

Geleitwort¹

In den letzten zehn Jahren nahm das Interesse an Software-Produktlinien ständig zu. Seit 1999 sind auch deutsche Partner an einer Reihe europäischer Kooperationsprojekte beteiligt, die sich mit diesem Thema beschäftigen. Dies sind die ITEA-Projekte ESAPS, CAFÉ und zurzeit FAMILIES (siehe Anhang A.1). Die Herausgeber und die meisten Autoren dieses Buches sind oder waren an diesen Projekten beteiligt. In den letzten Jahren wird auch außerhalb dieser Projekte deutlich, dass vermehrt Firmen Produktlinienkonzepte in ihren Organisationen einführen. Das vorliegende Buch repräsentiert den Stand der Technik in Deutschland. Der größte Teil der Arbeiten stammt aus den oben erwähnten ITEA-Projekten, es sind aber auch weitere Ergebnisse eingeflossen.

Der Markt fordert immer mehr Funktionalität in neuen Produkten, die darüber hinaus auch noch höhere Qualität bezüglich Betriebssicherheit, Zugriffssicherheit, Zuverlässigkeit und Geschwindigkeit aufweisen sollen. Wenn man diesen Anforderungen genügen will, so muss man in immer kürzeren Entwicklungszyklen eine Vielzahl komplexer Produkte entwickeln und auf den Markt bringen. Das bedeutet für softwareproduzierende Firmen, dass sie ihre Effizienz verbessern müssen, denn die Mannigfaltigkeit dieser komplexen Produkte muss mit der gleichen Anzahl von Personen entwickelt werden. Durch den Software-Produktlinienansatz versuchen wir dies zu erreichen, indem gemeinsame Teile nur einmal entwickelt werden und notwendige Variation gezielt integriert wird; dies kann über die ursprünglich geplante Variation hinausgehen. Um eine Vielzahl unterschiedlicher Produkte in kurzer Zeit anbieten zu können, ist individualisierte Massenfertigung notwendig. Wir müssen den Schritt von individuell gefertigten zu vorgefertigten Produkten gehen, wobei wir die Vielfalt an den Kundenwünschen ausrichten. Eine Software-Produktlinie ist also definiert als eine Gruppe von Systemen, die eine gemeinsam verwaltete Menge von Merkmalen aufweisen, welche die grundlegenden Bedürfnisse einer wohlbestimmten Domäne

1. Übersetzung von den Herausgebern. Das engl. Original ist auf der Website des Buches (www.dpunkt.de/spl) zu finden.

befriedigen. Kurz gesagt, wir müssen Vielfalt planen, entwerfen, vermarkten, entwickeln und verkaufen. Das Ziel ist, viele Varianten mit wenig Aufwand in kurzer Zeit zu entwickeln.

Die wesentliche Idee hinter dem Software-Produktlinienansatz besteht darin, dass wir ein neues System oder eine neue Anwendung aus einer gemeinsamen Menge von Artefakten (Domänenmodell, Requirements, Referenzarchitektur, Komponenten etc.) entwickeln. Diese kommen zumeist aus früher entwickelten Systemen, die zur gleichen Produktlinie gehören, beschreiben also eine partielle Lösung.

Der Übergang zur Software-Produktlinienentwicklung bietet viele Vorteile; die meisten lassen sich darauf zurückführen, dass man relativ einfach neue Produktvarianten entwickeln kann. Ein wesentliches Hilfsmittel dafür ist die strategische Wiederverwendung, die es erlaubt, Softwareartefakte auf vordefinierte Art und Weise für vordefinierte Zwecke wiederzuverwenden. Dies führt zu Synergien beim Entwicklungsaufwand und tritt damit der wachsenden Systemkomplexität entgegen. So lässt sich die Rate der Produktinnovationen aufrechterhalten unter Gewährleistung der garantierten System-Performance und -qualität. Dies ermöglicht eine große kommerzielle Vielfalt bei relativ kleiner technischer Vielfalt. Damit lassen sich einerseits Marktnischen bedienen, andererseits wird die Entwicklung und Evolution von Produktvarianten für verschiedene Kundengruppen, wie z. B. für verschiedene Länder, unterstützt. Außerdem wird die Time-to-Market für individuelle Systemvarianten signifikant reduziert.

Der Software-Produktlinienansatz zahlt sich nur aus, wenn ein großer Anteil der zukünftig benötigten Variabilität vorhergesehen werden kann, sonst besteht die Gefahr, dass sich die zusätzlichen Investitionen zum Aufsetzen einer Produktlinie nicht auszahlen. Die Einführung einer Software-Produktlinienorganisation bedeutet die Einführung unterschiedlicher Verantwortlichkeiten in der Organisation, die zusammenarbeiten müssen und die geleitet und finanziert werden müssen. Beispiele dafür sind die Bereitstellung der wiederverwendbaren Produktlinienartefakte und die Nutzung dieser Artefakte bei der Systementwicklung. Bei alledem muss die Evolution sowohl der Artefakte wie auch der Anforderungen an die einzelnen Produkte berücksichtigt und organisiert werden.

Dieses Buch präsentiert die wesentlichen Beiträge der deutschen Partner bei der Entwicklung und Einführung von Software-Produktlinien in die industrielle Praxis. Es wurden Techniken eingesetzt, die beim traditionellen Software Engineering weniger wichtig waren, aber durch Verbesserungen speziell der Software-Produktlinienentwicklung dienen. Besondere Bedeutung haben die Einführung und die Evolution von Produktlinien: Fast nie beginnt die Software-Produktlinienentwicklung bei null, fast immer ist die betroffene Firma schon im Geschäft und hat bereits viele Produkte und Artefakte, die in die Familie aufgenommen werden sollten. Aber während eines solchen Übergangs zur Produktlinienentwicklung muss die normale Produktentwicklung weitergehen und die Kunden

müssen versorgt werden. Die Erträge aus dem Verkauf der aktuellen Systeme und der gerade entwickelten Systeme werden benötigt, um den Übergang zur Software-Produktlinienentwicklung zu finanzieren. Und wenn eine Produktlinie einmal existiert, so gibt es viele Kräfte, die deren Evolution beeinflussen und die geführt werden müssen. Ein eigener Teil des Buches beschäftigt sich mit diesen Problemen, die sich während der gesamten Lebensdauer einer Produktlinie immer weiter fortsetzen.

Das Buch stellt das Know-how zusammen, das bei den deutschen Projektpartnern in den letzten vier Jahren in Zusammenarbeit mit den internationalen Projektpartnern entstanden ist. Dabei haben alle Beteiligten von den Erfahrungen der Praktiker mit den Problemen bei der Produktlinienentwicklung und dem Umgang mit diesen Problemen gelernt. Das Buch spricht die wichtigsten Themen der Software-Produktlinienentwicklung an. Einer Organisation, die sich zur Produktlinienentwicklung entschließt, steht harte Arbeit, aber auch großer Nutzen bevor. Dabei kann sie von dem vorliegenden Buch profitieren, sowohl was die Identifizierung der betreffenden Probleme betrifft als auch was zu deren Lösung auf jeden Fall getan werden muss.

Frank van der Linden

Philips Medical Systems

März 2004

Preface

Since 10 years, software product-lines get a lot of interest. Since 1999 German partners are involved in a series of European co-operation projects focussed on this topic, the ITEA projects ESAPS, CAFÉ, and presently FAMILIES. The editors of this book, and also many of the chapter authors, are involved in this series. In recent years we see that also outside the series of projects companies start to introduce product line thinking in their organisations. This book presents the state-of-the-art results of work done in Germany. A large part of the work is originating from the above-mentioned ITEA projects, but also other experiences are already covered.

The market asks for more functionality in systems, which should have a higher quality in safety, security, reliability, and speed. To serve these requests a variety of complex products has to be produced, and they have to be marketed. This has to be done in short time scales. The companies need to improve their efficiency: The diversity of complex products has to be produced with the same number of people. The software product-line approach aims at producing the common parts only once and provide variation in a managed way. Even if the variation is initially not accounted for. In order to provide a large amount of diverse products in a short time, mass customisation is needed. We have to move from custom-made to ready-made, and keep the diversity according to customers' requests. A software product-line is defined to serve these requests, as a group of systems sharing a common, managed set of features that satisfy core needs of a scoped domain. In short, we have to plan, design, market, develop and sell diversity. This is the main objective of software product-line development. Produce many variants in a short time.

The idea behind a software product-line approach is to build a new system or application from a common set of assets (domain model, reference architecture, components, platform) defined from earlier developed systems belonging to the same line. A software asset is a description of a partial solution. It might be a component, known requirements or design elements that an engineer uses to build or modify a software product.

There are many benefits for adopting a software product-line; most are related to the relative ease of producing variant products. A major tool is strategic reuse, which allows to reuse software artefacts in predefined ways for predefined purposes. This allows sharing of development effort within the software product-line and as such counters the impact of growing system complexity. This makes it possible to sustain the rate of product innovation, while keeping guaranteed levels of overall system performance and quality. It allows a large commercial diversity with a relative small technical diversity. For instance niche markets might be served. It supports the development and evolution of product variants for different customer groups, e.g. for different countries. It reduces significantly the time to market for individual system variants.

The software product-line approach only pays when there is a large amount of variability foreseen. Only then the additional investments of setting up a product-line organisation may benefit. Introducing a software product-line organisation involves

the introduction of distinct responsibilities in the organisation, which have to cooperate, be managed, and financed. The first responsibility involves the preparation of the software product-line assets to be reused. The second responsibility involves the actual use of the product-line assets in the production of systems. An additional complication is that evolution of both the product requirements and the product-line assets have to be managed. The organisation should be able to determine impacts of change and how changes propagate. Change to a product concerns modification related to defects or new requirements. It is then necessary to analyse the affected assets and determine whether changes have impact on product-line assets. It is also necessary to keep track of the proportion of assets that are faulty. In general, this enables us to predict the properties of variants or new reference assets before actually building them.

This book covers the main contributions of German project partners in the introduction of software product-lines. Techniques, which were less important in traditional software engineering were introduced and improved to serve product-line development. An important emphasis is in the introduction and evolution of software product-lines. Almost never the software product-line development starts from scratch. In almost all cases the company is already in business and has a lot of products and other assets, which should be incorporated in the family. In the mean time the shop has to be kept open. Income from present systems, and systems already in development, is needed to finance the transition towards software product-line development. When a software product-line development is existing, many forces will lead to evolution, which has to be managed. A special part of this book deals with the problems related to both changes, which continue indefinitely.

This book consolidates the knowledge obtained in Germany, in collaboration with other European partners over the last 4 years. The leader will learn from the experiences of practitioners, in the problems of software product-line development, and how they were solved within the participating companies. It focuses itself on the major topics related to software product-line development. Many software engineering topics, which are not special to product-line development are not considered, or only briefly. Of course an organisation that wants to adopt software product-line development, has to work hard. In the meantime they gain a lot from this book, both on learning how certain problems can be solved, and in what should at least be done.

March 2004
Frank van der Linden
Philips Medical Systems