
Mémoire de master II

Tedjou jospin delmas

Sous la direction de Dr. Amougou Ngoumou.

Le 20 septembre 2017

Résumé

Mon résumé : Les lignes de produit logiciel...Les lignes de produit logiciel...Les lignes de produit logiciel...Les lignes

Abstract

My abstract: Software prodcut lines.....Software prodcut lines.....Software prodcut lines.....Software prodcut

Remerciements

Merci à Namrod pour toute la partie sur la bibliographie. Retrouvez ses questions FAQ qui ont permis larédaction de cette

À mon père, ma mère, mes frères zé mes sœurs, oho...

Table des matières

Résumé	ii
Remerciements	vi
Table des matières	vii
Sommaire	vii
Introduction	1
1 Revue de la littérature	3
1.1 Ingénierie de lignes de produits	3
1.1.1 Ingénierie de domaine	3
1.1.2 Ingénierie de l'applicaton	3
1.1.3 Variabilité et Feature Models	3
1.2 Les méthodes de lignes de produits	3
1.2.1 FAST	3
1.2.2 DARE	3
1.2.3 Kobra	3
1.2.4 koala	3
1.2.5 PLUS	3
1.2.6 FORM	3
2 La méthode FORM/BCS	4
2.1	4
3 Mise en place de l'éditeur de caractéristiques	5
Conclusion	6
Glossaire	7
Liste des abréviations, des sigles et des symboles	7
Bibliographie	8
Table des figures	9
Liste des tableaux	10

Introduction

contexte

De nos jours, les logiciels sont des outils indispensables dans de nombreux secteurs d'activité. On les utilise dans le domaine de la finance, des transports, la gestion de stocks, l'éducation et bien d'autres. Leur mise en place est toutefois très coûteuse en termes de budget, de ressources humaines et de temps de réalisation et sujette à des risques d'échec de projets. Il devient alors intéressant de produire des logiciels de meilleure qualité à faible coût et rapidement ; c'est le but principal de l'ingénierie de ligne de produits logiciels [1]. Elle se base sur le fait que les logiciels développés aujourd'hui ne sont pas nouveaux mais sont des variantes de systèmes déjà développés dans le même secteur d'activité [2]. Dans le domaine de la gestion académique par exemple, un système logiciel pour une nouvelle université est une variante des systèmes existants, déployés dans les autres universités et qui a ses propres spécificité. Il traitera comme les autres des étudiants, des enseignants, des matières, des notes mais peut avoir son propre système de notes différent de celui des autres logiciels du domaine. Un domaine peut être vu comme un secteur d'activité ou une zone de connaissance dirigé par un ensemble de besoins et caractérisé par des concepts et des terminologies compréhensibles par les acteurs de ce secteur[1].

Au lieu de construire des logiciels en partant de rien (from scratch), on voudrait désormais construire des familles de logiciels qui serviront de base au développement des logiciels du domaine. On exploite alors les connaissances accumulées dans le domaine concerné. Comme le souligne [5], plusieurs définitions sont utilisées dans la littérature pour qualifier les Lignes de produits mais celle de Clements et Northrop est très souvent citée dans les travaux du domaine. Selon elle, une famille de logiciels ou ligne de produit logiciel (LPL) est un ensemble de logiciels partageant des propriétés communes, satisfaisant des besoins spécifiques pour un domaine donné et développées de façon contrôlée à partir d'un ensemble commun d'artéfacts [4].

problème et approches connues

Le développement des lignes de produits se fait grâce à un paradigme appelé Ingénierie de ligne de produits logiciels. Pohl et al. définissent l'ingénierie des LPL comme un paradigme de développement de logiciels utilisant une plateforme logicielle commune et la personnalisation de masse [1]. L'ingénierie de ligne de produits est constituée de deux étapes principales : l'ingénierie de domaine et l'ingénierie applicative [1]. L'ingénierie de domaine analyse le domaine pour en cerner les propriétés communes aux applications du domaine (commonalities en anglais) et les propriétés variables (variabilities en anglais) afin de produire l'ensemble des artéfacts logiciels réutilisables. L'ingénierie applicative quant à elle permet de dériver les logiciels membres de la famille selon les besoins spécifiques en exploitant la variabilité. Ainsi, le développement d'un nouveau logiciel revient à analyser les besoins spécifiques, paramétrer la plateforme support en choisissant les composants adéquats et générer le code du logiciel. Ngassam insiste sur le fait que l'ingénierie de ligne de produit fait intervenir l'Ingénierie Dirigée par les Modèles(IDM)[3]. Plusieurs méthodes de développement de lignes de produits ont été proposées : FAST(Family oriented Abstraction, Specification and Translation), PuLSE (Product Line Software Engineering) ,ODM (Organization Domain Modeling), DARE (Domain Analysis and Reuse Environment), FODA

(Feature Oriented Domain Analysis), FORM (Feature Oriented Reuse Method) etc. La méthode FORM étend la méthode FODA basée sur les caractéristiques métiers (features) qui a été longuement exploitée dans les applications industrielles. FORM a été étendue à son tour pour entre autre élargir son application aux systèmes d'information par la méthode FORM/BCS (Feature Oriented Reuse Method with Business Component Semantics) qui requiert notre attention dans le cadre de ce travail.

Description sommaire de form et form/bcs

Parler de FORM : Détailler les features, process, sub-system et module

question de recherche(sous problème) + résultats attendus

Cependant, la méthode FORM/BCS ne dispose pas encore d'outils support pour permettre son exploitation dans la pratique. C'est pourquoi dans le cadre de ce travail, nous voulons contribuer à la mise en place d'une plateforme support pour la manipulation dynamique des lignes de produits de FORM/BCS. Nous nous attèlerons plus spécifiquement à la conception et à l'implémentation de l'architecture d'un éditeur de composants métier caractéristique à l'aide de l'outil EMF (Eclipse Modeling Framework).

méthodologie

Pour réaliser ce travail, nous devons tout d'abord comprendre la méthode FORM/BCS en étudiant les propriétés du modèle, puis étudier l'embryon de plateforme support actuelle de FORM/BCS et enfin étendre cet embryon en y ajoutant un éditeur de composants métier caractéristique.

plan

Ce travail est structuré comme suit. D'abord le chapitre 1 présente l'état de l'art des méthodes d'ingénierie de lignes de produits existantes avec leurs points forts et leurs limites. Ensuite le chapitre 2 explique la méthode FORM/BCS en détaillant les caractéristiques du modèle de FORM ainsi que les composants réutilisables de FORM/BCS. Puis, le chapitre 3 présente la mise en oeuvre que nous allons effectuer, il récapitule les étapes de la conception et de l'implémentation de l'éditeur de composant métier caractéristique de FORM/BCS.

Chapitre 1

Revue de la littérature

1.1 Ingénierie de lignes de produits

1.1.1 Ingénierie de domaine

1.1.2 Ingénierie de l'application

1.1.3 Variabilité et Feature Models

INTRODUCTION A LA VARIABILITE

1.2 Les méthodes de lignes de produits

1.2.1 FAST

1.2.2 DARE

1.2.3 Kobra

1.2.4 koala

1.2.5 PLUS

1.2.6 FORM

Chapitre 2

La méthode FORM/BCS

2.1

Chapitre 3

Mise en place de l'éditeur de caractéristiques

Conclusion

Bibliographie

- [1] Frank van der Linden KLAUS POHL Günter Böckle. *Software Product Line Engineering Foundations Principles and Techniques*. Berlin Heidelberg : Springer, 2005. 467 p.
- [2] Amougou Ngoumou MARCEL FOUDA NDJODO. « The feature Oriented Reuse Method with Business Component Semantics ». In : *International Journal of Computer Science and Applications* 4 (2009), p. 63–82.
- [3] Ngassam Seuwa Florie MERVEILLE. « Conception et réalisation d'un éditeur des composants métiers caractéristiques de la méthode FORM/BCS : REAL ». Thèse de doct. Université de Douala, 2017.
- [4] Linda Northrop PAUL CLEMENTS. *Software Product Lines : Practices and Patterns*. SEI Series in Software Engineering. Addison-Wesley, 2002.
- [5] Simon URLI. « Processus Flexible de Configuration pour Lignes de Produits Logiciels Complexes ». Thèse de doct. Université de Nice Sophia-Antipolis, 2015.

Table des figures

Liste des tableaux