

# Evaluation eines modellbasierten Requirements-Engineering-Ansatzes für den Einsatz in der Motorsteuerungs-Domäne<sup>1</sup>

Ernst Sikora, Klaus Pohl

Universität Duisburg Essen  
Schützenbahn 70, 45127 Essen  
{ernst.sikora; klaus.pohl}@sse.uni-due.de

**Abstract:** Modellbasierte Entwicklungsansätze für softwareintensive, eingebettete Systeme sollen dazu beitragen, Fehler im Entwicklungsprozess zu vermeiden, die Entwicklungsdauer zu verkürzen und die hohe Systemkomplexität besser beherrschbar zu machen. Besonders vielversprechend erscheint der Einsatz von Modellen bereits in frühen Entwicklungsaktivitäten, in denen die Anforderungen und die Architektur definiert werden. COSMOD-RE ist ein ziel- und szenariobasierter Requirements-Engineering-Ansatz, der eine verzahnte Entwicklung von Anforderungs- und Architekturmodellen über mehrere Abstraktionsstufen eines softwareintensiven, eingebetteten Systems hinweg unterstützt. In diesem Beitrag werden die Erkenntnisse und Erfahrungen beschrieben, die bei der Evaluation von COSMOD-RE in einem spezifischen Entwicklungskontext, der Entwicklung von Anforderungen für die Anwendungssoftware einer Motorsteuerung, gesammelt wurden. Hauptmotivation für diese Evaluation war es, Erkenntnisse über den Nutzen und die Anwendbarkeit von COSMOD-RE in dieser Domäne zu gewinnen.

## 1 Motivation und Problemstellung

Bei der Entwicklung komplexer, softwareintensiver, eingebetteter Systeme müssen Anforderungen auf verschiedenen Abstraktionsstufen definiert und analysiert werden, beispielsweise Anforderungen an das Gesamtsystem sowie Anforderungen an die Systemsoftware. Modellbasierte Requirements-Engineering-Ansätze sollen dazu beitragen, Fehler in Anforderungen an das System bzw. die Software zu vermeiden, sowie in kürzerer Zeit zu einem umfassenden Verständnis der Anforderungen zu gelangen. Der COSMOD-RE-Ansatz (siehe [PS09] und [Po10]) unterstützt die modellbasierte Entwicklung von Anforderungen an softwareintensive, eingebettete Systeme. COSMOD-RE verknüpft dazu verschiedene, im RE etablierte Modellierungstechniken wie Ziel-, Szenario- und Funktionsmodellierung.

---

<sup>1</sup> Dieser Beitrag wurde gefördert durch die BMBF Innovationsallianz SPES 2020, Förderkennzeichen 01IS08045.

Um Erfahrungen für eine Weiterentwicklung des Ansatzes zu sammeln, wird COSMOD-RE in verschiedenen Teilgebieten der Domäne softwareintensiver, eingebetteter Systeme evaluiert. Dieser Beitrag beschreibt die Evaluation von COSMOD-RE im Kontext der Entwicklung von Anwendungssoftware für Motorsteuerungen im Automobilbereich (siehe [BDJ03]). Die Evaluation von COSMOD-RE in der Motorsteuerungs-Domäne stellt einen kritischen Anwendungsfall für den Ansatz dar, da Ziele und Szenarien im RE typischer Weise zur Beschreibung der Systemnutzung durch Personen und andere Systeme eingesetzt. Die Motorsteuerung hat jedoch (zum Beispiel im Vergleich zu einem Fahrerassistenzsystem) wenig direkte Nutzerinteraktion und wird weitgehend durch regelungstechnische Abläufe bestimmt. Der Beitrag ist wie folgt aufgebaut: Abschnitt 2 gibt einen Überblick über den COSMOD-RE-Ansatz. Abschnitt 3 berichtet über die durchgeführte Evaluation des Ansatzes in der Motorsteuerungs-Domäne. Abschnitt 4 zieht Schlussfolgerungen aus den Evaluationsergebnissen.

## 2 Überblick über den COSMOD-RE-Ansatz

COSMOD-RE [PS09] [Po10] ist ein ziel- und szenariobasierter RE-Ansatz für softwareintensive, eingebettete Systeme, der im Rahmen mehrerer Forschungs-Industrie-Kooperationen entwickelt und erprobt wurde. COSMOD-RE weist aufgrund der folgenden Eigenschaften eine prinzipielle Eignung für die Motorsteuerungs-Domäne auf:

- ❑ In der Motorsteuerungs-Domäne müssen Anforderungen auf verschiedenen Abstraktionsebenen betrachtet werden, wie zum Beispiel auf der Fahrzeugebene, der Systemebene und der Software/Hardware-Ebene. COSMOD-RE unterstützt die Spezifikation von Anforderungen auf verschiedenen Abstraktionsebenen mittels einer Abstraktionsstufenhierarchie sowie mittels Prozessstrukturen für die Entwicklung von Anforderungen über mehrere Abstraktionsebenen hinweg.
- ❑ In der Motorsteuerungs-Domäne ist eine funktionsorientierte Spezifikation von System- und Softwareanforderungen weit verbreitet. Von einigen Autoren wird darüber hinaus der Einsatz von Zielen und Szenarien angeregt [PK04]. COSMOD-RE unterstützt die modellbasierte Spezifikation von Anforderungen mittels Ziel-, Szenario- und Funktionsmodellen. Ziel- und Szenariomodelle unterstützen dabei die systematische Identifikation von System- und Softwarefunktionen.

In den folgenden Unterabschnitten wird ein kurzer Überblick über die wesentlichen Bestandteile des COSMOD-RE-Ansatzes gegeben. Eine umfassende Beschreibung von COSMOD-RE ist in [Po10] zu finden.

### 2.1 COSMOD-RE Abstraktionsstufenhierarchie

Eine wesentliche Eigenschaft komplexer Systeme ist deren Strukturierung in mehrere Abstraktionsebenen. COSMOD-RE unterstützt die Entwicklung von Anforderungen auf verschiedenen Abstraktionsebenen unter anderem durch eine vorgegebene, generische Abstraktionsstufenhierarchie. Diese Abstraktionsstufenhierarchie soll Requirements-Ingenieure dabei unterstützen, den richtigen Abstraktions- und Detaillierungsgrad für die Anforderungsartefakte auf einer bestimmten Abstraktionsebene zu wählen. Die

Abstraktionsstufenhierarchie führt zudem (auf höheren Abstraktionsstufen) zur Komplexitätsreduktion der erstellten Modelle, da sie die Abstraktion von Lösungsdetails fördert. Die Abstraktionsstufenhierarchie von COSMOD-RE ist in Abb. 1 dargestellt.

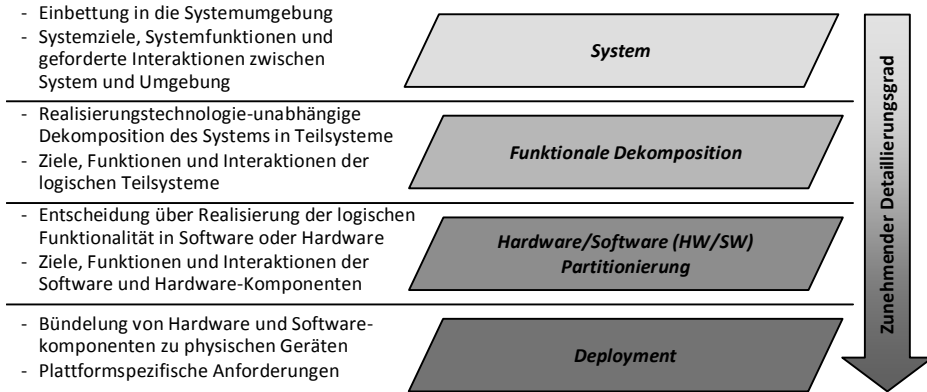


Abb. 1 Abstraktionsstufenhierarchie der COSMOD-RE-Methode

## 2.2 COSMOD-RE Artefakttypen

COSMOD-RE beinhaltet lösungsneutrale Anforderungsmodelle (Zielmodell und Szenariomodell), lösungsorientierte Anforderungsmodelle (Datenmodell, Funktionsmodell und Verhaltensmodell) sowie strukturelle Architekturmodelle. Tab. 1 zeigt einen Überblick über die wesentlichen von COSMOD-RE unterstützten Artefakttypen, deren Eigenschaften sowie mögliche Modellierungssprachen.

Artefakttyp	Kurzbeschreibung
<b>Zielmodell</b>	Definiert die Ziele, die das System erfüllen soll, einschließlich der hierarchischen Dekomposition der Ziele. Die Ziele werden unabhängig von einer möglichen oder intendierten Systemlösung definiert.
<b>Szenario-modell</b>	Definiert eine Menge von Szenarien und setzt diese zur Systemumgebung in Beziehung. Ein Szenario beschreibt eine Folge von Interaktionen, die zur Erfüllung eines bestimmten Ziels führen soll.
<b>Funktions-modell</b>	Definiert die Funktionen (Dienste), die das geplante System seiner Umgebung bereitstellen soll, die Eingabe- und Ausgabedatenflüsse der einzelnen Funktionen sowie wesentliche Datenspeicher des Systems.
<b>Datenmodell</b>	Definiert welche Informationen über die Umgebung im System abgebildet werden sollen und wie diese strukturiert sind.
<b>Verhaltens-modell</b>	Definiert die Betriebsmodi des Systems, Ereignisse die zum Wechsel des Betriebsmodus führen und, welche Funktionen in welchem Modus ausgeführt werden.
<b>Architektur-modell</b>	Definiert wesentliche Strukturen des geplanten Systems wie Schnittstellen zu anderen Systemen, die Dekomposition des Systems sowie Kommunikationsverbindungen zwischen den Systemkomponenten.

Tab. 1 Grobe Charakterisierung der in COSMOD-RE verwendeten Artefakttypen

Jedes in COSMOD-RE eingesetzte Anforderungs- oder Architekturmodell hat zwei wesentliche Bestandteile. Erstens beinhaltet ein Modell Diagramme, die die wesentlichen Modellelemente und deren Beziehungen grafisch dokumentieren. Zweitens beinhaltet jedes Modell textuelle Detailbeschreibungen der einzelnen Modellelemente. Damit soll sichergestellt werden, dass die Anforderungs- und Architekturartefakte möglichst verständlich und eindeutig interpretierbar sind. Die textuellen Detailbeschreibungen von Modellelementen werden mittels Schablonen strukturiert, die für den jeweiligen Artefakttyp die zu dokumentierenden Informationen festlegen.

Die in Tab. 1 aufgeführten Artefakttypen können auf jeder Abstraktionsebene eines Systems eingesetzt werden. Dies ist dadurch begründet, dass auf jeder Abstraktionsebene prinzipiell ähnliche Entwicklungsentscheidungen getroffen und die Übergänge zwischen verschiedenen Abstraktionsebenen möglichst frei von Brüchen sein sollten. Von der höchsten zur niedrigsten Abstraktionsebene hin steigt jedoch der Detaillierungsgrad der in den Modellen dokumentierten Anforderungen und Architekturinformationen.

### 2.3 COSMOD-RE Co-Design-Prozess

COSMOD-RE unterstützt durch so genannte „Co-Design“-Prozesse die verzahnte Entwicklung von Anforderungen und Architektur über mehrere Abstraktionsebenen hinweg. Abb. 2 zeigt die Struktur eines Co-Design-Prozesses in Form von Subprozessen und Informationsflüssen zwischen den Subprozessen. Ein Co-Design-Prozess verfolgt das Ziel, zunächst (lösungsneutrale) Anforderungen zu identifizieren und einen ersten, groben Architekturvorschlag zu erarbeiten. Erst nachdem die Problemstellung in Form von lösungsneutralen Anforderungsmodellen (Zielen und Szenarien) herausgearbeitet, und ein initialer Architekturvorschlag identifiziert wurden, erfolgt die Spezifikation detaillierter, lösungsorientierter Anforderungen.

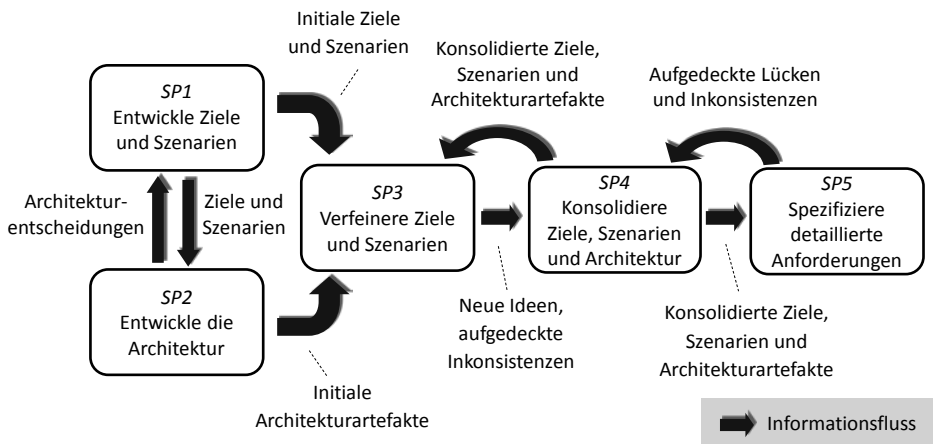


Abb. 2 Struktur eines COSMOD-RE Co-Design-Prozesses (siehe [PS09])

Die in Abb. 2 gezeigten Subprozesse können wie folgt charakterisiert werden:

- Subprozess  $SP_1$  ist für die Entwicklung initialer Ziele und Szenarien für das geplante System verantwortlich.
- Subprozess  $SP_2$  ist für die Entwicklung einer initialen, groben Systemarchitektur verantwortlich.
- Subprozess  $SP_3$  ist für die Definition verfeinerter Ziele und Szenarien und deren Zuordnung zu Architekturelemente verantwortlich.
- Subprozess  $SP_4$  ist für die Auflösung von Unstimmigkeiten zwischen Zielen, Szenarien und Architektur zuständig.
- Subprozess  $SP_5$  ist für die Spezifikation detaillierter, lösungsorientierter Anforderungen basierend auf den definierten und abgestimmten Zielen, Szenarien und Architekturartefakten zuständig.

### **3 Evaluation von COSMOD-RE in der Motorsteuerungs-Domäne**

Die nachfolgend beschriebene Evaluation von COSMOD-RE in der Motorsteuerungs-Domäne umfasst die folgenden, wesentlichen Evaluationsschritte:

- Definition von Evaluationskriterien durch Domänenexperten
- Dokumentbasierte Evaluation
- Geführte Anwendung von COSMOD-RE
- Evaluation des wahrgenommenen Nutzens und der Verständlichkeit des Ansatzes mittels eines Fragebogens

Im Folgenden werden zunächst der Entwicklungskontext und anschließend die einzelnen Evaluationsschritte und deren Ergebnisse beschrieben.

#### **3.1 Entwicklungskontext in der Motorsteuerungs-Domäne**

Die grundlegende Funktion einer Motorsteuerung ist die Umsetzung von Drehmomentanforderungen, die beispielsweise aus dem Beschleunigungswunsch des Fahrers resultieren. Dazu steuert die Motorsteuerung in geeigneter Weise den Verbrennungsprozess im Motor. Wesentliche Steuergrößen sind dabei Luftzufuhr, Kraftstoffzufuhr und Zündzeitpunkt (siehe Abb. 3). Eine umfassende Darstellung der Funktionen und der Systemstruktur einer Motorsteuerung ist z.B. in [BDJ03] zu finden.

Der Ausgangspunkt für die Entwicklung einer Motorsteuerung ist das Lastenheft des Fahrzeugherstellers. Basierend auf der Analyse der Anforderungen des Fahrzeugherstellers wird das Systemkonzept (meist eine existierende Plattform) festgelegt, auf deren Basis die Motorsteuerung realisiert wird. Bei der Funktionsentwicklung werden die Funktionen der Motorsteuerung ausgehend von den Anforderungen des Fahrzeugherstellers und des gewählten Systemkonzepts im Detail spezifiziert und in einem Pflichtenheft dokumentiert. Bei der Softwareentwicklung werden die spezifizierten Funktionen in Programmcode überführt. Die Software wird in

übergeordnete Steuerungsfunktionen (Anwendungssoftware) und Basisfunktionen unterteilt, welche der Ansteuerung von Hardwarekomponenten dienen.

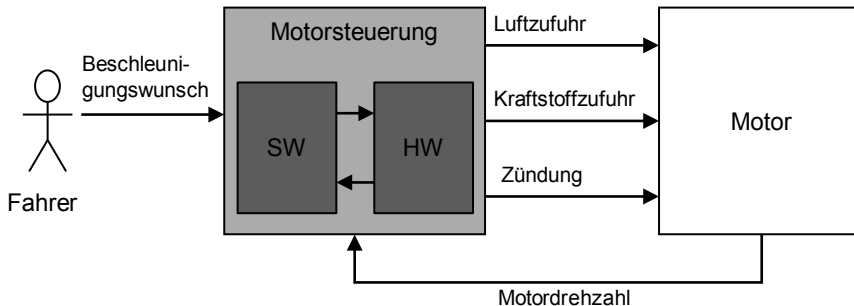


Abb. 3 Stark vereinfachtes, partielles Kontextmodell einer Motorsteuerung

### 3.2 Evaluationskriterien

Fokus der Evaluation von COSMOD-RE war die Durchführung des RE für die Anwendungssoftware (d.h. die Steuerungsfunktionen) der Motorsteuerung. Ziel war es somit, die Eignung von COSMOD-RE für diesen Teilbereich des Entwicklungsprozesses einer Motorsteuerung zu untersuchen. Zur Unterstützung der Evaluation des Ansatzes legten die Domänenexperten unter anderem die folgenden Evaluationskriterien fest:

1. Der Ansatz sollte die Spezifikation von Anforderungen auf zwei Abstraktionsebenen unterstützen (Produktebene und Systemebene).
2. Der Ansatz sollte den Übergang zwischen den Abstraktionsebenen unterstützen.
3. Der Ansatz sollte die Modellierung von zwei verschiedenen Problemklassen adressieren: diskrete und kontinuierliche Problemklassen.
4. Der Ansatz sollte die Modellierung verschiedener Technologien unterstützen: Software, digitale Hardware, Regelungstechnik und Mechanik.
5. Der Ansatz sollte einen durchgängigen RE-Prozess unterstützen.
6. Der Ansatz sollte einfach erlernbar sein.

### 3.3 Dokumentbasierte Evaluation

Die dokumentbasierte Evaluation von COMSOD-RE wurde von Domänenexperten basierend auf einer umfassenden Beschreibung des Ansatzes durchgeführt und erfolgte gegen die zuvor festgelegten Evaluationskriterien. Die Beschreibung des Ansatzes umfasste Definitionen der Grundprinzipien des Ansatzes, der zu erstellenden Entwicklungsartefakte, eine grobe Prozessdefinition sowie ein Beispiel, das den Ansatz veranschaulicht. Tab. 2 zeigt einen Überblick über die Ergebnisse der dokumentbasierten Evaluation des Ansatzes. Diese Ergebnisse spiegeln die subjektiven Eindrücke wieder, die die Domänenexperten aufgrund der dokumentbasierten Beschreibung des Ansatzes gewonnen haben.

Nr.	Kriterium	Erfüllungsgrad
1	Spezifikation von Produkt- und Systemanforderungen	erfüllt
2	Übergang von Produkt- zu Systemebene	teilweise erfüllt
3	Modellierung diskreter und kontinuierlicher Probleme	teilweise erfüllt (diskrete Probleme)
5	Modellierung der relevanten Technologien	teilweise erfüllt (Software)
6	Durchgängiger RE-Prozess	erfüllt
7	Einfache Erlernbarkeit	erfüllt

Tab. 2 Überblick über wesentliche Ergebnisse der dokumentbasierten Evaluation

### 3.4 Geführte Anwendung von COSMOD-RE

Da die dokumentbasierte Evaluation nur begrenzte Einblicke in Eigenschaften wie zum Beispiel die Durchgängigkeit des Ansatzes erlaubt, wurde zusätzlich ein gemeinsamer Workshop von Methoden- und Domänenexperten durchgeführt, bei dem der Ansatz exemplarisch zur Entwicklung von Anforderungen an die Anwendungssoftware einer Motorsteuerung eingesetzt wurde. Die Methodenexperten stellten dabei den Ansatz vor und unterstützten die Domänenexperten bei der Anwendung des Ansatzes. Im Folgenden werden die während der geführten Anwendung gewonnenen Erkenntnisse dargestellt. Die Erkenntnisse sind anhand der Bestandteile des COSMOD-RE-Ansatzes strukturiert.

#### Abstraktionsstufen

Die Unterstützung von Abstraktionsebenen war eine zentrale Anforderung an den Ansatz. Bei der geführten Anwendung des Ansatzes zeigte sich, dass während der Gewinnung von Anforderungen ein flexibler Umgang mit dem Abstraktionsstufenmodell von COSMOD-RE erforderlich ist:

- Der Ansatz muss auch dann anwendbar sein, wenn der Startpunkt des RE-Prozesses auf einer niedrigen Abstraktionsstufe liegt. In dem Beispiel wurde die Anwendungssoftware der Motorsteuerung als Gegenstand der Entwicklung gewählt. Somit lag der Startpunkt auf der Abstraktionsstufe „Hardware-Software-Partitionierung“. Die für die höheren Abstraktionsstufen relevanten Informationen wurden erst nach und nach ausgehend von diesem Startpunkt identifiziert. Dabei zeigte sich die Notwendigkeit, klare, domänenspezifische Kriterien vorzugeben, um die gewonnenen Informationen über das System jeweils in die richtige Abstraktionsstufe einzuordnen.
- Der Ansatz muss berücksichtigen, dass im Entwicklungsprozess binnen kurzer Zeit Informationen für mehrere verschiedene Abstraktionsstufen anfallen. Zum Beispiel kann es vorkommen, dass Ziele und Szenarien auf der höchsten Abstraktionsstufe und unmittelbar darauf lösungsorientierte Anforderungen auf einer sehr niedrigen Abstraktionsstufe definiert werden. Dies kann unter anderem damit begründet werden, dass Systemingenieure über ein umfassendes Wissen und Erfahrungen in Bezug auf die jeweilige Domäne verfügen und daher beispielsweise die Auswirkungen von Anforderungen einer höheren Abstraktionsstufe auf niedrigere Abstraktionsstufen unmittelbar, ohne mehrere Zwischenschritte bewerten können.

## Anforderungsartefakte

In Bezug auf die von COSMOD-RE unterstützten Artefakttypen wurden die folgenden Erkenntnisse gewonnen.

- ❑ Die Identifikation von Stakeholdern und die explizite Definition der Ziele der Stakeholder haben sich als sehr wichtig und hilfreich herausgestellt, um eine Ausgangsbasis für die Identifikation weiterer, detaillierter Anforderungen zu erhalten. Unter anderem wurden dabei wesentliche Entwicklungsziele der Motorsteuerungs-Domäne wie zum Beispiel das Ziel „Ressourcen-schonendes Fahren“ frühzeitig aufgedeckt und dokumentiert.
- ❑ Für eine erfolgreiche Anwendung der ziel- und szenariobasierten Vorgehensweise war es wesentlich, die Szenarien mit dem richtigen Abstraktionsgrad zu definieren. Dabei wurde von den Domänenexperten ein Abstraktionsgrad als hilfreich angesehen, bei dem die aus der Perspektive eines Systemingenieurs essenziellen Schritte zur Erfüllung des jeweiligen Ziels dokumentiert werden.
- ❑ Das Funktionsmodell, das Verhaltensmodell und das Architekturmodell erwiesen sich als sehr hilfreich, um die gewonnenen Informationen zu dokumentieren. Bei der Entwicklung dieser Modelle erwies sich ein Vorgehen sinnvoll, bei dem für jeweils ein spezifisches Ziel mehrere Szenarien definiert und dann für jedes einzelne Szenario die Architektur erweitert wird und die benötigten Softwarefunktionen und die benötigten Modi in den entsprechenden Modellen definiert werden.

## Co-Design-Prozess

Mit den von COSMOD-RE definierten Co-Design-Prozessen wurden die folgenden Erfahrungen gemacht:

- ❑ Prinzipiell wurden alle Subprozesse als sinnvoll erachtet und die Übergänge zwischen den Schritten als hinreichend durchgängig angesehen.
- ❑ Der Subprozess  $SP_2$  (Entwicklung einer initialen Architektur) wurde aufgrund des Startpunkts auf einer niedrigen Abstraktionsstufe (der Motormanagement-Anwendungssoftware) in einer modifizierten Form durchgeführt. Die zentrale Frage bei diesem Schritt war, wie die Einbettung der Software in das System gestaltet werden muss, um die identifizierten Szenarien erfüllen zu können. Dabei wurden beispielsweise notwendige Schnittstellen zu Sensoren, Aktuatoren sowie anderen Systemen im Fahrzeug identifiziert.
- ❑ Die identifizierte (System-)Architektur und die Szenarien wurden als gute Ausgangsbasis angesehen, um Softwarefunktionen zu identifizieren. Dabei zeigte sich die Notwendigkeit, die ausgehend von mehreren Szenarien identifizierten Softwarefunktionen in einem Folgeschritt zu konsolidieren (zum Beispiel durch Generalisierung oder Aggregation).

## 3.5 Evaluation mittels Fragebögen

Im Anschluss an die dokumentbasierte Evaluation und die geführte Anwendung von COSMOD-RE wurden mittels eines Fragebogens der wahrgenommene Nutzen und die Verständlichkeit des Ansatzes evaluiert. Der Fragebogen enthielt neben Fragen zum Hintergrund des Befragten jeweils Aussagen zum wahrgenommenen Nutzen und zur



Verständlichkeit des Ansatzes, die von den Domänenexperten auf einer 7-Punkt-Skala beantwortet wurden. Die Aussagen zum Nutzen wurden aus Hypothesen über die Vorteile des Ansatzes für den RE-Prozess abgeleitet. Zum Beispiel lautete eine Aussage „Durch die Verwendung des Ansatzes werden Fehler bei der Verfeinerung von Anforderungen vermieden“. Die Bewertung der Verständlichkeit des Ansatzes erfolgte mittels Aussagen wie zum Beispiel „Es wäre für mich einfach, den Ansatz anderen Mitarbeitern in meiner Abteilung zu erläutern.“

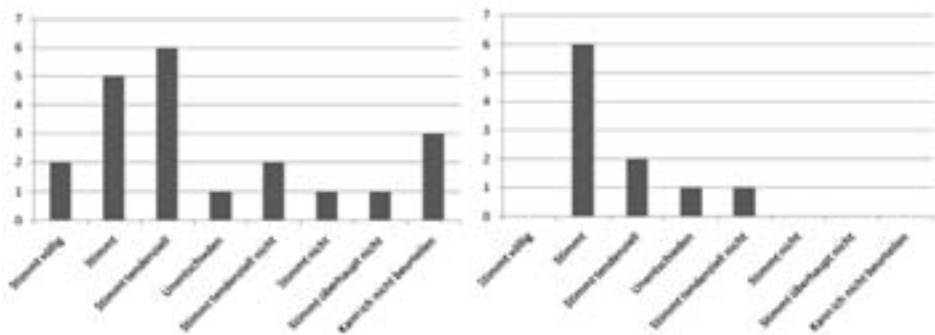


Abb. 4 Aggregierte Antworthäufigkeiten (*links*: Fragen zum wahrgenommenen Nutzen; *rechts*: Fragen zur Verständlichkeit und Erlernbarkeit)

Die aggregierten Häufigkeiten der Antworten auf Fragen zum wahrgenommenen Nutzen und zur Verständlichkeit sind in Abb. 4 dargestellt. Die Antworthäufigkeiten zeigen eine überwiegend positive Beurteilung des Ansatzes, weisen in einigen Detailfragen jedoch auch auf Verbesserungsmöglichkeiten hin. Beispielsweise wurde von den Befragten angeregt, eine Unterstützung für den Architekturentwurf basierend auf Zielen und Szenarien in den Ansatz zu integrieren. Zudem wurde der Bedarf nach einer Unterstützung für die automatisierte Prüfung von Qualitätseigenschaften wie Konsistenz und Vollständigkeit der Anforderungsmodelle festgestellt.

## 4 Schlussfolgerungen

Die eingesetzte Vorgehensweise zur Evaluation bestehend aus einer dokumentbasierten Evaluation, einer geführten Anwendung des Ansatzes durch Domänenexperten und einer schriftlichen Befragung hat sich bewährt, da jeder Schritt zu einem signifikanten Erkenntnisgewinn über den Nutzen und die Anwendbarkeit des Ansatzes für die betrachtete Domäne erbrachte. Die in Abschnitt 3 beschriebenen Evaluationsergebnisse legen folgende, zentrale Schlussfolgerungen nahe:

- Die Kombination aus Zielmodellierung, Szenariomodellierung, Funktionsmodellierung und Architekturmodellierung bildet eine gute Basis für die Realisierung eines durchgängigen RE-Prozesses in der anvisierten Domäne.
- Die explizite Unterstützung mehrerer Abstraktionsebenen wirkt sich in der betrachteten Domäne sehr positiv auf die Akzeptanz eines Ansatzes aus. Allerdings

können während der Entwicklung der Anforderungsmodelle Unklarheiten über die Einordnung der im Entwicklungsprozess anfallenden Informationen in die verschiedenen Abstraktionsebenen auftreten. Vor der Anwendung von COMSOD-RE empfiehlt es sich daher, mit den Domänenexperten einen Satz von Richtlinien zum Umgang mit Abstraktionsebenen während des RE-Prozesses zu definieren.

- Regelungstechnische Problemstellungen können durch Modellierungssprachen wie UML [OMG09] und SysML [OMG09] nur unzureichend abgebildet werden. Diese Problemstellungen sind für die Definition von System- und Softwareanforderungen in der betrachteten Domäne jedoch von zentraler Bedeutung. Zukünftige Versionen von COSMOD-RE sollten daher eine erweiterte Unterstützung für die Einbeziehung von regelungstechnischen Problemklassen in den RE-Prozess anbieten wie zum Beispiel Modellierungstechniken für hybride (diskret-kontinuierliche) Systeme.

Die weiteren Arbeiten an COSMOD-RE umfassen die systematische Evaluation des Ansatzes in unterschiedlichen Domänen, unter anderem, Avionik, Energietechnik und Automatisierungstechnik. Zudem wird der Ansatz basierend auf den gewonnenen Erkenntnissen und Erfahrungen überarbeitet und verbessert.

### **Danksagung**

Für die Mitwirkung an den beschriebenen Arbeiten danken wir Ulrich Freund, Franz Grzeschniok, Christine Haas, Peter Heidl, Igor Menzel, Mark Müller und Christoph Störmer. Wir danken Markus Ciolkowski für die kritische Begutachtung unseres Evaluationsfragebogens.

### **Literatur**

- [BDJ03] Bauer, H.; Dietsche, K., Jäger, T.: Gasoline-Engine Management – Motronic Systems. Yellow Jackets, Robert Bosch GmbH, 2003.
- [Po10] Pohl, K.: Requirements Engineering – Foundations, Principles, Techniques. Springer, erscheint im August 2010.
- [PK04] Puschnig, A.; Kolagari, R.T.: Requirements Engineering in the Development of Innovative Automotive Embedded Software Systems. In: Proc. 12th IEEE Int. Req. Eng. Conf., RE'04, Kyoto, Japan, IEEE Comp. Society, 2004, 328–333.
- [OMG08] Object Management Group: OMG Systems Modeling Language (OMG SysML). Version 1.1, 2008. <http://www.omg.org/spec/SysML/1.1/PDF/>
- [OMG09] Object Management Group: OMG Unified Modeling Language (OMG UML), Superstructure. Version 2.2, 2009. <http://www.omg.org/spec/UML/2.2/Superstructure/>
- [PS09] Sikora, E; Pohl, K.: COSMOD-RE – Verzahnung des Architektorentwurfs mit dem Requirements Engineering. OBJEKTSpektrum, Online-Themenspecial: Architekturen, 2009.