Feature-Oriented Software Product Lines

Sven Apel · Don Batory
Christian Kästner · Gunter Saake

Feature-Oriented Software Product Lines

Concepts and Implementation

Springer

Sven Apel
University of Passau
Passau
Germany

Christian Kästner
Carnegie Mellon University
Pittsburgh, PA
USA

Don Batory
The University of Texas
Austin, TX
USA

Gunter Saake
Otto von Guericke University
Magdeburg
Germany

ISBN 978-3-642-37520-0 ISBN 978-3-642-37521-7 (eBook)
DOI 10.1007/978-3-642-37521-7
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013935777

ACM Computing Classification (1998): D.2, K.6

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Foreword

Features are a fundamental notion in modern software engineering. Defined once by Pamela Zave as "incremental units of functionality", they are central to how software is developed now. Much of today's software is developed using scenario-driven approaches. In essence, scenarios, also known as use cases or user stories, are specifications of features.

Feature-oriented software development shines in the context of software product lines. Virtually any successful software faces the need to cater different feature combinations to different customer segments. Product line engineering accelerates product development by leveraging the commonalities among the product line members, while managing the differences, also known as variabilities, among them. Features play a key role in modeling commonalities and variabilities and in managing the development of product lines. Major organizations, including General Motors and Danfoss, use feature-oriented approaches to successfully develop complex software-intensive product lines.

The message of this book is that much of the tremendous power of features is yet to be unlocked by *making features explicit throughout the entire systems and software lifecycle*. The explicit treatment of features in requirements, architecture, implementation, and verification and validation can greatly improve the management of software. Features are abstractions that can be made understandable to all stakeholders, both technical and nontechnical, enabling effective communication among the stakeholders and planning, implementation, and evolution of complex software product lines. Many of the ideas and tools presented in this book are applicable not only to traditional software product lines, but also to a wide range of variability-intensive systems, including highly-configurable applications, computing platforms, and software ecosystems.

The book provides a systematic introduction to feature-oriented software product lines, and leads the reader to more advanced topics in its second half. The authors distill the concepts and principles underlying the field with remarkable clarity, providing a much-needed foundation for the field. They also illustrate these

concepts and principles using concrete examples, showcasing languages, tools, and systems from both industrial practice and latest research. The advanced part of the book covers recent research results, many of which the authors have helped to advance. The reader can also enhance his or her learning experience by completing the provided exercises. The book will make an excellent upper-year undergraduate or introductory graduate text; but also practitioners will find it invaluable to enhance their software engineering toolbox with the powerful concepts and techniques of feature-oriented software product lines.

There is no better team than these four authors to write about feature-oriented software product lines. The authors have made fundamental scientific and engineering contributions to the field. Don has pioneered feature-oriented composition of software with his work on GENESIS, an extensible database management system, in late 1980s, and generalized the concepts and principles underlying it in early 1990s. He has continued on this path, advancing the theory and designing languages and tools, but also inspiring generations of researchers to join the effort. I started working in the field after attending Don's tutorial on "Software Systems Generators, Architecture, and Reuse" at the International Conference of Software Reuse in 1996. Shortly after, I shared my excitement for Don's ideas on software generation with Ulrich Eisenecker, which led to our work on automating component assembly based on feature models and, eventually, the book on Generative Programming in 2000. The subsequent decade has seen tremendous progress. New generations of young researchers have worked on techniques for feature-oriented modularization, variability-aware analyses, and empirical studies of systems with variability. Sven and Christian, enjoying the creative and fertile environment of Gunter's research group in Magdeburg and inspired by Don's work, have played leading roles in this progress. Their research results on feature-oriented software product lines have reached wide audiences at major software engineering conferences, such as the International Conference on Software Engineering and the Conference on Foundations of Software Engineering. Today, the four authors are central figures in the growing, vibrant community, known as Feature-Oriented Software Development (FOSD).

The notion of features has already profoundly affected how software is engineered, and this is just the beginning. Features can substantially improve the communication among all stakeholders and will likely lead to new, more effective ways to modularize and develop software. Despite the tremendous progress so far, much potential and many more discoveries lie ahead. Future work topics include finding most effective ways to exploit features in software modularization, creating techniques for re-engineering legacy towards feature orientation and evolving feature-oriented software, and also supporting features more pervasively in tools and infrastructures, such as in configuration management. Years to come will bring new, unexplored topics, which none of us can possibly foresee.

It is a great joy to see the new generations of brilliant young researchers joining the thriving FOSD community. I invite you to join this exciting ride, too. This book is your ticket!

Waterloo, April 2013 Krzysztof Czarnecki

Preface

The idea for this book arose from a series of lectures on modern programming paradigms, feature-oriented programming, and software product lines that are continuously held at the Universities of Magdeburg, Marburg, Passau, Texas at Austin, and others. Our collaboration reaches back to 2006, when Sven and Christian visited Don's group in Austin. Don's lecture on feature-oriented programming was inspiration for the lecture series set up in 2007 at the Universities of Magdeburg and Passau, which is the basis for this book. In a joint effort, we developed and continuously refined the teaching material for the lectures since then, until the present day.

Our interest in this topic was always the developer's perspective of discussing implementation techniques that are suitable for constructing variable software. We would have preferred to use a textbook for our lectures from the shelf, but existing product-line textbooks said precious little on implementation techniques. Eventually, in 2011, the material was stable, such that the natural next step was to write a proper text on this topic, meant not only for our students, but for all students of computer science and related fields as well as researchers and practitioners interested in software product lines.

Writing a textbook is an enormous endeavor, and this book would not have been possible without the help of our colleagues, students, friends, and families. In particular, we thank Martin Kuhlemann, Jörg Liebig, Norbert Siegmund, and Thomas Thüm, who used and improved the teaching material that was the basis for this book. Furthermore, we thank David Broneske for his support in producing proper graphics for this book, Jörg Liebig, Sandro Schulze, Norbert Siegmund, and Thomas Thüm for feedback on selected chapters and the exercises proposed in the book, as well as pure-systems, Janet Siegmund, and Thomas Thüm for providing screenshots of the tools pure::variants, FeatureCommander, and FeatureIDE.

Besides feedback and support from colleagues, we acknowledge the financial support of the German Research Foundation and the National Science Foundation for a number of research projects related to the topics covered in this book (the NSF Science of Design projects: CCF-0438786 and CCF-0724979, the ERC

grant #203099, and the DFG projects: FAME-DBMS—Sa 465/32, Feature Foundation—AP 206/2, SafeSPL—AP 206/4, and Pythia—AP 206/5).

Finally, but not the least, we are grateful to our families and friends whose support was a necessary basis for the success of this endeavor.

Passau, February 2013 — Sven Apel
Austin — Don Batory
Pittsburgh — Christian Kästner
Magdeburg — Gunter Saake

Contents