

Feature-basierte Modellierung und Analyse von Variabilität in Produktlinienanforderungen

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften
der Rheinisch-Westfälischen Technischen Hochschule Aachen
zur Erlangung des akademischen Grades eines Doktors
der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Dipl.-Inform. Thomas von der Maßen
aus Aachen

Berichter:

Universitätsprofessor Dr. Horst Lichter

Universitätsprofessor Dr. Martin Glinz

Tag der mündlichen Prüfung: 26.01.2007

Berichte aus der Softwaretechnik

Thomas von der Maßen

**Feature-basierte Modellierung und Analyse von
Variabilität in Produktlinienanforderungen**

Shaker Verlag
Aachen 2007

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Zugl.: Aachen, Techn. Hochsch., Diss., 2007

Copyright Shaker Verlag 2007

Alle Rechte, auch das des auszugsweisen Nachdruckes, der auszugsweisen oder vollständigen Wiedergabe, der Speicherung in Datenverarbeitungsanlagen und der Übersetzung, vorbehalten.

Printed in Germany.

ISBN 978-3-8322-6027-9

ISSN 1433-9986

Shaker Verlag GmbH • Postfach 101818 • 52018 Aachen

Telefon: 02407 / 95 96 - 0 • Telefax: 02407 / 95 96 - 9

Internet: www.shaker.de • E-Mail: info@shaker.de

Zusammenfassung

Da der Markt fordert, dass Software-Produkte nach immer kürzeren Release-Zyklen zur Verfügung stehen müssen, wird ein hohes Maß an Effizienz und Flexibilität in der Entwicklung benötigt. Software-Produktlinien sind ein Entwicklungsparadigma, das diese Anforderungen erfüllen soll. Der grundlegende Gedanke ist, mehrere Software-Produkte aus möglichst vielen gemeinsamen Teilen zu entwickeln, damit der Entwicklungsaufwand für diese Teile nur einmal geleistet werden muss. Gleichzeitig besitzt jedes Produkt spezifische Teile, die für das jeweilige Produkt auszeichnend sind. Die Vorteile einer Software-Produktlinienentwicklung reichen von der Reduzierung der Entwicklungs- und Wartungskosten über eine Verkürzung der Release-Zyklen bei gleichzeitiger Verbesserung der Software-Qualität, bis hin zu einer erhöhten Kundenzufriedenheit in Bezug auf die ausgelieferten Produkte.

Die Entwicklung einer Software-Produktlinie stellt hohe Ansprüche an alle Aktivitäten des Entwicklungsprozesses, insbesondere auch an das Requirements Engineering. Als besondere Herausforderung gilt die Berücksichtigung von Variabilität in Anforderungen. Obwohl Variabilitätskonzepte in keiner Notation für die Modellierung von Anforderungen berücksichtigt werden, haben sich eine Reihe von Arbeiten damit beschäftigt, anerkannte Notationen um die benötigten Konzepte zu erweitern.

Obwohl es eine Reihe von Variabilitätsmodellierungsansätzen gibt, die größtenteils auf der Feature-Modellierung beruhen, fehlte bislang eine systematische Integration der Variabilitätsmodellierung im Kontext des Requirements Engineering. Insbesondere wurde bisher nicht berücksichtigt, welche Kriterien herangezogen werden müssen, um die zentralen Qualitäten Adäquatheit und Widerspruchsfreiheit von Anforderungsmodellen bewerten zu können, in denen Variabilität modelliert wurde.

Im Rahmen dieser Arbeit wird zunächst ein Metamodell für die Variabilitätsmodellierung präsentiert, welches die benötigten Konzepte und ihre Beziehungen verdeutlicht. Im Anschluss wird dargestellt, wie die Modellierung von Variabilität die Qualitätsmerkmale Adäquatheit und Widerspruchsfreiheit von Feature-Modellen beeinflussen kann. Es wird eine Kategorisierung von Defiziten vorgestellt, die sich aus dem Zusammenspiel von Variabilität und Abhängigkeiten zwischen variablen Anforderungen ergeben und ein Vorgehen erläutert, wie ein defizitbehaftetes Feature-Modell in eine so genannte normalisierte Form überführt werden kann. In Bezug auf die Adäquatheit wird der Variationsgrad eingeführt, der als Maß für die Anzahl von Produktspezifikationen bzw. Produkt-Feature-Modellen dient, die auf Basis eines so genannten Plattform-Feature-Modells abgeleitet werden können.

Abschließend wird eine prototypische Werkzeugunterstützung namens *RequiLine* skizziert, welche die Feature-Modellierung unterstützt und die Qualität von Feature-Modellen in Bezug auf ihre Adäquatheit und Widerspruchsfreiheit hin analysiert. *RequiLine* dient dazu, die in dieser Arbeit eingeführten Konzepte zu validieren.

Abschließend werden die eingeführten Konzepte bewertet und *RequiLine* anhand einer industriellen Fallstudie evaluiert.

Abstract

The market demands that software products should be available in short release cycles. This requires a very efficient and flexible software development process. Software product lines are a development paradigm that should address these requirements. The idea is to develop software products from a common set of core assets in a prescribed way. Therefore the development effort for these common assets must only be afforded once. At the same time, every product of the product line has specific parts, which distinguishes the product from others. The benefits that can be achieved reach from shorter release cycles, improved quality and broader product palettes, over reduced development and maintenance costs to standardization procedures in usage and distribution.

The advantages of performing a product line development are obvious, but will be purchased with high initial establishing costs and a serious and demanding development process. A particular challenge is the consideration of variability in requirements. Though no standard notation like UML for example provides support for variability modeling, some work has been done to extend existing notations with the required concepts.

Though there are some approaches of modeling variability - most of them are based on feature modeling - a systematic integration of variability modeling in the requirements engineering for product lines is missing so far. Especially it has been neglected, which criteria must be considered to determine the central qualities appropriateness and consistency of requirements models which contain variability.

Within the scope of this work, first of all a variability metamodel is presented which introduces the required concepts and their relationships. Furthermore it is analyzed, how variability influences the quality criteria appropriateness and consistency of feature models. A catalogue of deficiencies is presented, which arise from the interplay of variability concepts and dependencies between variable requirements. Additionally it is discussed, how a deficiency afflicted feature model can be transformed into a normalized feature model that does not show any of the identified deficiencies. With respect to the appropriateness of a feature model, the variation degree is introduced that reveals the number of product specifications, respectively product feature models, that can be derived from a so called platform feature model.

In addition, a prototype of a requirements management tool called *RequiLine* is presented that supports feature modeling and that allows to analyze the quality of feature models with respect to their appropriateness and their consistency. The aim of *RequiLine* is to validate the afore introduced variability concepts.

Finally, the introduced concepts and *RequiLine* are evaluated on basis of an industrial case study.

Einer meiner Freunde, ein Künstler, äußert immer mal wieder eine Ansicht, der ich mich beim besten Willen nicht anschließen kann. Er hält mir eine Blume hin und sagt: „Schau, wie schön sie ist“, und ich stimme zu. Doch dann fährt er fort: „Ich als Künstler, kann die Schönheit einer Blume sehen. Aber du als Wissenschaftler nimmst sie auseinander und zerstörst damit ihren ganzen Reiz.“ Das erscheint mir dumm. Erstens ist die Schönheit, die er sieht, allen zugänglich - und mir, will ich meinen, auch. Und selbst wenn mein ästhetisches Empfinden sich mit dem seinen nicht zu messen vermag, kann ich doch allemal die Schönheit einer Blume würdigen. Gleichzeitig aber sehe ich mehr in der Blume als er. Ich kann mir die Zellen in ihrem Inneren vorstellen, die eine eigene Schönheit besitzen. Denn es gibt nicht nur eine in Zentimetern messbare Schönheit - Schönheit ist schon in weit kleinerem Maßstab vorhanden.

Richard P. Feynman

Danksagung

Noch während ich zur Schule ging, las ich das Buch „Kümmert Sie, was andere Leute denken?“ von Richard P. Feynman, dem „brillantesten, respektlosesten und einflussreichsten theoretischen Physiker der Nachkriegszeit“ [New York Times]. Aus diesem Buch stammt auch der vorhergehende Prolog. Obwohl dieses Buch „Abenteuer eines neugierigen Physikers“ beschreibt und zeigt, dass Feynman auf amüsante Weise Geschichten erzählen konnte, so zeigt es auch, dass er wunderbar Physik zu erklären vermochte. Insbesondere diesem Buch habe ich es zu verdanken, dass mein Interesse für die Wissenschaft geweckt wurde. Obwohl ich mich für eine andere wissenschaftliche Disziplin als Feynman entschieden habe, nämlich für die Informatik, so glaube ich, dass ich mit ihm, so wie mit jedem anderen Wissenschaftler, zwei Dinge gemeinsam habe: Neben dem Drang, die Welt in der wir leben etwas besser verstehen zu wollen, war und ist es für mich immer wichtig, die Menschen die mich umgeben, an diesem Interesse und dem Wissen ebenfalls teilhaben zu lassen.

Die vorliegende Arbeit beschreibt, wie Feature-Modelle für die Modellierung und Analyse von Variabilität in Produktlinienanforderungen genutzt werden können. Auf dem noch recht jungen Forschungsgebiet der Software-Produktlinien ist in den letzten Jahren viel hervorragende und nützliche Arbeit geleistet worden. Neben dem deskriptiven Charakter dieser Arbeit, der einen Teil dieser Arbeiten konsolidiert, Mängel beseitigt und Lücken schließt und somit beschreibt, wie die Welt heute schon ist, hoffe ich darüber hinaus auch ein paar neue Denkanstöße geben zu können, die ein wenig beschreiben, wie man, beziehungsweise ich, die Welt gerne hätte. Ich hoffe mit dieser Arbeit einen Beitrag dazu leisten zu können.

Obwohl ich den Ruhm dieser Arbeit hoffentlich durch die Erlangung des Doktorgrads ernte, so möchte ich doch an dieser Stelle darauf hinweisen, dass diese Arbeit nicht das Werk eines einzelnen ist, sondern eine Vielzahl von Personen an diesem beteiligt waren. Diesen möchte ich nun danken:

Zunächst möchte ich Prof. Dr. Lichter danken, der mir die Möglichkeit zur Promotion und zum Erstellen dieser Dissertation gegeben hat. Weiterhin bedanke ich mich für seine wertvolle Unterstützung, für neue Denkanstöße und für die jederzeit konstruktive Kritik - während meiner Zeit als wissenschaftlicher Mitarbeiter als auch darüber hinaus. Ebenfalls möchte ich mich bei Prof. Dr. Martin Glinz, der sich bereit erklärt hat, das Zweitgutachten für diese Arbeit zu übernehmen, für die unkomplizierte Zusammenarbeit und die ebenfalls wertvolle Kritik bedanken, die es mir ermöglichte, die Arbeit weiter zu verbessern.

Ein besonderer Dank gilt auch meinen ehemaligen Kollegen am Lehr- und Forschungsgebiet für Softwarekonstruktion der RWTH Aachen, Thomas Weiler, Alexander Nyßen und Holger Schackmann. Die unvergessliche konstruktive, unkomplizierte und zuweilen auch humorvolle Zusammenarbeit haben einiges zum Gelingen dieser Arbeit beigetragen. Ebenfalls möchte ich auch den Diplomanden danken, für die ich die Betreuung ihrer Diplomarbeit übernehmen durfte. Ihre Arbeiten haben wesentlich die Ergebnisse dieser Arbeit beeinflusst und insbesondere die Entwicklung des Werkzeugs RequiLine mit getragen. Mein Dank also an Heiko Stein, Michael Matzutt, Holger Schackmann, Frank Thunig, Paul Herr und Nan Mungard. Ebenso gebührt auch Dank den zahlreichen Studenten, mit denen ich Themen der Informatik diskutieren durfte. Danken möchte ich auch unserer Sekretärin Frau Bärbel Kronewetter, die mir viele administrative Tätigkeiten abgenommen hat und immer unterstützend zur Seite stand.

Weiterhin danke ich den Mitgliedern der GI-Arbeitskreise, deren Mitglied ich war und die in zahl-

reichen Diskussionen mein Verständnis für die Thematik dieser Arbeit erweiterten und mir auch Einblick in die industriellen Herausforderungen gewährt haben. Mein Dank gilt Prof. Dr. Klaus Schmid, Isabel John, Gerald Heller, Dr. Klaus Müller, Dr. Andreas Birk, Heinrich Dreier, Dirk Janzen, Andreas Fleischmann, Dr. Ramin Tavakoli-Kolagari, Dr. Danilo Beuche, Andreas Wolfram, Heidi Galle, Prof. Dr. Barbara Paech und John McGregor.

Neben der notwendigen fachlichen Unterstützung bedarf eine solche Arbeit natürlich auch Unterstützung persönlicher und emotionaler Art. Deshalb möchte ich mich ganz herzlich bei meiner Frau Ruth und meiner Tochter Sophie für die unbezahlbare Unterstützung und den Ansporn in den letzten Jahren bedanken. Sie mussten oftmals Ehemann und Papa für eine ganze Weile entbehren. Zuletzt gebührt mein Dank meinen Eltern, die mir ein Studium ermöglicht haben und mich Beharrlichkeit und niemals Aufgeben gelehrt haben.

Inhalt

1 Einleitung	1
1.1 Software-Produktlinien	1
1.1.1 Grundlagen	2
1.1.2 Entwicklungsprozess	3
1.2 Beitrag dieser Arbeit	5
1.3 Aufbau der Arbeit	8
2 Requirements Engineering im Kontext von Produktlinien	11
2.1 Requirements Engineering für Einzelprodukte	11
2.1.1 Grundlagen des Requirements Engineering	11
2.1.2 Die Anforderungsspezifikation	12
2.1.3 Notationen für Anforderungen	13
2.2 Requirements Engineering für Produktlinien	15
2.2.1 Die Plattformanalyse	16
2.2.2 Die Produktanalyse	17
2.2.3 Herausforderungen	18
2.3 Zusammenfassung	18
3 Variabilität in Software-Produktlinien	21
3.1 Variabilität und Produktcharakteristika	21
3.1.1 Typen von Variabilität	22
3.1.2 Variationspunkt	23
3.1.3 Bindung	24
3.1.4 Produktlinienabhängigkeit	26
3.2 Klassifikation von Produktcharakteristika	27
3.2.1 Kerncharakteristika	28
3.2.2 Wahlcharakteristika	28
3.2.3 Variabilität und Produktlinien	28
3.2.4 Ableitungen	29
3.3 Zweck und Auswirkungen der Variabilität	30
3.4 Zusammenfassung	31
4 Modellierung von Variabilität	33
4.1 Ziele der Modellierung von Variabilität	33
4.1.1 Primärziele	33
4.1.2 Sekundärziele	34
4.2 Dokumentation von Anforderungen	35
4.3 Natürliche Sprache	36
4.3.1 Variabilität in natürlichsprachlichen Anforderungen	37
4.3.2 Bewertung	38
4.4 Anwendungsfälle	39
4.4.1 Anwendungsfallbeschreibungen	40

4.4.2	Anwendungsfalldiagramme	42
4.4.3	Variabilität in Anwendungsfallbeschreibungen und -diagrammen .	43
4.4.4	Erweiterung von Anwendungsfällen um Variabilitätskonzepte . . .	45
4.4.5	Anwendungsfälle und Abhängigkeiten	52
4.4.6	Bewertung	53
4.5	Feature-Modellierung	54
4.5.1	Der Featurebegriff in der Literatur	55
4.5.2	Einsatzgebiete	56
4.5.3	Methoden	56
4.5.4	Feature-Modelle	59
4.5.5	Bewertung	60
4.6	Zusammenfassung	61

5 Feature-Modellierung 65

5.1	Feature	65
5.1.1	Definition	65
5.1.2	Abgrenzung Feature und Anforderung	66
5.1.3	Eine Klassifikation für Features	67
5.2	Feature-Attribute	67
5.2.1	Attribut-Typen	68
5.2.2	Einsatzzweck für Attribute	69
5.3	Das Feature-Modell	70
5.3.1	Domänenbeziehungen	70
5.3.2	Abhängigkeitsbeziehungen	77
5.3.3	Feature-Mengen	78
5.3.4	Ein Feature-Metamodell	78
5.4	Repräsentationen für ein Feature-Modell	78
5.4.1	Das Feature-Diagramm	78
5.4.2	Textuelle Repräsentation	82
5.5	Feature-Modelle für die Produktlinien-Entwicklung	84
5.5.1	Das Plattform-Feature-Modell	85
5.5.2	Produkt-Feature-Modell	85
5.6	Zusammenfassung	86

6 Qualität von Plattform-Feature-Modellen 89

6.1	Ein Qualitätsmodell für PLFMe	89
6.1.1	Inhaltliche Qualitäten	89
6.1.2	Gebrauchsqualitäten	90
6.2	Defizite in PLFMen	91
6.2.1	Redundanz	92
6.2.2	Anomalie	94
6.2.3	Inkonsistenz	97
6.2.4	Normalisierung	99
6.2.5	Vermeiden von Defiziten	101
6.3	Variationsgrad	102
6.3.1	Definition	103
6.3.2	Variationsgrad und Domänenbeziehungen	104

6.3.3	Variationsgrad und Feature-Attribute	107
6.3.4	Variationsgrad und Abhängigkeiten	108
6.4	Zusammenfassung	114
7	Ableitung von Produkt-Feature-Modellen	115
7.1	Ableitung und Konfigurierung	115
7.1.1	PFM-Ableitung	115
7.1.2	Abgrenzung zum Konfigurationsmanagement	117
7.2	Qualität von PFMen	118
7.2.1	Vollständigkeit von PFMen	118
7.2.2	Widerspruchsfreiheit von PFMen	118
7.3	Ableitungsmethoden	119
7.3.1	Bisherige Arbeiten	120
7.3.2	Kategorien von PFM-Ableitungsmethoden	122
7.3.3	Ein generisches Modell für die PFM-Ableitung	124
7.4	Explorative PFM-Ableitung	125
7.5	Regelbasierte Ableitung	126
7.5.1	Ableitung unter Verwendung einer Grammatik	126
7.5.2	Ableitung durch Baum-Traversierung	127
7.6	Feature-Modell Spezialisierung	128
7.7	Klassifikation der PFM-Ableitungsmethoden	129
7.8	Zusammenfassung	129
8	Werkzeugunterstützung	131
8.1	Anforderungen	131
8.1.1	Anforderungen unabhängig vom Anforderungsmodell	132
8.1.2	Anforderungen an ein Feature-Management Werkzeug	134
8.2	Existierende Werkzeuge	135
8.2.1	CaptainFeature	136
8.2.2	ASADAL2	138
8.2.3	pure::variants	140
8.2.4	Vergleich der Werkzeuge	140
8.3	RequiLine	142
8.3.1	Motivation	142
8.3.2	Features	143
8.3.3	Architektur	146
8.3.4	Modellierung mit RequiLine	150
8.3.5	PFM-Ableitung	155
8.3.6	Analyse mit RequiLine	156
8.4	Zusammenfassung	159
9	Bewertung	161
9.1	Bewertung eingeführter Konzepte	161
9.1.1	Bewertung des Variabilitätsmetamodells	161
9.1.2	Bewertung des Feature-Metamodells	163
9.1.3	Bewertung des Qualitätsmodells	165

9.2	Bewertung RequiLine	168
9.2.1	Bewertung der Funktionalität	168
9.2.2	Bewertung der Qualität	170
9.2.3	Evaluierung	171
9.3	Zusammenfassung	172
10	Zusammenfassung	173
10.1	Resümee	173
10.2	Ausblick	175
10.2.1	Konzeptionelle Erweiterungen	175
10.2.2	Erweiterungen von RequiLine	176
A	Übersicht Notationselemente für Feature-Diagramme	179
B	Fallbeispiel CASE-Werkzeug	181
C	Ableitungsalgorithmus	185
D	Document Type Definitions	189
E	Weitere Materialien	197
	Glossar	199
	Literatur	203