

ANSATZ EINER WISSENSBASIERTEN BESCHREIBUNGSSPRACHE ZUR FUNKTIONSORIENTIERTEN BESCHREIBUNG VON HYBRIDEN LEISTUNGSBÜNDELN

Ewald G. Welp, Ralf Theiß, Tim Sadek, Matthias Köster

Zusammenfassung

Hybride Leistungsbündel (HLB)¹ folgen dem Grundgedanken, Kunden integrierte Leistungen bestehend aus domänenübergreifenden technischen Produktartefakten und industriellen Dienstleistungen zur Lösung von industriellen Problemstellungen anzubieten. Sie sind durch eine integrierte und sich gegenseitig determinierende Planung, Entwicklung, Erbringung und Nutzung von Sach- und Dienstleistungsanteilen einschließlich ihrer immanenten Softwarekomponenten gekennzeichnet [1]. Die Erfüllung der vom Kunden geforderten Leistung steht demnach im Vordergrund der Problemlösung, unabhängig davon, ob eine Leistung durch einen Produkt- oder Serviceanteil erfüllt wird. Dies führt dazu, eine starre Trennung bei der Entwicklung von Produkt- und Serviceanteilen durch eine integrierte Entwicklung der zu erbringenden Leistungen zu ersetzen [2]. Ausgangspunkt innerhalb der HLB-Konzeptentwicklung ist demnach eine funktionsorientierte Beschreibung der zu erbringenden Lösungsbestandteile unter Berücksichtigung der an sie gestellten Anforderungen und Restriktionen. Bisher verwendete Beschreibungssprachen weisen in dieser Hinsicht Defizite auf, z.B. durch eine einseitige Fokussierung auf Sach- oder auf Dienstleistungen, die durch den hier vorgeschlagenen Ansatz behoben werden sollen.

Gegenstand dieser Veröffentlichung ist die Darstellung eines Ansatzes für eine wissensbasierte Beschreibungssprache zur Abbildung der durch eine HLB-Problemlösung zu erfüllenden Funktionen mit ihren Anforderungen und Restriktionen sowie eine erste Umsetzung in einem rechnergestützten Anwendungsbeispiel. Grundlage für die Beschreibungssprache ist eine Notation bestehend aus Funktionselementen und Relationen, aus denen sowohl eine graphische Repräsentationsform für den Anwender als auch das Modell für die Implementierung in eine wissensbasierte Software abgeleitet werden können. Zur Implementierung des Modells werden geeignete Technologien des Semantic Web [3], wie z.B. Ontologien, ausgewählt und in einem Anwendungsbeispiel umgesetzt. Den Abschluss dieser Veröffentlichung bildet die Diskussion des Ansatzes anhand der mit dem Anwendungsbeispiel gesammelten Erfahrungen.

1 Einleitung

Die Entwicklung von Sach- und Dienstleistungen stellt eine im höchsten Maße kreative Tätigkeit dar, die in der heutigen Zeit intensiv wissensbasiert durchgeführt wird. Dies macht eine immer stärkere Unterstützung des Entwicklers durch wissensbasierte Methoden und Werkzeuge, insbesondere in Form wissensbasierter Software, erforderlich. Dabei ist in einem Entwicklungsprozess Wissen über vielfältige Bereiche zu berücksichtigen. Neben dem für das in der klassischen Entwicklung notwendige Wissen zu Anforderungen, Technologien und den im Unternehmen zur Verfügung stehenden Ressourcen wird verstärkt Wissen aus anderen Bereichen genutzt. Insbesondere Wissen aus der Produktnutzung sowie die Be-

¹ Hybride Leistungsbündel (HLB) werden im englischsprachigem Wissenschaftsbereich als Industrial Product Service Systems (IPS²) bezeichnet.

rücksichtigung von Wissen aus ökonomischen und juristischen Bereichen bis hin zu Wissen über den kulturellen Hintergrund der Käufer finden Eingang in die Produktentwicklung.

Hybride Leistungsbündel als Lösung für komplexe Problemstellungen wie z.B. für komplexe Fertigungsprobleme, zeichnen sich dadurch aus, dem Kunden nicht das Produkt und dessen Bestandteile zu verkaufen, sondern eine wandlungsfähige auf die jeweiligen Bedürfnisse des Kunden ausgerichtete Leistung bereitzustellen. Diese soll nicht nur durch die Verwendung von (neuen) Technologien sondern durch die gezielte Verbindung von Sach- und Dienstleistungen erreicht und erbracht werden. Das Ziel in der integrativen Entwicklung von HLB besteht darin, die einzelnen Bestandteile eines Produktes nicht mehr getrennt voneinander zu betrachten und zu entwickeln, sondern die bisher über die beteiligten Domänen verteilten Entwicklungsprozesse zusammenzuführen sowie bisher nicht berücksichtigte Bestandteile in einem umfassenden Entwicklungsprozess zu integrieren. Im Rahmen von HLB werden daher nicht allein Sach- und Dienstleistungskomponenten betrachtet sondern auch die zu deren Erbringung notwendige Infrastruktur. Weiterhin ist der HLB-Vertrag mit dem Kunden, bestehend aus einem innovativen Erlösmodell und der Vergabe der Eigentumsrechte für das betrachtete Geschäftsmodell notwendig.

Die Komplexität und der Umfang des durch diesen Anspruch notwendig werdenden Wissens sind für den einzelnen Entwickler und für Entwicklerteams nicht mehr handhabbar. Von daher wird es als notwendig angesehen, den Entwickler bei seiner Tätigkeit durch wissensbasierte Softwaresysteme gezielt zu unterstützen. Dabei sind neben der Bereitstellung von Wissen und der Validierung der modellierten Systeme im Rahmen der an sie gestellten Bedingungen weitere unterstützende Funktionen, wie z.B. eine teilautomatisierte Modellierung der Systeme auf Basis von vorhandenem Wissen erforderlich. Zudem ermöglicht die Anwendung von wissensbasierten Systemen das Ableiten von simulationsfähigen Modellen z.B. für die Kostenprognose oder für die genauere Planung der im HLB enthaltenen Dienstleistungsprozesse.

Als Grundlage für die gezielte Entwicklung von hybriden Leistungsbündeln sind zunächst die dafür notwendigen Funktionen zu ermitteln und in einem Modell nach Möglichkeit vollständig und konsistent zu beschreiben. Darin sind sowohl die durch den Kunden formulierten Anforderungen als auch die gesetzlichen und vertraglichen Bedingungen, in deren Rahmen das zu entwickelnde HLB erbracht wird, zu berücksichtigen. Hierfür wird ein neuartiger Ansatz zur funktionalen Beschreibung von hybriden Leistungsbündeln gewählt, der die in oder an einem System herrschenden Bedingungen (Restriktionen Domänen- und Lifecycle-übergreifender Art) als Grundlage der Funktionsbeschreibung heranzieht. Dazu wird zunächst in Kapitel 2 die konzipierte Beschreibungssprache mit ihren Elementen und Relationen erläutert. Daraus wird dann eine für den Menschen geeignete Repräsentationsform abgeleitet sowie eine erste Umsetzung in eine wissensbasierte Software auf Basis von Semantic Web Technologien, im speziellen Ontologien präsentiert. Den Abschluss bildet die Anwendung in einem Anwendungsbeispiel und der Diskussion der wesentlichen Vor- und Nachteile dieses Ansatzes.

2 Ansatz zur funktionalen Beschreibung von HLB

2.1 Bestehende Ansätze zur funktionalen Beschreibung von Systemen

Das klassische Verständnis des Funktionsbegriffs nach VDI 2221, wie es in Bild 1 abgebildet ist, fasst Funktionen als abstrakte Einheiten zur Umsetzung der Eingangs- in Ausgangsgrößen mit den drei Flussarten Energie, Stoff und Signal und einer noch unbekanntem Lösung auf. Dabei wird eine zu erfüllende Gesamtfunktion in Teilfunktionen detailliert und mittels Energie-, Stoff-, und Signalflüssen zu einem funktionalen, abstrakten Gesamtkonzept kombiniert. Dieser Ansatz ist geeignet, rein technische Systeme auf funktionaler Ebene zu be-

schreiben. Zur Integration von Dienstleistungsbestandteilen ist der Ansatz jedoch ungeeignet.

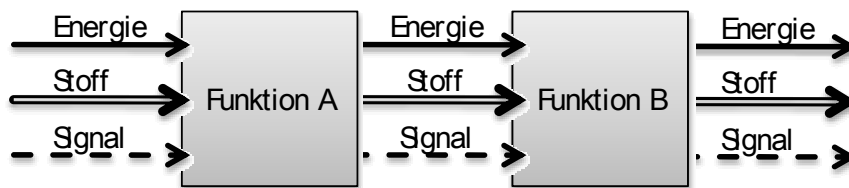


Bild 1: Funktionsbeschreibung nach VDI 2221

In Weiterführung dieses grundlegenden Ansatzes sind zwei Entwicklungen für die wissensunterstützte Beschreibung von Funktionszusammenhängen zu nennen. Zum einen lassen sich Funktionen aus abstrakten Metafunktionen, wie sie z.B. in [4] beschrieben werden, mit Hilfe von Ontologien ableiten und zum anderen können Ports als Schnittstellen zwischen den Funktionen bzw. mit Elementen zur Beschreibung des Verhaltens eines Systems wie in [5] beschrieben eingeführt werden. Ferner [4] beschreibt sechs Typen von Funktionen. Zu diesen zählen Typen wie z.B. *to Provide*, *to Control* (siehe Bild 2).



Bild 2: Funktionsbeschreibung nach [5]

Beide Ansätze zusammen repräsentieren das zurzeit vorherrschende Verständnis von Funktion und Funktionsstruktur. Es ermöglicht die wissensbasierte Modellierung von technischen Systemen sowohl mit Flüssen als auch mit komplexeren Relationen. Dabei werden weder direkte Beziehungen zu Anforderungen und Bedingungen noch die Möglichkeit zur Modellierung von Dienstleistungen berücksichtigt. Für die Beschreibung von sach- und dienstleistungsintegrierenden HLB sind das beschriebene Funktionsverständnis und die dabei verwendeten Relationen nicht mehr ausreichend. Sie bieten aber eine Grundlage für das in dieser Veröffentlichung erweiterte Verständnis der Beschreibung von HLB, dass die Funktionsebene als eine lösungsneutrale Beschreibung der durch das bzw. im HLB zu erbringenden Funktionen betrachtet. Die explizite Verwendung von Stoff-, Energie- und Signal-Flüssen auf einer funktionalen Ebene impliziert immer eine physikalische Lösung und widerspricht somit dem hier vorgestellten Verständnis. Im Rahmen des HLB-Konzeptmodells wird nicht auf eine Beschreibung der Flüsse verzichtet. Diese werden erst bei der Lösungssuche auf der HLB-Objekt- und der HLB-Prozess-Ebene betrachtet, die nicht Gegenstand dieser Veröffentlichung sind.

2.2 Grundlagen der Beschreibungssprache

Die Beschreibung der funktionalen Lösung eines HLB erfolgt innerhalb des heterogenen HLB-Konzeptmodells auf der HLB-Funktionsebene [6]. Die beiden wesentlichen Beschreibungselemente auf dieser Ebene sind Funktionselemente und Relationen. Zur Entwicklung der funktionalen HLB-Lösung werden die Funktionen aus Anforderungen und Restriktionen abgeleitet und über die Relationen miteinander verknüpft. Unter Anforderungen werden dabei nach [7] die angestrebten technischen und wirtschaftlichen Eigenschaften eines Systems verstanden, die mit Hilfe zweier Kriterien gegliedert werden. Zum einen lassen sich Anforderungen in funktionale wie nicht funktionale Anforderungen unterscheiden, wobei sich letztere nochmals in qualitative und quantitative Anforderungen, wie in Bild 3 dargestellt, differenzieren lassen. Funktionale Anforderungen beschreiben zumeist Aktionen des Systems oder in-

nerhalb eines Systems, die sich direkt als abstrakte Funktionen in eine Funktionsstruktur übertragen lassen. Diese Anforderungen werden dann durch die qualitativen und quantitativen Anforderungen parametrisiert.

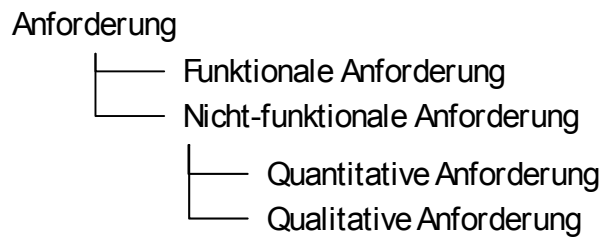


Bild 3: Klassifikation von Anforderungen

Das folgende Beispiel illustriert die Einteilung einer natürlich-sprachlich formulierten Anforderung in das o.g. Schema. Die Anforderung an ein HLB *100.000 Einheiten zu produzieren* setzt sich zusammen aus der funktionalen Anforderung *produzieren*, die zur Funktion *produzieren* wird, der qualitativen Anforderung *Einheiten* sowie der quantitativen Anforderung *100.000*. Innerhalb des Modelles werden zwei Arten von Relationen zur Verbindung von Funktionen verwendet. Die HLB-Lösung dient zur Bereitstellung einer vom Kunden geforderten Ausgangsleistung. Diese wird durch Transformation bestimmter Eingangsleistungen erreicht. Gleiches gilt für die Transformation eines materiellen Inputs in einen materiellen Output. Diese Transformation erfolgt durch die HLB-Systemfunktion. Diese beschreibt die Lösung auf abstrakte Weise auf dem niedrigsten Detaillierungsgrad. Eine Detaillierung dieser Funktion erfolgt durch eine Aggregation von Teilfunktionen. Demnach besteht beispielsweise in Anlehnung an das folgende Anwendungsbeispiel die Systemfunktion *Werkstück bearbeiten* aus weiteren Subsystemfunktionen wie etwa *Rotationsbewegung erzeugen*. Die Verknüpfung der HLB-Funktionen auf verschiedenen Detaillierungsebenen erfolgt über vertikale Relationen, die der Aggregation entsprechen.

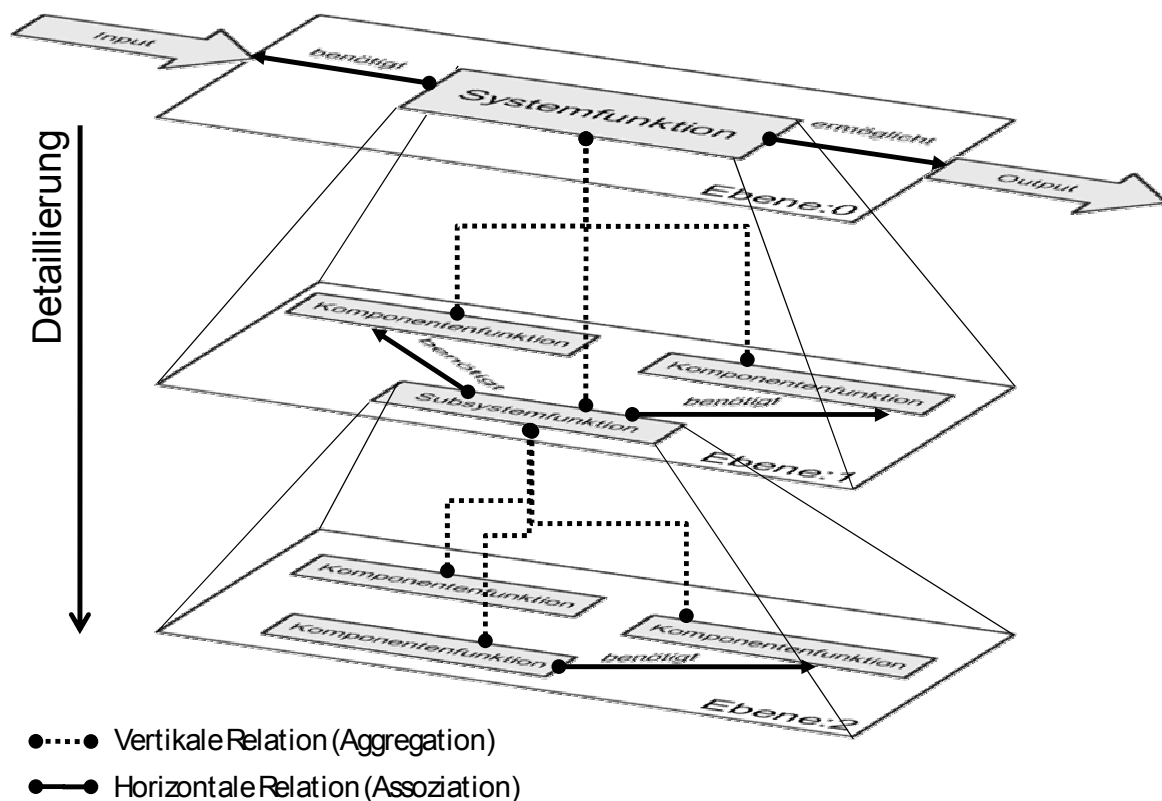


Bild 4: Relationen und Funktionen auf der HLB-Funktionsebene

Die Verknüpfung von HLB-Funktionen innerhalb einer Detaillierungsebene erfolgt über horizontale Relationen, die einer Assoziation entsprechen. Diese dienen insbesondere der Lösungsfindung und können je nach Leserichtung einen benötigenden oder ermöglichenden Charakter annehmen. So benötigt beispielsweise eine Funktion *Rotationsbewegung erzeugen* aufgrund von störenden Verschleißerscheinungen die weitere Funktion *Rotationsbewegung instandhalten*, damit die HLB-Lösung das im HLB-Vertrag festgelegte Leistungsversprechen einhalten kann. Ein Beispiel für die zuvor beschriebenen Relationen ist in Bild 5 abgebildet.



Bild 5: Horizontale Relation zwischen Funktionen

Anforderungen und Restriktionen, die als Basis der Funktionsermittlung dienen, werden in Form von Kontexten innerhalb der HLB-Funktionen abgebildet. Dies unterstützt einerseits durch die Sichtbarkeit der Anforderungen und Restriktionen auf der Funktionsebene den Entwickler bei der Lösungsfindung. Andererseits wird der sukzessive Charakter der HLB-Funktionsentwicklung betont. So werden notwendige Restriktionen erst durch die Strukturierung von Funktionen sichtbar und können direkt mit abgebildet werden. Die zweite Relation innerhalb des Modells ist die Aggregation. Sie verbindet die, im Gegensatz zur Assoziation, nicht gleichrangigen Funktionen sondern Funktionen die eine übergeordnete Funktion detaillieren. Ein Beispiel für die Detaillierung der Funktion *produzieren* stellen die Funktionen *Werkstücke anliefern* und *Werkstücke bearbeiten* dar. Diese können ebenfalls über Assoziationen miteinander verbunden sein. Bild 4 stellt das Konzept des Modells mit Funktionen und Relationen im Ganzen dar.

Die zweite Darstellungsform ist die Netzdarstellung gemäß Bild 6. Sie zeigt prinzipiell alle in einem Modell vorkommenden Relationen zwischen den Funktionen. Aufgrund der rasch zunehmenden Anzahl an Verknüpfungen, die selbst in einfachen Modellen auftreten, verliert die ungefilterte Darstellungsform schnell an Übersichtlichkeit. Aus diesem Grund müssen geeignete Filtermethoden entwickelt werden, die es dem Anwender ermöglichen, für eine Aufgabe gerade nicht relevante Anteile auszublenden und nur Teilmodelle zu selektieren. Eine Lösung dazu bietet der Einsatz von Semantic Web Technologien, mit deren Hilfe eine wissensbasierte Software Funktionen und Bedingungen bestimmten Kontexten, z.B. dem Antrieb (in Bild 6 dargestellt) oder sicherheitsrelevanten Komponenten zuordnen kann.

Zu beiden Darstellungsformen ist anzumerken, dass der Entwickler auf unterschiedliche Arten das Modell bearbeiten kann. So können von Hand Elemente eingefügt und Relationen gezogen werden oder die Elemente entsprechend mit Bedingungen ausreichend exakt beschreiben, damit eine rechnergestützte Generierung der Relationen erfolgen kann.

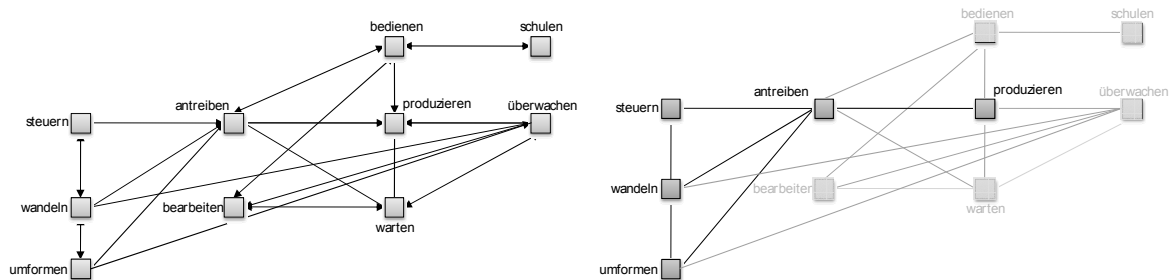


Bild 6: Netzdarstellung (links: vollständig, rechts: gefiltert)

2.3 Semantische Repräsentationsform für die wissensbasierte Funktionsmodellierung

Zur Unterstützung der Modellierung mit wissensbasierten Softwaresystemen, die auf Semantic Web Technologien basieren, ist es notwendig, das für die Entwicklung zu verwendende Wissen in Ontologien abzubilden. Durch die Abbildung des Wissens in einer maschinenlesbaren Form wird es den wissensbasierten Systemen ermöglicht, dem Anwender bei der Modellierung zu unterstützen. Beispiele hierfür sind das Vorschlagen von Funktionen zur Detaillierung oder von zu berücksichtigenden Anforderungen bis hin zur Konsistenzprüfung. Spätere Entwicklungen sollen aus der Funktionsstruktur simulationsfähige Modelle wissensbasiert ableiten. Basis hierfür bilden die Semantic Web Technologien, dabei insbesondere mittels OWL² beschriebene Ontologien. Für die Unterstützung der funktionalen Modellierung werden das Basiswissen der am Entwicklungsprozess beteiligten Domänen, Wissen zu den Strukturen sowie das aus bisherigen Entwicklungen gesammelte Wissen zur Kopplung von Anforderungen, Restriktionen und Funktionen verwendet. Dazu zählen z.B. geeignete Kombinationen aus Teilfunktionen, die eine übergeordnete Funktion erfüllen oder Anforderungen die in Verbindungen mit diesen verwendet werden.

Bild 7 zeigt den für die funktionale Modellierung verwendeten Ausschnitt aus einer am Lehrstuhl entwickelten Ontologie, die auch die Grundlage für mehrere andere Projekte darstellt. Diese Ontologie enthält neben den Elementen zur funktionalen Beschreibung auch Basiswissen aus der Domäne der Mechatronik (Systemelemente, Wirkprinzipien, physikalische Größen etc.). Die im Modell beschriebenen Funktionen, Anforderungen und Restriktionen sind dann Instanzen der in dieser Ontologie beschriebenen Klassen.

² Die Web Ontology Language (OWL) ist eine Beschreibungssprache zur Modellierung von Ontologien.

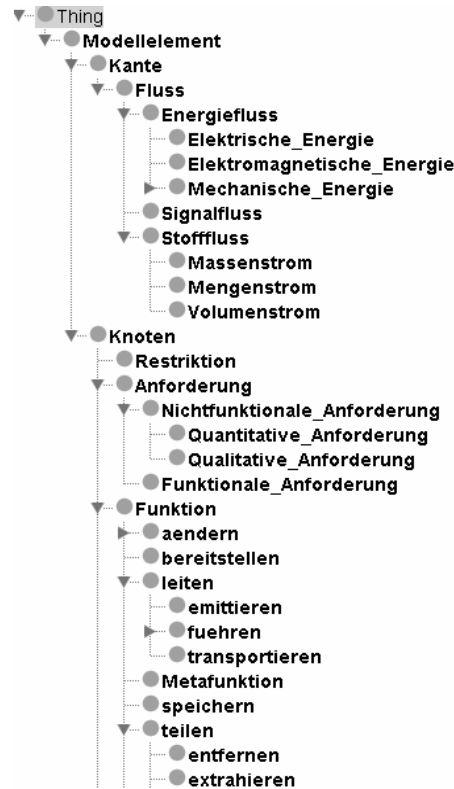


Bild 7: Ontologie zur funktionalen Beschreibung von HLB

Die hier dargestellte Ontologie bildet die Basis für die weitere Beschreibung und Integration des notwendigen Domänen- und Erfahrungswissens.

3 Evaluierung am Anwendungsbeispiel

Im Folgenden wird anhand eines Anwendungsbeispiels aus der Mikroproduktionstechnik die Problemlösung auf der HLB-Funktionsebene verdeutlicht. Ein Kunde sucht eine Problemlösung zur Fertigung rotationssymmetrischer Bauteile mit Abmessungen im Mikrometerbereich. Dazu wird von einem HLB-Anbieter eine Verfahrensvariante der Mikrofunkenerosion vorgeschlagen. Dieses Verfahren benötigt die Erweiterung einer Draht-EDM-Maschine um eine zusätzliche Rotationsachse, die die für die Bearbeitung notwendige Rotation des Werkstückes ermöglicht. Diese Rotationsbewegung soll durch die Integration eines tauchbaren Zusatzequipments, bestehend aus einer Rotationsspindel mit geeignetem Spannsystem und einer Werkstückaufnahme in den Maschinenraum der Draht-EDM-Maschine erzeugt werden. Als EDM-Maschine wird die kundeneigene Maschine verwendet. Das Zusatzequipment, das die Rotationsbewegung ermöglichen soll, soll von einem HLB-Anbieter in Form einer Verfügbarkeitsleistung dem Kunden bedarfsgerecht bereitgestellt werden. Die Ermittlung der zum einen technischen Funktionen des mechatronischen Systems Rotationsspindel und zum anderen der allgemeinen Funktionen, die zur Erfüllung des vom Anbieter angebotenen Leistungsversprechens notwendig sind, werden innerhalb der Konzeptentwicklung dieser HLB-Problemlösung auf der HLB-Funktionsebene abgebildet. (siehe Bild 8)

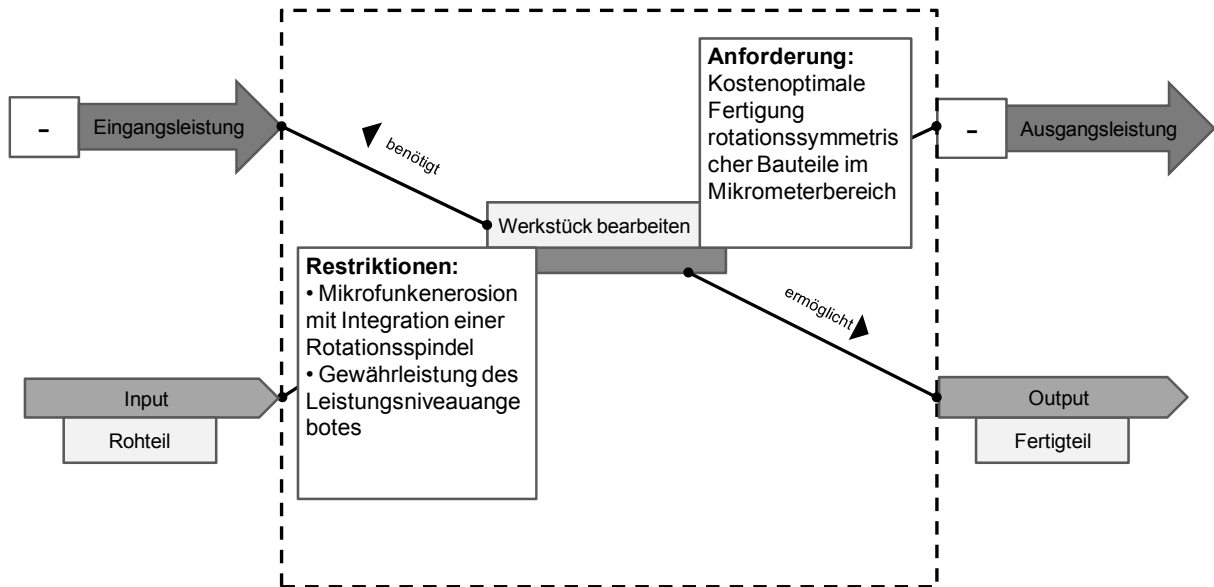


Bild 8: Systemfunktion

Diese Systemfunktion wird im Folgenden über eine Aggregation aus notwendigen Subsystem und Komponentenfunktionen detailliert.

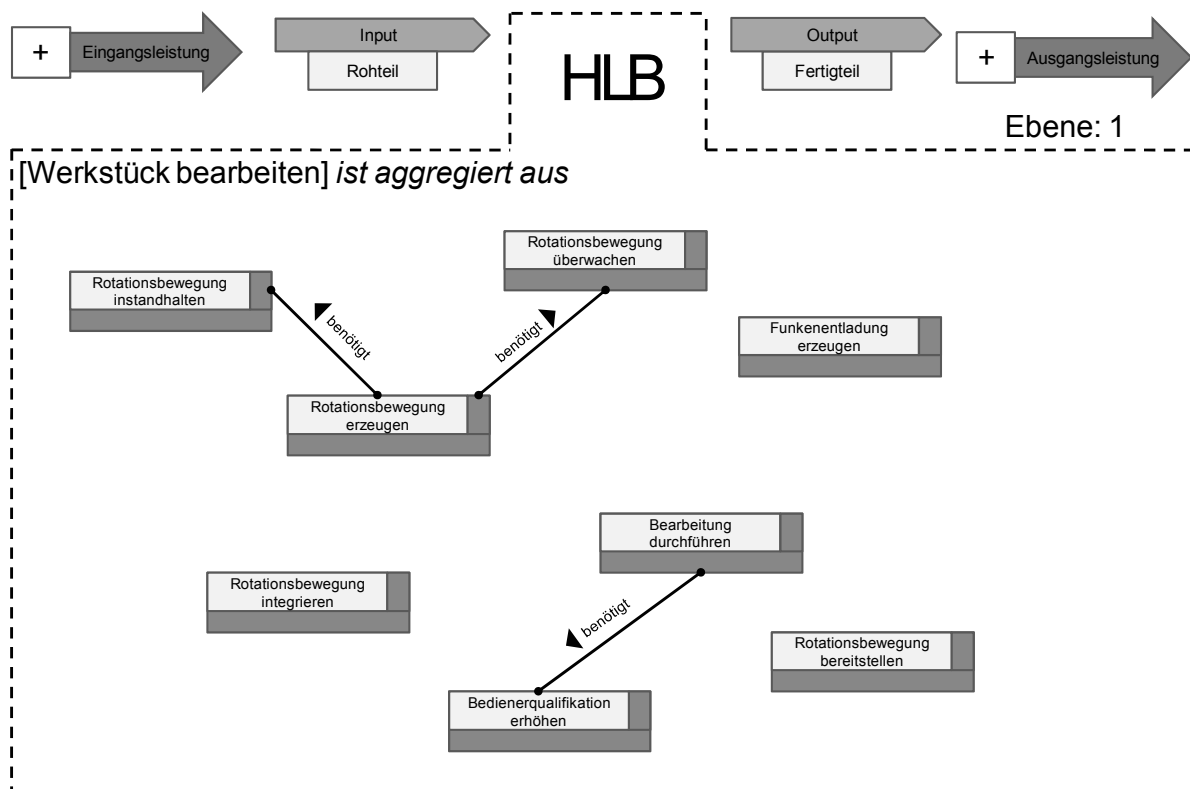


Bild 9: Detaillierung der Systemfunktion

Der Funktionsbaum, welcher sich aus der Detaillierung in den verschiedenen Modellierungsebenen ergibt, könnte wie in Bild 10 dargestellt, aussehen.

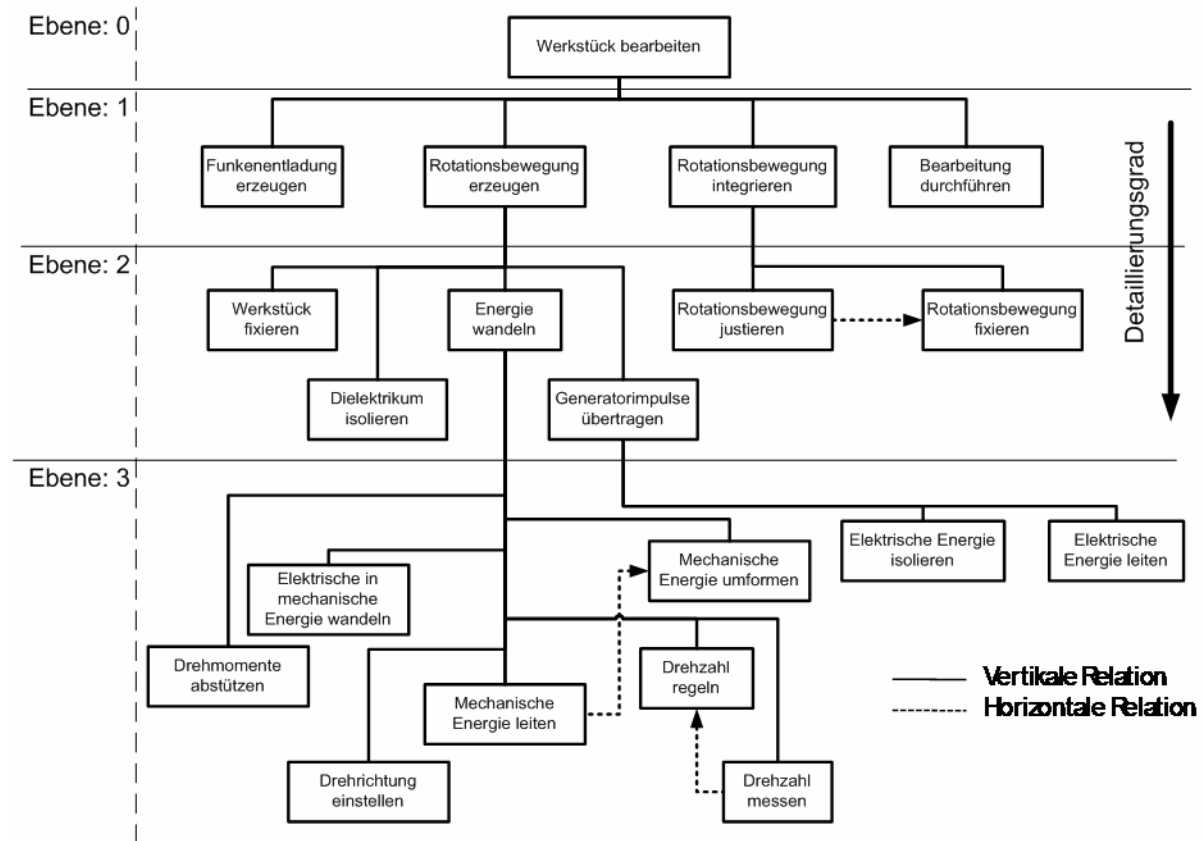


Bild 10: Funktionsbaum des Anwendungsbeispiels

4 Ausblick

Die mit dem Anwendungsbeispiel gesammelten Erfahrungen zeigen, dass der vorgestellte Ansatz die funktionale Modellierung von HLB ermöglicht, dabei jedoch eine wissensbasierte Unterstützung für den Anwender zur Verbesserung des Modellierungsprozesses als geeignet erscheint. Insbesondere ermöglicht der Ansatz, in einer integrierten Entwicklungsumgebung die Elemente der Funktionsstruktur zu modellieren und den Modellierungsprozess durch Semantic Web Services, die geeignete Vorschläge zur Detaillierung generieren, zu unterstützen. Ferner ist es möglich, auf Basis einer derart erstellten Funktionsstruktur, erste Objekte und Prozesselemente als Lösungselemente zur Erfüllung der geforderten Funktionen auszuwählen.

Zum anderen zeigen die mit dem Anwendungsbeispiel gesammelten Erfahrungen, dass sich die Schwierigkeiten im Umgang mit der Methodik im Wesentlichen auf zwei Bereiche konzentrieren. Zum einen ergeben sich Schwierigkeiten aus den in einem Modell enthaltenen Elementen und Relationen, da deren Anzahl bereits bei einem einfachen aus wenigen Funktionen bestehenden Modell schnell steigt. Somit ist es einem Entwickler ohne massive wissensbasierte Softwareunterstützung nicht möglich, komplexe Systeme aus Sach- und Dienstleistungen zu entwickeln und zu modellieren. Da der hier beschriebene Ansatz nur in Kombination mit geeigneten Softwarewerkzeugen möglich ist, werden besondere Anforderungen an diese Werkzeuge gestellt. Verbreitete Ontologie-Modellierungswerkzeuge wie z.B. Protege stoßen hierbei schnell an ihre Grenzen, da mit ihnen die Modellierung der Instanzen eine nicht unerhebliche Zeitdauer in Anspruch nimmt. Daher werden sich künftige Arbeiten mit der Entwicklung von Werkzeugen zur effizienten Modellierung sowie der Unterstützung bei der Formulierung von Instanzen befassen müssen.

Der zweite Bereich ist die Entwicklung der für die wissensbasierte Software notwendigen Ontologien. Sie sind gezielt für die einzelnen im Entwicklungsprozess verwendeten Domänen zu entwickeln. Für die Beschreibung von mechatronischen Systemen existieren bereits Ontologien, wohingegen Ontologien zur Beschreibung von Dienstleistungen, Bedingungen zu ökonomischen, juristischen und vergleichbaren Inhalten noch fehlen.

5 Literatur

- [1] TR29: Engineering hybrider Leistungsbündel – Dynamische Wechselwirkungen von Sach- und Dienstleistungen in der Produktion. <http://www.tr29.de>, Juni 2007.
- [2] Welp E.G., Sadek T., 2008, Conceptual Design of Industrial Product-Service Systems (HLB) based on the Extended Heterogeneous Modelling Approach, Proceedings of International Design Conference (DESIGN 2008), May 19 - 22 Dubrovnik, Croatia.
- [3] Bludau C., Welp E.G: Semantic Web Services for the Knowledge-Based Design of Mechatronic Systems. Proceedings of the 15th International Conference on Engineering Design (ICED); Globally Effective Engineering Design; Melbourne/Australia; 2005
- [4] Kitamura, Y.; Sano, T.; Kouji, N.; Mizoguchi R.: A Functional Concept Ontology and Its Application to Automatic to Identification of Functional Structures. Advanced Engineering Informatics, Vol. 16, No. 2, 2002
- [5] Cao, D.; Fu, M. W.; Gu, Y.; Haipeng, J.: Port-based Ontology Modeling For Product Conceptual Design. Proceedings of the ASME 2008 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, IDETC/CIE, 2008
- [6] Welp, E.G.; Meier, H.; Sadek, T.; Sadek, K.: Modeling Approach for the integrated Development of Product-Service Concepts in early Development Phases, 41st CIRP Conference on Manufacturing Systems 2008, 2008
- [7] Jansen, S.: Eine Methodik zur modellbasierten Partitionierung mechatronischer Systeme, Dissertation am Lehrstuhl für Maschinenelemente, Fakultät Maschinenbau, Ruhr-Universität Bochum, Bochum, 2006

Prof. Dr.-Ing. Ewald G. Welp, Dipl.-Ing. Ralf Theiß,
Dipl.-Ing. Tim Sadek, cand.-Ing. Matthias Köster
Lehrstuhl für Maschinenelemente und Konstruktionslehre (LMK)
Ruhr-Universität Bochum
Universitätsstraße 150, D-44801 Bochum
Tel. +49 (0)234 / 32-22636
Fax +49 (0)234 / 32-14159
E-Mail: welp@lmk.rub.de
URL: www.lmk.rub.de