Modellierung von Variabilität in der Geschäftsanalyse – eine industrielle Fallstudie

Thomas von der Maßen[1], Andreas Wübbeke[2]

[1]DC Application Development, arvato services,
An der Autobahn, 33310 Gütersloh
Thomas.vonderMassen@bertelsmann.de

[2]s-lab – Software Quality Lab, Universität Paderborn,
Warburger Str. 100, 33098 Paderborn
awuebbeke@s-lab.upb.de

Abstract: Der Umgang mit Variabilität ist eine zentrale Herausforderung bei der Einführung einer Software Produktlinie. Variabilität sollte dabei schon zu Beginn der Anforderungsanalyse besondere Beachtung finden. Dieser Beitrag beschreibt erste Schritte und die daraus resultierenden Herausforderungen beim Einsatz zweier bekannter Modellierungstechniken für Artefakte der Geschäftsanalyse im Rahmen einer industriellen Fallstudie.

1 Einleitung

Software Produktlinien (SPL), gemäß der Definition von [CN01], bieten für die industrielle Entwicklung von gleichartigen Softwareprodukten für eine bestimmte Domäne große Vorteile. Sie unterstützen die Produktentwicklung innerhalb einer Domäne, indem sie Maßnahmen zur Kostensenkung bzw. Beschleunigung der Entwicklung zum Beispiel durch verbesserte Wiederverwendbarkeit von Entwicklungsartefakten anbieten. Einführende Informationen zu SPL finden sich in [PBL05] und [CN01]. Im Kontext von SPL ist der Umgang mit Variabilität eine große Herausforderung.

Die in dieser Fallstudie vorgestellten Erfahrungen sind im Kontext der Einführung einer Softwareproduktlinien-Entwicklung bei arvato services entstanden. arvato services hat langjährige Erfahrung im Bereich der Dienstleistungen rund um die Domänen CRM, Fulfillment und Loyalty. DC als Geschäftseinheit von arvato services entwickelt dabei Software in enger Zusammenarbeit mit den jeweiligen Auftraggebern. In jeder Domäne entstehen individuelle Softwarelösungen, die aber gleichzeitig viele domänenübergreifende Gemeinsamkeiten besitzen und daher auch ein gemeinsames Vorgehensmodell für deren Entwicklung benötigen, um u. a. den flexiblen Einsatz von Mitarbeitern in unterschiedlichen Projekten garantieren zu können..Auf Basis der Projekte vergangener Jahre ist ein *Softwarekomponenten-Baukasten* entstanden, der eine Grundlage für neue Projekte bildet und auf der Ebene des Systementwurfs bzw. der Systemarchitektur sowie der Implementierung eingesetzt werden kann.

Dieser Baukasten enthält u.a. wiederverwendbare *Geschäftskomponenten*. Beim Einsatz einer Komponente in einem konkreten Entwicklungsprojekt wird diese dann durch Modifikation bzw. Erweiterung an die projektspezifischen Anforderungen angepasst. Aus Sicht des Systementwurfs und der Implementierung dient der Komponenten-Baukasten als Plattform im Sinne einer SPL.

Im Gegensatz dazu wird im Bereich der Geschäftsanalyse bisher dem Vorhandensein von projektübergreifenden, gemeinsamen Anforderungen und der davon abhängigen Betrachtung der individuellen Projektanteile unzureichend Aufmerksamkeit geschenkt. Somit ist die Wiederverwendung von Entwicklungsartefakten in der Geschäftsanalyse zu optimieren. Die in dieser Phase der Entwicklung entstehenden Entwicklungsartefakte, wie z. B. das Geschäftsmodell, bieten zum aktuellen Zeitpunkt keine Möglichkeit der expliziten Modellierung von Variabilität. An dieser Stelle würde ein generisches Modell (Plattform), welches für jedes Projekt abgeleitet werden kann, Vorteile hinsichtlich der Wiederverwendbarkeit und sinkende Spezifikationszeit bedeuten. Ableitung meint in diesem Kontext die Auflösung der im generischen Modell explizit modellierten Variabilität, d.h. die Bindung von variablen Geschäftsanforderungen an bestimmte Punkte des Modells. Nach der Ableitung des projektspezifischen Modells können dann weitere Geschäftsanforderungen angefügt werden, die nur im Kontext dieses Projekts relevant sind.

In diesem Beitrag wird anhand eines Praxisbeispiels diese Problematik der Behandlung von Variabilität in der Geschäftsanalyse erörtert und daraus Lösungsmöglichkeiten unter Einsatz zweier verschiedener Konzepte zur expliziten Modellierung von Variabilität skizziert. Die resultierenden Ergebnisse stellen dabei einen ersten Schritt zu einem ganzheitlichen und durchgängigen Ansatz zur Behandlung von Variabilität dar. Im folgenden Kapitel 2 wird unser Vorgehen bei der Geschäftsanalyse grundlegend dargestellt und anhand eines Beispiels die vorhandenen Unzulänglichkeiten bei der Behandlung von Variabilität und die daraus resultierenden Optimierungspotentiale dargestellt. In den Kapiteln 3 und 4 werden dann zwei etablierte Modellierungsansätze erläutert und auf ihr Potential zur Lösung der in Kapitel 2 identifizierten Unzulänglichkeiten hin untersucht. Abschließend werden die erarbeiteten Ergebnisse in Kapitel 5 zusammengefasst und ein Ausblick über diesen Beitrag hinaus gegeben.

2 Geschäftsanalyse bei DC

2.1 Konzepte und Vorgehensmodell

Der Entwicklungsprozess bei DC orientiert sich am Rational Unified Process (RUP) [Kr99]. Jedes Projekt beginnt mit einer sogenannten Akquisitionsphase. Diese dient der Aufwandsschätzung für das jeweilige Projekt, die darauf folgende eigentliche Geschäftsmodellierung (im RUP Geschäftsprozessmodellierung) dann der Ausformulierung des Geschäftsmodells. Die Akquisitionsphase unterteilt sich in die Teilbereiche Geschäftsanalyse, Systemanalyse (grob) und Bewertung. Die zentralen Artefakte der hier im Fokus stehenden Geschäftsanalyse sind die *Geschäftsskizze* und das *Geschäftsmodell,* bestehend aus allen *Geschäftsanwendungsfällen* (englisch Business Use Case, abgekürzt BUC) und deren *Geschäftsregeln*.

Die Geschäftsanwendungsfälle bilden in diesem Kontext die zugrundeliegenden Geschäftsprozesse ab. Die Geschäftsskizze dient als Grundlage für die Ausformulierung eines Geschäftsmodells und wird immer gemeinsam mit dem Auftraggeber erstellt. Dabei werden die BUCs nur namentlich beschrieben, eine Detaillierung findet dann im Geschäftsmodell statt. Dieses vollständige, konsistente und hinreichend spezifizierte Geschäftsmodell wird dann später in der eigentlichen Projektphase im Abschnitt Geschäftsmodellierung erstellt.

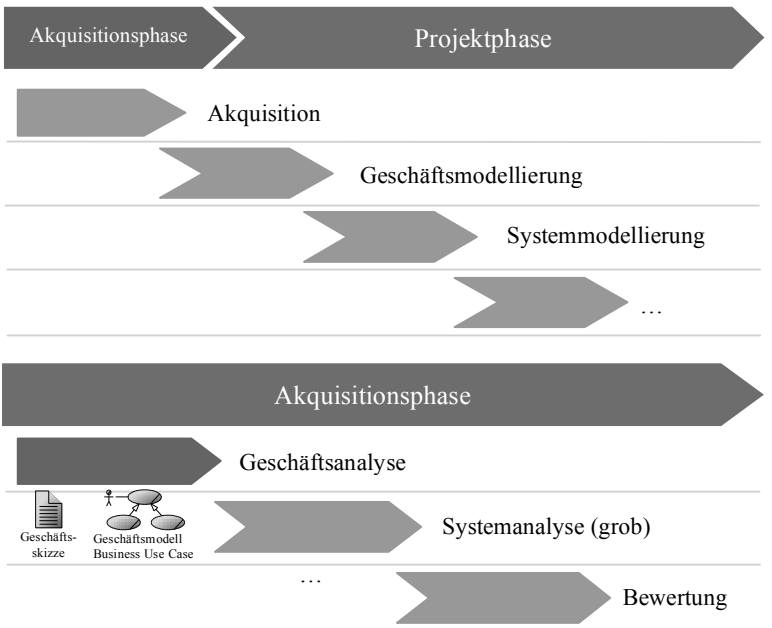

Abbildung 1: Vorgehensmodell

Der Prozess der Geschäftsanalyse sieht also zunächst das Erstellen einer Geschäftsskizze vor. Diese Skizze bietet thematisch gruppiert eine grobe Beschreibung der Geschäftsanforderungen (meist in Form einer Überschrift) an die zu realisierende Lösung. Die zu erstellende Geschäftsskizze wird aus einer Vorlage abgeleitet, welches bereits thematische Gruppen und ihnen zugeordnete Geschäftsanforderungen enthält. Diese Vorlage stellt die gesammelten Erfahrungen aus Geschäftsskizzen vergangener Projekte der gleichen Subdomäne dar. Bei der Ableitung einer Geschäftsskizze aus der Vorlage der Domäne wird folgendermaßen vorgegangen:

- Alle relevanten thematischen Gruppen und die jeweils in jeder Gruppe relevanten Geschäftsanforderungen müssen für das vorliegende Entwicklungsprojekt identifiziert und ausgewählt werden.

- Nach der Auswahl werden die noch fehlenden, projektspezifischen Gruppen bzw. Geschäftsanforderungen eingefügt.

Der nächste Schritt ist dann die Verfeinerung der grobgranularen Geschäftsskizze zu einem Geschäftsmodell. Dieses besteht aus Anwendungsfällen [OM04] (in diesem Fall den bereits erwähnten Geschäftsanwendungsfällen), die das zentrale und bei DC festgelegte Modellierungskonzept in allen Entwicklungsstufen darstellen. Es handelt sich dabei um eine natürlich sprachliche, textuelle Spezifikation eines Anwendungsfalls (vgl. [Ja92] und [Co01]). Diese Anwendungsfälle beschreiben Anwendungsszenarien durch Darstellung des Standardablaufs unter Berücksichtigung von Alternativen und Ausnahmen. Die Erstellung des Geschäftsmodells geschieht wiederum gemeinsam mit dem Auftraggeber des Entwicklungsprojekts, um die relevanten Geschäftsprozesse richtig auszugestalten.

Die in der Geschäftsmodellierung der Projektphase fertig gestellten BUCs des Geschäftsmodells werden im Rahmen der Systemmodellierung analysiert. Auf dieser Basis werden dann Systemanwendungsfälle (englisch System Use Cases, abgekürzt SUC) abgeleitet, die die Grundlage für den Systementwurf und den Systemtest bilden.

2.2 Beispiel einer Geschäftsskizze

Abbildung 2 zeigt anhand eines Beispiels, wie die einzelnen Entwicklungsartefakte zusammenhängen. *Antragsmanagement* und *Interessentenbetreuung* sind hierbei thematische Gruppen und die darunter angeordneten Punkte die Grobbeschreibung der jeweiligen Geschäftsanforderungen (*Automatisierte Antragsdatenverarbeitung*, *Manuelle Antragsdatenverarbeitung*, …). Unterhalb einer jeden Geschäftsanforderung werden weiterhin geforderte, übergreifende Geschäftsanforderungen aufgeführt. Übergreifend meint hier, dass die Anforderungen in mehreren anderen Geschäftsanforderungen benötigt werden.

Zur Geschäftsanforderung *Automatisierte Antragsdatenverarbeitung* gehören z. B. die Anforderungen *#Adressvalidierung* und *#Abgleich: Blacklist*. Die übergreifende Geschäftsanforderung *#Adressvalidierung* wird auch in der Anforderung *Manuelle Antragsdatenverarbeitung* verwendet.

An dieser Stelle lässt sich bereits erkennen, dass bei dieser Art der textuellen Modellierung die in den Geschäftsanforderungen geforderten übergreifenden Geschäftsanforderungen explizit in jeder Anforderung aufgelistet werden und somit redundant modelliert werden. Aus der Geschäftsskizze werden nun beispielhaft zwei BUCs für das Geschäftsmodell abgeleitet. Spezifizieren bedeutet in diesem Kontext das konstruieren von BUCs aus den existierenden Geschäftsanforderungen. Dies stellt einen kreativen Akt dar. Die BUCs in unserem Beispiel sind eine Spezifikation aus den beiden Geschäftsanforderungen *Automatisierte Antragsdatenverarbeitung* und *Manuelle Antragsdatenverarbeitung*. Diese Spezifikation muss nicht immer eine 1:1 Beziehung sein. Vielmehr sind einerseits eine Aufteilung einer Geschäftsanforderung in mehrere BUCs sowie die Konsolidierung mehrerer Geschäftsanforderung in einem BUC möglich und praktisch relevant (n:m Beziehung).

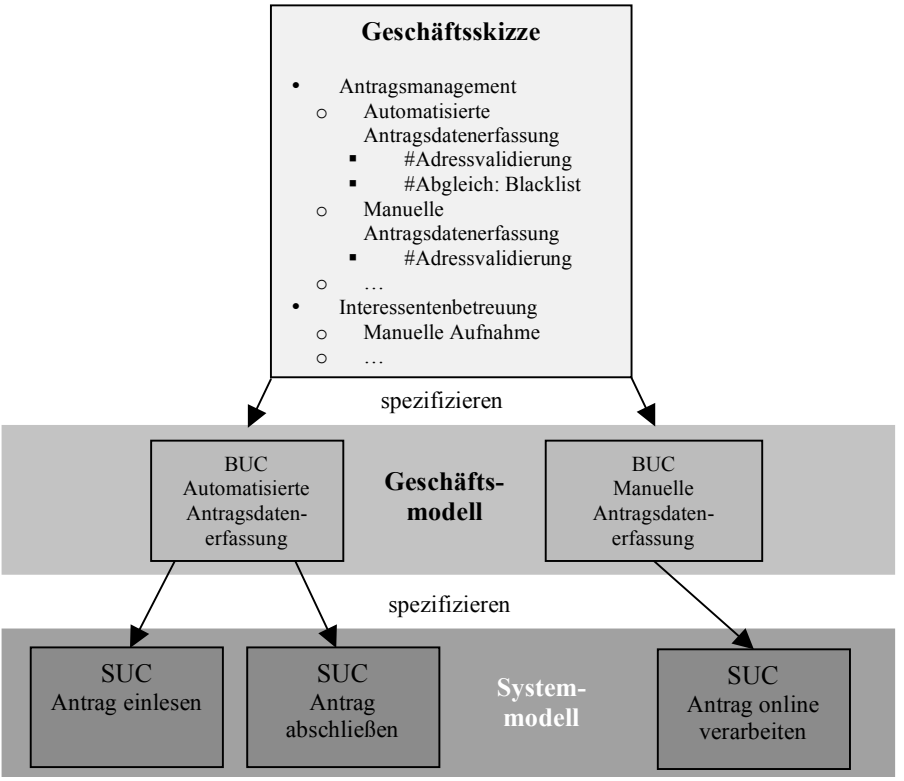

Abbildung 2: Entwicklungsartefakte in der Geschäftsanalyse und dem Anwendungsentwurf

Im letzten in Abbildung 2 gezeigten Schritt wird exemplarisch das Ableiten der Systemanwendungsfälle (englisch System Use Case, abgekürzt SUC) gezeigt. Auch zwischen BUCs und SUCs existiert eine n:m Beziehung.

2.3 Variabilität in der Geschäftsanalyse

Die explizite Modellierung von Variabilität ist eine der Grundlagen der erfolgreichen Geschäftsmodellierung einer Software Produktlinie. Die Forschung bietet verschiedene Ansätze, um dieser Herausforderung zu begegnen. Vor der Auswahl eines Vorgehens wird zunächst die in der vorgestellten Geschäftsanalyse implizit enthaltene Variabilität analysiert, um die Auswahl einer passenden Methodik zu unterstützen.

Die Variabilität in der hier vorgestellten Geschäftsanalyse wird in zwei verschiedenen Entwicklungsartefakten sichtbar. Zum einen werden in der Geschäftsskizze verschiedene übergreifende Anforderungen den einzelnen Geschäftsanforderungen zugeordnet. Diese Abhängigkeiten können in unterschiedlichen Projekten verschieden sein und sind daher projektübergreifend als variabel erkennbar.

Auch die verwendeten thematischen Gruppen und deren Geschäftsanforderungen sind abhängig von den Anforderungen des jeweiligen Projektes. Das zweite zentrale Artefakt, welches von der Variabilität betroffen ist, ist der BUC. In ihm werden bei der Modellierung je nach Projekt einerseits verschiedene Geschäftsanforderungen eingefügt und andererseits der Ablauf projektspezifisch modelliert. Weiterhin können BUCs untereinander referenziert werden, um die den verschiedenen Geschäftsanforderungen entsprechenden Prozesse darstellen zu können.

2.4 Resultierende Unzulänglichkeiten und Optimierungspotentiale

Bei der bisherigen Art der Modellierung einer Geschäftsskizze wird die in der Geschäftsskizzen-Vorlage implizit enthaltene Variabilität hinsichtlich der Geschäftsanforderungen und schließlich der möglichen, ableitbaren Geschäftsskizzen nicht explizit modelliert. Daraus resultieren folgende Probleme:

- Keine explizite Modellierung der Variabilität
- Die Abhängigkeiten einzelner Funktionalitäten untereinander werden nicht sichtbar, wodurch die Fehleranfälligkeit der Modellierung steigt
- Der geringe Grad an Formalisierung kann zu unterschiedlicher Granularität und schließlich auch Qualität bei der Modellierung einer Geschäftsskizze führen
- Die Ableitung einer Geschäftsskizze aus einer Vorlage bedeutet in jedem Projekt das Ableiten und Erstellen einer im Endeffekt vollständig eigenen Geschäftsskizze. Dadurch entsteht ein erhöhter Aufwand beim Ableiten jeder Skizze

Bei der nach der Geschäftsskizze folgenden Modellierung der BUCs werden die zuvor beschriebenen, übergreifenden Geschäftsanforderungen in die Prozessmodellierung des BUCs eingefügt. Hierbei entstehen projektspezifische BUCs. Zum aktuellen Zeitpunkt existiert hier keine explizite Modellierung der Variabilität in Bezug auf die übergreifenden Geschäftsanforderungen und den Ablauf des BUCs. Dadurch werden die Wiederverwendbarkeit der BUCs sowie die Austauschbarkeit der Geschäftsanforderungen stark eingeschränkt. Dies führt zu erhöhten Aufwänden.

Die bisher skizzierten Probleme in der Geschäftsanalyse sollen nun unter Einsatz von in der Literatur bekannten Techniken zur Modellierung von Variabilität und der Anwendungsfall basierten Entwicklung im Kontext von Software Produktlinien gelöst werden. Die Herausforderung bei der Modellierung der Variabilität innerhalb der Geschäftsskizze und der davon abhängigen BUCs besteht dabei in der ganzheitlichen Betrachtung der Variabilität in diesen beiden sehr unterschiedlichen Entwicklungsartefakten. Dabei beschreibt eine Geschäftsskizze eine statische Sicht auf die zu entwickelnde Lösung, wohingegen der BUC eine dynamische Sicht, nämlich die des Geschäftsprozesses, zeigt.

Auf Ebene der Geschäftsskizze wird eine Methodik benötigt, die sehr verschiedene Artefakte (thematische Gruppen und (übergreifende) Geschäftsanforderungen) behandeln und deren Beziehungen untereinander explizit machen kann. Es handelt sich dabei um eine statische Sicht auf das zu entwickelnde System. Dabei ist weiterhin zu beachten, dass nicht mehr für jedes einzelne Projekt eine vollkommen neue Geschäftsskizze erstellt werden soll, sondern eine Ableitung der Vorlagen-Skizze durch Bindung der Variabilität auf Ebene des Projekts erfolgt. Die Definition für Ableitung findet sich in [Ma07]. Auf Ebene der BUCs wird auch eine Methodik benötigt, die die Modellierung von Variabilität explizit ermöglicht. Dabei legen wir den Fokus auf die Variabilität in Bezug auf die Geschäftsanforderungen und vernachlässigen zunächst die Variabilität im Ablauf.

Aus den dargestellten sehr verschiedenen Anforderungen an die Modellierung von Variabilität in den beiden Entwicklungsartefakten, Geschäftsskizze und BUC, wird deutlich, dass jedem Artefakt mit einem eigenen Vorgehen für die Modellierung von Variabilität Rechnung getragen werden muss. Im Fall der Geschäftsskizze bietet die Literatur ein Featuremodell als semiformales Modellierungsinstrument an. Für die Modellierung der BUCs werden wir anschließend auf Produktlinien Anwendungsfälle (PLUC) eingehen. Beide Methoden sollen im Folgenden in unser Vorgehen integriert und anschließend kritisch betrachtet werden.

3 Featuremodellierung als Methodik für die Geschäftsskizze

Featuremodelle waren erstmals in der Domain-Engineering-Methode *Feature Oriented Domain Analysis (FODA)* [Ka90] als zentrales Element der Modellierungsaktivitäten für Software-Systeme zu finden. Sie bieten die Möglichkeit abgrenzbare, für den Auftraggeber relevante Eigenschaften eines Produkts und deren Abhängigkeiten (Variationspunkte) untereinander darzustellen. Diese als Feature bezeichneten Produkteigenschaften werden in Kernfeature und sogenannte optionale Feature unterteilt. Somit wird es möglich, variable (optionale) Produkteigenschaften explizit zu modellieren. Die Ableitung eines Featuremodells zu einem produktspezifischen Featuremodell geschieht durch Bindung von Features an Variationspunkte, wodurch die Variabilität aufgelöst wird [Ma07].

Das Featuremodell eignet sich für unsere Geschäftsskizzen-Modellierung aus folgenden Gründen:

- Es ist ein weit verbreitetes Konzept für die Modellierung von Variabilität
- Es besitzt verschiedene Variabilitätskonzepte (z.B. Auswahl, Alternativen, Implikationen, …)
- Es stellt die Abhängigkeiten zwischen den einzelnen Features dar
- Es bietet dem Auftraggeber die Möglichkeit zunächst statische Eigenschaften bzw. Funktionalitäten des späteren Systems auszuwählen

Wir verwenden deshalb ein Featuremodell zur stufenweisen Modellierung anhand unseres Beispiels Subdomäne Loyalitätsprogramme anstatt eines Modells auf Basis von Anwendungsfällen, da es zunächst nicht um die Modellierung dynamischer Abläufe geht. Im Featuremodell wird zunächst die Beziehung der einzelnen Geschäftsanforderungen zu der thematischen Gruppe modelliert. Bei dieser Modellierung werden, im Gegensatz zum bisherigen Vorgehen, die in den Geschäftsanforderungen geforderten übergreifenden Anforderungen unabhängig von den jeweiligen Geschäftsanforderungen konsolidiert modelliert. Dieses Vorgehen bietet folgende Vorteile:

- Es entstehen keine Redundanzen bei der Modellierung der übergreifenden Geschäftsanforderungen
- Die Abhängigkeiten der (übergreifenden) Geschäftsanforderungen untereinander können explizit modelliert werden

In Abbildung 3 wird ausschnittweise die thematische Gruppe *Antragsdatenmanagement* und zwei mögliche Geschäftsanforderungen dieser Gruppe dargestellt. Man kann hier erkennen, dass erst bei der Ableitung dieses Featuremodells in einem konkreten Projekt der Variationspunkt beim Feature *Antragsdatenmanagement* gebunden wird, in dem aus der Menge der zur Auswahl stehenden Geschäftsanforderungen gewählt wird. Die zuvor angesprochene Konsolidierung der übergreifenden Anforderungen geschieht im Feature *Antragsdatenprüfung*.

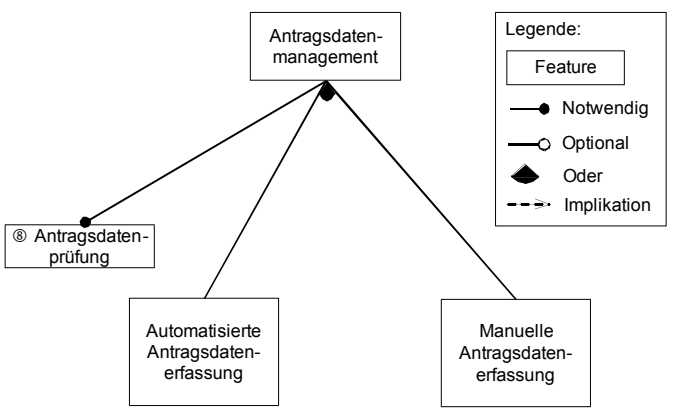

Abbildung 3: Featuremodell auf Ebene der thematischen Gruppierungen

Dieses Feature wird nun aus Platzgründen in Abbildung 4 dargestellt. Die Abbildung zeigt die übergreifenden Geschäftsanforderungen und ihre Abhängigkeiten untereinander. Die nun noch fehlenden Abhängigkeiten zwischen den Geschäftsanforderungen können einfach mit Hilfe von Implikationen modelliert werden (Beispiel in Abbildung 4: *Prüfung: Emaildubletten* impliziert *Prüfung: Email-Adresse)*.

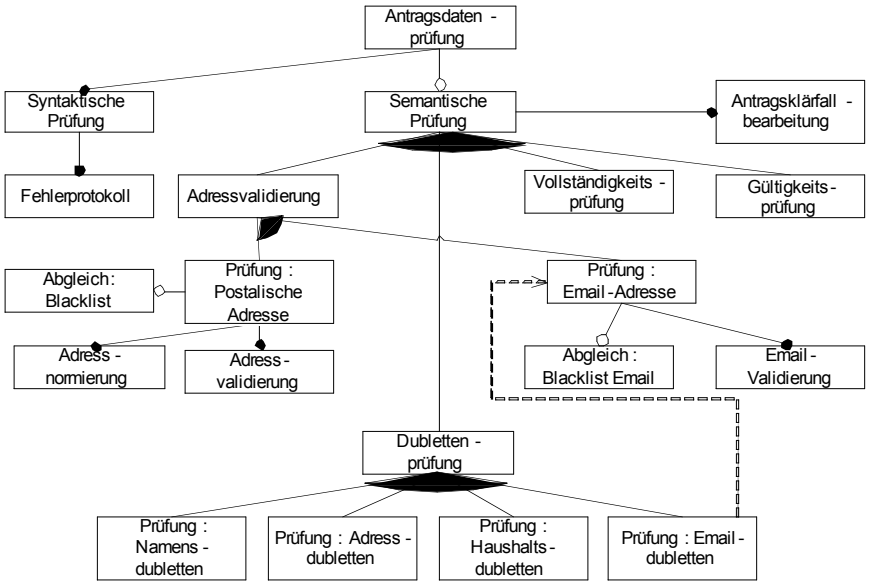

Abbildung 4: Featuremodell der übergreifenden Geschäftsanforderungen

Bei der hier vorgestellten Modellierung entsprechen die dargestellten Features Geschäftsanforderungen oder auch übergreifenden Geschäftsanforderungen. In Abschnitt 2.2 wurde bereits die existierende n:m Beziehung zwischen Geschäftsanforderungen und BUCs verdeutlicht. Auch innerhalb eines BUCs muss Variabilität explizit modelliert werden. Dabei handelt es sich nicht wie bisher in Bezug auf das Featuremodell um eine statische Sichtweise auf die Geschäftsanforderungen, sondern um eine dynamische Sicht auf die Geschäftsprozesse. Die Variabilität stellt sich hierbei in zwei Formen dar:

- Variabilität in der Ablaufbeschreibung
- Variabilität der Geschäftsanforderungen

Für die Modellierung von Variabilität innerhalb eines BUCs wird also ein eigenes Modell benötigt. Hier bietet die Literatur das Konzept der Product Line Use Cases (PLUC) [Be04] an, welches wir nun für die Modellierung der Geschäftsanforderungen in einem BUC verwenden werden.

4 Modellierung von BUCs mit der PLUC Methodik

Die Beschreibung von natürlich sprachlichen Anforderungen unter Berücksichtigung von Variabilität ist eine Aufgabe, die sich angemessen durch ein methodisches Vorgehen unterstützen lässt. Die Auswahl dieser Methode geschieht auf Basis der Tatsache, dass der Anwendungsfall als zentrales Modellierungselement in der Entwicklung bei DC vorgegeben ist.

In diesem Zusammenhang bietet sich die von Bertolino et. al. beschriebene Methodik des Produktlinien Anwendungsfalls (englisch Product Line Use Case, abgekürzt PLUC) [Be04] für eine Adaption auf den von uns beschriebenen Problembereich an. Im Folgenden wird nun ein kurzer Überblick über PLUCs gegeben.

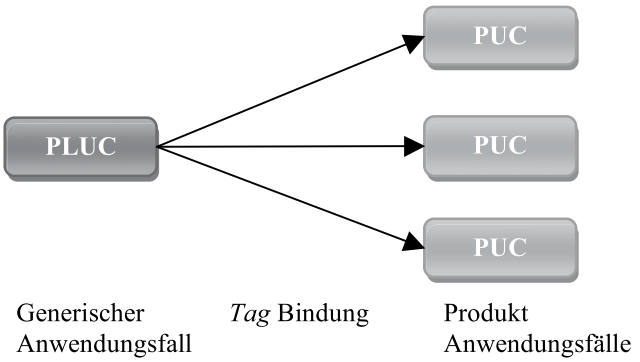

Abbildung 5: Zusammenhang PLUC und PUC

Die von Bertolino et. al. getätigte Erweiterung von Anwendungsfällen um Beschreibungsmöglichkeiten von Variabilität bietet mit *Tags* die Möglichkeit der Modellierung von Variabilität in der Form von Attributen. Diese Attribute stellen Wahlmöglichkeiten dar, die bei der Ableitung des PLUCs an ein Tag gebunden werden (Abbildung 5).

Name:		Automatisierte Antragsdatenerfassung
Vorbedingung:		Anträge liegen zur Verarbeitung bereit
Nachbedingung:		Anträge wurden in das System übernommen
Rollen:		IT-System
Erfolgreiches Szenario:		
Nr.	**Beteiligte Rollen**	**Beschreibung**
1.	IT-System	Antragsdateien einlesen
2.	IT-System	Antragsdateien auf syntaktische Korrektheit prüfen
3.	IT-System	Prüfung der **[ADR]**
4.	IT-System	Kundenvertrag erstellen
5.	IT-System	…
Variation:		
ADR: Parametrisiert [1..2]: • Prüfung: Postalische Adresse • Prüfung: Email-Adresse		

Abbildung 6: Abbildung der Variabilität mit der PLUC Methodik

Dabei entstehen produktspezifische Anwendungsfälle (englisch Product Use Cases, abgekürzt PUC). Unser BUC Beispiel *Automatisierte Antragsdatenerfassung* aus Abschnitt 4 wird nun in Abbildung 6 mit Hilfe der PLUC Methodik ausschnittsweise beschrieben. Dabei wird die Variabilität der beiden übergreifenden Geschäftsanforderungen *Prüfung: Postalische Adresse* und *Prüfung: Email-Adresse* berücksichtigt. Der Tag [ADR] zeigt hierbei die beiden Geschäftsanforderungen an, die bei der Ableitung dieses BUCs gebunden werden können. Dabei zeigt [1..2] an, dass mindestens ein und höchstens zwei, in diesem Fall also alle, Geschäftsanforderungen gebunden werden können. Mit diesem Vorgehen kann die Variabilität von Geschäftsanforderungen in Bezug auf PLUCs dargestellt werden.

5 Ergebnis und weitere Arbeit

In diesem Beitrag wurde exemplarisch gezeigt, dass Variabilität in der Geschäftsanalyse bei DC eine große Bedeutung hat. Erste Erfahrungen mit dem expliziten Umgang mit Variabilität wurden in einem Kundenprojekt gemacht. Dabei wurde zunächst die in der Geschäftsanalyse vorhandene Variabilität analysiert und die zum aktuellen Zeitpunkt existierenden Unzulänglichkeiten in diesem Kontext herausgestellt. Wichtig war dabei die Erkenntnis, dass der Variabilität in beiden Entwicklungsartefakten Geschäftsskizze (statische Sicht) und BUC (dynamische Sicht) durch zwei verschiedene Modelle Rechnung getragen werden muss. Dieser Beitrag unterscheidet sich dabei von [SP07] durch die Wahl des Anwendungsfalls zur Modellierung von Geschäftsprozessen im Vergleich zum dort gewählten BPMN Ansatz. Weiterhin wurden folgende Vorteile durch den Einsatz der neuen Modelle identifiziert:

- Das Featuremodell unterstützt die Qualität der Modellierung von statischer Variabilität und hilft bei der Vermeidung von Modellierungsfehlern, da die in der Geschäftsmodellierung vorhandene Variabilität explizit dargestellt wird
- Der Einsatz eines Featuremodells bei der Modellierung der Geschäftsskizze entfernt die Redundanzen in Bezug auf die übergreifenden Geschäftsanforderungen und stellt die Abhängigkeiten der einzelnen Geschäftsanforderungen explizit dar.
- Ein Featuremodell bietet darüber hinaus durch die Ableitung einen Mechanismus an, um für neue Produkte schneller Geschäftsskizzen abzuleiten. Dies stellt eine Zeitersparnis im Vergleich zum bisherigen Vorgehen dar.
- Die bei der Modellierung von BUCs existierende Variabilität wurde durch den Einsatz von PLUCs explizit dargestellt.
- PLUCs bieten durch die Ableitung in PUCs eine Zeitersparnis im Vergleich zum bisherigen Vorgehen.

Durch den Einsatz eines Featuremodells und den PLUCs entstehen offensichtlich Abhängigkeiten zwischen diesen beiden. Diese Abhängigkeiten führen zu neuen Fragestellungen:

- Die Wahl einer Geschäftsanforderung in einem BUC bedeutet auch eine Auswahl und Bindung eben dieser Anforderung auf Ebene des Featuremodells. Diese Abhängigkeit ist im Moment nicht explizit sichtbar bzw. modelliert und kann bei der Ableitung eines Produktes zu Problemen führen, wenn die Anforderung bei der Ableitung der Geschäftsskizze nicht gewählt wurde.
- Ein bisher unberücksichtigter Punkt ist die Modellierung von Variabilität in der Ablaufbeschreibung eines Anwendungsfalls (Abschnitt 3), welche nicht mit TAGs abzubilden ist.

Aus diesen Gründen wäre ein nächster Schritt die genaue Analyse der existierenden Abhängigkeiten zwischen Features und PLUCs und der Abbildung von Features auf PLUCs. Dabei ist die Betrachtung von statischer und dynamischer Variabilität von großer Bedeutung. In diesem Zusammenhang sollte auch grundlegend geklärt werden, welche Artefakte der Geschäftsanalyse wie auf das Artefakt Feature abgebildet werden sollen. Die in dieser Fallstudie getätigte Abbildung von Geschäftsanforderungen auf Features ist nur eine Möglichkeit.

Literaturverzeichnis

[Be04] Bertolino, A.; Fantechi, A.; Gnesi, S.; Lami, G.: Product Line Use Cases: Scenario-Based Specification and Testing of Requirements. In (Käkölä, T.; Ducnas, J. C.; Hrsg.): Software Product Lines: Research Issues in Engineering and Requirements, Springer Verlag, 2006.

[CN01] Clements, P.; Northrop, L.: Software Product Lines: Practices and Patterns, Addison-Wesley Professional, 2001.

[Co01] Cockburn, A.: Writing Effective Use Cases, Addition-Wesley, 2001.

[Ja92] Jacobson, I.; Christerson, M.; Jonsson, P.; Øvergaard, G.: Object-Oriented Software Engineering – A Use Case Driven Approach, Addision-Wesley, 2. Auflage, 1992.

[Ka90] Kang, K. C.; Cohen S.; Hess, J. A.; Novak, W. E.; Peterson, A. S.: Feature Oriented Domain Analysis (FODA) Feasibility Study, Technical Report CMU/SEI-90-TR21, 1990.

[Kr99] Kruchten, P.: Der Rational Unified Process - Eine Einführung, Addison-Wesley, 2. aufl. edition, 1999.

[OM04] OMG: UML 2.0 Superstrukture Specification, Verfügbar unter: http://www.omg.org, Januar 2008.

[PBL05] Pohl, K.; Böckle, G.; van der Linden, F. J.: Software Product Line Engineering: Foundations, Principles and Techniques, Springer, 2005.

[SP07] Schnieders, A.; Puhlmann, F.: Variability Modeling and Product Derivation in E-Business Process Families. In: Technologies for Information Systems, Springer, 2007.

[Ma07] von der Maßen, T.: Feature-basierte Modellierung und Analyse von Variabilität in Produktlinienanforderungen, Dissertation RWTH Aachen, 2007.