

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/327786002>

# Automated analysis of feature models: current state and practices

Conference Paper · September 2018

DOI: 10.1145/3233027.3233055

CITATIONS

3

READS

259

2 authors:



David Benavides

Universidad de Sevilla

129 PUBLICATIONS 4,373 CITATIONS

SEE PROFILE



José A. Galindo

Universidad de Sevilla

70 PUBLICATIONS 768 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



COPAS- [View project](#)



BELI: Technologies for Hybrid, Highly-Configurable and SLA-Aware Cloud Services [View project](#)

# Automated analysis of feature models. Current state and practices

David Benavides  
Dept. Lenguajes y Sistemas Informáticos  
University of Seville  
Seville, Spain  
benavides@us.es

José A. Galindo  
Dept. Lenguajes y Sistemas Informáticos  
University of Seville  
Seville, Spain  
jagalindo@us.es

## ABSTRACT

Software Product Lines (SPLs) are about developing a set of different software products that share some common functionality. Feature models are widely used to encode the common and variant parts of an SPL. The number of products encoded in a feature model grows with the number of features. Given  $n$  features and no constraints on valid feature combinations, there are  $2^n$  possible products. To deal with this complexity, automated mechanisms are used to extract information from feature models, such as features present in every product. A diversity of operations have been developed to model check, test, configure, debug, or compute relevant information by analyzing feature models. Moreover, such operations have been used in scenarios from different domains ranging from operating systems to video analysis optimization. In this tutorial, we go through the different automated analysis operations identifying its usage in the literature. Later present how to implement these operations within the FaMa framework.

## CCS CONCEPTS

• **Software and its engineering** → **Software product lines**; *Integrated and visual development environments*;

## KEYWORDS

SPL, Automated Analysis of Feature Models (AAFM), tool

### ACM Reference Format:

David Benavides and José A. Galindo. 2018. Automated analysis of feature models. Current state and practices. In *Proceedings of 22nd International Systems and Software Product Line Conference (SPLC '18)*. ACM, New York, NY, USA, 1 page. <https://doi.org/10.1145/3233027.3233055>

## 1 TOPIC

Variability-intensive systems (VIS) are those systems that need to cope with variability as part of its main functionality. SPLs are about developing a set of different software products that share some common functionality [4]. Feature Models (FMs) [6] have become the *de facto* standard to represent common and variable characteristics in variability-intensive systems.

---

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

*SPLC '18, September 10–14, 2018, Gothenburg, Sweden*

© 2018 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-6464-5/18/09.

<https://doi.org/10.1145/3233027.3233055>

Both graphical and textual notations have been proposed to represent FM and enable automated analysis support [2]. Currently, we can find many functionalities that have been benefiting from the formal structure of FMs. For instance, FMs can be used to facilitate the configuration process [7] and to analyze the system through automated analysis operations [2].

Concretely, there are more than thirty AAFM operations ranging from determining if a product is valid conforming to a feature model to seek for errors and bugs in a product line. Nowadays, we can find multiple tools that enable different SPL analyses in different contexts and domains such as cloud computing [5]. Some available tools are FaMa [3], FaMiLiAr [1] and FeatureIDE [8].

However, while these operations have been used in different domains and scenarios, users need to install, test and decide if the operation satisfies their requirements. Generally speaking, we find the following difficulties when using AAFM: *i*) It is hard to find the most appropriated operations to solve a problem. *ii*) It is required to install and configure an analysis tool; *iii*) Depending on the case, practitioners would need to adapt or extend and existing tool suite. *iv*) and; We need to integrate the tooling with the provided solution.

This tutorial can help to understand the catalog of available AAFM operations to users and practitioners. First we go through the different operations available in the literature and show where have been applied. Secondly, we analyze the current state of AAFM by identifying trends and research gaps. Then, we see how we can rely on an existing AAFM ecosystem for speeding up the development of new tools and operations.

## REFERENCES

- [1] Mathieu Acher, Philippe Collet, Philippe Lahire, and Robert B. France. 2013. FAMILIAR: A domain-specific language for large scale management of feature models. *Science of Computer Programming (SCP)* 78, 6 (2013), 657–681.
- [2] D. Benavides, S. Segura, and A. Ruiz-Cortés. 2010. Automated analysis of feature models 20 years later. *Information Systems* 35, 6 (2010), 615–636.
- [3] David Benavides, Pablo Trinidad, Antonio Ruiz Cortés, and Sergio Segura. 2013. *FaMa*. Springer Berlin Heidelberg, Chapter FaMa, 163–171. DOI : <http://dx.doi.org/10.1007/978-3-642-36583-6-11>
- [4] Paul Clements and Linda Northrop. 2002. *Software product lines*. Addison-Wesley Boston.
- [5] Jesús García-Galán, Omer F. Rana, Pablo Trinidad, and Antonio Ruiz-Cortés. 2013. Migrating to the Cloud: a Software Product Line based analysis. In *3rd International Conference on Cloud Computing and Services Science (CLOSER'13)*.
- [6] Kyo C Kang, Sholom G Cohen, James A Hess, William E Novak, and A Spencer Peterson. 1990. *Feature-oriented domain analysis (FODA) feasibility study*. Technical Report. DTIC Document. <http://www.sei.cmu.edu/reports/90tr021.pdf>
- [7] Andreas Pleuss and Goetz Botterweck. 2012. Visualization of variability and configuration options. *International Journal on Software Tools for Technology Transfer* 14, 5 (2012), 497–510. DOI : <http://dx.doi.org/10.1007/s10009-012-0252-z>
- [8] Thomas Thüm, Christian Kästner, Fabian Benduhn, Jens Meinicke, Gunter Saake, and Thomas Leich. 2014. FeatureIDE: An extensible framework for feature-oriented software development. *Science of Computer Programming* 79 (2014), 70–85.