

Modellierung von Funktionen in der modellbasierten Entwicklung von Systemverbänden kollaborierender cyber-physischer Systeme

Zuordnung zum Themengebiet: Modellierung und Entwurf

Alexander Hayward¹, Dr. Marian Daun², Dr. Wolfgang Böhm³, Ana Petrovska³, Lisa Krajinski², Prof. Dr.-Ing. Alexander Fay¹

Keywords: Funktionsmodellierung, Modellbasierte Entwicklung, Systemverbund, Requirements Engineering

Cyber-physische Systeme (CPS) werden zunehmend in offenen und dynamischen Umgebungen eingesetzt, in welchen sie ihre Funktionen nicht mehr nur allein, sondern im Zusammenwirken mit anderen CPS ausführen sollen. Durch diese Kollaboration soll ein Mehrwert geschaffen werden, den das einzelne CPS allein nicht erreichen könnte. Die Entwicklung und Realisierung dieser CPS und Systemverbände bringt zahlreiche neue Herausforderungen mit sich. Welche Aspekte in diesem Zusammenhang bei der Modellierung von Funktionen im Rahmen eines modellbasierten Entwicklungsprozesses berücksichtigt werden müssen, soll in diesem Beitrag vorgestellt werden. Die Aspekte sind in einem Metamodell zusammengefasst, das im Rahmen des vom BMBF geförderten Projektes CrESt (Förderkennzeichen 01IS16043) entwickelt worden ist.

1 Einführung und Motivation

Cyber-physische Systeme (CPS) werden heutzutage in Umgebungen unter sich häufig ändernden Anforderungen eingesetzt. Damit ein CPS möglichst viele dieser Anforderungen erfüllen kann, wird ein immer größerer Funktionsumfang angestrebt. Insbesondere durch die Vernetzung von verschiedenen CPS untereinander werden mehr Anwendungsmöglichkeiten erreicht. In der Entwicklung dieser vernetzten multifunktionalen Systeme steigen hierdurch die zu berücksichtigenden Abhängigkeiten, was den Bedarf nach durchgängigen Methoden zur Beherrschung der hieraus resultierenden Komplexität auslöst [Vo15].

In verschiedenen Forschungsprojekten wurde bereits untersucht, wie Entwicklungsprozesse von CPS durch modellbasierte Methoden unterstützt werden können. Unter anderem wurden dabei Fragestellungen behandelt, wie sich Anforderungen und Abhängigkeiten auf

¹ Helmut-Schmidt-Universität Hamburg, Institut für Automatisierungstechnik, Holstenhofweg 85, 22043 Hamburg, alexander.hayward@hsu-hh.de & alexander.fay@hsu-hh.de

² Universität Duisburg-Essen, paluno - The Ruhr Institute for Software Technology, Gerlingstraße 16, 45127 Essen, Marian.Daun@paluno.uni-due.de & lisa.krajinski@paluno.uni-due.de

³ Technische Universität München, Fakultät für Informatik, Boltzmannstraße 3, 85748 Garching bei München, boehm@in.tum.de & ana.petrovska@tum.de

verschiedenen Granularitätsebenen durchgängig und semantisch kohärent abbilden und aus unterschiedlichen Perspektiven betrachten lassen (s. [Me14], [Po12], [Po16]).

Die funktionszentrierte Entwicklung stellt in diesem Zusammenhang domänenübergreifend einen langjährig bewährten Weg dar, wie sich aus anfänglich oft informell beschriebenen Anforderungen sukzessive technische Lösungsmöglichkeiten ableiten lassen. Hierbei wird zunächst auf abstrakter Ebene der eigentliche Zweck des Systems durch Spezifikation der In- und Outputs von Funktionen und des an der Systemgrenze beobachtbaren Verhaltens beschrieben. Anschließend werden auf dieser Grundlage Dekompositionen in weitere Teilfunktionen durchgeführt und durch Verbindungen der In- und Outputs mögliche Lösungsprinzipien hergeleitet [Lu18]. Diese Verbindungen der Funktionen werden in bisherigen Modellierungsmethoden als statisch und zur Entwicklungszeit vollständig bekannt angenommen [Po16].

2 Problemstellung

Die Vernetzung der CPS untereinander unterliegt jedoch dynamischen Rahmenbedingungen und findet meist erst während der Betriebszeit statt, sodass relevante Details dieser Vernetzung zur Entwicklungszeit nur unvollständig vorgedacht werden können. Der Grundgedanke der dynamischen Vernetzung beinhaltet, dass unterschiedliche CPS sich ihre Funktionen gegenseitig bereitstellen, um gemeinsam ein Ziel zu erreichen, das ein einzelnes CPS alleine nicht erreichen kann. Durch die Aggregation von CPS-Funktionen entstehen neue funktionale Anwendungsmöglichkeiten.

Die beteiligten CPS bilden zur Zielerreichung einen kollaborativen Systemverbund, in dem sie zusammenwirken und hierbei die jeweiligen einzunehmenden Rollen und damit verbundenen Beiträge (in Form von Funktionen) festlegen. Da die konkreten Teilnehmer des Verbundes zur Entwicklungszeit nicht bekannt sind, kann die Kollaboration zur Entwicklungszeit auch nur bedingt geplant werden; sie erfordert eine Abstimmung über die die Funktionen und deren Verhalten zwischen den CPS zur Betriebszeit. Die dynamische Aggregation von Funktionen verschiedener CPS innerhalb eines Verbundes stellt die Entwicklung solcher CPS vor die Herausforderung, dass statische Modellierungsmethoden von Funktionen nicht mehr ausreichen. Die konkrete Ausprägung eines Verbundes ergibt sich in Abhängigkeit von sich ändernden Zielen erst zur Betriebszeit des CPS und kann daher nur bedingt in der Entwicklung vorgedacht werden.

3 Metamodell für die Modellierung von Funktionen von CPS und Systemverbänden

Welche Zusammenhänge im Rahmen der modellbasierten Entwicklung zwischen Funktionsmodellen und anderen Modellen aus der Anforderungsmodellierung berücksichtigt

werden müssen, ist im vom BMBF geförderten Projekt CrEst⁴, dem Nachfolgeprojekt von SPES2020 und SPES_XT, das sich mit der Entwicklung von Methoden zur modellbasierten Entwicklung solcher Systeme beschäftigt [Po16], untersucht worden.

3.1 Methodische Vorgehensweise

Um ein umfassendes Verständnis von den zu berücksichtigenden Aspekten der Funktionsmodellierung für die Entwicklung von kollaborierenden CPS zu erhalten, wurden innerhalb des Projekts CrEst all jene Methoden im Detail untersucht, die einen Bezug zur Modellierung von Funktionen aufweisen. Hierbei wurden insgesamt 49 unterschiedliche Methoden identifiziert und untersucht, inwiefern deren Ansätze über existierende Methoden der Vorgängerprojekte SPES2020 und SPES_XT zur Modellierung von statischen Funktionsverbänden hinausgehen. Diese Methoden adressieren hierbei Herausforderungen, die sich sowohl an der Beschreibung der Systeme als auch an dessen offenen und dynamischen Kontext ausrichten. Um diese unterschiedlichen Aspekte zusammenhängend darstellen zu können, wurden sie in einem Metamodell zusammengefasst, welches anschließend anhand von Beispielen evaluiert wurde. Dieses Metamodell und seine zugrundeliegenden Anforderungen sollen in diesem Beitrag vorgestellt und im Detail erläutert werden.

Aufgrund des großen Umfangs des Metamodells werden die einzelnen Elemente in mehreren Teilabschnitten vorgestellt. Außerdem werden in jedem Teilabschnitt Bezüge zu einem praktischen Anwendungsbeispiel aus dem Themenfeld der Smart Factory hergestellt (im Text durch kursive Schrift hervorgehoben). Vor der eigentlichen Darstellung des Metamodells soll dieses Beispiel im Folgenden kurz vorgestellt werden.

3.2 Anwendungsbeispiel: Auftragsgesteuerte Produktion in der Smart Factory

In dem Anwendungsbeispiel werden autonom agierende Fertigungsmodule (= CPS) innerhalb einer Smart Factory betrachtet. Diese Module verfügen über unterschiedliche Fertigungsfunktionen (z.B. bohren, drehen, transportieren oder montieren), mit denen sie zur Produktion von Gütern beitragen können. Für die Herstellung eines kundenindividuellen Produktes bilden die Module, die zu dieser Produktion beitragen, einen auftragsspezifischen Systemverbund, der von den für die Herstellung des Produktes benötigten Funktionen geprägt ist. Für die Bildung eines solchen Verbundes ist es erforderlich, dass zunächst überprüft wird, welche verfügbaren Module überhaupt einen Beitrag zur Produktion leisten können und sollen, sowie festzulegen, in welcher Reihenfolge und zu welchem Zeitpunkt sie ihre Funktionen ausführen sollen [Pl17].

⁴ Das CrEst Projekt zielt auf die Entwicklung einer umfassenden Methodensammlung für die Entwicklung von kollaborierenden CPS. Es wird öffentlich im Zeitraum 2017 bis 2020 durch das BMBF gefördert.

3.3 Metamodell zur Funktionsmodellierung von kollaborativen CPS

In der folgenden Abbildung ist das Metamodell in seiner Gesamtheit dargestellt. Aufgrund der Vielzahl der Elemente sind diese grafisch separiert worden. Die einzelnen farblich gemeinsam hervorgehobenen Elemente werden in den Teilabschnitten erläutert.

3.4 Systeme, Systemgruppen und ihre funktionalen Architekturen

Bei der Modellierung von Funktionen ist zunächst die Zugehörigkeit zu Systemen von Bedeutung. Aufgrund der Dynamik von kollaborativen Systemen ist die Betrachtung immer an einen bestimmten Zeitpunkt t gebunden, zu dem unterschieden werden kann, ob es sich bei einem betrachteten System um ein kollaborierendes CPS, ein nicht kollaborierendes CPS oder einen Systemverbund handelt. Bei einem nicht kollaborierenden CPS ist zu unterscheiden, ob dieses auf Grund seiner Beschaffenheit nicht zu einer Kollaboration fähig ist (nicht kollaboratives CPS) oder ob es ein kollaborationsfähiges aber aktuell nicht kollaborierendes CPS ist (kollaboratives CPS). Während das nicht kollaborative CPS zu keinem Zeitpunkt mit anderen CPS zusammenarbeiten kann, kann das kollaborative CPS zu einem späteren Zeitpunkt $t+1$ zu einem kollaborierenden CPS innerhalb eines Systemverbundes werden. Der Systemverbund setzt sich hierbei aus mehreren kollaborierenden CPS zusammen, die sich gegenseitig ihre Funktionen zur Verfügung stellen und Informationen austauschen, um ein gemeinsames Ziel zu erreichen. Jedes einzelne CPS (sowohl kollaborierend als auch nicht kollaborierend) besitzt eine funktionale Architektur, durch die beschrieben wird, wie ihre unterschiedlichen Funktionen miteinander interagieren können. Auch der Systemverbund besitzt eine funktionale Architektur, die sich aus den Funktionen der kollaborierenden CPS zusammensetzt.

Ein einzelnes Fertigungsmodul innerhalb der Smart Factory existiert zunächst losgelöst von anderen Modulen und bietet für eine Produktion von Gütern seine Funktionen an (kollaborationsfähiges CPS). Sofern die Funktionen dieses Moduls und diejenigen weiterer Module für einen Produktionsauftrag benötigt werden, bilden sie für die Erfüllung des Auftrages einen Systemverbund. Aus dem grundsätzlich kollaborationsfähigen wird ein tatsächlich kollaborierendes Modul. Der auftragsspezifische Systemverbund innerhalb der Smart Factory entsteht erst im Rahmen dieser Kollaboration.

3.5 Die Funktion: Bestandteile und Spezialisierungen

Im Kern des Metamodells steht die Funktion. Das Verhalten des Systems kann auf einer abstrakten Ebene durch seine Funktionen beschrieben werden. Die Funktion repräsentiert hierbei den eigentlichen Zweck, den ein System erfüllen soll. Eine Funktion verfügt über Schnittstellen, über welche sie Informationen, Material oder Energie aufnehmen, transformieren und wieder ausgeben kann. Je nach Domäne kann das Verständnis vom Begriff der Funktion im Detail leicht variieren, hat auf dieser allgemeinen Ebene jedoch Gültigkeit [Ei12].

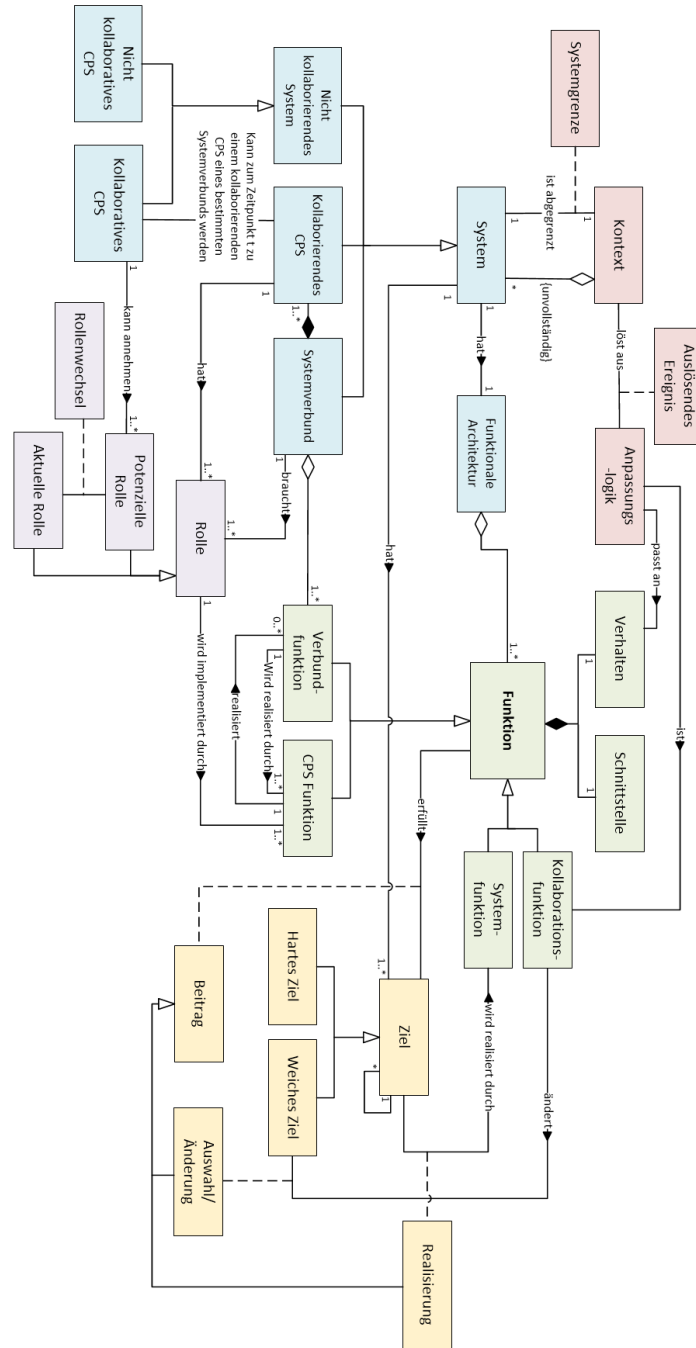


Abbildung 1: Metamodell für Funktionen von CPS und Systemverbänden

Sowohl in der klassischen Konstruktionsmethodik nach [Pa07] als auch in der heutigen modellbasierten Systementwicklung komplexer CPS [Po12, Me14] werden Funktionen in der frühen Phase der Entwicklung aus Anforderungslisten und -modellen abgeleitet, um die zu erbringende Leistung des CPS zu erfassen. Da auf Basis dieser abstrakten Funktionen nur sehr bedingt konkrete Lösungsprinzipien abgeleitet werden können, erfolgt eine weitere Dekomposition der Funktionen in Teilfunktionen, welche ebenfalls weiter unterteilt werden können. Auch diese Teilfunktionen verfügen über Schnittstellen, über die sie miteinander verbunden sind. Das hieraus entstehende Netz wird als funktionale Architektur bezeichnet und beschreibt den prinzipiellen Leistungsumfang des CPS. Durch das Verhalten kann außerdem beschrieben werden, welche Zuständen, Zustandsübergängen und Abhängigkeiten Funktionen besitzen [Ei12].

Innerhalb der Smart Factory stellen sich die Fertigungsmodule ihre teilweise sehr heterogenen Fertigungsfunktionen zur Verfügung, um einen Beitrag zur Herstellung von Produkten zu liefern. Eine Funktion, die einen Bohrvorgang eines Moduls beinhaltet, erstellt aus einem Input in Form von erforderlichem Material, benötigter Energie und Informationen über den geforderten Vorgang, ein mit den entsprechenden Bohrungen versehenes Werkstück. Das Verhalten dieser Funktion umfasst die einzelnen Schritte dieser Bohrung.

In Systemverbänden erbringen die kollaborierenden CPS ihre jeweiligen Funktionen im gegenseitigen Zusammenwirken (CPS Funktion im Metamodell) für die Erreichung eines gemeinsamen Ziels. Betrachtet man den Systemverbund als einzelnes System mit verschiedenen Komponenten und eigenen Funktionen (Verbundfunktion im Metamodell), so werden diese eigenen Funktionen des Verbundes durch die Aggregation und daraus resultierende Emergenz der an ihm beteiligten CPS realisiert.

Die Verbundfunktion eines innerhalb der Fabrik gebildeten Verbundes aus beteiligten Modulen würde zum Beispiel „Produkt XY herstellen“ beinhalten. Diese Verbundfunktion setzt sich dann aus den einzelnen Teilfunktionen der Module zusammen.

Während die Funktionen eines einzelnen CPS durch dessen Komponentenarchitektur realisiert werden, bleiben die Funktionen des Verbundes nach der Entwicklung zunächst ein vorgedachtes Konstrukt, das sich erst während der Kollaboration durch die CPS konstituiert. Durch Verbundfunktionen kann dargestellt werden, welche konkreten Beiträge beteiligte CPS liefern müssen, um das gemeinsame Ziel des Verbundes erreichen zu können.

Neben diesen beiden Spezialisierungen der Funktion wird im Metamodell weiter zwischen Systemfunktion und Kollaborationsfunktion unterschieden. Eine Systemfunktion beinhaltet hierbei den individuellen Beitrag, den ein CPS im Rahmen der Kollaboration leistet. Für das an der Produktion beteiligte Bohrmodul wäre dies erneut die Funktion des Bohrens. Im erweiterten Sinne sind hierunter auch weitere interne Funktionen des CPS zu verstehen, die, wenn auch indirekt, zur Zielerfüllung beitragen. Kollaborationsfunktionen umfassen hingegen ein allgemeines Set an Funktionen, das bei allen am Verbund beteiligten CPS gleichermaßen vorausgesetzt wird, um eine Kollaboration generell und unabhängig von konkreter Gestalt und Ziel möglich zu machen. Hierunter fallen unter anderem

Funktionen zur Wahrnehmung anderer Systeme und Kommunikation mit diesen, die Verhandlung von Zielen, der Abgleich zwischen benötigten und vorhandenen Funktionen, oder die Anpassung des eigenen Verhaltens an die Gegebenheiten des Systemverbundes.

Während durch die Unterteilung von CPS Funktion und Verbundfunktion hinsichtlich der Zugehörigkeit einer Funktion unterschieden worden ist, wird durch die Unterscheidung zwischen Systemfunktion und Kollaborationsfunktion die Art der Funktion weiter spezifiziert.

Sofern es innerhalb der Smart Factory ein Modul gibt, das die zentrale Koordination und Abstimmung ermöglicht, so können die hierarchisch untergeordneten Module diesem Koordinator durch ihre Kollaborationsfunktionen ihre Systemfunktionen, Verfügbarkeiten und individuelle Ziele (bspw. einen möglichst hohen Durchsatz zu erreichen) mitteilen. Sofern die Bearbeitung von Produktionsaufträgen dezentral durch die einzelnen Module selbst organisiert und abgestimmt wird, ist eine Abstimmung durch die Kollaborationsfunktionen zwischen den hierarchisch gleichgestellten Modulen erforderlich.

3.6 Rollen zur Definition von Systemgruppen

Ein weiteres Konzept, das die Modellierung und Realisierung von Systemverbänden unterstützen soll, ist die Verwendung von Rollen. Innerhalb eines Systemverbunds gibt es unterschiedliche Aufgaben, die zur Zielerreichung erfüllt werden müssen. Um festzulegen, welches CPS welche Aufgabe übernehmen soll, werden Rollen definiert, die von den CPS eingenommen werden können. Ein einzelnes CPS kann hierbei eine oder auch mehrere Rollen innerhalb eines Verbundes einnehmen. Durch die mit einer Rolle verbundenen Aufgaben werden Anforderungen und zur Erfüllung notwendigen Funktionen der jeweiligen CPS festgelegt. Neben den Anforderungen an die funktionale Beschaffenheit der CPS wird durch die Rollen außerdem das notwendige Verhalten der CPS während der Kollaboration definiert. Dieses notwendige Verhalten beinhaltet die damit verbundenen Abhängigkeiten zwischen den Funktionen auf Ebene des Verbundes und damit zwischen den einzelnen CPS [We19, Re19, Lu18].

Sofern für ein CPS bereits während der Entwicklung definiert wird, welche potenziellen Rollen es aufgrund seiner Funktionen innerhalb eines bestimmten Verbundes einnehmen kann, so ermöglicht dies während der Betriebszeit eine Zuordnung von an der Kollaboration beteiligten CPS zu den Rollen. Einem CPS, das eine bestimmte Rolle innerhalb des Verbundes eingenommen hat (Aktuelle Rolle im Metamodell), wird damit die Verantwortung für die rollenspezifischen Aufgaben übertragen. Sofern ein kollaborierendes CPS einen Verbund wieder verlässt, die mit der Rolle verbundenen Aufgaben aber weiterhin erfüllt werden müssen, ist es erforderlich, dass ein anderes kollaborierendes CPS diese Rolle übernimmt. Sofern diese Rolle nicht zusätzlich zur alten Rolle angenommen wird, ist ein Wechsel zwischen den Rollen notwendig.

Diese Vorgänge sind nur dann möglich, sofern die am Verbund beteiligten CPS ein gemeinsames Verständnis von einzunehmenden Rollen besitzen. Je nach Anwendungsdo-

mäne sind hierbei Systemverbände denkbar, die ein hohes Maß an Standardisierung besitzen, an denen sich Entwickler der CPS bereits im Entwurf orientieren. In anderen Domänen hingegen wird aufgrund der Heterogenität und Vielzahl der unterschiedlichen Verbände eine Standardisierung nicht möglich sein, sodass während des Betriebes zwischen den CPS im Rahmen der Verhandlung einer möglichen Kollaboration ein gemeinsames Verständnis von der Gestalt des Verbundes und den einzunehmenden Rollen sowie notwendigen Funktionen entwickelt werden muss.

Typische Rollen in der Smart Factory sind beispielsweise der bereits oben genannte Koordinator, oder auch Fertigungs-, Montage- oder Transportmodule. Sofern ein bestimmtes Modul die funktionale Beschaffenheit dafür besitzt Fertigungsfunktionen und Montageoperationen auszuführen, kann es auch beide dafür vorgesehenen Rollen des Verbundes einnehmen.

3.7 Ziele von Systemen und Systemgruppen

Wie in den vorherigen Abschnitten erwähnt, sind mit Systemen Ziele verbunden. Dabei wird ein Ziel als eine Bedingung oder Situation definiert, die das System mit Hilfe seiner Funktionen erfüllen oder herbeiführen möchte. Dies gilt sowohl für einzelne CPS als auch für den Systemverbund. Die individuellen Ziele der betroffenen CPS können voneinander und von den Zielen des Verbundes abweichen und sogar widersprüchlich sein. Bei (teilweise) unterschiedlichen oder widersprüchlichen Zielen müssen die Ziele zwischen den beteiligten CPS und dem Systemverbund ausgehandelt werden. Dies beinhaltet auch die Art und Weise, wie ein CPS zur Erreichung der Verbundziele beiträgt. Außerdem impliziert dies, dass die CPS ihre Ziele entsprechend den Ergebnissen der Verhandlungen ändern können. In diesem Sinne werden Ziele zur Laufzeit als dynamisch betrachtet. Verhandeltbare Ziele (Weiche Ziele) eines beteiligten CPS können dabei angepasst werden, wohingegen ein Zielkonflikt bei nicht-verhandelbaren Zielen (Harte Ziele) dazu führen kann, dass beispielsweise ein Beitritt des CPS zum Verbund nicht möglich ist⁵.

Im Anwendungsbeispiel der Smart Factory könnten harte Ziele eines Fertigungsmoduls etwa einzuhaltende Wartungsintervalle sein, wodurch das Modul für eine gewisse Zeit nicht mehr für den Produktionsverbund zur Verfügung stehen kann. Ein Beispiel für ein weiches Ziel könnte sein, dass ein Modul das Ziel hat, eine bestimmte Rolle im Verbund einzunehmen, jedoch auf Grund eines Ausfalls eines anderen Moduls gezwungen ist, dessen Rolle zu übernehmen.

Wie in Abschnitt 3.5 dargestellt, werden Ziele durch das Zusammenwirken der Systemfunktionen erreicht. Dieses Zusammenarbeiten muss sich möglicherweise ändern, wenn sich die Ziele ändern. Die (bei weichen Zielen dynamische) Abbildung der Ziele auf die Beiträge der Systemfunktionen wird durch die Kollaborationsfunktionen geleistet. Die

⁵ Hinweis: Die Begriffe "Hartes Ziel" und "Weiches Ziel" werden hier anders verwendet, als in der Literatur zur Zielmodellierung. Harte Ziele sind nicht Gegenstand von Verhandlungen und in gewissem Sinne „statisch“, während weiche Ziele dynamisch ausgehandelt und geändert werden können.

Kollaborationsfunktionen ermöglichen im Systemverbund, Zielen Systemfunktionen zuzuordnen, die diese Ziele realisieren. Dies wiederum impliziert eine Interaktion zwischen Kollaborationsfunktionen und den Systemfunktionen. Eine solche Interaktion zwischen Funktionen wurde im SPES-Framework als "Mode" bezeichnet.

3.8 Kontext und Anpassbarkeit

Zuletzt sollen nun der Kontext des Systems und die im Kontext auftretenden Ereignisse, die eine Anpassung des Systems auslösen, betrachtet werden. Im klassischen Verständnis des System Engineerings sind System und Kontext durch die Systemgrenze voneinander getrennt. Die Schnittstelle des Systems markiert hierbei diese Grenze und ermöglicht eine Interaktion mit dem Kontext. Da im Metamodell ausschließlich Funktionen betrachtet werden, die allgemeine Abgrenzung zwischen System und Kontext jedoch auch zahlreiche weitere Elemente, wie zum Beispiel Personen, Wissen oder strukturelle Komponenten von Systemen berücksichtigt, ist durch den Vermerk der Unvollständigkeit auf diesen Umstand hingewiesen. Da Systeme in einem dynamischen Kontext agieren, der sich über die Zeit ändert, treten Ereignisse im Kontext auf, die dazu führen, dass ein System sein Verhalten anpassen muss. Aus Sicht eines einzelnen kollaborierenden CPS innerhalb eines Systemverbundes kann ein solches Ereignis darin bestehen, dass ein kollaboratives CPS im Kontext zu einem späteren Zeitpunkt Teil des Systemverbundes wird.

Der Beitrag des neu hinzugekommenen CPS innerhalb des Verbundes kann eine Anpassung des eigenen Verhaltens der restlichen kollaborierenden CPS im Verbund erforderlich machen. Diese Anpassungsfähigkeit gehört zu den Kollaborationsfunktionen und ist notwendiger Bestandteil eines jeden CPS, das die Möglichkeit der Kollaboration innehaben soll. Betrachtet man den Verbund hingegen als ein System, bewirkt das Hinzukommen eines neuen CPS zum Verbund nicht nur eine Änderung des Kontextes, sondern auch eine Änderung der Systemgrenze und dessen innerer Struktur. In den bisherigen Ansätzen wurde die Systemgrenze als statisch angenommen. Die Möglichkeit der Veränderung dieser Grenze über die Zeit stellt zu berücksichtigende Neuerung in der Modellierung von System und Kontext dar.

Im Falle der Fabrik könnte diese Änderung im Kontext durch das Auftreten eines neuen Produktionsauftrages ausgelöst werden (Ereignis), der dazu führt, dass sich ein neuer Verbund bildet oder ein bestehender Verbund durch den Austausch zwischen notwendigen und nicht mehr benötigten Modulen verändert.

4 Ausblick

Auf Basis des vorgestellten Metamodells bestehen die weiteren Schritte in der Forschung in der Analyse und Erweiterung existierender Beschreibungsmittel, um die in diesem Beitrag dargestellten Inhalte in der Modellierung zu berücksichtigen. Da die Planung und Realisierung von Systemverbänden von Entwicklern nicht nur die Wahl einer geeigneten

Modellierungssprache verlangt, sondern auch, diese Sprache semantisch korrekt anzuwenden, sind entsprechende methodische Vorgehensweisen zu entwickeln. Da das Metamodell einen domänenübergreifenden Anspruch hat, soll eine Evaluation durch den Einsatz in möglichst unterschiedlichen Anwendungsszenarien erfolgen.

Danksagung: Dieser Beitrag wurde vom Bundesministerium für Bildung und Forschung unter den Förderkennzeichen 01IS16043A, 01IS16043U und 01IS16043V gefördert.

5 Literaturverzeichnis

- [Ei12] Eisenbart, B.; Blessing, L.; and Gericke, K.: Functional Modelling Perspectives Across Disciplines: A Literature Review. International Design Conference - Design 2012, Dubrovnik, Croatia, 2012.
- [Lu18] Ludewig, A. et. al.: Requirements for Modeling Dynamic Function Networks for Collaborative Embedded Systems. In: Schäfer, Karagiannis (Hrsg.): Fachtagung Modellierung 2018, Braunschweig, 2018.
- [Me14] Meissner, H. et.al.: Model-Based Development Process of Cybertronic Products and Production Systems. In: Advanced Materials Research, Vol. 1018, S. 539–546, 2014.
- [Pa07] Pahl, G. et. al.: Konstruktionslehre. Grundlagen erfolgreicher Produktentwicklung; Methoden und Anwendung. 7. Aufl., Heidelberg: Springer. 2007.
- [Pl17] Plattform Industrie 4.0: Anwendungsszenario trifft Praxis: Auftragsgesteuerte Produktion eines individuellen Fahrradlenkers. BMWi, Berlin, 2017.
- [Po12] Pohl, K. et. al.: Model-Based Engineering of Embedded Systems. The SPES 2020 Methodology. Heidelberg: Springer, 2012.
- [Po16] Pohl, K. et. al.: Advanced model-based engineering of embedded systems. Extensions of the SPES 2020 methodology. Cham, Switzerland: Springer, 2016.
- [Re19] Regnat, N. et. al.: Seamless Model Based Approach - MQ1.AP2.D4. Hrsg. v. Teilprojekt MQ1 des CrESt Projektes. Projektinterner Bericht des Projektes CrESt, 2019.
- [Vo15] Vogelsang, A.: Model-based Requirements Engineering for Multifunctional Systems. Dissertation. Technische Universität München, Institut für Informatik, 2015.
- [We19] Weiß, S. et. al.: Modeling of Dynamics in the Open Context of Collaborative Embedded Systems – EC4.AP2.D3. Hrsg. von Teilprojekt EC4 des CrESt Projektes. Projektinterner Bericht des Projektes CrESt, 2019.