Software product line engineering has proven to be the methodology for developing a diversity of software products and software intensive systems at lower costs, in shorter time, and with higher quality. In this book, Pohl and his co-authors present a framework for software product line engineering which they have developed based on their academic as well as industrial experience gained in projects over the last eight years. They do not only detail the technical aspect of the development, but also an integrated view of the business, organisation and process aspects are given. In addition, they explicitly point out the key differences of software product line engineering compared to traditional single software system development, as the need for two distinct development processes for domain and application engineering respectively, or the need to define and manage variability.
Software. Product Lines (SPLs), an innovative approach in the. Software Engineering has changed many things in the industrial area. A product derived from a .. engineering. Therefore it can be said that FODA is a domain analysis technique. The main objective of feature-oriented domain analysis if to create a domain.


and maintain a software product line during the parallel evolvement of several products is proposed. this process is . line Engineering (spLE), resulting in agile software product line Engineering (apLE). In this paper, we ... ing: Foundations, principles and techniques. springer-verlag new York, Inc., secaucus,. nJ, usa, 2005.

28 Sep 2004 . related variability issues in product line engineering, an estab- . lines. It holds for every evolving software system. However, even for single system development, it is common that one product is released in different versions and, thus, one has to ... Engineering: Foundations, Principles and Techniques.

Abstract: Software product line engineering is an es- tablished methodology for fast and effective develop- ment of software-intensive systems and services. To reap maximum benefits from the methodology, busi- nesses typically need to implement coordinated changes in development methodologies, tools, product architec.

20 Mar 2015 . A software version management system is proposed to support product line engineering by supporting product line evolution, product derivation and change promulgation from core assets and . encompasses the disciplines and techniques of initiating, ... Line Engineering Foundations, Principles and.

25 Sep 2017 . Variation techniques provide the capability to develop and deliver product configurations to platforms or programs. [17]. Figure 4 continues the narrative with an excerpt of a Lockheed. Martin slide showing how they characterize their PLE approach for AEGIS. Figure 4: AEGIS as a product line. It has been.

Software product line engineering aims to reduce development time, effort, cost, .. to different product requirements are captured in core assets. Those reusable assets are created during domain engineering. During application engineering, products are either .. Line Engineering Foundations, Principles and.


6 Mar 2010 . tive SPL implementation techniques. 2. BACKGROUND. Before we begin with the variability analysis and the evalu- ation of the collected data, we introduce SPLs, the prepro- cessor cpp, and cpp's role in SPL development. 2.1 Software Product Lines. An important concept in SPL engineering is that of a.

Context: Software Product Line (SPL) engineering is a specific approach to software development that offers strategic .. methods, techniques, and approaches of software product
Software product line engineering and the assured benefits of engineering foundations, principles, and techniques was used for validation and found the.

scale have pushed engineers to develop Software Product Lines (SPLs). As a result, testing a product line is challenging due to the enormous size of the possible products. Existing techniques focus on testing based on the product line's characteristics of aspect-oriented modeling techniques for product lines. A pacemaker product line, extracted from the real industry case, serves as a running example. http://www.omg.org/. [34] Pohl, K., Böckle, G., and van der Linden, F. J. Software. Product Line Engineering: Foundations, Principles and Techniques. Springer.

ABSTRACT. Product Derivation is one of the central activities in Software Product Lines (SPL). One of the main challenges of the process of product derivation is dealing with complexity, which is caused by the large number of artifacts and dependencies between them. Another major challenge is maximizing development. A product line approach for the development of e-Learning applications to satisfy the variable needs of customers, improve software quality and... Line. Engineering: Foundations, Principles, and Techniques. Springer, 2005. [6]. Northrop, L.M., Clements, C.C., “A Framework for Software Product Line Practice.” Version. 5.0.


Authors: Klaus Pohl; Gunter Bckle; Frank van der Linden. CiteWeb id: 20100000112. CiteWeb score: 2879. Software product line engineering has proven to be the methodology for developing a diversity of software products and software intensive systems at lower costs, in shorter time, and with higher quality. In this book.

using cloud services becomes hard, and the software product lines (SPL) approach is potentially promising for. Keywords. Cloud Computing, software product lines, cloud platforms, services, Health Watcher system. 1... G., Linden, F. 2005. Software Product Line Engineering: Foundations, principles, and techniques. 7 Jun 2015. The wide adaptation of product line engineering in software industry has enabled cost effective development of high quality software for diverse market. However, there is still a lack of methodology merged with techniques and tool supports to establish a time and cost effective product configuration.


introduced a methodological foundation for modeling and developing variant-rich. SOA-solutions by incorporating the principles of Software Product Line Engineering (SPLE) into the SOA development life. customization. SPLE research has proposed numerous approaches and techniques for the efficient production of.
technique. Mutation evaluates the ability of the test suite to detect defective versions of the FM, called mutants. In particular, it has been shown that existing test configurations achieving the . Keywords: Software Product Lines, Test Configuration Generation, Search-Based Software Engineering, Mutation, Feature Models.

24 Jan 2014. ABSTRACT. Software product line adoption is a challenging task in software development organisations. There are some reports in the literature of how software product line engineering has been adopted in several companies using different variability management techniques and patterns. However, to.

9 Jul 2017. Software Product Line (SPL) engineering is "a paradigm to develop software, not line. Optional. The feature F may or may not be added in a product line. Or. One or more features can be selected in a product line. Alternative One and only one feature from a ...

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software Product Line Engineering (SPLE) promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”

Software product lines constitute a new paradigm where industrial production techniques are adapted and applied. Software product line engineering promotes the development and maintenance of artifacts that can be reused in families of related software-intensive systems. As product lines are not necessarily disjoint and the development of product line artifacts is a demanding task, utilization of “related” or “similar”
have developed based on their academic as well as industrial experience gained in projects over the last eight years. They do not only.

Software and Systems Product Line (SSPL) engineering and management creates, exploits, and manages a common platform to develop a family of products (e.g. software products, systems architectures) at lower cost, reduced time to market, ... Software Product Line Engineering: Foundations, Principles and Techniques.


of a software product family approach at. Siemens AG. Our focus is on products and projects with an essential weight on software. Siemens AG is one of the world's largest electrical engineering and electronics companies. Most of Siemens' approximately 45,000 researchers and developers are working on software projects.

process components associated with software product lines. BAPO is considered critical in its consideration of how products resulting from software product lines make a profit. Software engineering, business, management and organizational sciences provide foundations for the concept of software product line engineering.


general, the software product line engineering paradigm involves two processes: core asset development. the scope definition techniques, analyzing their activities, roles, guidelines, concepts, strong points and. customization, relationship between SPL and development agile principles and metrics. 3. Research.

PLeTs' components, which led us to redesign its Software Product. Line Architecture (SPLA). In this paper, we . as Component-based development and Software Product Lines. (SPL) [11]. Component-based development is ... Software Product Line. Engineering: Foundations, Principles and Techniques. Springer–Verlag.


27 Aug 2015 . Software product line engineering consists of building a common platform for a set of products dedicated for a. and also a design for a co-evolution analysis framework based on biological techniques. . cladistics biological technique which was extensively used in biology to understand the evolution of a.


Course title and credit points. The course is titled “Software Product Line Engineering” and awards 7,5 credit. A software product line (SPL) is a set of software products that share a

18 Mar 2013 . software product lines; variability management; harmonization & variabilization; requirements engineering; case study. DOI: 10.12821/ijispm010103 . Through case study research. [15], an empirical research technique, it is largely ensured that any results found are also of value in a practical context.


3 Aug 2005 . I. Software Product Line Engineering Are you interested in producing software products or software-intensive systems at lower costs, in shorter time, and with higher quality? If so, you are holding the right book in your hands. Software product line engineering has proven to be the methodology for Higher.

T8: Leveraging Model Driven Engineering in Software Product Line Architectures (half-day), Monday afternoon · T9: Creating . What processes, methods, and technologies support PLE and what properties are required for the architecture and components that form the foundation of the product line? And what are the traps.


As part of the research carried out by IBM Haifa Research Lab, we are creating a model-driven framework for the development lifecycle of software and systems product lines. Our goal is to come up with a well-defined methodology and support for end-to-end software and systems product line development. Embedded.


1 Jan 2011 . evaluation research is needed to provide a better foundation for SPL testing. . Overviews of challenges and techniques are included in several earlier papers, as well as a couple of brief reviews. .. Software product line engineering is a development paradigm based on common software platforms, which.

products a key-challenge is to ensure correctness and safety of most of these products (if not all) at a low cost. Software. Product Line [1], [2] (SPL) techniques (and tools) allow engineering such families of related products. However, they rarely focus on testing the SPL as a whole. A software product line is usually modeled.

With more than 100 examples and about 150 illustrations, the authors describe in detail the essential foundations, principles and techniques of software product line engineering. The
authors are professionals and researchers who significantly influenced the software product line engineering paradigm and successfully.


Prof. Dr. Ralf Hahn, h_da Fachbereich Informatik,. Software Product Line Engineering, WS 13/14. 5 . . somehow all the same . . but all a little different and there are many more… 1. Motivation.


Abstract—Managing variability in a software family is cru- cial to software product line engineering. The existing vari- ability management techniques, however do not particularly address database design in the context of information systems product lines. This paper presents a practical approach to handle variability in.

I. Software Product Line Engineering Are you interested in producing software products or software-intensive systems at lower costs, in shorter time, and with higher quality? If so, you are holding the right book in your hands. Software product line engineering has proven to be the methodology for Higher quality, lower.


This textbook addresses students, professionals, lecturers and researchers interested in software product line engineering. With more than 100 examples and about 150 illustrations, the authors describe in detail the essential foundations, principles and techniques of software product line engineering. The authors are.

Determining whether a set of features can be composed, or safe composition, is a hard problem in software product line engineering because the number of feature combina- tions can be exponential. We argue that synergies between current approaches to safe composition should be exploited and propose a combined.


learned from this exploratory study of definition of a MAS-PL. Keywords: Product line, multi-agent systems, object-oriented, aspect-oriented. 1 Introduction. Software engineering aims to
produce methods, techniques and tools to develop software systems with high levels of quality and productivity. Software reuse [1] is one of.

I. Software Product Line Engineering Are you interested in producing software products or software-intensive systems at lower costs, in shorter time, and... Software Product Line Engineering: Foundations, Principles and Techniques. Authors: Pohl, Klaus, Böckle, Günter, van der Linden, Frank J.


24 Jan 2014. ABSTRACT. Dynamic software product lines (DSPL) constitute a promising approach for developing highly-configurable, runtime-adaptive systems in a feature-oriented way. A DSPL integrates both variability in time and space in a unified conceptual framework. For this, domain features are equipped.

rikard.land@mdh.se. Abstract. Software product line engineering has emerged as one of the dominant paradigms for developing variety of software products based on a shared platform and shared software artifacts. An important and challenging type of software maintenance and evolution is how to cost-effectively manage variability of the software product line that differentiates member products within a product line must be described with precise meaning and visualized so easy to select. Moreover, it should be easy to manage. Variability description approaches can largely be divided into two approaches, compositional symbolic execution technique that works in concert with a feature dependence graph to extract the set of execution technique.


Keywords: Product Line Engineering, Model Checking, Variability, Domain Artifact Verification. I. INTRODUCTION. Model checking ([5]) is a technique for quality assurance that facilitates the verification of properties (typically specified in CTL) of a system (typically specified in a state-transition model). In software.


ABSTRACT. Software Products Lines (SPLs) are families of products sharing common assets representing code or functionalities of a software product. These assets are represented as features, usually organized into Feature Models (FMs) from which the user can configure software products. Generally, few features are.


In addition, he is/was a member of numerous program committees. Klaus Pohl is (co-)author of over 100 refereed publications and several textbooks including “Software Product Line Engineering: Foundations, Principles, and Techniques”, Springer 2005 and “Requirements Engineering: Grundlagen, Prinzipien, Techniken”.


and therefore it is possible to apply techniques based on intensive soft- ware reuse, such as software product line engineering (SPLE). Although there has been much research on software product line engineering in the last years, the definition of a software product line for the domain of geographic information systems has.


Product line approaches from engineering, software quality assurance, and project management perspectives. driven the definition of new techniques, tools, and notations. These approaches share some common . The Agile Manifesto inspired 12 principles for. Agile process [2]. Among these principles, the highest.

Software Product Line (SPL) engineering is concerned with systematically reusing development assets in an . The technique uses merge operations to compose FMs defined in different SPLs. In Section 2 we define key terms and motivate our work. In particular, we distinguish .. Foundations, Principles and Techniques.


Software product line (SPL) is a set of software systems that share a . in a feature model for a new software product configuration [Pohl et al., 2005]. ... Software product line engineering: Foundations, principles and techniques. Springer-Verlag. Seinturier, L., Merle, P., Rouvoy, R., Romero, D.and Schiavoni, V., and Stefani.,

semantic web approach to model and verify product line requirements. We use . Software product line (SPL) is a software intensive system sharing a common and managed set of features that satisfy the .. [2] Pohl, K., Böckle, G., Linden, F.J.v.d.: Software Product Line Engineering: Foundations, Principles and Techniques.


verification techniques are applied that allow the analysis of products to be relativized on the properties of their. Software product line engineering [23] aims at developing a set of systems variants with well-defined commonalities and. In Section 3, we describe the foundations of our compositional verification technique.