

Feature-Based Determination of Product Line Asset Types: In-house, COTS, or Open Source?¹

Jaejoon Lee and Dirk Muthig

Fraunhofer Institute for Experimental Software Engineering (IESE),
Fraunhofer Platz 1, 67663 Kaiserslautern, Germany
{[jaejoon.lee](mailto:jaejoon.lee@iese.fraunhofer.de), [dirk.muthig](mailto:dirk.muthig@iese.fraunhofer.de)}@iese.fraunhofer.de

1 Introduction

One important activity in product line engineering is product line production planning [1,2], during which stakeholders of a product line determine what and how product line assets are developed and used for product development. Moreover, decisions on which assets should be developed as in-house assets or purchased as COTS are made.

Recently, there have been increasing interests in using open source software for product development [3,4]. As pointed out in [3], it seems reasonable to make some common components of a product line as open source software or acquire them from existing open source communities. However, it is still not clear what a “common” component means and how it can be identified. Suppose, for example, that a switching component for voice communications in a telephony product line is a common component and is required for every product of the product line. If the overall quality of a switching system mainly depends on the quality of the switching component, then it may be difficult to develop such components as open source software, as they may be developed based on lots of know-how of a company. Therefore, we need a systematic approach or guidelines that can be used to determine which product line assets to be developed as open source software.

In this position paper, we propose a feature-based approach to identifying product line assets and determining their development strategies during product line production planning. The approach is an extension of [5], and a feature model [6], which captures commonality and vari-

¹ This research is partially carried out in the Cluster of Excellence 'Dependable adaptive Systems and Mathematical Modeling' project, which is funded by the Program 'Wissen schafft Zukunft' of the Ministry of Science, Education, Research and Culture of Rhineland-Palatinate, Germany, AZ.: 15212-52 309-2/40 (30).

ability information of a product line, is used as primary input to the strategy selection.

2 Product Line Asset Type Determination

After features of a product line are identified, we group features into feature binding units, each of which includes features of the same binding time [5]. Then, we determine which features or feature binding units will be developed as core assets, product-specific assets, or open-source asset, or purchased as COTS. Therefore, for each feature or feature binding unit, its asset type (i.e., core asset, product-specific asset, open-source asset, or COTS) should be determined with consideration of the budget and time-to-market constraints and other business/technical considerations such as expected frequency of feature usage, estimated cost for development, availability of in-house expertise, and availability of open source software. (**Table 1** shows some of the identified product line assets of a Home Integration Systems (HIS) product line [2,5].)

Table 1 Identified Product Line Assets and Their Types

A representing name for a set of features with the same binding time	Constituent features	Frequency of feature usage	COTS availability / COTS price (which is compared to the estimated in-house development cost)	Open source software availability	Asset type
FIRE	Fire, Smoke, Smoke sensor, Sprinkler, ...	High	No / -	No	Core asset
FLOOD	Flood, Moisture, Moisture sensor, Alarm, ...	Medium	No / -	No	Core asset
MESSAGE	Message, Voice	Medium	Yes / Higher	No	Core asset
	Communication	Medium	Yes / Higher	Yes	Open-source asset
SECURITY	Security, Access-control, ...	Low	No / -	Yes	Product-specific asset
	Biometric	Low	Yes / Lower	No	COTS

For example, in the HIS product line, the *FIRE* feature binding unit has high frequency of usage (i.e., all products in the product line include it) and the estimated cost for development is low; the features of *FIRE* are identified as core assets. The *Communication* feature, however, has medium frequency of usage and is available as open source software; this feature is identified as an open-source asset, i.e., it will be acquired from an open source community when it is needed. For

another example, the *Biometric* feature, which is used to authenticate users, must be developed in a short period but in-house expertise and open-source software for the biometric technique is not available; COTS components will be purchased to realize this feature.

The considering factors (e.g., frequency of usage, etc.) and decision criteria for each type may vary from one organization to other. For example, if a company considers a feature as a ‘killing’ feature, then the company would not make the feature as open source software, even though similar features are available from an open source community. Also, some features may be open to an in-company-open-source community so that these features can be developed, improved, and shared by engineers belong to different departments/teams of the company.

3 Discussions

In this position paper, a feature-based approach to identifying product line assets and determining their development strategies during product line production planning is proposed. We claim that our approach can provide asset developers with an explicit way to identify core assets, and determine asset types with technical and business/management considerations. We believe that our approach makes it visible where to adapt open source software for product line asset development. We hope that this research will lead us to develop more detailed guidelines for open-source based development in the context of product line engineering.

4 References

1. Chastek, G., McGregor, J.D.: Guidelines for Developing a Product Line Production Plan, *Technical Report CMU/SEI-2002-TR-006*, Pittsburgh, PA, Software Engineering Institute, Carnegie Mellon University (2002)
2. Chastek, G., Donohoe, P., McGregor, J.D.: Product Line Production Planning for the Home Integration System Example, *Technical Note CMU/SEI-2002-TN-029*, Pittsburgh, PA, Software Engineering Institute, Carnegie Mellon University (2002)
3. Co-development using inner & Open source in Software Intensive products (COSI) project, <http://itea-cosi.org/modules/wikimod/index.php?page=WikiHome>
4. Free/Open Source Research Community, <http://opensource.mit.edu>
5. Lee, J, Kang, K., Kim, S. A Feature-Based Approach to Product Line Production Planning, SPLC2004, LNCS 3154, pp. 183-196, 2004

6. Lee, K., Kang, K., Lee, J. Concepts and Guidelines of Feature Modeling for Product Line Software Engineering. ICSR7, LNCS 2319. pp. 62-77, 2002