

Konzept für den MBSE-Einsatz zur automatisierten Individualisierung von komplexen Produkten

Fabian Dworschak¹, Christoph Zirngibl¹, Benjamin Schleich¹,
Sandro Wartzack¹

¹ *Engineering Design (KTmfk)*

Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Germany

Abstract

Due to increasing market transparency, companies are mainly focusing on development of customized products. In this course, a competitive advantage should be achieved through the automated adaptation of product characteristics. In order to achieve this, development engineers focus primarily on modifying existing designs and refrain from re-running the entire product development process (PDP). However, this approach is primarily limited to the CAD environment. Previous phases of the PDP are not integrated, leading to a lack of traceability and continuity. Media breaks and API conflicts arise which increase costs. In order to make use of this potential, this article shows a concept for integration of all phases of the RFLP approach into a tool for customization of adaptation constructions.

Keywords: Model Based Systems Engineering, RFLP, Design for X

1 Einleitung

Aufgrund der zunehmenden Markttransparenz setzen Unternehmen zur Wettbewerbsdifferenzierung verstärkt auf die Entwicklung kundenindividueller Produkte. Im Zuge dessen soll durch die möglichst automatisierte Anpassung der Produkteigenschaften ein Wettbewerbsvorteil erzielt werden. Um dies zu erreichen, setzen Entwicklungsingenieure vorrangig auf die Anpassung (Anpassungskonstruktion) vorhandener Baugruppen und verzichten auf den erneuten vollständigen Durchlauf (Neukonstruktion) des gesamten Produktentwicklungsprozesses (PEP) [1]. Dieses Vorgehen beschränkt sich jedoch vorrangig auf die CAD-Umgebung und berücksichtigt in den meisten Fällen lediglich die Produktausarbeitung. Die vorherigen Phasen des Entwicklungsprozesses sind hingegen nicht direkt integriert. Dies kann zu einer fehlenden Nachvollziehbarkeit und Durchgängigkeit zwischen den Prozessschritten der Produktentwicklung führen. Folglich entstehen Medienbrüche und Schnittstellenkonflikte, welche den PEP verlangsamen und schwer nachvollziehbar machen. Demzufolge werden die Möglichkeiten auf Produkthanpassungen ausreichend schnell reagieren zu können eingeschränkt und letztendlich die Entwicklungskosten erhöht. Um Medienbrüche zu vermeiden, müssen alle Phasen des PEP in einem einheitlichen Modell abgebildet werden. Hierfür bietet das modelbasierte Systems Engineering (MBSE) die Möglichkeit interdisziplinär und sprachenunabhängig Systemzusammenhänge graphisch darzustellen. Die Anwendungsmöglichkeiten für MBSE in der Produktentwicklung stellen ein weitreichendes und neuartiges Forschungsgebiet dar. Bisher wurde hierfür kein einheitliches Konzept oder Vorgehen definiert.

Dieser Beitrag stellt folglich ein mögliches Konzept für die Modellierung des PEP mit den Methoden des MBSE für Anpassungskonstruktionen vor. Es hebt sich gegenüber anderen Konzepten in drei Punkten ab. Erstens, verbindet das Konzept alle Phasen des RFLP-Ansatzes (Requirements, Functions, Logics, Physics) ohne Medienbrüche und Schnittstellenkonflikte konsistent mit Hilfe einer Datenbanklösung. Zweitens, werden die Zugriffsrechte des Anwenders durch eine graphische Nutzeroberfläche (GUI) reguliert. Drittens, unterstützt das Konzept durch die digital durchgängige Verknüpfung der Ebenen eine automatisierte und transparente Anpassung und Individualisierung des Produktes. Das vorgestellte Konzept soll keine generalisierte Methodik für die Anwendung von MBSE in der Produktentwicklung definieren. Vielmehr ist das Ziel, hinsichtlich der genannten Herausforderungen zu sensibilisieren und eine konzeptionelle Lösung vorzustellen.

Der nachfolgende Beitrag ist folgendermaßen gegliedert. Zunächst werden die verwendeten Methoden und der Stand der Technik in Abschnitt 2

vorgestellt. Aus den dargelegten Forschungslücken werden anschließend die Forschungsfrage und Zielsetzung abgeleitet und in Abschnitt 3 definiert. Anschließend wird in Abschnitt 4 das Konzept allgemein vorgestellt, bevor es anhand eines spezifischen Tools für ein mechatronisches Bauteil demonstriert wird. Im letzten Abschnitt erfolgt eine Zusammenfassung und Diskussion.

2 Hintergrund und Stand der Forschung

Bevor das erarbeitete Konzept im Detail vorgestellt wird, soll in diesem Abschnitt ein kurzer Überblick über die Integration des RFLP-Ansatzes innerhalb des PEP sowie die Modellierungspotentiale durch MBSE-Tools gegeben werden. Abschließend wird eine Auswahl an Konzepten für MBSE in der Produktentwicklung vorgestellt.

2.1 Strukturierung des PEP nach dem RFLP-Ansatz

Wie bereits zu Beginn beschrieben werden Anpassungen einzelner Funktionen und Elemente (Anpassungskonstruktion) sowie die Erstellung parametrischer Modelle (Variantenkonstruktion) vorrangig innerhalb der rechnergestützten Konstruktionsumgebung vorgenommen. Die resultierende Angleichung aller Dokumente der jeweiligen Produktlebensphasen stellt hierbei jedoch immer noch einen großen Aufwand dar [2]. Speziell in der interdisziplinären Produktentwicklung zeigen sich Herausforderungen in der durchgängigen Übertragung von Informationen zwischen den einzelnen Domänen [3]. Durch den RFLP-Ansatz kann auf Basis des Systems Engineering eine umfassende Unterstützung innerhalb des PEP realisiert werden. Die Abkürzung RFLP steht hierbei für Requirements (Anforderungsmodell), Functions (Funktionsmodell), Logics (Logisches Modell) sowie Physics (Physikalisches Modell). Im Allgemeinen beschreibt der Ansatz die systematische Produktentwicklung von der Analyse des Systems bis zur physikalischen Umsetzung [4]. Die Einordnung erfolgt in Bezug auf die VDI-Richtlinie 2206 im absteigenden Bereich (Synthese) des V-Modells. Die erste Ebene, auch als Anforderungsdefinition bezeichnet, beinhaltet die Anforderungsspezifikationen und umfasst das Requirements Engineering. Hierbei werden die Kundenanforderungen und Wünsche an das zu entwickelnde Objekt identifiziert und verwaltet. Als Methode eignen sich hierfür beispielsweise Feldbeobachtungen [5]. In der Funktionsanalyse findet eine Erweiterung des Anforderungsmodells statt. Die Funktionen werden hierfür zur Konzeptionierung und Spezifikation aus den Anforderungen abgeleitet. Als Basis für die Entwicklung der Funktionsstruktur dienen die erzeugten Haupt- und Unterfunktionen, welche zu Gesamt- oder Hauptfunktionen vereinigt werden können. Die Art dieser Verbindung kann je nach Anwendung variieren. Übereinstimmend mit der Konstruktionslehre nach Pahl/Beitz werden Verbindungen der Kategorie

Datenfluss als Stoff-, Signal- beziehungsweise Energiefluss dargestellt [6]. Resultierende Kontrollflüsse eignen sich hierbei zur Beschreibung der logischen Abhängigkeiten in der Funktionsebene und zur Deaktivierung oder Aktivierung unterschiedlicher Funktionen. In der dritten Ebene, der sogenannten logischen Ebene, wird für das System ein Architekturkonzept erstellt. Dieses beinhaltet Lösungen, welche in Form von Wirkprinzipien die zuvor festgelegten logischen Elemente charakterisieren. Anschließend wird jedem logischen Element ein modelliertes Systemverhalten zugewiesen. Dadurch kann ein Gesamtsystem erzeugt werden, welches ausreichend Inhalte für eine Simulationsdurchführung aufweist. Die detaillierte Ausarbeitung eines virtuellen Produktes erfolgt in der letzten Phase des RFLP-Ansatzes. Auf Basis eines 3D-CAD-Modells wird das logische Verhaltensmodell um physikalische Eigenschaften ergänzt.

2.2 Simulation und Modellierung im Systems Engineering

Bevor die Modellierung eines Systems im Detail beschrieben wird, müssen zuerst die grundlegenden Begriffe *Modell* und *System* definiert werden.

Für den folgenden Beitrag beschreibt der Begriff *Modell* eine eingeschränkte beziehungsweise abstrakte Darstellung der Realität [7]. Als wichtig gilt hierbei eine gute Balance aus den Begriffen Reduktion (filtern irrelevanter Details) und Abgrenzung (filtern irrelevanter Merkmale). Resultierend können komplizierte und/oder komplexe Systeme genauer beschrieben und für den Nutzer verständlicher abgebildet werden. Aufbauend darauf, definiert der Begriff *System* die theoretische Modellierung einer Gesamtheit von Elementen, welche durch Beziehungen zueinander verknüpft sind [8]. Die Anzahl der Beziehungen charakterisieren die Struktur, welche durch eine Systemgrenze von der Umwelt abgegrenzt wird. Dabei wird unterschieden, ob das System als offen oder geschlossen definiert ist. Im technischen Bereich wird ein System häufig durch die Betrachtung des Ein- und Ausgangsverhalten beschrieben. Hierbei tritt ein determiniertes Eingangssignal auf und führt zu Reaktionen innerhalb des Gesamtsystems.

Der Ausdruck des Systems Engineering (SE) wird von der internationalen Gesellschaft für Systems Engineering (INCOSE) als ein interdisziplinäres Vorgehen definiert, welches den Anwender bei der Entwicklung (technischer Systeme) unterstützt [9]. Der Fokus liegt hierbei auf der interdisziplinären Anwendbarkeit. Allgemein betrachtet das SE den ganzen Produktlebenszyklus, angefangen von der Entwurfsphase bis zur Produktentsorgung. Wobei die Überführung von Dokumenten in ein IT-interpretierbares Modell als eines der Hauptziele verfolgt wird. Das modellbasierte Systems Engineering (engl. *model-based System Engineering*) ist eine Vorgehensweise des SE, in welcher Informationen

durch eine strukturierte Modellierung verarbeitet werden. Generell sollen Systeme über den gesamten Lebenszyklus in den Bereichen Anforderungen, Entwurf, Analyse, Verifikation und Validierung unterstützt werden. Für die abstrakte und durchgängige Abbildung von Aspekten eines technischen Systems wird die interdisziplinäre Systemmodellierung verwendet. Diese setzt sich allgemein aus den drei Hauptbestandteilen: Modellierungssprache, Methode und Softwarewerkzeug zusammen [8]. Als Sprache wird in diesem Beitrag die Systems Modeling Language (SysML) verwendet, da diese neben standardisierten Diagrammarten (beispielsweise Blockdefinitionsdiagramme) eine einheitliche Notation realisiert. Neben der Verwendung des RFLP-Ansatzes als Methode, wird auf die Wahl des Softwarewerkzeuges in einem gesonderten Abschnitt eingegangen. Die Unterstützung des PEP durch MBSE wurde bereits in vorangegangenen Forschungsarbeiten untersucht. Beispielsweise die Anwendung im Design of Variety [10] oder in der Baukastenentwicklung [11] zeigten sich vielversprechende Resultate.

3 Forschungsproblem und Forschungsziel

Wie einleitend erläutert, ist die Anpassung von Produkten nach wie vor eine der Hauptaufgaben in der Produktentwicklung [12]. Obwohl viele Prozessschritte über die Produktgenerationen hinweg redundant ablaufen, fehlt es an automatisierten Abläufen. Die fehlende gesamtheitliche Digitalisierung der einzelnen Entwicklungsschritte und deren einheitliche Verknüpfung sind zwei der größten Herausforderungen für eine durchgängige Automatisierung der Produktpassung. Um diese zu überwinden, bietet das modellbasierte Systems Engineering Methoden für eine transparente und durchgängige Darstellung des Produktlebenszyklusses. Diese finden derzeit jedoch noch zu wenig Anwendung in den Produktentwicklungsprozessen, da mit deren Integration ein Umdenken vom dokumentenzentrierten hin zu modellzentrierten Prozessen einhergeht [13]. Um diese Lücke zu schließen, demonstriert der nachfolgende Beitrag ein Konzept für die Integration von MBSE in den Produktentwicklungsprozess. Im Vergleich mit anderen Ansätzen konzentriert sich die Vorgehensweise auf die Erstellung von durchgängig verknüpften, rechnerverarbeitbaren Modellen für jede Phase der Produktentwicklung. Obwohl sich deren Grundlagen auf wenige Methodiken zurückführen lassen, kann der eigentliche Prozess zwischen den Unternehmen stark variieren. Folglich wird das Konzept generisch dargestellt und anschließend anhand eines Use Cases veranschaulicht. Somit soll die Forschungsfrage beantwortet werden, wie die Methoden des MBSE für die Automatisierung des Produktentwicklungsprozesses von individualisierten Produkten eingesetzt werden können.

4 Konzept: Integration von MBSE in den PEP

Der Kern dieses Beitrags ist die Darstellung eines Konzepts zur durchgängig digitalisierten Verknüpfung des RFLP-Ansatzes durch den Einsatz von MBSE-Methoden. Es ermöglicht eine vollständige Darstellung aller Ebenen des RFLP-Ansatzes und gewährleistet eine konsistente und transparente Anpassung dieser für neue Anforderungskonfigurationen. Das Konzept ist in Bild 1 als Übersicht dargestellt.

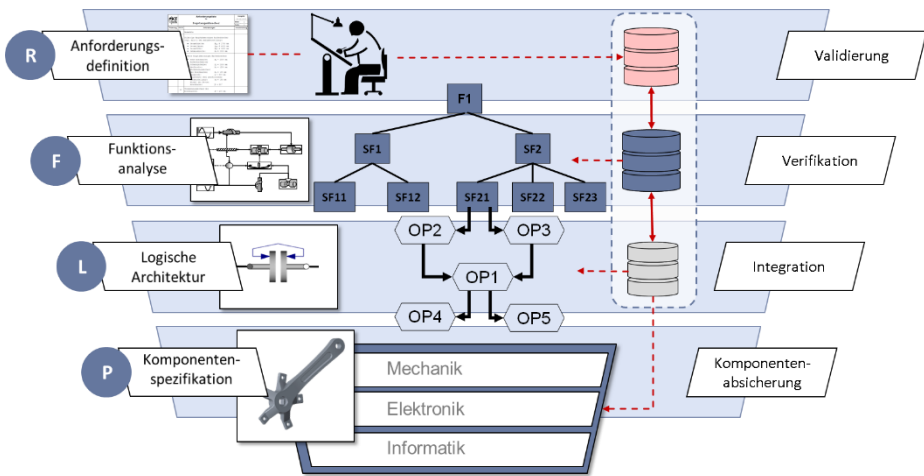


Bild 1: Konzept für den Einsatz von MBSE innerhalb des RFLP-Ansatzes

Zunächst wird die Anforderungsebene im Gegensatz zur klassischen Anforderungsliste durch eine graphische Benutzeroberfläche realisiert. Diese bietet einerseits dem Produktentwickler die Möglichkeit, auf neue Anforderungskonfigurationen direkt und anwenderfreundlich zu reagieren. Andererseits definiert die GUI die für eine Anpassungskonstruktion notwendigen Schnittstellen zur Datenbank und kontrolliert die Lese- und Schreibrechte im Backend. Die dort abgelegten Informationen bilden die kohärente Grundlage für die Modellierung der nachfolgenden Ebenen (FLP) und ermöglichen den anschließenden Wissenstransfer zwischen den einzelnen Entwicklungsschritten. Hierzu zählt unter anderem die Abbildung der funktionalen Ebene durch eine Entwicklungsumgebung mit integrierter Erweiterung für MBSE. Diese bietet dem Nutzer die Möglichkeit Diagramme mit der MBSE-Modellierungssprache SysML darzustellen und somit eine einheitliche Modellierungsnotation zu erzielen. Die Grundlage bildet hierbei die konsistente Abbildung sämtlicher Gesamtfunktionen und deren Teilfunktionen in einer hierarchischen Struktur. Für die automatisierte

Darstellung der logischen Ebene wird ebenfalls die SysML als Modellierungssprache verwendet. Dabei werden alle Wirkprinzipien in Form von Wirkstrukturdiagrammen abgebildet. Ziel ist die konsistente Beschreibung der Zusammenhänge zwischen stofflichen und geometrischen Wirkfaktoren, sowie deren Wechselwirkungen innerhalb des Prozesses. Die Informationen werden dabei aus dem Backend geladen und die Diagramme selbstständig auf Veränderungen innerhalb der Anforderungsebene angepasst. Für die automatisierte Generierung eines 3D-CAD-Modells (physikalische Ebene) wird eine Schnittstelle zwischen dem Front- und Backend sowie dem CAD Programm benötigt. Der wesentliche Fokus liegt hierbei auf dem Transfer und der Verarbeitung der durch die GUI eingepflegten individuellen Anforderungen. Erforderliche Berechnungen und Veränderungen innerhalb des 3D-CAD-Modells sollen selbstständig und durchgängig über alle Schnittstellen hinweg vorgenommen und übertragen werden. Durch die durchgängige Verknüpfung der einzelnen Ebenen des RFLP-Ansatzes kann somit gewährleistet werden, dass keine Medienbrüche oder Schnittstellenkonflikte die Individualisierung unterbrechen.

5 Anwendungsbeispiel: Generierung einer Leistungskurbel

Aus umwelttechnischen Aspekten nimmt das Fahrrad für viele Menschen eine immer größer werdende Rolle als bevorzugtes Verkehrsmittel ein. Hierbei gehört die Möglichkeit zur CO₂-Reduzierung beziehungsweise Stau-Vermeidung zu den größten Vorteilen des Fahrrades gegenüber dem PKW. Neben der Steigerung des Fahrkomforts für den täglichen Weg zur Arbeit, sind insbesondere ambitionierte Sportler stets auf der Suche nach der Steigerung der eigenen Leistungsprozente. Neben der körperlichen Fitness ist vor allem die richtige Konfiguration der verwendeten Rennradkomponenten und deren Relationen zum Nutzer ein entscheidender Faktor. Üblicherweise wird in sogenannten *Bike-Fittings* für den leistungsorientierten Nutzer – meist Bahn-, Rennrad-, MTB-Sportler oder Triathlet – ein auf ihn abgestimmtes Konzept erarbeitet. Dabei gibt es konfigurierbare Teile. Beispielsweise können die Sattelhöhe, die Länge des Vorbaus sowie dessen Überhöhung mit wenigen Handgriffen angepasst werden. Neben der Auswahl der richtigen Sitzposition beziehungsweise der Rahmenhöhe existieren Komponenten die nur mit deutlich höherem Aufwand angepasst werden können. Beispiele hierfür sind additiv gefertigte Griffe, Sättel oder Adapter zur Erweiterung der Standardgrößen. Eine weitere Komponente mit Optimierungspotential ist die Kurbel, da auch sie bei der Kraftübertragung zwischen Nutzer und Fahrrad involviert ist. Darüber hinaus entscheiden sich viele Rennradfahrer für die Verwendung einer Leistungskurbel. Hierbei handelt es sich um ein mechatronisches Bauteil, welches eine elektronische Sensoreinheit und den Kurbelarm zu einem Leistungsmesssystem kombiniert. Der Anwender hat dadurch die Möglichkeit, sich direkt Angaben zu seiner aktuellen

Fahrtintensität auf einem Fahrradcomputer anzeigen zu lassen. Bei der Auswahl der Kurbelarmlänge muss jedoch auf die gängigen Abstufungen der Hersteller zurückgegriffen werden. Diese liegen meist im Bereich von 170, 172,5 und 175mm. Eine durchgängige Anpassung der Bauteilkomponenten an die Nutzeranforderungen ist somit nicht möglich. Aus diesem Grund soll in diesem Beitrag die Individualisierung einer Leistungskurbel als Demonstrator für die Einbindung von MBSE in die Produktentwicklung dienen.

Die Requirements-Ebene wird für diesen Anwendungsfall durch eine in C# programmierte graphische Benutzeroberfläche (GUI) realisiert. Diese ist in Bild 2 dargestellt.

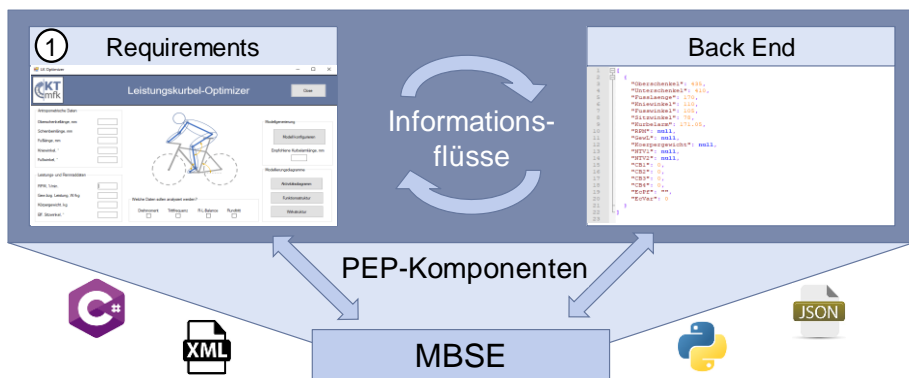


Bild 2: Darstellung des Front- und Backends

Für die Individualisierung der Kurbelarmlänge wird dem Nutzer die Möglichkeit für Änderungen an den anthropometrischen und leistungsbezogenen Personendaten direkt zur Verfügung gestellt. Der Konstrukteur nimmt hierbei Veränderungen an direkt beeinflussbaren Parameter vor, woraus Eigenschaften wie die Festigkeit oder das Gewicht der Leistungskurbel resultieren. Die übergebenen Informationen werden in eine JavaScript Object Notation Datei (JSON) gespeichert und bilden das Backend. Diese dient als Schnittstelle zur Übertragung und Regulierung sämtlicher Informationsflüsse. Folglich ist das Backend eine Datenbank auf welcher die Methoden des MBSE konsistent aufbauen. Hierfür wird zu Beginn durch den Produktentwickler eine Funktionsstruktur in Form von SysML-Blockdefinitionsdiagrammen (bdd) aufgebaut. Resultierend ergeben sich durchgängig verknüpfte, rechnerverarbeitbare Modelle, welche sämtliche Verzweigungen der jeweiligen Gesamtfunktionen enthalten und die Basis für eine anschließende individuelle Produkthanpassung bilden. Durch die GUI können unterschiedliche Anforderungen an die Leistungskurbel

(beispielsweise die Analyse der Trittfrequenz) konfiguriert und in das Backend übertragen werden. Die parametrische Modellierung der Diagramme und die Anpassung der jeweiligen Teilfunktionen an die neuen Nutzeranforderungen wird hierbei durch ein Python-basiertes Skript realisiert. Die dort hinterlegte Logik lädt zu Beginn Informationen aus dem Backend und iteriert anschließend mehrmals über die XML-Datei der Funktionsstruktur. Je nach Produktkonfiguration werden hierbei Diagrammzweige überschrieben oder hinzugefügt. Die resultierenden Diagramme werden dem Nutzer anschließend automatisch in der Entwicklungsumgebung Eclipse zur Verfügung gestellt. Für die Anpassung der Wirkstruktur werden die einzelnen Wirkprinzipien in einem ersten Schritt durch den Produktentwickler analysiert und ebenfalls als bdd modelliert. In Bild 3 ist hierfür beispielhaft eine Funktions- (2) und eine Wirkstruktur (3) dargestellt.

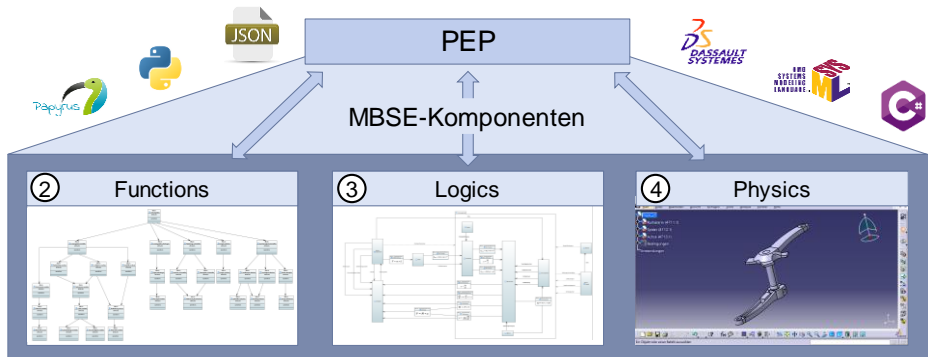


Bild 3: Darstellung der funktionalen, logischen und physikalischen Ebenen

Identisch zur Funktionsstruktur werden im zweiten Schritt die Informationen der zuvor definierten Produktkonfiguration aus dem Backend geladen. Die anschließende Realisierung der Wirkstrukturen erfolgt ebenfalls durch eine skriptbasierte Logik, welche definierte Diagrammbereiche auf die individuellen Spezifikationen anpasst. Nachfolgend werden die Diagramme wiederum automatisch in Eclipse geöffnet und dargestellt. Generell zeichnet sich das Konzept besonders durch die Gewährleistung einer durchgängigen Verknüpfung und der Vermeidung von Medienbrüchen oder Schnittstellenkonflikten, welche die Individualisierung unterbrechen, aus. Für die Generierung des 3D-CAD-Modells (physikalische Ebene) wird die Component Object Models (COM)-Schnittstelle Win32com verwendet. Diese ermöglicht es, eine Verknüpfung zwischen dem Front- und Backend sowie dem CAD Programm CATIA V5-6 zu erzeugen. Resultierend können Skizzen und Volumenkörper parametrisch aufgebaut und ein 3D-CAD-Modell der Leistungskurbel individuell modelliert werden.

6 Zusammenfassung und Diskussion

Im Kontext der Automatisierung für Anpassungskonstruktionen hat der vorliegende Beitrag ein Konzept zur durchgängig digitalisierten Verknüpfung des RFLP-Ansatzes durch den Einsatz von MBSE-Methoden vorgestellt. Der Vorteil gegenüber herkömmlichen Konstruktions-Templates oder Konstruktionsschablonen ist, dass alle Ebene des RFLP-Ansatzes direkt verknüpft sind und über einer einfachen Datenbank-Lösung konsistent gehalten werden. Diese wird ebenfalls genutzt, um die Zugriffsmöglichkeiten des Anwenders zu reglementieren. Das Konzept wurde an einem einfachen und spezifischen Beispiel, welches sich für die Individualisierung eignet, demonstriert. Für diesen bestimmten Anwendungsfall lassen sich Anpassungen schnell und automatisiert umsetzen. Der Demonstrator veranschaulicht die Möglichkeiten und Vorteile des MBSE für die Produktentwicklung. Allerdings zeigt sich auch, dass für einen spezifischen Anwendungsfall bereits sehr viel Entwicklungs- und Implementierungsaufwand notwendig ist. Folglich liefert dieser Beitrag, neben der detaillierten Lösung, abschließend Potentiale für die Weiterentwicklung des MBSE im Kontext der Produktentwicklung. Zum einen muss die Modellerstellung unterstützt werden, um den Implementierungsaufwand zu reduzieren. Da sich die Modelle in der Produktentwicklung schnell wiederholen, könnte ein modulares System in Form eines Diagrammbaukastens die Zeit für die Erstellung reduzieren. Auch ist die Reihenfolge der Diagramme häufig redundant, sodass musterbasierte Vorschläge für den folgenden Diagrammbaustein getroffen werden können. Mit Hilfe dieser zwei Ansätze kann die Generalisierung des gezeigten Konzeptes weiter vorangetrieben werden.

Danksagung

This research work is part of the FAU "Optimization-based design methodology in early phase of mechatronic product development" project (EFRE/OptMePro) and funded by the Bavarian program for the "Investment for growth and jobs" objective finance by the European Regional Development Fund (ERDF), 2014-2020. It is managed by the Bavarian Ministry of Economic Affairs and Media, Energy and Technology. The authors are responsible for the content of this publication.



Literatur

- [1] Naefe, P., Luderich, J.: „Anpassungskonstruktion. Konstruktionsmethodik für die Praxis“, Springer Verlag, Wiesbaden, 2016.
- [2] Böge, A., Böge, W.: „Handbuch Maschinenbau“, Springer Verlag, Wiesbaden, 2017.
- [3] Eigner, M. et al.: „Modellbasierte virtuelle Produktentwicklung“, Springer Verlag, Berlin/Heidelberg, 2014.
- [4] Kleiner, Sven.: „Entwerfen und Entwickeln mit Systems Engineering auf Basis des RFLP-Ansatzes in V6“, TUDpress – Verlag der Wissenschaft GmbH, Dresden, 2012.
- [5] Rupp, C.: „Requirements-Engineering und –Management: aus der Praxis von klassisch bis agil“, Carl Hanser Verlag, München, 2014.
- [6] Pahl, G. et al.: „Konstruktionslehre, Methoden und Anwendungen“, Springer-Verlag, Berlin, 2013.
- [7] Engeln, W.: "Produktentwicklung: Herausforderungen, Organisation, Prozesse, Methoden und Projekte", Vulkan-Verlag, Essen, 2018.
- [8] Alt, O.: „Modellbasierte Systementwicklung mit SysML“, Carl Hanser Verlag, München, 2012.
- [9] Walden, D. et al.: „INCOSE Systems Engineering Handbuch: Ein Leitfaden für Systemlebenszyklus-Prozesse und –Aktivitäten“, GfSE Verlag, München, 2017.
- [10] Hanna, M.: „Modelbasierte Unterstützung der Entwicklung modularer Produktfamilien“, Design for X, Beiträge zum 29. DfX-Symposium, Tutzing, 2018.
- [11] Bursać, N.: „Model Based Systems Engineering zur Unterstützung der Baukastenentwicklung im Kontext der frühen Phase der Produktgenerationsentwicklung. Dissertation“, Karlsruher Institut für Technologie (KIT), 2016.

-
- [12] Vajna, S.; Weber, C.; Zeman, K., Hehenberger, P., Gerhard, D., Wart-zack, S.: „CAx für Ingenieure: Eine praxisbezogene Einführung.“, 3. Auf-lage. Berlin, Heidelberg, 2018.
- [13] Scherer, H.: „Modellbasierte Methoden zur Modellierung des Zielsystems und des Funktions-Gestalt-Zusammenhangs zur Unterstützung der Se-rienenwicklung von Baukästen am Beispiel von Hybrid-Triebstrangsys-temen. Dissertation“, Forschungsberichte Institut für Produktentwick-lung am KIT Band 97. Karlsruhe: 2016.