A system to manage Multi-domain Software Product line

Yacine Djebar
Computer Science Department
May 8,1945 University of Guelma
djebyass@hotmail.fr

Mohamed Tahar Kimour
Computer Science Department
Badji Mokhtar Annaba University
mtkimour@hotmail.fr

Abstract

The development with Software product Line (SPL) often concerns one domain. The representation of multi-domain software product line is mainly based on complex Feature models. The latter is used to represent multi-domain component commonalities and variabilities. However, the management of multi-domain feature models needs an important effort. In addition, it is not easy to have a visual representation of this type of models. Split of multi-domain SPL into two models, the first to formulate the business needs which leads to a specific domain SPL and the second is devoted to compose a product in terms of selected SPL features according to business needs can lend a hand to overcome the management difficulties of this Type of SPL. In this paper, we propose a system to manage the multi-domain SPL. It is composed of a business model to express the needs in terms of business area configuration and a feature model to identify a concerned SPL according to selected business area needs in terms of feature configurations.

Keywords - business area, replacement configuration, business configuration, software product line, multi-domain.

1. Introduction

This a One of the more mainstream definitions of software product line is given by [Nor01].The latter defines SPL as "a set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way». In order to well configured and derived products, SPL engineering offers mechanisms to manage the product families through their common and variable features in all of the development steps[San08],[Voe07],[Coh90].

The Main activities of SPL are: identification and management of variability, management of constraints and derivation of products. The foremost representation of variability that can be applied for the three activities of SPL is undoubtedly a feature Model (FM)[Cza00],[Rei07].The latter can be used at any SPL abstraction level to model documents, code and other SPL useful artifacts. In Addition, a number of development approaches reported to Model-Driven Engineering are based on feature models to represent requirements through the common and variable FM features.

However, the management of multi-domain feature models is becomes a challenge for developers because of its complexity. In addition , creating and maintaining such large Feature Models can be a very fastidious activity[Har08], [Alv12], [Thu09], [Ben08], [Per08], [Whi07].To that end , several approaches have proposed solutions .Some of them are based on codes and offer tools [Har08], [Alv12], [Thu09], [Ben08], [Per08], [Whi07]. Other approaches are based strictly on models [San08], [Voe07], [Coh90], [Hey07], [Ghe06] and others used Aspect-oriented modeling approach[Sam16], [Tru17], [Lie18], [Ros18], [Dam18], and at the end the techniques that employ refined processes [Ben08], [Per08], [Whi07], [Ape08], [Asp17], [Ace10], [Kha13]. However, there is no ideal solution that addresses all aspects of the complexity of the feature models [Dam18].
Given the difficulty of handling a complex FM of multi-domain SPL, instead of manipulate in one way this type of FM, we focuses on a system that allows to

In: Proceedings of the 3rd Edition of the International Conference on Advanced Aspects of Software Engineering (ICAASE18), Constantine, Algeria, 1,2-December-2018, published at http://ceur-ws.org

manipulate a multi-domain SPL FM through two ways: a business model to formulate product needs, a feature model to derivate a product in terms of features and for this FM.The remainder of the paper is structured as follows. Section 2 gives an overview of the system .in Section 3, we describe the elements of the system.

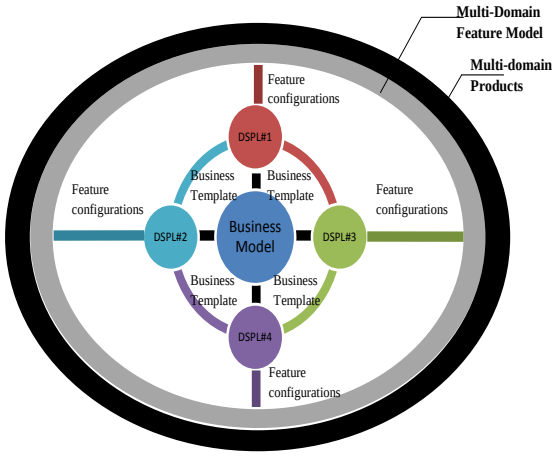

Figure 1: An overview of the system

linking the two models, a generic model of product configurations that allows to identify the domain and its SPL. Once, the SPL identified, the system manage the product configuration as an equivalent feature configuration in feature model. This business model expresses the needs of stakeholders in four business areas that are: category, profile, system and Article. Every area represents one or a set of business key characteristics of the targeted product. The system offers to stakeholder by a simple selection of a category of product to choose a single SPL domain and with more business areas to have a set of business configurations that correspond to products of a corresponding SPL domain products. The equivalent configuration is submitted to a stakeholder in order to be validated as If any equivalent configuration satisfies him; the system provides a replacement configuration .The latter is the closest in terms of matching features of the business configuration requested by the stakeholder.

The stakeholder can validate the replacement configuration if satisfied; else the integration of features without equivalence in the feature base becomes necessary. For this, our system allows to extend the Multi-domain FM by inserting features in

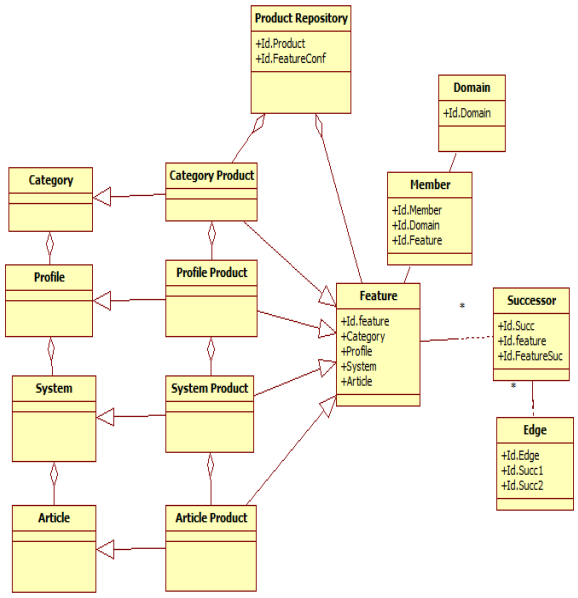

Figure 2: A Class diagram of our system

Section 4 details the replacement operation and the section 5 presents insert operation. Section 6 describes an illustrative iteration of a system. Section 7 presents an overview of the support-tool of our system and a section 8 concludes the paper and presents future work.

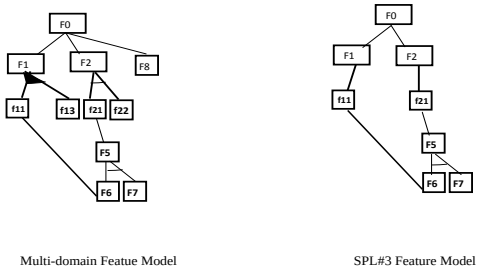

Figure 3: An Example of extraction of a Domain SPL FM from a multi-domain FM.

2 An overview of our system

A Multi-domain SPL can be perceived as a multi-Software product line of domains [Ros18],[Dam18].

Each domain is modeled by a specific SPL. So, the multi-domain FM can be perceived also as a set of domain Feature Models. Each domain feature model corresponds to a specific SPL feature model. The system as structured in Fig. 1 and modeled in Fig. 2 allows to extract a specific SPL FM from a multi-domain FM according to stakeholder business needs. The Fig. 3 illustrates this extraction. The targeted product of this type of domain must meet to business needs and at the same time to structural and architectural requirements. So, it can be described as a business area needs. So, the product is represented by unique business configuration of business areas. This configuration in terms of business areas has an equivalent configuration in terms of feature configuration. To meet this modeling, our system is structured in two models: a business model and a multi-domain feature model.

2.1 The Business Model

The selection of a category of product allows selecting a domain SPL. So, the corresponding SPL FM is

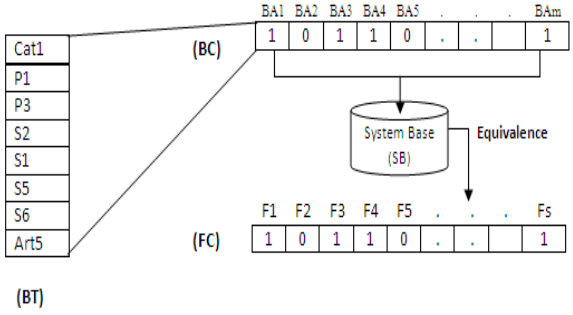

Figure 4: Relationship between (BT),(BC)and (FC)

extracted from a multi-domain FM and can generate a set of products according to the selected category .It means that all of these products have a same category in a business model.

The business model provides to stakeholder a second way to express his needs other than the only selection of FM components.

The business model is structured according to four business areas: the category, the profile, the system and the article (Fig. 2) .The stakeholder can select one business area or more by combining them .The aim is to formulate the business template (BT).

For each business area corresponds a number of business configurations (BC). The business configuration is composed of business features.

2.2 The feature Model

The multi-domain feature model defines the features, their successors, the edges and relationships between edges.

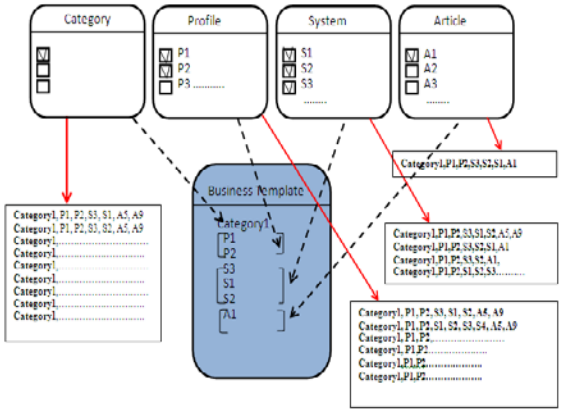

Figure 5:An overview of a business Template

For each element of business configuration correspond at most one feature in a domain SPL feature model. This correspondence is achieved through an equivalent feature configuration (FC) in a domain SPL feature model. Fig. 4 shows a relationship between (BT),(BC)

Table 1 : A structure of an Equivalent Feature configuration

F1	F2	F3	F4	F5	F6	….	F32	F33	….	Fn
1	**0**	**1**	**0**	**0**	**1**	**…**	**1**	**0**	**…**	**1**

and (EC).

The replacement configurations are proposed by the system to stakeholder in the case of a small 'No equivalence frame' without edge relationship impact. It means that the difference between (BC) and (FC) of a same (BT) only concerns the features .The relationships between edges in (BC) and (FC) are identical
If this configuration is not validated by the stakeholder, the No Equivalent Features in a system base must be inserted in the base.

The following pseudo algorithm summarizes the functioning of our system:

Running process Pseudo Algorithm

Begin.

1. Stakeholder select a business category in a business base (BB)
2. A system identifies the domain and its matching SPL (MSPL) in feature base (FB).
3. A system extracts a MSPL Feature Model from a feature base.
4. Stakeholder presents needs (select Profile and/or system and/or article) through Business Template (BT) from (BB).
5. System defines the Feature business Configuration (BC) of (BT).
6. System searches for the (BC) an equivalent Configuration (EC) in the Feature Base .
7. If (EC) found ((EC) = (BC)) Go to 10
8. If (EC) # (BC)
   5.1. System determines the Non Equivalent Frame (NEF) = ((BC)-(EC)).
   5.2. System proposes a replacement Configuration (RC) to (EC) from (NEF) and (FB).
   5.3. If (RC) accepted by stakeholder Go to 9.
   5.4. If (RC) Not accepted
   5.4.1 System Insert (NEF) in (FB).
   5.4.2 Go to 10.
9. System validates (RC) Go to 11.
10. System validates (EC) Go to 11.
11. (BT) satisfied.

End.

3 The System Elements

3.1 A Business Template

A Business Template (BT) is composed from a selection of one or more business areas according to stakeholder needs. For example the stakeholder can select only the category categ#1.In this case, the business template will be composed only by this category. If the selection of category "categ#1" is combined to "profile#2" profile, the system gives another BT composed by "categ#1" and "profile#2. The Fig. 5 illustrates an example of (BT).

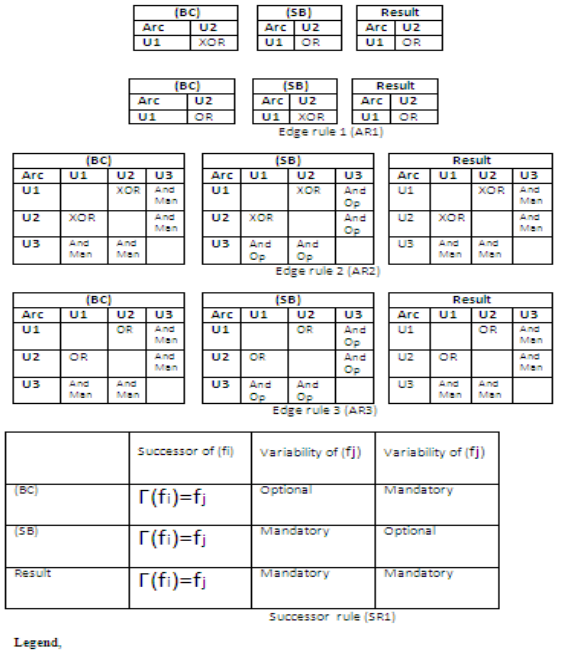

Figure 6 : The rules of Relationship predominance

3.1.1 A Business configuration

A Business Configuration (BC) is a Boolean expression of a business template in a business model.The (BC) structure is a Boolean array. Each cell of this array corresponds to a business area of selected (BT) elements .So; all of the business areas are represented in this array. The value '1' means that the corresponding area has been selected in the structure of the configuration and '0' means that the area is not selected. (BC) may be considered as a set of business features that have correspondence in a system base. Table 1shows an example of this modelling.

3.1.2 Equivalent Feature Configuration (FC)

Each (BC) element may have a representative feature in a system base (SB).it means that some ones may haven't. The set of the corresponding features in a system base forms an equivalent feature configuration (FC).

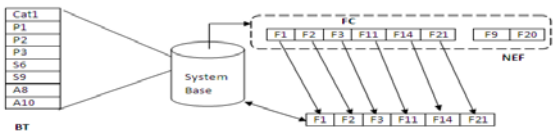

Figure 7: An example of NEF , (NEF=(F9,F10)

These configurations represent possible structures of products. Unlike the business configuration, the equivalent configuration is structured as a binary array as illustrated in Fig. 4. The Feature configuration is an expression of business configuration elements in a SPL domain feature model. The (FC) structure is a Boolean array. Each cell of this array corresponds to a feature of selected SPL Feature model .So; all of the FM features are represented in this array. The value '1' means that the corresponding feature has been selected in the structure of the product and '0' means that the feature is not selected. Table 1shows an example of this modeling.

The automatic generation of valid structures of (FC) may be achieved by one of the most techniques used in this context. However, the approach of [Dam18] that we've slightly adapted to our study is that we advocate for our system given the similarity of the structure of the configurations that are based on binary representations. The adaptation concerns only the pseudo-algorithm of generation of valid configurations. The latter is proposed as follows :

- Conf=0
- FC : Valid feature configuration
- I:=1
- CurrentDomain := domain#i
- CurrentFeatureModel:=FeatureModel(CurrentDomain)
- While (Conf<2^n) do

* n is a number of Current Feature Model (CFM) features

*2^{n-1} represents the number of possible configurations of products without considering of dependencies and different type of relationship (XOR, OR, Mandatory and Optional) between features.

- Generate binary representation of a Configuration

* Binary representation of a Configuration has a follow binary array structure:

1	1	0	1	0	1	0	
F1	F2	F3	F4	F5	F6	F7	...

F1, F2, F3..... represent features of (CFM)

- Select the corresponding features in (CFM)
- Compose (FC)
- Structural Verification
- If structure of (FC) valid then

*structure valid means the structure that consider of dependencies and relationships between each configuration representation features (FC).

- ProductRepository.Id.Feature.conf := structure of (FC)
- Endif
- Conf := Conf+1
- EndDo

3.1.3 The No Equivalence frame (NEF)

The BC elements that have not corresponding features in a SPL domain feature model (EC) forma no equivalence frame (NEF).The Fig. 4 shows an example

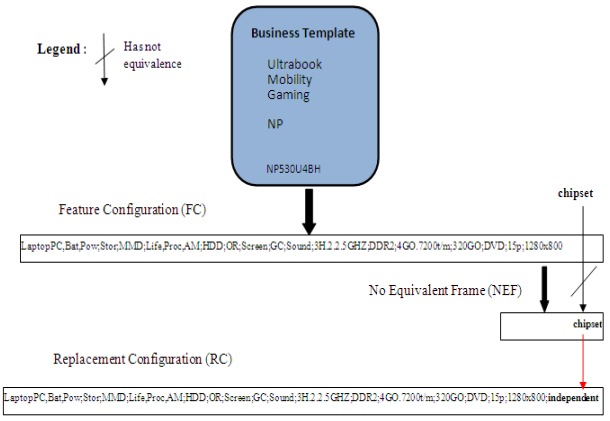

Figure 8: An example of a Replacement configuration of a NEF=("chipset").

of (NEF). The Fig. 7 shows an example of (NEF).

3.1.4 A system base

All of (FC) and validated (BC) configurations are stored in a system base. The feature model is mainly based on the following classes: product repository that contains all of configurations, feature, feature successors and edges through feature-successor-relationship .This organization combined to three insertion rules (Fig. 6) allows to system to identify rapidly the targeted feature nodes where the feature must be inserted or replaced in a SPL feature model graph.

4 The Replacement configuration

The replacement configurations are proposed by the system to stakeholder in the case of a small 'No equivalence frame' without edge relationship impact. It means that the difference between (BC) and (FC) of a same (BT) only concerns the features. Relationships between edges in (BC) and (FC) must be identical.

The stakeholder can validate the replacement proposal configuration. In this case, the system replaces (FC) by the valid (RC). The Fig. 8 shows an example of replacement configuration. We'll present hereinafter the pseudo-algorithm of the replacement proposal of (BC).

Pseudo Algorithm of replacement proposal of (BC)

Begin.

1. (NEF) = {Fk,Fl,...Fm}.
2. For each NEF(Fi) do
   - 2.1 (CF):=(Fi)
   - 2.2 Determine the parent P of (CF) in ((FB).Successor).
   - 2.3 Determine the set of (edges) U related to (CF) across P in ((FB).Edge).
   - 2.4 Determine the set of (successors) S of P in ((SB).Edge).
   - 2.5 For each (Sj) of (S) do
   - 2.5.1. Replace (Fi) with (Sj) in (FC) ((FC).Fi:=Sj)
   - 2.5.2. (RC):=(FC)
   - 2.5.3. If (RC) valid Go to 3
   - 2.5.4.Else
   - 2.5.4.1 j:=j+1
   - 2.5.4.2. Go to 2.5.1
   - Endif
2.6. i:=i+1
2.7.Go to 2.1
3. (BT) satisfied.

End.

5 The Insertion

The insertion of (NEF) elements in a system base is performed by the system in case of an important 'No equivalence frame' and/or NEF with an impact on relationships of edges. It means that relationships between (BC) edges and (FC) edges are different.

The system uses the three rules of insertion (see Figure 6) according to (BC) and (SB) features, feature-successors and edges.

There are two types of insertion rules: the simple insertion (successor rule) that implies one edge and two predominance cases and a complex insertion (edge rules) that implies many edges, many relationships and many predominance cases.

The system uses the three rules of insertion (see Figure 6) according to (BC) and (SB) features, feature-successors and edges . There are two types of insertion rules: the simple insertion (successor rule) that implies one edge and two predominance cases and a complex insertion (edge rules) that implies many edges, many relationships and many predominance cases. Figure 6 illustrates all of the predominance rules used by for insertion of (NEF) features by our system.

6 Illustrative Example

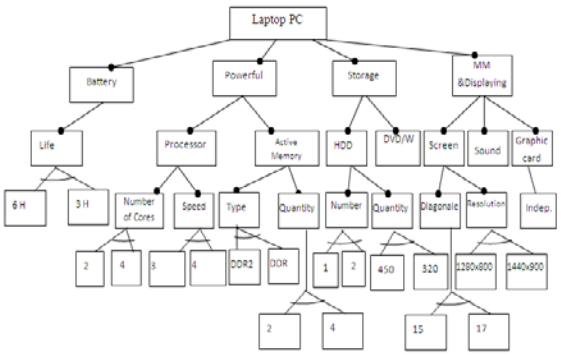

Figure 9: The feature Model of Samsung Laptop PC Domain.

Aiming to facilitate understanding different concepts of our system, we present an illustrative example of a Samsung PC multi-domain [Sam16] that we have adapted to our study. This system abbreviated « SMDSPL» allows to stakeholder to formulate his needs in a business template according to business areas. The Business areas are :

For Category :Essential , Ultrabook , MiniLaptop

The Domain of DesktopPC is represented by

For Profile : mobility , multimedia , gaming , versatility , fixed Office automation (FOA)

For System : NP , R , N
And for article : NP-NC10 ,RV-510 ,N100 , NP530U4BH, NP300V5AH , NP400U5AH
Multi-domain System contains three domains, each of them may be represented by a SPL FM. The domains of « SMDSPL» are as follows :
The Domain of LaptopPC is represented by SPL#1 .

The Domain of DesktopPC is represented by SPL#2 and the Domain of Hardware Server is represented by SPL#3.
The stakeholder selects the 'LaptopPC' category.The system select SPL#1and extracts automatically the corresponding SPL FM from the system base.The current SPL is SPL#1
Figure 9 presents a "Laptop PC" SPL FM.
The stakeholder can formulate the rest of (BT) according to the three remaining business areas

(profile, system and article)..The Fig. 10 illustrates the business model of Samsung multi domain .The Fig. 11 shows an example of composition of BT. Figure 8 illustrates an example of a replacement configuration.

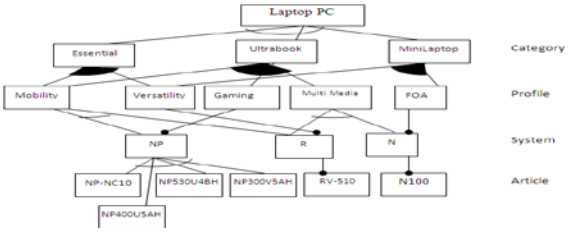

Figure 10: The business Model of Samsung Laptop PC

7 The Support Tool

We have implemented a tool called "MSPLSys" to support our system. The first version allows to create and update features, successors of features, edges, relationship between edges, business areas, business templates and dependencies between features. In addition, the tool allows to generate business

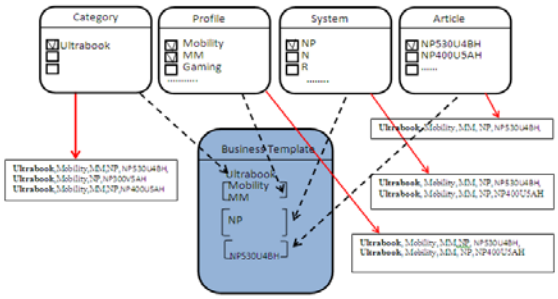

Figure 11: An example of Samsung Multi-domain template

configurations from a business template and system base. Figure 12 shows a replacement configuration report.

8 Conclusion

We have proposed in this paper, a new view of Multi-domain SPL. The latter can be perceived as a multi-Software product line of two views: a business and a feature views. Each of one is modeled by a specific SPLFM . So, instead of Multiple SPL , we use the multi-domain FM . Each domain feature model corresponds to a specific SPL feature model. The system as structured allows to extract a specific SPL

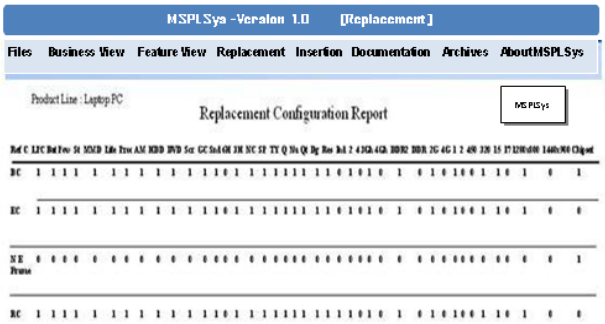

Figure 12 : MSPLSys Tool : A Replacement configuration Report

FM from a multi-domain FM according to stakeholder business needs. And the expected SPL product of this type of domain must meet to business needs and at the same time to structural and architectural needs .So; it can be described as a business area needs. This system employs two functions replacement and insertion to manipulate the multi-domain SPL. The system is powered by a tool that can support FM with a limited number of constraints. An extension can extend it to support a group of constraints between features.

References

[Ace10] A. Acher, M. Collet,P. Lahire and R. Robert. *Composing Feature Models*. France 2.1 University of Nice Sophia Antipolis, I3S Laboratory (CNRS UMR 6070), Sophia Antipolis Cedex, France, 2010.

[Alv12] T. Alves, V. Gheyi, R. Massoni,T. Kulesza, U. Borba and P. Lucena. *Refactoring product lines.* in GPCE'06: Proceedings of the 5th international conference on Generative programming & component engineering. ACM, NewYork,2012.

[Ape08] R. Apel, S. Lengauer,C. Moller and B. Kastner. *An algebra for features and feature composition*. In Meseguer, J., Roşu, G. (eds.)

AMAST 2008. LNCS, vol. 5140, pp. 36–50. Springer, Heidelberg ,2008.

[Asp17] Aspect-Oriented Modeling Workshop Series, http://www.aspect-modeling.org/., 2017.

[Ben08] S. Benavides, D. Ruiz-Cortes and A. Trinidad. *Automated merging of feature models using graph transformations*. in Generative and Transformational Techniques in Software Engineering II. LNCS, vol. 5235, pp. 489–505. Springer, Heidelberg ,2008.

[Coh90] K. Cohen, S. Hess, J. Novak and W. Peterson. *Feature-Oriented Domain Analysis (FODA) Feasibility Study*. Technical Report CMU/ SEI-90-TR-21, Software Engineering Institute,Nov, 1990.

[Cza00] R. Czarnecki and K. Eisenecker. *Generative Programming: Methods, Tools,and Applications* .Addison-Wesley Professional, Reading , 2000.

[Dam18] F.Damiani, R.Hähnle and E. Kamburjan, *Interoperability of Software Product Line Variants.* Proceeding of *splc2018* .Challenge Case1-1.pd , pp 99-07 , 09/*2018,* Gothenburg, Sweden , 2018.

[Ghe06] P. Gheyi, R. Massoni and T. Borba. *A theory for feature models in alloy*. in Proceedings of First Alloy Workshop, pp. 71–80, 2006.

[Har08] P. Hartmann and H. Trew. *Using feature diagrams with context variability tomodel multiple product lines for software supply chains.* in SPLC 2008: Proceedings of the 12th International Software , 2008.

[Hey07] P.Y.Heymans, P.Trigaux and J.C.Bontemps. *Generic semantics of feature diagrams*. Comput.Netw.51(2), 456–479 ,2007.

[Kha13] K. khalfelaoui, K. Chaoui, A. Foudil and C. Kerkouche. *Automatic Generation of SPL Structurally Valid Products using Graph Transformation Approach.* Dep of Computing Science, Univ of Jijel, MAAACA, v.488,pp.333–342.Springer,Switzerland, 2013.

[Lie18] M.Lienhardt, F.Damiani, S.Donetti and L. Paolini. *Multi Soft Product Lines in the Wild.* VAMOS 2018, Proceedding of the 12TH Inter.Workshop on Variability modelling of software-intense Systems – pp 89-96., 2018

[Nor01] P.Northorn.*Software Product Line :Practices and Patterns*. AddisonWesley Professional Reading, 2001.

[Per08] P.Perrouin,G. Klein,J.Guelfi and N Jezequel. *Reconciling automation and flexibility in product derivation*. In SPLC' 08: Proceedings of the 2008 12th International Software Product Line Conference, pp. 339–348. IEEE, Los Amitos, mexico, 2008.

[Rei07] M.Reiser and M.O.Weber. *Multi-level feature trees:a pragmatic approach to managing highly complex product families*. Requirements Engineering J,12,57–75 , 2007.

[Ros18] M.Rosenm and N.Siegmund .A*utomating the configuration of multi software product line.* School of Computer Science , University of Magdeburg , Germany @ovgu.de , 2018.

[Sam16] Samsung Laptop PC product catalog , http://www.samsung.com/n_africa/consumer/computers-peripherals/laptops ,2016.

[San08] P. Sanchez, P. Loughran, N. Fuentes, and L .*Engineering languages for specifying Product-Derivation processes in software product lines*. in Software Language Engineering journal,pp. 188–207 ,2008.

[Thu09] R.Thum, T.Batory and D.Kastner. *Reasoning about edits to feature models.* in Proceedings of the 31th International Conference on Software Engineering (ICSE IEEE), Los amitos,mexico,2009.

[Tru17] G.I,Trujillo and T.U. Juarez-Martíne . *Multiple SPL: applications and challenges.* division of Research and P.graduate Studies.Instituto Tech of Orizaba , Vera-cruz,Méxicodepi.edu.mx,aaguilar@itorizaba.edu.mx/kcortes@uv.mx OCT 2017.

[Voe07] R. Voelter and M. Groher. *Product line implementation using aspect-orientedand model-driven software development*. in SPLC'07:pp.233–242.,Los Alamitos ,2007.

[Whi07] P.K. Whittle, J. Elkhodary,A.M. and H. Gomaa. *Model compositionin product lines*

and feature interaction detection using critical pair analysis. In Engels, G., Opdyke, B., Schmidt, D.C., Weil, F.2007. LNCS,2007.