entwickelte ESS (*Earliest Start Scheduling*) Verfahren eine Alternative, bei der die *Runnables* anhand ihrer frühesten Start-Zeitpunkte einer vordefinierten Anzahl an Partitionen zugeordnet werden. Konkrete Umsetzungen der Verfahren können unter [HIK13] und [HI15] nachgelesen werden.

Im Rahmen des HVAC Beispiels soll Bild 5 einen groben Überblick über die Partitionierungsverfahren geben. Hierbei wird die minimale Laufzeit des HVAC Modells durch die LGP Partitionierung mit 5 Partitionen (siehe Bild 5 unten) erreicht und eine gleichmäßigere Lastenverteilung durch das ESS Verfahren mit 3 Partitionen (oben rechts). Die Aktivierungsgruppierung wird anhand der farblich hinterlegten *Runnables* deutlich, indem nur *Runnables* gleicher Aktivierungen und somit gleicher Farbmuster gruppiert werden.

4.5 Mapping

Die erzeugten Partitionen (*Tasks*) aus dem Partitionierungsschritt bilden gemeinsam mit den beiden Kernen des Hardware Modells [KK13] die Grundlage für den Mapping Prozess. Zur Bestimmung der Verteilung muss allerdings vorab eine der drei Verteilungsstrategien ausgewählt werden. Da bei dem HVAC sowohl Kontroll- als auch Datenfluss periodisch, d.h. beispielsweise alle 100ms aufgerufen werden, sind Strategien zur reinen Laufzeitminimierung der Perioden ineffektiv. Sinnvoller wäre es stattdessen den Energieverbrauch des Systems unter Gewährleistung der 100ms Periode zu minimieren, indem z.B. die Strategie *ILP⁷ based Energy Minimization*, welche das in [ZHC02] beschriebene Verfahren implementiert, genutzt wird. Das Verfahren behält die Perioden der Taskaufrufe bei und bestimmt eine valide Verteilung mit dem geringsten Gesamte-

⁷ Integer Linear Programming (z.dt. Ganzzahlige Lineare Programmierung)

nergieverbrauch der aus dem lastabhängigen Verbrauch der Kerne ermittelt wird. Um dies zu erzielen, nutzt das Verfahren die sog. Slacks (Leerlaufzeiten) zwischen zwei aufeinanderfolgenden *Tasks* aus. Dabei werden die Kerne in langsamere, dafür stromsparendere Modi geschaltet, wodurch sich zwar die Laufzeit der *Tasks* verlängert, die Deadlines aber aufgrund der Ausnutzung von Slacks beibehalten werden.

Um das Problem des Optimalitätsnachweises zu umgehen, wird als Abbruchbedingung ein sog. *GAP* von 10% eingestellt. Dieser bewirkt, dass erst die LP-Relaxation, d.h. eine Lösung des Gleichungssystems ohne die geforderte Ganzzahligkeit, ermittelt wird. Anschließend berechnet das Tool zu jeder Ganzzahligen Lösung die Abweichung zu der LP-Relaxation, und akzeptiert die erste Lösung, die maximal 10% von dieser abweicht. Da die Ganzzahlige Lösung eines GLP Problems niemals besser als seine LP-Relaxation sein kann, wird so gewährleistet, dass die ermittelte Lösung maximal um den eingestellten Prozentsatz von einer bestmöglichen Lösung abweicht.

Als Datensatz für die Verteilung wird das durch LGP partitionierte HVAC Modell verwendet, da dieses durch die kleinstmögliche Laufzeit die größten Slacks, und damit das höchste Potential zur Energieeinsparung, besitzt. Durch das Mapping-Verfahren werden die Aufgaben unter Verwendung einer Heuristik auf die Kerne verteilt, und die Kerne unter Einhaltung der Abhängigkeiten und Deadlines „verlangsamt“. Das Ergebnis des Mappings, d.h. die Verteilung von *Tasks* zu Kernen und die ermittelten Spannungsmodi für jede Allokation, werden für die weitere Verarbeitung im *Mapping Modell* gespeichert.

4.6 Code Generierung und Tracing

Die Code-Generierung erfolgt im Falle des HVAC-Beispiels über zwei Generatoren, die jeweils zur Code-Erzeugung für die Komponenten bzw. das Betriebssystem eingesetzt werden. Bei der Komponenten Generierung über Simulink und Yakindu Statecharts werden die Softwarekomponenten (SWCs) erstellt. Eine SWC kann beispielsweise die Ansteuerung für einen Lüfter oder Drucksensor implementieren. Auf Basis des Komponentenmodells erzeugt der Systemgenerator die Quell- und Headerdateien für das Verhalten vom System. Ferner werden in diesem Schritt die Datenstruktur des Systems sowie die Schnittstellen (APIs) für den Zugriff auf diese erstellt. Der erstellte systemspezifische Code ist dadurch unabhängig vom Betriebssystem auf dem er ausgeführt werden soll.

Um Daten aus dem System (d.h. der zugrundeliegenden Datenstruktur) abzurufen oder Daten in das System zu schreiben, werden für jeden Port der entsprechenden Komponente die zugehörigen Schnittstellenfunktionen definiert. Die jeweilige Funktionalität der Komponente wird in einer, vom Generator erzeugten, Funktion implementiert.

Beim Generieren des OSEK⁸ Betriebssystems (im Falle des HVAC Systems zwei Instanzen von Erika Enterprise⁹) wird als Basis eine OIL-Datei (OSEK Implementation Language) generiert. Diese beinhaltet die Konfigurationsdaten des OSEK-Betriebssystems und ist in formaler Metasprache nach Backus-Naur-Form [Knu64] beschrieben. Enthalten sind Definitionen und Deklarationen, wie beispielsweise die Verteilung der Task oder welche Prioritäten, Ressourcen sowie Betriebssystemdienste benötigt werden. Die aus der OIL-Datei generierten C-Dateien, die Dateien des OSEK-OS sowie weitere Quelldateien, werden zusammengeführt und kompiliert. Anschließend werden Treiber und Libraries durch den Linker hinzugefügt und so die endgültige Anwendung in Binärform für die gewählte Zielplattform erstellt, deren Ablauf auf der Zielplattform mit gängigen Software Tracing Tools nachverfolgt werden kann [Ama14-ol].

5 Resümee und Ausblick

Die Entwicklung moderner automobiler eingebetteter Multicore-Systeme ist ein komplexer Prozess, der eine Vielzahl von Arbeitsschritten und Werkzeugen umfasst. Insbesondere die Integration der Werkzeuge für einen durchgängigen Arbeitsfluss und die Gewährleistung des korrekten Datenaustausches zwischen den beteiligten Werkzeugen stellen dabei eine wesentliche Herausforderung dar. Dieser Beitrag befasste sich mit dieser Problemstellung und präsentierte mit der AMALTHEA Tool Plattform eine Möglichkeit den Entwicklungsprozess zu unterstützen. Hierfür wurde die Entwicklung eines industriellen Anwendungsfalls anhand einer mit der AMALTHEA Tool Plattform erstellten Werkzeugkette erfolgreich durchgeführt, die sich sowohl aus Drittwerkzeugen als auch den in AMALTHEA integrierten Werkzeugen zusammensetzt. Dadurch konnte gleichermaßen die Teilautomatisierung des Datenaustausches, wie auch die Eignung der in die Plattform integrierten Werkzeuge überprüft werden. Andererseits verdeutlichte der Anwendungsfall, dass insbesondere die Aktivitäten mit starkem Bezug zur Hardware nicht durchgängig automatisiert werden konnten, da beispielsweise Code-Generatoren manuell an die jeweilige Zielplattform angepasst werden mussten. In Zukunft soll die Plattform im Rahmen des AMALTHEA4public Projektes um weitere Aspekte der automobilen Softwareentwicklung, wie Verifikation, Tests und Safety-Aspekte, erweitert werden.

Literatur

- [Ama14-ol] AMALTHEA CONSORTIUM: Deliverable D 3.4 - Prototypical Implementation of Selected Concepts. Unter: <https://itea3.org/project/workpackage/document/download/1702/09013-AMALTHEA-WP-3-D34-Prototypicalimplementationofselectedconcepts.pdf>

⁸ Spezifikation für Echtzeitbetriebssysteme automobiler eingebetteter Systeme (<http://www.osek-vdx.org/>)

⁹ OSEK zertifiziertes Open-Source Echtzeitbetriebssystem (<http://erika.tuxfamily.org/drupal/>)

- [BJ14] BRINK, C. UND JATZKOWSKI, J.: EU-Forschungsprojekt AMALTHEA: Entwicklung einer open-source Entwicklungsplattform, erweiterbar durch die Community. SafeTRANS News, (1): S. 12-13, 2014.
- [BKP+14] BRINK, C.; KAMSTIES, E.; PETERS, M. UND SACHWEH, S.: On Hardware Variability and the Relation to Software Variability, 40th Euromicro Conference on Software Engineering and Advanced Applications, Verona, Italy, 2014
- [CSR+01] CORMEN, T. H.; STEIN, C.; RIVEST, R. L. UND LEISERSON, C. E.: Introduction to Algorithms, McGraw-Hill Higher Education, 2nd Edition, 2001
- [HIK13] HOETTGER, R.; IGEL, B. UND KAMSTIES, E.: Vector Clock Tracing and Model-Based Partitioning for Distributed Embedded Systems, International Journal of Computing, 4th Edition, Volume 12, 2013
- [HI15] HOETTGER, R.; KRAWCZYK, L. UND IGEL, B.: Model-Based Automotive Partitioning and Mapping for Embedded Multicore Systems, XIII International Conference on Parallel, Distributed Systems and Software Engineering, Volume 9 No. 1, 2015
- [KA96] KWOK, Y. UND AHMAD, I.: Dynamic Critical-Path Scheduling: An Effective Technique for Allocating Task Graphs to Multiprocessors, IEEE Transactions on Parallel Distributed Systems, Piscataway, NJ, USA, May 1996
- [KCJ+90] KANG, K. C.; COHEN, S.; HESS, J.; NOVAK, W. UND PETERSON, A.: Feature-Oriented Domain Analysis (FODA), Feasibility Study, Software Engineering Institute, Tech. Rep. CMU/SEI-90-TR-21, 1990.
- [KK13] KRAWCZYK, L. UND KAMSTIES, E.: Hardware models for automated partitioning and mapping in multi-core systems, 7th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS), Band 2, Seiten 721-725, Berlin, 2013
- [Knu64] KNUTH, D. E.: Backus Normal Form vs. Backus Naur Form, Communications of the ACM, vol. 7, Nr. 12, S. 735-737, 1964
- [ZD14] ZHU, Q. UND DENG, P.: Design synthesis and optimization for automotive embedded systems. International symposium on physical design (ISPD '14). ACM, New York, NY, USA, 141-148, 2014
- [ZHC02] ZHANG, Y.; HU, X.; CHEN, D.Z.: Task scheduling and voltage selection for energy minimization, 39th Design Automation Conference (DAC), New Orleans, LA, 2002

Autoren

M.Sc. Lukas Krawczyk ist Promotionsstudent im Rahmen einer kooperativen Promotion an der Universität Bielefeld und der Fachhochschule Dortmund. Er hat sein Masterstudium in der Informatik mit Vertiefung in technischer Informatik im Juli 2014 abgeschlossen und widmet sich der Forschung im Bereich Scheduling mit Fokus auf eingebettete Multi- und Manycore Systeme für Echtzeitanwendungen.

M.Eng. Robert Höttger hat sein Masterstudium in der Informations- und Elektrotechnik an der Fachhochschule Dortmund Anfang 2014 abgeschlossen und nahtlos sein Promotionsstudium am Informatik Lehrstuhl 12 der Technischen Universität Dortmund begonnen. Zu seinen wissenschaftlichen Tätigkeiten gehören die Erforschung adaptiver Systemverhalten mit Hilfe von Traces sowie Entwicklungen zu softwaretechnischen Parallelisierungsverfahren eingebetteter Echtzeitsysteme im Automobilbereich.

Dipl.-Inform. (FH) Christopher Brink Jahrgang 1984, ist Doktorand im Fachgebiet Softwaretechnik von Prof. Dr. Wilhelm Schäfer am Institut für Informatik der Universität Paderborn. Er hat im August 2008 sein Studium der Informatik als Dipl.-Inform. (FH) abgeschlossen, erlangte 2009 seine Promotionsberechtigung und forscht seitdem im Bereich der Hardware-/Software-Produktlinien.

M.Sc. Daniel Fruhner hat sein Masterstudium in der Informatik an der Fachhochschule Dortmund Anfang 2013 abgeschlossen und arbeitet seit dem beim AMALTHEA Projektpartner BHTC.

Prof. Dr. Carsten Wolff ist seit 2007 Professor für Technische Informatik an der Fachhochschule Dortmund. Er hat an der Universität Paderborn Elektrotechnik und Wirtschaftswissenschaften studiert und am Heinz Nixdorf Institut bei Prof. Hartmann und Prof. Rückert auf dem Gebiet der Digitalelektronik promoviert. In seiner siebenjährigen Industrietätigkeit war Dr. Wolff in der Halbleiterindustrie (Infineon AG) tätig und primär mit der Entwicklung von ASICs und Prozessoren sowie der entsprechenden Entwicklungsmethodik befasst. In diesem Kontext hat er sowohl als Projektleiter als auch als Abteilungsleiter gearbeitet und u.a. am Aufbau eines neuen Entwicklungszentrums mitgewirkt. Ferner ist Carsten Wolff Gesellschafter und Mitgründer der smart mechatronics GmbH, eines spin-offs des Heinz-Nixdorf-Instituts und der FH Dortmund mit dem Schwerpunkt Mechatronik Entwicklung.

Generierung von AUTOSAR-Modellen aus UML-Spezifikationen

Dr. Jan Meyer

HELLA KGaA Hueck & Co.

Beckumerstr. 130, 59556 Lippstadt

Tel. +49 (0) 2941 / 38 80 13, Fax. +49 (0) 2941 / 38 80 13

E-Mail: jan.meyer@hella.com

Jörg Holtmann, Thorsten Koch, Dr. Matthias Meyer

Projektgruppe Entwurfstechnik Mechatronik

Fraunhofer Institut für Produktionstechnologie IPT

Zukunftsmeile 1, 33102 Paderborn

E-Mail: {joerg.holtmann|thorsten.koch|matthias.meyer}@ipt.fraunhofer.de

Zusammenfassung

Die Komplexität moderner Fahrzeuge steigt aufgrund der zunehmenden Anzahl von Funktionen, die durch elektronische Systeme umgesetzt werden. Insbesondere nehmen die Abhängigkeiten zwischen den an der Entwicklung beteiligten Fachdisziplinen und der Softwareanteil massiv zu. Wir haben einen für die Automobilindustrie angepassten, zum Reifegradmodell Automotive SPICE konformen Prozess für die Entwicklung von Steuergeräten konzipiert, der ein fachdisziplinübergreifendes Systems Engineering und einen systematischen Übergang in die Softwareentwicklung unterstützt. Im Kontext dieses Entwicklungsprozess beschreiben wir in diesem Beitrag den Übergang vom UML-basierten Softwareentwurf zum in der Automobilindustrie etablierten AUTOSAR-Standard mit Hilfe einer automatischen Modelltransformation. So werden fehleranfällige und zeitaufwändige manuelle Tätigkeiten reduziert. Wir haben die Generierung von AUTOSAR-Modellen gemeinsam mit dem international tätigen Automobilzulieferer Hella KGaA Hueck & Co. in seriennahen Entwicklungsprojekten praktisch erprobt und Zeit- und Kostenersparnisse festgestellt.

Schlüsselworte

AUTOSAR, UML, Automotive, Modelltransformation, modellbasierte Entwicklung

Generating AUTOSAR Models from UML Specifications

Abstract

The complexity of modern cars increases due to the increasing amount of functions that are realized by electronic systems: there are more and more dependencies between the different engineering disciplines involved, and the amount of software increases significantly. In order to cope with this, we conceived a development process for electronic control units (ECUs), which is tailored to the automotive industry and in conformance with the process maturity model Automotive SPICE. The process supports a discipline-spanning systems engineering and a systematic transition to software development. In the context of this development process, we describe in this paper the automated transition from UML-based software design to the established automotive standard AUTOSAR by means of a model transformation. This way, we reduce error-prone and time-consuming manual tasks. In cooperation with the internationally operating automotive supplier Hella KGaA Hueck & Co., we evaluated the generation of AUTOSAR models in development projects close to serial production: as a major result, we observed time and cost savings.

Keywords

AUTOSAR, UML, Automotive, model transformation, model-based development

1 Einleitung

Die Automobilindustrie ist eine der bedeutendsten Branchen in Deutschland – sowohl im Hinblick auf Umsatz- und Beschäftigtenzahlen, als auch auf das Hervorbringen von Innovationen. Ein wesentliches Forschungs- und Entwicklungsthema ist die Emissionsreduktion: Regulierungen zur CO₂-Reduzierung erfordern innovative Antriebskonzepte wie Hybrid- oder Elektroantriebe in Verbindung mit der Verwendung neuer Materialien (Leichtbau). Die Elektronisierung und Digitalisierung von Fahrzeugen wird insbesondere durch fortschrittliche Fahrerassistenzsysteme (Advanced Driver Assistance Systems, ADAS) weiter zunehmen und letztlich ein autonomes Fahren ermöglichen. Hinzu kommt die Vernetzung des Fahrzeugs mit seiner Umgebung und dem Internet („Connected Car“): Fahrzeuge sind in Zukunft an Kommunikationssysteme z. B. zur Unfallvermeidung und Verkehrssteuerung angebunden (Car-2-Infrastructure) und koordinieren sich gegenseitig (Car-2-Car).

In der Folge steigt sowohl die Komplexität des Gesamtsystems Fahrzeug als auch die Komplexität einzelner Assistenzsysteme aufgrund der vielen neuen Funktionen, die durch elektronische Steuergeräte mit hohem Softwareanteil umgesetzt werden, massiv an. Insbesondere nehmen die Abhängigkeiten zwischen den verschiedenen an der Entwicklung beteiligten Fachdisziplinen und die Bedeutung der Softwareentwicklung deutlich zu. Die Entwicklung solcher Systeme in hoher Qualität erfordert einen ganzheitlichen Entwicklungsprozess mit Methoden des fachdisziplinübergreifenden Systems Engineering, der auch das Zusammenspiel von Fahrzeugherstellern (OEMs) und Zulieferern koordiniert. Zudem muss sich der Softwareentwicklungsprozess nahtlos in diesen Gesamtprozess integrieren.

Mit Automotive SPICE [ASIG10] wurde eine Spezialisierung des Reifegradmodells SPICE (ISO 15504) zur Bewertung der Leistungsfähigkeit insbesondere von Softwareentwicklungsprozessen für die Automobilindustrie eingeführt. Automotive SPICE macht Rahmenvorgaben u. a. für zu durchlaufende Entwicklungsphasen (so genannte Engineering Phases, z. B. System Architectural Design, Software Design oder Software Construction) und deren Arbeitsergebnisse. Insgesamt werden Anforderungen an den Systementwurf und vorwiegend an die Softwareentwicklung beschrieben, die Entwicklungsprozesse erfüllen müssen; die konkrete Gestaltung der Prozesse bleibt offen.

Ein weiterer internationaler Standard, der in der Automobilindustrie berücksichtigt werden muss, ist AUTOSAR [AUT11]. AUTOSAR definiert eine Referenzarchitektur und eine Entwicklungsmethodik, um die Integration und Wiederverwendung von Elektronik- und Software-Komponenten verschiedener Hersteller zu erleichtern. Um dieses Ziel zu erreichen, werden in der AUTOSAR-Referenzarchitektur drei Architekturebenen und standardisierte Schnittstellen dazwischen definiert. Die höchste Architekturebene ist die Applikations-Software. Die Applikations-Software ist hardwareunabhängig und definiert die Software zur Umsetzung der eigentlichen Funktionalität. Die zwei-

te Ebene Run-Time Environment (RTE) verbindet die Applikations-Software mit der dritten Ebene, der Basis-Software, und ist vergleichbar mit einer Middleware. Die RTE abstrahiert von der Hardware und der Basis-Software, so dass Komponenten der Applikations-Software auf verschiedenen Hardwareplattformen ohne Anpassungen ausgeführt werden können. Die unterste Architekturebene ist die Basis-Software. Sie stellt verschiedene Dienste und Treiber für den Zugriff auf die Hardwarekomponenten bereit. AUTOSAR fokussiert vor allem die Struktur von Software und bietet keine Möglichkeiten zur Verhaltensmodellierung. Unter anderem deswegen ist AUTOSAR erst für die späte, implementierungsnahe Phase der Softwareentwicklung geeignet.

Die Fahrzeughersteller fordern heute von ihren Zulieferern sowohl die Einhaltung bestimmter Reifegrade gemäß Automotive SPICE als auch zu AUTOSAR konforme Steuergeräte. Um dem gerecht zu werden, haben wir gemeinsam mit dem Automobilzulieferer Hella KGaA Hueck & Co. einen durchgängigen, zu Automotive SPICE und AUTOSAR konformen Entwicklungsprozess für Steuergeräte konzipiert [FHH+12, Mey15]. In dem Prozess (siehe Bild 1) haben wir insbesondere die Phasen des Systementwurfs und der Softwareentwicklung konkret ausgestaltet.

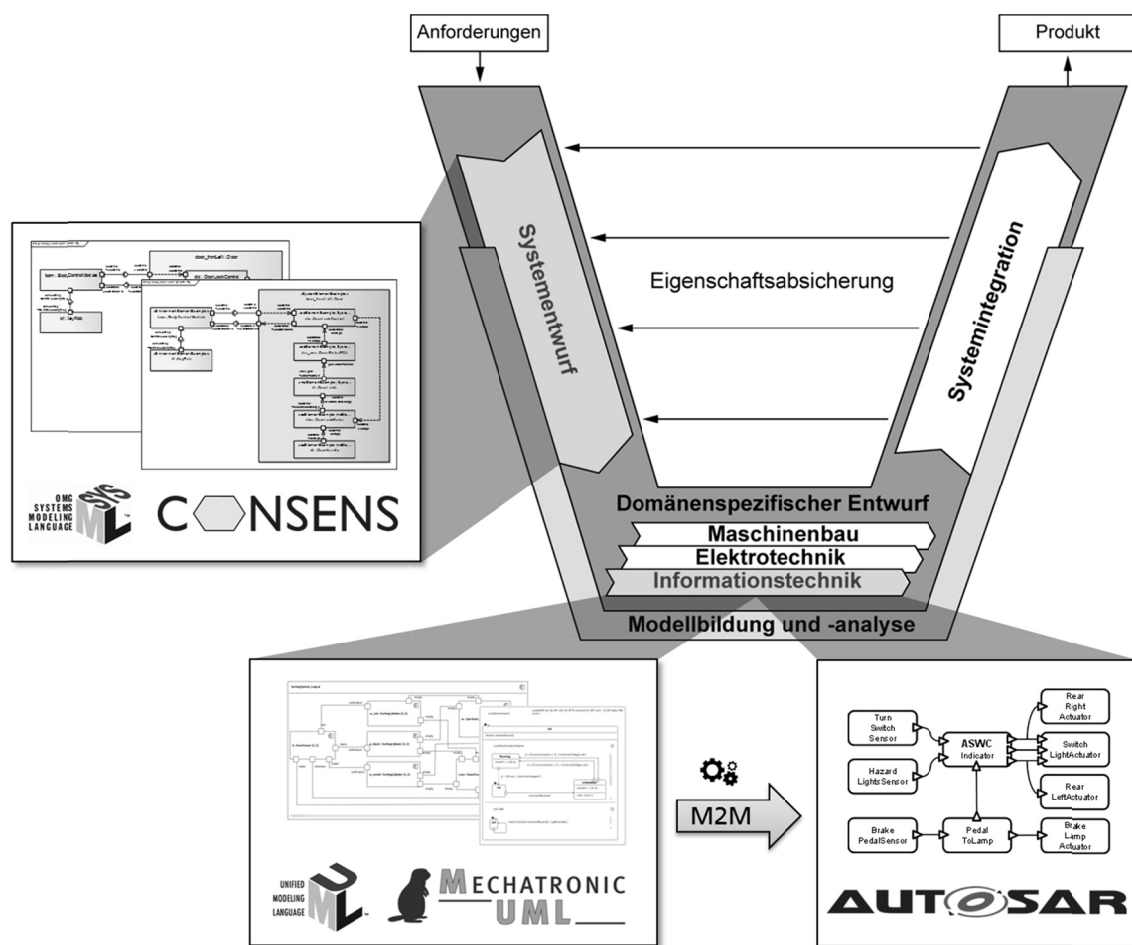


Bild 1: Durchgängiger, zu Automotive SPICE und AUTOSAR konformer Entwicklungsprozess für Steuergeräte in der Automobilindustrie

Wir haben für die Spezifika der Automobilindustrie angepasste Methoden und Modellierungssprachen für die Systementwicklung auf Basis der Systems Modeling Language (SysML) und der Systems-Engineering-Spezifikationstechnik CONSENS (CONceptual design Specification technique for the ENgineering of complex Systems) [GRS14] definiert, die eine fachdisziplinübergreifende Modellierung von Steuergeräten erlauben. Von dort erfolgt ein systematischer Übergang in die Softwareentwicklung, in der für den Softwareentwurf eine Kombination aus der Unified Modeling Language (UML) und der MechatronicUML [GRS14] zum Einsatz kommt. Damit können anders als mit AUTOSAR sowohl die Struktur als auch das Verhalten der Software präzise modelliert und früh mit Hilfe von Simulationsverfahren analysiert werden [NMK10].

In diesem Beitrag stellen wir den systematischen Übergang zu AUTOSAR-Modellen vor. Viele der für die Erstellung von AUTOSAR-Modellen benötigten Informationen liegen bereits in den UML-basierten Struktur- und Verhaltensmodellen vor und können somit aus diesen automatisch hergeleitet werden. Wir zeigen in diesem Beitrag, wie daraus AUTOSAR-Modelle mit Hilfe einer Modell-zu-Modell-Transformation (M2M, siehe Bild 1) generiert werden können. Auf diese Weise wird zum einen eine fehleranfällige und zeitaufwändige manuelle Übertragung der Informationen vermieden. Zum anderen werden die Zusammenhänge zwischen den Modellen gespeichert. So besteht die Grundlage für die Nachverfolgbarkeit (Traceability) und Konsistenzerhaltung der Informationen während der Entwicklung.

Im folgenden Kapitel beschreiben wir die Generierung der AUTOSAR-Modelle im Detail. Kapitel 3 berichtet über Evaluierungsergebnisse und Kapitel 4 stellt verwandte Arbeiten vor. Kapitel 5 fasst den Beitrag zusammen.

2 Generierung von AUTOSAR-Modellen

Für den Softwareentwurf (siehe Bild 2, linke Seite) verwenden wir zum einen Teile der UML und der MechatronicUML. Wir beschreiben die Softwarearchitektur mit strukturierten Klassen und das Verhalten mit Zustandsmaschinen und Aktivitätsdiagrammen. Zum anderen haben wir die UML über deren Profilmechanismus erweitert, um automotiv-spezifische Informationen wie die Betriebssystemeigenschaften und den Ressourcenverbrauch zu spezifizieren.

Im AUTOSAR-Standard benötigt eine Softwarekomponente Informationen für drei verschiedene Ebenen (siehe Bild 2, rechte Seite). Aus den in unserem Modellierungsansatz spezifizierten Informationen generieren wir mit einer Modelltransformation initiale AUTOSAR-Modelle für jede Ebene, die anschließend manuell verfeinert und ergänzt werden können. Dabei gehen die in unserem Modellierungsansatz spezifizierten Aspekte wie in Bild 2 skizziert in die verschiedenen Ebenen von AUTOSAR ein.

Die Bestandteile der Modelltransformation werden in den folgenden Abschnitten am Beispiel von Ausschnitten aus Softwaremodellen für die Zentralverriegelung und Innen-

lichtsteuerung eines Autos näher erläutert. Die Modelle sind ausschnittsweise in Bild 3 abgebildet und werden in den folgenden Abschnitten sukzessive erklärt.

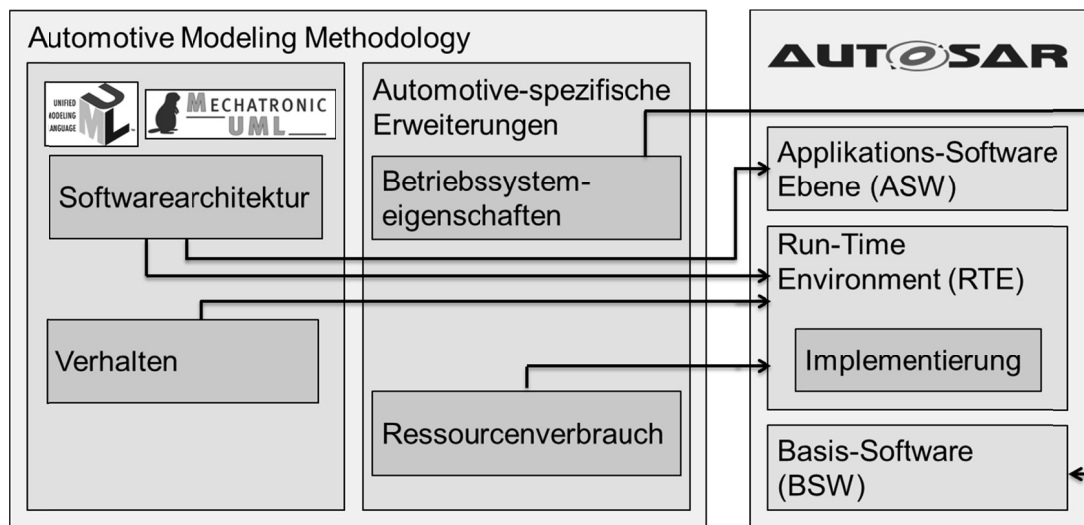


Bild 2: Übersicht der in den verschiedenen Modellen enthaltenen Informationen

2.1 Generierung der Applikations-Software-Ebene

In der Applikations-Software-Ebene (ASW-Ebene) (siehe Bild 2) werden Softwarekomponenten sowie ihre Ports und Schnittstellen spezifiziert. Da diese Informationen bereits in unserer UML-Spezifikation vorliegen, übertragen wir sie in die ASW-Ebene.

2.1.1 Generierung von Softwarekomponenten

In AUTOSAR gibt es verschiedene Arten von Softwarekomponenten, die von unserer Modelltransformation unterschieden werden.

Applikations-Softwarekomponenten (ASW-Komponenten) beinhalten die Logik der zu entwickelnden Software. In unserer UML-Spezifikation werden ASW-Komponenten in einem UML-Paket mit dem Stereotyp «application» definiert und besitzen ggf. zusätzlich den Stereotyp «singleton», sofern sie in AUTOSAR lediglich einmal instanziiert werden sollen. Im Beispiel in Bild 3 sind dies u. a. die Komponenten CentralLocking und InteriorLightManager im Paket BCM-Application-Software, welche die Logik für die Steuerung der Zentralverriegelung bzw. der Innenlichtsteuerung beinhalten. Diese werden in gleichnamige ASW-Komponententypen (Stereotyp «Application SW Component Type») in ein AUTOSAR-Paket (Stereotyp «ARPackage») Component-Library transformiert. Aufgrund der durch den Stereotyp «singleton» spezifizierten Information, dass diese beiden Softwarekomponenten nur einmal instanziiert werden sollen, legt die Modelltransformation direkt sog. Prototypen (Instanzen in AUTOSAR) an. Somit werden zusätzlich im AUTOSAR-Paket BCM-Application-Software die gleichnamigen Softwarekomponenteninstanzen (Stereotyp «SW Component Prototy-

pe)) CentralLocking und InteriorLightManager erstellt. Falls die Komponente mehr als einmal instanziiert werden soll, müssen diese Instanzen manuell spezifiziert werden, da diese Entscheidung nicht automatisch getroffen werden kann.

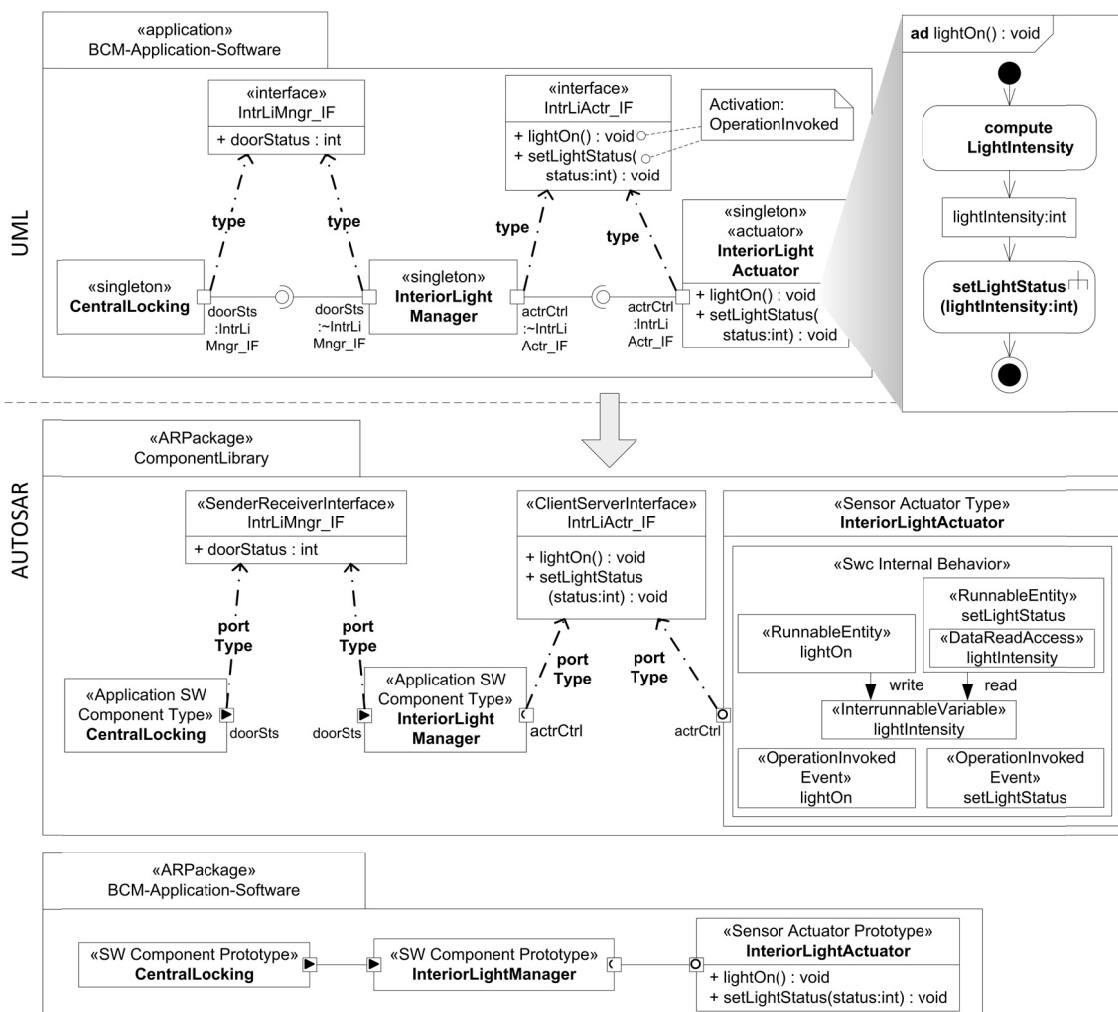


Bild 3: Beispiel-Modelle für die Transformation von UML in AUTOSAR

Sensor- und Aktorsoftwarekomponenten steuern eine Sensor- bzw. Aktorhardware und sind spezielle AUTOSAR-Softwarekomponenten. Diese Information hinterlegen wir im UML-Modell mit dem Stereotyp «sensor» bzw. «actuator», wie an der Komponente InteriorLightActuator in Bild 3 gezeigt. Die Modelltransformation legt analog zu der Transformation für ASW-Komponenten einen Typ und wegen des Stereotyps «singleton» eine Instanz für jede solche Komponente an, allerdings mit dem Stereotypen «Sensor Actuator Type» bzw. «Sensor Actuator Prototype».

Unser Modellierungsansatz und die Transformation umfassen mit Basis-Software-Komponenten und Service-Komponenten weitere Arten von Softwarekomponenten, die aber aus Platzgründen in diesem Beitrag nicht behandelt werden.

2.1.2 Generierung von Ports und Schnittstellen

Nach den Softwarekomponenten müssen deren in UML spezifizierte Ports und Schnittstellen nach AUTOSAR überführt werden.

In Bild 3 besitzt die Komponente `CentralLocking` einen Port `doorSts`, der über das Interface `IntrLiMngr_IF` typisiert ist (visualisiert durch die `type`-Verbindung). Das Interface besitzt das Attribut `doorStatus:int`, das den Verriegelungsstatus der Türen preisgibt. Der Port `doorSts` der Komponente `InteriorLightManager` ist ebenfalls über dieses Interface typisiert und konjugiert (erkennbar an `~` bei der Interfacetypisierung). Der nicht-konjugierte Port `doorSts` bietet das Interface und damit dessen Attribut an (sog. `provided interface`), während der konjugierte Port `doorSts` das Interface benötigt (sog. `required interface`). Über den Konnektor zwischen den beiden Ports kann die Komponente `InteriorLightManager` das Attribut `doorStatus` von `CentralLocking` lesen, um den Verriegelungsstatus der Türen in Erfahrung zu bringen. Wenn sich dieser Verriegelungsstatus ändert, kann der `InteriorLightManager` über die UML-Operation `lightOn` der Aktorkomponente `InteriorLightActuator` das Licht an- oder ausschalten.

Die Ports transformieren wir gemäß der Nutzung ihrer Interfaces (`provided` oder `required`) in entsprechende PPorts (z. B. Port von `CentralLocking`) oder RPorts (z. B. Port `doorSts` von `InteriorLightManager`) in AUTOSAR. Die Interfaces transformieren wir in entsprechende Interfaces im Paket `ComponentLibrary`. Dabei unterscheiden wir zwischen sog. `SenderReceiver`- und `ClientServer`-Interfaces. Erstere werden aus UML-Interfaces generiert, die lediglich Attribute besitzen (im Beispiel `IntrLiMngr_IF`). Letztere werden aus UML-Interfaces generiert, die UML-Operationen besitzen (im Beispiel `IntrLiActr_IF`). In Bild 3 wird ein `SenderReceiver`-Interface durch Pfeile in den Ports und ein `ClientServer`-Interface durch Kreise und Halbkreise in den Ports visualisiert.

2.2 Generierung der Run-Time-Environment-Ebene

Nachdem in der ASW-Ebene die AUTOSAR-Komponenten spezifiziert und instanziiert sowie die Ports und Schnittstellen der AUTOSAR-Komponenten definiert wurden, wird auf Run-Time-Environment-Ebene (RTE-Ebene) das Kommunikationsverhalten der AUTOSAR-Komponenten jeweils in einem sogenannten `Internal-Behavior-Element` spezifiziert. Das `Internal-Behavior-Element` beschreibt verschiedene Aspekte, wie zum Beispiel von der RTE aufgerufene Funktionen (`Runnables`), gemeinsam genutzte Variablen (`Interrunnable-Variablen`) oder exklusive Bereiche (`Exclusive Area`). Zur Generierung dieser Informationen nutzen wir das in UML spezifizierte Verhalten (siehe Abbildung von Verhalten auf RTE in Bild 2). Im Folgenden beschränken wir uns auf die Beschreibung von `Runnables` und `Interrunnable-Variablen`.

2.2.1 Generierung von Runnables

Runnables sind zentrale Elemente eines Internal-Behavior-Elements einer AUTOSAR-Komponente. Sie beschreiben Funktionen, die durch ein RTE-Ereignis (Event) aktiviert werden können. Das Verhalten der Funktion wird nicht innerhalb des AUTOSAR-Modells definiert. Stattdessen wird die Implementierung der Funktion über eine Referenz in einen Komponententyp eingebunden.

Die Transformation in die RTE-Ebene erzeugt für jede öffentliche UML-Operation einer Komponente ein Runnable-Element. Darüber hinaus werden Informationen über die Aktivierung benötigt. Der AUTOSAR-Standard definiert zwölf RTE-Ereignisse, die ein Runnable aktivieren können. Die RTE-Ereignisse werden in allgemeine Ereignisse für alle Schnittstellen, für Sender/Receiver und für Client/Server unterschieden. Die Information über das aktivierende Ereignis einer UML-Operation ist in UML nicht vorhanden und wurde in unserem Ansatz mit Hilfe einer UML-Annotation ergänzt.

In unserem Beispiel generieren wir das Kommunikationsverhalten der Komponente `InteriorLightActuator`. Dazu wird zuerst ein Internal-Behavior-Element (Stereotyp «Swc Internal Behavior») innerhalb der AUTOSAR-Komponente `InteriorLightActuator` im Paket `ComponentLibrary` angelegt. Anschließend werden für die beiden öffentlichen UML-Operationen `lightOn` und `setLightStatus` Runnable-Elemente (Stereotyp «RunnableEntity») erzeugt. Im UML-Modell ist definiert, dass beide UML-Operationen über das RTE-Event `OperationInvoked` gestartet werden (UML-Annotation `Activation` in Bild 3). Aus diesem Grund werden im Internal-Behavior-Element der AUTOSAR-Komponente `InteriorLightActuator` zwei Elemente mit dem Stereotyp «`OperationInvokedEvent`» und dem Namen der jeweiligen UML-Operation angelegt.

Neben der Aktivierung eines Runnables, fordert der AUTOSAR Standard zusätzlich Informationen darüber, auf welche Daten ein Runnable-Element lesend oder schreibend zugreift. Diese Informationen sind in unserem UML-Modell bereits durch die Argumente der UML-Operation vorhanden und können entsprechend übertragen werden. Wir definieren, dass ein Runnable-Element lesend auf einen Eingabeparameter und schreibend auf einen Rückgabewert zugreift.

In unserem Beispiel besitzt lediglich die UML-Operation `setLightStatus` ein Argument. Bei diesem Argument handelt es sich um einen Eingabeparameter. Daher wird bei der Transformation in die RTE-Ebene im Runnable-Element der UML-Operation `setLightStatus` ein `DataReadAccess`-Element mit dem Namen des Parameters (`lightIntensity`) zur Spezifikation des Datenzugriffs angelegt.

2.2.2 Generierung von Interrunnable-Variablen

Interrunnable-Variablen sind eine weitere Funktion der RTE-Ebene und erlauben es, dass Runnables eines Internal-Behavior-Elements ihre Daten untereinander austauschen.

Der AUTOSAR-Standard unterscheidet aufgrund der Art der Datenübergabe explizite und implizite Interrunnable-Variablen. Bei expliziten Interrunnable-Variablen kann immer auf die aktuellen Daten zugegriffen werden. Die RTE-Ebene stellt dabei sicher, dass die Daten immer konsistent sind. Bei impliziten Interrunnable-Variablen werden die Daten in der RTE zwischengespeichert und diese Kopie den anderen Runnables zur Verfügung gestellt. Eine Änderung der Variablen wird von diesen Runnables nicht sofort erkannt, sondern erst nach einem erneuten Aufruf. Standardmäßig werden bei der Generierung der RTE-Ebene implizite Interrunnable-Variablen erzeugt.

Die Verwendung von Interrunnable-Variablen kann aus dem UML-Modell über die UML-Aktivitätsdiagramme abgeleitet werden. Wird innerhalb einer UML-Operation eine weitere öffentliche UML-Operation derselben Komponente aufgerufen, so wird für die gemeinsam genutzten Daten jeweils eine Interrunnable Variable erzeugt. Darüber hinaus muss im AUTOSAR-Modell definiert werden, welches Runnable lesend bzw. schreibend auf die Interrunnable Variable zugreifen kann. In unserem Modellierungsansatz kann dies aus dem Datenfluss zwischen den UML-Operationen abgeleitet werden.

Das UML-Aktivitätsdiagramm rechts oben in Bild 3 zeigt das Verhalten der UML-Operation `lightOn`. Innerhalb des Aktivitätsdiagramms wird ein Wert für die interne Variable `lightIntensity` berechnet (Aktion `computeLightIntensity`). Abschließend wird dieser Wert als Argument beim Operationsaufruf `setLightStatus` übergeben. Da die UML-Operation `setLightStatus` zu derselben Komponente wie `lightOn` gehört, wird bei der Transformation in die RTE-Ebene eine Interrunnable-Variable mit dem Namen `lightIntensity` erzeugt. Das Runnable `setLightStatus` greift lesend und das Runnable `lightOn` schreibend auf die Interrunnable-Variable zu.

2.2.3 Generierung der Implementierungsebene

Für die Implementierung des Verhaltens der Runnable-Elemente setzt der AUTOSAR-Standard auf eine modellbasierte Entwicklung mittels sogenannter Behavior-Modeling Tools (BMT) [AUT11]. Dies können z. B. UML-Werkzeuge sein, aus denen Code generiert wird. Wie in Bild 3 dargestellt, ist in unserem UML-Modell neben der Struktur auch Verhalten in Form von Aktivitäts- und Zustandsdiagrammen beschrieben. Aus diesen lässt sich C-Code generieren, der zum Teil manuell ergänzt werden muss. Hierbei ist von Vorteil, dass im Gegensatz zu einer komplett manuellen Entwicklung, der generierte Teil konsistent zum Modell ist. Der Code muss mit Hilfe einer Referenz in der Komponentenbeschreibung hinterlegt werden. Dies ist der sogenannte SWCImplementation-Bereich mit einem XFile-Element für jedes Runnable. Die Informationen über den Speicherort des Codes und die Zugehörigkeit zum Runnable können im UML-Modell spezifiziert werden und werden durch die Modelltransformation berücksichtigt.

2.3 Generierung der Basis-Software-Ebene

Der bisher beschriebene Teil der Modelltransformation bezieht sich auf die Applikations-Software. Eine weitere wichtige AUTOSAR-Architekturebene ist die Basis-Software-Ebene. In dieser wird der Zugriff auf die Hardware definiert, die einen großen Einfluss auf die Qualitätsanforderungen wie Geschwindigkeit und Auslastung hat. Die Basis-Software ist in verschiedene Bereiche (Stacks) und einzelnen Komponenten aufgeteilt. Die einzelnen Komponenten werden größtenteils über Generatoren erzeugt, die über Konfigurationen gesteuert werden. Einige der Konfigurationen ergeben sich aus der System- und Softwarearchitektur in SysML/UML, wie z. B. die Betriebssystemspezifikation bzw. der Ressourcenverbrauch (siehe Bild 2).

Die Betriebssystemspezifikation wird bereits im UML-Modell vorgenommen, um die Aufteilung frühzeitig zu spezifizieren und mittels Echtzeitsimulation bereits in frühen Entwicklungsphasen zu überprüfen [NMK10]. Um diese bereits vorhandenen Informationen ebenfalls in die AUTOSAR-Konfiguration einfließen zu lassen, haben wir eine Transformation aus dem UML-Modell in die Konfiguration der AUTOSAR-Betriebssystemspezifikation entwickelt.

Hierbei werden die Tasks, Interrupts und die Zuordnung der Runnables darauf transformiert, um die bereits überprüften Ergebnisse aus dem UML Modell zu übernehmen. Hierzu werden die UML-Elemente auf die entsprechenden Konfigurationsparameter des AUTOSAR Betriebssystems abgebildet. Beispielsweise wird eine Task in der automotiv-spezifischen UML-Erweiterung (siehe Bild 2) auf eine AUTOSAR-Task abgebildet; dabei werden auch die Taskprioritäten berücksichtigt. Dieser Aspekt ist aus Platzgründen nicht im Beispiel enthalten.

3 Evaluierung

Der in diesem Papier beschriebene Ansatz wurde gemeinsam mit dem international tätigen Automobilzulieferer Hella KGaA Hueck & Co. entwickelt und implementiert. Die Implementierung erfolgte im Softwarewerkzeug IBM Rational Rhapsody. Dieses Werkzeug wurde gewählt, weil damit sowohl SysML- als auch UML-Modelle erstellt werden können und mittels mitgelieferter Profile ebenfalls die Modellierung von AUTOSAR-Modellen möglich ist. Die zuvor beschriebene Modelltransformation wurde als eine Erweiterung des Werkzeugs realisiert.

Bei der Implementierung der Transformation wurde besonders darauf geachtet, dass möglichst einfach verschiedene AUTOSAR-Versionen unterstützt werden können. Dies ist für einen Zulieferer besonders wichtig, da die Automobilhersteller unterschiedliche AUTOSAR-Versionen einsetzen und das Steuergerät des Zulieferers am Markt möglichst vielen Herstellern angeboten werden soll. So kann bei der Erstellung des AUTOSAR-Modells ohne Änderungen am UML-Modell die jeweilige zu nutzende AUTO-

SAR-Version ausgewählt werden. Dies ist ein großer Vorteil zum bisherigen Vorgehen, wo für jede unterschiedliche AUTOSAR-Version ein eigenes Modell erstellt wurde.

Die Transformation ist so in seriennahen Entwicklungsprojekten evaluiert worden, dass die verschiedenen Bereiche der Steuergeräteentwicklung, von Fahrerassistenzsystemen bis zu Komfortsteuergeräten, berücksichtigt wurden. Die Projekte waren weltweit verteilt und es arbeiteten mehrere Entwicklungsteams mit bis zu 100 Entwicklern daran. Als Ergebnis des Einsatzes konnten wir feststellen, dass die vorgestellte Transformation zu einer Zeit- und damit Kostenersparnis bei den Softwarearchitekten führt.

Darüber hinaus hat sich herausgestellt, dass die Transformation nur in der Richtung vom UML-Modell zum AUTOSAR-Modell nicht ausreicht. Die Automobilhersteller sind in letzter Zeit verstärkt dazu übergegangen, die Kommunikationsmatrix des Fahrzeugs, welche die Daten und die Verbindung des Steuergerätes mit anderen Geräten definiert, in Form von AUTOSAR-Modellen auszutauschen. Zudem ist ein Trend zu erkennen, dass einzelne Softwarekomponenten vom Automobilhersteller zugeliefert werden. Dies geschieht ebenfalls in Form einer AUTOSAR-Datei. Daher ist es wichtig, auch eine Transformation von AUTOSAR ins UML-Modell zu ermöglichen.

4 Verwandte Arbeiten

Die Einbindung des AUTOSAR-Standards in die Entwicklung von Steuergeräten mittels Modelltransformationen wird in der Literatur auch in einigen anderen Arbeiten beschrieben. In diesem Abschnitt werden diese Ansätze kurz beschrieben und wesentliche Unterschiede zu unserem Ansatz benannt.

In [SWCD13] stellen die Autoren eine Studie vor, wie aus einem UML-Modell mit Hilfe einer Modelltransformation ein AUTOSAR-Modell entsteht. In ihrem Lösungsansatz beschränken sich die Autoren aber auf die Transformation von System Templates des AUTOSAR-Standards, d.h. es werden nur AUTOSAR-Instanzen erzeugt. Weitere wichtige Informationen, wie die Definition von Runnables oder die Generierung der Basis-Software werden nicht berücksichtigt.

In [GHN09] werden Transformationsregeln zwischen einem SysML-Modell und der AUTOSAR-Applikations-Software beschrieben. In einer ausführlichen Evaluierung dieses Ansatzes [FHH+12] hat sich gezeigt, dass viele der für die Transformation benötigten Informationen in der Systemarchitektur noch nicht zur Verfügung stehen. Aus diesem Grund haben wir in unserem Modellierungsansatz weitere Elemente eingeführt, um die entsprechenden Informationen modellieren zu können. Darüber hinaus haben wir auf Basis dieser Elemente weitere Transformationsregeln erstellt (vgl. Abschnitt 2).

5 Resümee und Ausblick

Die Komplexität moderner Fahrzeuge steigt aufgrund der zunehmenden Anzahl elektronisch umgesetzter Funktionen und deren Vernetzung untereinander. Insbesondere nehmen die Abhängigkeiten zwischen den verschiedenen an der Entwicklung beteiligten Fachdisziplinen und der Softwareanteil massiv zu. In diesem Beitrag haben wir einen Teil eines für die Automobilindustrie angepassten, zum Reifegradmodell Automotive SPICE konformen Prozesses für die Entwicklung von Steuergeräten beschrieben, der ein fachdisziplinübergreifendes Systems Engineering und einen systematischen Übergang in die Softwareentwicklung unterstützt.

Die Softwareentwicklung muss des Weiteren konform zum in der Automobilindustrie etablierten AUTOSAR-Standard erfolgen. AUTOSAR unterstützt zwar die Erstellung von Komponentenmodellen, verfügt aber nur über eingeschränkte Möglichkeiten der Verhaltensmodellierung. Der Standard ist für die späte, implementierungsnahe Softwareentwicklung geeignet. Daher verwenden wir für die früheren Phasen besser geeignete, angepasste Modelle auf Basis der weit verbreiteten UML.

Ein großer Teil der für die AUTOSAR-Modellierung benötigten Informationen ist bereits in den UML-Modellen vorhanden. Damit diese nicht aufwändig und fehleranfällig manuell übertragen werden müssen, haben wir eine Modelltransformation konzipiert und im Softwarewerkzeug IBM Rational Rhapsody implementiert, die aus einem UML-Modell automatisch ein initiales AUTOSAR-Modell generiert. Die Transformation geht deutlich über verwandte Arbeiten hinaus.

Wir haben den Ansatz und insbesondere die automatische Transformation beim international tätigen Automobilzulieferer Hella KGaA Hueck & Co. in seriennahen Entwicklungsprojekten mit bis zu 100 Entwicklern praktisch erprobt. Dabei konnte zum einen die Praxistauglichkeit des Ansatzes gezeigt werden. Zum anderen wurde eine Zeit- und damit Kostenersparnis durch die Reduzierung fehleranfälliger, manueller Tätigkeiten festgestellt. Insbesondere die Unterstützung verschiedener AUTOSAR-Versionen hat sich aus Sicht eines Zulieferers als wertvoll erwiesen, um verschiedene Automobilhersteller effizient bedienen zu können.

Literatur

- [AUT11] AUTOSAR GBR.: AUTOSAR Standard, Version 4.0: www.autosar.org, 2011
- [ASIG10] AUTOMOTIVE SPECIAL INTEREST GROUP (SIG): Automotive SPICE – Process Reference Model. Release v4.5. 2010
- [FHH+12] FOCKEL, M.; HEIDL, P.; HOLTSMANN, J.; HORN, W.; HÖFFLINGER, J.; HÖNNINGER, H.; MEYER, J.; MEYER, M.; SCHÄUFFELE, J.: Application and Evaluation in the Automotive Domain. In: Klaus Pohl, Harald Hönniger, Reinhold E. Achatz und Manfred Broy (Hg.): Model-Based Engineering of Embedded Systems. The SPES 2020 Methodology. Berlin/Heidelberg: Springer, S. 157-175. 2012

- [GHN09] GIESE, H.; HILDEBRANDT, S.; NEUMANN, S: Towards Integrating SysML and AUTOSAR Modeling via Bidirectional Model Synchronization. In: Tagungsband Dagstuhl-Workshop MBEEs: Modellbasierte Entwicklung eingebetteter Systeme, Seiten 155-164. Informatik-Bericht 2009-01 TU Braunschweig, Institut für Software Systems Engineering Technische Universität Braunschweig, 2009
- [GRS14] GAUSEMEIER, J.; RAMMIG, F.-J.; SCHÄFER, W.: Design Methodology for Intelligent Technical Systems – Develop Intelligent Technical Systems of the Future. Berlin/Heidelberg: Springer (Lecture Notes in Mechanical Engineering). 2014
- [Mey15] MEYER, J.: Eine durchgängige modellbasierte Entwicklungsmethodik für die automobilen Steuergeräteentwicklung unter Einbeziehung des AUTOSAR Standards. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, Paderborn, 2015
- [NMK10] NICKEL, U.; MEYER, J.; KRAMER, T.: Wie hoch ist die Performance? Automobil-Elektronik, Seite 36-38, 2010
- [SWCD13] SELIM, G.M.K.; WANG, S., CORDY, J.R.; DINGEL, J.: Model transformations for migrating legacy deployment models in the automotive industry, In Journal of Software and Systems Modeling. Springer Verlag, 2013

Autoren

Dr. Jan Meyer arbeitete von 2006 bis 2011 als wissenschaftlicher Mitarbeiter am Software Quality Lab (s-lab) der Universität Paderborn. Er war dort zusammen mit industriellen Kooperationspartnern an verschiedenen Forschungsprojekten im Bereich der modellbasierten und verteilten Softwareentwicklung, insbesondere im Automotive-Umfeld, beteiligt. Seit 2011 arbeitet er bei der Hella KGaA Hueck & Co. in der Abteilung Prozesse, Methoden und Tools. In dieser ist er verantwortlich für die modellbasierte Architekturentwicklung mit SysML, UML, AUTOSAR und für Echtzeitsimulationen.

Jörg Holtmann ist wissenschaftlicher Mitarbeiter in der Abteilung »Softwaretechnik« der Projektgruppe Entwurfstechnik Mechatronik des Fraunhofer-Instituts für Produktionstechnologie IPT. Dort leitet er verschiedene, industrienahere Forschungsprojekte. Seine Arbeitsschwerpunkte liegen im formalen Requirements Engineering für software-intensive, eingebettete Systeme sowie im modellbasierten Systems Engineering.

Thorsten Koch ist wissenschaftlicher Mitarbeiter in der Abteilung »Softwaretechnik« der Projektgruppe Entwurfstechnik Mechatronik des Fraunhofer-Instituts für Produktionstechnologie IPT. Seine Schwerpunkte liegen im Requirements Engineering für software-intensive, eingebettete Systeme sowie im Variantenmanagement.

Dr. Matthias Meyer war von 2002 bis 2007 wissenschaftlicher Mitarbeiter im Fachgebiet Softwaretechnik und von 2008 bis 2010 Geschäftsführer im Software Quality Lab (s-lab) an der Universität Paderborn. Seit 2011 leitet er die Abteilung »Softwaretechnik« der Projektgruppe Entwurfstechnik Mechatronik des Fraunhofer-Instituts für Produktionstechnologie IPT. Seine Arbeitsschwerpunkte liegen im Bereich des Softwareentwurfs, des Re-Engineerings bestehender Software und der modellbasierten Entwicklung und Qualitätssicherung eingebetteter Softwaresysteme

Selbstdiagnose und Selbstoptimierung technischer Systeme auf Basis datenbasierter Prozessmodelle

Dr. Stefan Windmann, Prof. Dr. Oliver Niggemann
Fraunhofer-Anwendungszentrum Industrial Automation (IOSB-INA)
Langenbruch 6, 32657 Lemgo
Tel. +49 (0) 5261 / 94 29 20, Fax. +49 (0) 5251 / 60 62 68
E-Mail: stefan.windmann@iosb-ina.fraunhofer.de

Zusammenfassung

Industrielle Automatisierungssysteme werden aufgrund zunehmender Vernetzung und steigender Rechenleistung immer komplexer. Dadurch wird es für das Anlagenpersonal immer komplizierter, Prozessabläufe zu überwachen und zu optimieren. Eine Lösung für diese Problematik stellen intelligente Assistenzsysteme dar, die die Selbstdiagnose und -optimierung technischer Systeme ermöglichen. In diesem Artikel werden verschiedene Assistenzfunktionen dargestellt, die in Form der Toolbox proKNOWS schon heute in industriellen Anwendungen eingesetzt werden.

Eine datengetriebene Prozessmodellierung ermöglicht den flexiblen Einsatz der betrachteten Assistenzsysteme in komplexen Automatisierungssystemen. In diesem Ansatz werden Modelle diskreter und kontinuierlicher Prozessanteile, die zur Diagnose und Optimierung verwendet werden, aus Prozessdaten gelernt. Bei großen historischen Datenmengen ermöglicht die MapReduce-Technologie ein effizientes Lernen der Modelle. Zur Erfassung der Daten kommen verschiedene Konnektoren für proprietäre Protokolle und Datenbanken zum Einsatz, die entweder direkt oder über OPC UA an die Analysesoftware angebunden werden können.

Die Prozessüberwachung und Fehlerdetektion basiert auf dem Vergleich des mittels der gelernten Prozessmodelle prädizierten Prozessverhaltens mit Beobachtungen des tatsächlichen Prozessverlaufs. Separationsansätze ermöglichen zudem die Zuordnung erkannter Fehler zu einzelnen Systemkomponenten und damit eine genauere Fehlerdiagnose. Darüber hinaus werden die gelernten Prozessmodelle zur Selbstoptimierung technischer Systeme eingesetzt – beispielsweise zur Optimierung industrieller Automatisierungssysteme in Hinblick auf einen energieeffizienten Betrieb. Die beschriebenen Assistenzfunktionen wurden bereits erfolgreich in Automatisierungssystemen der Lemgoer Smart Factory sowie in mehreren Industrieprojekten erprobt.

Schlüsselworte

Intelligente Technische Systeme, Assistenzsysteme, Prozessanalyse, Optimierung, Diagnose, Data Mining

Self-Diagnosis and Self-Optimization of Technical Systems based on Data-Driven Process Models

Abstract

Industrial automation systems are growing more and more complex due to increasing computing capacities and expanding networks. These developments lead to a significant increase of data volume. Hence, monitoring and optimization of industrial processes is rendered more and more complex. Intelligent assistance systems provide solutions for these challenges by allowing for self-diagnosis and self-optimization of technical systems. The proposed paper describes some assistance functions which are already employed in industrial applications.

Data-driven process modelling allows for flexible use of intelligent assistance systems in complex automation systems. In this approach, models for process monitoring and optimization are learnt from process data. The applied methods comprise both self-learning of finite state machines and automatic parametrization of continuous process models. MapReduce technology allows for an efficient learning of process models from historical data.

Different interfaces for proprietary protocols and data bases are used for data acquisition. Data connections are directly integrated in a central infrastructure or connected via OPC-UA.

Condition monitoring, automatic failure detection and self-diagnosis are accomplished by comparison of predicted process behavior with observations of the real process. Furthermore, separation approaches allow for the assignment of recognized failures to particular system components and thus for a more precise diagnosis. The learnt process models are further applied for self-optimization of technical systems, e.g. for optimization of industrial automation systems with respect to energy efficient operation. The described assistance functions have been successfully evaluated in the Lemgo Smart Factory and in several industrial projects.

Keywords

Intelligent technical systems / assistance systems / process analysis / optimization / diagnosis / data mining

Dieser Beitrag ist in der Online-Version
leider nicht verfügbar.

Vernetzte mechatronische Systeme

Wissensaufbereitung und -bereitstellung durch Ontologien im Lebenszyklus von Produktionssystemen

***Dipl.-Inf. Marcus Petersen,
Dipl.-Ing. Gerald Rehage,
Prof. Dr.-Ing. Jürgen Gausemeier,
M.Sc. Dipl.-Ing.(FH) Frank Bauer,
Heinz Nixdorf Institut, Universität Paderborn
Fürstenallee 11, 33102 Paderborn
Tel. +49 (0) 5251 / 60 64 68, Fax. +49 (0) 5251 / 60 62 68
E-Mail: marcus.petersen@hni.uni-paderborn.de***

Zusammenfassung

Der Lebenszyklus von Produktionssystemen besteht nach Wiendahl et. al aus den drei Phasen Planung und Realisierung, Betrieb und Redistribution [WHW02]. Jede Phase enthält vielfältige Aufgaben die maßgeblich vom Fach- und Erfahrungswissen der Mitarbeiter abhängen. In der ersten Phase werden unter anderem die Fertigungsprozesse und -ressourcen festgelegt. Hierfür bedarf es genauer Kenntnisse über die zu fertigenden Bauteile, benötigten Prozesse und geeigneten Ressourcen selbst sowie deren Abhängigkeiten untereinander. In der Betriebsphase des Produktionssystems muss auf Maschinenausfälle oder Eilaufträge mit Um- bzw. Neuplanungen von Ressourcen reagiert werden. Das benötigte Wissen im Lebenszyklus von Produktionssystemen verteilt sich in der Regel auf verschiedene Personen und eine Vielzahl unterschiedlicher Dokumente; teilweise ist es auch nur implizit vorhanden. Demzufolge liegt es verteilt und unstrukturiert vor und eignet sich nicht für eine bedarfsgerechte Bereitstellung. Darüber hinaus fehlt eine Semantik, die das vorhandene Wissen in Beziehung setzt und durch automatisierte Schlussfolgerungen ergänzt. Hierfür bieten wissensbasierte Systeme (WBS) auf Basis von Ontologien einen vielversprechenden Lösungsansatz. Diese bieten eine Semantik und Inferenz zur Wissensmodellierung und ermöglichen den effizienten Zugriff auf das benötigte Wissen. Dies verspricht eine große Zeitersparnis bei den Aufgaben im Lebenszyklus von Produktionssystemen, auch im Hinblick auf die immer kürzer werdenden Produktlebenszyklen. Der Beitrag beschreibt den Aufbau von drei Ontologien für das Wissensmanagement im Rahmen verschiedener Aufgaben. Ferner werden die Vorteile durch den Einsatz von Ontologien praxisorientiert anhand eines Produktionssystems für eine Taschenlampe dargestellt.

Schlüsselworte

Wissensbasierte Systeme, Ontologie, Produktionssystemkonzipierung, Arbeitsvorbereitung, Prozess- und Ressourcenmodellierung

Ontology-based knowledge processing and provision within the lifecycle of production systems

Abstract

According to Wiendahl et al., the lifecycle of production systems consists of the three phases planning and realisation, operation and redistribution [WHW02]. In every phase, many tasks significantly depend on the expertise and experiences of the involved engineers. The processes and resources, for instance, are determined within the first phase of the lifecycle of production systems. For this purpose, profound knowledge about the components, processes and resources as well as their connections and interdependencies is required. Within the operational phase, the resource allocation often has to be changed after a machine break down has occurred or an urgent order has received. The required knowledge for the lifecycle of production systems is distributed in general to various persons and a variety of documents. Sometimes the knowledge only exists implicit. As a result, it is unstructured and unsuitable for a situation-based provision. In addition, a semantic for interrelating the knowledge and enhancing it through automated conclusions lacks. For this purpose, knowledge-based systems (KBS) with ontologies offer a promising approach. KBS provide a semantic and an inference for knowledge modelling and allow an efficient access to the knowledge. Thus, production systems can be adapted and rescheduled faster, which provides a great time saving with regard to the decreasing product lifecycles. For this purpose, the schema and application of three ontologies as part of the knowledge management for specific tasks within the lifecycle of production systems is described in this paper. Furthermore, the advantages due to the practical use of the ontologies are shown based on the production system of a flashlight.

Keywords

Knowledge-Based Systems, Ontology, Conceptual Design of a Production System, Work Planning, Modelling of Processes and Resources

1 Ausgangssituation und Handlungsbedarf

Kürzere Produktlebenszyklen zwingen die produzierenden Unternehmen ihre Produkte immer schneller auf den Markt zu bringen. Daraus resultiert die Herausforderung, die Zeit bis zum Produktionsanlauf signifikant zu verkürzen. Dies erfordert eine schnellere Entwicklung des zugehörigen Produktionssystems. Gleichzeitig werden Innovationen zunehmend in kleineren evolutionären Schritten am Markt eingeführt. Der geringere Innovationsgrad bei einem Produktwechsel ermöglicht die Wiederverwendung zahlreicher Module im neuen Produkt und ebenso im benötigten Produktionssystem. Prozesse und Ressourcen müssen nicht grundlegend neu konzipiert werden [ASS06]. Dabei nimmt die Komplexität der Produktionssysteme aufgrund der steigenden Komplexität der Produkte zu; eine zunehmende Herausforderung für Unternehmen. Dem kann nur mit einer durchgängigen Wissensaufbereitung und -bereitstellung im Lebenszyklus von Produktionssystemen entgegengewirkt werden. Basierend auf der Darstellung der Lebenszyklen im folgenden Abschnitt werden die Bedeutung und die Herausforderungen für die Nutzung von Fach- und Erfahrungswissen beschrieben. Ferner werden Ontologien als geeigneter Lösungsansatz vorgestellt.

1.1 Lebenszyklus von Produktionssystemen

Der Lebenszyklus eines Produktionssystems besteht aus den drei Hauptphasen *Planung und Realisierung*, *Betrieb* und *Redistribution*. Die Phase *Planung und Realisierung* umfasst alle Entwicklungstätigkeiten von der Konzeption des Produktionssystems bis zum Anlaufen der Produktion. Danach beginnt die *Betriebsphase* des Produktionssystems, welche mit dem Rückbau (*Redistribution*) endet [WHW02]. Das 3-Zyklenmodell der Produktentstehung nach GAUSEMEIER UND PLASS konkretisiert die Phase *Planung und Realisierung* in weitere Schritte. In der *Produktionssystemkonzipierung* wird das ganzheitliche Produktionssystemkonzept festgelegt, welches in der *Arbeitsplanung* domänenspezifisch ausgearbeitet und anschließend zu einem Gesamtsystem *integriert* wird [GP14]. Die Wertschöpfung des Produktionssystems wird im Wesentlichen durch die Phasen der Produktionssystemkonzipierung bis einschließlich des Betriebs beeinflusst. Diese Phasen liegen im Fokus des Beitrags und sind in Bild 1 dargestellt. Die Entwicklung von Produkt- und Produktionssystem erfolgt nach dem 3-Zyklenmodell der Produktentstehung parallel [GP14], die gegenüberliegenden Phasen im Bild sollen dies verdeutlichen.

1.2 Wissen im Lebenszyklus von Produktionssystemen

Die Globalisierung der Unternehmen und deren Produktion haben Auswirkungen auf das für Planung und Betrieb von Produktionssystemen notwendige Wissen. Dieses liegt zunehmend verteilt über Unternehmens- und Ländergrenzen hinweg vor, in den Köpfen einzelner Personen als auch in einer Vielzahl unterschiedlicher Dokumente. Die von

Industrie 4.0 geforderte Vernetzung der Produktion erfordert eine Harmonisierung und einen stärkeren Austausch von Informationen und Wissen. Die Durchdringung der Produktion mit Informations- und Kommunikationstechnologie (IKT) führt zu einem Anstieg des verfügbaren Wissens über die Produktion. Dieses verteilte und unstrukturierte Wissen gilt es aufzubereiten, um es bedarfsgerecht im Lebenszyklus eines Produktionssystems bereitzustellen. Darüber hinaus bedarf es einer Semantik, die das vorhandene Wissen in Beziehung setzt und durch automatisierte Schlussfolgerungen ergänzt. Hierfür bedarf es adäquater Wissens-, Planungs- und Erklärungsmodelle [FA13]. Die Wissensmodelle enthalten die benötigten Informationen in einer formalen und ggf. auch semantischen Struktur. Die Planungsmodelle nutzen diese Wissensbasis für die (teilautomatisierte) Lösung von konkreten Planungs- und Aufgabenstellungen, wie z.B. die Synthese von Ressourcen für eine gegebene Prozesskette. Je höher der Automatisierungsgrad dieser Planungsmodelle ist, desto wichtiger werden Erklärungsmodelle, die den Anwender beim Nachvollziehen einer Lösung unterstützen.

1.3 Ontologien als Lösungsansatz für die Wissensmodellierung

Ein Lösungsansatz für Wissensmodelle zur Aufbereitung und bedarfsgerechten Bereitstellung von Wissen im Lebenszyklus von Produktionssystemen sind Ontologien. Die Planungsmodelle generieren situationsabhängige Anfragen an die Ontologie und bereiten das Ergebnis für den Anwender auf.

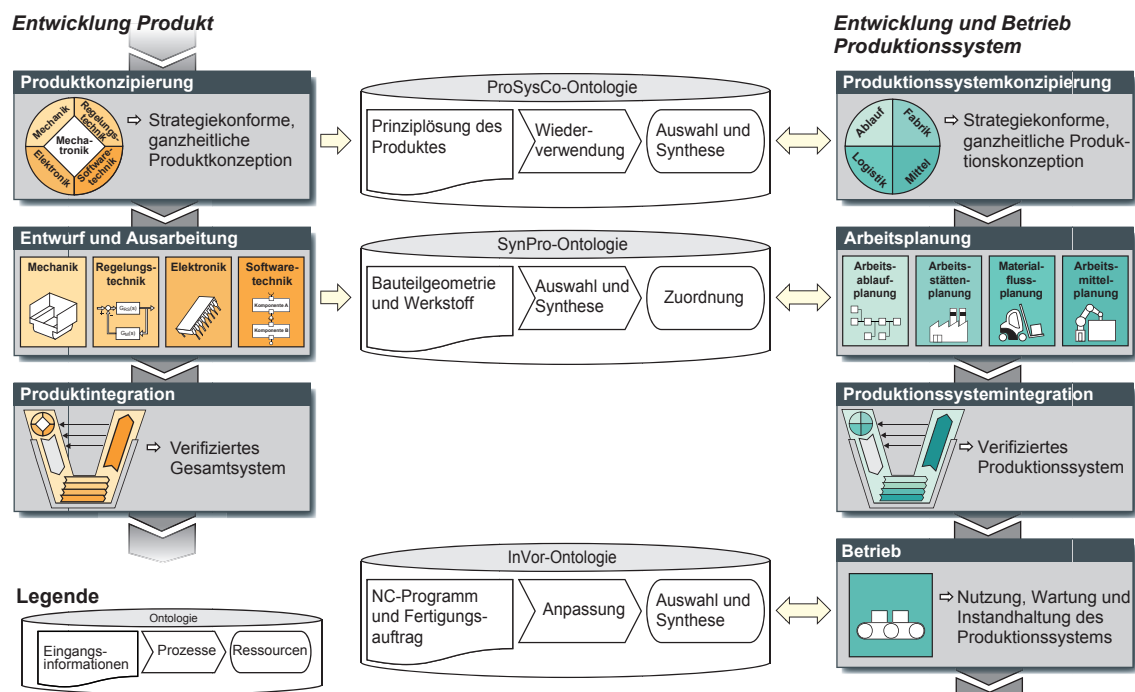


Bild 1: Einordnung der Ontologien des Beitrags in den Lebenszyklus von Produktionssystemen

Bild 1 zeigt die Phasen der Produktentwicklung und stellt diese den Phasen des Lebenszyklus von Produktionssystemen gegenüber. Die drei Ontologien des Beitrags sind in der Mitte des Bildes den Phasen zugeordnet. Die Ontologien ermöglichen eine gezielte Bereitstellung von Wissen und Unterstützen die Aufgaben während der Entwicklung und dem Betrieb des Produktionssystems. Die bedarfsgerechte Wissensbereitstellung erfolgt durch die Verarbeitung von Informationen aus der Produktentwicklung und den vorhandenen Zwischenergebnissen durch die jeweiligen Planungsmodelle. Die Pfeile in Bild 1 symbolisieren diesen uni- bzw. bidirektionalen Informationsaustausch.

Die Grundlagen der semantischen Wissensrepräsentation und deren Einsatz in der Produktion werden in Kapitel 2 erläutert. Kapitel 3 bildet den Kern dieses Beitrages und umfasst die Vorstellung der drei Ontologien sowie deren Einordnung in den Lebenszyklus von Produktionssystemen. In Kapitel 4 werden Anwendung und Nutzen der Ontologien am Beispiel einer Taschenlampe aufgezeigt.

2 Semantische Technologien für das Wissensmanagement

Das Wissensmanagement befasst sich mit dem Erwerb, dem Transfer, der Speicherung und der Nutzung von Wissen [VDI5610]. Wissensmanagementsysteme (WMS) sind nach LEHNER 2014 Softwaresysteme, die idealerweise die Identifikation, den Erwerb, die Entwicklung, Verteilung, Bewahrung und Bewertung von Wissen sowie das organisatorische Lernen und die organisatorische Effektivität unterstützen [Leh14]. Beispiele für WMS sind Dokumentenmanagementsysteme, Groupwaresysteme und die im World Wide Web verbreiteten Wikis. Diese dienen oftmals vorrangig dem Erwerb, der Speicherung und dem Transfer von Wissen und stellen Schnittstellen für eine Weiterverarbeitung des Wissens durch externe Softwarewerkzeuge zur Verfügung.

Wissensbasierte Systeme (WBS) als Untergruppe der WMS gehen einen Schritt weiter. Diese Systeme stellen das Fachwissen eines bestimmten Anwendungsbereichs dar und setzen es gezielt zur Problemlösung ein. Sie bestehen aus einer Wissensbasis und der Problemlösungskomponente. In der Wissensbasis wird das Fachwissen maschinenverständlich dargestellt. Anfragen an das System werden durch die Problemlösungskomponente verarbeitet und Schlüsse aus den vorliegenden Fakten gezogen. In konventionellen WBS liegt dieses Fachwissen als Datenstrukturen und das Lösungswissen als Algorithmen im Quellcode vor. Zur Anpassung oder Erweiterung des Fachwissens ist somit Programmieraufwand erforderlich [Kur89].

Einen Brückenschlag zwischen WMS und WBS verspricht die Weiterentwicklung der Inhalte im World Wide Web, hin zu einer auch für Maschinen verständlichen Darstellung. Bei dem als Semantik Web bezeichneten Konzept [BHL01] werden die Informationen von vornherein so aufbereitet, dass sie durch Maschinen verarbeitet werden können [HKR+08]. Die für die Umsetzung benötigten semantischen Technologien [PB06] umfassen u.a. Wissensrepräsentationssprachen zum Aufbau von Ontologien sowie Werkzeuge und Planungsmodelle für deren Erstellung und Anwendung. Abseits des Einsatz-

zes im World Wide Web werden Ontologien vermehrt für das industrielle Wissensmanagement eingesetzt [Bai08], [HKR+08].

2.1 Begriffsbestimmung Ontologie

In der Informatik wird darunter ein formales Repräsentationsmodell für Begriffe verstanden. Eine Ontologie stellt ein definiertes Vokabular zur Verfügung, um die Bedeutung von Objekten in einem Wissensbereich zu spezifizieren. Sie liefert die Grundlage für ein gemeinsames Verständnis über einen Bereich, für Schlussfolgerungen und für die Wiederverwendung von Wissen [GL02]. Die Bestandteile einer Ontologie sind Klassen (Konzepte), Instanzen von Klassen, Eigenschaften/ Restriktionen und Axiome. Mittels der formalen und expliziten Spezifikation wird die Konzeptualisierung in eine maschinenverständliche Sprache überführt [SS09], [PB06]. Nach der semantischen Ausdrucksstärke werden zwei Typen von Ontologien unterschieden: Leichtgewichtige (lightweight) Ontologien verwenden keinen oder nur sehr wenige Restriktionen zur Spezifikation der Objekte, z.B. für einen Glossar. Schwergewichtige (heavyweight) Ontologien beinhalten zusätzlich Axiome und Restriktionen, wodurch Aussagen über Objekte weiter eingrenzt und Schlussfolgerungen ermöglicht werden (Prädikatenlogik).

Eine weitere Unterteilung ergibt sich anhand der Spezialisierung, d.h. dem betrachteten Bereich und Detaillierungsgrad einer Ontologie. Sogenannte Top-Level-Ontologien definieren Konzepte, die in verschiedenen Wissensbereichen verwendet werden und so den Informationsaustausch zwischen den Bereichen unterstützen. Core-Ontologien umfassen Konzepte, die in mehreren (ähnlichen) Bereichen zum Einsatz kommen. Fachspezifische Ontologien, die einen bestimmten Bereich detailliert beschreiben, werden als Domain-Ontologien bezeichnet [Obe06], [Gua97].

2.2 Stand der Technik

Der modulare Aufbau von Ontologien bietet ein großes Potential zur Beschreibung der vielfältigen Wissensquellen im Lebenszyklus von Produktionssystemen. Zusätzlich ermöglicht die maschinenverständliche Wissensmodellierung verschiedene Schlussfolgerungen basierend auf dem enthaltenen Expertenwissen, bspw. die Komposition und Konfiguration von Prozessen und Ressourcen [MD09] oder die selbstständige Abwägung von Handlungsalternativen [SS12]. Dies ermöglicht die Automatisierung von zeitintensiven und wiederkehrenden Aufgaben der Mitarbeiter durch ein WBS.

In den letzten Jahren entstanden in diesem Bereich bereits einige prototypische Ontologien [SEG+10]. Das Einsatzfeld umfasst stets einzelne Aspekte wie das Supply-Chain-Management, z.B. durch die Suche geeigneter Sub-Unternehmer [Ame07], [NX13]; die semantische Modellierung von Produktionssystemen zur disziplinübergreifenden Zusammenarbeit [FKV14], [LSD+06]; die Entwicklungs- und Planungsunterstützung z.B. durch Wiederverwendung von Wissen [Loh06], [Har10] oder die Selbststeuerung von

Systemen auf Basis von Expertenwissen z.B. von logistischen Prozessen [Ber11]. Die Wissensmodellierung umfasst hierzu Fach- und Erfahrungswissen über die Produktionsressourcen (Maschinen, Werkzeuge etc.) und den darauf ausführbaren Prozessen (Produktions-, Logistik- und Hilfsprozesse) sowie den auf diese Weise herstellbaren Bauteilen bzw. Produkten. Ergänzt wird dieses Wissen durch die kausalen Zusammenhänge zwischen den Objekten; beispielsweise die maximal herstellbaren Bauteilmaße aufgrund des Bauraums einer Maschine. Die Ontologien gliedern sich daher typischerweise in die drei Top-Level-Konzepte *Produkt*, *Prozess* und *Ressource*. Diese werden durch weitere Unterkonzepte genauer beschrieben. Die Domain-Ontologien greifen häufig auf weitere Core-Ontologien zurück um die Konzepte durch standardisierte Geometrieangaben, Parameter, Schnittstellen und Maßeinheiten näher zu beschreiben [MD09].

3 Ontologien im Lebenszyklus von Produktionssystemen

Wie im Kapitel 1.1 dargestellt, entsteht die wesentliche Wertschöpfung eines Produktionssystems in den ersten beiden Hauptphasen des Lebenszyklus (*Planung und Realisierung, Betrieb*). Um die aufgezeigten Potentiale von Ontologien bei der Wissensaufbereitung und -bereitstellung im Lebenszyklus von Produktionssystemen erschließen zu können wurden drei Ontologien entwickelt. Jede Ontologie fokussiert dabei eine andere Aufgabe aus den Lebenszyklusphasen (siehe Bild 1).

Die Ontologie ProSysCo fokussiert die Produktionssystemkonzipierung und ermöglicht die Wiederverwendung und Parametrierung von Produktionsprozessen sowie die Synthese geeigneter Ressourcen für diese Prozesse. Im Rahmen der Arbeitsplanung können mit Hilfe von SynPro Herstellprozessketten für funktional gradierte Bauteile synthetisiert und die dafür notwendigen Ressourcen zugeordnet werden. Die dritte Ontologie unterstützt die Arbeitsvorbereitung und den Betrieb bei der Auswahl der wirtschaftlichsten, verfügbaren Ressourcen.

Im Folgenden werden Aufbau und Funktionsweise der Ontologien erläutert. Diese basieren alle auf ähnlichen Top-Level-Konzepten; außerdem nutzen sie identische Core-Ontologien, wie die QUDT-Ontologie [HK11-ol] zur Modellierung von Mengen und Einheiten. Auf diese Weise können die Ontologien ggf. später nach dem Bottom Up-Ansatz zusammengefügt werden, um komplexere Fragestellungen zu lösen.

3.1 Ontologie ProSysCo

Die Ontologie ProSysCo (Ontology for **Production System Concept**) dient als Grundlage für den Aufbau eines Planungswerkzeugs zur Produktionssystemkonzipierung. Das Werkzeug bietet dem Anwender Zugriff auf benötigtes Fach- und Lösungswissen während der grafischen Modellierung des Produktionssystems. Der Aufbau der Ontologie orientiert sich an den abzubildenden Aspekten und Elementen eines Produktionssystems. Diese wurden von REYES-PEREZ in einer Spezifikationstechnik für Produktions-

systeme festgelegt und von NORDSIEK und BRANDIS erweitert [Rey11], [Nor12], [Bra14]. Die Ontologie umfasst die Bereitstellung von Informationen über bekannte Materialelemente, Prozesse und Ressourcen sowie das Abfragen von Standardparametern oder zusätzlich erforderlichen Elementen des Produktionssystems. Bild 2 zeigt das Schema der Ontologie mit den Klassen und Relationen. Die Modellierung von spezifischen Materialelementen, Prozessen und Ressourcen erfolgt durch Instanzen der Klassen.

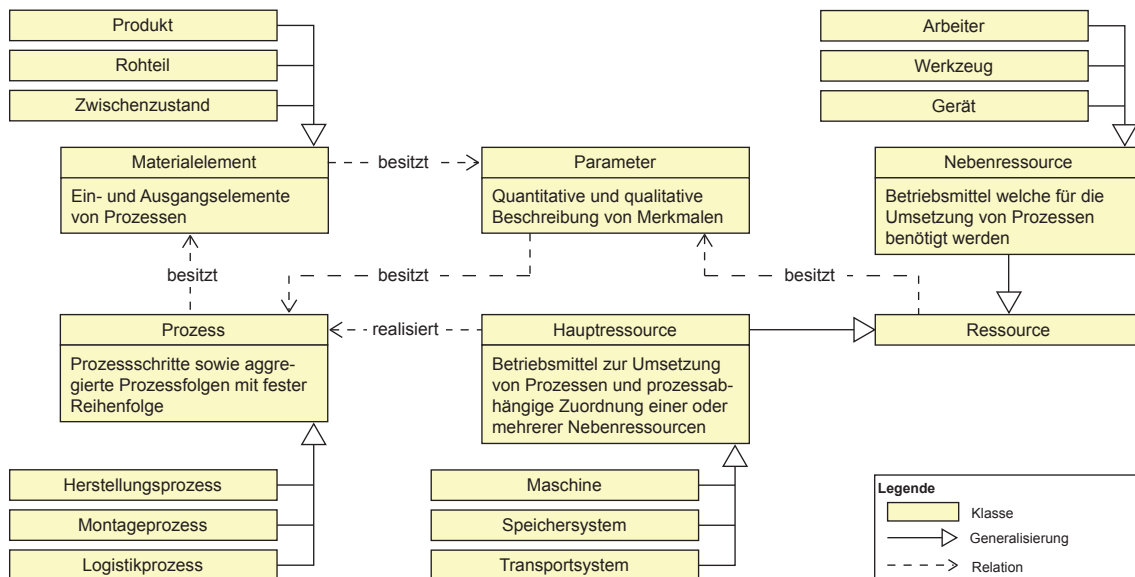


Bild 2: Ontologie ProSysCo zur Prozessauswahl und Ressourcensynthese

Die Unterstützung während der Modellierung auf Prozessebene umfasst zwei Arten von Suchanfragen:

1) Klassifizierung und Parametrierung von Materialelementen und Prozessen: Für eine gegebene Grundklasse (z.B. Prozess) werden mögliche Unterklassen (z.B. Montageprozess) zur Konkretisierung aus der Taxonomie ermittelt. Parallel werden auch vorhandene Instanzen einer Klasse abgerufen. Diese liefern spezifische Beschreibungen von Materialelementen oder Prozessen, bspw. für standardisierte Bauteile oder vorkonfigurierte Fertigungsprozesse. Die vorhandenen Instanzen können vom Nutzer ausgewählt und deren Parameter angepasst werden. Alternativ kann eine neue Instanz erstellt werden. Hierfür werden die Parameter aus der Klasse übernommen, welche vom Nutzer spezifiziert werden müssen.

2) Ermittlung von Prozessfolgen zur Herstellung eines Materialelements: Auf Basis der Instanzen und des gewünschten Ausgangsmaterialelements (z.B. Halbzeug) der zugehörigen Klasse sowie dessen Parameter und optional der Klasse eines bekannten Eingangsmaterialelements (z.B. Rohteil) werden geeignete Prozesse gesucht. Diese müssen den gleichen Klassen wie die Aus- und ggf. Eingangselemente zugeordnet sein und die gleichen Parameter aufweisen. Im Vergleich zu den Fertigungsprozessen sind die Konzepte zur Modellierung der Ressourcen deutlich umfangreicher. Diese umfassen Haupt- und Nebenressourcen, komplexe Ressourcen aus mehreren Hauptressourcen unter-

schiedlicher Klassen sowie Ressourcenkonfigurationen. Eine Ressourcenkonfiguration besteht aus genau einer Hauptressource sowie mindestens einer Nebenressource. Hierdurch kann eine Prozess-Instanz (z.B. Fräsen eines Standardbauteils) der erforderlichen Kombination, beispielsweise aus CNC-Fräsmaschine, der benötigten Werkzeuge und Vorrichtung, zugeordnet werden.

Die Unterstützung während der Modellierung auf Ressourcenebene umfasst drei unterschiedliche Suchanfragen:

1) Suche geeigneter Fertigungs- und Montagesysteme: Aus den spezifizierten Instanzen und den darin festgelegten Parametern, Merkmalen und Gestaltangaben ergibt sich für jeden Prozess eine Liste geeigneter Ressourcen.

2) Suche geeigneter Transportsysteme: Der Materialfluss ergibt sich aus den Fertigungsprozessen. Dieser wird durch Logistikprozesse und zu transportierende Materialelemente spezifiziert. Analog zu 1) erfolgt die Suche nach geeigneten Transportsystemen.

3) Ermittlung der Nebenressourcen: Zur Ergänzung einer gegebenen Hauptressource werden notwendige Werkzeuge, Hilfsmittel und Werker abgerufen. Die benötigten Nebenressourcen ergeben sich aus den in 1) gewählten Ressourcen.

Zusätzlich wird der Anwender bei der richtigen Modellierung der unterschiedlichen Produktionssystemaspekte unterstützt. Hierzu enthält die Ontologie eine Reihe von Modellierungsregeln, welche das Planungswerkzeug verarbeitet, um die formal richtige grafische Modellierung durch den Anwender sicherzustellen. Beispielsweise muss jeder Prozess über Ein- und Ausgangsmaterialelemente verfügen, Montageprozesse jedoch über mindestens zwei Eingangsmaterialelemente.

3.2 Ontologie SynPro

Die Planung resp. Synthese von Herstellprozessketten für ein Bauteil ist von den Abhängigkeiten und Wechselwirkungen zwischen Bauteil und Prozess bzw. zwischen den Prozessen selbst geprägt. Dies gilt insbesondere bei thermo-mechanisch gekoppelten Prozessen, bspw. zur Herstellung funktional gradierter Bauteile. Funktionale Gradierung beschreibt die gezielte Veränderung der Mikrostruktur eines Werkstoffs zur Einstellung von makroskopischen Bauteileigenschaften (Härte etc.). Das Ziel ist ein definierter Eigenschaftsverlauf über mindestens eine räumliche Dimension. An unterschiedlichen Stellen im Bauteil entstehen so scheinbar widersprüchliche Eigenschaften, die allerdings die spätere Funktion des Bauteils unterstützen. Funktional gradierte Bauteile stellen eine ressourcenschonende Alternative für moderne Verbundwerkstoffe dar [SMB09]. Jedoch werden für Synthese und Optimierung der erforderlichen Herstellprozessketten neue Methoden und Werkzeuge benötigt. Hierfür wird eine Planungsumgebung basierend auf der Ontologie SynPro (**S**ynthesis of **P**rocess chains) entwickelt.

Architektur und genereller Ablauf der Planungsumgebung sind in Bild 3 dargestellt. Die Planungsumgebung besteht aus drei Modulen und unterstützt den Anwender von der Bauteilbeschreibung bis zur Spezifikation der finalen Herstellprozesskette [BGH+13].

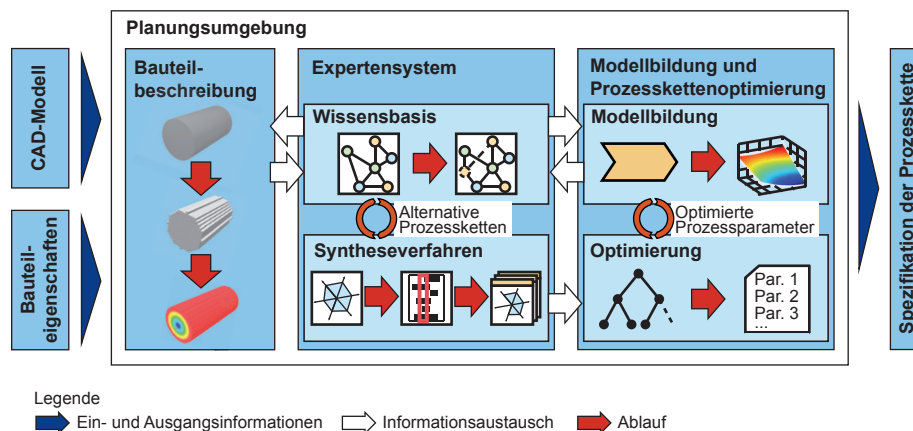


Bild 3: Planungsumgebung zur Synthese und Optimierung von Herstellprozessketten für funktional gradierte Bauteile

Das CAD-Modell des Bauteils bildet mit den Bauteileigenschaften die Eingangsinformationen für die Planungsumgebung. Im Rahmen der *Bauteilbeschreibung* wird das CAD-Modell in kleine quadratische Volumenelemente (Voxel) unterteilt, um die Grädierung im Bauteilinneren beschreiben zu können [BDG11]. Das *Expertensystem* besteht aus einer Wissensbasis und Syntheseverfahren. Die Wissensbasis enthält alle Informationen zu den Herstellprozessen, Ressourcen und Materialeigenschaften. Diese Informationen müssen strukturiert und zueinander in Beziehung gesetzt werden, um Schlussfolgerungen und eine automatisierte Verarbeitung zu ermöglichen. Ebenso müssen auch die vielfältigen Wechselwirkungen berücksichtigt werden [BGH+13]. Hierfür wird die Ontologie SynPro als Wissensbasis eingesetzt. Auf Grundlage der Wissensbasis schlägt das Syntheseverfahren mögliche Prozessketten zur Herstellung eines funktional gradierten Bauteils vor. Die Optimierung der Prozessparameter jeder alternativen Prozesskette erfolgt anschließend im Modul *Modellbildung und Prozesskettenoptimierung*. Dazu muss für jeden einzelnen Prozessschritt während der Modellbildung ein computerbasiertes Ersatzmodell erstellt worden sein. Diese Modelle werden dann im Hinblick auf die Bauteilbeschreibung hierarchisch entlang der Prozesskette optimiert [Wag13]. Anschließend wird die Prozesskette ausgewählt, die das Bauteil im Hinblick auf die Beschreibung am besten herstellen kann. Diese wird dem Planer mit Hilfe einer Spezifikationstechnik zusammen mit den optimierten Prozessparametern für weiterführende Planungen zur Verfügung gestellt [PBH+14].

Die Basis für die Synthese der Prozessketten bildet die Ontologie SynPro, welche in Bild 4 schematisch dargestellt ist. Das Schema zeigt die wichtigsten Klassen mit den entsprechenden semantischen Beziehungen. Einige Klassen besitzen für eine genauere Modellierung Unterklassen. So wird die Klasse *Prozess* bspw. in *Hilfsprozess* und *Fertigungsprozess* unterteilt, da letztere im Fokus der Synthese stehen. Die Relationen im

Bild zeigen lediglich die allgemeingültigen Beziehungen zwischen den Klassen. Abhängigkeiten oder Wechselwirkungen treten aber oft nur unter ganz bestimmten Voraussetzungen auf, daher werden diese über Regeln abgebildet. Besondere Bedeutung kommt dabei den (Material-)Eigenschaften zu, von denen das Ergebnis eines Prozesses am stärksten abhängt. Diese definieren ob zwei oder mehrere Prozesse kombiniert werden können und werden daher ebenfalls über Regeln abgebildet. Die Ontologie enthält Konzepte um alle Informationen aus der Bauteilbeschreibung ablegen zu können (CAD-Modell über Geometrie, Anforderungen als Eigenschaften etc.), da diese das Ergebnis der Herstellprozesskette definieren. Auf Basis dieser Informationen werden anschließend geeignete Herstellprozessketten synthetisiert. Dazu wird zuerst ein neues Materialelement für das geforderte Bauteil erstellt und in die Ontologie integriert. Im Anschluss wird die Ontologie inferiert, um Fehler erkennen und Schlüsse ziehen zu können. Danach werden alle Prozesse die das Bauteil herstellen können abgefragt und entsprechend ihrer Eignung zur Herstellung aller geforderten Materialeigenschaften in eine Rangfolge gebracht. Die vielversprechendsten Prozesse werden als Kernprozesse abgespeichert. Darauf aufbauend versucht die Planungsumgebung die vom jeweiligen Kernprozess noch nicht vollumfänglich erfüllten Anforderungen durch Abfrage und Verknüpfung neuer Prozesse zu minimieren. Dabei werden sowohl die Relationen als auch die Regeln überprüft, um die Konsistenz der Prozessketten gewährleisten zu können. Beinhaltet eine Prozesskette zwei inkompatible Prozessschritte, so wird diese in zwei Prozessketten mit jeweils nur einen der beiden Prozessschritte aufgespalten. Prozesse benötigen Prozessgrößen, um Materialelemente verarbeiten zu können. Diese werden von den Ressourcen bereitgestellt. Ressourcen bzw. deren Fähigkeiten werden durch Werkzeuge konkretisiert. So wird bspw. die endgültige Bauteilgeometrie einer Presse über deren Werkzeuge festgelegt. Daher finden weitere Abfragen zu den Ressourcen und Werkzeugen statt.

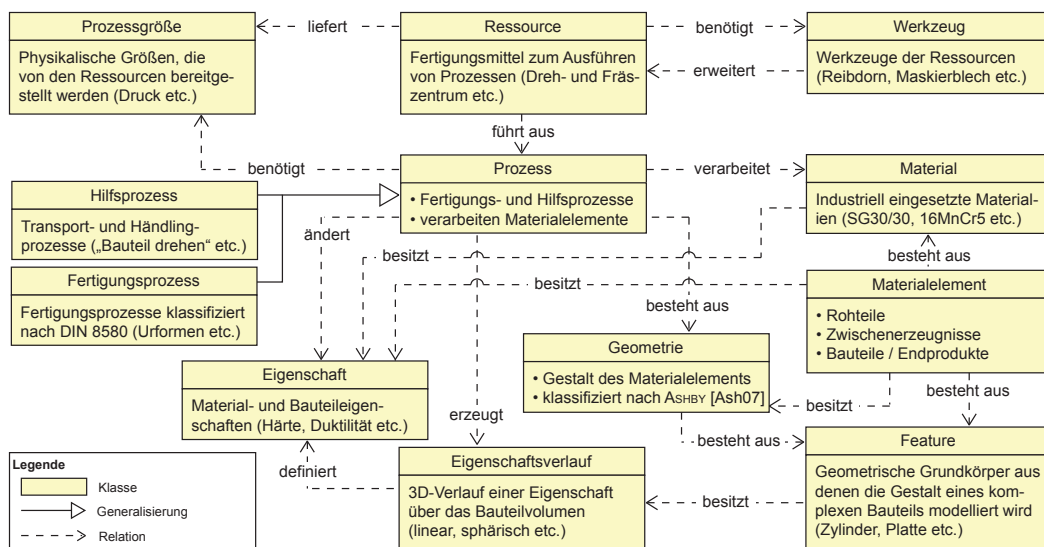


Bild 4: Schema der Ontologie SynPro zur Synthese von Herstellprozessketten

Die Planungsumgebung basiert auf den Arbeiten des Teilprojekts D5 aus dem Sonderforschungsbereich (SFB) Transregio 30. Ziel des Teilprojekts ist die Synthese und multikriterielle modellgestützte Optimierung von Herstellprozessketten für funktional gradierte Bauteile.

3.3 Ontologie InVor

Bei der Herstellung von Werkstücken mit geringen Stückzahlen wird die benötigte Flexibilität durch numerisch gesteuerte Werkzeugmaschinen erreicht [AR11]. Zur Bearbeitung eines breiten Spektrums von Werkstücken sind oft mehrere NC-Maschinen mit unterschiedlichen Eigenschaften in der Fertigung vorhanden. Bei der Auswahl einer geeigneten Maschine nutzen die Mitarbeiter ihr Fach- und Erfahrungswissen. Die Entscheidung erfolgt in der Regel nach technischen Gesichtspunkten für eine wahrscheinlich geeignete Maschine. Im Hinblick auf die Maschinenauslastung und -stundenkosten ist dies jedoch nicht unbedingt die wirtschaftlichste Wahl. Danach erfolgt die rechnerunterstützte Übersetzung der Werkstückgeometrie in einzelne Maschinenoperationen im Rahmen des CAM-Prozesses (Computer-Aided Manufacturing) [DA09]. Diese sind abhängig von der Maschinensteuerung, den -achsen und weiteren Eigenschaften. Das Ergebnis ist ein maschinenspezifisches NC-Programm, welches nur an der vorgesehenen Maschine korrekt ausgeführt wird [ZNN11].

In der Fertigung führen Maschinenstörungen oder Eilaufträge zu kurzfristigen Änderungen der Kapazitäten. Damit Verzögerungen in den folgenden Fertigungsschritten vermieden werden, gilt es die geplanten Aufträge möglichst schnell auf alternative Maschinen zu verteilen. Ein geeigneter Ansatz ist die Zuordnung mehrerer Maschinen für jeden Fertigungsschritt im Voraus. Hierfür wird ein ontologiebasiertes Entscheidungssystem zur automatisierten Selektion und Validierung alternativer NC-Fräsmaschinen im Rahmen einer **intelligenten Arbeitsvorbereitung** (Ontologie InVor) vorgestellt. Dies ermöglicht die Wahl der wirtschaftlichsten Maschine und möglicher Alternativen basierend auf den Anforderungen des erstellten NC-Programms. Ein schneller Maschinenwechsel bei veränderten Kapazitäten wird durch die spezifische Ausprägung des NC-Programms verhindert. Daher erfolgt eine Anpassung des NC-Programms an die Eigenschaften der neuen Maschine, wenn dies automatisiert möglich ist. Vor der Freigabe eines NC-Programms erfolgt zur Absicherung ein Testlauf auf einer virtuellen Werkzeugmaschine.

Kern des Entscheidungssystems ist eine Ontologie zur Suche alternativer Fräsmaschinen. Die Ontologie enthält formalisiertes Wissen der Arbeitsvorbereitung und Maschinenbediener, beispielsweise über die unterschiedlichen Fähigkeiten und Einsatzmöglichkeiten der Maschinen sowie die Zusammenhänge zwischen Anforderungen und Maschineneigenschaften. Eine schematische Darstellung der Ontologie mit den relevanten Konzepten und Relationen zeigt Bild 5. Die Konzepte entsprechen weitestgehend den Maschinenkomponenten; Relationen zwischen den Konzepten bilden die übergreifende

Kinematik und logische Zusammenhänge ab. Diese sind durch weitere Konzepte bzw. Relationen detailliert. In Bild 5 sind die Dreh- und Linearachsen als Unterklassen von Achsen dargestellt, die Generalisierungs-Beziehung bringt dies zum Ausdruck. Aus dem modellierten Wissen ergeben sich implizit die Bearbeitungsfähigkeiten der Maschine. Die Basisfähigkeiten werden mit Hilfe von Regeln explizit als Konfigurationen einer Maschine modelliert. Beispielweise kann der Arbeitsraum von einem nicht drehenden Werkstück besser ausgenutzt werden. Eine Regel definiert daher eine Konfiguration der Maschine mit effektiv größerem Arbeitsraum, wenn auf die Nutzung eines Drehtisches verzichtet wird. Weitere Regeln stellen die Beziehung zwischen Anforderungen des NC-Programms und den Maschinenparametern her, hier besteht selten eine direkte 1:1-Beziehung. Zum Beispiel ergibt sich die benötigte Arbeitsraumhöhe aus den Verfahrenswegen des NC-Programms und den Werkstückabmaßen. Die Anfrage an die Ontologie enthält die Anforderungen bzw. Parameter aus dem NC-Programm sowie die Werkstückbeschreibung. Die Antwort liefert als Ergebnis geeignete Maschinen, deren Konfiguration und Parameter. Die Anforderungen können auch aus einem CAD-Modell abgeleitet werden, um eine adäquate Maschine im CAM-Prozess auszuwählen.

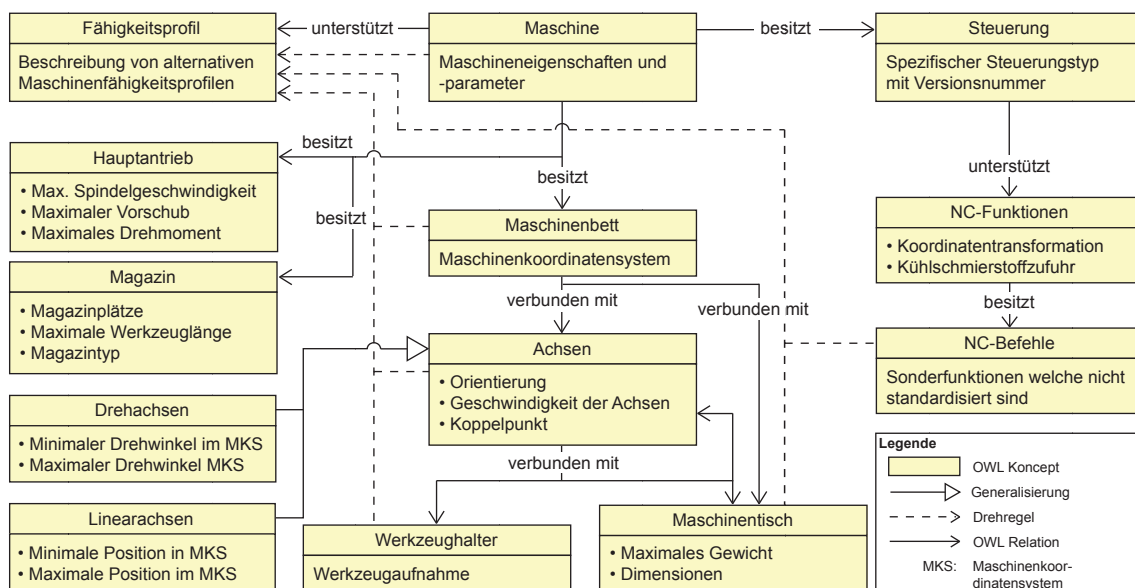


Bild 5: Schema der Ontologie InVor zur Auswahl alternativer Fräsmaschinen

Die Implementierung erfolgt in einem Forschungsprojekt des Spitzenclusters it's OWL mit DMG MORI. Ziel ist die Nutzung virtueller Werkzeugmaschinen zur Optimierung von spanenden Fertigungsprozessen [RGB+13]. Nach der Evaluierung des Systems für Fräsmaschinen wird die Betrachtung weiterer Werkzeugmaschinen angestrebt.

4 Anwendungsbeispiel »Produktion einer Taschenlampe«

Der Einsatz der drei vorgestellten Ontologien wird anhand der Konzipierung und Produktion einer Taschenlampe demonstriert. Zunächst wird der Demonstrator vorgestellt, anschließend folgen die Ergebnisse und Vorteile durch den Einsatz der Ontologien.

4.1 Demonstrator »Taschenlampe«

Für die Taschenlampe wurde eine Prinziplösung in Anlehnung an PAHL ET AL. [PBF+07] erstellt. Für eine detaillierte Beschreibung der Prinziplösung wird auf PETERSEN ET AL. [PBG+14] verwiesen. Die Gestalt der Taschenlampe ist in Bild 6 dargestellt. Es zeigt die drei Eigenfertigungsteile sowie die Zukaufteile (ZT).

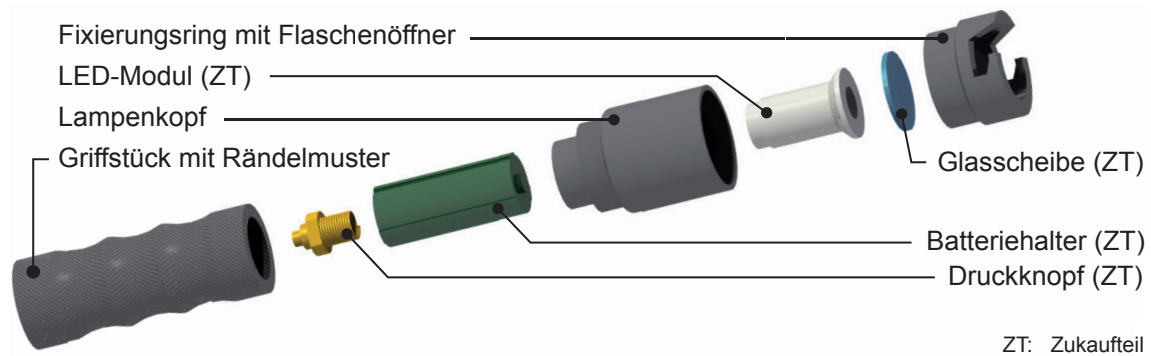


Bild 6: Gestaltmodell zum Demonstrator »Taschenlampe«

Im *Fixierungsring* ist ein Flaschenöffner integriert, ein neues Feature gegenüber einem früheren Modell. Des Weiteren besitzt das *Griffstück* ein Rändelmuster, damit die Taschenlampe gut in der Hand liegt.

4.2 Anwendung der Ontologien

Im Rahmen der integrativen **Produktionssystemkonzipierung** wird aus der Prinziplösung der Taschenlampe eine erste Prozesskette abgeleitet [GLL12]. Diese enthält alle bekannten Bauteile und deren Zusammenbau in einem mehrstufigen Montageprozess, ähnlich einem Montagevorranggraphen [BAD+86] und bildet den Ausgangspunkt für den Einsatz der Ontologie ProSysCo aus Kapitel 3.1.

Die erste Prozesskette gilt es in der Ansicht *Prozesse* im Planungswerkzeug zu modellieren. Hierfür steht eine Werkzeugleiste mit den entsprechenden Objekten zur Verfügung. Während der Modellierung wird das Modell mit Hilfe von Regeln auf die Einhaltung formaler Anforderungen geprüft. Im Folgenden werden die Fertigungsprozesse für die Eigenfertigungsteile der Taschenlampe und Montageprozesse konzipiert. Anschließend werden die benötigten Ressourcen synthetisiert. Bild 7 zeigt den Ausschnitt zur Herstellung des *Fixierungsring*s. Zur Bestimmung der Fertigungsprozesse wird der *Fixierungsring* zunächst der Klasse *Rotationssymmetrisches Bauteil mit subtraktiven Features* zugeordnet. Die erste Anfrage an die Ontologie liefert die Standardparameter wie *Material*, *Stückzahl* und *Geometrie* zur Beschreibung des Bauteils (①), welche durch den Anwender spezifiziert werden. Eine zweite Anfrage mit diesen Eingangsinformationen liefert den geeigneten Prozess *Fixierungsring fräsen* sowie das benötigte Eingangsmaterialelement *Aluminiumzylinder* (②). Die für diesen Prozess und das Materialelement bekannten Parameter werden vom Anwender auf den *Fixierungsring* angepasst

und dienen als Grundlage für eine dritte Anfrage nach einem Prozess der den benötigten *Aluminiumzylinder* herstellen kann (③).

Nachdem der Fertigungsprozess modelliert wurde, folgt im letzten Schritt die Auswahl und Synthese von geeigneten Ressourcen (④). Der Zugriff auf das Wissen über die bekannten Produktionssysteme, u.a. für die vorherige Taschenlampe, ermöglicht auch einem neuen Mitarbeiter die schnelle Konzipierung des benötigten Produktionssystems.

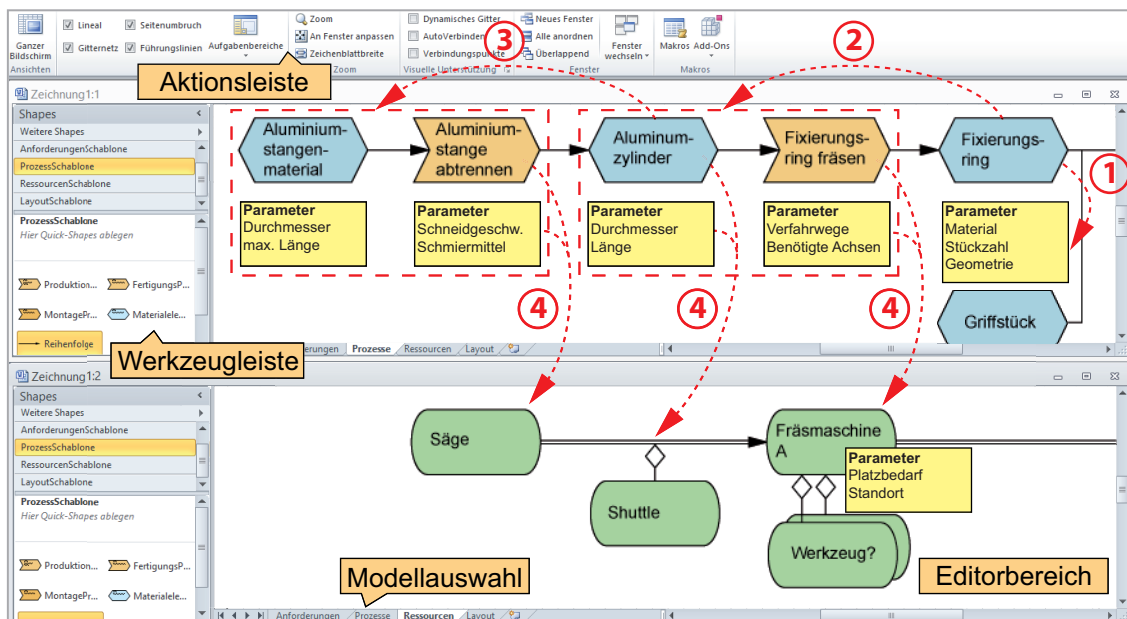


Bild 7: Planungswerkzeug basierend auf der Ontologie ProSysCo

Die Einbringung des Rändelmusters in das Griffstück ist aufgrund der verwendeten Hartlegierung mit hohem Werkzeugverschleiß und damit Kosten verbunden. Das Griffstück soll deshalb als funktional gradiertes Bauteil (vgl. Kapitel 3.2) mit einer weicheeren Oberflächenschicht ausgeführt werden, bei der die Härte im Inneren erhalten bleibt. Hierzu bietet sich eine Härtegradierung an, die mit Hilfe der Ontologie ProSysCo als Prozess *Griffstück gradieren* modelliert wurde. Dieser generische Gradierungsprozess wird im Rahmen der **Arbeitsplanung** mit der Planungsumgebung auf Basis der Ontologie SynPro ausgelegt. Dies umfasst die adäquate Beschreibung des Griffstücks mit der gewünschten Gradierung sowie die Synthese alternativer Prozessketten zur fertigungstechnischen Umsetzung. Die Planungsumgebung führt den Nutzer in sieben Hauptschritten durch den kompletten Planungsprozess.

Im ersten Schritt wird das CAD-Modell des Griffstücks geladen, bevor im zweiten Schritt die Werkstoffeigenschaften und die Anforderungen an die Gradierung charakterisiert werden. Anschließend wird das CAD-Modell gevoxelt, um die Beschreibung des Eigenschaftsverlaufs im nachfolgenden Schritt zu ermöglichen (siehe Bild 8). Für das Griffstück wurde eine Reduzierung der Härte von 90 HV (Vickershärte) an der Innenseite bis 65 HV an der Außenseite definiert. Auf Basis dieser Beschreibung erfolgt im fünften Schritt die Synthese alternativer Prozessketten. Hierzu werden die Informatio-

nen in die Ontologie SynPro integriert und mit einem Inferenzmechanismus alle Prozesse identifiziert die das Griffstück herstellen können. Die Eignung der jeweiligen Prozesse wird für den Anwender über Spinnennetz-Diagramme dargestellt. Hierbei wird der Erfüllungsgrad jeder Anforderung durch eine Achse repräsentiert. Ein geeigneter Prozess für das Griffstück ist *induktives Aufheizen*. Hierdurch wird jedoch die Härte im Bauteilinneren reduziert. Diese Restriktionen sind in der Ontologie hinterlegt und identifizieren die *Materialtemperatur* als prozessbestimmende Materialeigenschaft. Mit Hilfe weiterer Abfragen an die Ontologie werden zusätzliche Prozesse zur gezielten Beeinflussung der Materialtemperatur ermittelt, um die Härte im Bauteilinneren zu erhalten. Die einzelnen Parameter der Prozessketten werden nachfolgend in Schritt 6 optimiert. Auf Basis dieser Optimierung wird die beste Prozesskette zur Herstellung des Bauteils ausgewählt. Hierzu wird der von der Prozesskette herstellbare Eigenschaftsverlauf der geforderten Gradierung des Griffstücks gegenübergestellt. Im letzten Schritt wird die finale Prozesskette mit allen Anforderungen und Parametern angezeigt. Durch die Planungsumgebung können Herstellprozesse ohne umfassende Kenntnisse über funktionale Gradierungen geplant werden.

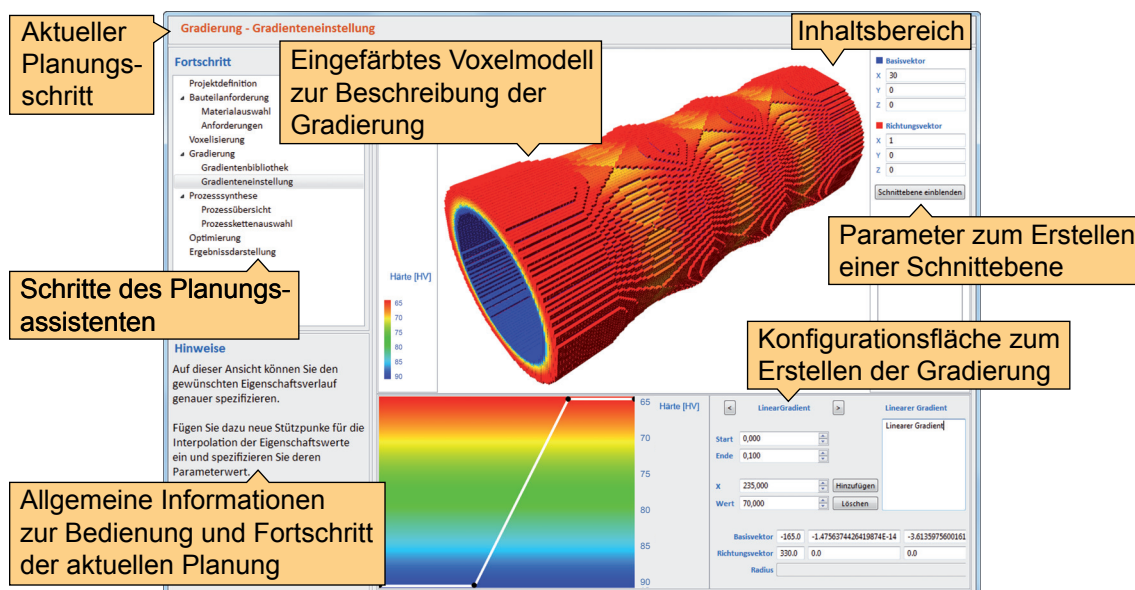


Bild 8: Anwendung der Planungsumgebung zur Beschreibung der Gradierung

Vor dem Hintergrund permanenter Kapazitätsänderungen in der **Betriebsphase** der Fertigung, sind die bei der Erstellung des NC-Programms festgelegten Werkzeugmaschinen auf ihre Wirtschaftlichkeit hin zu prüfen. Deshalb unterstützt die Ontologie InVor die Arbeitsvorbereitung bei der Suche nach geeigneten Maschinen für ein vorhandenes NC-Programm. Im Planungswerkzeug basierend auf der Ontologie ProSysCo wurden zur Fertigung der Taschenlampe drei Fräs-Dreh-Prozesse und Werkzeugmaschinen festgelegt. In der Benutzungsoberfläche der Ontologie InVor werden für diese Bearbeitungsschritte geeignete alternative Maschinen gesucht (siehe Bild 9). Als Suchparameter dienen die Anforderungen, die sich aus dem NC-Programm ableiten. Die Ergebnisse werden dem Nutzer angezeigt und ermöglichen die Priorisierung der geeigneten Ma-

schinen, z.B. anhand der Auslastung und Bearbeitungskosten. Dies stellt sicher, dass stets die wirtschaftlichste Maschine zum Einsatz kommt; hier ist die Werkzeugmaschine MT_2 wirtschaftlicher als die zuvor gewählte Maschine MT_1. Die gewählte Priorisierung ermöglicht zudem eine schnelle Umverteilung von Aufträgen bei einem Maschinenausfall.

The screenshot shows the 'InVorMa Web Interface' with a 'Machine selection' tab active. It displays 'Assisted machine selection for simulated manufacturing steps' with three steps listed. A callout box points to the step list with the text 'Auswahl der Fertigungsschritte'. Below this is a table comparing machines MT_1 and MT_2. A second callout box points to the 'User Priority' column with the text 'Priorisierung der ermittelten alternativen Maschinen durch den Nutzer'. To the right, a 'Manufacturing step details' panel shows process and setup times, and travel dimensions for Y, Z, and X axes. A 'Confirm' button is at the bottom right.

User Priority 0 = no relevance 1 = high; 9 = low	Machine Tool	Capacity forecast	Rating s = similarity e = efficiency	Time
2	MT_1 (pre-selected)	70%	s: 100% e: 100%	30 min
1	MT_2	60%	s: 90% e: 110%	23min
			s: 80% e: 90%	40 min

Manufacturing step details:
 Process time: 32 min
 Setup time: 20 min
 Travel on axis:
 Y: 45,5 mm A: 0 mm
 Z: 60,1 mm B: 0 mm
 X: 45,5 mm C: 0 mm

Bild 9: Benutzungsoberfläche zur Auswahl alternativer Werkzeugmaschinen

5 Resümee und Ausblick

Planung, Auslegung und Betrieb von Produktionssystemen erfordern viel Fach- und Erfahrungswissen. Für die Aufbereitung und situationsgerechte Bereitstellung dieses Wissens wurden drei Ontologien mit unterschiedlichen Schwerpunkten und die entsprechenden Werkzeuge für den Anwender vorgestellt. Anschließend wurde deren Anwendung und Nutzen anhand der Konzipierung und Produktion einer Taschenlampe demonstriert. Im Vergleich zu anderen Wissensbasierten Systemen sind Ontologien sehr flexibel und einfach zu erweitern.

In Zukunft sollen die drei Ontologien gebündelt werden. So kann das enthaltene Wissen bestmöglich für die Konzeption bzw. den Betrieb neuer Produktionssysteme genutzt werden. Aus der Gesamtheit des Wissens lassen sich dann neue Schlussfolgerungen ziehen. Beispielweise wird die Ressourcenauswahl bei der Konzipierung eines neuen Produktionssystems von Erfahrungen aus dem laufenden Betrieb direkt beeinflusst. Um das Erfahrungswissen bei der Konzeption neuer Produktionssysteme berücksichtigen zu können, muss eine Möglichkeit zum Abbilden des Wissens in abstrahierter Form geschaffen werden. Lösungsmuster mit Name, Kontext, Problem und Lösung wie ALEXANDER ET AL. [AIS+95] diese definieren bieten dafür eine gute Grundlage.

Literatur

- [AIS+95] ALEXANDER, C.; ISHIKAWA, S.; SILVERSTEIN, M.; JACOBSON, M.; FIKSDAHLKING, I.; ANGEL, S.; CZECH, H. (HRSG.): Eine Muster-Sprache – Städte, Gebäude, Konstruktion. Löcker Verlag, Wien, 1995
- [Ame07] AMERI, F.: Supply Chain Standardization – An ontological approach. VDM Verlag Dr. Müller, Saarbrücken, 2007
- [AR11] ABELE, E.; REINHART, G.: Zukunft der Produktion – Herausforderungen, Forschungsfelder, Chancen. Carl Hanser Verlag, München, 2011
- [Ash07] ASHBY, M. F.: Materials Selection in Mechanical Design – Das Original mit Übersetzungshilfen. Elsevier, München, 2007
- [ASS06] ALIZON, F.; SHOOTER, S.; SIMPSON, T.: Reuse of Manufacturing Knowledge to Facilitate Platform-Based Product Realization. Journal of Computing and Information Science in Engineering 6(2), 2006, S. 170-178
- [BAD+86] BULLINGER, H. J. (HRSG.): Ammer, D.; Dungs, K.; Seidel, U. A.; Weller, B.: Systematische Montageplanung – Handbuch für die Praxis. Carl Hanser Verlag, München, 1986
- [Bai08] BAIER, E.: Semantische Technologien in Wissensmanagementlösungen – Einsatzpotenziale für den Mittelstand. MFG-Stiftung Baden-Württemberg, Band 13, Stuttgart, 2008
- [BDG11] BAUER, F.; DETTMER, D.; GAUSEMEIER, J.: Feature-based component description for functional graded parts. Proceeding of the 4th Conference on Changeable, Agile, Reconfigurable and Virtual Production (CARV), Montreal, 2011, S. 239-244
- [Ber11] BERNDT, J. O.: Self-organizing Supply Networks: Autonomous Agent Coordination Based on Expectations. Proceedings of the 3rd International Conference on Agents and Artificial Intelligence (ICAART). INSTICC Press, 2011, S. 104-113
- [BGH+13] BIERMANN, D.; GAUSEMEIER, J.; HESS, S.; PETERSEN, M.; WAGNER, T.: Planning and Optimisation of Manufacturing Process Chains for Functionally Graded Components – Part 1: Methodological Foundations. Production Engineering – Research and Development 7(6), 2013, S. 657-664
- [BHL01] BERNERS-LEE, T.; HENDLER, J.; LASSILA, O.: The Semantic Web – A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. Scientific American – Mai 2001. Georg von Holtzbrinck Publishing Group, New York (NY), 2001, S. 34-43
- [Bra14] BRANDIS, R.: Systematik für die integrative Konzipierung der Montage auf Basis der Prinzipienlösung mechatronischer Systeme. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 325, Paderborn, 2014
- [DA09] DENKENA, B.; AMMERMANN, C.: CA-Technologien in der Fertigungs- und Prozessplanung. Carl Hanser Verlag, München, 2009
- [FA13] FORSCHUNGSUNION WIRTSCHAFT-WISSENSCHAFT; ACATECH – DEUTSCHE AKADEMIE DER TECHNIKWISSENSCHAFTEN (HRSG.): Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0, 2013
- [FKV14] FELDMANN, S.; KERNSCHMIDT, K.; VOGEL-HEUSER, B.: Combining a SysML-based modeling approach and semantic technologies for analyzing change influences in manufacturing plant models. Proceedings of the 47th CIRP Conference on Manufacturing Systems, Ontario, 2014
- [GL02] GRUNINGER, M.; LEE, J.: Ontology applications and design. Communications of the ACM – Februar 45(2), 2002, S. 39-41

- [GLL12] GAUSEMEIER, J.; LANZA, G.; LINDEMANN, U. (HRSG.): Produkte und Produktionssysteme integrative konzipieren – Modellbildung und Analyse in der frühen Phase der Produktentstehung. Carl Hanser Verlag, München, 2012
- [GP14] GAUSEMEIER, J.; PLASS, C.: Zukunftsorientierte Unternehmensgestaltung – Strategien, Geschäftsprozesse und IT-Systeme für die Produktion von morgen. Carl Hanser Verlag, München, 2014
- [Gua97] GUARINO, N.: Semantic Matching: Formal ontological distinctions for information organization, extraction, and integration. Information Extraction – A Multidisciplinary Approach to an Emerging Information Technology. Springer, Berlin, Band 1299, 1997, S. 139-170
- [Har10] HARMS, R.: Semantic-Web-Wissensbank für Planungsprozesse bei der Wiederverwendung von Produktionsanlagen. Fraunhofer-Verlag, Stuttgart, 2010
- [HK11-ol] HODGSON, R.; KELLER, P. J.: QUDT – quantities, units, dimensions and data types in OWL and XML. Unter: <http://www.qudt.org>, 28. November 2014
- [HKR+08] HITZLER, P.; KRÖTZSCH, M.; RUDOLPH, S.; SURE, Y.: Semantic Web – Grundlagen. Springer, Berlin, 2008
- [Kur89] KURBEL, K.: Entwicklung und Einsatz von Expertensystemen – Eine anwendungsorientierte Einführung in wissensbasierte Systeme. Springer, Berlin, 1989
- [Leh14] LEHNER, F.: Wissensmanagement – Grundlagen, Methoden und technische Unterstützung. Carl Hanser Verlag, München, 5. Auflage, 2014
- [Loh06] LOHSE, N.: Towards an ontology framework for the integrated design of modular assembly systems. PhD Thesis, University of Nottingham, Nottingham, 2006
- [LSD+06] LEMAIGNAN, S.; SIADAT, A.; DANTAN, J.-Y.; SEMENENKO, A.: Mason: A Proposal For An Ontology Of Manufacturing Domain. IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS), Prag, 2006, S. 195-200
- [MD09] MARTINEZ-LASTRA, J.; DELAMER, I.: Ontologies for Production Automation. Advances in Web Semantics I. Springer, Berlin, Band 4891, 2009, S. 276-289
- [Nor12] NORDSIEK, D.: Systematik zur Konzipierung von Produktionssystemen auf Basis der Prinzipienlösung mechatronischer Systeme. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 304, Paderborn, 2012
- [NX13] NING L.; XIAOPING L.: A multilevel modeling framework for semantic representation of cloud manufacturing resources. IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Whistler (BC), 2013,
- [Obe06] OBERLE, D.: Semantic management of middleware. Springer, Berlin, Band 1, 2006
- [PB06] PELLEGRINI, T.; BLUMAUER, A. (HRSG.): Semantic Web – Wege zur vernetzten Wissensgesellschaft. Springer, Berlin, 2006
- [PBF+07] PAHL, G.; BEITZ, W.; FELDHUSEN, J.; GROTE, K.-H.: Konstruktionslehre – Grundlagen erfolgreicher Produktentwicklung – Methoden und Anwendung. Springer, Berlin, 7. Auflage, 2007
- [PBG+14] PETERSEN, M.; BANDAK, S.; GAUSEMEIER, J.; IWANEK, P.; SCHNEIDER, M.: Methodik zur Berücksichtigung von Wechselwirkungen zwischen Produkt und Produktionssystem in den Frühen Phasen der Produktentwicklung – Ein Praxisbeispiel. 17. IFF-Wissenschaftstage, Magdeburg, 2014, S. 13-21
- [PBH+14] PETERSEN, M.; BAUER, F.; HESS, S.; GAUSEMEIER, J.; GRÄBLER, I.: Towards a Production System Specification Technique for Functionally Graded Components. Proceedings of the 13th International DESIGN Conference, Dubrovnik, 2014

- [Rey11] REYES-PEREZ, M.: A Specification Technique for the Conceptual design of manufacturing systems. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 291, Paderborn, 2011
- [RGB+13] REHAGE, G.; BAUER, F.; GAUSEMEIER, J.: Intelligent Manufacturing Operations Planning, Scheduling and Dispatching on the Basis of Virtual Machine Tools. Proceedings of the New Prolamat, Springer, Berlin, 2013
- [SEG+10] SHEA, K.; ERTELT, C.; GMEINER, T.; AMERI, F.: Design-to-fabrication automation for the cognitive machine shop. Advanced Engineering Informatics 24(3), 2010, S. 251-268
- [SMB09] STEINHOFF, K.; MAIER, H. J.; BIERMANN, D. (HRSG.): Functionally Graded Materials in Industrial Mass Production. Verlag Wissenschaftliche Scripten, Auerbach, 2009
- [SS09] STAAB, S.; STUDER, R.: Handbook on ontologies. Springer, Berlin, 2. Auflage, 2009
- [SS12] SCHUH, G.; STICH, V. (HRSG.): Produktionsplanung und -steuerung 2: Evolution der PPS. Springer, Berlin, 2012
- [VDI5610] VDI 5610: Wissensmanagement im Ingenieurwesen – Grundlagen, Konzepte, Vorgehen. Beuth Verlag GmbH, März 2009
- [Wag13] WAGNER, T.: Planning and Multi-Objective Optimization of Manufacturing Processes by Means of Empirical Surrogate Models. Vulkan Verlag, Essen, 2013
- [WHW02] WIENDAHL, H. P.; HEGENSCHIEDT, M.; WINKLER, H.: Anlaufrobuste Produktionssysteme. Werkstattstechnik (wt) online, Jahrg. 92, Ausgabe 11/12, 2002, S. 650-655
- [ZNN11] ZHANG, X.; NASSEHI, A.; NEWMAN, S.T.: Process comprehension for interoperable CNC manufacturing. IEEE International Conference on Computer Science and Automation Engineering (CSAE), Band 4, 2011, S. 225-229

Autoren

Dipl.-Inf. Marcus Petersen studierte Informatik an der Technischen Universität Clausthal. Seit 2011 ist er wissenschaftlicher Mitarbeiter in der Fachgruppe Produktentstehung von Prof. Gausemeier am Heinz Nixdorf Institut der Universität Paderborn. Seine Forschungsschwerpunkte liegen in der Planung von Herstellprozessketten für funktional gradierte Bauteile, der integrativen Produktionssystemplanung sowie dem Einsatz von wissensbasierten Systemen im Lebenszyklus von Produktionssystemen.

Dipl.-Ing. Gerald Rehage studierte Maschinenbau mit der Fachrichtung Produktentwicklung an der Universität Paderborn. Seit 2012 ist er wissenschaftlicher Mitarbeiter in der Fachgruppe Produktentstehung am Heinz Nixdorf Institut. Im Team „Integrative Produktionssystemplanung“ beschäftigt er sich mit Methoden zur Modellierung und Auswahl von Betriebsmitteln bei Planung und Betrieb von Produktionssystemen.

M.Sc. Dipl.-Ing. (FH) Frank Bauer war wissenschaftlicher Mitarbeiter in der Fachgruppe Produktentstehung von Prof. Gausemeier am Heinz Nixdorf Institut der Universität Paderborn. Im Rahmen mehrerer Forschungs- und Industrieprojekte hat Herr Bauer umfangreiche Kenntnisse im Bereich der Prozess- und Ressourcenmodellierung sowie dem Einsatz von semantischen Technologien in der Produktentstehung gesammelt. Seit Juni 2014 arbeitet Herr Bauer als Projektingenieur Automatisierungstechnik/Produktionssystemplanung bei der Infineon Technologies AG in Warstein.

Prof. Dr.-Ing. Jürgen Gausemeier ist Seniorprofessor am Heinz Nixdorf Institut der Universität Paderborn. Seine Arbeitsschwerpunkte sind Strategische Produktplanung und Systems Engineering. Er war Sprecher des Sonderforschungsbereiches 614 „Selbstoptimierende Systeme des Maschinenbaus“ und von 2009 bis 2015 Mitglied des Wissenschaftsrats. Jürgen Gausemeier ist Initiator und Aufsichtsratsvorsitzender des Beratungsunternehmens UNITY AG. Seit 2003 ist er Mitglied von acatech – Deutsche Akademie der Technikwissenschaften und seit 2012 Vizepräsident. Ferner ist Jürgen Gausemeier Vorsitzender des Clusterboards des BMBF-Spitzenclusters „Intelligente Technische Systeme Ostwestfalen-Lippe (it's OWL)“.

Technische Dokumentation in der Smart Factory

**Denis Störkle M.Sc., Dipl.-Wirt.-Ing. André Barthelmey,
Univ.-Prof. Dr.-Ing. Jochen Deuse, Univ.-Prof. Dr.-Ing. Bernd Kuhlenkötter**
*Technische Universität Dortmund, Institut für Produktionssysteme
Leonhard-Euler-Str. 2 , 44227 Dortmund
Tel. +49 (0) 231 / 75 55 601 , Fax. +49 (0) 231 / 75 55 616
E-Mail: denis.stoerkle@ips.tu-dortmund.de*

Dipl.-Inf. Stefan Magerstedt
*KHS GmbH
Juchostrasse 20 , 44143 Dortmund
Tel. +49 (0) 231 / 56 91 442 , Fax. +49 (0) 231 / 56 94 14 42
E-Mail: stefan.magerstedt@khs.com*

Zusammenfassung

Der vorliegende Aufsatz beschreibt neue Methoden zur Erstellung und Bereitstellung der technischen Dokumentation von Maschinen und Anlagen. Diese werden im Rahmen des vom BMBF geförderten Verbundprojektes CyberSystemConnector gemeinsam von Forschungseinrichtungen und Industrieunternehmen entwickelt und anhand von realen Anwendungsfällen validiert. Dabei bildet die Konformität der Dokumentations- und Bereitstellungskonzepte mit der EG-Maschinenrichtlinie ein wichtiges Kriterium.

Schlüsselworte

Cyber-physische Systeme, technische Dokumentation, Smart Factory, Virtuelles Anlagenabbild

Technical documentation in the Smart Factory

Abstract

In this paper, the authors propose new methodologies for creation and deployment of technical documentation for machinery and plants. These are developed in the research project CyberSystemConnector which is funded by the German Federal Ministry of Education and Research (BMBF). The project consortium consists of research institutions and industrial companies. The developed methodologies are validated using real industrial use-cases and still satisfy the Directive 2006/42/EC of the European Parliament and of the Council.

Keywords

Cyber-physical systems, technical documentation, smart factory, virtual machine representation

1 Einleitung

Hersteller von Maschinen und Anlagen sind bei der Übergabe und Inbetriebnahme von Maschinen und Anlagen nach der Maschinenrichtlinie 2006/42/EG verpflichtet, eine normgerechte und aktuelle technische Dokumentation an ihre Kunden zu übergeben [eg212]. Dem hohen Zeit- und Kostenaufwand für die Erstellung und Pflege der technischen Dokumentation ist es geschuldet, dass der Stand der Unterlagen nach der Auslieferung und dem Anlauf der Anlagen kaum aktuell gehalten werden kann (siehe Bild 1). In der Folge werden obligatorische Anpassungen, begründet durch Änderungen des Produktionssystems bspw. im Zuge von Produktwechsel, Produktionsvolumenanpassungen oder Optimierungsmaßnahmen, meist vollständig vernachlässigt oder nur als statischer Zusatz der Dokumentation hinzugefügt [SA01]. Auf eine vollständige Anpassung der Daten innerhalb der IT-Systeme wird dagegen aus kapazitiven Gründen in der Regel verzichtet [SPT+13]. Neben der rechtlichen Absicherung von Anlagenherstellern, bilden technische Dokumentationsinhalte in nahezu allen Unternehmensbereichen, jedoch insbesondere in der Instandhaltung die Datengrundlage zur Planung und Umsetzung der Prozesse.

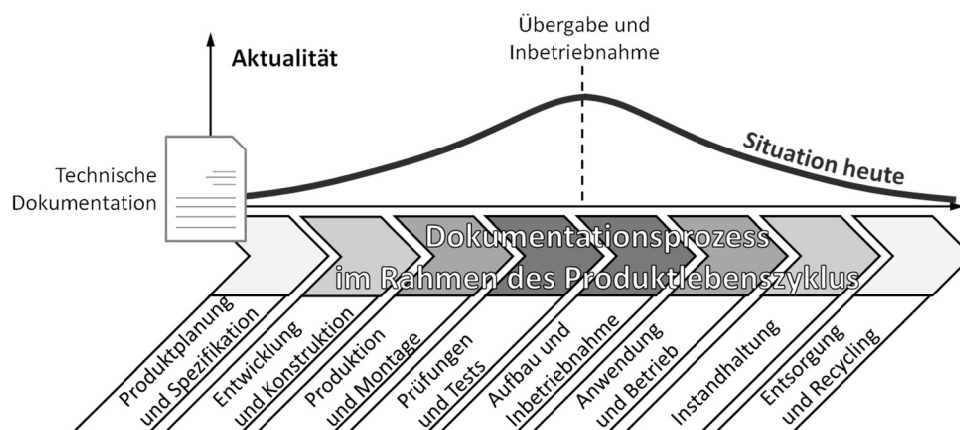


Bild 1: Aktualität der technischen Dokumentation entlang des Produktlebenszyklus

Mit dem Fortschreiten der IT-Integration auf dem Weg zur Smarten Fabrik und ferner der Modularisierung von Anlagenkomponenten [WEN+07] nehmen sowohl die Anforderungen an die technische Dokumentation als auch die technischen Möglichkeiten zu ihrer effizienten Erstellung und Nutzung stetig zu. Vor diesem Hintergrund sind angepasste Dokumentations- und Instandhaltungsprozesse erforderlich und mittlerweile umsetzbar, die zur Beherrschung der zunehmenden Komplexität und Variabilität von Produktionssystemen beitragen.

2 Stand der Technik

2.1 Technische Dokumentation im Maschinen- und Anlagenbau

Mit Hilfe der technischen Dokumentation soll der Anwender in die Lage versetzt werden, seine Maschine sicher zu betreiben. Dementsprechend sind alle hierfür relevanten Informationen in der technischen Dokumentation zusammengetragen.

Das zentrale Element der technischen Dokumentation bildet die Betriebsanleitung, deren integrale Bestandteile im Anhang I, Abschnitt 1.7.4 der Maschinenrichtlinie 2006/42/EG erläutert sind. Unter anderem zählen hierzu Hinweise zur Inbetriebnahme, Verwendung, Einrichtung, Wartung und Instandsetzung der Maschine mit den entsprechenden Warn- und Sicherheitshinweisen. Neben der Betriebsanleitung umfasst die technische Dokumentation die vollständige und aktuelle Teileliste. Außerdem fordern die Kunden vermehrt darüber hinausgehende, anwendungsspezifisch angepasste Dokumente. Hierzu zählen neben der Trainingsdokumentation für das Anlernen von Personal auch spezifische Dokumente, die einen notwendigen Arbeitsgang detailliert erklären, um Probleme und Missverständnisse zu vermeiden. Gängige Dokumentationstypen sind die s.g. SOP (Standard Operating Procedure), die eine genaue Ablaufbeschreibung für einen Prozess umfasst, sowie die s.g. OPL (One Point Lesson), eine verkürzte Form, die jeweils einen einzelnen Prozessschritt abbildet.

Im Rahmen der automatisierten Dokumentationserstellung ist zu beachten, dass es sich bei der Betriebsanleitung um ein „gegenständliches“ Dokument handelt, womit ein Ausdruck nach jeder Aktualisierung durch die Maschinenrichtlinie vorgegeben ist.

Im Allgemeinen wird im Rahmen der technischen Dokumentation zwischen der internen und der externen technischen Dokumentation unterschieden. Unter der externen technischen Dokumentation sind alle Dokumente, die der Lieferant dem Kunden für den Betrieb einer Maschine bereitstellt, zusammengefasst. Im Wesentlichen sind dies die Dokumente, die nach Maschinenrichtlinie bzw. nach Landesgesetzen gefordert werden sowie darüber hinausgehende anwendungsspezifisch angepasste Dokumente. Die Gesamtheit aller Daten und Dokumente, die der Lieferant über den Produktlebenszyklus hinweg erstellt, bilden die interne technische Dokumentation, die nicht vollständig ausgeliefert wird, sondern größtenteils beim Lieferanten verbleibt. Unter anderem gehören hierzu alle Daten und Dokumente, die während der Planung und Konstruktion erzeugt werden. Diese dienen dem Lieferanten als Grundlage, bspw. zur Planung von Umbauten an der ausgelieferten Maschine. Zusätzlich werden die Konstruktionsdokumente im Schadensfall herangezogen, um die Maschinensicherheit nachweisen zu können.

2.2 Instandhaltung und Industrie 4.0

Der aktuelle Trend in der Produktion hin zu Cyber-Physischen Produktionssystemen (CPPS) betrifft insbesondere den deutschen Maschinen- und Anlagenbau, da dieser in

Zukunft die Vorreiterrolle für diese hoch technologisierten Systeme einnehmen soll. Innerhalb eines CPPS besitzen alle beteiligten Akteure und Komponenten (Mensch, Maschine, Produkte) die Fähigkeit zur Erfassung des eigenen Status und der Umwelt. Weiterhin wird durch CPPS die Kommunikation der Arbeitsmittel untereinander, als auch die Kommunikation zwischen Arbeitsmittel und Menschen ermöglicht. Durch diese Erweiterung traditioneller Produktionssysteme entsteht im produzierenden Umfeld die sogenannte „Smart Factory“, in der die Verknüpfung der realen mit der digitalen Welt sowie die Ablösung der zentralen Produktionssteuerung durch eine dezentrale Selbstorganisation umgesetzt ist [WJE+05].

Damit der Mensch dieses komplexe, dynamische System beherrschen und darüber hinaus von der Vielzahl der neuen Informationen während seiner Arbeit profitieren kann, sind die vorhandenen Planungs-, Steuerungs- und Unterstützungssysteme mit den neuen technologischen und organisatorischen Möglichkeiten zu erweitern. Einen wesentlichen Methodenbaustein der „Smart Factory“ bildet Augmented Reality (AR). In mehreren Forschungsprojekten werden die Möglichkeiten von AR-Techniken für die Instandhaltung untersucht. An der Fakultät für Informatik der Columbia-Universität wurden bspw. bereits im Projekt „Augmented Reality for Maintenance and Repair“ AR-Applikationen für die Unterstützung von Instandhaltungsaufgaben in der Luftfahrt untersucht [HF11b][HF11a]. Hierbei wurde unter anderem gezeigt, dass Instandhalter, die mit der Maschine nicht vertraut waren, durch AR schneller ihre Arbeiten an dieser durchführen konnten. U.a. aufbauend auf dieser Erkenntnis werden zurzeit im Rahmen des Projektes „Telemaintenance mit Augmented Reality“ des Fraunhofer Institutes für Graphische Datenverarbeitung IGD und des Verbundprojektes „Ressourcen-Cockpit für Sozio-Cyber-Physische Systeme in der Produktion, Wartung und Instandhaltung“ Strategien zur Instandhaltung von Maschinen und Anlagen entwickelt [EKR+13][WBE+13]. Im Fokus steht die kontextbasierte Unterstützung von Instandhaltungsaufgaben durch den Einsatz mobiler Endgeräte. AR-Technologien sind darüber hinaus bereits aus dem wissenschaftlichen Umfeld in die industrielle Praxis überführt. Anbieter für industrielle AR-Anwendungen, sind bspw. die Firmen Metaio GmbH und Scope AR. Die Metaio GmbH entwickelt und vertreibt AR-Lösungen, die vom Kunden direkt genutzt, bzw. in Kundenapplikation integriert werden können. Die Expertise der Scope AR liegt in der AR-basierten Darstellung von Trainingsinhalten.

Die Standardisierung innerhalb der Produktdatentechnologie spielt auch in Zeiten von CPPS eine wesentliche Rolle. Darüber hinaus besteht der Trend zur zentralen, bzw. Cloud-basierten Datenhaltung zur Vermeidung von Redundanzen und überflüssiger Versionierung. Die Möglichkeit der einheitlichen Datenhaltung in einem unternehmensbereichs- und softwaresystemübergreifenden Format bietet bspw. AutomationML. Hierbei handelt es sich um einen offenen Standard zur Speicherung von Anlagenplanungsdaten mit dem Ziel, den verlustfreien Datenaustausch zwischen verschiedenen Softwarewerkzeugen in der Planungsphase einer Anlage realisieren zu können. AutomationML basiert auf der Extensible Markup Language (XML) und integriert weitere Da-

teiformate um Topologie, Geometrie, Kinematik und Logik abbilden zu können. Zur Beschreibung der Anlagentopologie wird das Datenformat CAEX (IEC 62424) herangezogen. Geometrische und kinematische Beziehungen von Objekten werden mit Hilfe von COLLADA implementiert. Die Abbildung der Ablauflogik in der Maschine sowie deren Verhalten findet in PLCopen statt [Dra10][BSK14]. Mit Hilfe von AutomationML ist es demnach möglich, eine Gesamtanlage vollständig zu beschreiben. Dies gilt sowohl für die mechanische, als auch für die elektrische Konstruktion.

3 Systematik smarter Dokumentation

3.1 Smarte technische Dokumentation von Maschinen und Anlagen

Die Anwendung der mit Industrie 4.0 einhergehenden Technologien und Ideen birgt großes Potenzial für die technische Dokumentation, sowohl auf Hersteller- als auch auf Kundenseite. Vor diesem Hintergrund wird im Verbundprojekt CyberSystemConnector (CSC) eine neue Systematik zur smarten Erstellung und Nutzung technischer Dokumentation entwickelt [BBH+12][BSKD14]. Zentrale Forschungsschwerpunkte sind hierbei die Verwendung der Methodenbausteine von CPPS, sowie die Berücksichtigung der Anforderungen einer „Smart Factory“, was die bisherigen Forschungsvorhaben zur Bereitstellung von Dokumentationsinhalten nicht adressierten.

Gegenüber dem traditionellen Dokumentationsprozess wird durch die Systematik Aktualität der technischen Dokumentation über den gesamten Produktlebenszyklus einer Anlage erreicht. Dadurch ist der Lieferant in der Lage, Umbauten effektiv, zu planen, sowie bei einem Defekt an der Anlage schneller reagieren zu können, da ihm alle Daten über den aktuellen Zustand der Maschine stets zur Verfügung stehen. Ein weiteres Potential liegt in der anwendungsorientierten Bereitstellung von Dokumentationsinhalten, da hierdurch kundenseitige Instandhaltungstätigkeiten schneller und effizienter durchgeführt werden können, was die Verringerung von Stillstandzeiten bewirkt.

3.2 Aktualität der technischen Dokumentation während des gesamten Lebenszyklus einer Anlage

Die Aufrechterhaltung der Aktualität technischer Dokumentation stellt zum einen eine durch die Maschinenrichtlinie gegebene Notwendigkeit dar und zum anderen ist sie seitens der Betreiber von Maschinen und Anlagen, ebenso wie herstellerseitig, eine Grundvoraussetzung für effiziente Planungsprozesse. Der Kunde betrachtet aktuelle technische Dokumentation als Mehrwert, da Anlagenstillstände in Folge von Service- und Instandhaltungstätigkeiten verkürzt werden. Der Zugang zum aktuellen Zustand der Maschinen versetzt Anlagenbauer in die Lage, jede durch den Kunden entstandene Mo-

difikation zu reproduzieren. Serviceaufgaben können so zielgerichtet vorbereitet werden, während Kunden via Fernwartung maximal unterstützt werden können.

Um eine aktuelle technische Dokumentation über den gesamten Produktlebenszyklus zu gewährleisten, müssen Änderungen des realen Systems unmittelbar mit ihrer Umsetzung vollständig erfasst und in der Dokumentation beschrieben werden, sowie alle resultierenden dokumentationsrelevanten Aspekte, wie bspw. veränderte Sicherheitsbestimmungen. Eine solche umfassende, bestenfalls automatische Anpassung der technischen Dokumentation ist jedoch zur Zeit noch eine Vision, da heute Schnittstellen zwischen Hardware-Komponenten und Softwaresystemen fehlen, die zur Beschreibung der Komponenten oder zur Dokumentationserstellung eingesetzt werden.

3.3 Anwendungsorientierte Bereitstellung von Dokumentationsinhalten

Aufgrund der Vielzahl an Informationen im Zusammenhang mit den verschiedenen Anwendungsgebieten steigt die Komplexität der technischen Dokumentation. Dies spiegelt sich insbesondere in nicht-kompatiblen Datentypen und in fehlender Struktur wider. Die Suche und Gewichtung relevanter Dokumentationsinhalte beschränkt die Effizienz nahezu aller Geschäftsprozesse. Durch eine intelligente Verwaltung der Ist-Daten kann eine zielgenaue Bereitstellung der für eine Anwendung relevanten Daten realisiert werden. Zusätzlich zum Anwendungsbezug, kann ein Personenbezug der Daten implementiert werden, so dass die Dokumentationsinhalte nicht nur auf die Tätigkeit, sondern darüber hinaus auf das Persönlichkeitsprofil des jeweiligen Anwenders zugeschnitten werden. Die Potentiale dieser effizienten Nutzung einzelner Dokumentationsinhalte sind unumstritten und werden bereits in vielen Forschungsvorhaben, u.a. im Rahmen der Industrie 4.0 Initiative fokussiert [BMBF14]. Der wesentliche Mehrwert anwendungsbezogen bereitgestellter Dokumentationsinhalte erschließt sich jedoch erst in Verbindung mit einer aktuellen Datengrundlage. Im Maschinen- und Anlagenbau entspricht diese der aktuellen Anlagenkonfiguration beim Kunden. Für diese Branche können folgende Gruppen unterschieden werden:

- Technischer Redakteur (bspw. Dokumentationserstellung und -Pflege)
- Interner Anwender (bspw. Angebotserstellung, Service)
- Externer Anwender (bspw. Instandhaltung, Anlagenbetrieb)

Durch die Umsetzung der Systematik smarterer Dokumentation wird der technische Redakteur insbesondere bei der Suche und Strukturierung von Dokumentationsinhalten bei der initialen Erstellung auf Komponentenebene durch eine weitgehende Automatisierung unterstützt. Darüber hinaus wird in der Folge der Pflegeaufwand der ausgelieferten Dokumentation minimiert. Interne Anwender profitieren durch aktuelle Dokumentationsinhalte innerhalb ihrer Planungstools hinsichtlich einer Optimierung der Vorbereitung und Planung von Instandhaltungstätigkeiten und Folgeaufträgen. Für externe Anwender ist Augmented Reality derzeit der am besten geeignete Ansatz, um komplexe

Tätigkeiten an der Anlage effektiv anzuleiten und bildet in diesem Zusammenhang eine sinnvolle Erweiterung der externen technischen Dokumentation, insbesondere für den Bereich Instandhaltung.

3.4 Zentrale Herausforderungen in der technischen Realisierung der Systematik

Smarte Komponenten wurden bereits von Forschungseinrichtungen und einigen Unternehmen pilothaft eingesetzt, insbesondere für Greenfield-Planungen, sind jedoch noch nicht Stand der Technik. Eine automatische Erkennung der einzelnen Komponenten innerhalb der realen Anlage und deren aktueller Konfiguration stellt daher weiterhin eine Herausforderung für die Umsetzung der automatisierten Aktualisierung von Anlagendokumentation dar.

Des Weiteren bildet eine durchgängige Struktur zur Ablage der im Unternehmen vorhandenen Anlageninformationen eine wesentliche Voraussetzung für die automatische Erstellung richtlinienkonformer Anlagendokumentation. Für eine automatisierte Bereitstellung dieser Inhalte in Form von kontextspezifischer Dokumentation muss die Struktur mit den Anlageninformationen zusätzlich mit dem sogenannten Rahmendokument verknüpft sein, welches die Struktur der Dokumentation vorgibt.

4 Methoden zur technischen Realisierung

4.1 Bauteilidentifikation für smarte Dokumentation

Mit der Industrie 4.0 Initiative wird der Einsatz von Informations- und Kommunikationstechnik in Produktionssystemen weiter forciert, so dass verbaute Anlagenkomponenten zunehmend „intelligenter“ werden. Eine umfassende Implementierung dieser Intelligenz in alle Komponenten einer Maschine oder Anlage ist heute noch nicht realisiert. Innerhalb des Forschungsvorhabens CSC werden daher zurzeit die im Folgenden erklärten Ansätze zur Bauteilerkennung verfolgt.

Einige Anlagenkomponenten auf hohen Aggregationsebenen verfügen bereits über eine eigene Steuerung. Komponenten der ersten Stufe, wie bspw. Roboter- oder Bearbeitungszellen sind durch die Steuerung in der Lage Informationen, wie den Bearbeitungsfortschritt, zu erfassen. Über eine Verbindung, bspw. Ethernet kann ein Netzwerk aus diesen Modulen aufgebaut werden. Über Protokolle, wie TCP/IP, können sich diese eindeutig identifizieren und Informationen, bspw. bezüglich der aktuellen Konfiguration austauschen. Komponenten der zweiten Stufe sind zu einer Selbstbeschreibung noch nicht fähig, jedoch über das Feldbussystem angebunden, über das sie bereits Prozessdaten bereitstellen und abfragen. Für diese Komponenten bietet die Verwendung von Netzwerkprotokollen eine Möglichkeit zur Identifikation und Kommunikation mit Rechnern in den Planungsebenen. Komponenten, die aufgrund ihrer Struktur und Funk-

tion nicht zu einer Anlagenkomponente aggregiert werden können, die an den Feldbus angebunden ist, können nicht mit solchen Protokollen erfasst werden und bilden hier die dritte Stufe. Chancen zur Erfassung der Komponenten der dritten Stufe birgt u.a. die Bildverarbeitung, da Kameras heutzutage kostengünstig und bereits vielfach in Produktionssystemen eingebaut und etabliert sind. Die optische Identifizierung könnte bspw. mit Hilfe von mobilen Endgeräten oder über zentral an der Anlage angebrachten Kameras geschehen. Die optische Identifikation liefert einen hervorragenden Ansatz, um die groben Strukturelemente einer Maschine oder Anlage, insbesondere der passiven Bauelemente, zu erkennen. Auch unbekannte Veränderungen an der Maschine können damit erfasst werden, um etwaige Rechtsansprüche gegenüber dem Kunden absichern zu können. Optische Identifikation kann zum einen dynamisch durch Kameras an beweglichen Objekten, wie mobilen Endgeräten (bspw. Smartphones, Tablets), automatisierten Transportwagen etc. erfolgen („structure from motion“) [Ull79]. Zum anderen können statische Kameras, welche in der jeweiligen Produktionsumgebung angebracht sind, einen permanenten Abgleich mit dem virtuellen Abbild liefern (bspw. 3D Scan mit Tiefensensor) [IKH+11]. Die erforderliche Technologie ist bereits in diversen Anwendungsfeldern im Einsatz. So werden mit statischen Kameras über Punktwolkenerkennung bereits heute dreidimensionale Gegenstände oder Personen erfasst. Bestehende Einführungsbarrieren im industriellen Umfeld bilden zurzeit bspw. die Komplexität der sicheren und eindeutigen Auswertung von Punktwolken, sowie Vorbehalte seitens Betriebsrat bzw. Compliance Restriktionen gegenüber dem Einsatz von Endgeräten mit integrierter Kamera. Voraussetzung für den Einsatz dieser Technologie ist die freie Zugänglichkeit der Komponenten, was bei eingehausten Anlagen nicht immer gewährleistet werden kann.

4.2 Datenstrukturierung für smarte Dokumentation

Während die Problematik der Identifikation durch technologische Entwicklungen und Standardisierung im Bereich CPPS einen aktuellen Forschungsschwerpunkt bildet, bleibt die Erstellung normkonformer Dokumentation weiterhin ein Randthema [BMBF14]. Aus diesem Grund stellt die automatische Überführung der Anlagendaten in die Dokumentationsstruktur die größte Herausforderung auf dem Weg zur „smarten Dokumentation“ dar.

Zur Bewältigung der strukturellen Unterschiede zwischen abgelegten Anlagendaten und dem Rahmendokument richtlinienkonformer Anlagendokumentation wird im Projekt CSC zurzeit die Verwendung des Datenaustauschformats AutomationML und gleichzeitig dessen Erweiterung um den Dokumentationsaspekt verfolgt. Ziel ist es, mit dem virtuellen Abbild eine Strukturierung zu erstellen, die eine eindeutige Anordnung der im Produktentstehungsprozess auftretenden Anlageninformationen ermöglicht. Bild 2 zeigt die Erstellung und Nutzung des virtuellen Abbilds der Anlage. Alle während der mechanischen Konstruktion und der Elektrokonstruktion erzeugten Daten sowie Ersatzteil- und Verschleißinformationen werden in AutomationML überführt. Im Rahmen der ini-

tialen Dokumentationserstellung pflegt die technische Redaktion die Zuordnungen von Dokumentationsinhalten zu den entsprechenden Anlagenkomponenten sowie fehlende Verknüpfungen zwischen mechanischen und elektronischen Komponenten nach. Bei Auslieferung der Anlage werden die realen Anlagenkomponenten mit ihrer jeweiligen virtuellen Repräsentation in Verbindung gebracht, sodass die Zuordnung zwischen Anlagenkomponenten und Dokumentationsinhalten eindeutig ist. Komponenten der ersten und zweiten Stufe sind bei Inbetriebnahme bereits mit ihrer virtuellen Repräsentation verknüpft, während die Verknüpfung für Komponenten der dritten Stufe dagegen erst im Zuge der Inbetriebnahme erfolgt. Dabei wird die aktuelle technische Dokumentation auf Maschinenebene weitestgehend automatisiert generiert. Jene Teile der Dokumentation, die nicht automatisiert erzeugt werden können, bedürfen eines Eingriffs durch den technischen Redakteur. Dieser wird hierbei bestmöglich unterstützt, indem ihm alle Informationen über die aktuelle Konfiguration der Anlagen zur Verfügung gestellt werden. Es handelt sich hierbei um Dokumentationsinhalte die sich aus dem komplexen Zusammenspiel mehrerer Komponenten ergeben, welches bei der initialen Erstellung der einzelnen Dokumentationen noch nicht berücksichtigt werden.

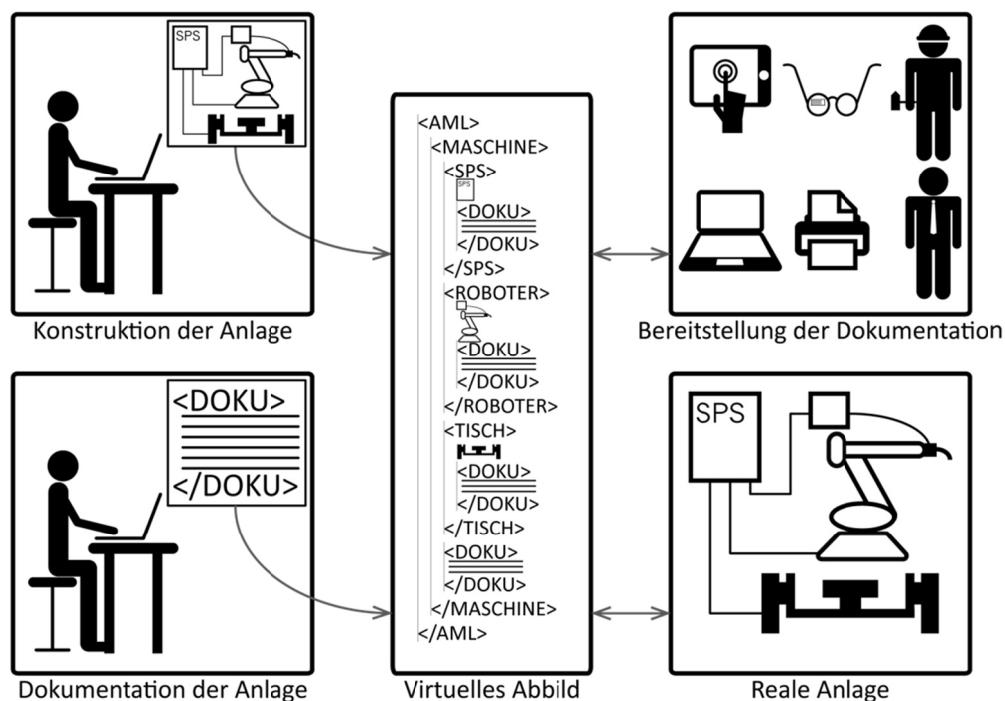


Bild 2: Erstellung und Nutzung des virtuellen Abbilds

Der kontinuierliche Abgleich der realen Anlage und dem virtuellen Abbild bildet im Zusammenspiel mit dem Ausgabemedium die ideale Grundlage für die anwendungsbezogene und personalisierte Bereitstellung der Dokumentation, da hierdurch der Benutzungskontext, die geplante Tätigkeit, das benötigte Ausgabeformat und der Anwender automatisch erfasst werden können. Ein Anwendungsszenario stellt bspw. eine mit AR angeleitete Inspektionsmaßnahme dar. Hierbei ist der Inhalt und dessen Darstellung von der aktuellen Anlagenkonfiguration, dem Instandhaltungsauftrag bezogen auf die Loka-

lität und den Fähigkeiten der Person abhängig, während die Darstellungsform durch das eingesetzte Medium zur Visualisierung, bspw. eines mobilen Endgeräts vorgegeben ist.

5 Resümee und Ausblick

Der Aufsatz unterstreicht die Notwendigkeit der Verbesserung von Dokumentationserstellung und Nutzung, um die heutigen Herausforderungen im Maschinen- und Anlagenbau und stellt ein neuartiges Konzept zur Übertragung der Methodenbausteine von CPPS auf den Dokumentationsprozess vor. Dazu werden mit jederzeit aktueller Dokumentation und anwendungsorientierter Bereitstellung von Dokumentationsinhalten die beiden wesentlichen Potentiale dieses neuen Ansatzes diskutiert.

Für die Sicherung der Aktualität technischer Dokumentation nach jeder Änderung des physischen Systems werden die automatisierte Identifikation dieser Änderungen und die Übertragung von Planungsdaten in die Struktur des Rahmendokuments als zu lösende Herausforderungen identifiziert. Darüber hinaus werden die Potentiale und die Realisierbarkeit des Einsatzes von AR im Rahmen des Konzepts erläutert.

Die bisherigen Projekterfahrungen lassen keinen Zweifel, dass ein Konzept, wie das des CyberSystemConnectors dazu beitragen kann, zurzeit entkoppelte Teilaspekte einer Unternehmung, wie die Produktentstehung und die Instandhaltung in einen unternehmensweiten Ansatz zum bereichsübergreifenden Informationsaustausch zu integrieren. Unternehmensprozesse können in Bezug auf Kontrollschleifen und Schnittstellen, die heute noch Fehlerquellen oder Kapazitätsengpässe darstellen, vereinfacht werden. Darüber hinaus kann situationsbezogenes Expertenwissen für jeden Mitarbeiter frei zugänglich gemacht werden.

6 Förderhinweis

Dieses Forschungs- und Entwicklungsprojekt wird mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) im Rahmenkonzept „Forschung für die Produktion von morgen“ gefördert und vom Projektträger Karlsruhe (PTKA) betreut. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung

BETREUT VOM



PTKA
Projektträger Karlsruhe
Karlsruher Institut für Technologie

Literatur

- [BBH+12] BILEK, EMINE ; BUSCH, FELIX ; HARTUNG, JOCHEN ; SCHEELE, CARSTEN ; THOMAS, CARSTEN ; DEUSE, JOCHEN ; KUHLENKÖTTER, BERND: Intelligente Erstellung und Nutzung von Maschinendokumentation. In: ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb (2012), September, S. 652-656. ISSN 0947-0085
- [BRI14] DIE BUNDESREGIERUNG: Industrie 4.0, <http://www.hightech-strategie.de/de/59.php>, Stand 18.11.2014
- [BMBF14] BUNDESMINISTERIUM FÜR BILDUNG UND FORSCHUNG: Industrie 4.0 – Innovationen für die Produktion von morgen. 2014
- [BSK14] BARTELT, MATTHIAS; SCHYJA, ADRIAN; KUHLENKÖTTER, BERND: More than a Mockup. In: Production Engineering 8 (2014), Nr. 6, S. 727-735. <http://dx.doi.org/10.1007/s11740-014-0575-6>. DOI 10.1007/s11740-014-0575-6. ISSN 0944-6524
- [BSKD14] BARTHELMEY, ANDRÉ; STÖRKLE, DENIS; KUHLENKÖTTER, BERND; DEUSE, JOCHEN: Cyber Physical Systems for Life Cycle Continuous Technical Documentation of Manufacturing Facilities. In: Procedia {CIRP} 17 (2014), Nr. 0, 207-211. <http://dx.doi.org/10.1016/j.procir.2014.01.050>. DOI 10.1016/j.procir.2014.01.050. ISSN 2212-8271. Variety Management in Manufacturing Proceedings of the 47th {CIRP} Conference on Manufacturing Systems
- [Dra10] DRATH, RAINER: Datenaustausch In Der Anlagenplanung Mit AutomationML: Integration von CAEX, PLCopen XML und COLLADA. Springer, 2010 (VDI-Buch). <http://books.google.ca/books?id=o5XfzefXSyAC>. ISBN 9783642046742
- [eg212] Richtlinie 2006/42/EG des Europäischen Parlaments und des Rates, 17. Mai 2006. <http://ec.europa.eu/enterprise/sectors/mechanical/documents/legislation/machinery>. Version: 02 2012
- [EKR+13] ENGELKE, T.; KEIL, J.; ROJTBERG, P.; WIENTAPPER, F.; WEBEL, S.; BOCKHOLT, U.: Content first – A concept for industrial augmented reality maintenance applications using mobile devices. In: Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on, 2013, S. 251-252
- [HF11a] HENDERSON, S.; FEINER, S.: Exploring the Benefits of Augmented Reality Documentation for Maintenance and Repair. In: Visualization and Computer Graphics, IEEE Transactions on 17 (2011), Oct, Nr. 10, S. 1355-1368. <http://dx.doi.org/10.1109/TVCG.2010.245>. DOI 10.1109/TVCG.2010.245. ISSN 1077-2626
- [HF11b] HENDERSON, S. J.; FEINER, STEVEN K.: Augmented reality in the psychomotor phase of a procedural task. In: Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on, 2011, S. 191-200
- [IKH+11] IZADI, SHAHRAM; KIM, DAVID; HILLIGES, OTMAR; MOLYNEAUX, DAVID; NEWCOMBE, RICHARD; KOHLI, PUSHMEET; SHOTTON, JAMIE; HODGES, STEVE; FREEMAN, DUSTIN; DAVISON, ANDREW; FITZGIBBON, ANDREW: KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera, ACM Symposium on User Interface Software and Technology, October 2011, S. 559-568
- [SA01] STORR, A., DREYER, J.: Informationsmodellierung für die technische Dokumentation. ZWF – Zeitschrift für wirtschaftlichen Fabrikbetrieb, (10) 2001, S. 559-565
- [SPT+13] SCHUH, G.; POTENTE, T.; THOMAS, C.; HAUPTVOGEL, A.: High Resolution Production Management – Hochauflösende Produktionssteuerung auf Basis kybernetischer Unterstützungssysteme. wt-online 2-2013 SONDERHEFT INDUSTRIE 4.0, S. 096-099
- [SA01] STORR, A., DREYER, J.: Informationsmodellierung für die technische Dokumentation. ZWF – Zeitschrift für wirtschaftlichen Fabrikbetrieb, (10) 2001, S. 559-565

- [Ull79] ULLMAN, S.: The Interpretation of Structure from Motion. In: Proceedings of the Royal Society of London B: Biological Sciences 203 (1979), Nr. 1153, S. 405-426.
<http://dx.doi.org/10.1098/rspb.1979.0006>. DOI 10.1098/rspb.1979.0006. ISSN 0080-4649
- [vdi06a] Richtlinie VDI 4499 Blatt 1: Digitale Fabrik – Grundlagen. 2006
- [vdi06b] Richtlinie VDI 4500-1: Technische Dokumentation – Begriffsdefinitionen und rechtliche Grundlagen. 2006
- [vdi06c] Richtlinie VDI 4500-2: Technische Dokumentation – Organisieren und Verwalten. 2006
- [vdi06d] Richtlinie VDI 4500-3: Erstellen und Verteilen von elektronischen Ersatzteilinformationen. 2006
- [vdi06e] Richtlinie VDI 4500-4: Dokumentationsprozess – Planen, Gestalten, Erstellen. 2006
- [WBE+13] WEBEL, SABINE; BOCKHOLT, ULI; ENGELKE, TIMO; GAVISH, NIRIT; OLBRICH, MANUEL; PREUSCHE, CARSTEN: An augmented reality training platform for assembly and maintenance skills. In: Robotics and Autonomous Systems 61 (2013), Nr. 4, 398 - 403.
<http://dx.doi.org/10.1016/j.robot.2012.09.013>. DOI 10.1016/j.robot.2012.09.013. ISSN 0921-8890. Models and Technologies for Multi-modal Skill Training
- [WJE+05] WESTKÄMPER, E.; JENDOUBI, L.; EISSELE, M.; ERTL, T.: Smart Factories – Bridging the gap between digital planning and reality; Tagungsbeitrag: The 38th CIRP-International Seminar on Manufacturing Systems, 16.-18.05.2005; Florianapolis, Brazil
- [WEN+07] WIENDAHL, H.-P.; ELMARAGHY, H.A.; NYHUIS, P.; ZÄH, M.F.; WIENDAHL, H.-H.; DUFFIE, N.; BRIEKE, M.: Changeable Manufacturing – Classification, Design and Operation. In: {CIRP} Annals - Manufacturing Technology 56 (2007), Nr. 2, S. 783-809.
<http://dx.doi.org/10.1016/j.cirp.2007.10.003>. DOI 10.1016/j.cirp.2007.10.003. ISSN 0007-8506

Autoren

Denis Störkle M.Sc. schloss 2011 sein Studium der Elektrotechnik mit dem Schwerpunkt Automatisierung an der Hochschule Darmstadt ab und begann seine Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Produktionssysteme der TU Dortmund. Seine Arbeitsschwerpunkte umfassen neben industriellen Echtzeitapplikationen, Cyberphysikalische Systeme und technische Dokumentation.

Dipl.-Inf. Stefan Magerstedt studierte Ingenieur-Informatik an der Universität Dortmund und durchlief nach seinem Abschluss mehrere Positionen innerhalb der KHS GmbH in Dortmund. Dazu zählen u.a. Client-/Server-Programmierung, IT-Projektleitungen und die Abteilungsleitung der technischen Dokumentation. Aktuell liegt sein Arbeitsschwerpunkt bei der Einführung eines unternehmensweiten, automatisierten Produktkonfigurators.

Dipl.-Wirt.-Ing. André Barthelmey studierte Wirtschaftsingenieurwesen mit der Vertiefung Produktionsmanagement an der TU Dortmund. Seit 2013 ist er am Institut für Produktionssysteme (IPS) an der TU Dortmund beschäftigt. Seine Arbeitsschwerpunkte liegen in den Bereichen technische Dokumentation und Instandhaltung sowie dem Industrial Engineering.

Univ.-Prof. Dr.-Ing. Jochen Deuse ist seit 2005 Leiter des Lehrstuhls für Arbeits- und Produktionssysteme (APS) und leitet seit 2012 das aus dem Lehrstuhl hervorgegangene Institut für Produktionssysteme (IPS) der Technischen Universität Dortmund. Er promovierte 1998 am Laboratorium für Werkzeugmaschinen und Betriebslehre (WZL) der RWTH Aachen. Anschließend war er in leitender Funktion für die Bosch-Gruppe im In- und Ausland tätig.

Univ.-Prof. Dr.-Ing. Bernd Kuhlenkötter promovierte 2001 am Lehrstuhl für Maschinenelemente, -gestaltung und Handhabungstechnik der TU Dortmund, an dem er bis 2005 als Oberingenieur und stellvertretender Lehrstuhlleiter tätig blieb. Er übernahm dann die Professurvertretung für das Fach „Industrielle Robotik und Handhabungssysteme“ am Institut für Roboterforschung. Während dieser Zeit engagierte er sich u. a. in den Sonderforschungsbereichen 696 und 708. Wesentliche Forschungsfelder waren hier die Prozesssimulation von robotergestützten Fertigungsverfahren und die Entwicklung innovativer Automatisierungslösungen.

Anfang 2007 wechselte Bernd Kuhlenkötter als Entwicklungsleiter zur ABB Automation GmbH. Hier umfasste sein Verantwortungsbereich die Entwicklung neuer Robotertechnologien in Zusammenarbeit mit internationalen ABB-Entwicklungszentren.

Zum 01.04.2009 folgte Bernd Kuhlenkötter dem Ruf der TU Dortmund auf den Lehrstuhl für „Industrielle Robotik und Produktionsautomatisierung“ der Fakultät Maschinenbau. Seit 2012 leitet er das Institut für Produktionssysteme in Dortmund und ist Präsident der Wissenschaftlichen Gesellschaft für Montage, Handhabung und Industrierobotik (MHI e. V.)

Portfolio-Management für Elektroantriebe in Powertools bei Hilti: Herausforderungen und Lösungsansätze

Dr.-Ing. Josef Ponn

Hilti Entwicklungsgesellschaft mbH

Hiltistraße 6, 86916 Kaufering

Tel. +49 (0) 8191 / 90 60 46

E-Mail: josef.ponn@hilti.com

Zusammenfassung

Das Hilti Produktportfolio beinhaltet ein breites Spektrum an Elektrogeräten für Profianwender in der Baubranche (z.B. Bohrhämmer, Schrauber, Sägen, Winkelschleifer und Diamantkernbohrsysteme). Der Elektroantrieb ist ein wichtiges Subsystem, ein entscheidender Treiber für die Differenzierung der Geräte, jedoch ebenso ein Verursacher von zunehmender Komplexität. Hierbei haben im Zuge der Mechatronisierung der Geräte insbesondere die Elektronik Hardware und Software ihren Anteil. An der Schnittstelle zwischen der Geräte- und Antriebsentwicklung ergeben sich somit Herausforderungen mit Bezug auf technische wie organisatorische Belange.

In diesem Beitrag wird ein Ansatz zum Portfolio- und Komplexitätsmanagement vorgestellt. Dieser beinhaltet zunächst das Verständnis für den Plattformbegriff sowie die Organisation der Plattformentwicklung für Hilti Elektroantriebe. Ein weiterer Aspekt ist der Ansatz zur Schaffung von mehr Transparenz im Geräte- und Antriebsportfolio basierend auf einer strukturierten Visualisierung der Systemarchitektur. Abschließend wird ein Konzept zur Unterstützung bewusster Portfolioentscheide präsentiert, welches eine Klassifikation von Entscheidungssituationen, ein Vorgehensmodell für Entscheidungsprozesse sowie eine Visualisierung von Entscheidungsauswirkungen auf die Portfoliostruktur bzw. Systemarchitektur beinhaltet.

Schlüsselworte

Portfoliomanagement, Modularisierung, Systemarchitektur, Entscheidungsfindung, Plattformen

Portfolio Management for Electric Drives in Powertools at Hilti: Challenges and Solution Approaches

Abstract

The Hilti product portfolio contains a wide spectrum of electric powertools for construction professionals (e.g. rotary drillhammers, screwdrivers, saws, angle grinders and diamond drilling systems). The electric drive is an important subsystem, a decisive driver for product differentiation, but also a source of growing complexity. In the context of an increasing mechatronic character of the powertools, the electronics hardware and embedded software have their particular share in this regard. At the interface between tool development and drive development, this leads to challenges with respect to technical as well as organizational aspects.

In this contribution an approach towards portfolio and complexity management is presented. The approach contains at first a definition of the platform term as well as the organization of platform development for Hilti electric drives. A second aspect deals with creating more transparency in the tools and drives portfolio based on a structured visualization of the system architecture. Finally a concept for supporting conscious portfolio decisions is presented, which includes a classification of decision situations, a procedural model for decision processes as well as a visualization of decision impacts on the portfolio structure respectively the system architecture.

Keywords

Portfolio Management, Modularization, System Architecture, Decision Making, Platforms

1 Ausgangssituation und Motivation

Hilti ist einer der weltweit führenden Bautechnologiekonzerne für Produkte, Systeme und Dienstleistungen für Profianwender. Das Unternehmen ist darauf spezialisiert, seinen Kunden Systemlösungen entlang der gesamten Prozesskette anzubieten. Das **Hilti Produktportfolio** umfasst ein breites Spektrum an Elektrogeräten, u.a. Bohrhämmer, Schrauber, Sägen, Winkelschleifer und Diamantkernbohrsysteme. Die Produktstruktur eines Elektrogeräts besitzt dabei modularen Charakter. Typische Module sind u.a. Werkzeugaufnahme, Schlagwerk, Getriebe, Motor, Elektronik und Handgriff (siehe Bild 1). Der Leistungsumfang des Elektroantriebs reicht dabei „vom Netzstecker bis zum Motorritzel“ und umfasst hiermit den Elektromotor sowie die zugehörige Leistungs- und Steuerungselektronik sowie den Schalter. Bei Netzgeräten wird das Netzkabel mit betrachtet, bei Akkusystemen der Akku und das Ladegerät.

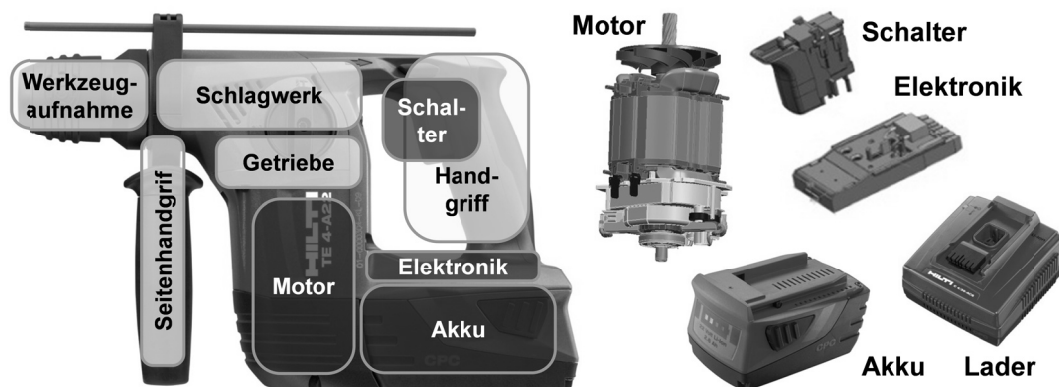


Bild 1: Produktstruktur von Hilti Powertools, Leistungsumfang Elektroantrieb

Die **Aufbauorganisation** besitzt typischen Matrixcharakter (siehe Bild 2). Die Verantwortung für die Produkte liegt in den Business Units, welche jeweils in die Funktionsbereiche Projektleitung, Entwicklung, Marketing und Supply gegliedert sind. Im Technologiebereich ist die Entwicklung für die Elektroantriebe angesiedelt (Electronics & Drives), bestehend aus Fachgruppen für die Themen Motor, Konstruktion, Elektronik Hardware (für Netzgeräte und Akkusysteme), Embedded Software und Messtechnik. Der Kompetenzbereich Elektroantriebe ist BU-übergreifend verantwortlich für die Spezifikation, Entwicklung und Qualifizierung der Antriebssysteme und ist vor allem mit den Business Units Power Tools & Accessories und Diamanttechnik eng verzahnt.

Die **Prozesslandschaft** bei Hilti kann vereinfacht in die Bereiche Technologie, TTM („Time to Money“ = Serienimplementierung) und Produktbetreuung gegliedert werden. Zwischen den einzelnen Bereichen gibt es definierte Übergaben. Beispielsweise ist vor dem Start eines TTM-Projektes der Reifegrad aller relevanten Technologien zu bewerten. Nur reife Technologien werden ins TTM-Projekt übernommen, um zeit- und kostenaufwändige Schleifen während der Serienimplementierung nach Möglichkeit zu vermeiden. Ist das Produkt eingeführt und die Qualität im Markt gewährleistet, erfolgt die Übergabe der Verantwortung an die Produktbetreuung.

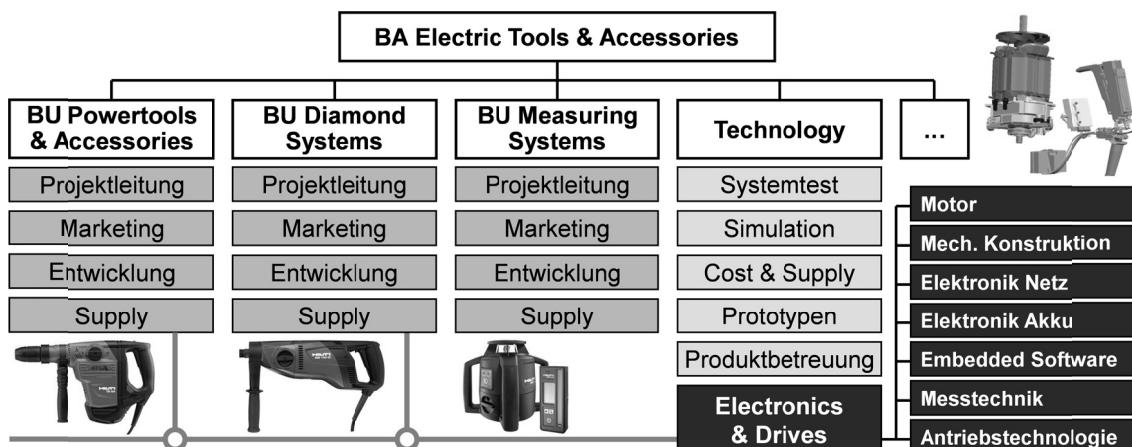


Bild 2: Organisation der Produktentwicklung bei Hilti

Das **Zielsystem** der Produktentwicklung bei Hilti beinhaltet u.a. die Faktoren Differenzierung, Wirtschaftlichkeit und Schlagrate. Eine Differenzierung gegenüber Wettbewerbslösungen bedeutet z.B. hohe Systemperformance und Qualität („Outperform, Outlast“). Bezogen auf den Anspruch nach Profitabilität und Wirtschaftlichkeit gewinnt neben optimierten Herstellkosten zunehmend die Betrachtung weiterer Kostenfaktoren an Bedeutung (z.B. Komplexitätskosten in Logistik, Reparatur und Service etc.). Der Faktor Schlagrate beinhaltet u.a. die Verkürzung der Innovationszyklen (time to market) sowie eine zuverlässige Einhaltung kommunizierter Markteinführungstermine.

Ein wichtiges Subsystem der Geräte und entscheidender Treiber für die Differenzierung ist der **Elektroantrieb**. Aufgrund begrenzter Entwicklungsressourcen werden im Fachbereich Antriebe system- und prozessseitig „Bündelungseffekte“ angestrebt, welche durch eine modulare Produktstruktur und Kommunalitäten mittels Plattform- bzw. Baukastenansätzen realisiert werden. Jedoch generiert die zunehmende technische und organisatorische Komplexität neue Herausforderungen, z.B. die Absicherung der Qualität, die Koordination des Informationsflusses zwischen den beteiligten Disziplinen sowie die Synchronisation der Entwicklungszyklen einzelner Systeme.

2 Stand der Technik und Handlungsbedarf

Das übergeordnete Thema – Portfoliomanagement für Elektroantriebe – wird im Folgenden aus der technischen wie organisatorischen Perspektive betrachtet. Dabei wird jeweils der Stand der Forschung mit dem Stand der Technik bei Hilti abgeglichen, um daraus Handlungsbedarfe und Anforderungen an den Lösungsansatz abzuleiten.

2.1 Perspektive Technik: Systemarchitektur und Plattformen

Unter System kann man im engeren Sinne das Einzelprodukt verstehen (Produktarchitektur siehe [Ulr95]). Im Kontext dieses Beitrags herrscht jedoch ein Verständnis analog zu [KL13], d.h. der Systembegriff bezieht sich auf das gesamte Produktprogramm. Die

Systemarchitektur umfasst die Dimensionen Modularität und Kommunalität. Modularisierung ist eine wichtige und weit verbreitete Maßnahme zur Reduktion der Komplexität in der Produktentwicklung [Göp 98, Ble11]. Modulare Produktstrukturen bzw. Systemarchitekturen ermöglichen u.a. die Parallelisierung von Entwicklungsabläufen und die Verkürzung von Entwicklungszyklen. Von entscheidender Bedeutung ist eine saubere Definition der Modulgrenzen sowie der Schnittstellen. Modulcharakter besitzen im Bereich Antriebe bei Hilti z.B. die Subsysteme Motor, Elektronik und Akku. Schwierigkeiten entstehen u.a. dadurch, dass verschiedene Sichten teilweise zu einer unterschiedlichen Zergliederung des Systems führen (Funktions- vs. Montagemodule). Kommunalitäten ermöglichen „Bündelungseffekte“. In Beschaffung und Produktion können dadurch über Volumensteigerung Herstellkosten reduziert, in der Entwicklung durch eine Mehrfachverwendung von Lösungen Ressourcen effizient genutzt werden. Eine Realisierung derartiger Kommunalitäten ist beispielsweise über Baukasten- und Plattformsätze möglich [ML 97, Grä11, Ble11].

Plattformsätze haben sich bei Hilti vor allem im Bereich Antriebe etabliert. Eine der zentralen Herausforderungen ist hierbei jedoch nach wie vor der Umstand, dass in der Organisation noch kein einheitliches Verständnis zum Plattformbegriff existiert. In der Antriebsentwicklung gab es in den letzten Jahren Bestrebungen, den Begriff zu schärfen. Folgende Definition ist die Grundlage der weiteren Betrachtungen: Plattformen stellen auf Komponentenebene die gemeinsame Basis dar, eine gemeinsame Schnittmenge von Dingen, die mehrfach genutzt werden. Als Komponenten sind dabei Subsysteme des Gesamtproduktes mit mittlerem bis hohem Anteil am Produkt zu verstehen, z.B. Motor, Elektronik, Akku, Lader. Die gemeinsame Basis kann sowohl auf physikalischer Ebene als auch auf konzeptioneller Ebene realisiert werden. Plattformen besitzen fixe und variable Umfänge. Die fixen Umfänge sind über alle Zielprodukte gleich, sie dienen den Synergien. Variable Umfänge sind über mindestens zwei Zielprodukte verschieden, sie dienen der individuellen Anpassung an das jeweilige Zielprodukt.

Den gemeinsamen Nenner und damit das konstituierende Merkmal der Plattform stellt bei den Motoren z.B. der Blechschnitt für Rotor und Stator dar. Kommunalitäten werden oftmals nicht rein auf physikalischer Bauebene, sondern auf konzeptioneller Ebene realisiert, bei Elektroniken beispielsweise auf Ebene des Schaltplans, bei Motoren bezogen auf das Kühlkonzept. Diese Tatsache erschwert das Verständnis für die Zusammenhänge, weil im Vergleich verschiedener Produktvarianten die konzeptionellen Gemeinsamkeiten oft nicht klar sichtbar sind.

Aufgrund der erhöhten Komplexität im Plattformkontext kommt einer übersichtlichen **Visualisierung** eine hohe Bedeutung zu. Zur Visualisierung der Zusammenhänge innerhalb der Produktstruktur eignet sich z.B. der „Module Interface Graph“ [Ble11]. Benötigt wird eine Visualisierung über das Einzelprodukt hinaus, da es bei Änderungen an Plattformteilen (bspw. des Blechschnitts für den Motor) produktlinienübergreifend zu massiven Auswirkungen im gesamten Produktprogramm kommen kann [KL13].

2.2 Perspektive Organisation: Prozesse und Entscheidungen

Seitens der **Organisation** führen Arbeitsteiligkeit und Spezialisierung zu einer Vielzahl am Prozess beteiligter Disziplinen (Entwicklung, Einkauf, Qualität etc.). In Kombination mit der Matrixorganisation resultiert das in einer Vielzahl an Schnittstellen und der Notwendigkeit, die Entwicklungsaktivitäten zu koordinieren. Eine zusätzliche Hürde für den Informationsaustausch stellen im Zuge der Mechatronisierung die unterschiedlichen Begriffswelten in den einzelnen Domänen dar (Mechanik, Elektronik, Software).

In den **Prozessen** führen die hohe Dynamik und Änderungsgeschwindigkeit von Anforderungen einerseits und unterschiedliche Taktzeiten in den Entwicklungszyklen (Gesamtgerät vs. Komponenten vs. Technologien) zu weiteren Herausforderungen. Zudem ist die Vorgehensweise in der Antriebsentwicklung stark „informell erfahrungsbasiert“. Es existieren nur wenige ausgeprägte bzw. explizit gelebte Standards, das Vorgehen in den Projekten ist teilweise sehr projekt- und personenabhängig.

Im Rahmen der Portfoliogestaltung sind **Entscheidungen** mit unterschiedlicher Tragweite zu treffen. Hierbei sind in der Regel viele verschiedene Stakeholder zu involvieren, die unterschiedliche Zielsetzungen verfolgen, was zu einem hohen Koordinationsaufwand führt. In derartigen Entscheidungsprozessen können Methodiken und Metriken unterstützen, die sich explizit auf Portfolioentscheidungen in multivarianten Produktprogrammen beziehen, z.B. [LGS12, KL13, HCS+3]. Unterstützung wird bei Hilti u.a. benötigt für die Entscheidung, welche Plattformtiefe für einen bestimmten Ausschnitt des Portfolios zielführend ist. Bei Elektroniken stellt sich z.B. häufig die Frage, ob eine Plattform auf Ebene Schaltplan oder auf Ebene Aufbaukonzept zielführend ist.

3 Lösungsansatz

Um den beschriebenen Herausforderungen zu begegnen, verfolgt Hilti verschiedene Lösungsansätze. Im Folgenden wird auf die Themen Plattformprozess, Transparenz im Portfolio und bewusste Portfolioentscheidungen eingegangen.

3.1 Plattformprozess

Der Antrieb wurde früher im Rahmen der TTM-Projekte der Zielgeräte mitentwickelt. 2009 wurden jedoch in Bezug auf die Entwicklung von Plattformantrieben zwei wesentliche organisatorische Neuheiten eingeführt: der Plattformprozess und die Rolle des Plattformprojektleiters. Der **Plattformprozess** besitzt ähnliche Phasen wie der TTM-Prozess, bezieht sich jedoch auf die zu entwickelnde Plattformkomponente. Bisher wurden nach diesem Prozess zwei neue Plattformantriebe zur Serienreife entwickelt.

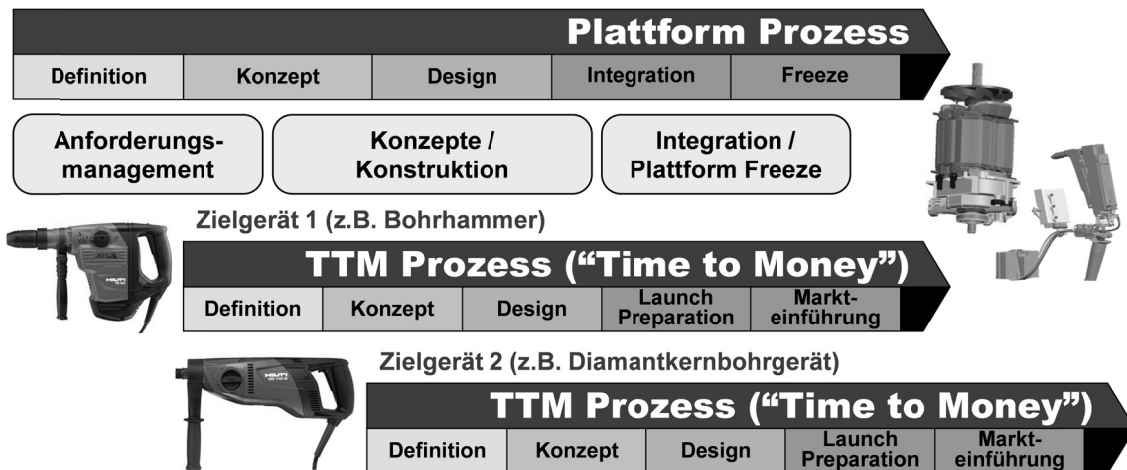


Bild 3: Ansatz zur Organisation der Plattformentwicklung bei Hilti

Der Plattformprozess lässt sich grob in drei Phasen gliedern (siehe Bild 3). Die erste Phase dient dem **Anforderungsmanagement**. Hier gilt es, die Anforderungen möglicher Zielprodukte zu ermitteln und zu analysieren. Auf Basis der Anforderungen werden Cluster von Produkten gebildet, für die ein gemeinsamer Nenner existiert, so dass der Kompromiss, den die Plattform immer darstellt, noch akzeptabel ist. Relevante Kriterien für die Clusterbildung in Bezug auf den Motor sind bspw. die Leistung, der Bauraum, die Robustheit (z. B. gegenüber Staub und Feuchtigkeit) und dynamische Eigenschaften (wie schnell muss der Antrieb abgebremst werden können?).

Sind die wesentlichen Anforderungen geklärt, folgen die Phasen **Konzepte und Konstruktion**. Hier werden die wesentlichen konzeptionellen Merkmale (z.B. Lüftungs-, Lagerungs-, Kommutierungs-, Verschaltungs-, Montagekonzept) sowie die wesentlichen Gestaltparameter (Geometrie des Blechschnitts, Durchmesser von Rotor und Stator, Länge des Blechpakets, Lagerabstände etc.) festgelegt. Hierbei ist eine intensive Abstimmung zwischen dem Plattformprojekt und den Verantwortlichen der einzelnen Zielgeräte notwendig. Hier werden die Weichen für eine möglichst intelligente Variantenbildung gestellt sowie fixe und variable Umfänge definiert.

Zum Abschluss der Plattformentwicklung erfolgen die **Integration in die Zielgeräte** und der **Plattformfreeze**. Mit dem „Einfrieren“ von Plattformkomponenten ist ein grundsätzlicher Änderungsstopp gemeint. Änderungen sind ab diesem Zeitpunkt in der Regel nur noch mit größerem Aufwand durchführbar. Kommt es dennoch zu Änderungsanträgen, muss zwischen verschiedenen Optionen bewusst abgewogen werden. Entweder die Änderung wird abgewiesen, d.h. es muss mit den Kompromissen gelebt oder das Problem an anderer Stelle gelöst werden. Eine weitere Option ist die gerätespezifische Änderung. Damit werden Auswirkungen auf bestehende Geräte im Markt vermieden, jedoch besteht die Gefahr, dass Plattformsynergien verloren gehen. Als letzte Option kommt die Änderung der Plattform in Frage, dies betrifft jedoch rückwirkend auch bereits bestehende Geräte, hier ist ggf. eine Nachqualifizierung erforderlich.

Die Plattformentwicklung führt zu einer wachsenden technischen und organisatorischen Komplexität. Es entsteht eine „Dreiecksbeziehung“ aus Plattformprojekt und den TTM-Projekten von zwei (oder mehr) Zielgeräten. Hier kommt der **Projektorganisation** eine wichtige Rolle zu, um den notwendigen Informationsfluss zu gewährleisten, aber dennoch effizient zu bleiben. Gewisse Projektrollen nehmen hier eine „Klammerfunktion“ zwischen den beteiligten Projekten ein, um den erhöhten Koordinationsbedarf zu decken. Diese Rolle wird u.a. vom Projektleiter Plattformen wahrgenommen und vom Projektverantwortlichen Antriebsentwickler (PVA).

3.2 Transparenz im Portfolio

Aufbauend auf dem grundlegenden Verständnis des Plattformbegriffs im Bereich Antriebe wurde in den letzten drei Jahren ein Ansatz in Richtung mehr **Transparenz im Portfolio** verfolgt (siehe [Loc13, Dei13]). Dieser umfasst ein Modell zur Darstellung der Systemarchitektur, einen Leitfaden zur Analyse der Systemarchitektur sowie eine grafische Visualisierung relevanter Ausschnitte aus dem Portfolio („Tapete“). **Systemarchitektur** umfasst in diesem Verständnis eine vertikale Komponente (Produktstruktur), in der die Modularität sichtbar wird, sowie eine horizontale Komponente (Programmstruktur), in welcher Kommunalität bzw. Variantenvielfalt sichtbar werden. Das Modell zur Darstellung der Systemarchitektur stellt beide Dimensionen in einer Matrix gegenüber und ermöglicht strukturierte Blicke auf Variantenspektren (siehe Bild 4).

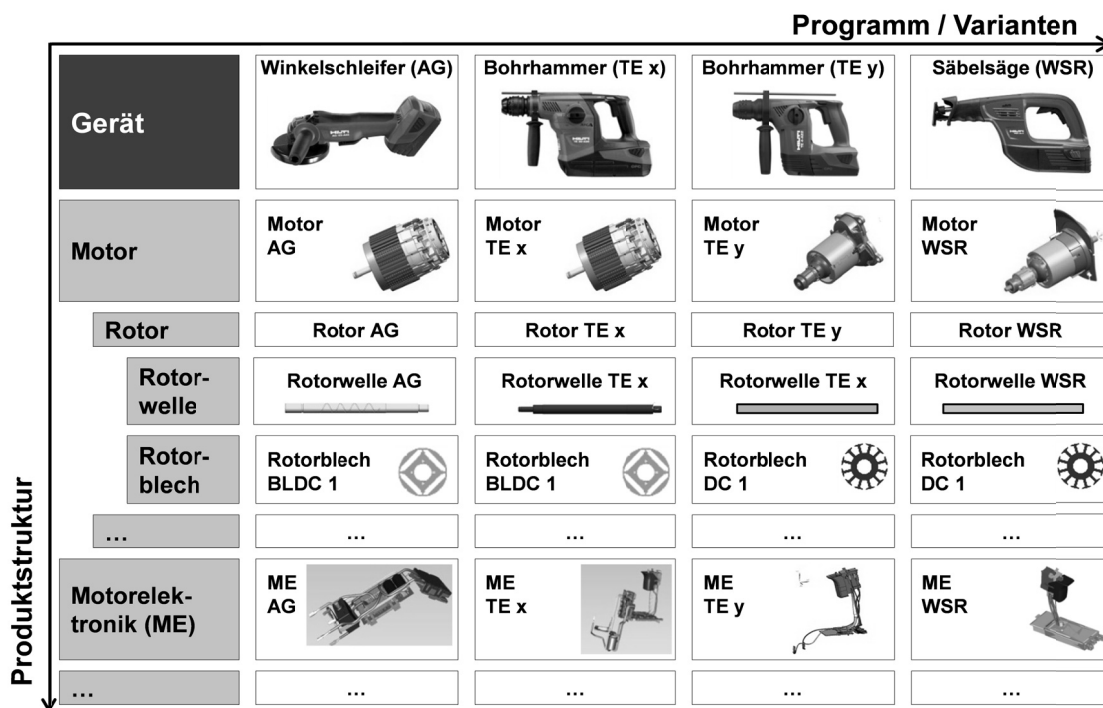


Bild 4: Modell zur Darstellung der Systemarchitektur

Der **Leitfaden zur Analyse der Systemarchitektur** zielt auf die Erzeugung von Transparenz in bestehenden Variantenspektren ab und beinhaltet vier Schritte [Dei13], siehe

Bild 5. Im **ersten Schritt** erfolgt eine Definition des Betrachtungsumfangs in Abhängigkeit von Projektsituation und Fragestellung. Zum einen sind hier der relevante Ausschnitt aus dem Produktprogramm zu wählen, zum anderen die relevanten Komponenten. Beispiel: „Welche Motoren haben wir im Portfolio, die wir selbst entwickeln und produzieren und die in Akkugeräten eingesetzt werden?“

Im **zweiten Schritt** erfolgt eine Analyse der Vielfalt auf Komponentenebene. Beispiel: „Wie viele verschiedene Rotorbleche, Rotorwellen etc. gibt es?“ Dadurch werden innerhalb der betrachteten Baugruppen die physikalischen Umfänge (Bauteile) sichtbar, die für den fokussierten Ausschnitt aus dem Produktprogramm gleich und unterschiedlich sind. Allerdings zeigt sich hier, dass auch trotz Plattformansatz die Vielfalt auf Komponentenebene in der Regel noch sehr hoch ist. Im Extremfall kann dies bedeuten, dass verschiedene Varianten innerhalb einer Motorenplattform für unterschiedliche Zielapplikationen als Gleichteile lediglich das Rotor- und Statorblech, sowie einige Standardkomponenten (Kugellager, Schrauben etc.) besitzen.

Daher erfolgt im **dritten Schritt** eine weitergehende Analyse hinsichtlich der konzept- bzw. gestaltrelevanten Beschreibungsmerkmale, die zur Unterscheidung bzw. Klassifikation der einzelnen Komponentenvarianten für den betrachteten Ausschnitt des Produktportfolios herangezogen werden können. Ergebnis ist die Identifikation relevanter Beschreibungsmerkmale der betrachteten Komponente, wie Motortyp, Motor Blechschnitt, Blechpaketlänge, Wicklungstyp etc. Im **vierten Schritt** erfolgt schließlich eine Analyse der Vielfalt auf Konzept- bzw. Merkmalsebene. Hierdurch werden die Logik der Variantenentstehung und die Zugehörigkeit verschiedener Motoren zu einer Plattform besser sichtbar. Außerdem können auf diesem Wege die fixen und variablen Merkmale innerhalb der Plattform unterschieden werden.

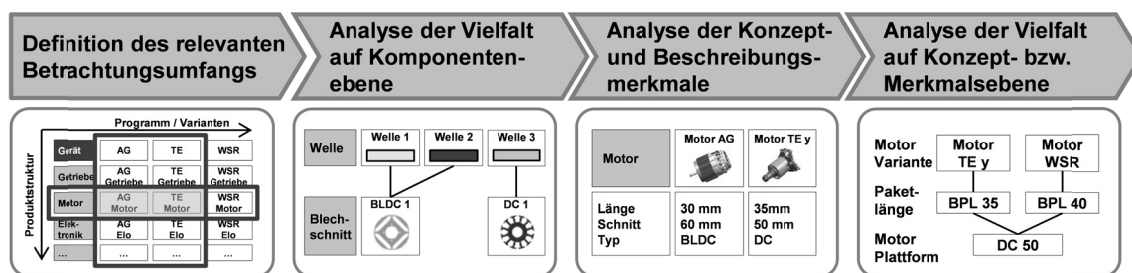


Bild 5: Leitfaden zur Analyse der Systemarchitektur

Das Ergebnis der Analyse lässt sich in eine **grafische Visualisierung** relevanter Ausschnitte aus dem Portfolio überführen, auch als „Tapete“ bezeichnet [Loc13]. Bild 6 enthält beispielhaft eine „Motortapete“. Im unteren Teil der Grafik ist die Logik der Variantenentstehung bei Motoren ersichtlich, die sich aus der Reihenfolge folgender Merkmale ergibt: 1) Motortyp, 2) Blechschnitt und 3) Blechpaketlänge. Eine Motorenplattform beinhaltet alle Motoren innerhalb eines Motortyps mit demselben Blechschnitt (z.B. UNI 1, UNI 2). Innerhalb der Plattform existieren in der Regel unterschiedliche Blechpaketlängen, vergleichbar mit Baureihen, die eine Leistungsstufung ermöglichen.

Im oberen Teil der „Tapete“ sind nun alle Geräte angeordnet, die einen entsprechenden Motor beinhalten. Der konkrete Motor (z.B. von Akku-Bohrhammer 1 oder Akku-Schrauber 1) enthält darüber hinaus immer noch variable Elemente (z.B. Wicklung = Drahtdurchmesser und Anzahl der Windungen), die eine Anpassung des Motors an die konkreten Leistungsbedarfe des Geräts ermöglichen.



Bild 6: Transparente Visualisierung von Portfoliozusammenhängen („Tapete“)

3.3 Bewusste Portfolioentscheidungen

Die Kenntnis über die Zusammenhänge im bestehenden Portfolio kann wiederum in die Gestaltung bzw. die gezielte Weiterentwicklung des Portfolios mittels **bewusster Portfolioentscheidungen** einfließen. Hierzu wurde eine Klassifikation von Entscheidungssituationen sowie ein Leitfaden zur Entscheidungsfindung erarbeitet, der ein Vorgehensmodell und Kriteriensystem zur Bewertung alternativer Szenarien beinhaltet [Loc13].

Sowohl auf strategischer als auch operativer Ebene werden ständig Entscheidungen mit mehr oder weniger großen Auswirkungen auf das Geräte- und Antriebsportfolio getroffen. Die **Klassifikation von Entscheidungssituationen** (siehe Bild 7) dient dazu, die Bedeutung einzelner Entscheidungen für das Portfolio transparent zu machen, und die Entscheidungen entlang der Prozesskette zu verorten. Grundsätzlich kann in Auswahlentscheidungen (Welche Technologie bzw. welches Konzept soll weiterverfolgt werden? Welcher Lieferant erhält den Zuschlag?) und Freigabeentscheidungen (Freigabe Technologiereife? Freigabe Werkzeuge? Freigabe Nullserie?) unterschieden werden.

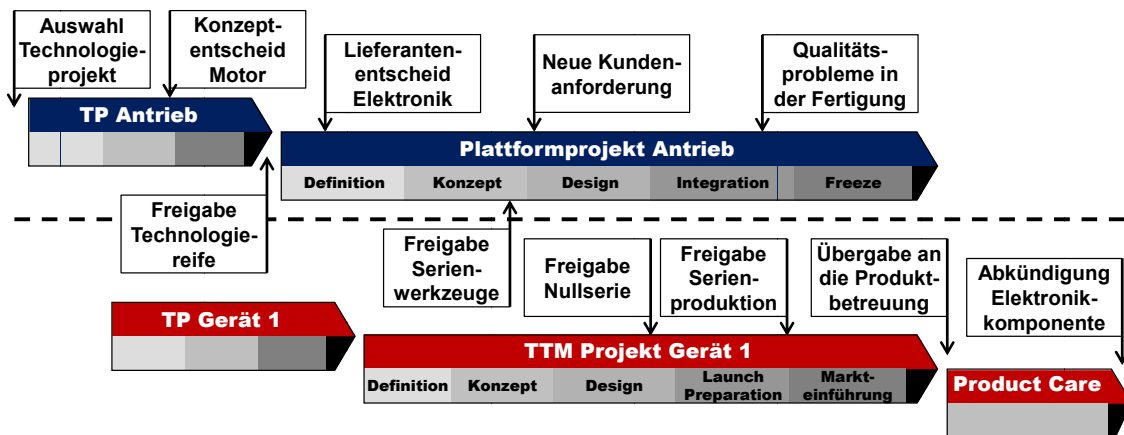


Bild 7: Klassifikation von Entscheidungssituationen

Der **Leitfaden zur Entscheidungsfindung** enthält eine grundlegende Vorgehensweise, die für alle Entscheidungsprozesse anwendbar ist. Die Schritte sind: 1) Initiierung, 2) Planung und Analyse, 3) Schaffung von Optionen, 4) Bewertung, 5) Entscheidung, 6) Umsetzung und Kontrolle. Je nach Entscheidungssituation und -typ sind konkrete Ausprägungen dieser Vorgehensweise denkbar. Schließlich können im Modell der Systemarchitektur (siehe Bild 4) Entscheidungen sichtbar gemacht werden, die Einfluss auf die Gestaltung des Portfolios nehmen. So können in den einzelnen Spalten Optionen bzgl. des **Modularisierungsgrads** visualisiert werden: Sollten zwei Funktionseinheiten einer Elektronik integriert oder separat realisiert werden (z.B. Leistungseinheit und Steuereinheit)? Sollte der Schalter fix oder trennbar mit dem Elektrobausatz verbunden werden? Eng verknüpft mit der Frage nach dem optimalen Modularisierungsgrad und der Gestaltung der Schnittstellen ist die Frage nach dem optimalen Grad an **Kommunalität** (Was sollte vereinheitlicht werden? Was sollte unterschiedlich ausgeprägt werden?). Eine modulare Gestaltung des Elektrobausatzes ergibt z.B. vor allem dann Sinn, wenn über die geräteübergreifende Vereinheitlichung der Steuereinheit (= Plattform!) Synergien im Entwicklungsprozess bzw. in der Beschaffung zu realisieren sind.

4 Resümee und Ausblick

Kern der vorgestellten Methodik ist ein Verständnis von Plattform als Subsystem innerhalb des Gesamtproduktes, welches fixe und variable Umfänge besitzt. Die Systemarchitektur im Produktportfolio lässt sich strukturiert in einem zweidimensionalen Schema visualisieren, in welchem die Produktstruktur in der Vertikalen und die Programmstruktur / Variantenvielfalt in der Horizontalen dargestellt sind. Einzelne Ausschnitte aus dem Portfolio lassen sich hinsichtlich der zugrunde liegenden Variantenbildungslogik analysieren, das Ergebnis ist in Form einer „Tapete“ darstellbar. Die „Tapete“ schafft Transparenz in Bezug auf bestehende Portfoliozusammenhänge und ermöglicht eine gezielte Weiterentwicklung des Portfolios. Durch die Nutzung dieser Visualisierung wird der Informationsaustausch gefördert und die Effizienz im Prozess, insbesondere an der Schnittstelle zwischen Antriebs- und Geräteentwicklung verbessert.

Die Methodik befindet sich seit etwa einem Jahr bei Hilti im Einsatz, die Resonanz und Wirkung ist bisher durchweg positiv. Darüber hinaus wird die Methodik stetig erweitert. Aktuelle Schwerpunkte sind Vorgehensweisen und Hilfsmittel zur besseren Berücksichtigung monetärer Aspekte bei Portfolioentscheidungen (Stichwort Komplexitätskosten) sowie zur automatisierten Ableitung geeigneter Visualisierungsformen in Abhängigkeit der Fragestellung und Projektsituation (Stichwort „Tapete auf Knopfdruck“).

Literatur

- [Ble11] BLEES, C.: Eine Methode zur Entwicklung modularer Produktfamilien. Dissertation TU Hamburg-Harburg. TuTech Verlag: Hamburg, 2011.
- [Dei13] DEIMLING, C.: Konzept zur Analyse und Gestaltung von Systemarchitekturen multivarianter Produktprogramme. TU München, 2013.
- [Göp98] GÖPFERT, J.: Modulare Produktentwicklung – Zur gemeinsamen Gestaltung von Technik und Organisation. Deutscher Universitäts-Verlag, Wiesbaden, 1998.
- [Grä11] GRÄBLER, I.: Balancing internal and external product variety in product development. 18th International Conference on Engineering Design (ICED11), Kopenhagen (2011).
- [HCS+13] HEILEMANN, M; CULLEY, S.; SCHLÜTER, M.; HAASE, H.-J.: Examination of Modularization Metrics in Industry. 19th International Conference on Engineering Design (ICED13), Seoul (2013).
- [KL13] KISSEL, M.; LINDEMANN, U.: System architecture change decisions in multi-variant product portfolios. 19th International Conference on Engineering Design (ICED13), Seoul (2013).
- [LGS12] LAUX, H.; GILLENKIRCH, R. M.; SCHENK-MATHES, H.Y.: Entscheidungstheorie. Berlin: Springer 2012.
- [Loc13] LOCK, C.: Konzept zur Unterstützung von Entscheidungen mit Einfluss auf die Systemarchitektur. TU München, 2013.
- [ML97] MEYER, M.H.; LEHNERD, A.P.: The Power of Product Platforms: Building Value and Cost Leadership. The Free Press: New York, 1997.
- [Ulr95] ULRICH, K.: The Role of Product Architecture in the Manufacturing Firm. Research Policy, Vol. 24, No. 3, 1995, S. 419-440.

Autor

Dr.-Ing. Josef Ponn ist seit 2011 als Projektleiter für Plattformen im Bereich Electronics & Drives der Hilti Entwicklungsgesellschaft in Kaufering tätig. Zu seinen Aufgaben zählt unter anderem die Serienimplementierung von Plattformantrieben. Nach seinem Maschinenbau-Studium an der TU München promovierte Herr Ponn am Lehrstuhl für Produktentwicklung der TU München zum Thema „Situative Unterstützung der methodischen Konzeptentwicklung technischer Produkte“. Die operative Anwendung der Entwicklungsmethodik praktizierte Herr Ponn von 2007 bis 2011 bei Hilti im Bereich Prozesse und Methoden. Hier leitete er neben methodenbasierten Workshops (u.a. FMEA, kreative Lösungsfindung, Variantenentscheide) eine Reihe strategischer Projekte, beispielsweise zu den Themen „Antriebfertigung“ und „Lean Development“.

Intelligente Elektronikentwicklung

Integrativer Ansatz zur rechnergestützten Entwicklung räumlicher optomechatronischer Baugruppen

Jochen Zeitler, Christian Fischer, Jörg Franke

Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik

Friedrich-Alexander-Universität Erlangen-Nürnberg

Egerlandstraße 7-9, 91058 Erlangen

Tel. +49 (0) 9131 / 85 28 972, Fax. +49 (0) 9131 / 30 25 28

E-Mail: Jochen.Zeitler@faps.fau.de

Zusammenfassung

Ein Trend in der Mechatronik ist die stetige Nachfrage nach Miniaturisierung zur Erstellung funktionsintegrierter, bauraumoptimierter Baugruppen. Es werden kleinere Bauteile mit immer feiner werdenden Strukturen verwendet, um mehr Funktionalität in die entstehenden Produkte zu bringen. Insbesondere durch die funktionale Integration von Mechanik und Elektronik entstehen neue mechatronische Baugruppen, sogenannte Molded Interconnect Devices (MID). Diese 3D-Schaltungsträger können die Anforderungen zunehmender Miniaturisierung lösen. Eine Herausforderung im Prozess der Konstruktion ist die Integration von Elektronik und Mechanik auf Baugruppen mit komplexer Oberflächenstruktur. Zukünftig wird die elektrische Datenübertragung auf Baugruppenebene zunehmend auch durch optische Kommunikationstechnologien ergänzt werden. Insbesondere in der rechnergestützten Entwicklung stellen sich hier neue Herausforderungen. Derzeit existieren keine Designwerkzeuge, welche diese spezifischen Anforderungen an 3D-optomechatronische Baugruppen abdecken. Folgend werden bestehende Methoden zum rechnergestützten Entwurf für 3D-MID vorgestellt. Aus diesen werden Herausforderungen im Bereich optomechatronischer Baugruppen und neue Lösungsansätze aufgezeigt.

Schlüsselworte

3D-MID, ECAD-MCAD, Optomechatronik

Integrated Design Methods for Computer-Aided Development of Spatial Opto-Mechatronic Assemblies

Abstract

One trend in mechatronics is the increasing demand for miniaturization for the creation of functionally-integrated, space-saving assemblies. Components are getting increasingly smaller and use finer structures to reduce the size of products. Within the functional integration of mechanics and electronics new mechatronic assemblies evolved, so called molded interconnect devices (MID). These assemblies connect electronic and mechanical functionality within the same part. On component level the electrical data transfer will be complemented by optical communication technologies in future. Regarding to this development novel hybrid mechatronic parts will occur with integrated electronic, as well as integrated optical functions. Especially in the field of computer-aided development completely new challenges have to be solved. Though currently in this field no design-tools are available which could cover this specific field of requirements. In following existing methods for computer-aided design of spatial mechatronic assemblies will be presented. Also requirements and processes for a new optomechatronic design-tool will be unveiled.

Keywords

3D-MID; ECAD-MCAD, Optomechatronics

1 Einführung

Durch die mögliche Verbindung einer Vielzahl technischer, wie z.B. mechanischer, elektrisch/elektronischer, optischer, thermischer oder chemischer Funktionen ist die Mechatronik fundamentale Triebfeder für Innovationen. Allerdings steigen durch diese Innovationen auch stetig die Anforderungen an die verwendeten Bauteile. Produzierende Unternehmen sind bei der Entwicklung und Herstellung mechatronischer Produkte wachsenden Herausforderungen hinsichtlich Miniaturisierung, Funktionsintegration und Zuverlässigkeit ausgesetzt. Eine Klasse mechatronischer Produkte, die den Gedanken der Funktionsintegration und Miniaturisierung in den Mittelpunkt stellt, sind räumliche elektronische Baugruppen. Molded Interconnect Devices oder Mechatronic Integrated Devices (3D-MID) bieten räumliche Gestaltungsfreiheit und integrieren mechanische und elektronische Funktionalität, indem diese mit Leiterbildstrukturen und elektronischen Bauteilen versehen werden [Fra13]. Eine weitere Anwendungsmöglichkeit, für die in Zukunft ein hohes Potential erwartet wird, liegt in der Integration von optischen Kommunikationsstrukturen in elektronische und mechatronische Bauteile und -gruppen. Diese integrierten optischen Baugruppen haben vor allem Vorteile aufgrund der elektromagnetischen Verträglichkeit sowie der Fähigkeit hohe Datenraten zu übertragen. Im Folgenden sollen zunächst aktuelle Entwicklungsmethoden und Forschungsprojekte zu 3D-MID vorgestellt werden. Anschließend werden diese auf das Feld der integrierten optomechatronischen Baugruppen erweitert.

2 Rechnergestützte Entwicklung von 3D-MID

Für die Herstellung der thermoplastischen Grundkörper im Spritzgussverfahren sowie für die elektronische Aufbau- und Verbindungstechnik von 3D-MID sind serienreife Lösungen vorhanden. Geeignete Softwarewerkzeuge für die MID-Entwicklung sind jedoch noch immer wenig verbreitet [Fra13]. Dies ist dem Umstand geschuldet, dass Unternehmen im Bereich 3D-MID meist auf ihre schon etablierten Systeme setzen. Zudem rechtfertigt erst eine gewisse Anzahl von Entwicklungsprojekten die Einführung von zusätzlicher Software, da dies mit einem hohen Zeit- und Kostenaufwand verbunden ist. Die Beherrschung kurzer Innovationszyklen mechatronischer Produkte ist, aufgrund hoher Komplexität, ohne den Einsatz rechnergestützter Werkzeuge kaum zu realisieren. Insbesondere lässt sich dies auf Produkte der MID-Technologie übertragen. MIDs erfordern eine logisch-elektrische als auch eine räumlich-mechanisch formgebende Konstruktion. Meist werden in der Praxis, nach heutigem Stand der Technik, die Domänen Elektrotechnik und mechanische Konstruktion getrennt behandelt. Der Austausch der benötigten Daten erfolgt zumeist aufwändig über unterschiedlichste Datenformate [KF05].

Eine Besonderheit von elektronischen CAD-Werkzeugen für das Schaltungslayout (E-CAD) liegt darin, dass im Unterschied zur mechanischen Konstruktion, zunächst der

logische Entwurf in einem eigenen CAE-Modul entwickelt wird. Erst nach dem Entwurf des fertigen Logikmodells wird die Geometrie der Leiterbahnen und Pad-Flächen bestimmt. Im CAD-System für die mechanische Konstruktion (MCAD) wird hingegen die räumliche Formgebung des Schaltungsträgers festgelegt. Zudem können hier Flächen definiert werden, welche für die Platzierung elektronischer Bauteile oder zur Leiterbahnenflechtung nicht verwendet werden dürfen. Zunächst entsteht der Entwurf zur räumlichen Gestalt, der anschließend durch Simulation des Schaltungsträgers validiert werden kann. Aus dieser verteilten Vorgehensweise zur Produktentwicklung ergeben sich jedoch unerwünschte Iterationen. Diese resultieren aus der Trennung der beiden Ingenieursdisziplinen Maschinenbau und Elektrotechnik. So fehlen der mechanischen Seite Daten zu elektrischen Eigenschaften sowie Informationen über Verbindungen der Bauelemente untereinander. Der Elektronikentwickler kennt im Gegenzug weder Gehäuse noch Bauraum und kann daher keine Rücksicht darauf nehmen [Zhu07]. Während die klassische parallele Entwicklung bei Printed Circuit Boards- (PCB-) Baugruppen durchaus Sinn macht, ist im Falle von räumlichen Schaltungsträgern eine integrierte Entwicklungsweise notwendig.

2.1 Integration domänenspezifischer Funktionen

Zur Kopplung der mechanischen Konstruktion und des elektrischen bzw. elektronischen Layouts gibt es bereits kommerzielle Lösungen. Das gleichzeitige Bearbeiten von Mechanik und Elektronik unter Berücksichtigung der Abhängigkeiten, als ECAD/MCAD Co-Design bezeichnet, wird beispielsweise mit dem von der Firma Zuken entwickelten BoardInterchanger realisiert, der einen Austausch von Leiterplattengeometrien mit dem MCAD-System Catia erlaubt [Zuk14-ol]. In Pro/ENGINEER ist das Modul ECAD-MCAD Collaboration Extension (ECX) für die datentechnische Verbindung zu Mentor Graphics und Cadence eingebettet. Damit lassen sich Änderungen auf Mechanik- und Elektronikseite gleichzeitig anzeigen [Uhl08]. Die Lösungen der großen Softwareanbieter vereinfachen zwar die Zusammenarbeit zwischen Konstrukteuren und Layoutern, bieten allerdings keine Unterstützung für räumliche Schaltungsträger an [FFF09]. Deshalb ist es notwendig, einheitliche Funktionen und eine gemeinsame Entwicklungsbasis für die grundverschiedenen Domänen Maschinenbau und Elektrotechnik zu schaffen (siehe Bild 1).

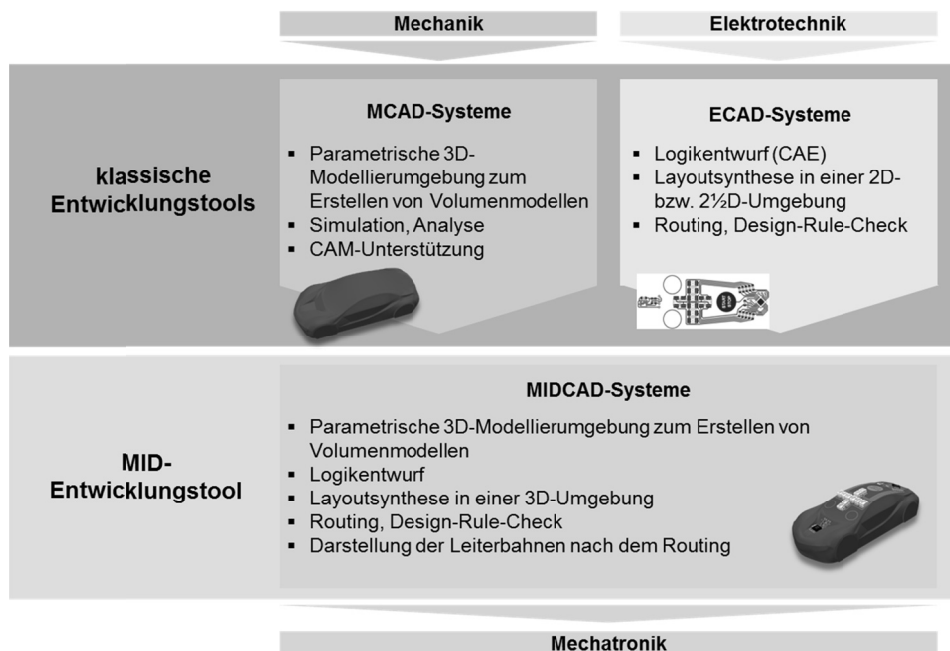


Bild 1: Synthese der Konstruktionsdisziplinen

Dies betrifft standardisierte MCAD-Funktionalitäten, wie die parametrische 3D-Modellierung oder das automatisierte Entflechten der Leiterbahnen aus dem ECAD-Bereich. Dazu kommen Design-Regeln, die für 3D-MID von Bedeutung sind. Dies können Kantenradien oder Montageflächen sein, die in jedem Fall berücksichtigt werden müssen. Aus dieser Synthese der beiden Domänen entsteht ein integriertes Softwaremodell zur Entwicklung räumlicher mechatronischer Baugruppen.

2.2 Integrative Entwicklung durch Kopplung der Partialmodelle

Am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik (FAPS) wurden in diesem Bereich zwei Ansätze verfolgt: Zum einen das in Pro/ENGINEER integrierte System MIDCAD und zum anderen das Projekt MID-Layout [ZFG+14]. In beiden Systemen wurden automatisierte Routingverfahren auf räumlichen Schaltungsträgern prototypisch umgesetzt. Als Grundlage dienten jeweils proprietäre CAD-Systeme wie Siemens NX, Pro/ENGINEER oder Autodesk Inventor. In MIDCAD wurde zunächst die Umsetzbarkeit eines 3D-ECAD-Systems für MID-Baugruppen auf Basis herkömmlicher MCAD-Werkzeuge untersucht. Dabei wurden neue MID-spezifische Funktionsmodule wie das automatische Routing oder die Integration von CAE-/CAM-Funktionen berücksichtigt. Die Funktionen wurden direkt über die Programmierschnittstellen im CAD-System integriert [Zhu07]. Im Projekt MID-Layout wurde der Rechenaufwand im CAD-System minimiert und ausgelagert. Komplexe Berechnungen zum Leiterbahnrouting sind in einem eigens dafür entwickelten Modul realisiert. Die Kopplung zwischen den CAD-Systemen und dem Berechnungsmodul basiert auf einem MID-spezifischen Produktmodell, welches auf der Auszeichnungssprache XML basiert (siehe Bild 2). Hier befinden sich z.B. die nötigen Daten aus dem logischen Verhal-

tensmodell der elektronischen Schaltung, zu der auch technologische Design-Regeln gehören. Im mechanischen Bereich existieren geometrisch-mathematische Beschreibungen der Schaltungsträger. Hier sind auch 3D-Bauteilpositionen oder sogenannte Keep-Out-Flächen, Flächen die zumeist mechanische Funktionen erfüllen und nicht für die Elektronik verwendet werden dürfen, zu finden. Die eigentliche Berechnung der Leiterbahntflechtung geschieht im Projekt MID-Layout über drei wesentliche Prozeduren. Aus der in das XML-Format extrahierten Volumengeometrie der CAD-Systeme wird ein neues Datenmodell erstellt. Dieses enthält die geometrisch-mathematische Beschreibung der Volumenkörper, die sogenannte Boundary Representation (BRep). Die Berechnung einer in die Ebene projizierten Darstellung der Oberflächengeometrie wird hierdurch ermöglicht. Eine Herausforderung dabei ist die Abwicklung unebener Geometrien, wie Flächen mit elliptischer Grundform oder Freiformflächen. Insbesondere letztere sind eine große Herausforderung, da sich diese nur näherungsweise berechnen lassen. Aus den in die Ebene projizierten Flächen und den vordefinierten MID-spezifischen Informationen, wie Design Regeln und Bauteilpositionen, entsteht ein neues planarisiertes Produktmodell. In diesem Modell sind Routing-Algorithmen implementiert und berechnen Pfade für Leiterbahnen. Die Ergebnisse werden in die 3D-Umgebung rücktransformiert und über die XML-Schicht zurück an die 3D-MCAD-Systeme geleitet.

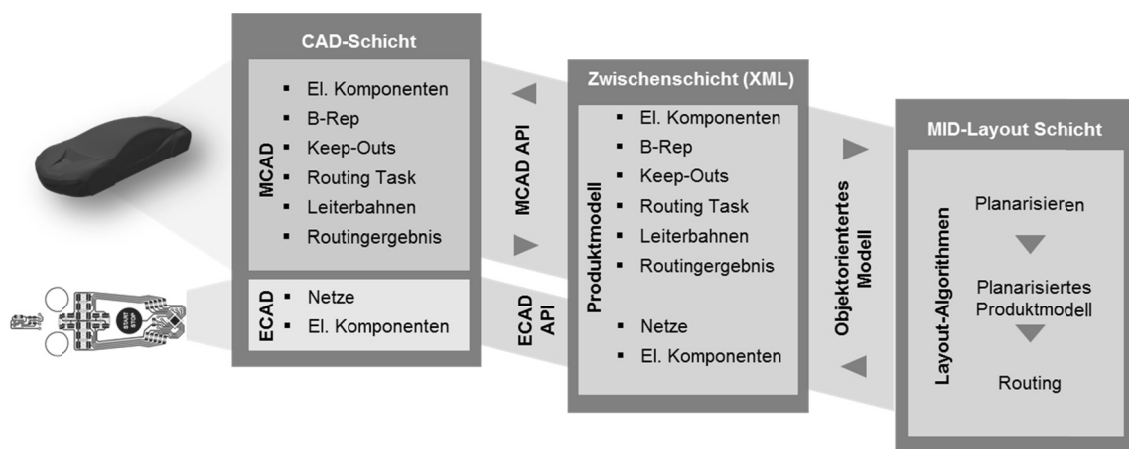


Bild 2: Datenaustausch zwischen CAD-Systemen, Produktmodell und Layout-Algorithmen

Der Vorteil dieser Vorgehensweise ist vor allem die Möglichkeit unterschiedlichste CAD-Systeme anzubinden. Ebenso bietet sie modulare Erweiterbarkeit über das XML-basierte MID-Produktmodell unter Beibehaltung der systemeigenen CAD-Formate. Dadurch können Modellhistorien der nativen CAD-Formate erhalten werden. Diese würden bei der Bearbeitung durch neutrale CAD-Formate wie STEP oder IGES verloren gehen. Insbesondere letzterer Punkt ist bedeutsam, wenn Änderungen auf parametrischer Basis vorgenommen werden sollen.

Ein kommerziell verfügbares Tool für die 3D-MID-Konstruktion ist mit dem industriell bereits eingesetzten 3D-ECAD-System NEXTRA® der Firma MECADTRON GmbH gegeben [KF05]. NEXTRA® beinhaltet grundlegende Funktionen wie die 3D-Platzierung elektronischer Bauelemente auf räumlichen Schaltungsträgern, das manuelle Entflechten der elektrischen Leiterbahnen sowie die Anbindung von MCAD- und ECAD-Tools über Dateischnittstellen. NEXTRA® bietet ähnliche Funktionen für die MID-Konstruktion wie MIDCAD und MID-Layout. Keines der genannten Werkzeuge bietet jedoch spezifische Konstruktionsroutinen für die Optoelektronik.

3 Entwicklung integrierter optomechatronischer Schaltungsträger

Die optische Datenübertragung ist auf Grund ihrer vielen Vorteile bereits im Kommunikationsnetz erfolgreich etabliert. Elektromagnetische Unempfindlichkeit und Steigerung der Datenrate durch Wellenlängenteilung sind nur einige vielversprechende Vorteile dieser Technologie [Sch08]. Auch der Chip-zu-Chip Kommunikation wird aufgrund der Möglichkeit, große Datenvolumina zu verarbeiten, erhebliches Potential zugeschrieben.

Im Allgemeinen wird bei der optischen Datenübertragung ein optisches Signal mittels einer Laserdiode oder einem VCSEL (Vertical Cavity Surface Emitting Laser) in einen Lichtwellenleiter eingekoppelt und über diesen zum Empfänger geleitet. Bei elektrooptischen Leiterplatten kommt zu den verschiedenen Kupferlagen der herkömmlichen Leiterplatten eine zusätzlich integrierte optische Lage. Diese Lage kann einen oder mehrere Lichtwellenleiter enthalten. Die einzelnen Lichtwellenleiter sind von der Oberfläche der Leiterplatte optisch zugänglich und ermöglichen das Ein- und Auskoppeln von Lichtsignalen. [FC09], [FCR08]

In Zukunft wird sich das Potential auch auf den Bereich integrierter Schaltungsträger ausweiten. Insbesondere der geringe Platzbedarf und die hohe Beständigkeit gegen Umwelteinflüsse im Vergleich zu herkömmlichen Leitern sind ausschlaggebend für deren Einsatz. Folgend wird der gegenwärtige Stand der Technik zum rechnergestützten Entwurf optoelektronischer Designwerkzeuge dargestellt. Aus diesen werden Anforderungen und Handlungsstrategien für die rechnergestützte optomechatronische Entwicklung abgeleitet.

3.1 Gegenwärtiger Stand und Anforderungen im Bereich optomechatronischer Designwerkzeuge

Das Prinzip der Verknüpfung mehrerer Disziplinen ist bei MID-Produkten durch die untrennbare Verbindung von elektronischer Funktion und Bauteilgeometrie noch deutlicher ausgeprägt als bei herkömmlichen mechatronischen Systemen. Aus diesem Grund ist ein Werkzeug notwendig, welches in der Konstruktion und der Herstellung den besonderen Anforderungen des elektronisch-mechanischen Aufbaus von MID-

Komponenten Rechnung trägt [Mei02]. Die Anforderungen lassen sich dabei nach solchen aus der geometrischen Gestaltungsfreiheit, der Integration elektronischer und mechanischer Funktionen sowie nach design- und fertigungsbezogenen Anforderungen klassifizieren. Zu diesen Anforderungen kommt nun die Domäne der Optik oder Optoelektronik hinzu.

Im 2D-Bereich wird das Design optoelektronischer PCBs bereits durch spezielle CAD-Tools unterstützt. So wurde vom Cooperative Computing & Communication Laboratory (C-LAB) in Paderborn der OptoBoard Designer für Flachbaugruppen entwickelt [BSS09]. Diese Software stellt eine Simulations- und Entwurfsumgebung für leiterplattenbasierte Multimode-Wellenleiter-Verbindungen.

Die Eigenschaften des Materials, der Signalerzeugung und -leitung sowie die teilweise aktiv überwachten Montageverfahren unterscheiden sich zu dem Erfahrungsbereich der Entwicklungsmethodik für räumliche elektronische Baugruppen. Konstruktionsfunktionen, die Koppelstellen zu elektrisch-optischen Funktionen liefern könnten, sind nicht vorhanden. Es existieren derzeit auch noch keine Design-Richtlinien. Vorgaben aus dem MID-Bereich sind deshalb nicht einfach auf räumliche optomechatronische Baugruppen übertragbar.

Bisher ist keine Design-Software für die Konstruktion räumlich angeordneter optomechatronischer Baugruppen verfügbar. Die am Lehrstuhl FAPS entwickelten Systeme MIDCAD und MID-Layout können jedoch als Basis für das Design optomechatronischer Baugruppen dienen (siehe Bsp. Bild 3). In zweidimensionalen elektronischen CAD-Systemen (2D-ECAD) spielt die Bereitstellung von Bauelementbibliotheken ebenso eine wichtige Rolle wie die Verwendung von Features in 3D-MCAD-Systemen. Bereits bestehende Funktionen wie die Bauelementplatzierung auf beliebigen Flächen sowie die 3D-Leiterbahnentflechtung auf der Oberfläche des Schaltungsträgers sind bereits in den Projekten MID-Layout und MIDCAD realisiert [ZFG+14], [Zhu07]. Auch in einem Entwicklungswerkzeug für räumliche optische Baugruppen sind für eine sinnvolle Entwicklung entsprechende Funktionen notwendig. Dies sind spezifische optomechatronische 3D-Bauelementbibliotheken und zusätzliche optische Konstruktionselemente, wie beispielsweise Durchkontaktierungen.

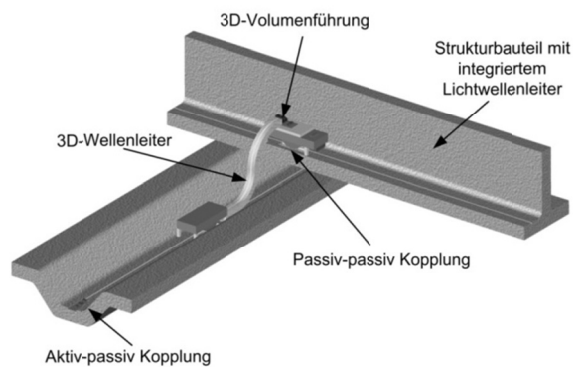


Bild 3: Konzept für einen 3D-Wellenleiter im Rahmen der Forschergruppe OPTAVER

Für Lichtwellenleiter (LWL) sind optoelektronische Design-Regeln zu übertragen. Beim Entwurf können so frühzeitig fertigungstechnische Schwierigkeiten vermieden werden. Spezifische Regeln für Gestalt und Anordnung optoelektronischer Elemente sowie deren Toleranzanforderungen für 3D-Schaltungsträger werden noch im Rahmen paralleler Forschungsprojekte identifiziert.

3.2 Umsetzung eines spezifischen Konstruktionswerkzeugs

Um diese Problemstellung anzugehen, wird am Lehrstuhl FAPS, in Zusammenarbeit mit dem Institut für Optik, Information und Photonik in Erlangen, ein Werkzeug zur Simulation und zum Entwurf optischer Baugruppen entwickelt. Weiterhin sind das IAVT der Universität Dresden sowie das LZH und ITA der Universität Hannover, als in anderen Teilprojekten der DFG-Forschergruppe OPTAVER (Optische Aufbau- und Verbindungstechnik für baugruppenintegrierte Bussysteme) beteiligt. Dabei sollen die Fähigkeiten von ausgewählten bestehenden Softwaresystemen der 3D-MID-Technologie sowie der 2D-Wellenleiter-Modellierung und Simulation zu neuer Funktionalität aggregiert werden. Dazu müssen entsprechende Datenschnittstellen genutzt oder bei Bedarf neu geschaffen werden. Entsprechende Vorgehensweisen werden folgend vorgestellt:

Für den Datenaustausch zwischen zwei Programmen ist es möglich, systemneutrale Schnittstellen, wie z.B. XML, zu verwenden. Dazu werden üblicherweise je ein Prä- und ein Postprozessor benötigt. Ein Präprozessor wandelt das systemeigene, native Datenformat in ein systemneutrales Schnittstellenformat, z.B. Initial Graphics Exchange Specification (IGES), um. Der Postprozessor des anderen Programms setzt das Schnittstellenformat wiederum in sein natives Datenformat um. Beim Datenaustausch über ein Produktmodell wird dieses als zentrales Organ zur Verwaltung und Speicherung aller für den Datenaustausch vorgesehenen Produktdaten definiert. Administrativ-, Meta-, Berechnungs- und Simulationsdaten können logisch verknüpft werden.

Ein elektromechanisches Produktmodell für 3D-MID Baugruppen bezüglich deren Entwicklung in verschiedenen CAD-Systemen wurde am Lehrstuhl FAPS entwickelt

[Zhu07], [Kre96]. Die Realisierung eines systemneutralen Produktdatenmodells geschieht meist auf der Basis des sogenannten „Standard for the exchange of product model data“ (STEP), da dieser sowohl für den Produktdatenaustausch mittels Dateien als auch für die Integration der Produktdaten in Datenbanken genutzt wird [Had00].

Es lassen sich aus den vorgestellten Methoden und Systemen für die CAD-Integration demnach zwei grundlegende Strategien für optomechatronische Funktionen ableiten:

1. *Entwicklung eines integrierten Produktmodells:* Diese Strategie versucht vor allem, bestehende oder neu entworfene Produktmodelle zu erweitern. Diese sollen alle Daten, die in der Produktentwicklung vorhanden sind, erfassen. Im Falle einer Erweiterung können Spezifikationen und Anforderungen direkt an ein Produktmodell angehängt werden. Dies geschieht analog zu den Vorarbeiten der Forschungsprojekte MIDCAD oder insbesondere MID-Layout, welches auf einem dateineutralen Produktmodell basiert.
2. *Programmgesteuerte Zugriffsmethoden:* Zugriffe auf bestehende Modelle erfolgen zentral über eine Programmierschnittstelle (API – Application Programming Interface). Entsprechende Funktionen und Spezifikationen können so benutzt werden. Das Spektrum der Zugriffsmethodik reicht dabei von Kommunikationsstandards wie TCP/IP über CAD-System-spezifische Bibliotheken bis hin zur objektorientierten Kapselung und Integration in eine verteilte Arbeitsumgebung. Dies setzt voraus, dass entsprechende Zugriffsmöglichkeiten der Softwarepakete bestehen.

Die Kapselung komplexer Berechnungsfunktionen, wie das Monte-Carlo Raytracing für Lichtwellenleiter, kann in externen Programmen erfolgen. Dies setzt voraus, dass entsprechende Geometrien aus dem 3D-Umfeld des verwendeten CAD-Systems in andere Simulationsumgebungen exportiert werden. Weiterhin müssen zu identifizierende Konstruktionsrichtlinien und Design-Strategien im 3D-Umfeld berücksichtigt werden. Diese müssen in einem eigenen Produktmodell realisiert werden.

4 Resümee und Ausblick

Räumliche elektronische Baugruppen benötigen spezifische Entwicklungssysteme für den fertigungsgerechten Entwurf. Eine einfache parallele Entwicklung der unterschiedlichen Domänen ist im Unterschied zum konventionellen Leiterplattenentwurf aufgrund der gegenseitigen Abhängigkeiten von Mechanik und Elektronik in mechatronischen Bauteilen nicht mehr möglich. Insbesondere bei der Aggregation mit optischen Bauelementen verschärft sich diese Anforderung zusätzlich. Zu den schon existierenden Forschungsarbeiten im Bereich integrierter Entwicklungssysteme kommen neue Anforderungen aus der Lichtwellenleitertechnik hinzu. Durch die DFG-Forschergruppe OPTAVER kann ein erster Schritt in diese Richtung erfolgen. Voraussetzung dafür sind entsprechende Forschungsarbeiten im Bereich der Fertigung sowie der Aufbau- und Ver-

bindungstechnik. Die Lösungen sind in einem entsprechenden Konstruktions- und Simulationswerkzeug umzusetzen. Auf Basis eines neuen Produktmodells werden Design- und Fertigungsrichtlinien implementiert. Ein rechnergestütztes Konstruktionswerkzeug für optomechatronische Baugruppen berücksichtigt über die schon bekannten Anforderungen aus Elektrotechnik und Maschinenbau hinaus auch optische Herausforderungen. Die beschriebenen Analogien aus dem MID-Bereich zeigen auf, dass eine wirtschaftliche Lösung möglich scheint.

Literatur

- [Fra13] FRANKE, J.: Räumliche elektronische Baugruppen (3D-MID) : Werkstoffe, Herstellung, Montage und Anwendungen für spritzgegossene Schaltungsträger. München: Hanser, 2013
- [KF05] KREBS, T.; FRANKE, J.: Konstruktionswerkzeuge für elektronisch/mechanisch integrierte Produkte : ECAD- und MCAD-Funktionen in einem dreidimensionalen Entwicklungssystem integriert. In: *Elektronik 2005* (2005), Nr. 18, S. 60-66
- [Zhu07] ZHUO, Y.: Entwurf eines rechnergestützten integrierten Systems für Konstruktion und Fertigungsplanung räumlicher spritzgegossener Schaltungsträger (3D-MID). Bamberg: Meisenbach, 2007 (Fertigungstechnik Erlangen Bd. 180)
- [Zuk14-ol] ZUKEN GMBH. URL: <http://www.zuken.com/de/products/pcb-design/cr-5000/products/cr-products-a-z/board-interchanger> (Abfragedatum 30.11.2014)
- [Uhl08] UHL, K.: PTC Pro/Engineer Wildfire 4. In: *CAD CAM* (2008), 1-2, S. 60-62
- [FFF09] FRANKE, J.; FELDMANN, K.; FISCHER, C.: Konstruktionswerkzeuge für Molded-Interconnect-Device (3D-MID) Technologie mit Bezug zu CAM (Kongress Internationales Forum Mechatronik). Linz, 2009-11-10
- [ZFG+14] ZEITLER, J.; FISCHER, C.; GOETZE, B.; MOGHADAS, S. H.; FRANKE, J.: Integration of Semi-Automated Routing Algorithms for Spatial Circuit Carriers into Computer-Aided Design Tools. In: *Proceedings of the 13th Electronic Circuits World Convention*, 2014
- [Sch08] SCHEEL, W.: *Optische Aufbau- und Verbindungstechnik in der elektronischen Baugruppenfertigung: Einführung*. 1. Aufl. Templin/Uckermark: Detert, 2002 (Optische Aufbau- und Verbindungstechnik)
- [FCR08] FELDMANN, K.; CRAIOVAN, D.; RÖSCH, M.: Entwicklung und Qualifizierung einer spezifischen Montagetechnologie zur Herstellung elektrooptischer Baugruppen. In: DVS; GMM (Hrsg.): *Elektronische Baugruppen und Leiterplatten - EBL 2008: Systemintegration und Zuverlässigkeit*. Berlin, Offenbach: VDE Verlag GmbH, 2008 (GMM-Fachbericht Band 55, 55), S. 75-80
- [FC09] FRANKE, J.; CRAIOVAN, D.: Modified assembly systems and processes for the mounting of electro-optical components. In: IEEE; ISAM (Hrsg.): *Proceedings of the International Symposium on Assembly and Manufacturing 2009 (ISAM)*: IEEE, 2009, S. 143-148
- [Mei02] MEIER, R.: *Strategien für einen produktorientierten Einsatz räumlicher spritzgegossener Schaltungsträger (3-D MID)*. Bamberg : Meisenbach Verlag, 2002 (Fertigungstechnik Erlangen Bd. 135)
- [BSS09] BIERHOFF, T.; STÜBBE, O.; SCHRAGE, J.: Computer Aided Design of Board-level Optical Interconnects. In: *C-Lab Reports 2009* (2009), Vol 8, No. 4

- [Kre96] KREBS, T.: Integration elektromechanischer CA-Anwendungssysteme über einem STEP-Produktmodell. Bamberg: Meisenbach, 1996 (Fertigungs-technik Erlangen Bd. 59)
- [Had00] HADERER, G.: Integration von Gestaltung und Berechnung mittels CORBA. Berlin, Technische Universität Berlin, Fachbereich 11 – Maschinenbau und Produktionstechnik: Dissertation. 2000

Autoren

M.Sc. Jochen Zeitler studierte Mechatronik und Wirtschaftsingenieurwesen an der Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) sowie der Hochschule Hof. Seit 2013 ist er wissenschaftlicher Mitarbeiter am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik (FAPS) der Friedrich-Alexander-Universität Erlangen-Nürnberg. Sein Forschungsschwerpunkt liegt auf dem integrierten Layoutentwurf räumlicher mechatronischer Schaltungsträger und der Entwicklung von CAD/CAM-Systemen.

Dipl.-Inf. Christian Fischer studierte Informatik an der FAU. Er ist wissenschaftlicher Mitarbeiter in der Gruppe System Engineering des Lehrstuhls FAPS und beschäftigt sich mit den softwaretechnischen Lösungen in Bezug auf die Konstruktion mechatronischer Produkte und 3D-MID im Speziellen.

Prof. Dr.-Ing. Jörg Franke ist seit 2009 Leiter des Lehrstuhls FAPS an der Friedrich-Alexander-Universität Erlangen-Nürnberg. Dort legt er den Forschungsschwerpunkt auf die Entwicklung neuer Fertigungsverfahren für mechatronische Produkte, insbesondere für elektronische Baugruppen, die Montage von Elektromotoren, sowie die Planung und Simulation komplexer mechatronischer Systeme. Prof. Franke ist unter anderem Vorsitzender der Forschungsvereinigung Räumliche elektronische Baugruppen 3D-MID e.V. sowie Sprecher des bayrischen Clusters Mechatronik und Automation.

Systematik zur Kostenbewertung von mechatronischen Systemen in der Technologie Molded Interconnect Devices

Thomas Schierbaum, Prof. Dr.-Ing. Jürgen Gausemeier

Heinz Nixdorf Institut

Fürstenallee 11, 33102 Paderborn

Tel. +49 (0) 5251 / 60 62 67 , Fax. +49 (0) 5251 / 60 62 68

E-Mail: {thomas.schierbaum|juergen.gausemeier}@hni.upb.de

Zusammenfassung

Hohe Funktionsdichte auf kleinem Bauraum und damit einhergehende Miniaturisierung sind Erfolgsfaktoren für eine Vielzahl von mechatronischen Produkten. Ein Innovationsstreiber in diesem Bereich ist die Technologie MID (Molded Interconnect Devices). Sie ermöglicht es, Elektronik direkt in mechanische Bauteile zu integrieren. Planare Schaltungsträger lassen sich so durch räumliche ersetzen. Das spart Raum, Teile und Kosten. Speziell die korrekte Kostenbewertung ist oftmals eine schwierige Hürde bei innovativen MID-Projekten. Bereits in der MID-Studie 2011 „Markt- und Technologieanalyse“ wurde die Bewertung als einer von vier wesentlichen Schlüsselfaktoren für erfolgreiche MID-Projekte ausgemacht [FGG+11]. Dies wird durch Aussagen aus der Praxis untermauert. Wir präsentieren in diesem Beitrag eine Systematik, die eine korrekte Kostenbewertung von MID-Produkten ermöglicht. Dazu gehört die Definition des Betrachtungsgegenstandes, welcher die Basis darstellt; Das Kostenmodell beschreibt alle relevanten Aspekte des Produkts und Produktionssystems, die bewertet werden müssen. Abschließend wird die Systematik anhand des Miniaturroboters BeBot angewendet.

Schlüsselworte

Molded Interconnect Devices, MID, Kostenbewertung, Kostenmodell, Entwicklungsmethodik, Mechatronik

Cost Appraisal of Mechatronic Systems in the Technology Molded Interconnect Devices

Abstract

High functional density on small space including strong miniaturization are factors of success of a plurality of mechatronic products. One innovation driver is the technology Molded Interconnect Devices (MID). It enables the integration of electronic and mechanical functions. Common devices such as FR4 boards can be replaced by spatial devices. This reduces space, parts and costs. A central challenge is the economic efficiency analysis which was identified as a main key factor for successful MID-projects in the MID study 2011 “market and technology analysis” of the research association molded interconnect devices 3-D MID e.V. A central need for action was identified by considering the comparability of the whole process chain [FGG+11]. In this Paper we present a systematics, which enables a correct cost evaluation of MID-products. Therefore the subject matter needs to be defined, which is the basis for further investigation. The cost model describes all relevant aspects of the product and production system, which need to be evaluated. In the end the systematics will be applied to the miniature robot BeBot.

Keywords

Molded Interconnect Devices, MID, cost evaluation, cost model, design methodology, mechatronic systems

1 Einleitung

Sich immer schneller ändernde Märkte bilden eine Herausforderung für produzierende Unternehmen. Nahezu alle Einflussfaktoren (Kunden, Technologien, Gesetze etc.), die auf Unternehmen wirken, verändern sich dynamisch. Diese Situation ist grundsätzlich nicht neu. So führte beispielsweise die Industrialisierung zu Beginn des letzten Jahrhunderts zur Einführung der Arbeitsteilung und der Trennung von planenden und ausführenden Tätigkeiten. Heute führt die zunehmende Individualisierung von Produkten zu einer zunehmenden Variantenvielfalt und kürzeren Produktlebenszyklen. Gleichzeitig steigen die Komplexität der Produkte und die Qualitätsanforderungen. Dies führt bei den Unternehmen zu einem stetig wachsenden Zeit- und Innovationsdruck in der Produkt- und Prozessentwicklung. Die räumliche Integration mechanischer und elektronischer Komponenten, verbunden mit Funktionsintegration und Miniaturisierung eröffnet dabei neue Möglichkeiten in der Produktgestaltung. Spritzgegossene Schaltungsträger (Molded Interconnect Devices) bilden hierfür ein leistungsfähiges Beispiel:

MID-Teile sind räumliche elektro-mechanische Funktionsträger. Elektronische Funktionen lassen sich durch Leiterbahnen erzeugen. Mechanische Funktionen lassen sich in Form des Spritzgussteils integrieren. Das spart Raum, Teile und Kosten.

Speziell die korrekte Kostenbewertung ist oftmals eine schwierige Hürde bei innovativen MID-Projekten. Bereits in der MID-Studie 2011 „Markt- und Technologieanalyse“ [FGG+11] wurde die Bewertung als einer von vier wesentlichen Schlüsselfaktoren für erfolgreiche MID-Projekte identifiziert. Dies wird durch Aussagen aus der Praxis untermauert. In vielen Fällen wird die Technologie MID fälschlicherweise als Kostentreiber ausgemacht. Eine korrekte Kostenkalkulation setzt die Betrachtung der gesamten Prozesskette sowie des Gesamtsystems voraus. Es müssen alle Prozesse von dem Spritzguss über die Metallisierung bis hin zur Aufbau und Verbindungstechnik sowie weitere vor- und nachgelagerte Prozesse betrachtet werden.

In diesem Beitrag wird zunächst die Technologie MID näher vorgestellt. Anschließend erfolgt eine Analyse der aktuellen Arbeiten. In Kapitel 4 werden die Bestandteile der Systematik näher erläutert. In Kapitel 4.1 wird zunächst das Anwendungsbeispiel, der Miniaturroboter BeBot, vorgestellt. In Kapitel 4.2 wird die Definition des Betrachtungsgegenstandes erläutert. In Kapitel 4.3 wird das Kostenmodell vorgestellt, welches einen zentralen Aspekt der Systematik darstellt. Im darauffolgenden Kapitel werden die Kosten des BeBot betrachtet. Der Beitrag endet mit einem Resümee und Ausblick.

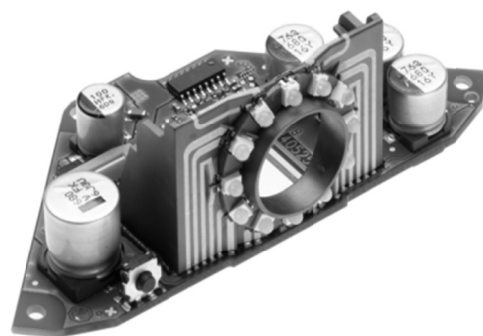
2 Die Technologie Molded Interconnect Devices (MID)

Die MID-Technologie bietet Vorteile wie 3D Gestaltungsfreiheit, hohe Zuverlässigkeit und mitunter geringere Herstellungskosten im Vergleich zu herkömmlichen elektronischen Baugruppen. **MID-Produkte** sind spritzgegossene thermoplastische Bauteile mit integrierten dreidimensionalen elektronischen Leiterbahnen. Die Oberfläche der ver-

wendeten Thermoplaste kann selektiv metallisiert werden. Dies ermöglicht die Integration von elektronischen Komponenten mittels Aufbau- und Verbindungstechnik. Darüber hinaus können elektronische Funktionen wie Antennen, elektromagnetische Abschirmungen oder Wärmebrücken realisiert werden. Das Design des thermoplastischen Schaltungsträgers selbst erfüllt dabei die mechanischen Funktionen des Bauteils. Montage und Anschlussfunktionen sind möglich, ebenso wie der Schutz vor Umwelteinflüssen oder Wärmemanagement. Die Zahl der serienmäßig hergestellten MID-Anwendungen hat in den letzten Jahren deutlich zugenommen. Bild 1 zeigt zwei Anwendungsbeispiele.



Quelle: KaVo Dental, 2E mechatronics



Quelle: Harting Mitronics, Sick

Bild 1: Multi LED-Baugruppe für Zahnarztbehandlungseinheiten (links) und Kameramodul für Sicherheitssysteme (rechts)

Die räumliche Gestaltungsfreiheit ermöglicht eine optimale Ausrichtung der elektronischen Komponenten entsprechend ihrer Einsatzumgebung, z.B. bei optischen Sensoren. Insbesondere bei Anwendungen mit begrenztem Platzangebot können neue Produktkonzepte umgesetzt werden. Diese Miniaturisierung geht oft auch mit einer Gewichtsreduktion einher. Auf Herstellerseite lässt sich die Prozesskette in der Produktion rationalisieren. Die Integralbauweise unterstützt zudem die Teilereduktion. Dies wirkt sich positiv auf die Handling- und Montagekosten aus. Integrierte Installationshilfen, wie Schnapphaken, vereinfachen die Montage und reduzieren die Anzahl fehlerhafter Verbindungselemente innerhalb der Produktion.

Zur **Herstellung der MID-Baugruppen** steht eine Vielzahl unterschiedlicher Verfahren zur Verfügung. Zu den am weitverbreitetsten Verfahren zählen das Zweikomponentenspritzgießen, das Laserstrukturieren, das Heißprägen und das Folienhinterspritzen. Gleichläufig wächst die Bedeutung von Plasma- und Druckverfahren. Die unterschiedlichen Verfahren werden durch gemeinsame Prozessstufen gekennzeichnet. Dabei verbindet der Begriff MID-Herstellverfahren die Prozesse des Spritzgießens, Strukturierens sowie Metallisierens. Die Prozessschritte sind notwendig um den MID-Grundkörper zu fertigen. Die nachgelagerten Prozesse der Aufbau- und Verbindungstechnik (AVT) dienen dem Aufbringen der elektronischen Komponenten auf dem Grundkörper. Die Zusammenführung aus MID-Herstellverfahren und AVT wird als MID-Prozesskette be-

zeichnet (siehe Bild 2). Das MID-Produkt ist stark durch die Restriktionen der einzelnen Prozessschritte determiniert. Die für MID-Bauteile typische Integralbauweise verstärkt diese Abhängigkeiten, zudem beeinflussen sich Änderungen am Produkt oder Produktionssystem gegenseitig. Die Integration von mechanischen und elektronischen Funktionsträgern führt zudem zu zahlreichen Wechselwirkungen innerhalb des MID-Bauteils [Fra13].

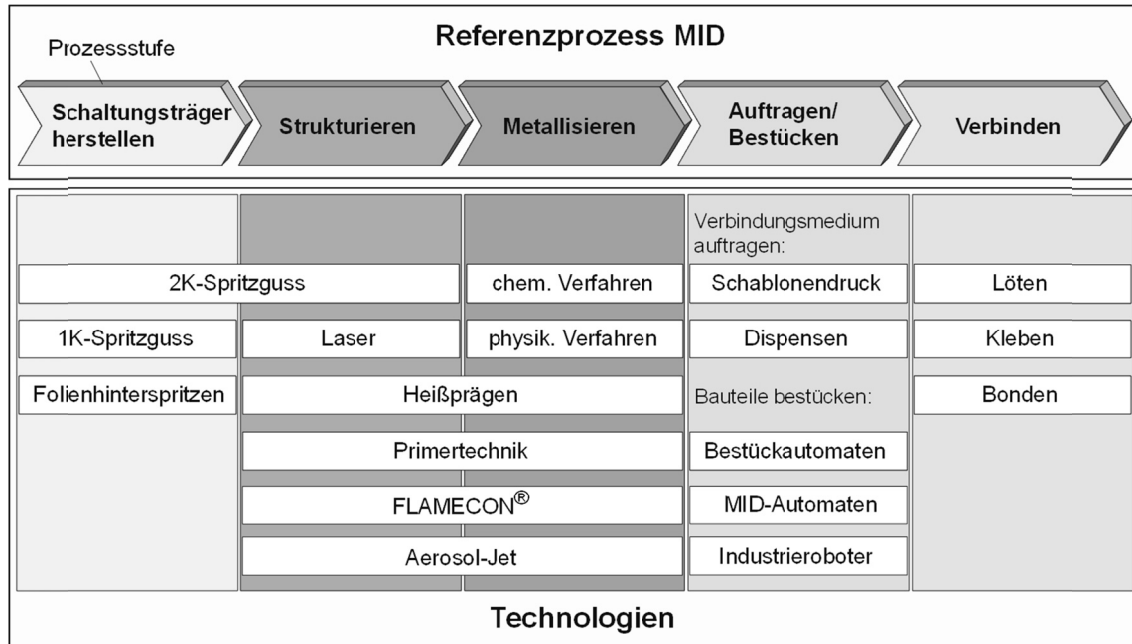


Bild 2: Referenzprozess für MID-Baugruppen nach [GDG10]

3 Analyse bisheriger Arbeiten der Kostenbewertung

Vereinzelt existieren Methoden zur Kostenbewertung von MID. FRANKE entwickelte einen Ansatz, der Kostenstrukturen und Nutzenpotentiale der Technologie MID transparent machen soll. Basis für diesen Ansatz ist das Bewertungsverfahren der Prozessübergangskosten. Es wird ermittelt, wann ein Übergang von konventionellen Technologien zu MID wirtschaftlich sinnvoll ist. Hierbei steht die strategische Entscheidung im Vordergrund. Konkreten Bezug auf Fertigungstechnologien erarbeitete FRANKE indem er Kostenstrukturen für MID-Herstellverfahren definierte. Es wurde jedoch lediglich ein Teil der Prozesskette abgebildet [Fra95]. Zur vollumfänglichen Kostenbetrachtung gilt es die gesamte Prozesskette abzubilden, damit MID-Fertigungsverfahren mit herkömmlichen Verfahren verglichen werden können. Methoden aus der Betriebswirtschaftslehre befassen sich mit nur einem unzureichenden Detaillierungsgrad mit der Bewertung. Sie geben oftmals lediglich Kennzahlen (Return-On-Investment oder Break-Even-Point) aus [WD10]. Technologiespezifische Aspekte bleiben häufig unberücksichtigt. Speziell im Hinblick auf die große Anzahl unterschiedlicher MID-Herstellverfahren, die jeweils einen starken Einfluss auf die Produktgestalt und Fertigung haben, ergeben sich in diesem Spannungsfeld Herausforderungen für die Kostenbewertung [Fra95]. Die Vielzahl

von Einflussfaktoren auf die Kosten macht eine Betrachtung schwierig. Der Kostenvorteil der MID-Lösung ergibt sich nicht einfach durch die Betrachtung der Einzelkosten; vielmehr durch die Integration mehrerer Bauteile/Funktionen und einer entsprechend vereinfachten Bauteil- und Systemmontage [Pöh98].

Weitere allgemeinere Bewertungsmethoden sind die Nutzwertanalyse, technische-wirtschaftliches Konstruieren nach VDI-Richtlinie 2225, der Analytic Hierarchy Process (AHP) oder die Wertanalyse nach VDI-Richtlinie 2800. Die Nutzwertanalyse ist vielfältig einsetzbar und es existieren etliche Varianten dieser Methode. Ursprünglich wurde sie Mitte der 1970er Jahre von ZANGEMEISTER und BECHMANN entwickelt. Sie wird häufig bei komplexen Entscheidungsproblemen eingesetzt. Das Gesamtproblem wird, mit dem Ziel der Vereinfachung des Entscheidungsproblems, fragmentiert. Die entstanden Teilprobleme werden anhand von Kriterien bewertet. In der Regel reicht die Bewertung von 0 (schlecht) bis 10 (sehr gut). Die Variante mit dem höchsten Nutzwert gilt als die vorteilhafteste [Zan76], [Kue14]. Die Methode nach VDI-Richtlinie 2225 fokussiert im Wesentlichen die Ermittlung von Bauteilkosten auf Basis der Geometrie und des verwendeten Werkstoffs. Es wurde ein umfangreiches Tabellenwerk erarbeitet, welches für verschiedene Erzeugnistypen (z.B. Krane oder Staubsauger) angewandt werden kann [VDI2225]. Der Analytic Hierarchy Process wurde 1971 von SAATY entwickelt. Wie bei der Nutzwertanalyse fokussiert der AHP komplexe Entscheidungsprobleme. Die Bewertung wird relativ zwischen zwei Entscheidungsalternativen durchgeführt. Alle Teilbewertung werden anschließend aufsummiert. Die Alternative mit der größten Summe ist die vorteilhafteste und ist auszuwählen. Abschließend wird eine Sensitivitätsanalyse durchgeführt. Sie besagt, wie robust das Ergebnis gegenüber einer Änderung ist bzw. wie sensibel es auf eine Änderung der Kriteriengewichtung reagiert. Ein Ergebnis gilt als robust, wenn marginale Änderungen keinen Einfluss auf die Rangfolge der Alternativen haben [SV12]. Die Wertanalyse wurde 1947 in den USA entwickelt. Ihr Fokus liegt auf der Verbesserung existierender Produkte sowie der Entwicklung neuer Produkte. Im Fokus der Betrachtung stehen jene Kosten, die sich nicht auf die Qualität, den Gebrauch, die Lebensdauer, die Optik oder den Verkauf beziehen. Für die Bewertung stellt die VDI-Richtlinie 2800 ein Methodenportfolio (z.B. Quality Function Deployment oder Return-On-Investment) bereit. Ziel ist entweder die Nutzenmaximierung bei gegebenen Kosten oder die Kostenminimierung bei gegebenem Nutzen [VDI2800]. Die Zielsetzung entspricht dem ökonomischen Prinzip nach Erich Gutenberg.

Fazit: Es fehlt an einer durchgängigen Systematik zur Kostenbewertung, welche alle relevanten Aspekte mit ins Kalkül zieht. Die Aspekte umfassen unter anderem die Bewertung der Produkt- als auch der Produktionssystemkosten. Ein weiterer wesentlicher Aspekt für eine Kostenbewertung ist die korrekte Definition des Betrachtungsgegenstandes, welcher für alle Lösungskonzepte gleich ist und die Basis für die Bewertung darstellt. Zudem sind die Entwickler mit geeigneten Methoden zu unterstützen, da diese

zumeist wenig Erfahrung mit dem Umgang der Technologie MID haben. Daher wurde eine Systematik zur Kostenbewertung entwickelt. Sie wird im Folgenden vorgestellt.

4 Systematik zur Kostenbewertung

Die Systematik zur Kostenbewertung umfasst mehrere Bestandteile: Zu Beginn muss der Betrachtungsgegenstand (vgl. Kap 4.2) definiert werden. Hierfür wurde eine entsprechende Methode entwickelt, welche auf dem Systems Engineering (SE) basiert. SE wird als durchgängige, fachdisziplinübergreifende Disziplin zur Entwicklung technischer Systeme verstanden. Das Model-Based Systems Engineering unterstützt modellbasiert, bereits in der Konzipierung, bei der Entwicklung technischer Systeme [GDS+13]. Zur Modellierung wird die Spezifikationstechnik CONSENS genutzt, welche am Heinz Nixdorf Institut entwickelt worden ist. Mit CONSENS werden Produkt und Produktionssystem mit Hilfe eines zusammenhängenden Systems von Partialmodellen beschrieben [GFD+09], [GLL06]. Aufbauend auf den Modellen wird eine Kostenbewertung durchgeführt. In Kapitel 4.3 wird das Kostenmodell vorgestellt. Es enthält alle relevanten Aspekte und gibt Aufschluss über die Berechnung der einzelnen Kosten. Die Berechnungsformeln sind als Orientierung zu verstehen und lassen sich vom Anwender individuell anpassen oder erweitern. In Kapitel 4.4 wird die Berechnung der Kosten anhand des Miniaturroboters BeBot durchgeführt. Zunächst erfolgt eine kurze Erläuterung des Anwendungsbeispiels.

4.1 Miniaturroboter BeBot

Der Miniaturroboter wurde am Heinz Nixdorf Institut als Demonstrator im Rahmen des Sonderforschungsbereichs 614 – Selbstoptimierende Systeme des Maschinenbaus – entwickelt. Er ist eine Forschungsplattform für Algorithmen der Schwarmintelligenz und Multi-Agenten Systeme. Ferner ist er ein Technologiedemonstrator für MID. Sein Gehäuse ist eine 2,5D MID-Baugruppe, welche mittels Laserdirektstrukturierung hergestellt wurde. Das Gehäuse sowie der BeBot sind in Bild 3 abgebildet. Auf der unteren Hälfte ist der gesamte BeBot zu sehen, mit Antrieben, Leiterplatten, Lichtleiter und Deckel mit integrierter W-LAN Antenne. Das MID-Gehäuse (oben) verfügt über zwölf Infrarotsensoren welche einen 360° Blick ermöglichen. Der BeBot ist so in der Lage Gegenstände und Hindernisse zu detektieren. Zusätzlich, zu weiteren elektronischen Bauteilen, ist ein Mikrokontroller auf der rückwärtigen Gehäuseinnenseite angebracht. Der Mikrokontroller ist für die Sensorauswertung und das Motorenmanagement zuständig. Zudem können zwei Leiterplatten in das Gehäuse gesteckt werden.

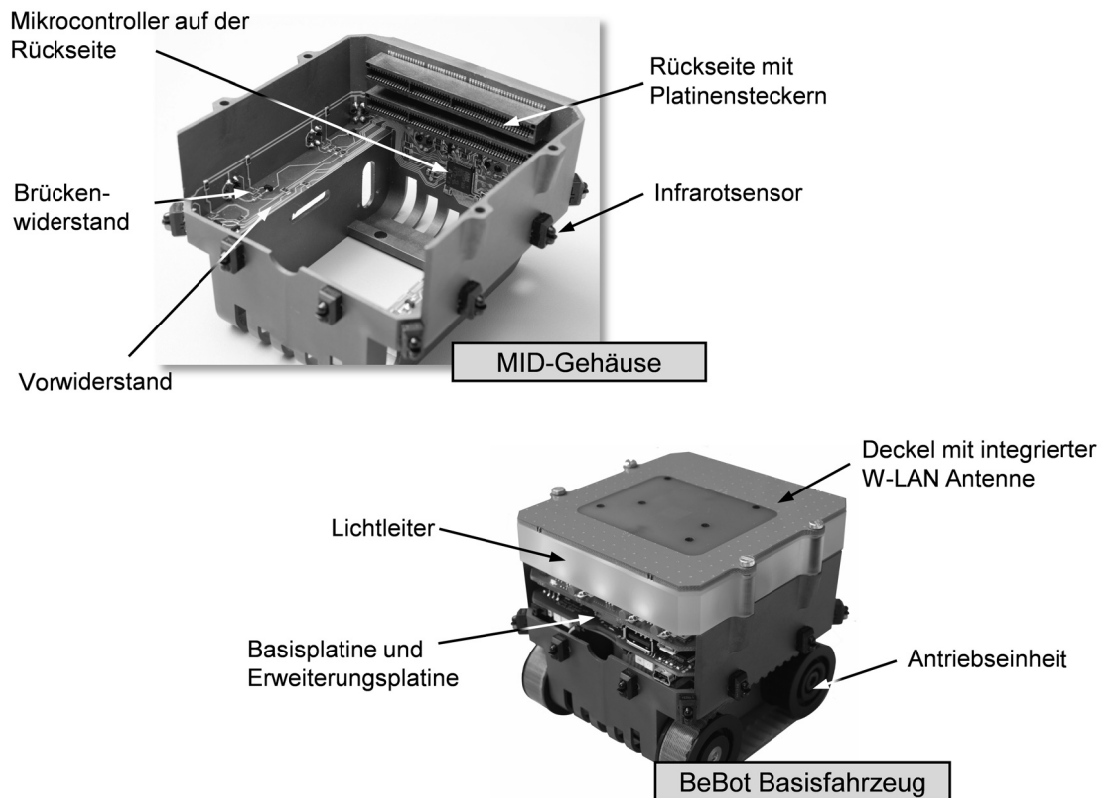


Bild 3: MID-Gehäuse (oben) und der Miniaturroboter BeBot (unten)

4.2 Definition des Betrachtungsgegenstandes

Bei der Definition des Betrachtungsgegenstandes sind zwei Aspekte wesentlich: (1) Alle relevanten Aspekte des Konzeptes müssen für die Bewertung Beachtung finden und (2) die Bewertung konkurrierender Konzepte muss auf einer einheitlichen Basis durchgeführt werden. Dafür wurde eine funktionale Sicht auf das System gewählt. In einer Funktionshierarchie werden alle Funktionen beschrieben, die vom System umgesetzt sind. Im oberen Bildbereich von Bild 4 ist ein Auszug der Funktionshierarchie des BeBot abgebildet. Alle fünf gezeigten Funktionen werden von dem MID-Gehäuse realisiert. Im nächsten Schritt wird die Wirkstruktur analysiert.

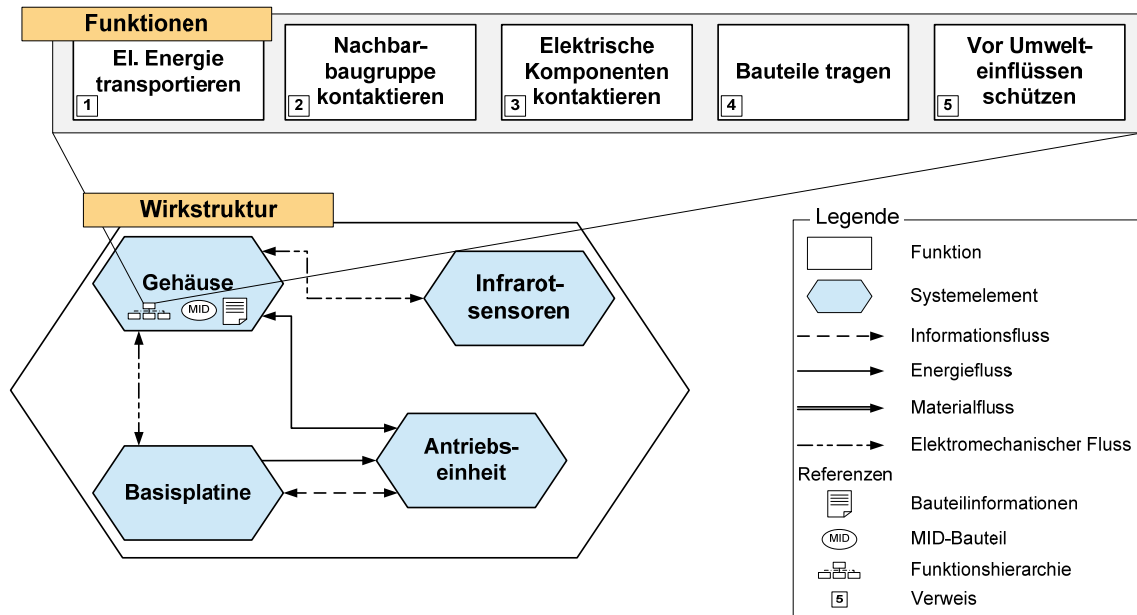


Bild 4: Auszug der Funktionshierarchie und Wirkstruktur des Miniaturroboters BeBot

Die Systemelemente und deren Flussbeziehungen die mit dem MID-Gehäuse in Beziehung stehen müssen betrachtet werden. Speziell mechanische oder elektromechanische Flussbeziehungen sind von großer Relevanz. Sie geben Aufschluss über die Fertigung der Baugruppe. Es lassen sich z.B. Montagezusammenhänge erkennen. Aufbauend auf dieser Erkenntnis wird der Fertigungsprozess modelliert.

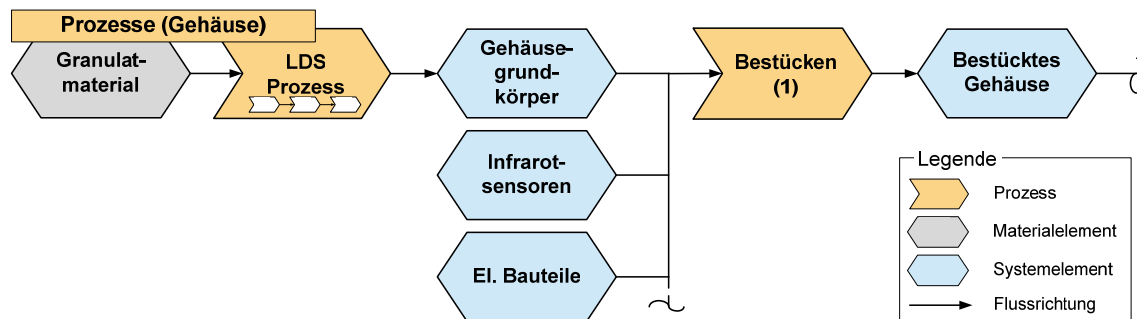


Bild 5: Auszug aus dem Fertigungsprozess für den Miniaturroboter BeBot

Bild 5 zeigt einen Auszug aus dem Fertigungsprozess für den BeBot. Die MID-Prozesskette ist als LDS-Prozess zusammengefasst. Das Piktogramm der zusammenhängenden Prozesspfeile verdeutlicht dies. Nach der Fertigung des Gehäuses wird dieses u.a. mit Infrarotsensoren und elektronischen Bauteilen bestückt. Anschließend erhält man das bestückte Gehäuse. Alle Prozesse die unmittelbar mit dem Gehäuse bzw. MID-Bauteil in Beziehung stehen, sind mit ins Kalkül zu ziehen. So wird der Anforderung der Betrachtung der gesamten Prozesskette Rechnung getragen.

4.3 Kostenmodell

Das Kostenmodell beschreibt alle relevanten Kostenaspekte des Produkts und Produktionssystems. Bild 6 zeigt die oberste Ebene des Kostenmodells für die Herstellkosten. Diese setzen sich aus den Materialkosten (Produktkosten) und den Fertigungskosten (Produktionskosten) zusammen. Das weiße Dreieck zeigt an, dass eine weitere Detaillierung der Kostenzusammensetzung existiert.

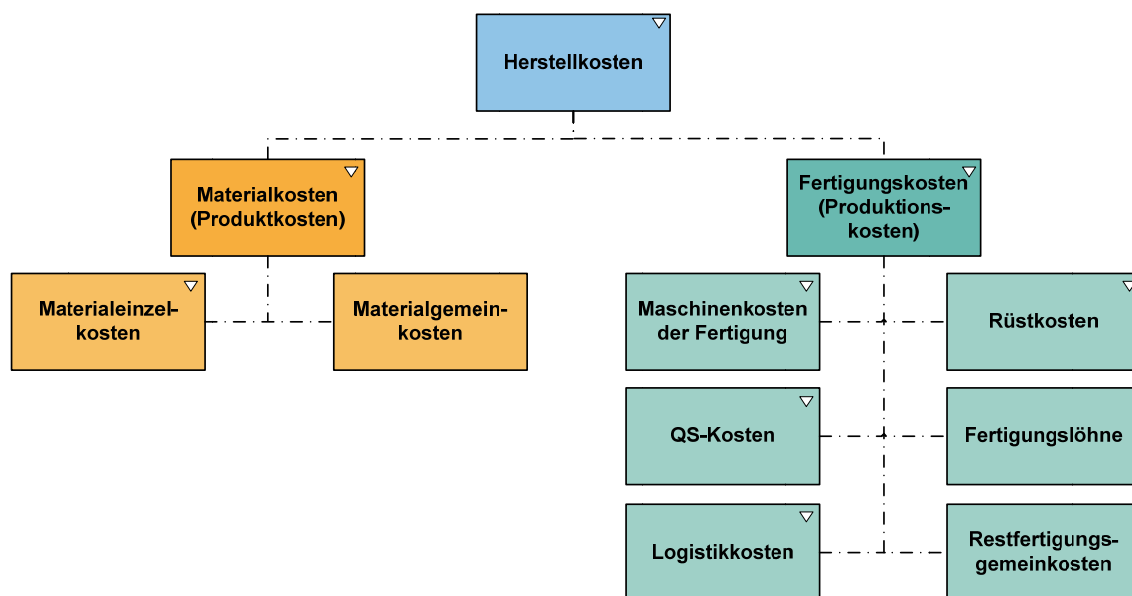


Bild 6: Kostenmodell für die Bestimmung der Herstellkosten (Auszug)

Zur Berechnung der Materialkosten sind die Materialeinzelkosten sowie die Materialgemeinkosten zu bestimmen. Die Materialeinzelkosten setzen sich aus Werkstoffkosten und den Bauelementkosten zusammen. Die Werkstoffkosten (WK) wiederum werden aus dem Volumen des Bauteils (V_{BT}), Volumen des Angusses (V_A), der Dichte des Werkstoffs (ρ_W), dem Kostenkoeffizienten des Werkstoffs (KK_W) und den Kosten des Referenzwerkstoffs (K_{RW}) gebildet: $WK = (V_{BT} + V_A) \cdot \rho_W \cdot KK_W \cdot K_{RW}$

Zur vereinfachten Berechnung werden Zuschlagssätze für die Werkstoffe gebildet. Es liegt die Annahme zugrunde, dass die Preisdifferenz zwischen Werkstoffen über lange Zeiträume annähernd konstant ist [VDI2225]. Es muss nur der Preis des Referenzwerkstoffs bekannt sein. Dies reduziert den Aufwand bei der Berechnung der Werkstoffkosten. Die Materialgemeinkosten sind mit einem Aufschlagssatz angegeben, wie in der Kosten und Leistungsrechnung üblich. Die Fertigungskosten setzen sich aus den Maschinenkosten der Fertigung zusammen sowie den Fertigungslöhnen. Hinzu kommen Rüstkosten, Qualitätssicherungskosten, Logistikkosten und Restfertigungsgemeinkosten. Es müssen nicht alle Kosten entstehen, einzelne Aspekte können mit unter vernachlässigt werden, da sie entweder nicht existieren oder marginal sind.

4.4 Betrachtung der Kosten für den Miniaturroboter BeBot

Wie oben beschrieben, wurde der Betrachtungsgegenstand definiert. Um die Materialkosten (Produktkosten) zu bestimmen, wurden alle relevanten Informationen (soweit bekannt) an das Systemelement des MID-Bauteils annotiert. Dazu gehören der Werkstoff sowie alle elektronischen Bauteile. Auf Basis des CAD-Modells kann das Volumen des Grundkörpers berechnet werden. Um die Fertigungskosten (Produktionskosten) zu bestimmen, ist die Prozesskette für den definierten Betrachtungsgegenstand zu erstellen. Jedem Prozessschritt kann eine Prozesskarte [SGD14] zugewiesen werden. Bild 7 zeigt einen Ausschnitt der annotierten Informationen des Gehäuses. An die Prozesse sind die Prozesskarten annotiert. Sie geben vielfältige Informationen über den Prozess, wie z.B. die Kostenzusammensetzung und benötigte Betriebsmittel. Auf Basis dieser Informationen können die Kosten bestimmt werden. Je detaillierter und genauer die Informationen sind, desto genauer ist die Kostenbestimmung des MID-Bauteils. Da sich alle Prozesse einzeln bewerten lassen, kann jeder Prozessschritt oder Teile der Prozesskette mit Angeboten von einem Auftragsfertiger verglichen werden. Eine make or buy Entscheidung für einzelne Prozesse bis hin zur gesamten Fertigung kann so herbeigeführt werden.

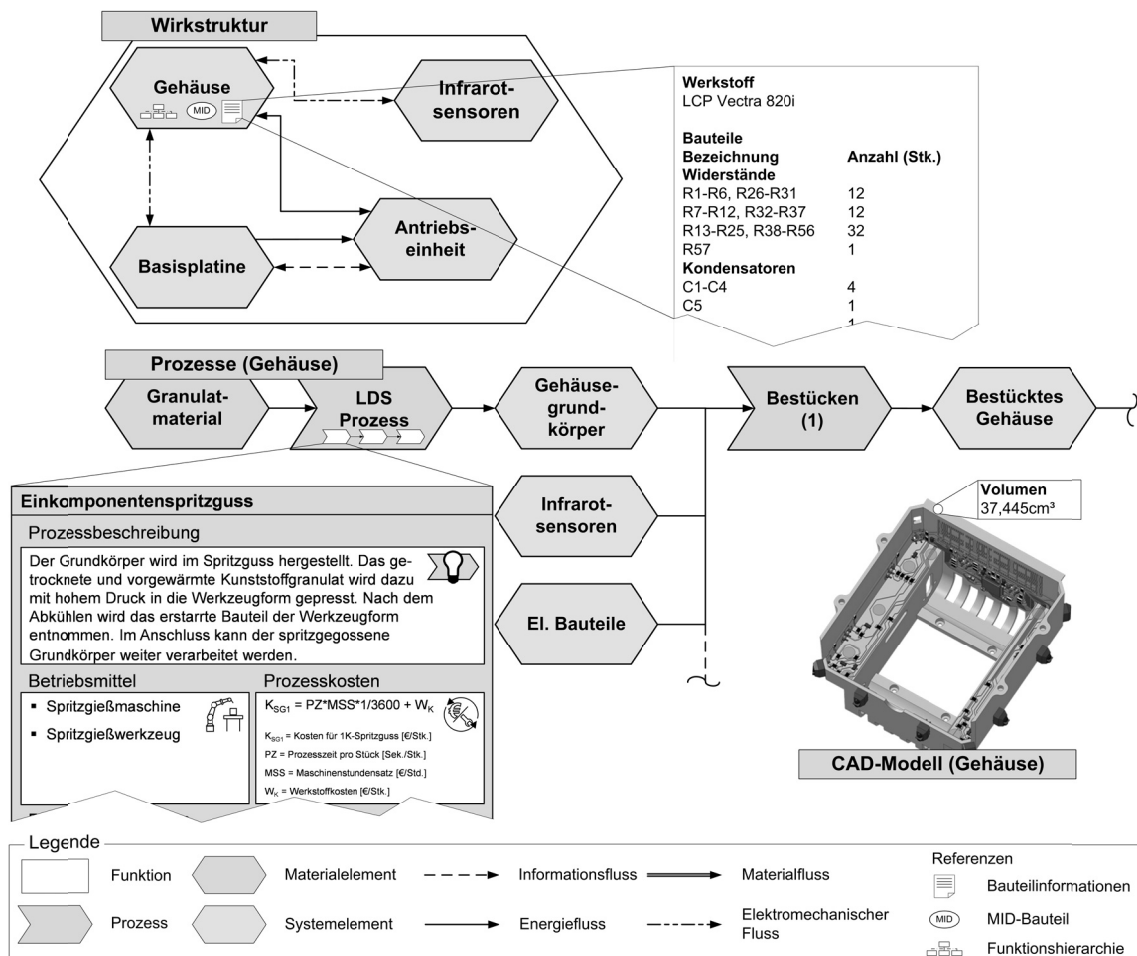


Bild 7: Betrachtung der Herstellkosten für den Miniaturroboter BeBot

5 Resümee und Ausblick

Der vorliegende Beitrag stellt eine Systematik zur Kostenbewertung eines MID-Bauteils vor. Dabei unterstützt sie u.a. die Definition des Betrachtungsgegenstandes. Er ist der wesentliche und entscheidende Faktor für die korrekte Bewertung. Zudem wird ein Modell für die Bewertung der Herstellkosten vorgestellt. Es umfasst alle relevanten Kosten auf Seiten des Produktes und Produktionssystems. Anhand des Anwendungsbeispiels BeBot (Miniaturroboter), wird die Systematik vorgestellt. Dafür wird auf die Spezifikationstechnik CONSENS zurückgegriffen. Sie ermöglicht eine ganzheitliche Modellierung des BeBots. Die Spezifikationstechnik wurde um Aspekte der Kostenbewertung erweitert: Es lassen sich relevante Informationen (z.B. Werkstoff) an die Systemelemente (z.B. Bauteile, Module) annotieren. Zudem wurden für die Fertigungsprozesse Prozesskarten entwickelt, welche dem Entwickler Informationen zur Verfügung stellen; u.a. über die Zusammensetzung der Prozesskosten je Prozessschritt. Mit Hilfe der vorgestellten Systematik können die Kosten für ein MID-Bauteil korrekt bestimmt werden. Die Güte der Genauigkeit der Kostenvorhersage hängt von der Güte der vorliegenden Informationen ab. Die Systematik unterstützt auch weniger mit der Technologie MID vertraute Anwender. Weitere Arbeiten gliedern die vorgestellten Aspekte in eine ganzheitliche Systematik zur Wirtschaftlichkeitsbewertung von MID-Produkten ein.

Literatur

- [FGG+11] FRANKE, J.; GAUSEMEIER, J.; GOTH, C.; DUMITRESCU, R.: MID Studie 2011 – Markt und Technologie Analyse. Forschungsvereinigung räumliche elektronische Baugruppen 3-D MID E.V., Erlangen, 2011
- [Fra13] FRANKE, J. (Hrsg.): Räumliche elektronische Baugruppen (3D-MID). Carl Hanser Verlag, München, 2013
- [Fra95] FRANKE, J.: Integrierte Entwicklung neuer Produkt- und Produktionstechnologien für räumliche spritzgegossene Schaltungsträger (3-D MID). Dissertation, Carl Hanser Verlag, München, 1995
- [GDG10] GAUSEMEIER, J.; DUMITRESCU, R.; GAUKSTERN, T.: Chancen der Technologie MID erkennen und nutzen. In: Produktion von Leiterplatten und Systemen (PLUS) 2/2010, Leuze Verlag, Bad Saulgau, 2010
- [GDS+13] GAUSEMEIER, J.; DUMITRESCU, R.; STEFFEN, D.; CZAJA, A. M.; TSCHIRNER, C.; WIEDERKEHR, O.: Studie – Systems Engineering in der industriellen Praxis, 2013
- [GFD+09] GAUSEMEIER, J.; FRANK, U., DONOTH, J.; KAHL, S.: ‘Specification Technique for the Description of Self-Optimizing Mechatronic Systems’, Research in Engineering Design, 20(4), Springer, London, 2009
- [GLL06] GAUSEMEIER, J.; LANZA, G.; LINDEMANN, U.: Produkte und Produktionssysteme integrativ konzipieren – Modellbildung und Analyse in der frühen Phase der Produktentstehung. Carl Hanser Verlag, München, 2006
- [Kue14] KUEHNAPFEL, J. B.: Nutzwertanalysen in Marketing und Vertrieb. Springer Fachmedien, Wiesbaden, 2014
- [Pöh98] PÖHLAU, F.: Entscheidungsgrundlagen zur Einführung räumlicher spritzgegossener Schaltungsträger (3-D MID). Bamberg, Meisenbach Verlag, 1998

- [SGD14] SCHIERBAUM, T.; GAUSEMEIER, J.; DUMITRESCU, R.: Method for the Identification and Comparison of Alternative Process Chains Focusing on Economics Efficiency Analysis During the Conceptual Design of Mechatronic Integrated Devices. In Proceedings 11th International Congress Molded Interconnect Devices, September 2014
- [SV12] SAATY, T. L.; VARGAS, L. G.: Models, Methods, Concepts & Applications of the Analytic Hierarchy Process. Springer Verlag, New York, 2. Auflage, 2012
- [VDI2225] VEREIN DEUTSCHER INGENIEURE (VDI): Technisch-wirtschaftliches Konstruieren – Vereinfachte Kostenermittlung. VDI-Richtlinie 2225, Blatt 1, Beuth Verlag, Berlin, 1997
- [VDI2800] VEREIN DEUTSCHER INGENIEURE (VDI): Wertanalyse. VDI-Richtlinie 2800, Blatt 1, Beuth Verlag, Berlin, 2010
- [WD10] WÖHE, G.; DÖRING, U.: Einführung in die Allgemeine Betriebswirtschaftslehre. Vahlen, München, 24. Auflage, 2010
- [Zan76] ZANGEMEISTER, C.: Nutzwertanalyse in der Systemtechnik – Eine Methodik zur multidimensionalen Bewertung und Auswahl von Projektalternativen. Dissertation, Wittmansche Buchhandlung, München, 1976

Danksagung

Das IGF-Vorhaben (18445N) der Forschungsvereinigung (Räumliche Elektronische Baugruppen 3-D MID e.V.) wurde über die AiF im Rahmen des Programms zur Förderung der Industriellen Gemeinschaftsforschung (IGF) vom Bundesministerium für Wirtschaft und Energie aufgrund eines Beschlusses des Deutschen Bundestages gefördert.

Autoren

Thomas Schierbaum studierte Wirtschaftsingenieurwesen mit der Fachrichtung Maschinenbau an der Universität Paderborn und der Qingdao University of Science and Technology (China). Seit 2010 ist Herr Schierbaum wissenschaftlicher Mitarbeiter von Herrn Prof. Gausemeier am Heinz Nixdorf Institut der Universität Paderborn. Seine Arbeitsschwerpunkte sind Systems Engineering, Entwicklungsmethodik Mechatronik sowie Molded Interconnect Devices (MID).

Prof. Dr.-Ing. Jürgen Gausemeier ist Seniorprofessor am Heinz Nixdorf Institut der Universität Paderborn. Seine Arbeitsschwerpunkte sind Strategische Produktplanung und Systems Engineering. Er war Sprecher des Sonderforschungsbereiches 614 „Selbstoptimierende Systeme des Maschinenbaus“ und von 2009 bis 2015 Mitglied des Wissenschaftsrats. Jürgen Gausemeier ist Initiator und Aufsichtsratsvorsitzender des Beratungsunternehmens UNITY AG. Seit 2003 ist er Mitglied von acatech – Deutsche Akademie der Technikwissenschaften und seit 2012 Vizepräsident. Ferner ist Jürgen Gausemeier Vorsitzender des Clusterboards des BMBF-Spitzenclusters „Intelligente Technische Systeme Ostwestfalen-Lippe (it’s OWL)“.

Datenflussmodellierung als Methode zur Optimierung von Entwicklungsprozessen am Beispiel der Leiterplattenentwicklung

***Dipl.-Ing. Stefan Herbrechtsmeier, Dr.-Ing. Thorsten Jungeblut,
Dr.-Ing. Mario Porrmann***

*CITEC – Universität Bielefeld – AG Kognitronik & Sensorik
Inspiration 1 (Zehlendorfer Damm 201), 33619 Bielefeld
Tel. +49 (0) 521 / 10 66 73 67, Fax. +49 (0) 521 / 10 61 23 48
E-Mail: sherbrec@cit-ec.uni-bielefeld.de*

Zusammenfassung

Entwicklungsprozesse sind zumeist historisch gewachsen und folgen einem linearen Verlauf. Die fachdisziplinübergreifenden Schnittstellen zwischen Entwicklungsprozessen werden oft nicht betrachtet und ein genaues Verständnis der Wechselwirkungen zwischen Entwicklungsaufgaben fehlt. Die in dieser Arbeit vorgestellte Datenflussmodellierung ermöglicht die Identifikation und Modellierung der Datenflüsse im Entwicklungsprozess. Die Modellierung basiert auf einer standardisierten Geschäftsprozessmodellierungssprache. Diese wurde hinsichtlich einer einfachen Modellierung von Datenflüssen angepasst und um eine Typisierung der Datenobjekte erweitert. Am Beispiel eines optimierten Leiterplattenentwicklungsprozesses und seinen Schnittstellen zu anderen Fachdisziplinen werden die Einsatzmöglichkeiten der entwickelten Datenflussmodellierung dargestellt.

Schlüsselworte

Datenflussmodellierung, Entwicklungsprozess, Leiterplattenentwicklung, Leiterplattenentwurf, Geschäftsprozessmodellierung

Data flow modelling as a method for optimization of development processes illustrated by the printed circuit board development

Abstract

Development processes usually grow historically and follow a linear progression. The interdisciplinary interfaces between development processes are often not considered and a detailed understanding of the interactions between development tasks is missing. The data flow modelling presented in this work enables the identification and modelling of data flows in the development process. The modelling is based on a standardized business process modelling language. This was adapted to a simple modelling of data flows and extended by a typing of the data objects. Using the example of an optimized printed circuit board design process and its interfaces to other disciplines the capabilities of the developed data flow modelling are presented.

Keywords

Data flow modelling, development process, printed circuit board development, printed circuit board (PCB) design, Business process modelling

1 Einleitung

Viele Entwicklungsprozesse in Unternehmen sind historisch gewachsen und folgen einem linearen Verlauf. Neue Produktentwicklungen sind oft eine Weiterentwicklung bestehender Produkte. Wenn auch nicht explizit im Entwicklungsprozess ersichtlich, profitiert doch der gesamte Prozess von dem Wissen und den Erfahrungen der Entwickler aus vorhergehenden Projekten. Insbesondere ihre Fähigkeit zur Abschätzung der Wechselwirkungen zwischen unterschiedlichen Fachdisziplinen beeinflusst den Verlauf der Entwicklung. Diese Fähigkeit kommt an ihre Grenzen, sobald auf neue Anforderungen reagiert werden muss, neue Fachdisziplinen zum Entwicklungsprozess hinzukommen oder Mitarbeiter aus dem Projekt ausscheiden. Im Zusammenhang mit Industrie 4.0 ist mit einer steigenden Komplexität und neuen Herausforderungen zu rechnen, so dass die Abschätzungen immer ungenauer werden. Bewährte Entwicklungsprozesse stoßen dadurch an ihre Grenzen und es ist damit zu rechnen, dass entweder immer seltener effiziente Lösungen gefunden werden oder die Anzahl der benötigten Iterationen im Entwicklungsprozess steigen. Um diesem Vorgang entgegen zu wirken, müssen die Wechselwirkungen zwischen Entwicklungsaufgaben als Teil des Entwicklungsprozesses betrachtet werden. Dies benötigt ein genaues Verständnis der Informationsflüsse zwischen den Entwicklungsaufgaben. Die im Folgenden vorgestellte Datenflussmodellierung adressiert diesen Bedarf.

Im Unterschied zur Geschäftsprozessmodellierung liegt der Schwerpunkt hierbei nicht in der Modellierung eines vollständigen Geschäftsprozesses, sondern in der Identifikation und Modellierung der Datenflüsse in einem Entwicklungsprozess. Die Datenflussmodellierung hilft, Abhängigkeiten und Wechselwirkungen im Entwicklungsprozess zu identifizieren und den Entwicklungsprozess anschließend zu strukturieren. Die strukturierte Datenflussmodellierung des Entwicklungsprozesses gibt dem Entwickler ein Verständnis über seine Beziehungen zu anderen Entwicklungsaufgaben und ermöglicht ihm, seine Kommunikationspartner zu identifizieren und somit selbstständig Abstimmungen, Optimierungen oder Konfliktlösungen im laufenden Entwicklungsprozess durchzuführen. Dies bedeutet gleichzeitig, dass die vorhergehenden Entwicklungsprozesse unter Umständen neu ausgeführt werden müssen. Die Grundidee ist, den Entwickler zu einem mündigen Teil des Entwicklungsprozesses zu machen und ihn nicht auf die reine Konkretisierung zu beschränken. Im Vergleich zu einer Matrix der Datenflüsse zwischen einzelnen Aufgaben visualisiert die Datenflussmodellierung zusätzlich den Prozess. Dies ist insbesondere bei der Identifizierung von Datenobjekten hilfreich.

Die von uns entwickelte und im Rahmen dieses Beitrags vorgestellte Datenflussmodellierung basiert auf der Geschäftsprozessmodellierung des Standards „Business Process Model and Notation“, die in Kapitel 2 vorgestellt wird. In Kapitel 3 folgt eine Beschreibung unserer Datenflussmodellierung. Kapitel 4 nutzt die entwickelte Datenflussmodellierung, um den Leiterplattenentwicklungsprozess und seine Schnittstellen zu anderen Fachdisziplinen zu modellieren und zu optimieren. Abschließend gibt Kapitel 5 ein Resümee und einen Ausblick.

2 Geschäftsprozessmodellierung

Innerhalb der Geschäftsprozessmodellierung haben sich unter anderem folgende grafische Modellierungssprachen etabliert:

- Business Process Model and Notation (BPMN) [BPMN2.0]
- Ereignisgesteuerte Prozesskette (EPK) [Sch99]
- Integrated DEfinition Method 3 (IDEF3) [MMP+95]
- Unified Modeling Language – Aktivitätsdiagramm [UML2.4]

Einen Schwerpunkt der Sprachen bildet die Prozessmodellierung. Die Betrachtung der Datenflüsse ist von untergeordneter Bedeutung. Die Business Process Model and Notation (BPMN) zeichnet sich durch eine ausgeprägte Werkzeugunterstützung und weite Verbreitung aus [All2009-ol, DGS+10, FR14]. Die Modellierungssprache ermöglicht die Geschäftsprozessmodellierung und grafische Darstellung von Arbeitsabläufen. Sie wurde ursprünglich von dem IBM Mitarbeiter S. White entwickelt [Whi04] und im Jahr 2004 von der Business Process Management Initiative (BPMI) veröffentlicht [BPMN1.0a]. Im Jahr 2006 wurde sie als ein offizieller Standard von der Object Management Group (OMG) veröffentlicht [BPMN1.0b]. Im Jahr 2011 verabschiedete die OMG die Version 2.0 [BPMN2.0] und im Jahr 2013 wurde die Version 2.0.1 zum internationalen Standard ISO/IEC 19510:2013 erhoben [ISO19510].

Die Modellierungssprache definiert Symbole für Aufgaben, Ereignisse und logische Verzweigungen sowie Verbindungen zwischen Symbolen zur Beschreibung von Sequenzflüssen und Assoziationen (siehe Bild 1). Im Folgenden werden nur Elemente der Modellierungssprache beschrieben, welche für die entwickelte Datenflussmodellierung relevanten sind. Die Texte basieren auf den Einführungen von [Har11] und [Goe11].

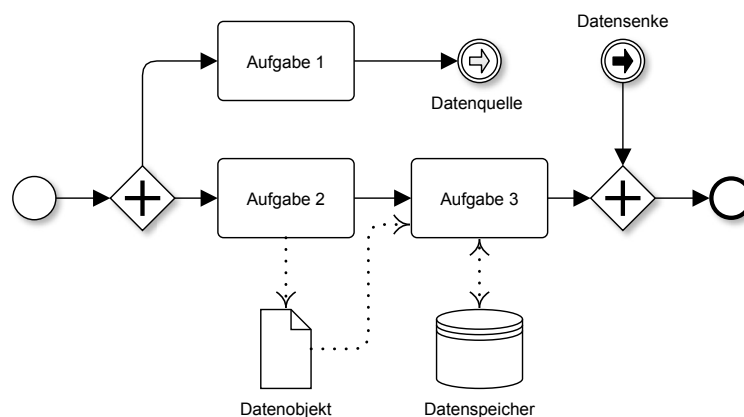


Bild 1: Beispiel einer Geschäftsprozessmodellierung

Aufgaben

Eine Aufgabe modelliert eine einzelne Tätigkeit in einem Entwicklungsprozess und wird durch ein abgerundetes Rechteck dargestellt. Ein Name innerhalb des Rechtecks beschreibt die Aufgabe und gibt Aufschluss über den Inhalt der Tätigkeit.

Sequenzflüsse

Ein Sequenzfluss bestimmt die Abhängigkeiten zwischen Aufgaben oder weiteren Elementen und wird durch einen durchgezogenen Pfeil symbolisiert. Der Sequenzfluss gibt dem Entwicklungsprozess eine Ablaufstruktur.

Ereignisse

Ein Ereignis tritt im Verlauf eines Prozesses auf und wird durch einen Kreis dargestellt. Generell werden drei Arten von Ereignissen unterschieden: Startereignisse, Zwischenereignisse und Endereignisse. Außerdem existieren verschiedene Spezialisierungen der Ereignisse.

- **Startereignisse**

Ein Startereignis steht am Anfang eines Prozesses oder Prozessabschnitts und wird durch einen Kreis mit dünner Randlinie dargestellt. Ein Startereignis hat keinen eingehenden Sequenzfluss.

- **Zwischenereignisse**

Ein Zwischenereignis kennzeichnet ein Ereignis im Verlauf einer Prozessausführung und wird durch einen Kreis mit doppelter dünner Randlinie dargestellt. Eine Spezialisierung des Zwischenereignisses modelliert einen Sprung im Prozessmodell. Dieser wird als Link bezeichnet und durch einen Pfeil gekennzeichnet. Es wird zwischen einer Quelle mit ausgefülltem Pfeil und einem Ziel mit nicht ausgefülltem Pfeil unterschieden.

- **Endereignisse**

Ein Endereignis steht am Ende eines Prozesses oder Prozessabschnitts und wird durch einen Kreis mit dicker Randlinie dargestellt. Ein Endereignis hat keinen ausgehenden Sequenzfluss.

Gateways

Ein Gateway ermöglicht eine Aufspaltung und Zusammenführung von Sequenzflüssen. Er beeinflusst den weiteren Verlauf eines Prozesses und wird durch eine Raute symbolisiert.

- **Exklusive Gateways**

Ein exklusiver Gateway dient der Darstellung von alternativen Flüssen und wird durch eine leere Raute dargestellt. Der Fluss genau eines eingehenden Sequenzflusses wird, basierend auf einer vordefinierten Bedingung, genau auf einen der ausgehenden Flüsse weitergeleitet.

- **Parallele Gateways**

Ein paralleler Gateway dient der Darstellung von parallelen Flüssen und wird durch eine Raute mit einem Pluszeichen in der Innenfläche symbolisiert. Es wird zwischen verzweigendem und zusammenführendem parallelem Gateway unterschieden. Der verzweigende parallele Gateway teilt einen Sequenzfluss in zwei oder mehr Flüsse auf. Alle ausgehenden Pfade eines parallelen Gateways werden unabhängig voneinander ausgeführt. Die ausgehenden Flüsse können so-

wohl gleichzeitig als auch in einer beliebigen Reihenfolge ausgeführt werden. Der zusammenführende parallele Gateway vereint zwei oder mehr Flüsse zu einem Sequenzfluss. Der ausgehende Sequenzfluss wartet, bis alle eingehenden Flüsse angekommen sind.

Datenobjekte

Ein Datenobjekt modelliert eine gespeicherte Information und wird durch eine Seite mit abgeknickter Ecke dargestellt. Ein Name unterhalb des Symbols beschreibt das Datenobjekt. Weiterhin wird durch einen ausgefüllten oder nicht ausgefüllten Pfeil ein Dateneingang oder Datenausgang spezifiziert.

Datenspeicher

Ein Datenspeicher symbolisiert ein Element, auf das Aufgaben lesend und schreiben zugreifen können und wird durch einen Zylinder mit zwei Linien am oberen Ende gekennzeichnet. Der Datenspeicher wird über einen Namen unterhalb des Elements benannt.

Datenassoziationen

Eine Datenassoziation verknüpft ein Datenobjekt mit Aufgaben oder Sequenzflüssen und wird durch einen gepunkteten Pfeil oder eine Linie symbolisiert.

3 Datenflussmodellierung

Im Gegensatz zur Geschäftsprozessmodellierung bildet die Datenflussmodellierung keine streng fortlaufende Abfolge von Entwicklungsaufgaben ab. Der Schwerpunkt liegt in der Modellierung der Datenflüsse und somit der Schnittstellen zwischen einzelnen Entwicklungsaufgaben. Unsere Datenflussmodellierung basiert auf einer Untermenge der oben vorgestellten Modellierungssprache. Diese wurde durch Einschränkungen und Anpassungen an das Modellierungsziel adaptiert (siehe Bild 2). Dies soll die Komplexität der Diagramme reduziert, die Gestalt der Diagramme vereinheitlicht und die Anwendung der Modellierungssprache vereinfachen.

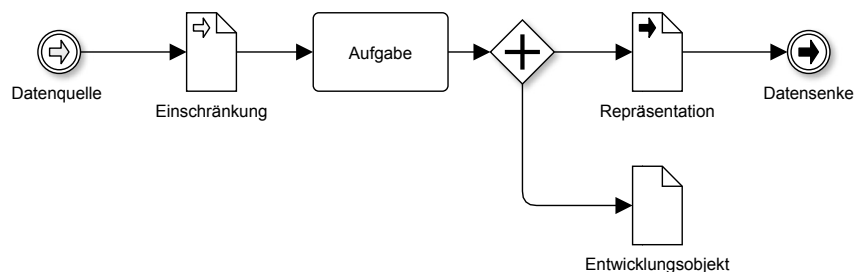


Bild 2: Beispiel einer Datenflussmodellierung

Aufgaben

Eine Aufgabe beschreibt einzelne Entwicklungsaufgaben. Die Granularität sollte sich an dem Austausch von Datenobjekten orientieren. Im Unterschied zum Modellierungsstan-

dard wird die Anzahl eingehender und ausgehender Sequenzflüsse von Aufgaben jeweils auf eins reduziert.

Sequenzflüsse

Ein Sequenzfluss beschreibt den Datenfluss zwischen Aufgaben in einem Entwicklungsprozess ohne Rekursionen. Der Datenfluss ist zwar unidirektional, auf Grund von Wechselwirkungen im Prozess existiert aber eine parallele bidirektionale Abstimmung zwischen den Aufgaben. Datenobjekte werden nicht einmalig ausgetauscht, sondern entstehen im Rahmen der Entwicklung sukzessive zwischen den in Beziehung stehenden Aufgaben. Rücksprünge zu einzelnen Aufgaben sind jederzeit möglich. Im Fall einer Änderung eines Datenobjektes geht der Rücksprung einher mit einer wiederholten Ausführung des gesamten restlichen Entwicklungsprozesses. Weiterhin erfolgt keine explizite Modellierung von Konfliktlösungen. Diese Einschränkung basiert auf der Annahme, dass Konflikte in einer Entwicklungsaufgabe nur durch die Anpassung eines eingehenden Datenobjektes gelöst werden können. Die Konfliktlösung erfolgt somit durch eine inverse Interpretation der eingehenden Sequenzflüsse.

Ereignisse

In der Datenflussmodellierung wird nur der Link verwendet. Er kennzeichnet externe Datenquellen und -senken. Datenquellen werden mittels eines nicht ausgefüllten Pfeils und Datensenzen mittels eines ausgefüllten Pfeils symbolisiert.

Datenobjekte

Die Datenflussmodellierung unterscheidet drei Arten von Datenobjekten im Entwicklungsprozess:

- Entwicklungsobjekte werden direkt von einer Aufgabe zu einer anderen Aufgabe transferiert. Die Informationen werden in der Regel in Form von Modellen transferiert.
- Repräsentationen werden aus Entwicklungsobjekten abgeleitet und dienen dem Datenaustausch zwischen verschiedenen Fachdisziplinen. Der Austausch erfolgt meist über ein Datenaustauschformat und die enthaltenen Informationen sind entweder reduziert oder um weitere Informationen aus der empfangenden Fachdisziplin angereichert. Sie werden durch einen ausgefüllten Pfeil symbolisiert.
- Einschränkungen beeinflussen den Lösungsraum einer anderen Entwicklungsaufgabe und werden oft dokumentenorientiert ausgetauscht. Sie werden durch einen nicht ausgefüllten Pfeil symbolisiert und dienen der expliziten Modellierung von direkten Wechselwirkungen zwischen Entwicklungsaufgaben.

Im Unterschied zum Modellierungsstandard wird in unserer Modellierung auf gerichtete Datenassoziationen verzichtet. Stattdessen werden die Datenflüsse mittels Sequenzflüssen modelliert. Diese Einschränkung basiert auf der Annahme, dass die Sequenzflüsse zwischen Entwicklungsaufgaben auf dem Austausch von Datenobjekten basieren und nicht willkürlich sind. Außerdem wird auf eine Datenassoziation zwischen Datenobjekt und Sequenzfluss verzichtet und das Datenobjekt direkt in den Sequenzfluss eingefügt.

Dieses reduziert die Komplexität der Diagramme und ermöglicht die Verwendung von einfachen Graphen und entsprechenden Softwarewerkzeugen, wie z. B. *yEd* von yWorks [yEd-ol]. Auf eine direkte Einfügung der Datenobjekte in die Sequenzflüsse kann zugunsten einer Kompatibilität mit dem Modellierungsstandard verzichtet werden.

4 Beispiel Leiterplattenentwicklung

Die entwickelte Datenflussmodellierung wird genutzt, um die Abhängigkeiten und Wechselwirkungen in einem Entwicklungsprozess zu identifizieren und den Entwicklungsprozess zu optimieren. In einem ersten Schritt werden die Datenflüsse in einem existierenden Entwicklungsprozess modelliert. Aufbauend auf diesem Modell wird der Prozess angepasst und erweitert.

Der Einsatz der Datenflussmodellierung wird im Folgenden am Beispiel der Leiterplattenentwicklung demonstriert. Die Leiterplattenentwicklung beginnt nach dem interdisziplinären Systementwurf. Sie lässt sich in die Phasen Entwurf, Produktion, Analyse und Validierung unterteilen und beginnt mit dem Leiterplattenentwurf. Auf Basis des Leiterplattenentwurfs erfolgt eine prototypische Fertigung und Bauteilbestückung der Leiterplatte. Die bestückte Leiterplatte wird einer Inbetriebnahme und verschiedenen Tests unterzogen. Dabei wird der gesamte Prozess oder Teilprozess rekursiv ausgeführt. Das folgende Beispiel konzentriert sich auf den Leiterplattenentwurf, da dieser die meisten Wechselwirkungen zu anderen Fachdisziplinen aufweist.

Der klassische Leiterplattenentwurf umfasst die Aufgaben Blockschaltbild-, Schaltplan- und Layoutentwurf. In der Literatur wird hauptsächlich der Layoutentwurf detailliert [Coo08, HAB+09]. Der Übergang von einem interdisziplinären Systementwurf zum Leiterplattenentwurf wird dabei nicht betrachtet. Häufig wird sogar auf das Blockschaltbild verzichtet und der detaillierte Schaltplan bildet den Einstiegspunkt in den Leiterplattenentwurf. Wechselwirkungen mit anderen Fachdisziplinen werden nicht betrachtet. Unsere Modellierung eines angepassten Leiterplattenentwurfsprozess zeigt neben den Wechselwirkungen innerhalb der Elektrotechnik auch Wechselwirkungen mit dem interdisziplinären Systementwurf, der Informationstechnik und dem Maschinenbau (siehe Bild 3).

Eine Wechselwirkung zum interdisziplinären Systementwurf besteht über die Systemarchitektur. Sie beeinflusst die Bauteilauswahl, Leiterplattenbestückungs- und Leiterplattenfertigungstechnologie. Außerdem steht der Auswahlprozess in starker Wechselwirkung mit anderen fachfremden Entwicklungsaufgaben. Beispielsweise muss die Informationsverarbeitung in Abstimmung mit der Softwaretechnik oder die Sensorik und Leistungselektronik in Abstimmung mit der Regelungstechnik ausgewählt werden. Auch der Leiterplattenbauraum beeinflusst die Bauteilauswahl. In unserem Leiterplattenentwurfsprozess haben wir daher die Systemelemente in der interdisziplinären Systemarchitektur um Einschränkungen erweitert und das nicht genau spezifizierte Blockschaltbild durch eine Komponentenarchitektur ersetzt. Mittels Einschränkungen wird

der Auswahlprozess von fachfremden Entwicklungsaufgaben beeinflusst. Die Komponentenarchitektur ordnet jedem Funktionsträger der Leiterplatte eine Komponente zu. Eine Komponente wird selbst über einen Teilschaltplan spezifiziert. Der Teilschaltplan enthält z. B. einen integrierten Schaltkreis und dessen komplette Beschaltung aus passiven Bauteilen.

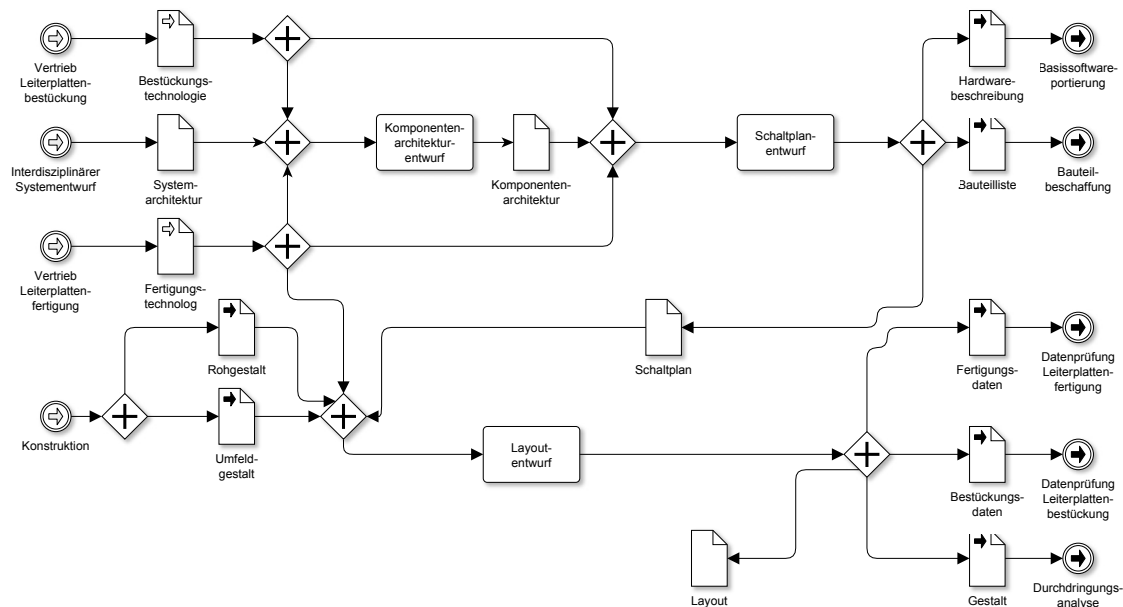


Bild 3: Vereinfachte Datenflussmodellierung eines angepassten Leiterplattenentwurfsprozesses

Eine Wechselwirkung zur Informationstechnik besteht über den Schaltplan. Die Bauteilstruktur sowie Bauteil- und Anschlusskonfigurationen werden für die Entwicklung der Basisssoftware bzw. des Betriebssystems benötigt. Gleichzeitig beeinflussen die Konfigurationsmöglichkeiten in der Informationstechnik den Entwurf des Schaltplans. Der Datenaustausch erfolgt zurzeit manuell. Erschwerend hierbei ist die Tatsache, dass einige dieser Informationen nur implizit im Schaltplan spezifiziert werden. Unser Leiterplattenentwurfsprozess wurde daher um eine Hardwarebeschreibung erweitert. Diese beinhaltet unter anderem eine Bauteileliste sowie die Bauteil- und Anschlusskonfiguration für die Hardwaretreiber.

Eine Wechselwirkung zum Maschinenbau existiert über das Layout der Leiterplatte. Insbesondere die Leiterplattenrohgestalt und Bauteilplatzierung beeinflussen die mechanische Konstruktion des Systems. Einerseits werden Gehäusegestalt und Verbindungstechnik spezifiziert und andererseits elektrische Bauteile in ihrer zweidimensionalen Gestalt platziert. Ein Datenaustausch zwischen den Softwarewerkzeugen der Fachdisziplinen wird bereits unterstützt und es existieren Forschungen hinsichtlich einer gemeinsamen Entwurfsumgebung [DVP+13, SNK+14]. Unser Leiterplattenentwurfsprozess wurde entsprechend um die Rohgestalt, Umfeldgestalt und Gestalt der Leiterplatte erweitert.

5 Resümee und Ausblick

Im Rahmen dieser Arbeit wurde die von uns entwickelte Datenflussmodellierung vorgestellt. Diese basiert auf einer Untermenge der Modellierungssprache für Prozessdiagramme des Standards „Business Process Model and Notation (BPMN)“. Der Modellierungsstand wurde durch Einschränkungen und Anpassungen an das Modellierungsziel angepasst. Der Einsatz der Datenflussmodellierung wurde am Beispiel der Leiterplattenentwicklung demonstriert und ein Ausschnitt eines angepassten Leiterplattenentwurfsprozesses vorgestellt. Allein dieser Ausschnitt weist schon einige Wechselwirkungen zu anderen Fachdisziplinen auf und widerspricht damit der verbreiteten Annahme einer getrennten fachdisziplinspezifischen Entwicklung. Dieser Widerspruch bestätigt die Notwendigkeit einer Untersuchung der Datenflüsse in Entwicklungsprozessen. Insbesondere historisch gewachsene und lineare Entwicklungsprozesse bedürfen einer Überprüfung. Die entwickelte Datenflussmodellierung unterstützt die Analyse und anschließende Optimierung der Entwicklungsprozesse.

Nächste Schritte sind die Anwendung der Datenflussmodellierung auf weitere Entwicklungsprozesse und die Diskussion der Entwicklungsprozesse mit Experten der einzelnen Fachdisziplinen. Ein verstärkter Einsatz der Modellierung sollte helfen, diese weiter zu optimieren. Zudem soll der entwickelte Leiterplattenentwicklungsprozess weiter detailliert und optimiert werden.

Danksagung

Dieses Forschungs- und Entwicklungsprojekt wird mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) im Rahmen des Spitzenclusters „Intelligente Technische Systeme OstWestfalenLippe“ (it’s OWL) gefördert und vom Projektträger Karlsruhe (PTKA) betreut. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.

Literatur

- [All09-ol] ALLWEYER, T.: BPMN setzt sich durch in der Praxis. <http://www.computerwoche.de/a/1886445>, 2009
- [BPMN1.0a] BUSINESS PROCESS MANAGEMENT INITIATIVE (BPMI): Business Process Modeling Notation (BPMN) – Version 1.0. 3. Mai 2004
- [BPMN1.0b] OBJECT MANAGEMENT GROUP (OMG): Business Process Modeling Notation (BPMN) – Version 1.0. 1. Februar 2006
- [BPMN2.0] OBJECT MANAGEMENT GROUP (OMG): Business Process Model and Notation (BPMN) – Version 2.0. 3. Januar 2011
- [Coo+08] COOMBS, C. F.: Printed Circuits Handbook. 6th Edition, McGraw-Hill, 2008
- [DGS+10] DRAWEHN, J.; GAYER, S.; SCHNEIDER, P.; SPATH, D.; WEISBECKER, A.: Business Process Modeling 2010 – Modellierung von ausführbaren Geschäftsprozessen mit der Business Process Modeling Notation. Fraunhofer Verlag, Stuttgart 2010

- [DVP+13] DANKOVIC, D.; VRACAR, L.; PRIJIC, A.; PRIJIC, Z.: An Electromechanical Approach to a Printed Circuit Board Design Course. IEEE Transactions on Education, Vol. 56, No.4, 2013
- [FR14] FREUND, J.; RÜCKER, B.: Praxishandbuch BPMN 2.0. 4. Auflage, Carl Hanser Verlag München Wien, 2014, ISBN 978-3-446-44255-9
- [Goel11] GÖTZ, M.: BPMN 2.0 Tutorial – Kompakte Einführung in die BPMN 2.0. iTransparent GmbH | Business Process Architects, Bayreuther Str. 31, 90409 Nürnberg, Juli 2011
- [HAB+09] HOLDEN, H.; ANDRESAKIS, J.; BOGATIN, E.; CARANO, M.; CARPENTER, K.; DIETZ, K. H.; LAING, M.; VAUCHER, C.; VIKLUND, P.; WUENSCH, M.: The HDI Handbook. BR Publishing Inc., Oregon, USA, 2009
- [Har11] HARDER, A.: Einführung in die Prozessdarstellung mit BPMN (Business Process Modelling Notation). Universität Stuttgart, April 2011
- [ISO19510] ISO/IEC 19510:2013: Information technology – Object Management Group Business Process Model and Notation. Internationale Organisation für Normung (ISO), 1. Juli 2013
- [MMP+95] MAYER, R.; MENZEL, C.; PAINTER, M.; DEWITTE, P.; BLINN, T.; PERAKATH, B.: Information Integration for concurrent Engineering (IICE) – IDEF3 Process Description Capture Methode Report. <http://www.idef.com>, September 1995
- [Sch99] SCHEER, A.-W.: ARIS – Business Process Modelling. Springer Verlag, 1999
- [SNK+14] SON S., NA S., KIM K., LEE S.: Collaborative design environment between ECAD and MCAD engineers in high-tech products development. International Journal of Production Research, Vol. 52, Iss. 20, 2014
- [UML2.4] OBJECT MANAGEMENT GROUP (OMG): Unified Modeling Language (UML) – Version 2.4.1. 5. August 2011
- [Whi04] WHITE, S.: Process Modeling Notations and Workflow Patterns. IBM Corporation, 2004
- [yEd-ol] yWorks: yEd Graph Editor. Tübingen, <http://www.yworks.com/en/products/yfiles/yed/>

Autoren

Dipl.-Ing. Stefan Herbrechtsmeier ist seit 2010 wissenschaftlicher Mitarbeiter in der Arbeitsgruppe Kognitronik und Sensorik am CITEC, Universität Bielefeld. Von 2002 bis 2007 studierte er Ingenieurinformatik an der Universität Paderborn. Ab 2007 war er wissenschaftlicher Mitarbeiter in der Fachgruppe Schaltungstechnik am Heinz Nixdorf Institut, Universität Paderborn.

Dr.-Ing. Thorsten Jungeblut ist akademischer Rat in der Arbeitsgruppe Kognitronik und Sensorik am CITEC, Universität Bielefeld. Im Jahr 2011 promovierte er mit seiner Arbeit „Entwurfsraumexploration ressourceneffizienter VLIW-Prozessoren“. Zuvor war er ab 2005 wissenschaftlicher Mitarbeiter in der Fachgruppe Schaltungstechnik am Heinz Nixdorf Institut, Universität Paderborn.

Dr.-Ing. Mario Pormann ist Akademischer Direktor in der Arbeitsgruppe Kognitronik und Sensorik am CITEC, Universität Bielefeld. Im Jahr 2001 promovierte er mit seiner Arbeit zur „Leistungsbewertung eingebetteter Neurocomputersysteme“ am Heinz Nixdorf Institut, Universität Paderborn. Von 2001 bis 2009 war er Akademischer Ober-

rat und von 2010 bis 2011 Vertretungsprofessor am Fachgebiet Schaltungstechnik, Universität Paderborn.

Modellierung und Simulation mechatronischer

Integrierte Entwicklung komplexer Systeme mit modellbasierter Systemspezifikation und -simulation

Jürgen Roßmann, Michael Schluse, Malte Rast, Martin Hoppen
RWTH Aachen, Institut für Mensch-Maschine-Interaktion, Ahornstr. 55,
52074 Aachen, Tel. +49 (0) 241 / 80 26 101, E-Mail: rast@mmi.rwth-aachen.de

Roman Dumitrescu, Christian Bremer, Michael Hillebrand
Fraunhofer Projektgruppe Entwurfstechnik Mechatronik
Zukunftsmeile 1, 33102 Paderborn, Tel. +49 (0) 5251 / 54 65 173

Oliver Stern, Florian Blümel
RIF Institut für Forschung und Transfer e.V.
Joseph-von-Fraunhofer Str. 20, 44227 Dortmund, Tel. +49 (0) 231 / 97 00 101

Christoph Averdung
CPA Redev GmbH
Auf dem Seidenberg 3a, 53721 Siegburg, Tel. +49 (0) 2241 / 25 94 213

Zusammenfassung

Entwicklungsprojekte im Bereich der Raumfahrtrobotik sind geprägt durch die tiefgreifende Integration unterschiedlichster Fachdisziplinen und die damit einhergehende Komplexität der zu entwickelnden Systeme. Zur Unterstützung der multidisziplinären Zusammenarbeit bietet das Paradigma des Model-Based Systems Engineering (MBSE) einen geeigneten Ansatz. Eine weitere wesentliche Herausforderung bei Raumfahrtprojekten stellt die frühzeitige Validierung und die Verifikation des modellbasierten Entwurfs dar. Tests mit realen Prototypen sind mit erheblichen Kosten verbunden und können die tatsächlichen Gegebenheiten oft nur eingeschränkt abbilden (Mikrogravitation, Strahlung, Startbelastung, etc.). Es werden daher Virtuelle Testbeds eingesetzt, in denen Teile oder das gesamte System simuliert, validiert und verifiziert werden können. Ziel des Forschungsprojekts INVIRTES ist die Integration der disziplinübergreifenden Systemspezifikation mit der disziplinübergreifenden Testkonzeption mittels Virtueller Testbeds zu einem ganzheitlichen Systemmodell. Dadurch kann ein durchgehendes simulationsbasiertes Engineering im Entwurf von Raumfahrtanwendungen erreicht werden. Somit wird sichergestellt, dass die Anforderungen aus der frühen Entwurfsphase systematisch zu Test und Integration der Detailmodelle aus den verschiedenen Fachdisziplinen genutzt werden können. Hierzu ist eine gemeinsame Datenbasis mit einer Systemmodellverwaltung nötig, die das verteilte Arbeiten auf einer gemeinsamen Datengrundlage ermöglicht. Das Systemmodell beschreibt das System in unterschiedlichen Varianten und Ausschnitten in seiner Gesamtheit. Das INVIRTES-Systemmodell bietet

das Potenzial, das in nationalen und internationalen Vorhaben gewonnene Know-How effizient zusammenführen zu können, um die Entwicklungsschritte einer Mission von der Analyse über Entwurf, Visualisierung, Steuerung, Bedienung, Training bis hin zu Marketing und Technologietransfer konsequent unterstützen zu können. Damit können zukünftige Projekte preiswerter, robuster und schneller realisiert werden. Dieses Konzept wird mit dem vorliegenden Beitrag vorgestellt.

Schlüsselworte

Raumfahrtrobotik, PLM/PDM, Model-Based Systems Engineering, Virtuelle Testbeds, Requirements Engineering

Integrated Development of Complex Systems with Model-Based System Specification and Simulation

Abstract

R&D projects in space robotics are characterized by the profound integration of various disciplines and the associated complexity of the systems to be designed. To support this multidisciplinary collaboration, the paradigm of Model-Based Systems Engineering (MBSE) is a suitable approach. Another essential challenge of space projects is the early verification and validation of the model-based design. Tests with physical prototypes give rise to considerable costs and can reproduce the real conditions only to a certain extent (microgravity, radiation, launch stresses etc.). Therefore, Virtual Testbeds are used to simulate and validate parts of the overall system. The objective of the research project INVIRTES is the integration of the cross-domain system specification with the cross-domain test design with Virtual Testbeds to a comprehensive system model. This enables a continuous simulation-based engineering in the design of space applications. Requirements from the early design phase can be used systematically to test and integrate detailed models from the various disciplines. This requires a common data base with a system model management for distributed and parallel collaboration. The system model describes the system in various alternative layouts and sections in its entirety. The INVIRTES system model offers the potential to efficiently integrate national and international know-how to support the development phases of a mission from the requirements analysis to design, visualization, control, operation, training up to marketing and technology transfer. Future projects can be realized faster, less expensive and more robust. This concept is presented in this contribution.

Keywords

Space Robotics, PLM/PDM, Model-Based Systems Engineering, Virtual Testbeds, Requirements Engineering

1 Einleitung

Autonome und intelligente robotische Systeme sind Schlüsseltechnologien für die Erkundung des Weltraums. Ferner ist die Raumfahrtrobotik eine „Sprungbretttechnologie“ für die Anwendbarkeit von Weltraumentwicklungen auf der Erde, die damit zugleich weitreichende terrestrische Anwendungsfelder und globale Zukunftsmärkte erschließt.

Entwicklungsprojekte im Bereich der Raumfahrtrobotik sind geprägt durch die tiefgreifende Integration unterschiedlichster Fachdisziplinen und die damit einhergehende Komplexität der zu entwickelnden Systeme. Zur Unterstützung der multidisziplinären Zusammenarbeit bietet das Paradigma des Model-Based Systems Engineering (MBSE) einen geeigneten Ansatz. Eine weitere wesentliche Herausforderung bei Raumfahrtprojekten stellt die frühzeitige Validierung und die Verifikation des modellbasierten Entwurfs dar. Tests mit realen Prototypen sind mit erheblichen Kosten verbunden und können die tatsächlichen Gegebenheiten nur eingeschränkt abbilden (Mikrogravitation, Strahlung, Startbelastung, etc.). Es werden daher Virtuelle Testbeds (VTB) eingesetzt, in denen Teile oder das gesamte System simuliert, validiert und verifiziert werden können.

Der Übergang von der frühzeitigen Modellierung im Sinne des MBSE zur Detailmodellierung mit dem Ziel der Simulation ist von manueller Arbeit geprägt. Da im Laufe der Entwicklung, insbesondere durch Testsimulationen, fortwährend weitere Informationen generiert werden, ist eine Konsistenzerhaltung zwischen den Modellen nur schwer möglich. Diese ist aber notwendig, da ein iterativer Entwicklungsprozess Änderungen in unterschiedlichen Ebenen der Systembeschreibung erzeugt. Es bedarf eines Ansatzes zur integrativen Modellierung des zu entwickelnden Systems.

Ziel des Forschungsprojektes INVIRTES ist die Integration der disziplinübergreifenden Systemspezifikation mit der disziplinübergreifenden Testkonzeption mittels Virtueller Testbeds. Dadurch kann ein durchgehendes, simulationsbasiertes Engineering im Entwurf von Raumfahrtanwendungen erreicht werden. Somit wird sichergestellt, dass die Anforderungen aus der frühen Entwurfsphase systematisch zu Test und Integration der Detailmodelle aus den verschiedenen Fachdisziplinen genutzt werden können. Hierzu ist eine gemeinsame Datenbasis mit einer Systemmodellverwaltung nötig, die das verteilte Arbeiten auf einer gemeinsamen Datengrundlage ermöglicht (Kapitel 2). Das Systemmodell beschreibt das System in unterschiedlichen Varianten und Ausschnitten in seiner Gesamtheit und bildet somit die Datengrundlage für den Entwurf. Die Datenbasis (Kapitel 3) ermöglicht die Integration von Systemspezifikation und disziplinspezifischen Detailmodellen (bspw. CAD, FEM, semantische Umfeldmodell) bis hin zur virtuellen Inbetriebnahme. Daneben bedarf es einer geeigneten Vorgehenssystematik (Kapitel 4), die in eine durchgehende und benutzungsfreundliche Werkzeugunterstützung eingebettet ist. Für Raumfahrtvorhaben gilt, dass das flugfähige Gesamtsystem nicht final verifiziert werden kann; die eigentliche Mission ist der erste Gesamtsystemtest. Durch die Gesamtsystems simulation (Kapitel 5) in Virtuellen Testbeds kann demgegen-

über ein maßgebliches „notwendiges“ Kriterium („Wenn ein System nicht im VTB funktioniert, dann auch nicht real!“) für die Beurteilung der Funktionsfähigkeit der Systeme bereitgestellt werden. In Kapitel 6 wird der praktische Nutzen am Beispiel realer Anwendungen untersucht.

2 INVIRTES-Systemarchitektur

Bild 1 zeigt die INVIRTES-Systemarchitektur. Ein Systemmodell wird von einer zentralen nD-Datenhaltungskomponente (hier vom objekt-orientierten SupportGIS Java (SGJ)) verwaltet. SGJ muss dazu ein übergreifendes INVIRTES-Schema importieren, welches die notwendigen Schemata der zu modellierenden Anwendungen (siehe z.B. Kapitel 6) und Entwicklungsstufen (Systemspezifikation, Detailmodell, ...) integriert. Auf die Datenhaltung können nun Software-Werkzeuge zugreifen, das Datenbank-Schema mit ihrem internen Schema abgleichen und Daten austauschen. Ein Systemmodell kann dann mittels verschiedener Software-Werkzeuge entwickelt werden, ohne Konvertierungen und Medienbrüche. Im Rahmen von INVIRTES erfolgt die Systemspezifikation mit dem auf Eclipse basierenden Papyrus nach der CONSENS-Spezifikationstechnik („CONceptual design Specification technique for the ENgineering of complex Systems“, [GLL12]). Die Gesamtsystemsimulation zur Analyse, Bewertung und Optimierung eines Systems erfolgt mit der Simulationssoftware VEROSIM [RSS+13]. Dazu wird die Systemspezifikation zu einem experimentierbaren Modell ausdetailliert (siehe Abschnitt 4.2).

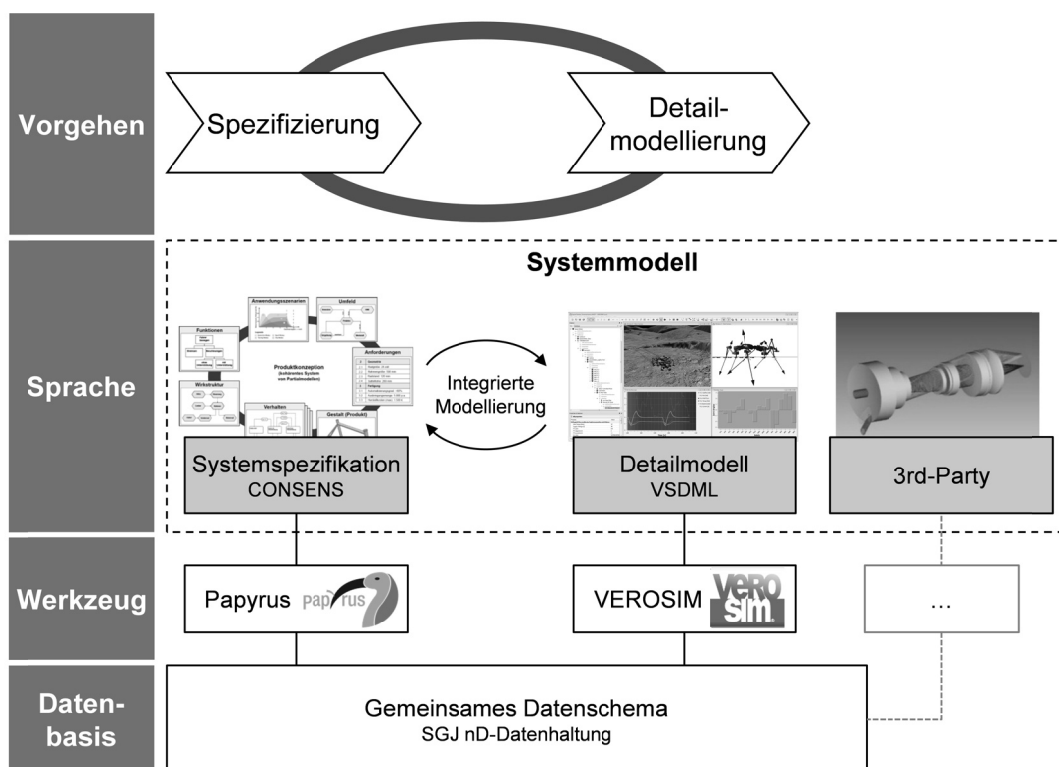


Bild 1: INVIRTES-Systemarchitektur

2.1 Systemmodell

Das Systemmodell enthält neben einer anwendungsübergreifenden Systemspezifikation die Geometrie, sämtliche relevanten Kenn- und Zustandsgrößen, die Beschreibung interner und externer Schnittstellen zwischen den Komponenten, Programme von Steuerungskomponenten (z. B. für Roboter oder Speicherprogrammierbare Steuerungen), die Konfiguration vordefinierter Simulationsalgorithmen bzw. die Beschreibung benutzerspezifischer Simulationsalgorithmen sowie weitere für den Simulations- und Entwicklungsprozess wesentliche Informationen. Durch diese umfassende Beschreibung des Systems eignet sich das Systemmodell nicht nur als Grundlage für die Simulation sondern ist darüber hinaus ideal zur Strukturierung der Entwicklungen in komplexen Entwicklungsvorhaben geeignet, da es über ein gemeinsames Datenschema und gemeinsam verwaltete Daten ein gemeinsames Systemverständnis definiert.

3 Aktive, echtzeitfähige nD-Datenhaltung

Für eine Speicherung des Systemmodells in persistenten Datenstrukturen kann grundsätzlich auf sogenannte Standard-Datenbankmanagementsysteme (DBMS) zurückgegriffen werden. Sie verfügen von Haus aus über geeignete Mechanismen zur transaktions-sicheren Verwaltung von Daten mit und ohne Raumbezug. Zugleich wird über die hier verwendete Datenbanksprache SQL die weitgehende Unabhängigkeit einer Anwendung vom eingesetzten DBMS erreicht.

Mithilfe dieser Systemumgebungen zur Speicherung der Daten des Systemmodells werden jedoch nur die formalen Voraussetzungen zur Herstellung einer Persistenzschicht geschaffen. In der Folge gilt es, die darüber hinausgehenden Anforderungen an die datenmäßige Führung des Systemmodells im Hinblick auf die Eigenschaften *aktiv*, *echtzeitfähig*, *mehrdimensional* und *temporal* mithilfe dieser Basistechnologie umzusetzen.

Im Zentrum der Betrachtung steht dabei die Anforderung an die Hersteller-unabhängigkeit bei der Beschreibung und Implementation des Systemmodells. Umgesetzt werden diese Anforderungen durch den konsequenten Verzicht auf proprietäre Entwicklungen in Bezug auf den Aufbau der Datenstrukturen innerhalb des DBMS sowie durch die Verwendung von SQL als Datenbanksprache auch für die Realisierung des Zeitbezugs innerhalb des Systemmodells.

Als Schnittstelle zwischen der Beschreibung eines Systemmodells und den verwendeten DBMS kommt innerhalb des Forschungsvorhabens INVIRTES die SupportGIS-Technologie der CPA zum Einsatz. Sie gestattet die Transformation von beispielsweise in der Unified Modeling Language (UML) beschriebenen Systemmodellen in ISO 19100-konforme Datenbankstrukturen und bildet über die sprachlichen Mittel von Standard-SQL die temporalen Eigenschaften der Objekte, Relationen und Attribute innerhalb der Tabellenstrukturen eines objektrelationalen DBMS ab. Ein wesentlicher Vorteil der SupportGIS-Technologie ist dessen Eigenschaft zur deklarativen (beschreibenden)

Abbildung der Systemmodelle in objektrelationalen Datenbanktabellen. Diese Vorgehensweise erlaubt die Verwaltung einer nahezu beliebigen Anzahl von Systemmodellen in den Datenstrukturen des DBMS. Entsprechend lassen sich die weiteren Attribute wie *aktiv*, *mehrdimensional* und *temporal* ebenso generisch wie abstrakt mit den objektorientierten Eigenschaften eines Systemmodells verknüpfen, so dass sich in letzter Konsequenz die allgemein gültige Grundlage für eine aktive, echtzeitfähige nD-Datenhaltung der Systemmodelle ergibt.

Dabei steht insbesondere der Zeitbezug eines Systemmodells im Mittelpunkt des Interesses. Zeitbezüge lassen sich hier in mannigfaltiger Art und Weise identifizieren. Zum Beispiel in Form von verschiedenen Versionen eines gesamten oder von Teilen des Systemmodells, als Varianten einzelner Komponenten des Systemmodells oder bei der echtzeitfähigen Aufzeichnung diskreter Zustandsveränderungen in Virtuellen Testbeds. Entsprechend der durchzuführenden Aufgabe wechselt die Betrachtungsweise des Systemmodells innerhalb von INVIRTES von geringdynamisch nach hochdynamisch bis hin zu einer dann individuell zu spezifizierenden Echtzeitfähigkeit. Die dazu notwendigen Performanceeigenschaften können die Verwendung von sich über den Kernspeicher eines Computers abzubildenden transienten Speicherstrukturen erforderlich machen.

4 Vorgehenssystematik

Zur Erarbeitung des Systemmodells bedarf es einer Vorgehenssystematik, die das Vorgehen des Entwicklers strukturiert. Im Folgenden werden die Schritte hin zur Systemspezifikation und zum Detailmodell näher erläutert.

4.1 Systemspezifizierung

Ziel der Systemspezifizierung ist ein Spezifikationsmodell, das das zu entwickelnde System im Sinne des Model-Based Systems Engineering disziplinübergreifend beschreibt. Dieses Modell dient der Kommunikation und Kooperation zwischen den Entwicklungsbeteiligten, ermöglicht die Beherrschung der Komplexität und unterstützt das Problemverständnis [HWF+12], [MK11-ol]. Es dient außerdem als Grundlage für die anschließende Detaillierung. Die Spezifikationstechnik CONSENS [GLL12] sowie andere auf der SysML basierenden Vorgehen wie etwa SYSMOD [Wei06] oder OOSEM [FSM08] entspringen diesem Verständnis. Keiner der genannten Ansätze adressiert jedoch die integrierte Konzipierung von Testfällen. Es besteht der Bedarf nach einem Vorgehen, mit dem frühzeitig Testfälle beschrieben werden. Im Rahmen von INVIRTES wird CONSENS als Grundlage gewählt. Hierbei werden die Aspekte Umfeld, Anwendungsszenarien, Anforderungen, Funktionen, Wirkstruktur und Verhalten erarbeitet. Die Spezifikationstechnik zielt auf die Beschreibung mechatronischer Systeme und ist auch für die in dieser Arbeit betrachteten Systeme geeignet.

Bei der Konzipierung von Testfällen gilt es, sicherzustellen, dass alle Anforderungen, die an das System gestellt werden, erfüllt sind und die Systemfunktionen in den jeweiligen Anwendungsszenarien abgesichert sind. Hierzu wird jede definierte Anforderung mit einem Testfall assoziiert, der diese validiert. Dabei wird jedoch übersehen, dass die Anforderungen häufig nicht alle zu testenden Einzelheiten enthalten, weil diese erst im Laufe der Entwicklung festgelegt werden und häufig nicht in den Anforderungen nachgepflegt werden. Dazu kommt die generelle Frage, mit welcher Konstellation an Test-szenarien ein System möglichst effizient und dennoch hinreichend getestet wird.

Daher werden auszuführende Testszenarien in ergänzenden CONSENS-Modellen erarbeitet. Dazu werden vor allem die Aspekte Umfeld, Anwendungsszenarien, Wirkstruktur und Verhalten benötigt. Mit zunehmendem Entwicklungsstand sind verschiedene Testszenarien zum Nachweis der Funktionsfähigkeit in der Raumfahrtanwendung zu adressieren. Hierfür hat sich das Konzept der Technologiereifegrade (TRL) etabliert [ISO16290]. Dieses Prinzip gilt als Rahmen für die Vorgehenssystematik und wird für die Anwendung in Virtuellen Testbeds adaptiert.

Um sicherzustellen, dass die System- und Testspezifikation plausibel und vergleichbar ist, werden Modellierungsrichtlinien und –überprüfungen eingeführt. So kann etwa überprüft werden, ob die energetische oder stoffliche Beschreibung den Regeln der Energie- und Stoffhaltung genügen. Zudem kann bspw. ermittelt werden, welche Schnittstellen des System innerhalb der Testszenarien getestet werden oder ob allen Funktionen Testfälle zugeordnet sind.

4.2 Detailmodellierung

Die Detailmodellierung beschreibt den Prozess der Detaillierung einer abstrakten Systemspezifikation hin zu einem konkreten Modell des Systems. Während also das Ergebnis einer Systemspezifikation noch keine Details für die konkrete Wahl bzw. Umsetzung der beteiligten Komponenten liefert, werden diese im Zuge der Detailmodellierung ergänzt. Spricht man also beispielsweise in der Systemspezifikation noch allgemein von Komponenten wie einer Sensorik, die mit einer Navigationskomponente verbunden ist, so wird in der Detailmodellierung ganz konkret beschrieben, welcher Navigationsalgorithmus mit welchen Sensoren genutzt wird (Geometrie, Parameter, usw.).

Zu diesem Zwecke wird eine geeignete allgemeine Modellierungssystematik benötigt. Im Rahmen von INVIRTES sollen dazu die flexibel erweiterbaren Modellierungsstrukturen der Simulationsdatenbank VSD (VEROSIM Active Simulation Database) Verwendung finden. Sie bietet einen grafenbasierten Modellierungsansatz mit Modellelementen u.a. zur Modellierung von Geometrie, Kinematik, Mehrkörperdynamik oder Ein-/Ausgabe-Netzwerken. Durch ihre flexiblen Metastrukturen kann die VSD applikationsspezifisch erweitert werden und beliebige Fachschemata abbilden. Die Detailmodellierung kann automatisiert, werkzeuggestützt oder manuell erfolgen. Beispiele sind die Generierung physikalischer Ersatzmodelle für die Starrkörpersimulation aus CAD-

Daten oder die Ableitung von ausführbaren SOML-Netzen (State Oriented Modeling Language, [FS04]) aus Zustandsdiagrammen der Spezifikation. Wichtig ist hierbei, dass agile Vorgehensweise und paralleles Arbeiten explizit unterstützt werden, so ist bei nachträglicher Anpassung der Spezifikation keine erneute Detailmodellierung notwendig, sondern die Integrität des Systemmodells bleibt erhalten und das Detailmodell muss lediglich aktualisiert werden.

5 Ganzheitliche Simulation in Virtuellen Testbeds

Der Begriff „Virtuelles Testbed“ wurde bislang zur Beschreibung sehr unterschiedlicher Systemarchitekturen verwendet. Methoden der 3D-Simulation werden bereits heute intensiv zur Planung, zur Entwicklung und zum Betrieb mobiler Systeme verwendet. Speziell zur Simulation planetarer Explorationsrover wurden Systeme wie ROAMS [Hun08], RCAST [BBL+08] oder Kombinationen speziell angepasster Standardsoftware wie Matlab/Simulink und SIMPACK (z.B. [SGK+08]) entwickelt. [RWA+10] verfolgt einen vergleichbaren Ansatz zum Abgleich von Simulation und realem Testbed, ebenfalls speziell für planetare Explorationsrover. Die Hersteller bekannter CAD-Systeme integrieren Simulationskomponenten zur Analyse strukturmechanischer oder thermodynamischer Aspekte (z.B. Autodesk® Simulation oder Catia® Dymola). Ein Ansatz der speziell die Methoden der virtuellen Realität nutzt und Anknüpfungspunkte zum Virtuellen Testbed aufweist, ist das Smart Hybrid Prototyping [SBS+09]. Dieses Konzept setzt auf die Kombination physischer und virtueller Elemente, um so sehr schnell zu einem ersten Prototyp für die Produktentwicklung zu kommen. Ein umfassender Ansatz, der unterschiedlichste Simulationsalgorithmen zur ganzheitlichen Systemanalyse einschließlich aller Wechselwirkungen miteinander integriert fehlt bislang. Darüber hinaus beschränken sich viele Ansätze auf konkrete Anwendungsbereiche und die Analyse des Roboters selbst und nicht der Gesamtmission.

Das Systemmodell (Kapitel 2.1) ist die einzige Eingangsgröße für die ganzheitliche Simulation komplexer Systeme in Virtuellen Testbeds. Während der Simulation kommunizieren die unterschiedlichen an der Simulation beteiligten Komponenten über die zentrale Datenhaltung und tragen ihre Simulationsergebnisse dort ein. Hierdurch kann das Simulationsergebnis im Nachgang der Simulation interaktiv nachvollzogen und analysiert werden. Neben der Datengrundlage ist die Leistungsfähigkeit Virtueller Testbeds und damit der hier vorgeschlagenen Entwicklungsmethodik entscheidend abhängig von den zur Verfügung stehenden Simulations- und Daten verarbeitenden Algorithmen sowie den Schnittstellen (Daten, Hard- und Softwaresysteme, Algorithmen). Zur Realisierung des Konzepts „Virtueller Testbeds“ und zur Integration der vielfältigen hierfür notwendigen Aspekte war entsprechend die Entwicklung neuer Softwarearchitekturen im Bereich der 3D-Simulationstechnik notwendig. Die hier vorgeschlagenen Methoden basieren daher auf dem 3D-Simulationssystem VEROSIM [RSS+13]. Dieses Simulationssystem basiert auf einer Mikrokern-Architektur, wobei der Mikrokern durch eine Echtzeit-Simulationsdatenbank gebildet wird.

Auf dieser Simulationsdatenbank aufbauend sind bereits heute unterschiedlichste Simulationsalgorithmen verfügbar. Diese reichen von Software-Modulen zur Simulation von Kinematik [RSS+11], Starrkörperdynamik [RJR10], Bodenmechanik [RJR10], Hydraulik, Pneumatik, von Sensoren [ERS+11] und Aktoren [YJR+10] und Steuerungskomponenten (z.B. Roboter oder speicherprogrammierbare Steuerungen [RSS+11]) bis hin zur Simulation weiterer physikalischer Effekte (z.B. Feuer) oder spezialisierter Mechanismen. Methoden zustandsorientierter Modellierung [FS04] stellen darüber hinaus eine Skriptsprache bereit, die klassische objektorientierte Entwicklungsmethoden mit der zustandsorientierten Betrachtungsweise von Petrinetzen integriert und so die einfache Realisierung spezialisierter Algorithmen, Schnittstellen oder Benutzeroberflächen ermöglicht. Alle diese Softwarekomponenten sind so genannte „Erweiterungen“ (Plug-Ins) der Simulationsdatenbank und können mit dieser sowie über diese auch miteinander zielgerichtet und in Echtzeit kommunizieren. Sie wurden auf Grundlage einer einheitlichen Programmierschnittstelle entwickelt, welche auch die Grundlage für die Realisierung neuer zur Entwicklung mechatronischer Systeme für die Weltraumrobotik notwendiger Komponenten bildet. Auf diese Weise sollen im Rahmen dieses Vorhabens neue Simulationsaspekte sowie ggfls. neue Schnittstellen z.B. zu Systemen für die Simulation thermo- oder strukturdynamischer Effekte realisiert werden.

6 Anwendungen

Ein zentrales Ziel von INVIRTES ist die Zusammenführung unterschiedlicher Raumfahrtvorhaben. Dieses Ziel spiegelt sich auch in den Referenzanwendungen wider. In deren Rahmen wird zunächst das Virtual Space Robotics Testbed-Konzept aus früheren Forschungsprojekten auf INVIRTES-Strukturen übertragen. Darüber hinaus werden die Ergebnisse der laufenden Vorhaben iBOSS (On-Orbit-Servicing modularer Satelliten) sowie ViTOS (Rendezvous & Docking von Satelliten im Orbit) in diese Strukturen integriert. Damit sollte das Ergebnis eine hervorragende Grundlage für die Realisierung zukünftiger Projekte bereitstellen.

6.1 Virtual Space Robotics Testbed

Idee des Projekts FastMap [RSS11] ist es, während des Landeanflugs für die Explorationsmission hochauflösende Bilder von der Planetenoberfläche aufzunehmen, daraus ein digitales Höhenmodell und Landmarken zu generieren und diese für die dann folgende durch SELOK [RSS11] bereitgestellte Selbstlokalisierung auf der Planetenoberfläche zu nutzen. Mit dem Beitrag von Virtual Crater [YJR+10] können neben radgetriebenen Systemen können auch Laufroboter und kooperierende Mehrrobotersysteme auf dieser Basis entwickelt werden (Bild 2).

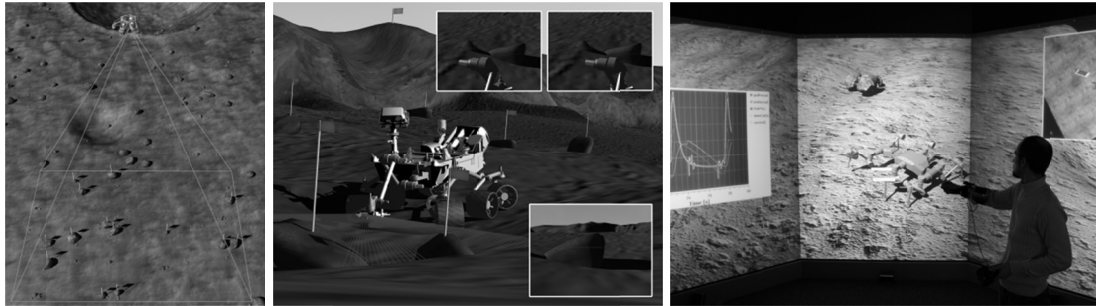


Bild 2: Landung, Kartographierung, Selbstlokalisierung und Exploration auf planetaren Oberflächen im Virtual Space Robotics Testbed (VSRT)

6.2 On-Orbit-Servicing (OOS)

Heutige Satelliten sind meist monolithische Systeme ohne die Möglichkeit zur Wartung oder Instandhaltung. Am Ende seiner Lebensdauer wird der Satellit dann zu Weltraumschrott, der andere Satelliten, die ISS oder bei unkontrolliertem Absturz sogar besiedelte Gebiete auf der Erde gefährdet. Im Projekt iBOSS [WBA+12] wird deshalb ein Konzept für modular aufgebaute Satelliten untersucht, welche dann im Orbit von Servicer-Satelliten rekonfiguriert (siehe Bild 3a), in andere Orbits verbracht oder kontrolliert zum Absturz gebracht werden können. Im Projekt ViTOS wird ein Virtuelles Testbed für optische Sensoren in robotischen Raumfahrtssystemen entwickelt, welches der Entwicklung von Rendezvous & Docking-Strategien dient (Bild 3b).

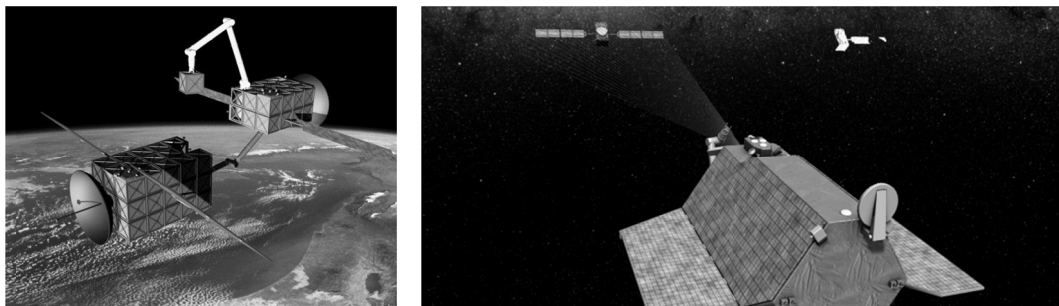


Bild 3: a) Rekonfiguration eines modularen Satelliten (nach [WBA+12]);
b) Rendezvous gestützt durch optische Sensoren

6.3 Resümee und Ausblick

Dieser Beitrag stellt ein Konzept für eine durchgängige rechnergestützte Entwicklung mechatronischer Systeme im Allgemeinen und robotischer Raumfahrtssysteme im Speziellen vor. Die aktuell betrachteten Entwicklungsphasen umfassen den Weg von der Systemspezifikation bis zur detaillierten Gesamtsystemsimulation. Konkret wird hier die CONSENS-Spezifikationstechnik verwendet, um darauf aufbauend Virtuelle Testbeds zu erstellen, in denen das Zielsystem simuliert, analysiert, bewertet und optimiert werden kann, bevor es real umgesetzt wird. Die Durchgängigkeit wird dabei realisiert durch ein anwendungs- und entwicklungsphasenübergreifendes Systemmodell, welches

alle Daten mit Bezug zum Zielsystem integriert und in einer aktiven Datenhaltungs-komponente bereitgestellt wird. Dabei können beliebige Datenschemata integriert werden.

Damit wird ein Überblick über das Forschungsprojekt INVIRTES gegeben (welches sich aktuell selbst in der Spezifikationsphase befindet). Im weiteren Verlauf des Projektes wird in den kommenden Jahren ein Prototyp des INVIRTES-Konzepts entstehen und dieser anhand konkreter Entwicklungsprozesse aus dem Bereich der Raumfahrtrobotik (z.B. Rendezvous & Docking, On-Orbit-Servicing, etc.) evaluiert werden.

Danksagung

Diese Arbeit entstand im Rahmen des Projekts „INVIRTES“, gefördert von der Raumfahrt-Agentur des Deutschen Zentrums für Luft- und Raumfahrt e.V. mit Mitteln des Bundesministeriums für Wirtschaft und Technologie aufgrund eines Beschlusses des Deutschen Bundestages unter den Förderkennzeichen 50RA1306 – 50RA1309.

Literatur

- [BBL+08] BAUER, R.; BARFOOT, T.; LEUNG W.; RAVINDRAN, G.: Dynamic Simulation Tool Development for Planetary Rovers, International Journal of Advanced Robotic Systems, Vol. 5, No. 3 2008.
- [ERS+11] EMDE, M.; ROßMANN, J.; SONDERMANN, B.; HEMPE, N.: Advanced Sensor Simulation in Virtual Testbeds: A Cost-Efficient Way to Develop and Verify Space Applications, AIAA Space 2011, Long Beach, California
- [FS04] FREUND, E.; SCHLUSE, M.: Realisierung moderner 3D-Simulations- und Steuerungsverfahren mit Zustandsorientierter Modellierung, ROBOTIK 2004, München
- [FSM08] FRIEDENTHAL, S.; STEINER, R.; MOORE, A.: A Practical Guide to SysML, Elsevier, Amsterdam, 2008
- [GLL12] GAUSEMEIER, J.; LANZA, G.; LINEMANN, U. (Hrsg.): Produkte und Produktionssysteme integrativ konzipieren – Modellbildung und Analyse in der frühen Phase der Produktentstehung, Carl Hanser, München 2012
- [Hun08] HUNTSBERGER, T.: Characterization of the ROAMS Simulation Environment for Testing Rover Mobility on Sloped Terrain, i-SAIRAS 2008, California
- [HWF+12] HABERFELLNER, R.; DE WECK, O.; FRICKE, E.; VÖSSNER, S. (Hrsg.): Systems Engineering – Grundlagen und Anwendung. Orell füssli Verlag, Zürich, 2012
- [ISO16290] INTERNATIONAL STANDARDIZATION ORGANISATION (ISO): Space Systems – Definition of the Technology Readiness Levels (TRLs) and their criteria assessment. Beuth-Verlag, Berlin, 2013
- [MK11-ol] MULLER, G.; KORFIATIS, P.(Eds.):Architecture Modeling and Simulation – White Paper Resulting from Architecture Forum Meeting. Unter: http://www.architectingforum.org/whitepapers/SAF_WhitePaper_2011_12.pdf, November 2011
- [RJR10] ROSSMANN, J.; JUNG, T.; RAST, M.: Developing Virtual Testbeds for Mobile Robotic Applications in the Woods and on the Moon, IEEE/RSJ IROS 2010, Taipei

- [RSS11] ROSSMANN, J.; SPRINGER, M.; SONDERMANN, B.: Ein Virtuelles Testbed für Planeten-Landemissionen – Hintergründe und Konzept. WInTeSys Bd. 295. Heinz Nixdorf Institut, Universität Paderborn, 2011, S. 107-118
- [RSS+11] ROSSMANN, J.; SCHLETTE, C.; SCHLUSE, M.; EILERS, K.: Simulation, Programming and Control of Kinematics and Other Articulated Mechanisms based on a Uniform Framework, ISPRAS 2011, Cambridge, UK
- [RSS+13] ROSSMANN, J.; SCHLUSE, M.; SCHLETTE, C.; WASPE, R.: A New Approach to 3D Simulation Technology as Enabling Technology for eRobotics. In: Proceedings of the 1st International Simulation Tools Conference & EXPO (SIMEX). Brüssel, 2013, S. 39-46
- [RWA+10] REBELE, B.; WEDLER, A.; APFELBECK, M.; HIRSCHMÜLLER, H.; KUSS, S.; GIBBESCH, A.; SCHÄFER, B.; HIRZINGER, G.: Advanced testbed and simulation environment for planetary exploration and mobility investigations, i-SAIRAS 2010, Sapporo
- [SBS+09] STARK, R.; BECKMANN-DOBREV, B.; SCHULZE, E.-E.; ADENAUER, J.; ISRAEL, J.H.: Smart Hybrid Prototyping zur multimodalen Erlebbarkeit virtueller Prototypen innerhalb der Produktentstehung, VDI-Verlag, 2009, ISBN:978-3-18-302922-8.
- [SGK+08] SCHAEFER, B.; GIBBESCH, A.; KRENN, R.; REBELE, B.: Verification and Validation Process on 3D Dynamics Simulation in Support of Planetary Rover Development, ASTRA 2008, Noordwijk
- [WBA+12] WEISE, J.; BRIESS, K.; ADOMEIT, A.; REIMERDES, H.-G.; GÖLLER, M.; DILLMANN, R.: An Intelligent Building Blocks Concept for On-Orbit-Satellite Servicing. i-SAIRAS 2012 Turin
- [Wei06] WEILKIENS, T.: Systems Engineering mit SysML/UML, Dpunkt-Verlag, Heidelberg, 2006
- [YJR+10] YOO, Y.-H.; JUNG, T.; RÖMMERMANN, M.; RAST, M.; KIRCHNER, F.; ROßMANN, J.: Developing a Virtual Environment for Extraterrestrial Legged Robots with Focus on Lunar Crater Exploration, i-SAIRAS 2010, Sapporo

Autoren

Prof. Dr.-Ing. Jürgen Roßmann, Dr.-Ing. Michael Schluse, Dipl.-Phys. Malte Rast und Dipl.-Inform. Martin Hoppen sind Institutsleiter, Oberingenieur, Gruppenleiter bzw. Mitarbeiter des Instituts für Mensch-Maschine-Interaktion der RWTH Aachen und kombinieren Techniken aus dem Bereich Virtual Reality und 3D-Simulationstechnik um virtuelle Entwicklungsumgebungen für komplexe technische Systeme bereitzustellen.

Dr.-Ing. Roman Dumitrescu, M.Sc. Christian Bremer, M.Sc. Michael Hillebrand sind Abteilungsleiter bzw. Mitarbeiter der Fraunhofer Projektgruppe Entwurfstechnik Mechatronik und erarbeiten Methoden und Werkzeuge für die marktorientierte und kosteneffiziente Entwicklung mechatronischer Produkte.

Dipl.-Inform. Oliver Stern und Dipl.-Inf. Florian Blümel sind Abteilungsleiter bzw. Mitarbeiter des RIF Institut für Forschung und Transfer e.V. und widmen sich dem Technologietransfer aus der Forschung in die industrielle Praxis.

Dr.-Ing. habil. Christoph Averdung ist Geschäftsführer und Gesellschafter der CPA ReDev GmbH, die normenkonforme und datenbankgestützt arbeitende Technologien und Anwendungen in den Bereichen 3D, Simulation und Raumfahrt entwickelt.

Simulationsgetriebene Systemmodellierung zur Analyse und Optimierung von Stückgut-Förderanlagen

Dipl.-Ing. Lukas Weingartner, Dr. Peter Hehenberger
Institut für Mechatronische Produktentwicklung und Fertigung
Johannes Kepler Universität Linz
Altenberger Straße 69, 4040 Linz
Tel. +43 (0) 732 / 2468/6550, Fax. +43 (0) 732 / 2468/6542
E-Mail: lukas.weingartner@jku.at

Dr. Stefan Boschert, Dipl.-Math. Roland Rosen
Siemens AG Corporate Technology
Otto-Hahn-Ring 6, D-81730 München
Tel. +49 (0) 89 / 636/633322
E-Mail: stefan.boschert@siemens.com

Zusammenfassung

In diesem Beitrag wird ein „simulationsgetriebener MBSE-Ansatz“ für die zielgerichtete Aufbereitung, Bereitstellung und Einbindung von Modellen mit hohem Detaillierungsgrad (wie z.B. 3D-CAD-Modellen und Konstruktionsdaten) in Simulationsmodellen aus der Kategorie Anlagensimulation vorgestellt. Dieser Ansatz kann zur Kopplung mehrerer Autorensysteme (z. B. einer Konstruktions- und einer Simulationsplattform) verwendet werden, wobei das Ziel die Implementierung in einer integrierten Umgebung mit zentraler Datenbasis ist (vgl. PLM-Systeme). Die zur Kopplung zur Verfügung stehenden generischen Arbeitsschritte können je nach Anwendungsfall angepasst und in „Arbeitsabläufen“ kombiniert werden. Charakteristische Aufgaben, die darin ausgeführt werden, sind „Aggregation/ Anreicherung, Filterung/Reduktion, Verarbeitung, Transformation, Export/Import“. Der Einsatz von individualisierten und automatisierten Arbeitsabläufen erleichtert bzw. beschleunigt den Modellaufbau und auch den Berechnungsvorgang in der Simulation. Implementiert wird der Ansatz an einem Industriebeispiel, bei dem basierend auf einem umfangreichen 3D-CAD-Modell Simulationsmodelle einer Fördertechnik-Anlage aufgebaut werden. Die Methode kann auf unterschiedliche Werkzeugketten und ähnliche Aufgabenstellungen adaptiert werden, wobei Arbeitsschritte/-abläufe und auch Zwischenergebnisse direkt bzw. leicht modifiziert wiederverwendet werden können.

Schlüsselworte

Systemmodellierung und -simulation, Modellbasierte Produktentwicklung, Modellreduktionsmethoden, Anlagensimulation, Fördertechnik

Simulation driven system modelling for analysis and optimization of cargo conveying systems

Abstract

This paper presents a “simulation-driven MBSE approach” for the preparation, deployment and integration of detailed design models (such as 3D-CAD models and design data) for the usage in simulation models from the category plant simulation. This approach can be used to couple several authoring tools (for instance a design and a simulation platform). The aim is an implementation within an integrated environment with a central data base (like PLM systems). All generic steps to implement the coupling can be combined to “workflows” depending on the specific application. Typical tasks are “aggregation, filtering, preprocessing, export and import”. The use of these individualized workflows facilitates and accelerates the modelling and also the calculation in the simulation. The approach was implemented on an industrial example in which simulation models of a material handling conveyor system are constructed based on a detailed 3D-CAD model. The method can be adapted to different tool chains and similar tasks. Steps, work processes and also intermediate results can be reused directly or slightly modified.

Keywords

System modelling and simulation, model-based product development, model reduction methods, plant simulation, conveyor technology

1 Einleitung

Im Anlagenbau (z.B. bei der Entwicklung von Paketfördersystemen) treten Fragen nach den tatsächlich zu erwartenden Leistungen/Eigenschaften des Systems (wie z.B. Durchsatz, Qualität,...) über die Entwicklungsphasen immer wieder auf und müssen meist mithilfe von Simulationen beantwortet werden. Ab einem bestimmten Entwicklungsfortschritt stehen typischerweise vom mechanischen Teil der Anlage detaillierte (Geometrie)-Modelle in 3D-CAD-Systemen zur Verfügung, die dann als Basis für umfangreiche Systemsimulationen dienen. Da Förderanlagen aus einer Vielzahl von Unterbaugruppen und Komponenten (meist über 100000 Elemente) bestehen, stellt die Handhabung der zugehörigen Detailmodelle hohe Anforderungen an Rechnerleistung, an eingesetzte Werkzeuge und an die Konstruktions- und Arbeitsmethodiken. Werden die komplexen Modelle innerhalb einer Softwarewerkzeugkette über Schnittstellen ausgetauscht oder in verschiedenen Werkzeugen disziplinübergreifend weiterverwendet, sind Ansätze notwendig, um die hohe Anzahl der Objekte, Parameter und Interaktionen zu beherrschen. Moderne 3D-CAD-Systeme und Simulationswerkzeuge unterstützen zwar verschiedenste Schnittstellenstandards, eine Durchgängigkeit des Informations- und Datenaustausches zwischen den Systemen wird in der industriellen Praxis dennoch oft nicht erreicht. Aufwändige manuelle Tätigkeiten wie werkzeugübergreifendes Kopieren von Informationen sowie redundante und inkonsistente Daten sind oft die Folge.

In traditionell organisierten Maschinen- und Anlagenbauunternehmen verantwortet und verwaltet oft eine eigene Konstruktionsabteilung die CAD-Modelle. In einer Gesamtkonstruktion werden üblicherweise Baugruppen aus vorangegangenen Projekten in angepasster Form wiederverwendet und detaillierte Modelle von Sublieferanten (Motorenhersteller, Fördertechnikhersteller,...), deren Detaillierungsgrad oft höher als für den konkreten Einsatzfall erforderlich ist, verbaut. Die dabei notwendigen Konvertierungen zwischen verschiedenen Versionen und Dateiformaten sowie die sehr hohe Anzahl an Elementen und Querbeziehungen erschweren die Handhabbarkeit und den Austausch dieser Modelle. Da diese CAD-Modelle ab einem gewissen Entwicklungsgrad einen großen Teil der Produktdokumentation darstellen, werden neben Stücklisten und Fertigungszeichnungen auch Animationen (Kundenpräsentationen) oder Montageanleitungen daraus abgeleitet (in [Lee05] werden dazu die übergeordneten Themenbereiche „Virtual Prototyping“, „Virtual Manufacturing“ usw. angeführt). Komplexere nachgeschaltete Anwendungsfelder wie Simulationen (FE-Simulationen, Kollisionsüberprüfungen, Durchsatzsimulationen,...) nutzen ebenfalls diese CAD-Modelle, die aber meist noch geeignet aufbereitet und mit zusätzlichen Informationen angereichert werden müssen, bevor sie zum Einsatz kommen können. Diese Aktivitäten erfordern oft die Zusammenarbeit mehrerer Stakeholder und Experten aus unterschiedlichen Disziplinen, die sich durch mehrere Faktoren schwierig gestaltet. Ein Beispiel, das in der Literatur [AB14] genannt wird, ist die „personelle Trennung der Konstruktion von der Berechnung“. Spezialisten der Berechnung oder Simulation haben teilweise nur geringe Erfah-

rung mit 3D-CAD-Systemen und auf ihrem Arbeitsplatz sind oft weder die erforderlichen Zugriffsrechte noch die Lizenzen verfügbar, um mit einer gemeinsamen Datenbasis und mit gemeinsamen Modellen zu arbeiten. Die Stakeholder können darüber hinaus auf unterschiedliche Abteilungen, Standorte oder mehrere Firmen (z.B. Zulieferbetriebe) verteilt sein. Wenngleich die Tätigkeiten innerhalb von Konstruktionsabteilungen üblicherweise durch eigene Konstruktionsrichtlinien unterstützt werden, sind für den Austausch zwischen den Anwendungen bzw. Abteilungen – auch aufgrund komplizierter Verantwortungsverhältnisse – oft keine oder nur sehr allgemein gehaltene Prozesse definiert. Nach ANDERL und BINDE [AB14] führt diese unzureichende Prozessorientierung beispielsweise bei Freigabe- und Änderungsprozessen, in denen Konstruktions- sowie Berechnungsabteilungen beteiligt sind, zu Integrationsproblemen. Da die beschriebenen kritischen Punkte von den Autoren auch in laufenden Industrieprojekten beobachtet und auch von Firmenpartnern aufgezeigt wurden, besteht diesbezüglich Forschungs- und Handlungsbedarf. Um die genannten Situationen sowohl durch eine methodische als auch durch eine softwareseitige Unterstützung zu verbessern, ist die Entwicklung konkreter Arbeitsabläufe für die disziplinen- und abteilungsübergreifende Zusammenarbeit und den dabei notwendigen Austausch von Daten sinnvoll. Im Idealfall können diese Arbeitsabläufe direkt innerhalb der genutzten Werkzeuge hinterlegt werden, sodass je nach Bedarf und Aufgabenstellung ein geeigneter Arbeitsablauf zur Verfügung steht, der eine (semi-)automatische Bereitstellung der benötigten Daten in der gewünschten Form unterstützt bzw. ausführt. Bild 1 zeigt schematisch an einer exemplarischen CAD-CAE-Toolkette, wie in einem Arbeitsablauf die im Simulationsmodell benötigten Daten (Hauptabmessungen, Materialparameter, Randbedingungen, Belastungen,...) je nach Modellzweck (z. B. Ermittlung der Durchbiegung oder Eigenfrequenz eines Trägers) aggregiert/aufbereitet werden. Eine Methodik für die Erstellung derartiger Arbeitsabläufe, die sich aus Arbeitsschritten wie z.B. Extraktionen oder Transformationen zusammensetzen können, wird in den folgenden Kapiteln vorgestellt.

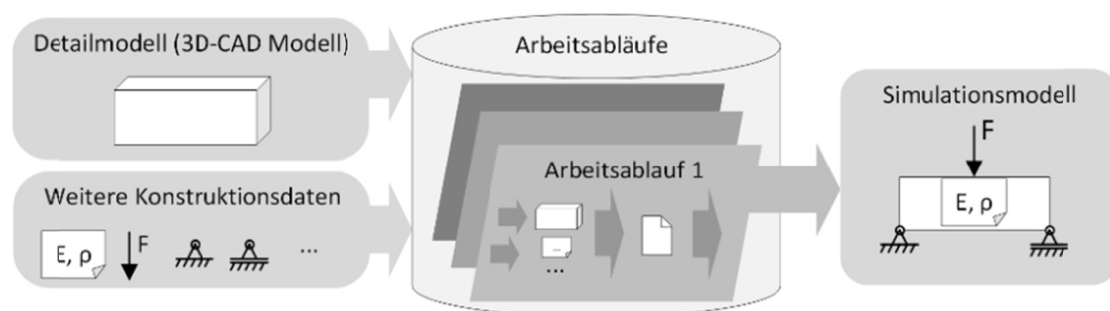


Bild 1: Vordefinierte und angepasste Arbeitsabläufe unterstützen die Aufbereitung und den zielgerichteten Austausch von Daten zur Nutzung in Simulationen

2 Stand der Technik

2.1 Klassifizierung von Simulationswerkzeugen

Moderne Simulationswerkzeuge für Produktion und Logistik unterscheiden sich stark im Modellierungskonzept (siehe [Ele12] und [AIK+08]). Grundlegend können Simulationsmodelle sehr flexibel mithilfe allgemeiner Programmiersprachen erstellt werden. In den Hierarchien darüber existieren spezielle Simulationssprachen, Simulator-Entwicklungsumgebungen und Simulationsumgebungen (meist durch Bibliothekselemente mit Anwendungsbezug unterstützt) bis hin zu Spezialsimulatoren mit geringer Allgemeingültigkeit bzw. Flexibilität. Diese Arbeit bezieht sich hauptsächlich auf Modellierungskonzepte, die auf der Konfiguration von Bibliothekselementen basieren.

2.2 Methoden zur Integration von Softwarewerkzeugen

Notwendige Maßnahmen und Arbeitsschritte, um eine stärker automatisierte Kopplung zwischen CAD und FEM zu erreichen, wurden bereits im Jahr 1993 in [ABS93] diskutiert. Dabei wird ausgehend von einem Produktdatenmodell, über eine CAD-FEM-Transformation (Attributeditor, Detaileditor, Dimensionalitätsreduktion und Gliederung) ein Netz generiert und das Analysemodell erstellt. In [Amf02] werden mehrere Konzepte für den Datenaustausch zwischen CAD- und FEM-Systemen sowie grundlegende Möglichkeiten der Schnittstellenkommunikation vorgestellt und zusammengefasst. Bei der dabei vorgestellten modellorientierten Schnittstellenkommunikation wird ein vorab nicht vorhandenes Modell B aus einem vorhandenen Modell A generiert, hingegen werden bei einer parameterorientierten Vorgehensweise gemeinsam auftretende Parameter aus zwei Modellen A und B verbunden. An dieser Stelle wird auf die detaillierten Ausführungen in [VWB+09] in Bezug auf Kopplungsmöglichkeiten (z. B. Direktschnittstellen, Möglichkeiten durch parametrische Modellierung,...) und Vorgehensweisen bei der Modellerstellung (z. B. CAD-, FEM-, MKS-Modellierungstechniken) verwiesen. In [Rou14] werden ausgehend von CAD typische CAx-Prozessketten als „formale und hierarchische Strukturierung von Informationsprozessen“ im Umfeld der Mechanischen Konstruktion vorgestellt.

2.3 Datenaustauschformat und Aufbereitung der Daten

Neben der Frage der geeigneten Schnittstelle muss vorab ein Konzept für die konkret zu übertragenden Informationen erarbeitet werden, das auch eine geeignete Aufbereitung der Daten bzw. der Modelle einschließt. Nicht immer ist es notwendig ein 3D-Geometrieformat zur Übertragung von Geometrieinformationen zu verwenden. Beispielsweise kann durch die Nutzung von Symmetrien eine Reduktion auf wesentliche Parameter erfolgen (im Fall eines Zylinders sind das beispielsweise Radius und Anfangs-/Endpunkt), was andere (eventuell zweckmäßigere) Methoden für den Datenaus-

tausch ermöglicht (Möglichkeiten für derartige Modellreduktionen werden in [NTA+14] vorgestellt). Im genannten Beispiel kann eine Textdatei (XML,...) zur Übertragung dieser wesentlichen Hauptparameter einem 3D-Datenformat überlegen sein, weil dadurch eine einfach lesbare, interpretierbare und manipulierbare Datei entsteht. Unterschiedliche Techniken zur Vereinfachung von Oberflächen- und Volumenrepräsentationen von 3D-Geometriemodellen für die Anwendungsbereiche FEM oder andere physikbasierte Simulationen sind in [TBS09] zusammengefasst. In [Lee05] wird beschrieben, wie Feature-basierte Konstruktionen idealisiert und für andere Anwendungen (Rendering, Virtual Manufacturing,...) bereitgestellt werden können. Eine Evaluierung der derzeit am weitesten verbreiteten neutralen 3D-Dateiformate IGES 5.3, STEP AP214, VRML97, X3D 3.2, JT und 3D-PDF hinsichtlich geometriespezifischer Eigenschaften, Metadaten, Leichtgewichtigkeit und anderen Kriterien wird in [FLL+11-ol] durchgeführt. Darin wird auch die Leistungsfähigkeit der Exportmodule verbreiteter Autorenwerkzeuge sowie von 3rd-Party-Konvertierungswerkzeugen gegenübergestellt.

2.4 Möglichkeiten zur transparenten Darstellung der Zusammenhänge zwischen Modellen

Um eine effiziente Zusammenarbeit zwischen den Stakeholdern und die Integration der relevanten Softwarewerkzeuge zu ermöglichen, ist eine genaue Situationsanalyse erforderlich. Speziell zur Analyse von Änderungsauswirkungen ist eine grafische Modellierung und Darstellung hilfreich. Eine Möglichkeit, die oft komplexen Verknüpfungen und Prozesse transparent zu machen, wird in [FHS+14] vorgestellt. Dabei werden die Themenbereiche Entwicklungsprozess, Modelle, Organisation und Kommunikation analysiert und die Abhängigkeiten in einer „Model Dependency Map“ dargestellt. Auf in der Literatur weit verbreitete Modellierungstechniken wie UML und SysML wird beispielsweise in [Lin11] eingegangen.

3 Generische Methodik zur simulationsgetriebenen Systemmodellierung

In der entwickelten Methodik bilden Modelle mit hohem Detaillierungsgrad den Ausgangspunkt für die Arbeitsabläufe. Im Beispiel eines 3D-CAD-Modells sind darin sowohl geometrische als auch textuelle Produktdaten enthalten. Die Methodik kann aber auch an andere Anwendungsfälle (wie z.B. ereignisdiskrete Simulation auf Basis eines detaillierten 2D-CAD-Layouts) adaptiert/angewendet werden. In Abschnitt 3.1 werden die Teilschritte beschrieben, aus denen sich unidirektionale (rückwirkungsfreie) Arbeitsabläufe zur Erstellung eines Simulationsmodells auf Basis bestehender Detailmodelle zusammensetzen. In Abschnitt 3.2 werden bidirektional notwendige Maßnahmen, um die Konsistenz der Modelle innerhalb der verschiedenen Softwarewerkzeuge sicherzustellen, und Möglichkeiten zur Implementierung der Arbeitsabläufe dargestellt. Für eine Integration der Softwarewerkzeuge ist eine Zuordnung zwischen Elementen aus

dem Detailmodell und den zum Aufbau der Simulation benötigten Elementen erforderlich. Um diesen wesentlichen Schritt deutlich zu machen, wird an dieser Stelle konkret auf das Beispiel 3D-CAD und Fördertechnik-Simulation eingegangen. Die Schwierigkeit bei der Zuordnung von Elementen aus dem Geometriemodell zu Modellbausteinen der Simulation entsteht dadurch, dass unterschiedliche Ordnungsstrukturen (Sichten) zugrunde liegen (vgl. „Phasenabhängige Ausprägung von Produktstrukturen“ in [ES09]). In CAD-Systemen wird beim Modellaufbau i. Allg. eine fertigungs- oder prüforientierte Baugruppenstrukturierung verwendet, während beim Aufbau eines Simulationsmodells meist eine funktionsorientierte Betrachtung (vgl. Verhaltenssimulation) im Zentrum steht. Nachdem in 3D-CAD-Systemen oft die funktionale Sicht bzw. die Funktionen der einzelnen Baugruppen nicht explizit abgebildet und modelliert werden, müssen diese erst identifiziert und nachgebildet werden, damit eine Zuordnung zu den in der Simulation erforderlichen Modelltypen oder Klassen erfolgen kann. Zur Identifikation der für eine Klasse relevanten Geometriedaten können Filter definiert werden. Als Filterkriterien eignen sich Merkmale, die im CAD-Modell direkt vorhanden oder dem Modell zugeordnet wurden. Beispiele dazu sind Geometrie-Merkmale (Abmessungen, Fläche, Volumen, Abstand von einem Referenzpunkt, Lage axial/radial/tangential, außen-/innenliegende Fläche), Parameter (Materialvergabe, Masse), Attribute (Zuordnung zu einer Gruppe, Benennung, Status - siehe auch Nummern- und Klassifizierungssysteme in [ES09]). Wenn die in den Modellen bereits enthaltenen Merkmale zur Filterung nicht ausreichen, können zusätzliche Attribute definiert und den Modellen entweder manuell oder automatisch (aufgrund bestimmter Kriterien und Regeln) zugeordnet werden. Eine aus Sicht der Simulation günstige Methode, bei der erst ein Funktionsmodell gebildet wird, das in den Phasen der Detaillierung mit CAD-Modellen angereichert wird, präsentiert [Lin11] im Kontext der „Virtuellen Inbetriebnahme“.

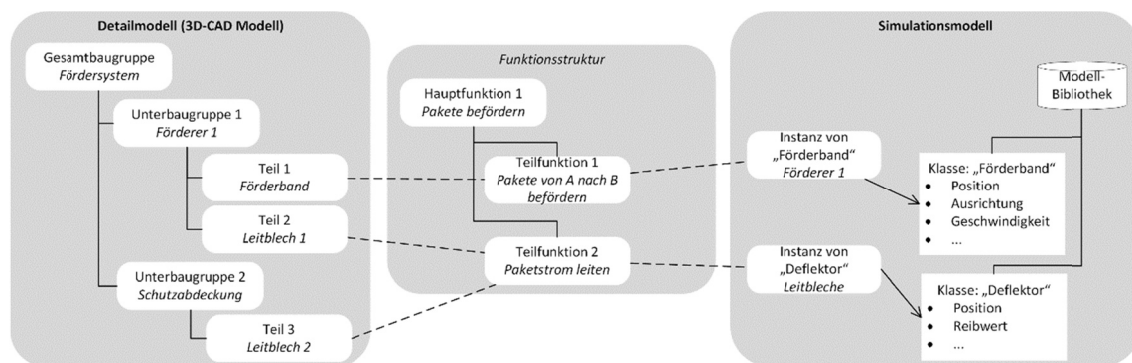


Bild 2: Beispiel einer Funktionsstruktur zur Verknüpfung von Elementen aus einem Detailmodell (CAD) mit Elementen aus einem Simulationsmodell

3.1 Erstellung und Nutzung unidirektionaler Arbeitsabläufe zum Aufbau eines Simulationsmodells

Nachdem der Zweck des Simulationsmodells bereits hinreichend bekannt und festgelegt ist, wird in einer Abstimmung zwischen allen beteiligten Stakeholdern grundsätzlich

abgeklärt, welche Informationen in der Simulation benötigt werden, ob diese Informationen im bestehenden Detailmodell vorhanden sind und wie sie übernommen werden können. Es sollten nur simulationsrelevante Daten übertragen werden, um den Berechnungsvorgang, die Konsistenz zwischen den Daten sowie die Interpretation der Simulationsergebnisse möglichst einfach zu halten. Bei der Abstimmung der Datenformate für die Übergabe muss abgeklärt werden, welche Schnittstellen von beiden Werkzeugen unterstützt werden bzw. welche sich für die jeweilige Art der ausgetauschten Information eignen. Im Fall einer Direktschnittstelle ist zwar kein Import/Export notwendig, die Leistungsfähigkeit/Eignung (vgl. Kapitel 2) muss allerdings ebenfalls evaluiert werden, da mitunter nicht alle benötigten Informationen geeignet übertragen werden.

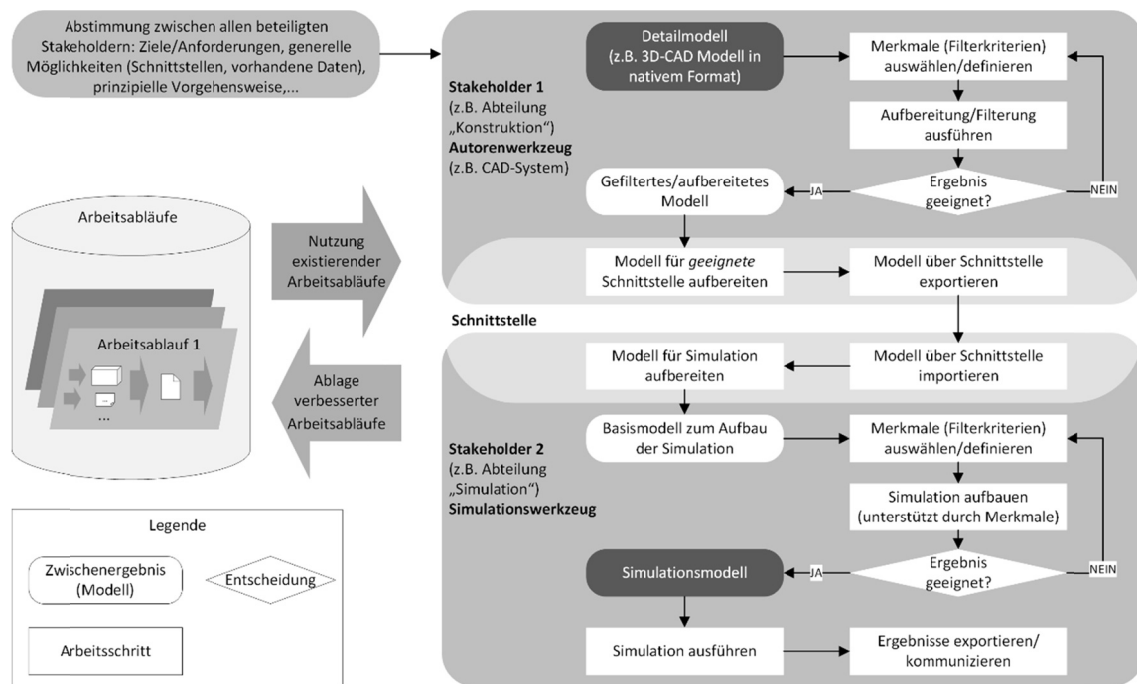


Bild 3: Schema zur Methodik für den unidirektionalen Informationsaustausch zum Aufbau eines Simulationsmodells auf Basis eines Detailmodells

Für die Auswahl und die Vorbereitung zur Übernahme sind im Autorensystem mehrere Schritte notwendig. Zuerst müssen die benötigten Daten aus dem detaillierten Modell gefiltert und geeignet aggregiert werden. Merkmale (Filterkriterien) werden dabei solange angepasst und verfeinert, bis das gewünschte Ergebnis erreicht wird. Die bei dieser Filterung ausgeführten Arbeitsschritte bzw. Arbeitsabläufe werden abgelegt und stehen als Basis für zukünftig benötigte Arbeitsabläufe zur Verfügung. Nach der Filterung erfolgt die Übertragung der geeignet aufbereiteten (siehe Methoden aus Abschnitt 2.3) Modelle über die gewählte Schnittstelle. Nach dem simulationsseitigen Import stehen je nach Werkzeug ebenfalls Methoden zur Filterung zur Verfügung. Damit können auf Basis der übertragenen Modelle und der dazu im Autorensystem vergebenen Merkmale die zugehörigen Simulationsbausteine identifiziert, eingesetzt und parametrisiert werden.

3.2 Bidirektionale Assoziativität zwischen den Modellen

Bidirektionale Beziehungen zwischen Geometrie- und Simulationsmodell spielen speziell beim Management von Änderungen eine wichtige Rolle. Um die Konsistenz zwischen den Modellen sicherzustellen muss zu passenden Zeitpunkten ein Abgleich der Modelle erfolgen. Einerseits treten Änderungen im Detailmodell auf und müssen im Simulationsmodell korrigiert werden. Andererseits können Simulationsergebnisse Anweisungen über notwendige Änderungen, die im Detailmodell umgesetzt werden müssen, enthalten. Eine geeignete Modularisierung und Parametrisierung der Modellelemente in der Simulation (vgl. parametrischer Bibliotheksmechanismus für die Wiederverwendung von Komponenten in [Lin11]) ermöglicht einen Ersatz durch neue Versionen. Damit diese Vorgehensweise zielführend ist, muss klar zugeordnet sein, welche Auswirkung Änderungen im Detailmodell auf die Modellelemente in der Simulation haben. Tiefgreifende strukturelle Änderungen im Detailmodell sind nicht erlaubt, wenn in diesem Fall auch die Zuordnungen angepasst werden müssten. Werden strukturelle Änderungen notwendig, ist eine Abstimmung unter den Stakeholdern notwendig (z.B. automatische Benachrichtigung). Um derartige Änderungen erkennen zu können, müssen Regeln festgelegt werden, in denen die Freiräume (z.B. „Maximalkonfiguration“ mit allen erlaubten Varianten) definiert werden. Die Verknüpfungen auf Basis einer Funktionsstruktur können konkret entweder im Autorensystem (z.B. CAD), innerhalb der Simulationsplattform oder in einer dazwischengeschalteten Anwendung (z.B. zentrale Datenbasis, wie sie PDM- oder PLM-Systeme bieten) implementiert werden. Der Aspekt der unterschiedlichen Sichten auf die Daten ist eine ureigene Aufgabe von Datenbanken. Die CAD-Modelle, Konstruktionsdaten sowie Simulationsmodelle und -ergebnisse werden dabei im zentralen PLM-Backbone in PDM- bzw. den Autorensystemen nahestehenden TDM-Systemen [ES09] abgelegt. Werden für die Simulation Daten aus detaillierten Modellen benötigt, können die aktuellen Daten durch Ausführen des jeweiligen Arbeitsablaufes direkt aus dem PLM-Backbone generiert werden.

4 Umsetzung der Methodik am Beispiel „Paketfördersystem

Bei dem betrachteten Gesamtsystem handelt es sich um eine Paket-Sortieranlage, in der spezielle vom Firmenpartner entwickelte Module (z.B. Paket-Entladesystem) mit Standard-Förderkomponenten externer Hersteller zusammengeschaltet werden. In der Phase der Detailkonstruktion ergeben sich Fragestellungen wie Durchsatzermittlung, Geometrieoptimierung oder regelungstechnische Stau-/Störungsbeseitigung, die bereits vorab ohne die Notwendigkeit eines physischen Prototyps der Anlage mithilfe von Simulationen bearbeitet werden müssen. Ein dazu verwendetes Softwaretool Demo3D® ist ein bausteinorientiertes Materialfluss-Simulationswerkzeug, in dem Förderanlagen virtuell aufgebaut, programmiert und animiert werden können. Die Einflüsse von Aktoren (Fördertechnik, Roboter,...) auf das beförderte Stückgut können dabei unter Einbeziehung physikalischer Effekte (Gravitation, Reibung, Massenträgheit,...) simuliert werden.

Ausgehend von einem detaillierten CAD-Modell in Siemens NX® (CAD/CAM/CAE-Plattform) [Sie14-ol] wird im Simulationswerkzeug Demo3D® Enterprise [Emu14-ol] ein Simulationsmodell aufgebaut. Eine Integration zwischen den Werkzeugen ist dafür nicht gegeben, was aufwendige manuelle Arbeiten und hohen zeitlichen Aufwand verursacht. Der Modellaufbau sollte daher durch möglichst effektive und weitgehend automatisierte Arbeitsabläufe unterstützt werden und auf der vorhandenen CAD-Geometrie basieren. Unter Berücksichtigung etablierter Schnittstellen wurde mithilfe der vorgestellten Methodik eine Schnittstellenarchitektur basierend auf Funktionen bzw. einer Funktionsstruktur entwickelt und umgesetzt. Für den Aufbau des Simulationsmodells in Demo3D® sind grundsätzlich vier Modelltypen/Klassen notwendig. „Visualisierungselemente“ sind Geometrielemente, die ausschließlich zur visuellen Darstellung (zum Zweck von Plausibilitätsprüfungen, Animationen,...) verwendet werden und somit in der Simulation mit keinen zusätzlichen Funktionalitäten belegt werden müssen (z.B. Außenhülle des CAD-Modells). „Deflektoren“ sind Geometrielemente, die vom beförderten Stückgut (z.B. Pakete) nicht durchdrungen werden dürfen (z.B. Leitbleche) und in der Simulation durch Zuordnung zum Modelltyp „Deflektor“ als solche berücksichtigt werden. „Konfigurierbare Bibliothekselemente“ aus dem mitgelieferten Katalog von Demo3D® sind vorgefertigte Modellbausteine (z.B. Förderbänder, Rollenbahnen, Aufzüge) mit bestimmten charakteristischen Funktionalitäten (u.a. steuerbare Förderfunktionen) und konfigurierbaren Parametern (Abmessungen, Fördergeschwindigkeit,...), die vor der Verwendung in der Simulation den Anforderungen entsprechend angepasst werden können. „Sensoren“ sind Standard-Geometrielemente (z.B. Zylinderelemente), die in der Simulation mit spezieller Funktionalität (Modelltyp „Sensor“) ausgestattet werden können und dann beispielsweise als Lichtstrahl eines Sensors wirken.

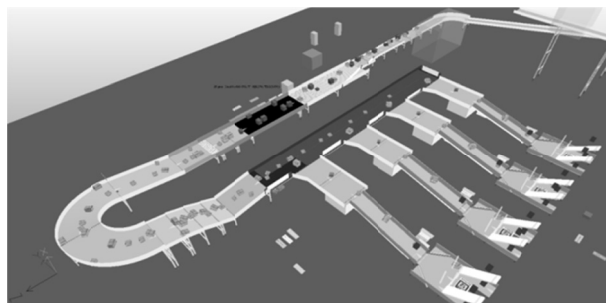


Bild 4: Simulationsaufbau einer Paketförderanlage in Demo3D® (mit freundlicher Genehmigung von Siemens AG - Logistics and Airport Solutions)

Bei den im konkreten Beispiel (in Abstimmung mit allen beteiligten Stakeholdern) erarbeiteten Arbeitsabläufen wird die Filterung und Anpassung der Geometriedaten ausschließlich auf CAD-Seite durchgeführt, da die Simulationsumgebung im Gegensatz zum CAD-System diesbezüglich nur sehr eingeschränkte Funktionalitäten bereitstellt. Im ersten Vorfilterungsschritt können nicht benötigte Geometrie- und Konstruktionselemente (z.B. Hilfsgeometrie, Normteile,...) durch Ausblenden der betreffenden „Layer“ ausgeschlossen werden, sofern die Layer-Zuordnung vorab geeignet durchgeführt

wurde (Konstruktionsrichtlinien). Für die Visualisierungselemente ist im betrachteten Fall nur die Außengeometrie relevant. Um die relevanten Teile zu erfassen wurden mehrere Filter (z.B. Vorgabe von Mindestvolumen/-oberfläche zum Ausschluss sehr kleiner Teile) und spezielle CAD-Funktionalitäten (z.B. Außenhülle generieren) kombiniert. Das Ergebnis der Filterung wird durch ein Text-Attribut „Visualisierung“ gekennzeichnet und einer Sichtprüfung unterzogen. Korrekturen (z.B. Wiederhinzufügen relevanter kleiner Teile wie Griffe, Anzeigeelemente oder Entfernen nicht benötigter Teile,...) werden durch Anpassung der Filterkriterien oder durch manuelles Setzen des Attributs durchgeführt. Für Deflektoren ergibt sich eine ähnliche Vorgehensweise wie für die Filterung der Visualisierungselemente. Das CAD-System bietet hierzu hilfreiche Auswahlmethoden für außenliegende Flächenelemente, bei denen alle aus definierten Ansichten (sowie aus einer Blickrichtung ausgehend von einem Punkt) sichtbaren/zugewandten Flächen automatisch markiert werden. Damit lassen sich Leitelemente (z.B. Leitbleche) durch eine geeignete Wahl von Perspektiven oder Ursprungspunkten entlang des Materialflusses rechnergestützt selektieren. Förderbändern werden gewöhnlich spezielle Materialien zugewiesen (z.B. Gummi/Textil) was eine einfache Filterung aufgrund dieses Merkmals ermöglicht. Sensoren sind auf einem eindeutigen Layer abgelegt, was die Filterung ebenfalls erleichtert. Nach Ausführung der beschriebenen Arbeitsabläufe für die Filterung stehen im CAD-System vier (teilweise überschneidende) Gruppierungen von Geometrieelementen für den Export zur Verfügung. Die Übertragung der Geometrie der Visualisierungselemente erfolgt im speziell für 3D-Darstellung geeigneten Dateiformat VRML. Für die Deflektoren-Geometrie wird STEP gewählt, da sowohl die übertragene Teilebezeichnung als auch Baugruppenstruktur in der Simulation weiterverwendet werden kann. Zur Übertragung der Hauptparameter von Förderbändern und Sensoren werden aus dem CAD ausschließlich Punktkoordinaten als Textdatei exportiert. Förderbänder werden durch drei Koordinaten (Anfangs- und Endpunkt sowie Anfangspunkt/seitlich), Sensoren durch Anfangs- und Endpunkt in ihrer Geometrie spezifiziert. Im Simulationswerkzeug werden diese Geometriedaten der Visualisierungselemente und Deflektoren importiert, durch eine Koordinatentransformation geeignet ausgerichtet und den jeweiligen Modelltypen zugeordnet. Konfigurierbare Bibliothekselemente (Förderer) und Sensoren werden mithilfe eines Skripts gemäß den übertragenen Koordinaten (Hauptparameter) geeignet parametrisiert und in das Simulationsmodell eingebaut. Die verwendeten Arbeitsabläufe können in den Autorenssystemen durch „API“ bzw. „Scripting“ zu einem hohen Grad automatisiert (programmiert) werden und stehen für zukünftige ähnliche Aufgabenstellungen zur Verfügung. Bei der Umsetzung am Beispiel wurden neben einer Zeitersparnis beim Aufbau der Simulationsmodelle auch eine Erhöhung der Qualität des Modells hinsichtlich Detailtreue und Leistungsfähigkeit erreicht. Die konsequente Umsetzung der Schnittstelle zwischen CAD und Simulationswerkzeug mithilfe unterschiedlicher Arbeitsabläufe für die Modelltypen/-klassen bietet Möglichkeiten für den Umgang mit Änderungen, indem beispielsweise „Konfigurierbare Bibliothekselemente“ durch überarbeitete Parametersätze aktualisiert

werden können. Anschließende Forschungsarbeiten behandeln die im Kapitel 3.2 erläuterten Herausforderungen im Fall tiefgreifender struktureller Änderungen.

5 Resümee und Ausblick

Der vorgestellte Ansatz zeigt Möglichkeiten, wie systematisch erarbeitete Arbeitsabläufe zu einer besseren Kopplung zwischen Entwicklungswerkzeugen beitragen können. Im PLM-Gedanken (siehe auch [ES09]) sind viele der präsentierten Schritte bereits grundlegend verankert. Nachdem viele Softwarewerkzeuge nur eingeschränkte Möglichkeiten zur Integration bieten und eine flächendeckende Umsetzung derartiger Arbeitsabläufe in PLM-Systemen in der industriellen Praxis nur in Ansätzen beobachtet werden konnte, besteht offenbar Bedarf an einfach anwendbaren Methoden und praktikablen Lösungen. Die Hersteller der Softwaretools sollten den Anwendern verstärkt generische Templates für typische Arbeitsabläufe zur Integration mehrerer Werkzeuge zur Verfügung stellen. Erfahrungsgemäß sind derartige Arbeitsabläufe in modernen PLM-Systemen ebenfalls nur durch aufwendige Anpassungen und zusätzliche Programmierungen implementierbar. Um die hohen Ansprüche, die im Maschinen- und Anlagenbau auch bezüglich Flexibilität der Integrationen von Werkzeugen gestellt werden, systematisch erfüllen zu können, müssen zukünftige Forschungen die konkreten Arbeitsweisen analysieren und Möglichkeiten zur Umsetzung in modernen Werkzeugen liefern.

Danksagung

Diese Arbeit wurde im Rahmen eines K2 Projektes in der LCM GmbH durchgeführt. Die K2 Projekte werden aus Mitteln des österreichischen COMET-K2 Programms finanziert. Die COMET K2 Projekte im LCM werden unterstützt von der Österreichischen Bundesregierung, dem Land Oberösterreich, der Johannes Kepler Universität und allen wissenschaftlichen Partnern, die Teil des K2-COMET Konsortiums innerhalb der LCM GmbH sind.

Literatur

- [Amf02] AMFT, M.: Phasenübergreifende bidirektionale Integration von Gestaltung und Berechnung. Dissertation, Fakultät für Maschinenwesen, Technische Universität München, 2002
- [AB14] ANDERL, R.; BINDE, P.: Simulationen mit NX. Kinematik, FEM, CFD, EM und Datenmanagement. Mit zahlreichen Beispielen für NX 9. Carl Hanser Verlag, München, 2014
- [ABS93] ARABSHAHI, S.; BARTON, D. C.; SHAW, N. K.: Steps Towards CAD-FEA Integration. Engineering with Computers, 9. Jg., Nr. 1, S. 17-26, 1993
- [AIK+08] ARNOLD, D.; ISERMANN, H.; KUHN, A.; TEMPELMEIER, H.; FURMANS, K. (Hrsg.): Handbuch Logistik. 3., neu bearbeitete Auflage, Springer-Verlag, Berlin Heidelberg, 2008

- [Ele12] ELEY, M.: Simulation in der Logistik. Eine Einführung in die Erstellung ereignisdiskreter Modelle unter Verwendung des Werkzeuges „Plant Simulation“. Springer-Verlag, Berlin Heidelberg, 2012
- [ES09] EIGNER, M.; STELZER, R.: Product Lifecycle Management. Ein Leitfaden für Product Development und Life Cycle Management. 2., neu bearbeitete Auflage, Springer-Verlag, Berlin Heidelberg, 2009
- [Emu14-ol] EMULATE3D INC.: Demo3D® Presentation and Prototyping Software Overview. http://www.demo3d.com/show/Demo3D_OV.html, 17.11.2014
- [FHS+14] FRIEDL, M.; WEINGARTNER, L.; HEHENBERGER, P.; SCHEIDL, R.: Model Dependency Maps for transparent concurrent engineering processes. In: De Vin, L. J.; Solis, J. (Hrsg.): Proceedings of the 14th Mechatronics Forum International Conference, Mechatronics 2014, Karlstad, Sweden, S. 614-621, 2014
- [FLL+11-ol] FRIEDWALD, I. A.; LÖDDING, I. H.; LUKAS, U. V.; MESING, B.; ROTH, M.; SCHLEUSENER, S.; TITOV, F.: Benchmark neutraler Formate für den prozessübergreifenden Datenaustausch im Schiffbau. Bericht, Fraunhofer IGD, <http://publica.fraunhofer.de/documents/N-194622.html>, 2011
- [Lee05] LEE, S. H.: A CAD–CAE integration approach using feature-based multi-resolution and multi-abstraction modelling techniques. Computer-Aided Design, 37. Jg., Nr. 9, S. 941-955, 2005
- [Lin11] LINDWORSKY, A.: Teilautomatische Generierung von Simulationsmodellen für den entwicklungsbegleitenden Steuerungstest. Forschungsberichte IWB, Band 249, Herbert Utz Verlag, München, 2011
- [NTA+14] NOLAN, D. C.; TIERNEY, C. M.; ARMSTRONG, C. G.; ROBINSON, T. T.: Defining Simulation Intent. Computer-Aided Design, 59. Jg. (2015), S. 50-63, 2014
- [Rou14] ROUBANOV, D.: Produktmodelle und Simulation (CAE). In: Eigner, M.; Roubanov, D.; Zafirov, R. (Hrsg.): Modellbasierte virtuelle Produktentwicklung. Springer-Verlag, Berlin Heidelberg, 175-195, 2014
- [Sie14-ol] SIEMENS PLM: NX Übersichtsbrochure. <http://www.siemens.de/plm/nx>, 17.11.2014
- [TBS09] THAKUR, A.; BANERJEE, A. G.; GUPTA, S. K.: A survey of CAD model simplification techniques for physics-based simulation applications. Computer-Aided Design, 41. Jg. (2009), Nr. 2, S. 65-80, 2009
- [VWB+09] VAJNA, S.; WEBER, C.; BLEY, H.; ZEMAN, K.; HEHENBERGER, P.: CAx für Ingenieure: Eine praxisbezogene Einführung. 2. Auflage, Springer-Verlag, Berlin Heidelberg, 2009

Autoren

Dipl.-Ing. Lukas Weingartner ist seit der Absolvierung des Diplomstudiums Mechatronik an der Johannes Kepler Universität Linz als „Wissenschaftlicher Mitarbeiter“ am Lehrstuhl für Mechatronische Produktentwicklung und Fertigung tätig.

Dipl.-Ing. Dr. Peter Hehenberger ist seit 2000 am Institut für Mechatronische Produktentwicklung und Fertigung tätig und promovierte 2004 auf dem Gebiet „Mechatronische Produktentwicklung“. Die Schwerpunkte seiner Forschung liegen im Bereich „Model-based Mechatronic System Design“, CAx-Technologien, Vorgehensmodelle in der Produktentwicklung.

Dr. Stefan Boschert studierte Physik an der Universität Freiburg wo er 2000 promovierte. Nach seiner Tätigkeit als Berater und Simulationsingenieur bei GiMS Basel wechselte er in 2004 zur Siemens AG in die zentrale Forschungseinheit Corporate Technology. Als Senior Key Expert arbeitet er als Fachexperte für simulationsbasierte Systementwicklung im Technologiefeld Automation & Control. Aktuelle Arbeitsgebiete sind Multiphysik-Simulationen, Simulationsdaten und -prozessmanagement sowie die Kopplung von daten- und physikbasierten Modellierungsaspekten.

Dipl.-Math. Roland Rosen ist bei Siemens Corporate Technology als Principal Key Expert für die internationale Fokus Area „Modeling & Simulation“ verantwortlich. Sein Arbeitsschwerpunkt liegt in der Realisierung simulationsbasierter Methoden für die Entwicklung mechatronischer Systeme und komplexer Anlagen. Spezifisches Interesse liegt in der Einbindung der Simulation in das Lifecycle Management.

Online-Simulation fluiddynamischer Prozesse in der Konzeptionsphase

M.Sc. Dipl.-Ing.(FH) Stefan Krottil, Prof. Dr.-Ing. Gunther Reinhart

Projektgruppe RMV des Fraunhofer IWU

Beim Glaspalast 5, 86153 Augsburg

Tel. +49 (0) 821 / 56 88 376, Fax. +49 (0) 821 / 56 88 350

E-Mail: stefan.krottil@iwu.fraunhofer.de

Zusammenfassung

Kürzere Produktlebenszyklen und eine steigende Variantenvielfalt rücken den Produktentwicklungsprozess immer weiter in den Fokus. Zur Validierung unterschiedlicher Konzepte und Entwürfe hat sich die Simulation als effektives Hilfsmittel etabliert. Jedoch ist die Ausführung von Fluidsimulationen noch ein stark iteratives Vorgehen. Daher wird eine Methode dargestellt, welche den heutigen iterativen Zyklus zwischen Konzeptions- und Simulationsphase dynamisiert und parallelisiert. Darüber hinaus werden softwaretechnische Anforderungen spezifiziert und die Funktionalität der Methode durch ein Umsetzungsbeispiel dargestellt.

Schlüsselworte

Fluidsimulation, CFD, Echtzeit, Modellierung

Online-Simulation of fluiddynamic processes in the design process

Abstract

The design process gets more and more important having shorter product life cycles as well as a growing range of products. To validate and ensure functionality of product, simulation has been established as a helping tool. But the simulation of fluid dynamic simulation is still a problem because of its strict sequential and iterative procedure. Thus, a new method is introduced, which dissolves this drawback in making the construction and simulation phases executed parallel and more dynamic. In addition, the technical requirements get specified. Furthermore, the functionality of the method gets proved in an filling process example.

Keywords

Computational Fluid Dynamics, modelling, real-time. fluid simulation

1 Die frühe Phase des Produktentwicklungsprozess

1.1 Konzeptionsphase

Der Produktentwicklungsprozess gewinnt durch die steigende Variantenvielfalt und kürzeren Produktlebenszyklen immer mehr an Bedeutung. Darüber hinaus führt eine wachsende Interdisziplinarität zu stärkeren Abhängigkeiten zwischen den unterschiedlichen Ingenieurdisziplinen. [AR11]

Vor allem die frühen Phasen des Produktentwicklungsprozesses gewinnen immer mehr an Bedeutung. Die Konzeptionsphase [ERZ14, S. 16] entscheiden maßgeblich über die späteren Kosten und die Qualität der Produkte [EKL03]. Durch die Verifikation und Validierung sowie dem Vergleich möglichst vieler unterschiedlicher Alternativen können die Risiken von Fehlentwicklungen entscheidend reduziert werden. Die Simulationstechnik hat sich dabei als effektives Hilfsmittel etabliert. Die Gründe hierfür sind zahlreich. Neben dem wirtschaftlichen Aspekt, dass reale Versuchsaufbauten teuer und aufwändig zu realisieren sind, spielen vor allem Faktoren wie die Beobachtbarkeit von Systemen und die einfache Zerlegung von Problemstellungen einen entscheidenden Vorteilsfaktor dar [Döb08].

1.2 Simulation von Fluiden

Die Simulation von fluidischen Prozessen in produktionstechnischen Anlagen, beispielsweise das Abfüllverhalten oder Tauch- und Lackierprozesse, können simulationstechnisch durch Ansätze der Computational Fluid Dynamics (CFD) noch immer nur schwer abgebildet werden. Zum einen sind Fluidsimulationen auch mit heutiger Berechnungskapazität noch sehr zeitaufwändig, zum anderen ist die Erstellung der Simulationsmodelle sehr komplex. Weitere Hinderungsgründe stellen sowohl die eingeschränkte Wiederverwendbarkeit der einzelnen Modelle als auch die Notwendigkeit von Simulationsexperten sowie hohe Kosten ein dar. [Dav02]

Aus diesem Grund werden diese Problemstellungen in vielen Produktentwicklungen nicht oder erst spät nach der Konzeptionsphase simuliert.

1.3 Iterativer Simulationszyklus

In heutigen Entwicklungsprozessen erfolgt die Simulation in iterativer Abfolge mit einer Konzeptionsphase (siehe Bild 1). Eine Iteration lässt sich hierbei im Groben in vier Phasen untergliedern: Zuerst wird ein neues Konzept erstellt, welche als Grundmodell für die Simulation dienen. Darauf folgend muss dieses Modell (CAD-Daten) für die Simulation aufbereitet werden. In der Simulationsvorbereitung werden beispielsweise die Randbedingungen spezifiziert und die Fluiddomäne vernetzt. Auf dieser Basis erfolgt der eigentliche Simulationsdurchlauf. Nach Abschluss des Prozessschrittes werden

die Ergebnisse und Resultate dargestellt. Die daraus gewonnen Erkenntnisse fließen anschließend in eine Verbesserung der Konzepte, woraufhin die Änderungen in einem neuen Zyklus validiert werden. [Hed13], [Lec14]

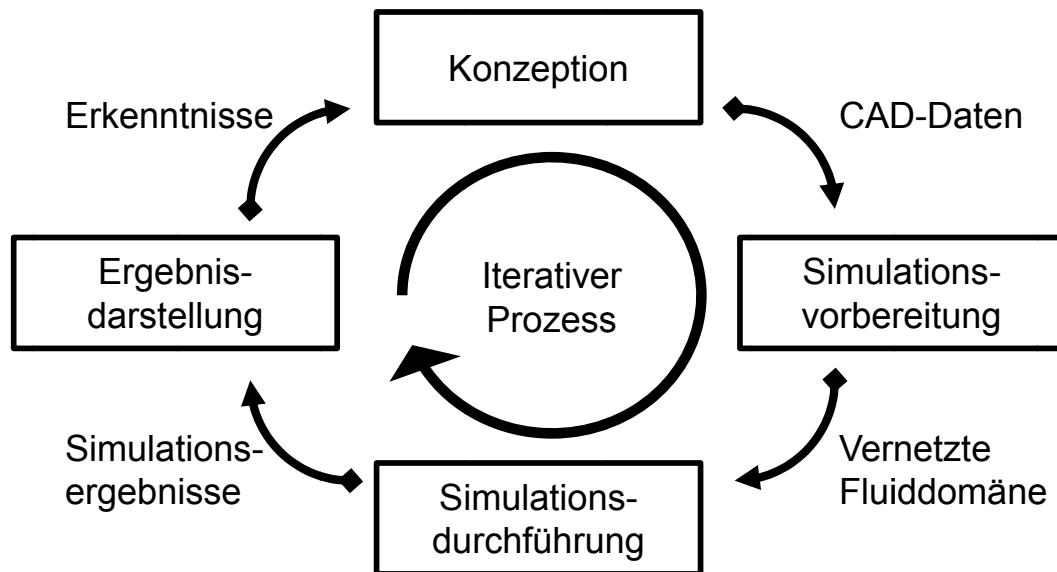


Bild 1: Iterativer Simulationsprozess im Produktentwicklungsprozess

2 Computerarchitektur

Die Dauer der Simulationsdurchführung hängt im Wesentlichen von zwei Faktoren ab, zum einen ist die angewandte Simulationmethode wichtig (vgl. Kapitel 3), zum anderen die Hardware der Rechnerarchitektur.

Diese hat sich in den letzten Jahren merklich verändert. In Bild 2 ist die Entwicklung der Berechnungsgeschwindigkeit der Central Processing Unit (CPU) und der Graphics Processing Unit (GPU) in GFlops (Giga Floating operation points per second \triangleq Milliarden-Gleitkommazahlberechnungen pro Sekunde) aufgezeigt. Wie zu sehen ist, ist in den letzten Jahren die mögliche Geschwindigkeit der GPU im Gegensatz der CPU erheblich gestiegen. Dies ist vornehmlich auf Grund der unterschiedlichen Hardwarearchitekturen zu erklären. Während die CPU der Generalist ist, welche unterschiedlichste Aufgaben mit hoher Taktrate erfüllt, ist die GPU als ein Spezialist entworfen, welche die gleiche Aufgabe tausendfach parallel in kürzester Zeit ausführen kann. [LCC+10]

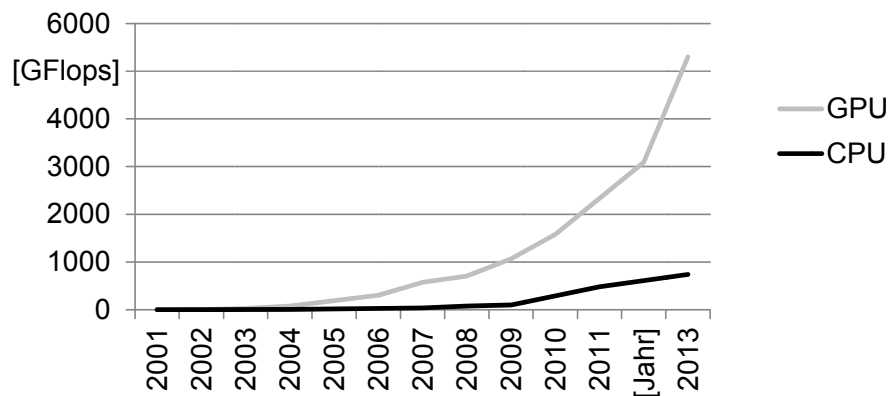


Bild 2: Historische Entwicklung der Rechenkapazität (Bild in Anlehnung an [Nvi14-ol])

3 Heutige Methoden zur Simulation von Fluiden

Bei fluiddynamischen Problemstellungen bildet die Navier-Stokes Gleichung die Grundlage für die Simulationsmethoden. Auf Grund ihrer Komplexität sind analytische Lösungen bzw. Näherungen nur in Ausnahmefällen möglich. Daher wurden mehrere numerische Lösungsmöglichkeiten entwickelt, welche durch die Diskretisierung komplexer Simulationsdomänen die Problemstellungen vereinfachen und lösbar machen. Die Methoden zur Lösung der numerischen Strömungsmechanik lassen sich in diesem Zusammenhang grundsätzlich in die zwei Bezugsräume der Eulersche- und den Lagrange-Methoden unterscheiden.

3.1 Eulersche-Methoden

Eulersche-Methoden basieren auf einer Diskretisierung der Fluidomäne durch eine Vernetzung des gesamten Raumes. Diese stellen die Grenzen für das Fluid dar. Die Granularität und Qualität der Diskretisierung entscheidet zum einen über die Ergebnisgenauigkeit, zum anderen ist es eine bestimmende Größe für die Berechnungszeit [Kor09]. Um den Grad der Diskretisierung zu reduzieren wurden mehrere Methoden entwickelt. Die Direkte Numerische Simulation (DNS) abstrahiert die Fluidomäne nicht, daher ist eine sehr feingranulare Vernetzung notwendig [GRF10]. Einen höheren Abstraktionsgrad besitzen die Methoden der Large Eddy Simulation [SD09]. Da auch hier die Berechnungszeiten für die industrielle Nutzung zu hoch sind, wird in der Industrie hauptsächlich die Reynolds-Averaged-Navier-Stokes (RANS) Methode eingesetzt [Dav02].

Um trotzdem den Anforderungen von hoher Genauigkeit, hoher Flexibilität und geringer Berechnungszeit gerecht zu werden, wurden unterschiedliche Verfahren zur Verbesserung der RANS-Methode entwickelt. Beispielsweise können lokale Fehler der Be-

rechnungsverfahren durch adaptive Netzverfeinerungen an kritischen Stellen reduziert werden [Chu02].

Zusammenfassend sind Eulersche-Methoden jedoch mit dem Fokus einer hohen Präzision entwickelt worden. Eine starke Beschleunigung der Berechnungsgeschwindigkeit wird durch mehrere inhärente Eigenschaften beschränkt. So ist zum einen die hohe Komplexität in der Parallelisierbarkeit zu nennen, zum anderen unterliegt die Zeitschrittweite auf Grund der Notwendigkeit einer hohen Stabilität starken Restriktionen. [KSR14]

Zudem ist auf Grund der Notwendigkeit der Vernetzung der gesamten Fluiddomäne die Flexibilität stark eingeschränkt. Bei Änderungen des Eingangsmodells muss die Simulationvorbereitung nochmals gänzlich durchgeführt werden. Darüber hinaus muss auf Grund ungelöster Probleme weiterhin ein Experte die Resultate bewerten [Dav02], [HH04], da trotz einer langen Forschungshistorie beispielsweise das chaotische Verhalten von Turbulenzen noch immer große Probleme aufgibt [WKA+13].

3.2 Lagrangesche-Methoden

Im Gegensatz zu Eulersche-Methoden werden in Lagrangesche-Methoden die Fluiddomäne nicht als Ganzes diskretisiert, stattdessen werden die Fluide nur an ihrem Auftrittort durch Partikel repräsentiert. Dadurch können die methodeninhärenten Schwierigkeiten umgangen werden. Beispielsweise sind eine zeit- und berechnungsaufwändige Netzgenerierung sowie das adaptive Regenerieren der Netzqualität nicht notwendig. Darüber hinaus können sich bewegende Randbedingungen wie bei Mehrphasenströmungen ohne Probleme modelliert werden. [Liu10], [NRB+08].

Zur Lösung von fluidischen Problemstellungen gibt es zahlreiche Methoden. Auf Grund der stark fortgeschrittenen Forschung eignet sich beispielsweise die Smoothed Particle Hydrodynamics (SPH) Methode für den industriellen Einsatz. [LL03], [Mon05]

Ein großer Vorteil der SPH-Methode ist eine gute Skalierbarkeit und eine große Zeitschritttoleranz [ZSL+12], [SR09], [KSR14]. Auf Grund der Tatsache, dass die einzelnen Partikel in einem Zeitschritt weitestgehend unabhängig voneinander berechnet werden können, ist zudem eine hoher Parallelisierungsgrad der Berechnungen möglich. [CDB+11]

4 Online-Simulation von Fluiden in der Konzeptionsphase

Die Vorteile der Lagrangesche-Methoden können hin zu einer schnelleren Modellbildung und Simulationsdurchführung genutzt werden. Statt einer ganzheitlichen Diskretisierung der Fluiddomäne können die direkten Interdependenzen zwischen den Festkörpergeometrien und dem Fluid gelöst. Dadurch bedingt eine Änderung der Geometrie keiner neuen Vernetzung der gesamten Fluiddomäne.

Durch diese Umstellung der Simulation hin zu einer dynamischeren Erstellung von Modellen kann die Untersuchung unterschiedlicher Alternativen in der Konzeptionsphase verbessert werden. Wie in Bild 3 dargestellt ist, kann somit der iterative Prozess zwischen Konzeption und Simulation parallelisiert werden. Dies bedeutet dass Konzeptänderungen direkt in der Simulationsumgebung angepasst werden. Der eigentliche Simulationslauf wird jedoch nicht verändert oder neu gestartet, lediglich das zu untersuchende Modell wird angepasst. Darüber hinaus können auf der Basis einer echtzeitfähigen Simulationen die Auswirkungen der Konzeptänderungen und -modifikationen dem Anwender online zu visualisieren. Die Prozessschritte werden in Kapitel 4.3 weiter spezifiziert.

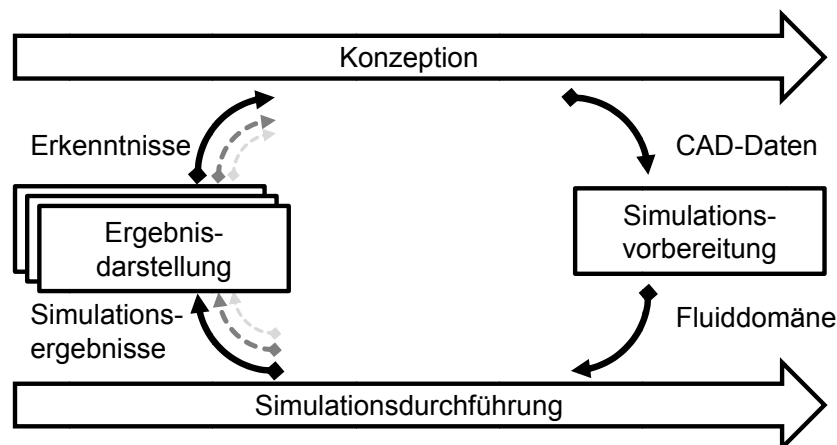


Bild 3: Online-Simulationsprozess in der Konzeptionsphase

4.1 Anforderungen für die Realisierung

Um diese Methode zu ermöglichen, bedarf es einer spezifischen Anpassung der Softwarestruktur, welche folgende Anforderungen erfüllt.

- Anforderung 1: Um der industriellen Diversität an Werkzeugen in der Konzeptionsphase gerecht zu werden, muss die Simulationsumgebung unabhängig von spezifischen Werkzeugen sein.
- Anforderung 2: Die Simulation von Fluiden ist sehr berechnungsaufwändig. Um Ergebnisse in Echtzeit zurückzugeben, ist es daher für die Simulation notwendig, eine möglichst schnelle und effiziente Berechnungszeit mit möglichst hoher Zeitschritttoleranz unter stabilen Bedingungen zu gewährleisten.
- Anforderung 3: Die Simulationsvorbereitung, d. h. die Aufbereitung der Modellgeometrien, soll einen möglichst geringen Einfluss auf die eigentliche Simulationsdurchführung haben.
- Anforderung 4: Modelle sollen online in der Simulation geändert werden können.
- Anforderung 5: Ergebnisse sollen während der laufenden Simulation für den Anwender bereitstehen und visualisiert werden.

4.2 Systemtechnische Beschreibung der Prozessschritte

4.2.1 Anforderungsspezifische Aufteilung der Hardwareressourcen

Zur Erfüllung der Anforderungen müssen die Methodenschritte auf die unterschiedlichen Ressourcen GPU und CPU aufgeteilt werden (siehe. Tabelle 1). Um die Anforderung 2 zu erfüllen, wird die eigentliche Berechnung der Simulationsdurchführung komplett auf die GPU ausgelagert. Dies hat mehrere Gründe: Zum einen besitzt diese in modernen Rechnern die höchste Rechenkapazität (vgl. Kapitel 2), zum anderen werden die Berechnungen von azyklischen Einflussfaktoren wie dem Import neuer Geometrien in der Simulationsvorbereitung nicht beeinflusst. Die Simulationsvorbereitung erfolgt daher rein auf der CPU. Die Ergebnisdarstellung wird je nach Auswertungsanfragen auf der GPU oder CPU berechnet.

4.2.2 Simulationsdurchführung

Vergleicht man die für die Simulationsdurchführung betreffenden Anforderungen (2, 3, 4) mit den möglichen Methoden der Fluidsimulation, so eignen sich Lagrangesche Methoden aus den Gründen höherer Parallelisierbarkeit sowie größerer Zeitschritttoleranz [KSR14]. Darüber hinaus entfällt die Notwendigkeit der Vernetzung, wodurch Geometrien während der Simulationslaufzeit angepasst werden können. Modifizierte Geometrien werden stattdessen als sich ändernde Randbedingungen modelliert. Darüber hinaus ist in Bereichen, in denen keine Fluidpartikel sind, die Änderung der Geometrie möglich.

Tabelle 1: Aufteilung der Berechnungsaufgaben auf Hardwareressourcen

Methodenschritt	Auftreten		Hardwareressource	
	zyklisch	azyklisch	CPU	GPU
Simulationsvorbereitung		X	X	
Simulationsdurchführung	X			X
Ergebnisdarstellung	X		X	X

4.2.3 Simulationsvorbereitung

Die Simulationsvorbereitung dient der Aufbereitung der Eingangsdaten für die Simulation. Prinzipiell kann in drei unterschiedliche Simulationsbestandteile unterschieden werden, welche sich hauptsächlich durch deren Definition der Randbedingungen unterscheiden.

Der Erste ist eine Quelle, in welcher neue Fluidpartikel generiert werden. Die Instanzierungslokalität wird dabei durch die Oberfläche einer 2-dimensionalen Geometrie defi-

niert. Darüber hinaus beinhaltet die Quelle fluidspezifische Parameter wie Viskosität, Dichte u. a.

Darüber hinaus werden nach einem ähnlichen Prinzip Senken zur Eliminierung von Fluidpartikeln genutzt. Diese sind essentiell, da die Berechnungsgeschwindigkeit maßgeblich von der Partikelanzahl abhängt. Diese können über 3-dimensionale Geometrien definiert werden. Das Fluid wird dabei bei Kontakt mit diesen virtuellen Simulationspartizipanten gelöscht.

Als letzte Möglichkeit sind die aus dem Konzeptionswerkzeug stammenden Geometrien zu nennen. Diese werden in eine tesselierte Oberflächengeometrie transformiert. Den einzelnen Dreiecken werden die vom Anwender definierten Oberflächeneigenschaften zugewiesen. Werden die Geometrien online modifiziert, so wird der Unterschied zwischen der vorhandenen und der neuen Geometrie analysiert. Aus Performancegründen werden auf dieser Basis nur die jeweils notwendigen Oberflächengeometrien aus der Simulationsumgebung gelöscht bzw. neu hinzugefügt.

Die Aufbereitung dieser Geometrien sowie die Berechnung erfolgt auf der CPU. Alle Änderungen werden dabei auf eine Kopie der Simulationsdatenstruktur eingepflegt. Nachdem dies geschehen ist, werden diese auf die GPU übertragen. Durch dieses Vorgehen wird der Simulationszyklus der eigentlichen Simulation nicht direkt beeinflusst. Auf Modellierungsebene haben somit nur noch die Größe und Komplexität der Fluiddomäne einen Einfluss auf die Berechnungsgeschwindigkeit.

4.2.4 Ergebnisdarstellung

Um die Vorteile der Online-Simulation dem Anwender zu ermöglichen, müssen die Zwischenergebnisse der Simulation dem Nutzer direkt aufbereitet werden. Für diesen Prozessschritt werden fallspezifisch sowohl die GPU als auch die CPU genutzt. So wird eine partikelspezifische Darstellung direkt von der GPU (oder einer zweiten GPU) berechnet und auf der graphischen Ausgabe gerendert. Im Gegensatz dazu werden Auswertungen wie beispielsweise Minima- und Maximaberechnungen auf der CPU durchgeführt.

5 Softwaretechnische Umsetzung

Eine softwaretechnische Umsetzung einer Online-Simulation von Fluiden muss mehrere Möglichkeiten bereitstellen. Zum einen müssen die Konzepte, d. h. beispielsweise die Geometrien, echtzeitnah in die Simulation importiert werden, zum anderen müssen Simulationsparameter wie Quellen und Senken, Fluidspezifika wie Dichte und Viskosität und geometriespezifische Haftbedingungen vorab oder online definiert werden. Zur Umsetzung der Online-Konzeption wurde ein modulweiser Aufbau gewählt. Bild 4 zeigt das Simulationsmodell zur Online-Simulation während der Konzeption. Aus den Konzeptionstools werden in allen Phasen permanent Modelle extrahiert und in ein Si-

Simulationsmodell konvertiert. Auf Basis der vom Nutzer konfigurierten Simulationsparameter erfolgen eine dauerhafte Simulation des aktuellen Modelles sowie eine permanente Ergebnisvisualisierung auf einer graphischen Oberfläche. Die Simulationsvorbereitung, -durchführung und Ausgabe der Ergebnisse werden im Simulationskern durchgeführt.

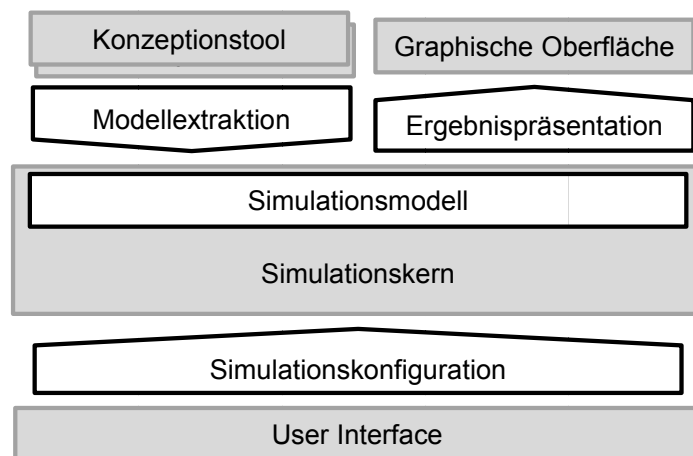


Bild 4: Prototypische Softwarestruktur zur Online-Simulation von fluidischen Prozessen

6 Beispielhafte Umsetzung

Die beschriebene Methode wurde in einem produktionstechnischen Entwicklungsprozess bei der Entwicklung eines Abfüllvorganges umgesetzt. Als Lagrangesche-Simulationsmethode wurde die SPH-Methode auf Grund ihrer breiten Verfügbarkeit sowie hohen Parallelisierbarkeit [NK12] gewählt. Zur Erstellung der geometrischen Konzepte wurde SolidWorks genutzt. Bild 5 zeigt einen beispielhaften Ablauf des Konzeptionsprozesses beim Entwurf eines Sammelbeckens. In Schritt 1 wurde eine Quelle für das Simulationsprogramm gewählt. Als Basis für das Sammelbecken wurde ein Würfel konstruiert (2). Dieser wird durch den Prozessschritt der Simulationvorbereitung aufbereitet und direkt in die Simulation eingefügt (3). Ein Neustart ist nicht notwendig. In Schritt 4 und 6 erfolgen weitere Konstruktionsschritte. Die Änderung werden individuell sofort in die Simulation importiert, woraufhin sich das Verhalten des Fluid an die neuen Geometrien online anpasst (5,7).

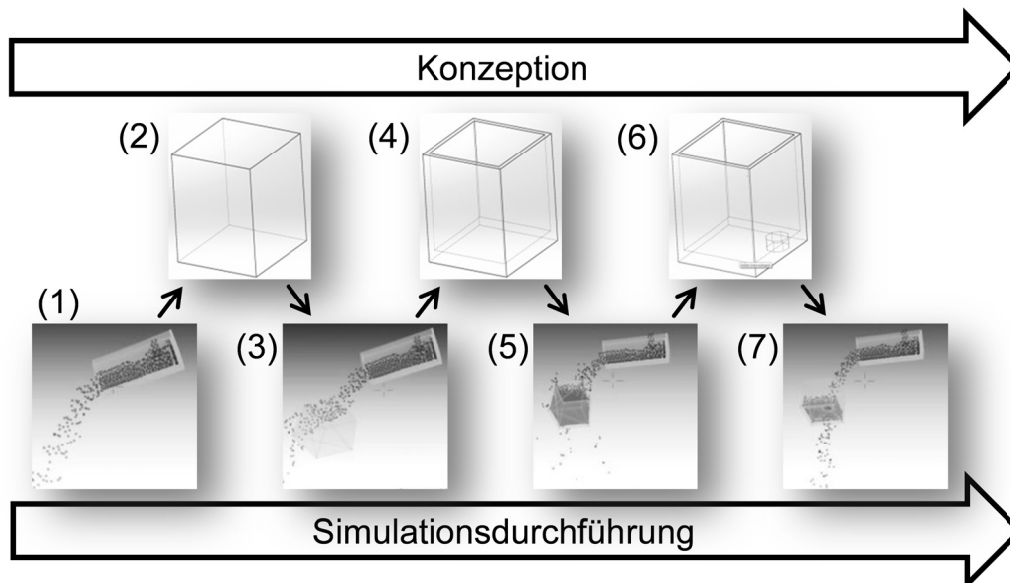


Bild 5: Umsetzungsbeispiel für die Online-Simulation in der Konzeptionsphase

6.1 Nutzen der Online-Simulation

Die dargestellte Methode bietet vor allem folgende Vorteile gegenüber dem bisherigen iterativen Ablauf:

- **Ein Simulationsdurchlauf muss nicht gänzlich berechnet werden:** Durch die direkte Online-Darstellung der Simulationsergebnisse während des Simulationslaufes können auch nach einem Bruchteil der Simulationsdauer erste Erkenntnisse gewonnen werden. Bisherige Werkzeuge beschränken die Ergebnisdarstellung meist auf das Ende der Simulationsdurchführung.
- **Die Dauer eines Konzeptionszyklus reduziert sich:** Dieser Vorteil resultiert direkt aus der Eigenschaft, dass während des Simulationslaufes Erkenntnisse gewonnen werden können. Wird ein fehlerhaftes Verhalten festgestellt, so muss nicht auf weitere Simulationsergebnisse gewartet werden, stattdessen kann sofort das Grundmodell nach den Erkenntnissen angepasst werden.
- **Konzeptänderungen haben sofortige und direkte Auswirkung auf die Simulation:** Die Simulation läuft parallel zum Konzeptionsprogramm. Änderungen an der Geometrie können direkt in die laufende Simulation gegeben werden. Somit ist im Gegensatz zur bisherigen Methode ein Neustart der Simulation nur eine Option. Darüber hinaus können kleine Anpassungen auf einfache Weise validiert werden.

7 Resümee und Ausblick

Es wurde eine Methode beschrieben, mit deren Hilfe eine echtzeitfähige Simulation von Fluiden möglich ist. Darüber hinaus können die Simulationsmodelle permanent geändert, wodurch eine sofortige Auswertung von Konzepten ermöglicht wird.

Bisher stellen sowohl die Latenz der Modellaufbereitung als auch die Berechnungsgeschwindigkeit stellt für die Methode einen kritischen, beschränkenden Faktor dar. Aus diesem Grund werden weitere Möglichkeiten zu einer strukturierteren Modellaufbereitung erforscht. Den Engpass der Berechnungsgeschwindigkeit wird durch stetig bessere Computerhardware reduziert, jedoch werden Performanceverbesserungen durch eine simulationsmethodenspezifische Modellaufbereitung untersucht.

Danksagung

Die Autoren danken dem Bayerischen Staatsministerium für Wirtschaft, Infrastruktur, Verkehr und Technologie sowie dem Freistaat Bayern für die Förderung des Projektes.

Literatur

- [AR11] ABELE E.; REINHART G.: Zukunft der Produktion. Hanser, München, 2011
- [Chu02] CHUNG T. J.: Computational Fluid Dynamics. Cambridge University Press, England, 2002
- [CDB+11] CRESPO, A. C.; DOMINGUEZ, J. M.; BARREIRO, A.; GÓMEZ-GESTEIRA, M.; ROGERS, B. D.: GPUs, a new tool of acceleration in CFD: efficiency and reliability on smoothed particle hydrodynamics methods. PloS one 6.6, 2011
- [Dav02] DAVIDSON D. L.: The Role of Computational Fluid Dynamics in Process Industries. The Bride Vol. 32 No. 4, S. 9-14, 2002
- [Döb08] DÖBLER T.: Simulation und Visualisierung in der Produktentwicklung. MFG-Stiftung Baden-Württemberg, Deutschland, 2008
- [EKL03] EHRENSPIEL K.; KIEWERT A.; LINDEMANN U.: Kostengünstig Entwickeln und Konstruieren. Springer, Berlin, 2003
- [ERZ14] EIGNER M.; ROUBANOV D.; ZAIFROV R.: Modellbasierte virtuelle Produktentwicklung. Springer, Berlin, 2014
- [GRF10] GEORGIADIS N. J.; RIZZETTA D. P.; FUREBY C.: Large-Eddy Simulation: Current Capabilities, Recommended Practices, and Future Research. AIAA Journal Vol. 48 No. 8, S. 1772-1784, 2010
- [Hed13] HEDTSTÜCK U.: Simulation diskreter Prozesse. Springer, Deutschland, 2013
- [HH04] HÖLLING M.; HERWIG H.: CFD-TODAY: Anmerkungen zum kritischen Umgang mit kommerziellen Software-Programmpaketen. Forschung im Ingenieurwesen Vol. 68 No. 3, S. 150-154, 2004
- [Kor09] KORTELAJNEN J.: Meshing tools for open source CFD-a practical point of view. VTT, Espoo, Finland, 2009
- [KSR14] KROTIL S.; STICH P.; REINHART G.: Anforderungen und Einsatzmöglichkeiten der Fluidsimulation in der frühen Phase des Produktentwicklungsprozesses. ASIM 2014: 22. Symposium Simulationstechnik, Tagungsband Teil 2, S. 259-266, 2014

- [Lec14] LECHER S.: Numerische Strömungsberechnung. Springer, Deutschland, 2014
- [LCC+10] LEE, V. W.; CHANGKYU K.; CHHUGANI J.; DEISHER M.; DAEHYUN K.; NGUYEN A. D.; SMELYANSKIY M.; CHENNUPATY S.; HAMMARLUND P.; SINGHAL R.; DUBEY P.: Debunking the 100X GPU vs. CPU myth: an evaluation of throughput computing on CPU and GPU. ACM SIGARCH Computer Architecture News, Vol. 38. No. 3, ACM, 2010
- [LL03] LIU G. R.; LIU M. B.: Smoothed Particle Hydrodynamics – a meshfree particle method. World Scientific Publishing, Singapur, 2003
- [Liu10] LIU G. R.: Meshless Methods. CRC press, USA, 2010
- [Mon05] Monaghan J. J.: Smoothed particle hydrodynamics. Reports on progress in physics 68.8, S. 1703-1759, 2005
- [NRB+08] NGUYEN V. P.; RABCZUK T.; BORDAS S.; DUFLOT M.: Meshless methods: a review and computer implementation aspects. Mathematics and computers in simulation, Vol. 79 No. 3, S. 763-813, 2008
- [NK12] NULI, U. A.; KULKARNI, P. J.: Sph based fluid animation using cuda enabled gpu. International Journal of Computer Graphics and Animation, 2012
- [Nvi14-ol] NVIDIA CORPORATION: CUDA C Programming Guide. Unter: <http://docs.nvidia.com/cuda/cuda-c-programming-guide/>, 6. November 2014
- [SD09] SAGAUT P.; DECK S.: Large eddy simulation for aerodynamics: status and perspective. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 367, S. 2849-2860, 2009
- [SR09] SOLENTHALER B.; RENATO P.: Predictive-corrective incompressible SPH. ACM transactions on graphics (TOG). Vol. 28 No. 3. ACM, S. 40, 2009
- [WKA+13] WANG Z. J.; KRZYSZTOF F.; ABGRALL R.; BASSI F.; CARAENI D.; CARY A. DECONINCK H.; HARTMANN R.; HILLEWAERT K.; HUYNH H. T.; KROLL N.; MAY G.; PERSSON P.-O.; VAN LEER B.; VISBAL M.: High-order CFD methods: current status and perspective. International Journal for Numerical Methods in Fluids Vol. 72: S. 811-845, 2013
- [ZSL+12] ZHANG F.; SHEN X.; LONG X.; ZHAO B.; HU L.: Real-time particle fluid simulation with wcsph. Pacific Graphics Short Papers, S. 29-34, 2012

Autoren

Stefan Krottil hat Elektro- und Informationstechnik von 2004–2008 an der FH Ingolstadt studiert. Anschließend vertiefte er sein Wissen durch ein Ergänzungsstudium an der TU München in selbiger Disziplin mit Abschluss im Jahr 2011. Seitdem arbeitet Herr Krottil bei der Projektgruppe RMV des Fraunhofer IWU in Augsburg und befasst sich mit der Simulationstechnik im produktionstechnischen Bereich.

Prof. Dr.-Ing. Gunther Reinhart ist Ordinarius für Betriebswissenschaften und Montagetechnik an der Technischen Universität München. Nach dem Maschinenbaustudium schloss er seine Promotion unter Prof. Dr.-Ing. J. Milberg ab. Nach mehreren leitenden Tätigkeiten in der Industrie wurde Prof. Reinhart 1993 auf den Lehrstuhl für Betriebswissenschaften und Montagetechnik an der Technischen Universität München und in die Leitung des *iwb* berufen.

Gleichzeitig ist er Vorstandsvorsitzender des Bayerischen Clusters für Mechatronik und Automation e.V. Seit dem 1. Januar 2009 ist Prof. Reinhart darüber hinaus Leiter der

Fraunhofer IWU Projektgruppe für Ressourceneffiziente Mechatronische Verarbeitungsmaschinen (RMV) in Augsburg.

Modellbasierte Analyse und Simulation industrieller Großwäschereien

*Dipl.-Ing. Alexander A. Albers, M.Sc. Christian Bremer,
Dr.-Ing. Torsten Bruns, Dr.-Ing. Roman Dumitrescu, Dr.-Ing. Martin Krüger*
Fraunhofer IP – Projektgruppe Entwurfstechnik Mechatronik
Zukunftsmeile 1, 33102 Paderborn
Tel. +49 (0) 5251 / 54 65 113, Fax. +49 (0) 5251 / 54 65 102
E-Mail: alexander.albers@ipt.fraunhofer.de

Zusammenfassung

In diesem Beitrag wird ein Ansatz für die modellbasierte Analyse und Simulation industrieller Großwäschereien präsentiert. Der Fokus liegt hierbei auf bestehenden Anlagen. Daher wird zunächst auf die effektive Aufnahme solcher Systeme eingegangen. Hierzu werden Methoden des Model-Based Systems Engineerings angewandt. Da bestehende Ansätze nicht den spezifischen Ansprüchen der hier vorliegenden Prozesssysteme genügen, wurde ein vierschrittiges Konzept entwickelt. Mit den hierin erarbeiteten Partialmodellen lässt sich das betrachtete Prozesssystem ganzheitlich beschreiben; es stellt außerdem die Basis für die Ableitung von Simulationsmodellen dar.

Zur Simulation und Analyse wird eine Modellbibliothek vorgestellt, aus deren Komponenten beliebige industrielle Großwäschereien im Hinblick auf die interne Wäschelogistik abgebildet und simulationsbasiert analysiert werden können. Anhand eines (validierten) Simulationsmodells lässt sich damit eine Wäscherei bzgl. der Auslastung ihrer einzelnen Verarbeitungsmaschinen, ihrer Speichereinrichtungen und bzgl. der Durchlaufzeiten von Wäscheposten analysieren und bewerten. Mit diesem Modell ist die Grundlage zur mathematischen Optimierung industrieller Wäschereien bereitet.

Schlüsselworte

Industrie 4.0, Systems Engineering, Produktionssystem, Industrielle Wäscherei, MBSE

Model based analysis and simulation of industrial laundries

Abstract

In this paper, the model-based analysis and simulation of large industrial laundries is presented. The focus is on existing plants. It first discusses the effective capture of these systems. For this purpose, methods of model-based systems engineering are applied. Since existing approaches do not meet the specific needs of the present process systems, a four-step approach was developed. With the developed four-step approach and the partial models, the process system can be described holistically; it also provides the basis for the development of simulation models.

A model library for simulation and analysis is presented. With the components of the library arbitrary industrial laundries can be represented concerning to their internal logistics by a simulation model. Such a (validated) model makes it possible to analyze and evaluate a laundry with regard to the utilization of its processing machines, its storage facilities and the processing times of the laundry batches. Last but not least the model is the prerequisite for the mathematical optimization of industrial laundries.

Keywords

Industry 4.0, Systems Engineering, Production System, Industrial Laundry, MBSE

Dieser Beitrag ist in der Online-Version
leider nicht verfügbar.

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, Mobilität, Vernetzung: Eine neue Schule des Entwurfs der 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 mit insgesamt 200 Mitarbeiterinnen und Mitarbeitern. Etwa ein Viertel der Forschungsprojekte der Universität Paderborn entfallen auf das Heinz Nixdorf Institut und pro Jahr promovieren hier etwa 30 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, Mobility, Integration: En-route to the 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 tomorrow's economy.

Today eight Professors and 200 researchers work at the Heinz Nixdorf Institute. The Heinz Nixdorf Institute accounts for approximately a quarter of the research projects of the University of Paderborn and per year approximately 30 young researchers receive a doctorate.

Zuletzt erschienene Bände der Verlagsschriftenreihe des Heinz Nixdorf Instituts

- Bd. 314 VON DETTEN, M.: Reengineering of Component-Based Software Systems in the Presence of Design Deficiencies. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 314, Paderborn, 2013 – ISBN 978-3-942647-33-5
- Bd. 315 MONTEALEGRE AGRAMONT, N. A.: Immunorepairing of Hardware Systems. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 315, Paderborn, 2013 – ISBN 978-3-942647-34-2
- Bd. 316 DANGELMAIER, W.; KLAAS, A.; LAROQUE, C.: Simulation in Produktion und Logistik 2013. HNI-Verlagsschriftenreihe, Band 316, Paderborn, 2013 – ISBN 978-3-942647-35-9
- Bd. 317 PRIESTERJAHN, C.: Analyzing Self-healing Operations in Mechatronic Systems. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 317, Paderborn, 2013 – ISBN 978-3-942647-36-6
- Bd. 318 GAUSEMEIER, J. (Hrsg.): Vorausschau und Technologieplanung. 9. Symposium für Vorausschau und Technologieplanung, Heinz Nixdorf Institut, 5. und 6. Dezember 2013, Berlin-Brandenburgische Akademie der Wissenschaften, Berlin, HNI-Verlagsschriftenreihe, Band 318, Paderborn, 2013 – ISBN 978-3-942647-37-3
- Bd. 319 GAUSEMEIER, S.: Ein Fahrerassistenzsystem zur prädiktiven Planung energie- und zeitoptimaler Geschwindigkeitsprofile mittels Mehrzieloptimierung. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 319, Paderborn, 2013 – ISBN 978-3-942647-38-0
- Bd. 320 GEISLER, J.: Selbstoptimierende Spurführung für ein neuartiges Schienenfahrzeug. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 320, Paderborn, 2013 – ISBN 978-3-942647-39-7
- Bd. 321 MÜNCH, E.: Selbstoptimierung verteilter mechatronischer Systeme auf Basis paretooptimaler Systemkonfigurationen. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 321, Paderborn, 2014 – ISBN 978-3-942647-40-3
- Bd. 322 RENKEN, H.: Acceleration of Material Flow Simulations - Using Model Coarsening by Token Sampling and Online Error Estimation and Accumulation Controlling. Dissertation, Fakultät für Wirtschaftswissenschaften, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 322, Paderborn, 2014 – ISBN 978-3-942647-41-0
- Bd. 323 KAGANOVA, E.: Robust solution to the CLSP and the DLSP with uncertain demand and online information base. Dissertation, Fakultät für Wirtschaftswissenschaften, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 323, Paderborn, 2014 – ISBN 978-3-942647-42-7
- Bd. 324 LEHNER, M.: Verfahren zur Entwicklung geschäftsmodell-orientierter Diversifikationsstrategien. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 324, Paderborn, 2014 – ISBN 978-3-942647-43-4
- Bd. 325 BRANDIS, R.: Systematik für die integrative Konzipierung der Montage auf Basis der Prinziplösung mechatronischer Systeme. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 325, Paderborn, 2014 – ISBN 978-3-942647-44-1
- Bd. 326 KÖSTER, O.: Systematik zur Entwicklung von Geschäftsmodellen in der Produktentstehung. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 326, Paderborn, 2014 – ISBN 978-3-942647-45-8
- Bd. 327 KAISER, L.: Rahmenwerk zur Modellierung einer plausiblen Systemstrukturen mechatronischer Systeme. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 327, Paderborn, 2014 – ISBN 978-3-942647-46-5
- Bd. 328 KRÜGER, M.: Parametrische Modellordnungsreduktion für hierarchische selbstoptimierende Systeme. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 328, Paderborn, 2014 – ISBN 978-3-942647-47-2

Zuletzt erschienene Bände der Verlagsschriftenreihe des Heinz Nixdorf Instituts

- Bd. 329 AMELUNXEN, H.: Fahrdynamikmodelle für Echtzeitsimulationen im komfortrelevanten Frequenzbereich. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 329, Paderborn, 2014 – ISBN 978-3-942647-48-9
- Bd. 330 KEIL, R.; SELKE, H. (Hrsg.): 20 Jahre Lernen mit dem World Wide Web. Technik und Bildung im Dialog. HNI-Verlagsschriftenreihe, Band 330, Paderborn, 2014 – ISBN 978-3-942647-49-6
- Bd. 331 HARTMANN, P.: Ein Beitrag zur Verhaltensantizipation und -regelung kognitiver mechatronischer Systeme bei langfristiger Planung und Ausführung. Dissertation, Fakultät für Wirtschaftswissenschaften, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 331, Paderborn, 2014 – ISBN 978-3-942647-50-2
- Bd. 332 ECHTERHOFF, N.: Systematik zur Planung von Cross-Industry-Innovationen. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 332, Paderborn, 2014 – ISBN 978-3-942647-51-9
- Bd. 333 HASSAN, B.: A Design Framework for Developing a Reconfigurable Driving Simulator. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 333, Paderborn, 2014 – ISBN 978-3-942647-52-6
- Bd. 334 GAUSEMEIER, J. (Hrsg.): Vorausschau und Technologieplanung. 10. Symposium für Vorausschau und Technologieplanung, Heinz Nixdorf Institut, 20. und 21. November 2014, Berlin-Brandenburgische Akademie der Wissenschaften, Berlin, HNI-Verlagsschriftenreihe, Band 334, Paderborn, 2014 – ISBN 978-3-942647-53-3
- Bd. 335 RIEKE, J.: Model Consistency Management for Systems Engineering. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 335, Paderborn, 2014 – ISBN 978-3-942647-54-0
- Bd. 336 HAGENKÖTTER, S.: Adaptive prozessintegrierte Qualitätsüberwachung von Ultraschalldrahtbondprozessen. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 336, Paderborn, 2014 – ISBN 978-3-942647-55-7
- Bd. 337 PEITZ, C.: Systematik zur Entwicklung einer produktlebenszyklusorientierten Geschäftsmodell-Roadmap. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 337, Paderborn, 2015 – ISBN 978-3-942647-56-4
- Bd. 338 WANG, R.: Integrated Planar Antenna Designs and Technologies for Millimeter-Wave Applications. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 338, Paderborn, 2015 – ISBN 978-3-942647-57-1
- Bd. 339 MAO, Y.: 245 GHz Subharmonic Receivers For Gas Spectroscopy in SiGe BiCMOS Technology. Dissertation, Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 339, Paderborn, 2015 – ISBN 978-3-942647-58-8
- Bd. 340 DOROCIAC, R.: Systematik zur frühzeitigen Absicherung der Sicherheit und Zuverlässigkeit fortschrittlicher mechatronischer Systeme. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 340, Paderborn, 2015 – ISBN 978-3-942647-59-5
- Bd. 341 BAUER, F.: Planungswerkzeug zur wissensbasierten Produktionssystemkonzipierung. Dissertation, Fakultät für Maschinenbau, Universität Paderborn, HNI-Verlagsschriftenreihe, Band 341, Paderborn, 2015 – ISBN 978-3-942647-60-1
- Bd. 342 GAUSEMEIER, J.; GRAFE, M.; MEYER AUF DER HEIDE, F. (Hrsg.): 12. Paderborner Workshop Augmented & Virtual Reality in der Produktentstehung. HNI-Verlagsschriftenreihe, Band 342, Paderborn, 2015 – ISBN 978-3-942647-61-8
- Bd. 343 GAUSEMEIER, J.; DUMITRESCU, R.; RAMMIG, F.; SCHÄFER, W.; TRÄCHTLER, A. (Hrsg.): 10. Paderborner Workshop Entwurf mechatronischer Systeme. HNI-Verlagsschriftenreihe, Band 343, Paderborn, 2015 – ISBN 978-3-942647-62-5