

# Vom Projekt zum Produkt durch Produktlinien und Variantenmanagement

Kim Lauenroth

paluno – the Ruhr Institute for Software Technology  
Universität Duisburg-Essen  
Gerlingstraße 16  
45127 Essen  
kim.lauenroth@paluno.uni-due.de

**Abstract:** Die Produktlinienentwicklung ist ein etabliertes Paradigma zur parallelen Entwicklung gleichartiger Softwaresysteme mit hoher Qualität und in kurzer Zeit. Dieser Beitrag gibt einen Überblick über die wesentlichen Konzepte und Bestandteile der Produktionsentwicklung und diskutiert, wie Konzepte der Produktlinienentwicklung einen erfolgreichen Übergang vom Projekt- zum Produktgeschäft unterstützen können.

## 1 Einleitung

Die Produktlinienentwicklung ist ein etabliertes Paradigma zur parallelen Entwicklung gleichartiger Softwaresysteme mit hoher Qualität und in kurzer Zeit [PBL05]. Die zentrale Idee der Produktlinienentwicklung ist die proaktive Wiederverwendung von Entwicklungsartefakten.

Die Wiederverwendung beschränkt sich hierbei nicht nur auf Realisierungsartefakte wie beispielweise Komponenten oder Codefragmente. Die Proaktivität der Wiederverwendung zeichnet sich dadurch aus, dass bereits das Requirements Engineering und das Design in der Produktlinienentwicklung auf die Wiederverwendung ausgerichtet werden. Eine zentrale Rolle bei dieser Ausrichtung spielt das Varianten- bzw. Variabilitätsmanagement, das darauf abzielt, die notwendige Anpassbarkeit in den Entwicklungsartefakten (d.h. Anforderungen, Architektur, Code, etc.) zu identifizieren, zu dokumentieren und in den einzelnen Entwicklungsphasen zu verwalten.

Dieser Beitrag gibt einen Überblick über die wesentlichen Entwicklungsprozesse der Produktlinienentwicklung (Abschnitt 2) und über das Variantenmanagement (Abschnitt 3). Anschließend wird diskutiert, wie Konzepte der Produktlinienentwicklung einen erfolgreichen Übergang vom Projekt- zum Produktgeschäft unterstützen können (Abschnitt 4).

## 2 Entwicklungsprozesse der Produktlinienentwicklung

Die Produktlinienentwicklung besteht im Unterschied zur Einzelsystementwicklung aus zwei Entwicklungsprozessen (vgl. [WL99]). Abbildung 1 zeigt das Rahmenwerk der Produktlinienentwicklung nach [PBL05]. Das Rahmenwerk unterscheidet die Entwicklungsprozesse Domänen-Engineering und Applikations-Engineering, welche im Folgenden vorgestellt werden.

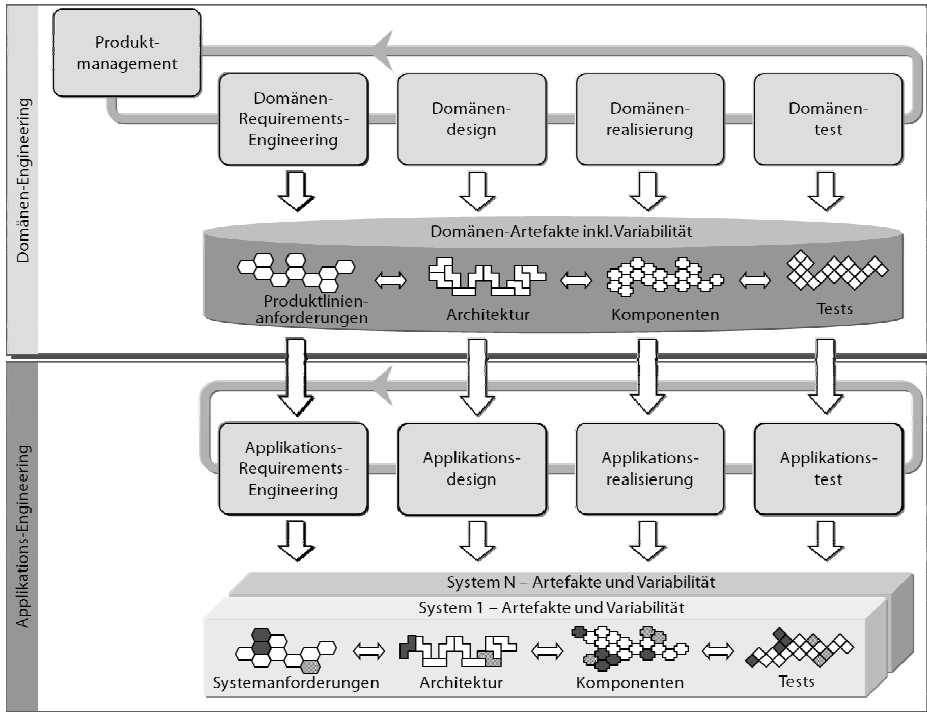


Abbildung 1: Rahmenwerk der Produktlinienentwicklung [PBL05]

### 2.1 Domänen-Engineering

Das Ziel des *Domänen-Engineering* besteht darin, Domänen-Artefakte der Produktlinie zu planen und zu entwickeln. Unter Domänen-Artefakten werden alle Entwicklungsartefakte verstanden, die für die Entwicklung von Produkten der Produktlinie bereitgestellt werden. Es werden des Weiteren zwei Arten von Domänen-Artefakten unterschieden:

- *Gemeinsame Domänen-Artefakte* sind Bestandteil aller Produkte der Produktlinie. Dies sind zum Beispiel gemeinsame Komponenten, die in allen Produkten vorhanden sind oder Anforderungen, die von allen Produkten erfüllt werden müssen.

- *Variable Domänen-Artefakte* sind wählbar bzw. anpassbar, um kunden- oder marktspezifische Produkte der Produktlinie zu entwickeln. Dies sind zum Beispiel Komponenten, die für spezifische Anwendungsfälle konfiguriert werden können oder auch als Ganzes gewählt werden können.

Zur Entwicklung von Domänen-Artefakten werden im Rahmen des Domänen-Engineering alle aus der Einzelsystementwicklung bekannten Aktivitäten ausgeführt [PBL05]:

- Das *Produktmanagement* befasst sich mit den ökonomischen Aspekten der Produktlinie und ist für marktstrategische Fragestellungen der Produktlinie verantwortlich.
- Das *Domänen-Requirements-Engineering* hat das Ziel, Requirements Engineering für die Produktlinie zu betreiben, um Anforderungen an die Produktlinie zu gewinnen. Im Unterschied zur Einzelsystementwicklung besteht die Herausforderung im Domänen-Requirements-Engineering insbesondere darin, gemeinsame und variable Anforderungen der Produktlinie zu identifizieren und für die nachgelagerten Aktivitäten verfügbar zu machen.
- Das *Domänen-design* entwickelt die Architektur der Produktlinie basierend auf den Produktlinienanforderungen. Die besondere Herausforderung für das Domänen-design besteht darin, die gemeinsamen und variablen Produktlinienanforderungen möglichst effizient umzusetzen und eine Architektur zu entwickeln, die möglichst robust gegenüber Änderungen der Produktlinienanforderungen ist.
- Die *Domänenrealisierung* implementiert die wiederverwendbaren Bestandteile der Produktlinie. Die besondere Herausforderung der Domänenrealisierung besteht darin, die gemeinsamen und variablen Anforderungen möglichst effizient umzusetzen.
- Der *Domänentest* hat die Aufgabe, eine möglichst hohe Qualität der Domänen-Artefakte zu gewährleisten. Die besondere Herausforderung des Domänentests besteht zum einen darin, die Variabilität der Domänen-Artefakte bei der Qualitätssicherung zu berücksichtigen. Dies beinhaltet zum einen, die noch nicht implementierten Bestandteile der Domänen-Artefakte zu berücksichtigen, und zum anderen, die Qualität aller möglichen Variantenausprägungen sicherzustellen.

Ein zentraler Unterschied zwischen der Einzelsystementwicklung und dem Domänen-Engineering ist die explizite Betrachtung und Behandlung von Variabilität in allen Aktivitäten des Domänen-Engineering. Diese zu allen Aktivitäten des Domänen-Engineering querschneidende Aktivität ist Teil des Variantenmanagements und wird im Detail in Abschnitt 3.2 beschrieben.

## 2.2 Applikations-Engineering

Im *Applikations-Engineering* werden spezifische Produkte der Produktlinie basierend auf den gemeinsamen und variablen Domänen-Artefakten der Produktlinie abgeleitet. Hierzu werden im Applikations-Engineering die folgenden Aktivitäten ausgeführt [PBL05]:

- Im *Applikations-Requirements-Engineering* werden Anforderungen an das abzuleitende Produkt definiert. Die besondere Herausforderung besteht darin, möglichst viele Produktlinienanforderungen wiederzuverwenden und die Abweichungen zwischen kunden- bzw. marktspezifischen Anforderungen und Produktlinienanforderungen zu erkennen.
- Im *Applikationsdesign* wird die Architektur für das abzuleitende Produkt durch Wiederverwendung der Produktlinienarchitektur definiert. Die besondere Herausforderung besteht darin, einen möglichst hohen Grad an Wiederverwendung zu erzielen.
- In der *Applikationsrealisierung* wird das Produkt durch Wiederverwendung der Produktlinienkomponenten realisiert. Die besondere Herausforderung besteht darin, kunden- oder marktspezifische Anpassungen der jeweiligen Produkte mit möglichst geringem Aufwand zu realisieren.
- Im *Applikationstest* wird die Qualität des abgeleiteten Produktes gesichert. Die besondere Herausforderung im Applikationstest besteht darin, die Ergebnisse des Domäne-Test wiederzuverwenden und möglichst keine zum Domänentest redundanten Tests vorzunehmen, d.h. es sollten nur solche Qualitätssicherungsmaßnahmen durchgeführt werden, die kunden- oder marktspezifische Teile des Produktes betreffen.

Ein zentraler Unterschied zwischen der Einzelsystementwicklung und dem Applikations-Engineering besteht darin, dass das alle Aktivitäten des Applikations-Engineering auf den Domänen-Artefakten der Produktlinie aufsetzen und damit auch die Varianteninformation berücksichtigen und nutzen können. Diese Berücksichtigung und Ausnutzung der Varianteninformation ist ebenfalls Bestandteil des Variantenmanagements und wird in Abschnitt 3.3 beschrieben.

### 3 Variantenmanagement

Die explizite Betrachtung und Behandlung der Variabilität ist, neben der Unterscheidung der zwei Entwicklungsprozesse Domänen- und Applikations-Engineering, ein weiteres wesentliches Unterscheidungsmerkmal zwischen Produktlinien- und Einzelsystementwicklung (vgl. [Kr02]). Variabilität wird in der Literatur definiert als die Fähigkeit eines Domänenartefakts verändert bzw. angepasst zu werden (vgl. [GB02]).

Aufgrund der zentralen Rolle der Variabilität in der Produktlinienentwicklung hat es sich in Forschung und Praxis durchgesetzt, die Variabilität einer Produktlinie in einem eigenständigen Modell, dem sogenannten Variabilitätsmodell, zu definieren. Das Variabilitätsmodell einer Produktlinie hat die Aufgabe zu dokumentieren, in welcher Weise variable Domänenartefakte einer Produktlinie variieren können bzw. anpassbar sind (vgl. [PBL05]), z.B. kann das Variabilitätsmodell definieren, dass sich bestimmte variable Artefakte ausschließen. Dieses eigenständige Variabilitätsmodell bildet die Arbeitsgrundlage für alle Aktivitäten in der Produktlinienentwicklung, die Bezug zur Variabilität der Produktlinie haben.

Im Folgenden wird daher zunächst das orthogonale Variabilitätsmodell als eine mögliche Ausprägung für die Dokumentation von Variabilität im Variantenmanagement vorgestellt. Im Anschluss daran wird in Abschnitt 3.2 bzw. 3.3 das Variantenmanagement im Domänen- bzw. Applikations-Engineering beschrieben.

### 3.1 Dokumentation von Variabilität mit dem orthogonalen Variabilitätsmodell

Das orthogonale Variabilitätsmodell (OVM, vgl. [PBL05]) unterscheidet zur Dokumentation von Variabilität die folgenden Konzepte:

- *Variationspunkte* beschreiben, was oder welcher Aspekt in einer Produktlinie variiert. Es werden obligatorische und optionale Variationspunkte unterschieden. Obligatorische Variationspunkte müssen bei der Ableitung eines Produktes betrachtet werden, optionale Variationspunkte können betrachtet werden. Darüber hinaus können interne und externe Variationspunkte unterschieden werden. Externe Variationspunkte sind allgemein sichtbar, interne Variationspunkte sind nur für Entwickler und nicht für Kunden sichtbar.
- *Varianten* beschreiben, wie etwas in einer Produktlinie variiert, d.h. Varianten beschreiben mögliche Ausprägungen eines Variationspunktes.
- *Variabilitätsabhängigkeiten* zwischen Variationspunkt und Variante beschreiben, welcher Weise eine Variante an einem Variationspunkt gewählt werden darf. Es werden obligatorische, optionale und alternative Variabilitätsabhängigkeiten unterschieden.
- *Bedingungsabhängigkeiten* beschreiben Einschränkungen in der Auswahl von Varianten und Variationspunkten. Es werden Erfordert- und Ausschlussabhängigkeiten unterschieden.

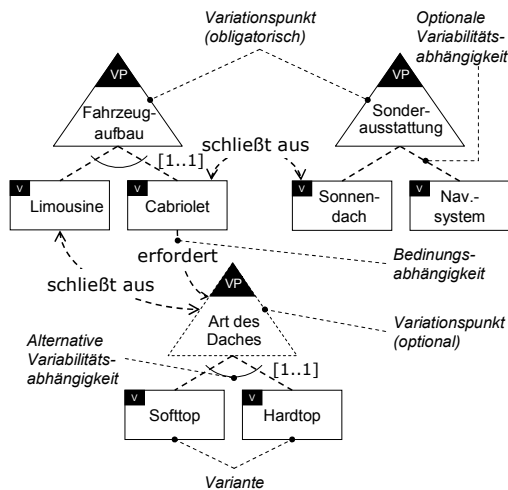


Abbildung 2: Beispiel für ein OVM [LP09]

Abbildung 2 zeigt ein einfaches orthogonales Variabilitätsmodell für PKWs als Beispiel. Es spezifiziert, dass ein Kunde über den Fahrzeugaufbau und die Sonderausstattung entscheiden muss (beides obligatorische Variationspunkte). Der Kunde kann entweder Limousine oder Cabriolet wählen (alternative Abhängigkeit<sup>1</sup>). Wenn der Kunde ein Cabriolet wählt, muss er über die Art des Daches entscheiden (optionaler Variationspunkt, verbunden mit der Variante Cabriolet durch eine Erfordert-Bedingungsabhängigkeit). Gleichzeitig darf er kein Sonnendach als Sonderausstattung wählen (Ausschluss-Bedingungsabhängigkeit). Wenn der Kunde die Variante Limousine wählt, darf er die Art des Daches nicht entscheiden (Ausschluss-Bedingungsabhängigkeit).

Neben der eigentlichen Variabilitätsinformation müssen auch die Auswirkungen der Variabilität auf die verschiedenen Domänen-Artefakte dokumentiert werden. Hierzu definiert das orthogonale Variabilitätsmodell die sogenannten *Artefaktabhängigkeiten*. Eine Artefaktabhängigkeit dokumentiert die Auswirkungen der Variabilität auf die Domänen-Artefakte durch eine Beziehung zwischen Varianten und (Teilen von) Domänen-Artefakten. Wenn eine Variante z.B. mit einer Anforderung in Beziehung steht, ist diese Anforderung nur dann für ein Produkt relevant, wenn die entsprechende Variante gewählt ist. Wenn ein Artefakt(teil) mit keiner Variante in Beziehung steht, handelt es sich um ein gemeinsames Artefakt (oder einen gemeinsamen Artefakteil), der Bestandteil aller Produkte der Produktlinie ist.

Abbildung 3 zeigt ein Beispiel für Artefaktabhängigkeiten. Zum Beispiel ist die Variante Sonnendach über eine Artefaktabhängigkeit mit der Anforderung R-2 verbunden. Damit ist diese Anforderung nur dann für ein Produkt zu erfüllen, wenn die Variante Sonnendach gewählt ist. Die Anforderung R-3 steht mit keiner Variante in Beziehung. Damit muss diese Anforderung für alle Produkte der Produktlinie erfüllt werden (und damit hat jedes Fahrzeug eine Klimaanlage als Sonderausstattung).

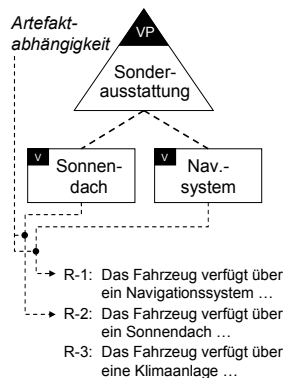


Abbildung 3: Beispiel für Artefaktabhängigkeiten [LP09]

<sup>1</sup> Die Kardinalität in Form von „[1..1]“ gibt an, dass bei dieser Alternative mindestens eine und höchstens eine Variante gewählt werden darf. Andere Kardinalitäten sind ebenfalls möglich, z.B.: beschreibt „[0..2]“, dass keine oder höchstens zwei Varianten gewählt werden dürfen.

### 3.2 Variantenmanagement im Domänen-Engineering

Das Variantenmanagement im Domänen-Engineering ist eine zu den regulären Entwicklungsaktivitäten querschneidende Aktivität und erfüllt die folgenden Aufgaben [LP09]:

- *Identifikation von Variabilität:* Die Aufgabe der Identifikationsaktivität besteht darin, während der Durchführung der Aktivitäten des Domänen-Engineering Variabilität in den Domänen-Artefakten zu identifizieren. Im Sinne des orthogonalen Variabilitätsmodells besteht die Identifikation von Variabilität darin, mögliche Variationspunkte und Varianten zu identifizieren.
- *Auswirkungsanalyse:* Die Aufgabe der Auswirkungsanalyse besteht darin, die Auswirkung der identifizierten Variabilität auf die Domänen-Artefakte und die übrige Variabilität der Produktlinie zu untersuchen. Im Sinne des orthogonalen Variabilitätsmodells bedeutet die Auswirkungsanalyse zum einen die Analyse der neu identifizierten Variabilität auf neue Bedingungsabhängigkeiten zu anderen Elementen des Variabilitätsmodells und zum anderen die Analyse der Variabilität in Bezug auf Artefaktabhängigkeiten zu Domänen-Artefakten der Produktlinie.
- *Dokumentation von Variabilität:* Die Aufgabe der Dokumentationsaktivität besteht darin, die identifizierte Variabilität zu dokumentieren. Zur Dokumentation der identifizierten Variabilität kann das orthogonale Variabilitätsmodell inklusive der Artefaktabhängigkeiten verwendet werden. (siehe Abschnitt 3.1).

### 3.3 Variantenmanagement im Applikations-Engineering

Das Variantenmanagement im Applikations-Engineering ist analog zum Domänen-Engineering eine querschneidende Aktivität und erfüllt die folgenden Aufgaben [BHL06]:

- *Kommunikation der Variabilität:* Bei der Ableitung eines Produktes von einer Produktlinie muss die Variabilität der Produktlinie in geeigneter Form an diejenigen Stakeholder kommuniziert werden, die für die jeweilige Entwicklungsaktivität die Entscheidungen treffen sollen. Dies können beispielsweise Kunden, Nutzer des Produktes oder auch Entwickler der Produktlinie sein. Die adäquate Kommunikation der Variabilität an die Stakeholder ist ein essenzieller Erfolgsfaktor für ein Produktlinienprojekt, damit die Stakeholder eine umfassende Grundlage für eine fundierte Entscheidung treffen können.
- *Bindung der Variabilität:* Nachdem eine Entscheidung in Bezug auf die Variabilität getroffen wurde, müssen die Auswirkungen der Variabilität auf die Domänen-Artefakte realisiert werden. Dieser Prozess wird im Allgemeinen auch als Bindung der Variabilität bezeichnet. Zu diesem Prozess gehört beispielsweise, dass nicht gewählte Anforderungen verworfen oder nicht gewählte Komponenten entfernt werden.
- *Auswahldokumentation:* Die Aufgabe der Auswahldokumentation besteht darin, die getroffenen Auswahlentscheidungen der Stakeholder zu dokumentieren.

- *Dokumentation applikationsspezifischer Anpassungen:* Typischerweise können nicht alle Kundenwünsche und Anforderungen durch eine Produktlinie erfüllt werden. Kundenindividuelle Anpassungen an ein Produkt einer Produktlinie werden in diesem Zusammenhang auch als applikationsspezifische Anpassungen bezeichnet und können für die Weiterentwicklung einer Produktlinie von großer Relevanz sein.

## 4 Unterstützung des Übergangs vom Projekt zum Produkt

Ein Projekt wird im Folgenden dahingehend verstanden, dass ein Projekt ein auf einen spezifischen Kunden hin zugeschnittenes System erstellt. Ein Produkt hingegen ist ein standardisiertes System, das mit dem Ziel erstellt wurde, die Anforderungen eines Marktes zu erfüllen. In der weiteren Diskussion wird insbesondere der Fall betrachtet, dass ein laufendes Projekt unter der Prämisse durchgeführt wird, dass das Ergebnis des Projektes in einem nachgelagerten Schritt zu einem Produkt ausgebaut wird.

### 4.1 Implikationen aus der Trennung von Domänen- und Applikations Engineering

Im Projektgeschäft werden typischerweise alle Projektaktivitäten auf die Erfüllung der Kundenwünsche hin ausgerichtet, d.h. die konzeptuellen Aktivitäten fokussieren ausschließlich die Anforderungen und Bedürfnisse des Projektes, die Realisierungsaktivitäten setzen diese um und die Qualitätssicherungsmaßnahmen prüfen die korrekte Umsetzung. Die Trennung von Domänen- und Applikations-Engineering in der Produktlinien-entwicklung motiviert sich unter anderem aus der Erkenntnis, dass die erfolgreiche Entwicklung von wiederverwendbaren Domänen-Artefakte eine gezielte Planung und strategische Realisierung erfordert.

Für den erfolgreichen Übergang vom Projekt zum Produkt impliziert dies, dass neben den Anforderungen des konkreten Projektes auch mögliche Anforderungen aus einem späteren potenziellen Produktkontext erfasst, analysiert und dokumentiert werden müssen. Das Requirements Engineering eines Projektes kann parallel zur Erhebung der Anforderungen für das Projekt eine Analyse in Bezug auf mögliche gemeinsame Anforderungen vornehmen, die auch vom späteren Produkt erfüllt werden müssen. In dem Fall, dass Anforderungen nur für den spezifischen Projektkunden relevant sind, kann eine zusätzliche Erhebung möglicher Anforderungen an das geplante Produkt erfolgen. Mögliche Ergebnisse dieses Schrittes sind:

- *Gemeinsame Anforderungen für Projekt und Produkt:* Analog zu gemeinsamen Anforderungen in der Produktlinienentwicklung sind dies Anforderungen, die sowohl für das Projekt als auch für das geplante Produkt gelten.
- *Bekannte variable Anforderungen:* Analog zur Produktlinienentwicklung sind dies Anforderungen, deren Ausprägungen sowohl für das Projekt als auch für das geplante Produkt bekannt sind.
- *Potenziell variable Anforderungen:* Dies sind Anforderungen des Projektes, für die jedoch vermutet wird, dass diese für das geplante Produkt anders ausgeprägt sein können.



- *Spezifische Anforderungen für das Projekt:* Dies sind Anforderungen, die ausschließlich für das Projekt definiert wurden und keine Rolle für das Produkt spielen.

Die zuvor beschriebenen Erkenntnisse über Gemeinsamkeiten und Unterschiede in den Anforderungen von Projekt und Produkt stellen für genommen bereits einen großen Mehrwert dar. Nach Abschluss der Projektstätigkeiten können diese Informationen genutzt werden, um das Projektergebnis zu einem Produkt weiterzuentwickeln. Die Produktlinienentwicklung bietet allerdings noch darüberhinausgehende Möglichkeiten. Der zweite Schritt der strategischen Vorgehensweise in der Produktlinienentwicklung ist ein gezieltes Design und eine gezielte Realisierung von Domänen-Artefakten basierend auf den Ergebnissen aus dem Domänen-Requirements-Engineering.

Für den Übergang vom Projekt zum Produkt impliziert dies, dass die Design- und Realisierungsaktivitäten für das aktuelle Projekt nicht ausschließlich auf den Anforderungen des Projektes basierend sollten, sondern nach Möglichkeit auch bereits die Ergebnisse in Bezug auf das geplante Produkt berücksichtigen sollten. Das zusätzliche Wissen um das geplante Produkt (bspw. in Form von gemeinsamen oder variablen Anforderungen) erlaubt einen wesentlich gezielteren Entwurf der Architektur und eine gezieltere Realisierung in Bezug auf eine schnelle und leichte Anpassung des Projektes hin zum Produkt. Mögliche Ergebnisse dieses Schrittes sind:

- *Gemeinsame Realisierungsartefakte für Projekt und Produkt:* Analog zu gemeinsamen Realisierungsartefakten in der Produktlinienentwicklung sind dies Artefakte, die unverändert im Projekt und im Produkt verwendet werden können.
- *Spezifische Realisierungsartefakte für das Projekt:* Dies sind Realisierungsartefakte, die ausschließlich für das Projekt erstellt werden.
- *Potenziell variable Realisierungsartefakte:* Dies sind Realisierungsartefakte, die für das Projekt entwickelt wurden, wobei Anpassungsmöglichkeiten vorgesehen sind.
- *Variable Realisierungsartefakte:* Dies sind Realisierungsartefakte, die durch definierte Anpassung sowohl im Projekt als auch im Produkt verwendet werden können.

Zusammenfassend kann festgehalten werden, dass das Wissen um die Gemeinsamkeiten und Unterschiede (bzw. Varianten) zwischen dem Projekt und dem geplanten Produkt einen entscheidenden Beitrag für eine erfolgreiche Produktentwicklung darstellt.

## 4.2 Implikationen aus dem Variantenmanagement

Für die Produktlinienentwicklung ist das Variantenmanagement ein essenzieller Erfolgsfaktor, um die Komplexität der Gemeinsamkeiten und Varianten der Produktlinie zu beherrschen. Im vorherigen Abschnitt wurde gezeigt, dass das Wissen um Gemeinsamkeiten und Varianten zwischen Projekt und Produkt einen großen Wert besitzt. Durch ein angemessenes Variantenmanagement im Projekt kann dieses Wissen für den Entwicklungsprozess und das geplante Produkt verfügbar und damit nutzbar gemacht werden. Prinzipiell kann der Reifegrad des Variantenmanagement wie folgt klassifiziert werden:

- *Zufällig:* Gemeinsamkeiten und Varianten werden zufällig identifiziert, aber nicht dokumentiert. Das Wissen über mögliche Gemeinsamkeiten und Varianten befindet sich ausschließlich im Kopf der Entwickler.
- *Unstrukturiert:* Gemeinsamkeiten und Varianten werden zufällig identifiziert und unstrukturiert dokumentiert, beispielsweise durch textuelle Annotationen in Entwicklungsartefakten.
- *Strukturiert:* Gemeinsamkeiten und Varianten werden bewusst identifiziert, sowie eigenständig und strukturiert dokumentiert, beispielsweise in gesonderten Dokumenten oder Modellen.
- *Umfassend:* Zusätzlich zu den Gemeinsamkeiten und Varianten werden die Auswirkungen der Gemeinsamkeiten und Varianten auf die jeweiligen Entwicklungsartefakte analysiert und strukturiert durch geeignete Nachvollziehbarkeitsinformationen dokumentiert.

In Abhängigkeit davon, in welcher Ausprägung das Variantenmanagement für ein Projekt betrieben wird, ergeben sich verschiedene Vorteile für den Übergang von Projekt zum Produkt. Im Requirements Engineering für das Projekt kann bereits ein erster Mehrwert erzielt werden, wenn die Gemeinsamkeiten und Varianten von Projekt und Produkt unstrukturiert gemanagt werden. Durch das Bewusstsein über die Existenz von Varianten können Gemeinsamkeiten und Unterschiede identifiziert und für weitere Entwicklungsaktivitäten verfügbar gemacht werden. Ein strukturiertes Variantenmanagement macht dieses Wissen strukturiert für weitere Entwicklungsaktivitäten im Projekt verfügbar. Das umfassende Variantenmanagement im Requirements Engineering bietet den größten Mehrwert für den Übergang vom Projekt zum Produkt, da nicht nur die Gemeinsamkeiten und Varianten, sondern auch die Auswirkungen in Bezug auf die Anforderungen verstanden und dokumentiert werden. In dieser Ausprägung würde das Requirements Engineering für das Projekt in wesentlichen Aspekten dem Requirements Engineering für eine Produktlinie entsprechen.

Die Design- und Realisierungsaktivitäten profitieren aufgrund der typischen Komplexität von Design und Realisierungsartefakten nur in geringer Weise von einem unstrukturierten Variantenmanagement. Ein strukturiertes Variantenmanagement in Bezug auf Design- und Realisierungsaktivitäten erlaubt es hingegen, dass die Gemeinsamkeiten und Unterschiede zwischen Projekt und Produkt explizit erfasst werden und ermöglicht dadurch einen wesentlich effektiveren Übergang vom Projekt zum Produkt. Zum einen sind die gemeinsamen Aspekte bekannt, d.h. die Design- und Realisierungsartefakte können gezielt entwickelt und auf Bestandteile untersucht werden, die unverändert in das Produkt übernommen werden können. Des Weiteren sind die variablen Aspekte bekannt. Dies erlaubt wiederum eine gezielte Entwicklung und Suche nach Anpassungen der vorhandenen Design- und Realisierungsartefakte.

Ein umfassendes Variantenmanagement der Design- und Realisierungsartefakte bietet den größten Mehrwert für den Übergang vom Projekt zum Produkt, da neben den explizit dokumentierten Gemeinsamkeiten und Varianten auch die Auswirkungen auf die Design- und Realisierungsartefakte bekannt und dokumentiert sind. Dies ermöglicht zum einen den unmittelbaren Zugriff auf diejenigen Bestandteile der Design- und Realisierungsartefakte, die unverändert in das Produkt übernommen werden können und zum anderen werden explizit diejenigen Stellen dokumentiert, die einer Anpassung im geplanten Produkt bedürfen.

## **5 Zusammenfassung und Fazit**

In diesem Beitrag wurden die wesentlichen Konzepte der Produktlinienentwicklung vorgestellt: Trennung von Domänen- und Applikations-Engineering sowie die explizite Betrachtung von Gemeinsamkeiten und Varianten durch das Variantenmanagement. Es wurde gezeigt, dass diese Konzepte einen erfolgreichen Übergang vom Projekt zum Produkt unterstützen können.

Eine wesentliche Erkenntnis für den Übergang vom Projekt zum Produkt aus der Produktlinienentwicklung ist das Wissen um die Gemeinsamkeiten und Unterschiede von Projektergebnis und Produkt. Das Vorhandensein dieses Wissens erlaubt eine wesentlich strukturiertere Ausrichtung der Entwicklungsaktivitäten des Projektes in Bezug auf das geplante Produkt. Beispielsweise können frühzeitig gemeinsame Bestandteile realisiert werden, die unverändert vom Projekt in das Produkt übertragen werden können. Verschiedene Ausprägungen des Variantenmanagements können dieses Wissen in mehr oder minder stark strukturierter Weise den jeweiligen Entwicklungsaktivitäten zuführen.

## Literaturverzeichnis

- [BHL06] Bühne, S., Halmans, G., Lauenroth, K., Pohl, K.: Scenario-based Application Requirements Engineering. In: Software Product Lines - Research Issues in Engineering and Management. Springer, 2006.
- [GB02] Geyer, L.; Becker, M.: On the Influence of Variabilities on the Application Engineering Process of a Product Family. Proceedings of the 2nd the Second Software Product Line Conference 2002.
- [Kr02] Krueger, C.: Variation Management for Software Product Lines – A Case Study. Proceedings of the 2nd the Second Software Product Line Conference 2002.
- [LP09] Lauenroth, K.; Pohl, K.: Variabilität als eigenständige Sicht auf Software-Produktlinien. In: Produkt-Variabilität im gesamten Lebenszyklus, Workshop im Rahmen der Tagung Software Engineering 2009.
- [PBL05] Pohl, K.; Böckle, G.; van der Linden, F.: Software Product Line Engineering – Foundations, Principles, and Techniques. Springer, Heidelberg, 2005.
- [WL99] Weiss, D.; Lai, C.: Software Product-Line Engineering – A Family-Based Software Development Process. Addison-Wesley, Reading, 1999.