

TUM

INSTITUT FÜR INFORMATIK

Herausforderungen für das Requirements Engineering eingebetteter Systeme

Andreas Fleischmann, Eva Geisberger, Markus Pister



TUM-I0414
September 04

TECHNISCHE UNIVERSITÄT MÜNCHEN

TUM-INFO-09-I0414-0/1 .-FI

Alle Rechte vorbehalten

Nachdruck auch auszugsweise verboten

©2004

Druck: Institut für Informatik der
 Technischen Universität München

Inhaltsverzeichnis

Inhaltsverzeichnis	1
1 Einführung	2
Requirements Engineering (RE)	2
Eingebettete Systeme	3
Charakteristika eingebetteter Systeme	3
2 Steuerung technisch/physikalischer Prozesse	5
Modellieren technischer bzw. physikalischer Prozesse	5
Konkretisierung und Bewertung der Prozesse	6
Beschreibung echtzeit- und sicherheitskritischer Funktionen	6
3 Menschliche Kontrolle	8
Erfassen der Nutzungs-/Anwendungsprozesse	8
Identifizieren der Mensch-Maschinen-Schnittstellen	9
Spezifizieren des erforderlichen Nutzungsverhaltens	9
Spezifizieren einer transparenten Mensch-Maschine-Schnittstelle	10
4 Kombination Mechanik/Elektrik/Software	11
Gemeinsamer interdisziplinärer Systementwurf	11
Übergang digitaler zu analoger regelungstechnischer Steuerungslogik	12
Integration der verschiedenen interdisziplinären Sichten und Modelle	13
5 Einbettung in Systemlandschaft	14
Herausarbeiten und Beschreiben der Schnittstelle zur physikalischen Umwelt	14
Zusammenwirken mit anderen eingebetteten Systemen	15
Restriktionen für das Design	15
Modularisierung und Verwendung von Standards	16
6 Produktlinien	17
Umgang mit Variabilität	17
Zuordnung von Anforderungen zu Produkten und Plattform	18
Modularität und Standardisierung	19
Umgang mit Innovation und Concurrent Engineering	20
7 Zusammenfassung	21
Literatur	23

1 Einführung

Ziel des Projektes REFOCUS ist die Entwicklung einer Requirements Engineering Methodik für eingebettete Systeme. Insbesondere soll die Methodik die spezifischen Problemstellungen der Anwendungsdomänen eingebetteter Systeme adressieren. Im Folgenden werden die spezifischen Problemstellungen und Herausforderungen der Entwicklung eingebetteter Systeme aufgezeigt und entsprechende Anforderungen an das Requirements Engineering definiert.

Zu Beginn erfolgen eine knappe Klärung der Ziele und Aufgaben des Requirements Engineering und eine grundlegende Definition eingebetteter Systeme.

Requirements Engineering (RE)

Requirements Engineering ist innerhalb des Systems und Software Engineering eine iterative systematische Vorgehensweise mit dem Ziel, eine explizite, mit den Stakeholdern abgestimmte Anforderungs- und Systemspezifikation zu erstellen, die als Basis für die weitere Entwicklung dient. Kernaufgaben des Requirements Engineering sind das gemeinsame:

- Herausarbeiten (Elicitation)
- Analysieren der Anforderungen
- Spezifizieren von Anforderungen
- Validieren der Anforderungen
- Dokumentieren der Abstimmungsergebnisse und
- Verwalten der Artefakte des RE (Requirements Management).

Im RE sind unterschiedlichste Personengruppen mit verschiedenen Perspektiven und Interessen (Stakeholder) an der Produktentwicklung beteiligt. Es ist Aufgabe des RE, aus diesen unterschiedlichen, anfangs vagen, impliziten und oft widersprüchlichen Anforderungen der beteiligten Stakeholder ein gemeinsames Problem- und Zielverständnis und daraus abgeleitete präzise, vollständige und messbare Systemanforderungen zu erarbeiten.

Requirements Engineering findet im Kontext von Markt-/Kunden-Lieferanten-Beziehungen statt. Spezifikationsdokumente sind Bestandteil der Auftragsvergabe und von Verträgen. Daher ist die Abstimmung zwischen den Zielen und Anforderungen der markt-/kundenorientierten und der entwicklungsorientierten Stakeholder notwendig. Aufgabe ist

- die Definition einer Anforderungsspezifikation, die die Problemstellung der Nutzer/Kunden des zu entwickelnden Systems adäquat erfasst und aus dieser Sicht Anforderungen an Produkt und Produktentwicklung spezifiziert, und
- die Erarbeitung einer Systemspezifikation, die die definierten Problemanforderungen in Abstimmung mit Technologie-, gegebenen Systemanforderungen (Constraints), wirtschaftlichen und anderen Zielen der Produktentwicklung verfeinert und ein entsprechende Anforderungen an ein Systemkonzept spezifiziert.

Die erstellten Spezifikationsdokumente sind Grundlage für Abstimmungsprozesse und das Projektmanagement der gesamten Entwicklung. Abhängig von dem Anwendungsgebiet der Systementwicklung kommen spezifische Anforderungen an die Methodik des RE hinzu.

Eingebettete Systeme

Trotz der Unterschiede bezüglich Anwendungsgebiet und Komplexität heutiger eingebetteter Systeme – Beispiele sind Avionik-Steuerungen im Flugzeug, Türsteuerungen in der Automobiltechnik oder Produktionssteuerungen in der Automatisierungstechnik - trifft im Kern die folgende Definition für diese Systeme zu:

Eingebettete Systeme sind Software-/Hardwaresysteme, die als Teil eines größeren Systems für den Benutzer verborgen arbeiten, aber für die Funktions- und Leistungsfähigkeit des Gesamtsystems entscheidend sind. Ihre Aufgabe besteht darin, eine Anzahl technischer Geräte und physikalischer Prozesse der Umgebung zu überwachen und/oder gemäß einer definierten Funktionalität über Sensoren und Aktuatoren zu steuern und zu regeln. Sie sind reaktiv. D. h. abhängig von aktuellen Eingaben und Zustandsinformation der Umgebung werden die relevanten Größen und Prozesse über Aktuatoren gesteuert.

Diese Anwendungen sind meist zeit- (Echtzeitsysteme) und oft sicherheitskritisch. D.h. die Beschreibung von Sicherheitseigenschaften und die Formulierung von Zeitbedingungen sind wichtige Bestandteil der Spezifikation eines eingebetteten Systems.

Eingebettete Systeme werden von Menschen benutzt oder überwacht. Insbesondere bei sicherheitskritischen Anwendungen ist es notwendig, die Funktionalität und die Eingriffsmöglichkeit bei der Bedienung des eingebetteten Systems herauszuarbeiten und für den Menschen begreifbar/transparent zu vermitteln.

Neben der heutigen steigenden Komplexität, Verteiltheit und Parallelität der Steuerungsaufgaben eingebetteter Systeme findet eine ständige Entwicklung von dezidierten, auf Einzelanwendungen zugeschnittenen Anwendungen hin zu eingebetteten Systemen als Komponenten ganzer Produktsysteme mit verschiedenen Variationsmöglichkeiten für unterschiedliche Kunden- und Marktanforderungen statt.

RE-Charakteristika eingebetteter Systeme

Aus der obigen Definition und Beschreibung lassen sich wesentliche Aspekte des Requirements Engineering eingebetteter Systeme identifizieren, die besondere Herausforderungen an die Entwicklung dieser Anwendungen – insbesondere an das Herausarbeiten einer problemadäquaten Anforderungs- und Systemspezifikation – bedeuten:

- Eingebettete Systeme steuern technische/physikalische Anwendungsprozesse und technische Geräte.
- Diese Prozesse und Geräte werden von Menschen bedient und kontrolliert.

- Diese technischen Systeme werden zunehmend eingebettet in komplexe bestehende Systemlandschaften/Steuerungsprozesse mit verteilter Steuerungslogik und wechselnden Kontrollhierarchien.
- Eingebettete Systeme sind integriert in physikalisch-mechanische Komponenten, deren Verhalten und Wirken sie steuern bzw. wesentlich mitbestimmen, d.h. das Verhalten des Gesamtsystems wird bestimmt durch die Zusammenarbeit mechanischer, elektrischer und softwarebasierter Komponenten.
- Sie sind oft Teil einer für einen spezifischen globalen Markt entwickelten Produktlinie.

Diese spezifischen Aspekte eingebetteter Systeme werden in den folgenden Abschnitten anhand eines Anwendungsbeispiels im Automobilbereich - der Spezifikation einer Türsteuerung - erläutert, und es werden entsprechende Anforderungen an das Requirements Engineering eingebetteter Systeme abgeleitet.

Die erläuternden Beispiele (kursive Absätze) sind zum Teil der Spezifikation einer Türsteueranlage von DaimlerChrysler [HP02] entnommen, die dem aktuellen Stand der Praxis entspricht.

2 Steuerung technisch/physikalischer Prozesse

Ein eingebettetes System beeinflusst maßgebend das Verhalten eines technischen Gesamtsystems. Die Aufgabe liegt dabei in der Überwachung, in der Steuerung und in der Regelung von technischen und physikalischen Prozessen (siehe Definition). Dieser Umgang mit Prozessen fordert vom Requirements Engineering

- das Modellieren (Erfassen bzw. Definieren) der technischen bzw. physikalischen Prozesse und
- das Konkretisieren und Bewerten der Prozesse hinsichtlich ihres Nutzens.

Die Beeinflussung der physikalischen Umwelt kann bei einem Fehlverhalten des Systems große, teilweise lebensgefährliche Situationen (z.B. im Automobil) verursachen. Deshalb ist eine Aufgabe des Requirements Engineering

- das präzise und umfassende Beschreiben und Analysieren von Echtzeit- und sicherheitskritischen Funktionen.

Modellieren technischer bzw. physikalischer Prozesse

Die Steuerung physikalischer und technischer Prozesse fordert vom Requirements Engineering das Erfassen dieser Prozesse. Mit den Prozessbeschreibungen werden die Einsatzmöglichkeiten des Systems in der Umgebung erfasst (siehe [Dei98]) und damit kann auch die Aufgabe des Systems dargestellt, sowie der Nutzen des Systems bewertet werden.

*Der Fahrer **entriegelt** das Fahrzeug, steigt ein, fährt von a nach b, steigt aus und **verriegelt** das Fahrzeug.*

*Der Fahrer **entriegelt** das Fahrzeug, steigt ein, **verriegelt** das Fahrzeug, fährt von a nach b, **entriegelt** das Fahrzeug, steigt aus, und **verriegelt** das Fahrzeug.*

Neben der Aufgabenbeschreibung wird mit den Prozessbeschreibungen die funktionale Nutzungsschnittstelle beschrieben. Diese Nutzungsschnittstelle wird über konkrete Anwendungsprozesse beschrieben.

*Der Fahrer **schließt** das Fahrzeug an der Fahrer- oder Beifahrertüre auf und die **Zentralverriegelung entriegelt** alle Türen*

Eine Prozessbeschreibung wie in obigem Beispiel enthält eine Menge von Aktivitäten, die von Gegenständen der Umwelt angestoßen oder ausgeführt werden. Bei der Modellierung eines technischen Prozesses müssen diese Informationen über Aktivitäten und Beteiligte ermittelt werden.

Die Herausforderung für das Requirements Engineering ist es, Prozesse herauszuarbeiten, die die Gesamtaufgabe beschreiben, und Prozesse zu ermitteln, die die konkrete Nutzungsschnittstelle, also die unmittelbare Interaktion mit der Umgebung, darstellen. Hierbei ist zu berücksichtigen, dass sowohl die konkreten wie auch die allgemeinen Prozesse zueinander konsistent sein müssen.

Konkretisierung und Bewertung der Prozesse

Die Prozesse können, wie in vorherigem Abschnitt dargestellt, mit verschiedenen Präzisionsgraden beschrieben werden. Auf äußerer Ebene werden die übergreifenden Prozesse beschrieben, in denen noch keine besondere Rücksicht auf die detaillierte Gestaltung der Interaktion zwischen dem technischen Gesamtsystem und der Umgebung genommen wird (siehe erstes Beispiel in vorherigem Abschnitt).

Je präziser diese Prozesse jedoch beschrieben werden, desto mehr werden funktionale Designentscheidungen über die Schnittstelle gefällt ([GW03], [GS+04]).

Der Fahrer entriegelt das Fahrzeug: Der Fahrer schließt das Fahrzeug an der Fahrer- oder Beifahrertüre auf und die Zentralverriegelung entriegelt alle Türen.

→ *Alternative 1: Der Fahrer nähert sich mit einem Sender dem Fahrzeug und ab einer Entfernung von fünf Metern entriegelt die Zentralverriegelung alle Türen.*

→ *Alternative 2: Der Fahrer drückt auf den „Unlock“-Knopf der Fernbedienung und die Zentralverriegelung entriegelt alle Türen.*

Eine Herausforderung beim Requirements Engineering ist es, Entscheidungen zwischen verschiedenen Alternativen zu hinterfragen und für alle beteiligten Stakeholder nachvollziehbar zu machen.

Beschreibung echtzeit- und sicherheitskritischer Funktionen

Die Anwendung in einem physikalischen Umfeld erfordert, dass auf die Reize der Umwelt entsprechend schnell reagiert wird. Damit verbunden ist auch, dass in den meisten Fällen die Steuerung physikalischer Prozesse sicherheitskritisch ist (siehe [Lev95]). So ergeben sich zum Beispiel für eine Zentralverriegelung folgende Anforderungen:

Das Öffnen der vorderen Türen von innen ist immer möglich, unabhängig davon, ob die Zentralverriegelung offen oder geschlossen ist.

Die hinteren Türen lassen sich von innen nur öffnen, wenn die Kindersicherung ausgeschaltet ist.

Das Entriegeln der Türen des Fahrzeugs nach Erhalten des Öffnungsbefehls darf nicht länger als 0,5 Sekunden dauern.

Ein Fehlverhalten des Systems kann hier hohe Folgeschäden mit sich bringen. Dementsprechend ist eine umfassende Beschreibung des Verhaltens bei diesen Anwendungen notwendig, um frühzeitig Fehler zu vermeiden.

Eine Herausforderung für das Requirements Engineering ist es, Beschreibungen zu finden, die einen möglichst weiten Betrachtungswinkel haben, also möglichst viele Einflussfaktoren aus der Umwelt berücksichtigen, und unter Berücksichtigung dieser Faktoren ein möglichst vollständiges Verhalten zu spezifizieren. So stellen sich im Rahmen der Zentralverriegelung z. Bsp. die Fragen:

Was passiert, wenn bei einem Unfall die Kindersicherung an ist?

Was passiert, wenn der Nutzer bei angeschalteter Kindersicherung das Fahrzeug am vorderen Türschloss aufschließt?

Der Einsatz der Systeme in einer Anwendung fordert in besonderem Maße die Berücksichtigung der Menschen, die mit dem System umgehen müssen. Bereits in frühen Phasen ist daher ihre Einbeziehung bei Realisierungsentscheidungen erforderlich, wie im nächsten Abschnitt zu lesen.

3 Menschliche Kontrolle

Visuell ist das eingebettete System für den Menschen verborgen. Seine Funktionalität erfährt und steuert der Mensch über die physikalischen Schnittstellen des Gesamtsystems. Für das Requirements Engineering bedeutet dies, dass die Mensch-Maschinen-Schnittstelle in diesem Kontext zu erfassen, herauszuarbeiten und der weitere Übergang von der Gesamtsystem-Schnittstelle zum eingebetteten System zu definieren ist.

Eingebettete Systeme unterstützen den Menschen bei der Nutzung, Steuerung und Bedienung von technischen Maschinen und Geräten im Rahmen spezifischer Anwendungsprozesse und Situationen. Beispielhafte Anwendungsprozesse für die Nutzung der Türverriegelung beim Automobil sind:

Der Fahrer entriegelt das Fahrzeug, steigt ein, fährt von a nach b, steigt aus und verriegelt das Fahrzeug.

Der Fahrer entriegelt das Fahrzeug, steigt ein, verriegelt das Fahrzeug, fährt von a nach b, entriegelt das Fahrzeug, steigt aus, und verriegelt das Fahrzeug.

Der Fahrer schließt das Fahrzeug an der Fahrer- oder Beifahrertüre auf und die Zentralverriegelung entriegelt alle Türen.

Wesentliche Aufgabe des Requirements Engineering ist es daher, die relevanten Prozesse und Situationen aus Sicht der Mensch-Maschinen-Nutzung zu erfassen und alle Anforderungen an die Schnittstellen und Interaktionen zwischen Mensch und Maschine transparent zu spezifizieren (transparentes Spezifizieren der Nutzungsfunktionalität).

Im Einzelnen sind hier die Aufgaben des Requirements Engineering:

- das Erfassen aller relevanter insbesondere kritischer Nutzungs-/Anwendungsprozesse und Situationen,
- das Identifizieren und Berücksichtigen aller dabei beteiligten Schnittstellen (Interaktionen) zwischen Mensch und Maschine
- das Spezifizieren des erforderlichen Nutzungsverhaltens
- die Definition einer transparenten Bedienungsschnittstelle auf Ebene des Gesamtsystems und der zugehörigen Schnittstellen zum eingebetteten System.

Erfassen aller relevanten insbesondere kritischer Nutzungs-/Anwendungsprozesse

Wichtig ist es, möglichst alle für den Menschen relevanten Einsatzsituationen des technischen Gerätes zu erfassen und die Anforderungen an das in dieser Situation notwendige Systemverhalten zu spezifizieren. (siehe auch vorheriges Kapitel). So ist im Beispiel der Zentralverriegelung folgende Frage zu untersuchen:

Wie soll das Systemverhalten im Falle eines Unfalles mit verriegelten Türen aussehen? Die Insassen sind evtl. eingeschlossen und können von Außen nicht gerettet werden. Eine mögliche Forderung daraus wäre, dass die Zentralver-

riegelung in diesem Falle sofort alle Türen entriegelt, eine Andere, dass es einen Notknopf gibt, der in der Situation leicht und schnell zu bedienen ist.

In diesem Fall ist es notwendig, diese Problemsituationen zu analysieren, alternative Lösungsmöglichkeiten abzuwägen, zu entscheiden und die entsprechenden Systemanforderungen zu spezifizieren.

Identifizieren aller relevanten (auch physikalischen) Mensch-Maschinen-Schnittstellen

Für die Erarbeitung situationsgerechter Anforderungen an die Nutzungsschnittstelle ist es notwendig, mit Hilfe der Beschreibung der Anwendungsprozesse alle Interaktionen zwischen Mensch und Maschine zu identifizieren und in die Spezifikation miteinzubeziehen. Dazu gehören insbesondere auch alle direkten (physikalische) Interaktionen, die nicht durch explizite Ein- oder Ausgaben in das Steuerungssystem erfolgen. Das Beispiel aus der Türverriegelung macht das deutlich:

Der Fahrer entriegelt das Fahrzeug, steigt ein, fährt von a nach b, steigt aus und verriegelt das Fahrzeug.

Wie entriegelt der Fahrer das Fahrzeug? Mit dem Schlüssel mechanisch? Per Knopfdruck mit der Fernbedienung? Woran erkennt der Fahrer ob die Türen entriegelt sind oder nicht? An dem Geräusch? An einem Aufleuchten der Lampen? Sind alle Türen entriegelt worden? Gibt es Anzeigen über den Zustand der Verriegelung/Türen?

Aus diesem Beispiel ist ersichtlich, dass es notwendig ist, die physikalische Umgebung und Systemschnittstellen anschaulich/realistisch zu erfassen und zu beschreiben, um mögliche Systeminteraktionen zu spezifizieren und entsprechend den Bedürfnissen der Nutzer zu definieren.

Spezifizieren des erforderlichen Nutzungsverhaltens

Diese Analyse der Mensch-Maschinen-Schnittstelle bedeutet insbesondere, dass bei der Spezifikation der funktionalen Schnittstellen zwischen Mensch und Maschinen nicht das technische System im Vordergrund steht, sondern der Mensch und seine Nutzungsprozesse. Dem Menschen sollen nur die Informationen zur Verfügung gestellt werden, die für ihn – insbesondere aus sicherheitstechnischen Gründen – relevant sind und einer aktuell sinnvollen und möglichen Bedienung bedürfen ([Lev95], [Wei03]).

Für das Requirements Engineering erfordert dies eine Betrachtung und Strukturierung der Bedienprozesse nach Dringlichkeit (z.B. nach Komfort- und sicherheitskritischen Prozessen), sowie ihrer Beziehungen untereinander, und eine entsprechende Definition der funktionalen Mensch-Maschinen-Interaktion.

Im sicherheitskritischen Bereich ist eine Analyse komplexer Kontroll- und Entscheidungsaufgaben und möglicher Fehler bzw. Fehlbedienung erforderlich („Task and Human Error Analysis Techniques“ [Lev05]).

Spezifizieren einer transparenten Mensch-Maschine-Schnittstelle

Essentiell für die Entwicklung eingebetteter Systeme ist die Definition einer transparenten Mensch-Maschinen-Schnittstelle. Transparent heißt, dass das Verhalten des Systems für die Nutzer durchschaubar und an ihre jeweiligen Bedürfnisse und Fähigkeiten angepasst ist. Gerade in kritischen Situationen müssen die (möglichen) Folgen einer Bedienung für den Benutzer erkennbar und verständlich sein [Lev95]. Die Funktionalität des Systems, seine Eingriffsmöglichkeiten und deren Wirkung müssen für den Benutzer intuitiv erfahrbar definiert und vermittelt werden.

Beispiel der Zentralverriegelung: Falls die Funktion „Entriegeln aller Türen durch die Zentralverriegelung“ von einem Insassen bedient wird, werden dabei auch die rückwärtigen Türen, die durch die Kindersicherung verschlossen sind entriegelt. Falls das der Fall ist, und Kinder hinten sitzen, muss das dem Fahrer mitgeteilt werden, ansonsten kann er nicht situationsgerecht reagieren und die Kinder sichern.

Abgesehen von der Frage, ob der Fahrer diese Funktionalität so will, ist bei der Definition dieser Schnittstelle darauf zu achten, dass sie dem jeweiligen Nutzer entsprechend vermittelt wird und er die Funktionalität seinen Bedürfnissen anpassen kann.

Neben der transparenten und den Erfordernissen entsprechenden Definition und Vermittlung der Nutzungsfunktionalität ist es Aufgabe des RE unterschiedliche „Nutzerprofile“ zu berücksichtigen ([Wei03], [Maa94]). Die verschiedenen Bedürfnisse, Erwartungen und Fähigkeiten der möglichen Nutzer sind herauszuarbeiten und bei der Bestimmung der Systemanforderungen und der Anforderungen an die Mensch-Maschinen-Schnittstelle zu berücksichtigen. Das bedeutet,

- dass Systemkonzept und Funktionalität nach den Bedürfnissen/Fähigkeiten der Nutzergruppen auszurichten sind,
- dass Systemkonzept und Funktionalität verständlich zu definieren sind bzw. eine anschauliche intuitive Gestaltung und Vermittlung der Mensch-Maschinen-Schnittstelle und Bedienführung notwendig ist und Nutzer der Systeme beim Aufbau ihrer mentalen Modelle der Systemfunktionalität unterstützt werden müssen.

4 Kombination Mechanik/Elektrik/Software

Die Steuerungsaufgaben heutiger eingebetteter Systeme sind zunehmend komplexer und verteilt auf verschiedene Komponenten heterogener Systemlandschaften, die parallel und in Kooperation arbeiten. Mechanische und softwarebasierte Komponenten bilden ein integriertes Gesamtsystem, in dem die Software in steigendem Maße die wesentlichen Steuerungsaufgaben übernimmt. Damit verbunden ist, dass die traditionell von der Maschinenbau-Disziplin beherrschte Systementwicklung mit ihren räumlich konstruktiven und physikalischen Sichtweisen zunehmend funktionale und abstrakte Betrachtungen des Anwendungs- und Systemverhaltens in den Mittelpunkt der Systemspezifikation stellen muss.

Diese Charakteristika der Systeme und ihrer Entwicklung erfordern für das RE eingebetteter Systeme:

- die Anforderungen an das Verhalten des zu entwickelnde Produkts auf Gesamtsystemebene gemeinsam mit allen beteiligten Disziplinen Mechanik, Steuerungssoftware und Elektronik/Elektrik zu erarbeiten und zu definieren (siehe dazu die vorherigen Abschnitte *Steuerung technisch physikalischer Prozesse* und *Menschliche Kontrolle*).
- den Systementwurf gemeinsam und interdisziplinär durchzuführen,
- den Übergang von der digitalen zu analogen regelungstechnischen Steuerungslogik zu spezifizieren,
- die Integration der verschiedenen Sichten und Modelle für eine interdisziplinäre Spezifikation zu ermöglichen
- die Einbettung des Systems in eine komplexe interdisziplinäre Systemlandschaft zu beschreiben (siehe dazu den nachfolgenden Abschnitt *Einbettung in Systemlandschaft*).

Gemeinsamer interdisziplinärer Systementwurf

Wie in den vorhergehenden Abschnitten erläutert ist es notwendig die Anforderungen an das Systemverhalten auf Gesamtsystemebene gemeinsam und in gegenseitiger Abstimmung der der mechanischen, elektrischen und softwaretechnischen Erfordernisse und Restriktionen durchzuführen. Das Selbe gilt für den gemeinsamen interdisziplinären Systementwurf: Da der Entwurf der physikalischen Hardware und der Steuerungsfunktionalität des eingebetteten Systems voneinander abhängen, ist es notwendig, die Systemspezifikation der Mechanik, Elektrik und Steuerungsfunktionen parallel und in gegenseitiger Abstimmung zu erstellen und die Schnittstellen - insbesondere die Interaktionen zwischen diesen Disziplinen - gemeinsam zu spezifizieren. Diese Schnittstellenspezifikationen sind Grundlagen für den weiteren disziplinspezifischen Entwurf/Realisierung und die Systemintegration ([GW03], [GS+04], [VA EPS 02]).

So muss zum Beispiel die Frage wie die Kindersicherung der Zentralverriegelung im Auto realisiert wird Ergebnis intensiver Abstimmung zwischen Konstruktion, Elektrik/Elektronik und Softwareentwicklung sein. Soll etwa die Kindersicherung

automatisch bei Erreichen einer bestimmten Geschwindigkeit eingeschaltet werden, bedarf es einer entsprechenden Schnittstelle zwischen Drehzahlmesser (oder anderen Messsystemen) und Verriegelungssteuerung. Soll es ein Einschalten der Kindersicherung durch „Knopfdruck“ geben, muss die Schnittstellen zwischen „Bedienknopf“ und Softwaresteuerung definiert werden. Ist zusätzlich ein mechanisches (von Hand) Einschalten der Kindersicherung möglich ist die Schnittstelle insbesondere die Kooperation zwischen diesen verschiedenen Systemen abzustimmen und zu entwerfen.

Eine wesentliche Aufgabe des interdisziplinären Systementwurfes ist die Entscheidung über die Realisierung der einzelnen Steuerungsfunktionen. Abhängig von Performanz-, Kosten- und Sicherheits-/Zuverlässigkeitsaspekten müssen sie durch Software oder Hardware bzw. durch zu entwickelnde oder bestehende Komponenten realisiert werden. Aufgabe des RE ist es, hier diese Rahmenbedingungen und Constraints herauszuarbeiten, gemeinsam zu bewerten und entsprechend der Problemstellung zu definieren.

Übergang digitaler zu analoger regelungstechnischer Steuerungslogik

Kennzeichnend für die Schnittstellen des eingebetteten Systems zur Mechanik ist der Übergang von der digitalen diskreten Steuerungslogik zur analogen regelungstechnischen Darstellung physikalischer Prozesse (mittels Sensoren und Aktuatoren).

Betrachtet man die Steuerung der Türverriegelung im Auto, mit der Aufgabe, im Falle des Überschreitens einer bestimmten Fahrgeschwindigkeit automatisch die Türen des Autos zu verriegeln, so gilt es zu spezifizieren wie die Geschwindigkeit des Fahrzeugs „gemessen“ und im Sinne der Anwendung definiert werden kann, wie die Verriegelung der Türschlösser physikalisch erfolgen soll (Motorsteuerung, magnetische Steuerung oder anderes), wie das physikalische Prinzip dieser Steuerung ist, welche Steuergrößen dazu notwendig und welche Steuerungsinformationen berechnet und an entsprechende Aktuatoren gesendet werden müssen.

Für das Requirements Engineering bedeutet dies,

- dass eine geeignete Abbildung der relevanten physikalischen Systeme und ihrer Funktionsweise in die ereignisgesteuerte zustandsbasierte Logik der Steuerungssoftware spezifiziert werden muss,
- dass die komplexen Wirkbeziehungen zwischen physikalischen Funktionen modelliert und die Schnittstellen zur digitalen Logik spezifiziert werden müssen.

Insbesondere ist es notwendig, die grundlegende physikalische Funktionsweise und die zu steuernden Größen der Anwendung zu spezifizieren (z. B. Funktionsweise eines Stepmotors oder Geschwindigkeitsmessung im Auto).

Integration der verschiedenen interdisziplinären Sichten und Modelle

Wie bereits erläutert werden eingebettete Systeme als zentraler Teil eines Gesamtsystems aus Mechanik, Elektrik/Elektronik und Software entwickelt. Wichtige Stakeholder für die gemeinsame Erarbeitung der Anforderungsspezifikation und die Systemdefinition sind

- Kunden und Nutzer des Systems
- Marketing
- Maschinenbau-Ingenieure, Mechaniker
- Elektro-Ingenieure, Elektriker
- Mess- und Regelungstechniker
- Software-Ingenieure
- Produktmanager, Plattformmanager/Entwickler (bei Produktlinien)

Jede dieser Disziplinen hat entsprechend ihres Hintergrundes und ihrer Zielsetzung eine eigene Sicht auf das Produkt, eigene Denkstrukturen, Sprachen, Modelle und Vorgehensweise. Traditionell sind in den Anwendungsgebieten eingebetteter Systeme eine mechanisch-konstruktive Sichtweise und regelungstechnische Beschreibungsmodelle für die Steuerung von Systemverhalten vorherrschend.

Um den genannten Herausforderungen der Entwicklung eingebetteter Systeme begegnen und eine adäquate Spezifikation der Anforderungen an das Verhalten des zu entwickelnden Systems und seiner Eigenschaften erstellen zu können, ist es notwendig grundlegende logische Beschreibungsmethode und Techniken zu entwickeln bzw. einzusetzen, die

- die logische abstrakte Beschreibung von Anwendungsprozessen, Systemverhalten, Systemstrukturen unterstützen,
- es erlauben, bewährte Modelle und Sichten der einzelnen Disziplinen zu integrieren und miteinander in Beziehung setzen,
- es unterstützen, die geforderten Eigenschaften und Constraints der Kunden, Nutzer und Ingenieure auf diese Modelle abzubilden,
- die gemeinsame Abstimmung und Bewertung der Anforderungs- und Systemspezifikation nach Wirtschaftlichkeit und Machbarkeit ermöglichen,
- die Definition und Spezifikation der Schnittstellen zwischen den Disziplinen einfordern, die wiederum Grundlage für Entwicklung und Abstimmungsaufgaben bilden.

Siehe hierzu auch [ProMiS02], [VA EPS 02], [GW03].

5 Einbettung in Systemlandschaft

Innerhalb eines technischen Gesamtsystems hat ein eingebettetes System die Aufgabe, eine Anzahl technischer Geräte und physikalische Prozesse der Umgebung zu überwachen, zu regeln oder zu steuern. Als Software-/Hardwaressystem kann es diese Funktionalität nach außen nur zusammen mit elektrischen und physikalischen Komponenten zur Verfügung stellen. Diese Tatsache fordert vom Requirements Engineering

- das Herausarbeiten und Beschreiben der Schnittstelle zur physikalischen Umwelt des eingebetteten Systems.

Des Weiteren beinhalten komplexe technische Gesamtsysteme aufgrund der Modularisierung meist mehrere eingebettete Systeme. Dies führt dazu, dass eine Funktionalität des Gesamtsystems in Zusammenarbeit mit anderen eingebetteten Systemen gestellt wird. Die Einbettung eines SW-/HW-Systems in diese Systemlandschaft fordert vom Requirements Engineering folgende Aufgaben:

- Das Zusammenwirken mit anderen eingebetteten Systemen muss geklärt und die Schnittstelle zu diesen Systemen beschrieben werden.
- Die sich aus der Systemlandschaft ergebenden Restriktionen für das Design des eingebetteten Systems müssen ermittelt und beschrieben werden.
- Funktionen und Komponenten eingebetteter Systeme müssen modular und mit Hilfe standardisierter Schnittstellen aufgebaut werden.

Herausarbeiten und Beschreiben der Schnittstelle zur physikalischen Umwelt

Die Schnittstelle zur physikalischen Umwelt setzt sich per Definition aus Aktuatoren und Sensoren zusammen (siehe [Fie99]). Ein Sensor übermittelt dem eingebetteten System Reize von der Umwelt und ein Aktuator übernimmt Steuerungsaufgaben (siehe [Bro97]).

Jeweils ein Aktuator, der Befehle zum Öffnen und Schließen entgegennimmt, soll die Verriegelung der jeweiligen Türe steuern. Ob eine Türe verriegelt ist, kann jeweils von einem Sensor angegeben werden. Jede Tür hat einen Sensor, der dem System mitteilt, ob diese offen ist.

Ebenso hat jeder Türgriff einen Sensor, der angibt, ob der Griff angezogen wird, und einen Sensor, der die Türschlossstellung angibt.

Es existieren oft bereits fertige Aktuatoren/Sensoren und Komponenten anderen Disziplinen, oder diese werden zumindest parallel zum Requirements Engineering (und nicht im Anschluss daran) entwickelt. Beim Herausarbeiten dieser Schnittstellen wird die Beschreibung stets konkreter. Je konkreter eine Beschreibung wird, desto mehr müssen die Lösungen der anderen Disziplinen abgestimmt bzw. berücksichtigt werden. Das letztendliche Ziel ist es, den Übergang von der digitalen Signalwelt zu analogen Strömen als Schnittstelle zur Elektrik zu beschreiben.

Aktuator: Die Verriegelung der Türen wird über einen Elektromotor gesteuert. (Charakteristik: 12V für 2 Sek. +- 0.3 Sek. verriegelt, -12 V für 2 Sek. +-0,3 Sek. entriegelt.) Gesteuert über Anschluss (Signal) ZENTR_MOT1 (Pin 20) und ZNTR_MOT2 (Pin 21).

Die Beschreibung einer disziplinübergreifenden Schnittstelle wie in obigem Beispiel muss beide Disziplinen berücksichtigen. Aus elektrotechnischer Sicht liegt der Schwerpunkt auf analogen Strömen, die an Pins angelegt werden. Aus informationstechnischer Sicht wiederum liegt der Schwerpunkt auf dem Zusammenhang der logischen Signale und der daraus resultierenden Wirkung (z. B. ZENTR_MOT1 für 2 Sek -> Türe verriegelt).

Zusammenwirken mit anderen eingebetteten Systemen

Eingebettete Systeme haben zunehmend komplexere Steuerungsaufgaben. Die Steuerung der physikalischen bzw. technischen Prozesse wird in Kooperation mit anderen eingebetteten Systemen geleistet. Je nach Einsatzaufgabe herrschen unterschiedliche Kontrollhierarchien zwischen den verschiedenen Steuerungskomponenten. Ein maßgebender Faktor ist, dass die einzelnen Systeme nicht neu bzw. parallel entwickelt werden, sondern neu hinzukommende Steuerungsaufgaben und Komponenten in eine bestehende Systemlandschaft integriert bzw. auf einander abgestimmt werden müssen ([Fie99], [GW03]). Aufgabe des Requirements Engineering ist hier die Definition der Schnittstellen zwischen diesen eingebetteten Systemen.

Die Auswertung und Anzeige der Fehlermeldungen, die von der Zentralverriegelung gemeldet werden, übernimmt das Kombiinstrument hinter dem Lenkrad. Dieses entnimmt dem CAN-Bus die spontan auftretenden Signale B_LOW_KEY (id:6ff, Bit 2) und ERROR_KEY (Id:6ff, Bit 3).

Wie in diesem Beispiel dargestellt, wird die Aufgabe, dem Fahrer Informationen über den Zustand der Zentralverriegelung zu geben, zusammen mit einem Kombiinstrument geleistet. Die Beschreibung der Schnittstelle zu diesem System ist ein essentieller Bestandteil einer Spezifikation. Hier ist es aus Sicht des Requirements Engineering notwendig zu klären, welches System welche Aufgabe übernehmen soll und wie sich die Systeme beeinflussen.

Restriktionen für das Design

Neben der Betrachtung dessen, wie ein eingebettetes System aus funktionaler Sicht mit anderen Systemen interagiert, spielt auch die Berücksichtigung der gemeinsamen Ressourcen eine Rolle. Das Vorhandensein bestimmter konstruktiver Vorgaben, die z.B. aus bereits existierenden technischen Gesamtsystemen hervorgehen, kann weitere Anforderungen an das eingebettete System zur Folge haben.

Es gibt bereits jeweils ein Steuergerät in den beiden vorderen Türen. Auf diesen soll die Funktionalität aus Kostengründen implementiert werden. Die Kommunikation der Geräte erfolgt über den systemweiten CAN-BUS.

Mit dem Signal ZV_SCHL_L teilt das linke Steuergerät einen Schließbefehl mit ...

Im obigen Beispiel teilen sich mehrere eingebettete Systeme die gemeinsame Ressource CAN-Bus für ihre Kommunikation. Hier ist es im Rahmen des RE notwendig, die Abstimmung auf der Ressource zu klären, da es ohne diese Abstimmung zu Konflikten mit anderen bestehenden oder zu entwickelnden Systemen kommen kann.

Modularisierung und Verwendung von Standards

Die Integration in eine Systemlandschaft bedeutet für die Anforderungsanalyse, dass all die daraus resultierenden Anforderungen, Rahmen- und Randbedingungen herausgearbeitet und entsprechende Systemanforderungen und Constraints an das zu entwickelnde System, aber auch an die bestehenden Nachbarsysteme definiert bzw. berücksichtigt werden müssen. Abbildung 1 stellt nochmals die herauszuarbeitenden internen Schnittstellen eines technischen Gesamtsystems dar. Wichtig ist hierbei die stetige Berücksichtigung der in der Zeichnung nicht dargestellten Nutzungsschnittstelle des Gesamtsystems, die mit den internen Schnittstellen harmonisieren muss.

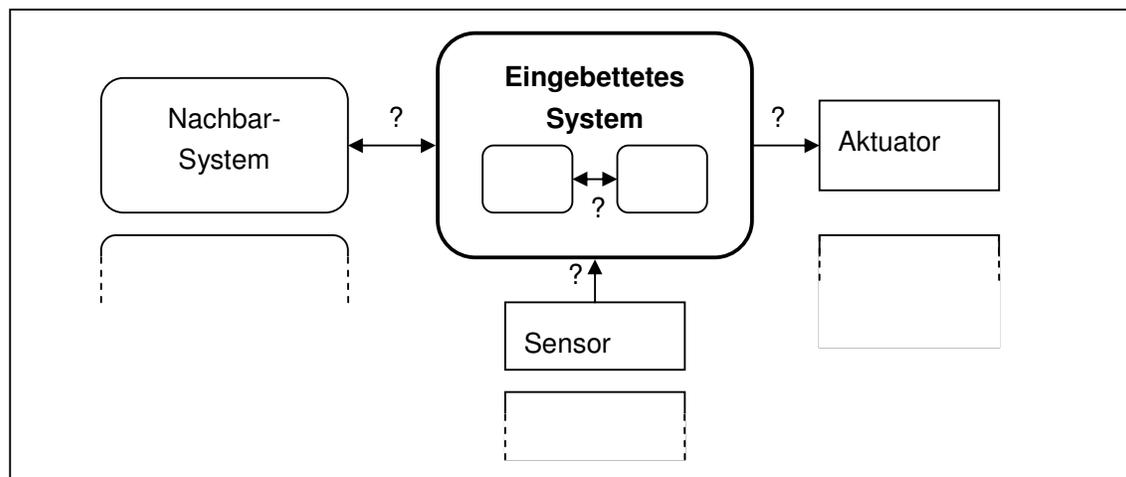


Abbildung 1: Herauszuarbeitende Schnittstellen bei der Einbettung in eine Systemlandschaft

Um die Komplexität der Systemlandschaft bewältigen zu können, ist es notwendig, Systemanforderungen und Design der zu entwickelnden Komponenten mit Hilfe standardisierter und modular gestalteter Dienst- und Schnittstellendefinition zu spezifizieren. Dies erleichtert einen flexiblen und allgemein verständlichen Aufbau der Systeme und somit die Möglichkeiten, Systeme mit einem angemessenen Aufwand in eine komplexe Systemlandschaft einzubetten. Die Herausforderung an das Requirements Engineering ist es, eine passende Strukturierung mit Modulen zu finden und so weit wie möglich Standards zu verwenden.

6 Produktlinien

Eine Produktlinie/Produktfamilie ist eine Menge von gemeinsam verwalteten Produkten, die einerseits eine definierte Menge von gemeinsamen Funktionen und Eigenschaften haben (die "Plattform", die gemeinsame Kernfunktionalität der Produktlinie enthält), und die sich andererseits in spezifischen Punkten voneinander unterscheiden (die konkreten Produkte der Produktlinie) [BN+00]. Produktlinien spielen bei eingebetteten Systemen eine große Rolle.

Zum einen werden Aufträge für eingebettete Systeme (zum Beispiel für eine Türsteuerung für einen Mercedes) oft an Unterauftragnehmer (zum Beispiel Bosch) vergeben, die ähnliche Systeme auch für andere Kunden (zum Beispiel BMW, Audi) entwerfen. Zum anderen erfordern Baureihen (zum Beispiel die Mercedes S-Klasse) entsprechende Produktlinien auf Seite der eingebetteten Systeme.

Zielsetzung des Entwickelns von solchen Systemen in Form von Produktlinien ist es, durch Ausnutzung von Gemeinsamkeiten verwandter Produkte Wiederverwendungsmöglichkeiten nutzen zu können. Dieses Produktlinien Engineering gliedert sich in der Regel in zwei Prozesse: Im so genannten Domain Engineering geht es um die Erfassung der Gemeinsamkeiten und u.a. um die Definition der grundlegenden Architektur/Plattform für die gesamte Produktlinie, während im Application Engineering ausgehend von aktuellen Produkthanforderungen und der gemeinsamen Basis durch Komposition, Anpassung und Neuentwicklung die spezifischen Produkte definiert werden.

Im folgenden Abschnitt wird beschrieben, welche Herausforderungen für das Requirements Engineering konkreter Systementwicklungen im Kontext von Produktlinien identifiziert werden können:

- Umgang mit Variabilität
- Zuordnung von Anforderungen zu Produkten und Plattform
- Modularität und Standardisierung
- Umgang mit Innovation und Concurrent Engineering

Diese Herausforderungen werden in den folgenden Unterabschnitten näher erläutert.

Umgang mit Variabilität

Die einzelnen Produkte innerhalb einer Produktlinie variieren in der Regel in einer ganzen Reihe von Funktionalitäten und Eigenschaften. Man unterscheidet dabei zwei Arten von Variabilität [BH+03]: Im Falle der optionalen Variabilität unterscheidet sich ein Produkt von einem anderen durch die Anwesenheit/Abwesenheit einer Funktion. Im Falle der alternativen Variabilität unterscheiden sich zwei Produkte in der Ausprägung einer Funktion.

	Mercedes C330 Limousine	Mercedes C320 Cabrio	Mercedes S500 Cabrio
alternative Variabilität	"Die Verriegelung der 4 Fahrzeugtüren wird daraufhin geöffnet"	"Die Verriegelung der 2 Fahrzeugtüren wird daraufhin geöffnet"	"Die Verriegelung der 2 Fahrzeugtüren wird daraufhin geöffnet"
optionale Variabilität	-	-	"Zum Auf/Ab- schließen kann der Benutzer einen Funk- sender einsetzen"

Tabelle 1: Beispiele für optionale und alternative Variabilität

Die Unterschiede zwischen den spezifischen Produkten einer Produktlinie bestehen in der Regel sowohl aus alternativen als auch aus optionalen Variabilitäten. Eine Herausforderung an das Requirements Engineering besteht daher darin,

- dass die Gemeinsamkeiten und die Unterschiede (sowohl alternative als auch optionale Variabilität) der einzelnen Produkte explizit modelliert werden können müssen,
- dass ausgehend von Gemeinsamkeiten und Variabilitäten spezifiziert werden muss, welche konkreten Produkte instanziiert werden,
- dass Beziehungen zwischen einzelnen Anforderungen bzw. einzelnen Anforderungsmodulen verfolgt werden müssen (Tracing), um festzustellen, wie sich Varianten zusammensetzen können.

Zuordnung von Anforderungen zu Produkten und Plattform

Im Rahmen des Requirements Engineering muss entschieden werden, ob Anforderungen sich an die gesamte Produktpalette richten, oder ob sie lediglich spezifische Produkte betreffen. Bei dieser Entscheidung ist zunächst wichtig, wie viele Produkte betroffen sind: ist die Mehrzahl der Produkte betroffen, wird eine Anforderung in der Regel der Plattform zugeordnet (Redundanzvermeidung). Es müssen aber noch weitere Gesichtspunkte berücksichtigt werden:

- **Architektur:** Bei der Bewertung und Einordnung von Anforderungen muss die bereits vorhandene Architektur berücksichtigt werden ([BD+04], [BH+03], [BN+00]); dies kann zum Beispiel durch intensive Kommunikation zwischen den Designern und den Anforderungsingenieuren gewährleistet werden, oder durch Übernahme der Architektur in die Spezifikation in Form von Rahmenbedingungen und Restriktionen.
- **Marketing/Strategie:** Produktlinien sind marktgetrieben und dienen neben der Kostenersparnis durch Redundanzvermeidung auch der schnelleren Reaktion auf Bedürfnisse des Markts. Bei der Bewertung und Einordnung einer Anforderung muss daher die Wahrscheinlichkeit, dass in Zukunft weitere Produkte (die bereits Teil der

Produktlinie sind, oder die ganz neu in der Produktlinie entstehen werden) diese Anforderung erfüllen werden müssen, berücksichtigt werden.

Im Lebenszyklus einer Produktlinie wird es nötig sein, die Zuordnung einer Anforderung zu Plattform oder spezifischen Produkten zu ändern [BN+00]. So kann sich eine spezifische Funktion eines Produkts bewähren, so dass auch für die anderen Produkte übernommen wird (d.h. in die Plattform verschoben wird); oder es kann eine Funktion überflüssig werden, so dass sie nur noch für wenige spezifische Produkte relevant ist und daher von der Plattform verschoben wird. Um diese Änderungen effizient durchführen zu können, ist es nötig, dass zum einen die Spezifikation modular aufgebaut ist, und dass zum anderen die Anforderungen mit den entsprechenden Elementen des Designs und der Implementierung verknüpft sind (Tracing).

Anforderungen, die in der Spezifikation einem konkreten Produkt zugeordnet wurden, können gleichzeitig in der Architektur aus marktstrategischen oder anderen Gründen der Plattform zugeordnet werden. Im Design des jeweiligen Produktes wird dann nur entschieden, ob eine Funktionalität der Plattform "frei geschaltet" wird oder nicht.

So kann zum Beispiel das Feature "Türöffnung per Funksender" zunächst nur in der Spezifikation des teureren Mercedes S500 vorgesehen sein. Da es aber eventuell mittelfristige Marktstrategie ist, einige Zeit später auch die S350 Reihe mit Funksendern auszustatten, kann es sich anbieten, dieses Feature von Anfang an in der gemeinsamen Plattform zu implementieren. So können später die S350 schneller und kostengünstiger mit dem neuen Feature aufgerüstet werden.

Modularität und Standardisierung

Um variabel mit verschiedenen Produkthanforderungen umgehen und Plattformen entsprechend flexibel aufbauen zu können, muss bereits die Spezifikation der Plattform und der spezifischen Produkte modular und standardisiert aufgebaut werden (Baustein-Prinzip). Eine Herausforderung an das Requirements Engineering ist, die Bausteine und ihre Abhängigkeiten (sowie Konsistenzbedingungen) explizit zu beschreiben, damit klar ist, wie die einzelnen Bausteine miteinander kombiniert und dadurch konkrete Produkte definiert werden können ([VA EPS 02], [BH+03]).

*Beispiel: Ist der Baustein "Funksender-Teilspezifikation" mit "[...] **die Türen** können auch per Funksender auf/abgeschlossen werden [...]" beschrieben, ist der Baustein (auf Spezifikationsebene) unabhängig davon, ob eine Autovariante mit 2 oder 4 Türen spezifiziert wird. Wurde hingegen etwa eine Formulierung wie "[...] **alle 4 Türen** können auch per Funksender auf/abgeschlossen werden [...]" benutzt, kann der Funksender nicht ohne weiteres in Kombination mit der Variante "Zweitürer" verwendet werden.*

Da im Leben einer Produktlinie viele verschiedene Artefakte verwaltet werden müssen, die in unterschiedlichsten Beziehungen zueinander stehen, wird ein komplexes Konfigurationsmanagement benötigt, um die verschiedenen Varianten verwalten zu können [BH+03, BN+00].

Produktlinien für eingebettete Systeme befinden sich oft in Marktsegmenten (Kontexten), in denen eine Standardisierung angestrebt wird oder bereits erreicht ist.

Im Beispiel der Zentralverriegelung könnten solche Standards zum Beispiel die Steckerbelegungen oder sicherheitskritische Spezifikationen des Funksenders betreffen.

Die sich daraus ergebende Herausforderung für Requirements Engineering liegt darin, dass zum einen Standards in Spezifikationen angemessen eingebunden und dargestellt werden müssen, und dass andererseits Spezifikationen leicht an neue Standards angepasst werden müssen.

Umgang mit Innovation und Concurrent Engineering

Neben dem Ziel der Produktlinienansätze durch die grundlegende Definition von Plattformen und Produktvariationen bei der Spezifikation neuer Produkte bereits existierende Spezifikationen wieder verwenden zu können, ist es insbesondere Aufgabe des Requirements Engineering die definierten Plattformen und Produktlinien zukünftigen Erfordernissen anzupassen (bzw. anpassen zu können). Eine langfristige strategische Planung von Produkten, in der Anforderungen nicht nur für die Gegenwart auf Produkte verteilt werden, sondern auch bereits über zukünftige Anforderungen an zukünftige Produktversionen entschieden wird (Releaseplanung) ist erforderlich. Entsprechend werden Produktlinie, Plattform und die spezifischen Produkte, die darauf aufsetzen, von verschiedenen Teams (Produktlinien-, Plattformteam und Produktteams) betreut und parallel entwickelt (Concurrent Engineering). Diese arbeiten mit unterschiedlichen manchmal entgegen gesetzten Zielsetzungen und Anforderungen, die es gilt intensiv zu analysieren und abzustimmen ([VA KPE 00], [BD+04], [Dei98]). Die damit verbundenen Herausforderungen an das Requirements Engineering sind:

- Bei der Definition eines neuen Produkts/Produktreleases unter Rückgriff auf bereits bestehenden Spezifikationen innerhalb der Produktlinie müssen die dort bereits existierenden Anforderungen und Rahmenbedingungen im neuen Kontext hinterfragt und ggf. verändert oder aussortiert werden.
- Es muss möglich sein, in der Spezifikation auch zukünftige Anforderungen formulieren zu können, bzw. Aussagen über zukünftige Releases zu treffen.
- die Kommunikation und Kooperation zwischen den verschiedenen Teams unterstützt werden muss, zum Beispiel durch geeignete Schnittstellen im Prozess oder gemeinsame Modelle.
- ganz unterschiedliche Arten von Anforderungen erfasst werden müssen (zum Beispiel konkrete Produkthanforderungen, langfristige strategische Entscheidungen).

7 Zusammenfassung

Die Eigenschaften eingebetteter Systeme führen beim Requirements Engineering zu einer Fülle von Herausforderungen an die Entwicklung und damit auch an das Herausarbeiten einer problemadäquaten Anforderungs- und Systemspezifikation dieser Systeme. Diese Eigenschaften und die daraus resultierenden Herausforderungen sind:

- Eingebettete Systeme steuern technische/physikalische Anwendungsprozesse und technische Geräte. Diese Anwendungsprozesse sind zu beschreiben, zu verfeinern und zu bewerten. Besonders bei sicherheitskritischen Systemen sind sie umfassend und detailliert zu ermitteln.
- Die Prozesse, die eingebettete Systeme steuern, werden von Menschen bedient und kontrolliert. Dies fordert das Erfassen aller relevanter, insbesondere kritischer Nutzungs-/Anwendungsprozesse und Situationen, sowie das Identifizieren und Berücksichtigen aller dabei beteiligten Schnittstellen und Interaktionen zwischen Mensch und Maschine. Um unerwartetes und unerwünschtes Verhalten des Systems auszuschließen gilt es komplexe Kontroll- und Entscheidungsaufgaben im Rahmen der Bedienung hinsichtlich möglicher Fehler zu analysieren und eine transparente und intuitive Bedienschnittstelle auf Ebene des Gesamtsystems zu definieren.
- Eingebettete Systeme sind in physikalisch-mechanische Komponenten integriert. Sie steuern bzw. beeinflussen das Verhalten des Gesamtsystems und das Zusammenspiel aller mechanischen, elektrischen und softwarebasierten Teilkomponenten. Dieses Verhalten ist gemeinsam zu spezifizieren; d.h. die Disziplinen müssen in einer gemeinsamen Sprache und mit Hilfe integrierter Modelle miteinander kommunizieren, die Anforderungen an das Gesamtsystem und seine Teilkomponenten miteinander abstimmen und die entsprechenden Schnittstellen zwischen den Disziplinen definieren. Insbesondere gilt es die Komponenten hinsichtlich ihrer abstrakten Funktionalität zu beschreiben und ausgehend davon den Übergang von der digitalen diskreten Steuerungslogik zur analogen regelungstechnischen Modellierung zu spezifizieren.
- Eingebettete Systeme werden zunehmend in komplexe bestehende Systemlandschaften/Steuerungsprozesse mit verteilter Steuerungslogik und wechselnden Kontrollhierarchien eingebettet. Dies bedeutet, dass sowohl die Schnittstellen zur physikalischen Umwelt, wie auch die Schnittstellen zu und das Zusammenspiel mit anderen eingebetteten Systemen und die sich daraus ergebenden Restriktionen geklärt werden müssen. Hinsichtlich der einfachen Einbettung in Systemlandschaften und deren zukünftigen Erweiterungen gilt es die Systeme bzw. ihre Schnittstellen modular und unter Verwendung von Standards aufzubauen.
- Eingebettete Systeme sind meist Teil einer für einen spezifischen globalen Markt entwickelten Produktlinie. Dies bedeutet, dass innerhalb einer Produktentwicklung die spezifischen Produkthanforderungen mit den langfristigen Zielen und Produktstrategien der Produktlinien abgestimmt werden müssen. Um den verschiedenen

Kundenwünschen entsprechende Produkte definieren zu können müssen Anforderungen variabel gestaltet und spezifischen Produkten und Plattformen zugeordnet werden können. Auch hier gilt es, hinsichtlich zukünftiger Erweiterungen und Änderungen der Produktstrategie und Plattformen (Systemlandschaften) Anforderungen, Systemkonzepte und Schnittstellen modular und unter Verwendung von Standards aufzubauen.

Die hier diskutierten Herausforderungen des Requirements Engineering eingebetteter Systeme resultieren neben den oben genannten inhaltlichen Aussagen in Anforderungen an den RE-Prozess:

- Es gilt, die Abstimmung und Bewertung von Anforderungen und Systemkonzepten zwischen den verschiedenen Disziplinen zu unterstützen bzw. einzufordern. Dazu gehört die Abstimmung der Ziele und Anforderungen
 - zwischen Productline Engineering und Application Engineering,
 - zwischen Marketing, Kunden und Entwicklungsingenieuren und
 - zwischen den Entwicklungsdisziplinen Mechanik, Elektrik/Elektronik, Software.
- Diese Abstimmungen müssen Teil eines iterativen Verfeinerungsprozesses sein, in dem Anforderungen und Lösungskonzepte sukzessive er-/überarbeitet, validiert und vervollständigt werden.
- Das RE muss auf Basis eines definierten und standardisierten Prozess mit festgelegten Iterationsstufen, Abstimmungskonzepten und Entscheidungspunkten/-kriterien erfolgen.
- Aufgrund des intensiven Abstimmungsbedarfes und einer potentiell hohen Änderungsrate während der Anforderungsanalyse und Systemdefinition muss ein flexibles aber kontrolliertes Änderungsmanagement auf verschiedenen Ebenen mit entsprechendem Risikomanagement zentraler Bestandteil des RE-Prozesses sein.

Die hier zusammengefassten Herausforderungen bzw. Anforderungen an das Requirements Engineering eingebetteter Systeme bilden gemeinsam mit der allgemeinen Aufgaben- und Problemanalyse des Requirements Engineering (REFOCUS-Dokument: *Requirements Engineering - Recherche und Problembeschreibung* [FGP04]) die Grundlage für die Definition der RE-Methodik REFOCUS, aber auch für die Betrachtung und Bewertung weiterer RE-Ansätze eingebetteter Systeme.

Literatur

- [BD+04] H. Bacher, J. Dörr, E. Geisberger, S. Harbhajan, F. Houdek, T. von der Maßen, J. McGregor, K. Müller, B. Paech, H. Wußmann: Einsatz von Features im Softwareentwicklungsprozess. Abschlussbericht des Arbeitskreises „Features“ im Rahmen der GI Fachgruppe 2.1.6. Requirements Engineering, in Bearbeitung, Erscheinung Herbst 2004.
- [BN+00] L. Bass, L. Northrop et al: Fourth Product Line Practice Workshop Report- Software Engineering Institute, Carnegie Mellon University, Pittsburgh, 2000
- [BH+03] A. Birk, G. Heller, I. John, S. Joos, K. Müller, K. Schmid, T. von der Maßen: Report of the GI Work Group "Requirements Engineering for Product Lines". 2003.
- [Bro97] M. Broy: Requirements Engineering for Embedded Systems. In: FemSys97, München, April 1997.
- [Dei98] B. Deifel: Theoretische und praktische Ansätze im Requirements Engineering für Standardsoftware und Anlagenbau. FORSOFT, 1998.
- [Fie99] J. Fierz: The CIP Method: Component- and Model-Based Construction of Embedded Systems. In: Proceedings of ESEC/FSE'99: 7th European Software Engineering Conference, Toulouse, France, Lecture Notes in Computer Science Software Engineering 1999.
- [HP02] F. Houdek und B. Paech: Das Türsteuergerät - Eine Beispielspezifikation. IESE-Report 002.02/D, Januar 2002.
- [GS+04] E. Geisberger, M. Saynisch, R. Schmidt, H. Wußmann: Integriertes Projektmanagement und Requirements Engineering für die Entwicklung von eingebetteten Systemen. Projektmanagement Forum 2004.
- [GW03] E. Geisberger, H. Wußmann: Requirements Engineering eingebetteter Systeme. GI-Software-technik-Trends, Bd. 23, Heft 1, 2003.
- [Lev95] N. Leveson: Safeware: System Safety and Computers. Addison Wesley, 1995.
- [Maa94] S. Maaß: Transparenz, eine zentrale software-ergonomische Forderung. Report Nr. FBI-HH-B-170/94, Hamburg, 1994.
- [ProMiS02] Pflichtenheft BMBF-Verbundprojekt „Projektmanagement für Interdisziplinäre Systementwicklung“, 2002.
- [FGP04] A. Fleischmann, E. Geisberger, M. Pister: REFOCUS – Requirements Engineering – Recherche und Problembeschreibung. (Unveröffentlichter) Projektbericht REFOCUS, 2004.
- [VA EPS 02] BMBF-Untersuchungsprojekt: VA „Entwicklung, Produktion und Service von Software für eingebettete Systeme“, 2002.

- [VA KPE 00] BMBF-Untersuchungsprojekt: VA „Kooperatives Produktengineering“, Abschlussbericht, 2000.
- [WM85] P.T. Ward, S. J. Mellor: Structured Development for Real-Time Systems. Yourdon Press, Prentice Hall, 1985.
- [Wei03] B. Weisshaupt: Intuitive Prozessführung. SE / STZ März 2003.