

1. Record Nr.	UNINA9910739406603321
Titolo	Systems and software variability management : concepts, tools, and experiences // Rafael Capilla, Jan Bosch, Kyo-Chul Kang, editors
Pubbl/distr/stampa	New York ; ; London, : Springer, c2013
ISBN	3-642-36583-3
Edizione	[1st ed. 2013.]
Descrizione fisica	1 online resource (xiv, 317 pages) : illustrations (some color)
Collana	Gale eBooks
Altri autori (Persone)	CapillaRafael BoschJan KangKyo-Chul
Disciplina	005.1
Soggetti	Software product line engineering Computer science Software engineering
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Description based upon print version of record.
Nota di bibliografia	Includes bibliographical references at the end of each chapters and index.
Nota di contenuto	Part I Variability Management -- Software Product Line Engineering -- Variability Modeling -- Variability Scope -- Binding Time and Evolution -- Variability Implementation -- Variability Realization Techniques and Product Derivation -- Visualizing Software Variability -- Variability in the Software Product Line Lifecycle -- Part II Research and Commercial Tools -- COVAMOF -- PLUM - Product Line Unified Modeller Tool -- FaMa -- pure::variants -- Part III Industry Experiences -- Philips Healthcare Compositional Diversity Case -- Variability in Power Plan Control Software -- Second Generation Product Line Engineering - A Case Study at General Motors -- Part IV Emerging and Research Topics in Software Variability -- Dynamic Software Product Lines -- Variability in Autonomic Computing -- Variability in Web services -- Service-Oriented Product Lines -- Software Variability and Design decisions -- Variability and Aspect Orientation.
Sommario/riassunto	The success of product line engineering techniques in the last 15 years has popularized the use of software variability as a key modeling approach for describing the commonality and variability of systems at all stages of the software lifecycle. Software product lines enable a

family of products to share a common core platform, while allowing for product specific functionality being built on top of the platform. Many companies have exploited the concept of software product lines to increase the resources that focus on highly differentiating functionality and thus improve their competitiveness with higher quality and reusable products and decreasing the time-to-market condition. Many books on product line engineering either introduce specific product line techniques or include brief summaries of industrial cases. From these sources, it is difficult to gain a comprehensive understanding of the various dimensions and aspects of software variability. Here the editors address this gap by providing a comprehensive reference on the notion of variability modeling in the context of software product line engineering, presenting an overview of the techniques proposed for variability modeling and giving a detailed perspective on software variability management. Their book is organized in four main parts, which guide the reader through the various aspects and dimensions of software variability. Part 1 which is mostly written by the editors themselves introduces the major topics related to software variability modeling, thus providing a multi-faceted view of both technological and management issues. Next, part 2 of the book comprises four separate chapters dedicated to research and commercial tools. Part 3 then continues with the most practical viewpoint of the book presenting three different industry cases on how variability is managed in real industry projects. Finally, part 4 concludes the book and encompasses six different chapters on emerging research topics in software variability like e.g. service-oriented or dynamic software product lines, or variability and aspect orientation. Each chapter briefly summarizes "What you will learn in this chapter", so both expert and novice readers can easily locate the topics dealt with. Overall, the book captures the current state of the art and best practices, and indicates important open research challenges as well as possible pitfalls. Thus it serves as a reference for researchers and practitioners in software variability management, allowing them to develop the next set of solutions, techniques and methods in this complicated and yet fascinating field of software engineering.
