Extracting orthogonal variability models from Debian repositories

José A. Galindo
Department of Computer
Languages and Systems
University of Seville
Seville, Spain
jagalindo@us.es

Fabricia Roos-Frantz
Department of Computer
Languages and Systems
University of Seville
Seville, Spain
fabriciaroos@us.es

Jesús García-Galán
Department of Computer
Languages and Systems
University of Seville
Seville, Spain
jgalan@us.es

Antonio Ruiz-Cortés
Department of Computer
Languages and Systems
University of Seville
Seville, Spain
aruiz@us.es

Software Product-Line (SPL) engineering is about management of a set of similar software products from a set of reusable core assets [2]. Variability Models (VM) are used to describe the variability present in the different stages of an SPL. The automated analysis of variability models involves a large number of techniques and tools. Currently, there are a variety of commercial and open-source tools to automate the reasoning on VMs such as $FaMa$[1], $FaMa\text{-}OVM$[2], $pure::variants$[3], $SEGOS$[4] are some of them.

Although there are several kinds of variability models, the majority of the research works on analysis of these models has focused on Feature Models (FM) and Orthogonal Variability Models (OVM). The former focuses on the domain and variability present in an SPL and the latter in the variability.

A well known problem in SPL community is the lack of real models available to researchers [1]. There exists references to real VMs, but those models are not public, probably this is mainly caused by the apprehension of the industry to reveal their business models. Real models are needed for providing an idea of program's behaviour in a real scenario [5].

As other researchers[4], we looked for real variability models in the opensource community. We found that Debian based distributions define the variability model of their product line [3] in a description file. We propose using Debian repositories as a good source of realistic and complex OVM. These can be used as motivations input problems for those tools dealing with the automation of OVM models.

In OVM a variation point (VP) documents what can vary from one product to another, and a variant (V) documents how this variation point can vary. Every line of the Debian description file describes an attribute or a relation of a package. A package definition is composed of a set of attributes and their relations with other packages. In our work we describe a mapping to obtain an OVM from the the variability present in Debian based distributions. The models obtained using our proposal could have up to 28.000 vari-ability elements (Ubuntu 8.04) and more depending on the distribution used.

Materials. In www.isa.us.es/materials/fmsple you can find the OVM model (in FaMa-OVM format) of the Ubuntu 8.04 distribution, and also a eclipse workspace with the sources, libraries and the implementation of the algorithm.

Acknowledgements

1. REFERENCES

[1] Don Batory, David Benavides, and Antonio Ruiz-Cortes. Automated analysis of feature models: challenges ahead. *Commun. ACM*, 49(12):45–47, 2006.

[2] P. Clements and L. Northrop. *Software product lines*. Addison-Wesley, 2001.

[3] J.A. Galindo, D. Benavides, and S. Segura. Debian packages repositories as software product line models. towards automated analysis. In *Proceeding of the First International Workshop on Automated Configuration and Tailoring of Applications (ACOTA)*, 2010.

[4] Steven She, Rafael Lotufo, Thorsten Berger, Andrzej Wasowski, and Krzysztof Czarnecki. The variability model of the linux kernel. In *Fourth International Workshop on Variability Modelling of Software-intensive Systems (VAMOS'10)*, Linz, Austria, January 2010.

[5] S.Segura, D.Benavides, and A. Ruiz-Cortés. Functional testing of feature model analysis tools. a first step. In P.Knauber, A.Metzger, and J.McGregor, editors, *Proceedings of SPLiT 2008 - Fifth International Workshop on Software Product Lines Testing*, pages 36–39, Limerick, Ireland, September 2008.

---

[1] www.isa.us.es/fama

[2] www.isa.us.es/famaovm

[3] www.pure-systems.com

[4] www.software-productline.com/SEGOS-VM-Tool