Variation Mechanisms and Multi-view Architecting in Platform-based Product Family Development

Wijnstra, Jan Gerben

Rijksuniversiteit Groningen

VARIATION MECHANISMS AND MULTI-VIEW ARCHITECTING IN PLATFORM-BASED PRODUCT FAMILY DEVELOPMENT

Proefschrift

ter verkrijging van het doctoraat in de
Wiskunde en Natuurwetenschappen
aan de Rijksuniversiteit Groningen
op gezag van de
Rector Magnificus, dr. F. Zwarts,
in het openbaar te verdedigen op
vrijdag 3 december 2004
om 13.15 uur

door

Jan Gerben Wijnstra
geboren op 24 september 1969
te Dantumadeel

Promotor:	Prof. dr. ir. J. Bosch
Beoordelingscommissie:	Prof. dr. S. Demeyer Prof. dr. ir. P.B. Kruchten Prof. dr. K. Pohl

ISBN 90-74445-63-2

Abstract

Over the past decades, the size and complexity of software within products has grown rapidly. Furthermore, the variation in products has increased. One way to cope with these trends is to develop products within the context of a product family. Product family development introduces a structured approach for reusing software across the products within the family. Product families are gaining in importance in the software industry.

This thesis presents the work we have performed at Philips Research using industrial cases. It consists of three parts, each addressing an area of research related to product family development.

In the first part we deal with the question how variation can be supported in a product family. We describe two methods for dealing with variation. The first method uses component frameworks with plug-in components. The component frameworks capture the generic functionality and form a stable part of the software architecture. The plug-in components provide the specific functionality that is needed to make a specific product. We focus especially on component frameworks, where the functionality is defined in the form of services. We describe a second method that is used for the development of a platform with reusable components. For a specific product components can be reused from this platform. An important element of this method is the explicit handling of interfaces, allowing the exchange of components providing the same interface. Another important element is the use of information models, which describe the data that are exchanged via the interfaces.

In the second part of this thesis we describe a method that contributes to the management of complexity, especially product families. This method uses design aspects, which are related to views that crosscut the main decomposition of a system, such as data logging or error handling. They provide a valuable structuring mechanism for the various phases of development; we will give various examples of this. Design aspects help to obtain components that can be easily reused. Since design aspects do not deal with domain specific functionality, the design aspects and their relation to the architecture requirements we describe in our study are broader applicable.

In the third part we deal with the question how a product family approach can be successfully introduced and maintained. In this part, we deal not only with technical issues, but also with business, process and organization issues. Based on a number of case studies, we identify important factors that must be considered to obtain successful product family development. Furthermore, we

introduce a classification scheme for product family approaches, which is based platform coverage and variation mechanisms. This classification scheme supports the selection of a suitable product family approach and can be used to evaluate and compare existing product families. We also discuss product family introduction strategies using the scheme.

Samenvatting

De laatste jaren is de omvang en complexiteit van software in producten snel gegroeid. Bovendien is de productvariatie toegenomen. Eén manier om met deze trends om te gaan is het ontwikkelen van producten in de context van een productfamilie. Productfamilie-ontwikkeling introduceert een gestructureerde aanpak voor het hergebruiken van software in de producten van de familie. Het belang van productfamilies groeit in de software industrie.

Dit proefschrift presenteert het werk dat we bij Philips Research hebben uitgevoerd, gebruik makende van studies van industriële systemen. Het bestaat uit drie delen, die elk een onderzoeksgebied adresseren gerelateerd aan productfamilie-ontwikkeling.

In het eerste deel behandelen we de vraag hoe variatie ondersteund kan worden binnen een productfamilie. We beschrijven twee methoden om met variatie om te gaan. De eerste methode maakt gebruik van raamwerkcomponenten en insteekcomponenten. De raamwerkcomponenten bevatten de generieke functionaliteit en vormen een stabiel onderdeel van de software architectuur. De insteekcomponenten bieden de specifieke functionaliteit die nodig is om specifieke producten te maken. Wij richten ons met name op service-raamwerkcomponenten, waarbij de functionaliteit is gedefinieerd in de vorm van services. We beschrijven een tweede methode die gebruikt wordt voor de ontwikkeling van een platform met herbruikbare componenten. Voor een specifiek product kunnen componenten worden hergebruikt uit dit platform. Een belangrijk element van deze methode is het expliciet omgaan met interfaces, waardoor het mogelijk wordt om componenten uit te wisselen die dezelfde interface aanbieden. Een ander belangrijk element is het gebruik van informatiemodellen, die de data beschrijven die via interfaces worden uitgewisseld.

In het tweede deel van dit proefschrift beschrijven we een methode die bijdraagt tot het beheersen van de complexiteit, met name in productfamilies. Deze methode maakt gebruik van ontwerpaspecten, die gerelateerd zijn aan gezichtspunten die de hoofdopdeling van een systeem doorsnijden, zoals het loggen van data of het afhandelen van fouten. Ze bieden een waardevol structureringsmechanisme voor de verschillende ontwikkelfases; we zullen hiervan verschillende voorbeelden tonen. Ontwerpaspecten dragen bij tot het verkrijgen van componenten die gemakkelijk hergebruikt kunnen worden. Aangezien ontwerpaspecten geen betrekking hebben op domeinspecifieke

functionaliteit, zijn de ontwerpaspecten en hun relatie tot de architectuureisen, die we beschrijven in onze studie, breder toepasbaar.

In het derde deel behandelen we de vraag hoe een productfamilie-aanpak op een succesvolle manier kan worden geïntroduceerd en onderhouden. In dit deel behandelen we niet alleen technische zaken, maar ook bedrijfs-, proces- en organisatiezaken. Gebaseerd op een aantal systeemstudies, identificeren we belangrijke factoren die in acht genomen moeten worden om te komen tot succesvolle productfamilie-ontwikkeling. Daarnaast introduceren we een classificatieschema voor productfamilie-aanpakken dat gebaseerd is op platformdekking en variatiemechanismen. Dit classificatieschema ondersteunt de keuze van een geschikte productfamilie-aanpak en kan gebruikt worden om bestaande productfamilies te evalueren en vergelijken. We bespreken ook introductiestrategieën voor productfamilies, gebruikmakende van dit schema.

Acknowledgements

During my work at Philips Research Laboratories I have had the opportunity to work in various projects related to software architecture and product families. These projects have been performed in cooperation with Philips Kommunikations Industrie (PKI) in Nuremberg (Germany) and Philips Medical Systems (PMS) in Best (Netherlands). Results of this work have been published in various papers on software architectures and product families. Six of these papers form the core of this thesis. I would like to thank my bosses at Philips Research, Jaap van der Heijden and Henk Obbink, for their support and for giving me the opportunity to work in these challenging projects. I would like to thank my bosses Ronald Begeer and Madhusudan Iyer at the Innovation Center Eindhoven of Philips Semiconductors for supporting the completion of this thesis.

I would like to thank my promotor Jan Bosch for his support and advice during my work on this thesis. After a number of discussions, we chose to base this thesis on six of the papers. His valuable feedback helped me to improve the thesis. It has been a pleasure working with him.

The work presented here is the result of a number of different projects. I am grateful to the various groups for their willingness to discuss their systems and architecting topics. The project that formed an important basis for this work is the Building Block project for the TSS (Telecommunication Switching System) group at PKI in Nuremberg. I would like especially to thank Lothar Baumbauer (main architect) and Jürgen Müller from this group. Other projects were performed at PMS in Best. Here, too, many people contributed to the various research projects. I would especially like to thank Jan Willem van den Beukel, Jan Willem Dijkstra, Luc Koch, René Krikhaar, Wim Pasman, Ben Pronk, Jacco Wesselius, Stef Zaicsek and Jan van Zoest for their valuable cooperation. These projects at the Product Divisions gave me the opportunity to work with large systems in practice, which is invaluable when you are working in the area of architecting and product families.

In the various projects I worked together with various colleagues from Philips Research. In this context I would especially like to thank René Krikhaar, with whom I spent about eight months in Nuremberg in the context of the Building Block project. I have fond memories of our stay in Nuremberg. I would also like to thank Frank van der Linden and Reinder Bril. In the projects for PMS, I also worked with various colleagues. I would like to thank Pierre America, Wim

Christis, Henk Obbink, André Postma, Eelco Rommes, Tobias Rötschke, Marc Stroucken and William van der Sterren for the enjoyable cooperation.

In addition to the projects for the Product Divisions, the composable architectures project at Philips Research played an important role. It is from here that the BAPO/CAFCR thinking model originated. Members of this project were Pierre America, Hans Jonkers, Frank van der Linden, Jürgen Müller, Gerrit Muller, Henk Obbink, Rob van Ommering and William van der Sterren.

The papers included in this thesis have been reviewed by various persons, including Pierre America, Lothar Baumbauer, Reinder Bril, René Krikhaar, Frank van der Linden, Gerrit Muller, Jürgen Müller, Rob van Ommering, André Postma, Ben Pronk, Eelco Rommes, Tobias Rötschke, William van der Sterren and Marc Stroucken. I would like to thank them for their critical and constructive comments. I would also like to thank Philips Translation Services for correcting the English grammar, and Henny Herps for the cover design.

I want to thank the reading committee for providing feedback and approving my thesis: Serge Demeyer (University of Antwerp, Belgium), Philippe Kruchten (University of British Columbia, Canada) and Klaus Pohl (University of Duisburg/Essen, Germany).

I want to express my gratitude to my parents Jan and Greta Wijnstra and my brother Folkert Jan for their support. Finally, I would like to thank my wife, Ingrid, for her love and encouragement.

Contents