Variability Extraction and Modeling for Product Variants

Lukas Linsbauer
Johannes Kepler University Linz
Linz, Austria
lukas.linsbauer@jku.at

Roberto Erick Lopez-Herrejon
École de Technologie Supérieure
Montreal, Canada
roberto.lopez@etsmtl.ca

Alexander Egyed
Johannes Kepler University Linz
Linz, Austria
alexander.egyed@jku.at

ABSTRACT

Fast changing hardware and software technologies in addition to larger and more specialized customer bases demand software tailored to meet very diverse requirements. Software development approaches that aim at capturing this diversity on a single consolidated platform (i.e. software product lines [5]) often require large upfront investments of time and money. Alternatively, companies resort to developing one variant of a software product at a time by reusing as much as possible from already existing product variants (i.e. clone-and-own [2]). However, identifying and extracting the parts to reuse is an error-prone and inefficient task.Hence, more disciplined and systematic approaches are needed to cope with the complexity of developing and maintaining sets of product variants. Such approaches require detailed information about the product variants, the features they provide and their relations.

Variability is the capacity of software to vary [6]. Its effective management requires *variability information* such as the set of possible product variants, the features they provide, how they are related, and how they are implemented. In this work we present an approach for extracting variability information from sets of related product variants. We compute traces from features and feature interactions to their implementation artifacts and dependencies between features. A *Trace* is a link between a source and a target artifact (in this context between features and artifacts) [1]. *Traceability* is the potential for traces to be established and used. Variability management is paramount for coping with scenarios where multiple product variants must be developed and maintained such as:

Supporting and Enhancing Clone-and-Own Reuse. Clone-and-own is a manual ad hoc software reuse approach where new product variants are created by reusing parts from already existing variants [2]. The parts to be reused must first be located in the existing variants, then extracted, merged, and completed to obtain the new working variant. This process is repeated for each new variant required. Variability information helps to locate reusable features and their implementing artifacts. It even makes it possible to partially automate reuse [3].

Reverse Engineering Software Product Lines (SPLs). An SPL consists of a set of related product variants that are distinguished by the set of features they provide [5]. In cases where the

existing variants into a fully integrated SPL platform.Variability information is not only highly useful but even necessary for reverse engineering sets of variants into an SPL [3].

Extending an SPL. In cases where an SPL already exists, the need of providing new features to meet new customer requirements may still arise. In such cases product lines have to be extended to provide such new features, a process that requires knowledge about already existing features and how they are related and implemented.

CCS CONCEPTS

• **Software and its engineering** → **Software product lines**; *Software reverse engineering*;

KEYWORDS

Feature, Trace, Dependency, Interaction, Variant, Variability

ORIGINAL JOURNAL PAPER

ACKNOWLEDGMENTS

REFERENCES

[1] Jane Cleland-Huang, Olly Gotel, and Andrea Zisman (Eds.). 2012. *Software and Systems Traceability.* Springer.
[2] Yael Dubinsky, Julia Rubin, Thorsten Berger, Slawomir Duszynski, Martin Becker, and Krzysztof Czarnecki. 2013. An Exploratory Study of Cloning in Industrial Software Product Lines. In *17th European Conference on Software Maintenance and Reengineering, CSMR 2013, Genova, Italy, March 5-8, 2013.* 25–34