

Proposal for construction of an approach for defining feature models in Software Product Lines based on Collaborative Work

Jazmín Gómez^{1[0000-0001-5559-1854]} Vanessa Agredo Delgado^{1[0000-0003-0870-6895]} and Pablo H. Ruiz^{1[0000-0003-2098-2614]}

¹Corporación Universitaria Comfacauca - Unicomfacauca, Popayán, Colombia
{jgomez, vagredo, pruiiz}@unicomfacauca.edu.co

Abstract. The Software Product Lines (SPL) provide a software production approach that is based on planned reuse, which seeks to improve productivity, development times, commercialization, and increase the quality of software products. Features modeling is one of the most popular domain analysis techniques, and is one of the essential elements for the good definition of SPLs, since they allow analyzing the similarity and variability to develop core product assets highly reusable. But defining a features model is a complex activity, because it involves the participation and coordination of a group of people, who use different techniques to manage and elicit sources of information, aspects that, if not well coordinated and related, can lead to a poor definition of the model and therefore, these will be of little use. Consequently, this paper presents the initial progress of the research that has allowed the articulation and base a proposal that seeks to define an approach for the definition of the features models, which allows in an assisted and collaborative way its construction using the own elements of SPL's Engineering in combination with Collaboration Engineering.

Keywords: Software Product Lines, Collaborative Work, Features Model.

1 Problem statement

The Colombian software industry has problems in its production, which affect low levels of quality, innovation, development, and reuse, causing customers and end-users to disagree about the development of the project and the final product [1]. The academic community and companies have directed efforts to mitigate these problems, have sought to use, adapt and/or define production approaches that allow increasing the quality of the software and consequently improve competitiveness [2] [3]. The Software Product Lines (SPL) provide a software production approach that is based on planned reuse, which seeks to improve productivity, development times, commercialization, and increase the quality of software products [2] [3]. But the construction of an SPL is not trivial, because it requires specialized knowledge of the definition and use of its components, for this it is necessary to follow Domino Engineering (DI) and Application Engineering (AI). The DI is responsible for providing guidelines for the definition of

elements that make up the SPL. For its part, the IA uses the infrastructure defined in the DI to develop specific products [4]. Specifically, in the DI, activities are carried out to determine the features of the products by identifying common elements, optional features, points of variation, and variant features, which must be configured and selected depending on the product to be generated, this identification of features is materialized in a Features Model, which is a vital and necessary element for the definition of an SPL. This model captures the features of the software products that will make up the SPL. In addition, the relationships between the features are described and the set that is valid to be used in a given software product is defined [5]. But defining feature models is a complex, time-consuming activity due to the involvement of different people with different ideas and knowledge, where it is necessary to properly understand and interpret heterogeneous opinions of the model needs that should emerge from the modeling team. In addition, when identifying features, it is necessary to abstract and understand the domain knowledge of various experts, as well as to search for information in different sources such as: books, user manuals, design documents, and source programs [6]. All the aforementioned aspects must be combined to define the features model, which makes the activity of its definition complex, since mainly an adequate collaboration between people who use techniques to manage and elicit sources of information is required, the aspect that if they are not well related they can lead to a poor definition of the model and therefore that it will be of little use [6].

On the other hand, collaborative work requires that participants who carry out an activity share the process of creating knowledge [7] [8] which implies providing spaces for interaction between people to meet an objective based on the use of different elements of collaboration (roles, tasks, strategies, tools, metrics, etc.) that allow incorporating the perspectives of the participants and their teamwork [9]. According to the general problem and considering that in the features models construction it is necessary to collaborate effectively and that the adoption of collaborative work elements could support in the features models definition in SPL's, it is pertinent to ask *How to support the definition of feature models in product lines using elements of collaborative work?* Given that it is convenient to support the definition of these models so that it is of relevant knowledge in the strengthening of the SPL's. Consequently, this paper presents the initial progress of the research that has allowed the articulation and support of a proposal that seeks to define an approach for the definition of feature models, that allows in an assisted and collaborative way its construction using own elements of SPL's Engineering in combination with Collaboration Engineering.

The rest of this paper is structured as follows: Section 2 describes the rationale. Section 3 presents and structures the related works. Section 4 defines the objectives of the proposal. Section 5 describes the methodology to be used for the execution of the project, and finally, section 6 describes preliminary conclusions of the work

2 Justification

SPLs are a software development paradigm that seeks to create product families instead of creating individual products. These families group a set of products that share common features while having some variations from each other. SPLs promise benefits such as reuse, decreased time and development costs [10]. In the SPL definition, several elements are used, among them the features models [5]. This research work aims to define an approach to define feature models in SPLs that incorporates elements of collaborative work, providing the business community with an approach that supports the construction of these models that are fundamental to the success and incorporation of SPLs in the industry, as well as providing benefits for the software quality improvement. In addition, it is intended to offer the research community an approach that incorporates own elements of Product Line engineering in combination with Collaboration engineering to define in an assisted and collaborative way, features models. Finally, to the academic community, it is intended to provide an approach that is useful and appropriate to the context of software engineering laboratories so that it can be considered as part of the software development methodologies used in this field.

3 Related work

Approaches to defining feature models

The features model was first presented as part of the features-oriented domain analysis (FODA) [11], its main focus is the identification of outstanding or distinctive features of the software systems that are produced from the domain analysis. In [6] the concept of features is explained, the objectives of its modeling and it provides guidelines for the domain analysis that arise from the execution of various industrial cases. On the other hand, in [12] FODA integrates with reuse-driven software engineering work products and processes. In [13] A semi-automatic approach is proposed to build feature models based on grouping of requirements. For its part, in [14] an algorithmic and parameterizable approach is presented to compute a logical and appropriate hierarchy that includes feature groups, written feature attributes, domain values, and relationships between attributes. In [15] the application of the automated analysis of feature models by means of computer-assisted information extraction is proposed. On the other hand, in [16] a method is proposed, which uses a two-layer feature model: application and infrastructure.

Classification and implementation of features

In [17] a cardinality-based notation for feature modeling is proposed, integrating a number of existing extensions of previous approaches. Introduces the staged configuration concept, which can be accomplished through the gradual specialization of feature models. In [18] a proposal is made to model and reason in an SPL, using constraint programming where functional and extra-functional features are considering. And [5] introduces and emphasizes the configuration and implementation of features.

Tools for creating feature models

In [19] a dynamic web is presented for the visualization, creation, modification, elimination, the definition of relationships, and restrictions of the feature models. On the other hand, [20] presents a reasoning and configuration system for the feature models that allows configuring and reasoning about the SPLs. In [21] present an automated approach to build feature models, from publicly available product descriptions. In [22], an approach is presented for the specification of feature models encoded in a relational database that allows validating and derive specific products.

Collaborative work in SPL's

In [23] is an approach for the configuration of collaborative products is proposed that provides effective support to coordinate the decision making of the teamwork in the configuration of the SPL products. On the other hand, in [24] [25] and [26] collaborative processes are proposed, from their own perspective, to define the scope of the SPL's in accordance with some guidelines of Collaborative Engineering. For his part in [27] a proposal to create a World-Class Collaborative Software Product Line Engineering Laboratory is showed to support the maturation of SPLs.

Systematic literature review

The Systematic literature review shown in [28] seeks to identify the state of the art of the feature-oriented software development paradigm FOSD (Feature oriented software development). In addition, it organizes and structures the development process of the Product Line approach in terms of characteristics and artifacts

The previous works show different perspectives related to the features models and some approaches that use collaborative work in the framework of SPLs. These works have different visions and strategies of how to create, manipulate, and manage feature models and how to use collaboration engineering in SPLS, but none involve or rely on collaborative work for building feature models. Therefore, none use elements of collaboration engineering that specifically support the creation of feature models, this being the major contribution of this research work.

4 Objectives

General objective

To propose a feature model definition approach¹ in Software Product Lines that incorporates collaborative work elements.

Specific objectives

¹ Approach: For this work, an approach can be: techniques, processes, activities, guidelines, that support the feature models creation.

- To characterize from the available literature, techniques, processes, activities, guide-lines, methods, and models used in the definition of feature models.
- To identify the best elements of collaborative work that are appropriate to the SPL development environment.
- To construct a feature models definition approach that is supported by elements of collaborative work
- To validate the proposed approach in an academic or industrial context.

5 Methodology

This work will be carried out following the multi-cycle research-action methodology with bifurcation which defines three cycles: conceptual, methodological and evaluation [29]. The cycles and their respective activities are briefly described below:

Conceptual cycle: planning and execution of systematic mapping are carried out to identify the features of the different approaches existing in the feature model creation. The activities related to the first and second goal are:

- Planning of systematic mapping. In this activity, the different research questions and objectives will be defined, inclusion and exclusion criteria will be defined, the databases and search engines to be used in the entire systematic mapping will be identified.
- Execution. In this activity, the search terms will be defined, scientific databases will be searched and relevant content and data will be extracted
- Analysis. In this activity, information is collected to identify the characteristics of the different approaches in modeling SPL characteristics.
- Identify, according to the literature, the collaborative work practices that can support the creation of a model of characteristics in SPL.
- Classify and/or define the collaborative work elements to be integrated into the definition of feature models in SPL.

Methodological cycle: the base structure of elements that will be part of the proposed approach is defined. The activities related to the third goal are:

- Define the base structure of elements that will be part of the proposed approach.
- Incrementally incorporate each of the components of the proposed approach

Evaluation cycle: an evaluation mechanism will be chosen with which the proposed approach will be validated, where the validation must be planned, designed, and executed to refine the approach considering the analysis of the results obtained. The activities related to the fourth goal are:

- Select an evaluation mechanism with which the proposed approach will be validated
- Plan, Design, and Execute the validation of the proposed approach
- Refine the approach taking into account the analysis of the results obtained in its validation

6 Conclusions

This paper presents the initial progress of the research that has allowed to articulate and substantiate a proposal that seeks to define an approach for the feature models definition, which allows in an assisted and collaborative way its construction using own elements of SPL's Engineering in combination with the Collaboration's Engineering. In addition, it is identified that defining feature models is a complex activity, which is too time-consuming, due to the fact that diverse people with different ideas and knowledge intervene, where it is necessary to properly understand and interpret heterogeneous opinions that emerge from the modeling team. According to the related works analyzed, there are different perspectives in the literature concerning the feature models, but none, involves or is based on collaborative work, therefore, this aims to be the greatest contribution of this research work, seeking it should be noted that coordination and collaboration between people who share knowledge, techniques, and tools can improve the feature models definition, contributing to the strengthening of the SPLs definition.

References

1. Sindy, M., Santiago, A., & Robledo, J. El Crecimiento de la Industria del Software en Colombia: un Análisis Sistemático. *Revista EIA*, 12, 95-96, (2015)
2. Peñalvo, D. F. Capítulo 7. Ingeniería del Software. En *Ciencia de la Computación e Ingeniería Artificial* (págs. 277-388). Salamanca, España, (2018)
3. Apel, S., Batory, D., Kästner, C., & Saake, G. *Feature-Oriented Software Product Line*. New York: Springer, (2013).
4. Bécan, G., Behjati, R., & Gotlieb, A. Synthesis of Attributed Feature Models From Product Descriptions. *Proceedings of the 19th International Conference on Software Product Line - SPLC'15*, (2015)
5. Benavides, D., Trinidad, P., & Ruiz-Cortés, A. Automated Reasoning on Feature Models. Springer, (2005)
6. Camacho, O. M., Álvarez, R. F., & Collazos, C. A. Identifying Collaborative Aspects During Software Product Lines Scoping. *Proceedings of 23rd International Systems and Software Product Line Conference*, (2019)
7. Chen, K., Zhang, W., Zhao, H., & Mei, H. An Approach to Constructing Feature Models Based on Requirements Clustering. *13th IEEE International Conference on Requirements Engineering*, (2005)
8. Clements, P., & Northrop, L. *Software Product Lines: Practices and Patterns*. Addison-Wesley. (2001)
9. Czarnecki, K., Helsen, S., & Eisenecker, U. Staged Configuration Using Feature Models. Springer, (2004).

10. Dillenbourg, P., Baker, M. J., Blaye, A., & O'Malley, C. The evolution of research on collaborative learning. Learning in humans and machine: Towards an interdisciplinary learning science, (1996)
11. Davril, J.-M., Delfosse, E., Hariri, N., Acher, M., Cleland-Huang, J., & Heymans, P. Feature Model Extraction from Large Collections of Informal Product Descriptions. *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*, 290-300, (2013)
12. Farahani, E. D., & Habibi, J. Feature model configuration based on two-layer modeling in Software Product Lines. *International Journal of Electrical and Computer Engineering (IJECE)*, 9(4), 2648-2658, (2019)
13. Galindo, J. A., Benavides, D., Trinidad, P., Gutiérrez-Fernández, A.-M., & Ruiz-Cortés, A. Automated analysis of featuremodels: Quo vadis? (2018)
14. Griss, M. L., Favaro, J., & d'Alessandro, M. Integrating Feature Modeling with the RSEB. *Proceedings. Fifth International Conference on Software Reuse*, 76-85, (1998)
15. Kan, K. C., Cohen, S. G., Hess, J. A., & Peterson, A. S.. *Feature Oriented Domain Analysis (FODA)*, (1990)
16. Kozar, O. Towards Better Group Work: Seeing the Difference between Cooperation and Collaboration. En *English Teaching Forum* (págs. 16-23), (2010)
17. Lee, K., Kang, K. C., & Lee, J. Concepts and Guidelines of Feature Modeling. En *Software Reuse: Methods, Techniques and Tools. Lecture Notes in Computer Science* (págs. 62-77). Springer, (2002)
18. Mendonca, M., Branco, M., & Cowan, D. S.P.L.O.T. - Software Product Lines Online Tools. *Proceeding of the 24th ACM SIGPLAN Conference Companion on Object Oriented Programming Systems Languages and Applications - OOPSLA '09*, (2009)
19. Mendonca, M., Bartolomei, T. T., & Cowan, D. (s.f.). Decision-Making Coordination in Collaborative Product Configuration. *Proceedings of the 2008 ACM Symposium on Applied Computing - SAC '08*, 108-113, (2008)
20. Noor, M. A., Grünbacher, P., & Briggs, R. O. A Collaborative Approach For Product Line Scoping: A Case Study In Collaboration Engineering. *IASTED International Multi-Conference SOFTWARE ENGINEERING*, 216-223, (2007)
21. Noor, M. A., Rabiser, R., & Grünbacher, P. Agile product line planning: A collaborative approach and a case study. *The Journal of Systems and Software*, 81(6), 868-882, (2008)
22. Northrop, L. M., Clements, P. C., Bachmann, F., Bergsey, J., Chastek, G., Cohen, S., . . . Jones, L. A Framework For Software Product Line Practice, Versión 5.0, (2012)
23. Oscar, D., & Salva, T. Lineas de Producto Software1. En *Fábricas de Software: experiencias, tecnologías y organización (2° edición)*. Ra-Ma, (2010)
24. Pino, F. J., Piattini, M., & Horta-Travassos. Managing and developing distributed research projects in software engineering by means of action engineering by means of action-research. *Revista Facultad de Ingeniería Universidad de Antioquia*(68), 61-74, (2013)
25. Roschelle, J., & Teasley, S. D. The Construction of Shared Knowledge in Collaborative Problem Solving. In: *Computer supported collaborative learning*. Berlin: Heidelberg:Springer, (1995)
26. Shatnawi, H., & Cunningham, H. C. Automated Analysis and Construction of Feature Models in a Relational Database Using Web Forms. *ACM Southeast Conference – ACMSE 2020*, 233-238, (2020)
27. Shatnawi, H., & Cunningham, H. C. Mapping SPL Feature Models to a Relational Database. *Proceedings of the SouthEast Conference on - ACM SE '17*, (2017)
28. Weiss, D. M., & Li, D. Workshop on Collaborative Software Product Line Engineering. *25th IEEE Conference on Software Engineering Education and Training*, 77-78, (2012)
29. Marx, K. La Neue Rheinische Zeitung. Organ der Demokratie (Nueva Gaceta del Rin. Órgano de la Democracia). 3, 4-26, (1849)

30. Neil, C., De Vincenzi, M., Battaglia, N., & Martínez, R. Herramientas Colaborativas Multiplataforma en la Enseñanza de la Ingeniería de Software. *XVIII Workshop de Investigadores en Ciencias de la Computación*, 899-904, (2016)
31. Ku, K., & Malaiyappan, P. M. A Systematic Literature Review on Feature Oriented Software Development Paradigm. *International Journal of Engineering and Technology*, (2017)