

Praktische Ansätze zur Etablierung einer Software-Produktlinie in eine bestehende Mehr-Produkt-Entwicklung

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der
RWTH Aachen University zur Erlangung des akademischen Grades eines
Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Dipl.-Wirt.-Inf. Holger Rendel
aus Braunschweig

Berichter: Universitätsprofessor Dr. rer. nat. Bernhard Rumpe
Professorin Dr.-Ing. Ina Schaefer

Tag der mündlichen Prüfung: 22.03.2016

Aachener Informatik-Berichte, Software Engineering

herausgegeben von
Prof. Dr. rer. nat. Bernhard Rumpe
Software Engineering
RWTHAachen University

Band 23

Holger Rendel

**Praktische Ansätze zur Etablierung
einer Software-Produktlinie in eine
bestehende Mehr-Produkt-Entwicklung**

Shaker Verlag
Aachen 2016

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.: D 82 (Diss. RWTH Aachen University, 2016)

Copyright Shaker Verlag 2016

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

Printed in Germany.

ISBN 978-3-8440-4584-0

ISSN 1869-9170

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

Kurzfassung

Umfang und Komplexität von Softwaresystemen steigen kontinuierlich weiter an. Gründe dafür sind zum einen die steigende Zahl funktionaler Anforderungen an die Systeme und zum anderen, dass nicht nur ein Produkt entwickelt wird, sondern eine Vielzahl verwandter Produkte. Die Komplexität steigt durch Funktionserweiterungen und die Einbindung von Variabilität. Optimierung von bestehenden Funktionalitäten und Fehlerbeseitigung erhöht den Variantenreichtum und damit die Komplexität zusätzlich. Zudem ist ein immer höherer Aufwand notwendig, um ein System über den gesamten Lebenszyklus hinweg zu begleiten. Dadurch, dass sich aus einem System viele Produktvarianten ergeben, ist für jedes Produkt ein gewisser Aufwand für die Entwicklung, den Test und die Dokumentation des Produkts notwendig.

Die negativen Effekte der steigenden Komplexität und des steigenden Aufwands sollen durch eine möglichst hohe Wiederverwendung von Entwicklungsartefakten verringert werden. Dies führt zur Nutzung von programmatischen Möglichkeiten wie Vererbung und der Entwicklung von Bibliotheken und Komponenten, die in mehreren Produkten Anwendung finden. Ein Produkt enthält in der Regel mehrere Bibliotheken bzw. Komponenten, die jeweils einen definierten Funktionsumfang bereitstellen. Bei einer Software-Produktlinie wird diese Vorgehensweise ebenfalls angewendet, verbunden mit der Möglichkeit für den Kunden, das entstehende System nach seinen Wünschen zu konfigurieren.

Wie eine Entwicklung mittels Produktlinien-Methodiken ausgestaltet sein kann, ist mittlerweile in einigen Publikationen beschrieben. In der Praxis ist es jedoch oft so, dass schon mehrere Einzelproduktentwicklungen vorliegen, die auf eine produktlinienbasierte Entwicklung umgestellt werden sollen. Die theoretischen Ansätze aus der Literatur für diesen Schritt lassen sich nicht direkt in die Praxis übertragen, sondern müssen den projektspezifischen Gegebenheiten angepasst werden. Das Ergebnis ist eine praxistaugliche Adaption der Methodiken, die für den aktuellen Kontext relevant sind. Es ist die projektspezifische Kombination aus mehreren theoretischen Vorgehen notwendig, um ein optimales Ergebnis zu erhalten.

In dieser Arbeit werden Ansätze zur Etablierung einer Software-Produktlinie im Rahmen einer Steuergeräte-Entwicklung beschrieben. Dazu werden die allgemeinen und individuellen Einflussfaktoren skizziert und bewertet. Basierend darauf wird eine Vorgehensweise definiert, die zur Einführung von Produktlinien-Methodiken in ein existierendes Projekt

beiträgt. Dabei werden zunächst im Entwicklungsprozess mehr Modelle verwendet, die den Automatisierungsgrad erhöhen. Auf Basis dieser Modelle wird Variabilität erfasst, im Entwicklungsprozess erstmals effizient modelliert und mittels Werkzeugen in weiteren Artefakten des Entwicklungsprozesses strukturiert berücksichtigt. Das beschriebene Vorgehen wurde in der Praxis angewandt und mit messbarem Erfolg umgesetzt. Somit wurde eine Optimierung des Entwicklungsprozesses erreicht, damit dieser zukünftig weiter in eine Produktlinie überführt werden kann.

Abstract

The size of software systems as well as their complexity is continuously growing. This is caused by additional functional requirements and the fact that not only one system is developed but a set of related systems. Complexity increases through improvement of functionality and the use of variability. Optimizations and error corrections increase also the number of variants. Furthermore a higher effort is required to maintain a system throughout its complete life cycle. Due to the number of product variants resulting from one system an appropriate effort is required to develop, test and document each product.

The negative effects of the complexity should be compensated with an efficient reuse of development artifacts. Using inheritance and libraries as well as components, which can be used in more than one product is a common strategy when implementing reuse. A product contains several libraries or components, which realize specific functionality. Software product lines also use this mechanism but with the enhancement for the customer to configure the resulting system for his needs.

The development with software product lines is described in several publications. In practice often multiple single products should be evolved into a product line based development. The theoretical approaches cannot be easily transferred into practice but must be adjusted to a project-specific situation. Thus, an optimal result can be gained by combining different theoretical approaches to a practical adaption of product line methodologies.

This thesis describes approaches to establish a software product line for the development of an electronic control unit. Therefore, common and individual factors are examined and evaluated to define a strategy to implement product line methodologies into an existing project. The key is to increase the usage of models to raise the level of automation. Next, variability is defined in these models and hence, efficiently described for the first time. Tools use the variability in further artifacts in the development process. This strategy was implemented in practice and empirically evaluated resulting in an improved development process, which can be further transformed into a product line.

Danksagung

Ganz besonders möchte ich mich an dieser Stelle bei Prof. Dr. Bernhard Rumpel bedanken, der meine Arbeit betreut hat. Zahlreiche hilfreiche Ratschläge und Diskussionen haben mich bei der Anfertigung der Dissertation sehr unterstützt. Insbesondere möchte ich mich bei ihm für die Möglichkeit bedanken, die Inhalte der Arbeit bei Volkswagen anzuwenden, was den Grundstein für meine weitere berufliche Karriere gelegt hat.

Des Weiteren möchte ich mich bei Prof. Dr. Ina Schaefer bedanken, die sich als weitere Gutachterin für die Dissertation zu Verfügung gestellt hat. Die Zusammenarbeit an zahlreichen Veröffentlichungen haben sehr zu dem positiven Ausgang der Promotion beigetragen. Vielen Dank geht auch an Prof. Dr. Joost-Pieter Katoen als Prüfer und Leiter der Prüfungskommission und an Prof. Dr. Stefan Decker als Prüfer.

Meine Arbeit am Lehrstuhl und diese Dissertation wurde ebenfalls durch eine Reihe von Mitarbeitern am Lehrstuhl unterstützt. Vor allem möchte ich hier Holger Krahn, Christian Berger und Tim Gülke erwähnen, mit denen ich in der Zeit sehr intensiv zusammengearbeitet habe. Auch vielen weiteren aktuellen und ehemaligen Mitarbeitern möchte ich hier danken (in alphabetischer Reihenfolge): Marita Breuer, Timo Greifenberg, Hans Grönniger, Sylvia Gunder, Arne Haber, Christoph Herrmann, Andreas Horst, Carsten Kolassa, Thomas Kurpick, Achim Lindt, Markus Look, Cem Mengi, Antonio Navarro Pérez, Claas Pinkernell, Dirk Reiß, Jan Oliver Ringert, Martin Schindler, Steven Völkel, Galina Volkova, Ingo Weisemöller, Andreas Wortmann

Ein großer Dank geht an die Entwicklung der Lenkungselektronik bei Volkswagen, wo ich ebenfalls viel Unterstützung bei der Umsetzung der Arbeit erfahren habe. Hier möchte ich insbesondere Carsten Busse, Fabian Wolf, Christian Hopp, Lars Gottwaldt und Karsten Rowold hervorheben, die seitens Volkswagen sich als wertvolle Ansprech- und Diskussionspartner bereitgestellt haben. Dank geht auch an die Doktoranden, die ich bei Volkswagen kennengelernt habe und die mich auf meinem Weg begleitet haben.

Meiner Familie und Schwiegerfamilie möchte ich danken, dass sie mich während der ganzen Zeit unterstützt und motiviert haben. Nicht zuletzt haben die Promotionsvorhaben des einen oder anderen Bruders dazu beigetragen, dass ich meine Promotion erfolgreich abschließen wollte. Die Unterstützung in der finalen Phase und am Tag der Prüfung werde ich euch nie vergessen. Vielen Dank geht auch an alle weiteren Verwandte, Freunde und Bekannte, die zum Gelingen der Promotion beigetragen haben.

Der größte Dank geht an meine Frau Franziska, die mir die notwendige Zeit für die Dissertation gegeben hat, und die mich das eine oder andere Mal wieder aufgebaut hat, wenn es mal nicht nicht so lief. Ihre Sichtweise auf die Dinge hat mir geholfen mich auf das Wesentliche zu konzentrieren und mich nicht in Details zu verlieren. Mit ihr als verlässliche Partnerin an meiner Seite konnte ich mich intensiv mit meiner Arbeit beschäftigen. Zuletzt geht mein Dank auch an meinen Sohn Moritz, dessen Lächeln und Art mich jederzeit von den Sorgen des Alltags abgelenkt haben.

Disclaimer

Die Ergebnisse, Meinungen und Schlüsse dieser Dissertation sind nicht
notwendigerweise die der Volkswagen AG.

The results, opinions and conclusions expressed in this thesis are not necessarily those
of Volkswagen AG.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation und Kontext	1
1.2	Wissenschaftliche Fragestellung	3
1.3	Wesentliche Ergebnisse	4
1.4	Aufbau der Arbeit	5
2	Einführung in Software-Produktlinien	7
2.1	Grundlegende Begriffe	7
2.2	Entwicklungsprozess von Software-Produktlinien	11
2.3	Variabilitätsmodelle	15
2.3.1	Problem Space	17
2.3.2	Solution Space	20
2.4	Gestaltung einer Entwicklung als Software-Produktlinie	23
2.5	Beispiele für den erfolgreichen Einsatz von Software-Produktlinien	26
2.6	Zusammenfassung	29
3	Methodik zur Einführung einer Software-Produktlinienentwicklung	31
3.1	Praktische Erprobung im Projektkontext	32
3.1.1	Produkte	33
3.1.2	Entwicklungsprozess	35
3.2	Beschreibung der Maßnahmen	36
3.3	Abhängigkeiten der Maßnahmen	40
3.4	Zusammenfassung	41
4	Analyse der bestehenden Software	43
4.1	Datenmodell zur Analyse	44
4.1.1	System	45
4.1.2	Ähnlichkeitsfunktion	46
4.2	Kennzahlen zur Analyse	47
4.3	Exemplarische Durchführung der Analyse	50
4.4	Anwendung der Kennzahlen bei Volkswagen	52
4.5	Weitere Analysemöglichkeiten	54
4.6	Zusammenfassung	55

5	Bewertung der Einflussfaktoren für eine Produktlinienentwicklung	57
5.1	Quantitative Einflussfaktoren	58
5.1.1	Personen und Material	58
5.1.2	Kosten	59
5.1.3	Zeit	59
5.1.4	Projektgruppengröße	60
5.1.5	Komplexität	60
5.2	Qualitative Einflussfaktoren	61
5.2.1	Kundenbezogenheit	61
5.2.2	Lebenszyklus	62
5.2.3	Innovationskultur	62
5.2.4	Regularien	63
5.2.5	Einsatz von Modellen	64
5.2.6	Technische Infrastruktur.	65
5.3	Unternehmerische Einflussfaktoren	66
5.3.1	Unternehmensziele	67
5.3.2	Wettbewerbspositionierung	67
5.3.3	Betriebsklima	69
5.4	Variabilität als Einflussfaktor	69
5.4.1	Verteilung von Variabilität über mehrere Ebenen	70
5.4.2	Bindung von Variabilität	71
5.4.3	Auftreten und Realisierung von Variabilität	72
5.5	Zusammenfassung	75
6	Strategie bei der Implementierung von Software-Produktlinien	79
6.1	Zuordnung von Maßnahmen zu Einflussfaktoren	79
6.2	Auswahl der Maßnahmen für die Realisierung	82
6.3	Erstellung einer Übersicht über die vorhandenen Projekte	84
6.3.1	Anforderungen an den Projektraum	85
6.3.2	Realisierungsmöglichkeiten	87
6.3.3	Umsetzung mittels einer Präsentation	88
6.4	Zusammenfassung	90
7	Modellbasiertes Vorgehen	93
7.1	Überblick über die modellbasierte Softwareentwicklung	93
7.2	Methodik zur Identifikation von Modellen in einer Softwareproduktlinie	97
7.2.1	Metamodell dokumentieren	98
7.2.2	Artefakte identifizieren und Typ der Artefakte bestimmen	98
7.2.3	Modelleigenschaften der Artefakte untersuchen	99
7.2.4	Nutzbarkeit der Modellartefakte auswerten	99
7.2.5	Artefakte definieren	100
7.3	Anwendung der Methodik bei Volkswagen	101
7.3.1	Metamodell dokumentieren	101
7.3.2	Artefakte identifizieren und Typ der Artefakte bestimmen	109

7.3.3	Modelleigenschaften der Artefakte untersuchen	113
7.3.4	Nutzbarkeit der Modellartefakte auswerten	113
7.3.5	Artefakte definieren	115
7.4	Definition von Variabilität in Modellen	116
7.4.1	MontiArc für 150%-Modelle	116
7.4.2	DeltaMontiArc	117
7.4.3	Vergleich der Ansätze	120
7.4.4	Entscheidung im Produktlinien-Einführungsprojekt	125
7.5	Zusammenfassung	125
8	Erstellung von Werkzeugen	127
8.1	Methodik zur Entwicklung von Werkzeugen für den Produktlinieneinsatz .	128
8.1.1	Werkzeug definieren	128
8.1.2	Zugriff auf die Quellen definieren	132
8.1.3	Werkzeug erstellen	133
8.2	Konstruktion konkreter Werkzeuge bei Volkswagen	134
8.2.1	Werkzeug definieren	134
8.2.2	Zugriff auf die Quellen definieren	137
8.2.3	Werkzeug erstellen	144
8.2.4	Erfahrungen beim Einsatz des Werkzeuges	149
8.3	Weitere Beispiele für den Einsatz im Produktlinien-Einführungsprojekt .	150
8.3.1	Generierung von Diagnose-Daten aus DOORS (ODXGen)	150
8.3.2	Generierung von Schnittstellen-Code (EVFGen)	152
8.3.3	Generierung von Test-Infrastruktur (TestbedGen)	154
8.4	Zusammenfassung	155
9	Zusammenfassung und Ausblick	157
9.1	Zusammenfassung	157
9.2	Ausblick	158
	Abbildungsverzeichnis	161
	Tabellenverzeichnis	163
	Listingverzeichnis	165
	Literaturverzeichnis	167
A	DOORS-Offline-Adapter	189
A.1	DXL-Skript für den DOORS-Export	189
A.2	DSL für den DOORS-Import	205
B	Grammatiken	211
B.1	DSL für das DBC-Format	211

