



HAL
open science

Configuration guidée des lignes de produits basée sur les traces de processus

Houssem Chemingui

► To cite this version:

Houssem Chemingui. Configuration guidée des lignes de produits basée sur les traces de processus. Génie logiciel [cs.SE]. Université Paris 1 Panthéon-Sorbonne; Université de la Manouba (Tunisie), 2021. Français. NNT: . tel-03277036v1

HAL Id: tel-03277036

<https://hal.science/tel-03277036v1>

Submitted on 2 Jul 2021 (v1), last revised 21 Jan 2022 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**THÈSE DE DOCTORAT DE L'UNIVERSITÉ PARIS I – PANTHÉON
SORBONNE ET L'UNIVERSITÉ DE LA MANOUBA**

Spécialité : INFORMATIQUE

Présentée par

Houssem CHEMINGUI

Pour l'obtention du titre de

**DOCTEUR DE L'UNIVERSITÉ PARIS I - PANTHÉON
SORBONNE ET L'UNIVERSITÉ DE LA MANOUBA**

Sujet de la thèse :

**Configuration guidée des lignes de produits
basée sur les traces de processus**

Soutenue le 30 Juin 2021 devant le jury composé de :

Pr. Naoufel KRAIEM Professeur à l'ISI, Université de Tunis Manar, Tunisie.	Président
Pr. Saïd ASSAR Professeur à l'Institut Mines Télécom, France.	Rapporteur
Pr. Olfa BELKAHLA-DRISS Professeure à l'ESCT, Université de la Manouba, Tunisie.	Rapporteur
Pr. Bénédicte LE GRAND Professeure à l'Université Paris 1 Panthéon-Sorbonne, France.	Examineur
Pr. Camille SALINESI Professeur à l'Université Paris 1 Panthéon-Sorbonne, France.	Directeur (France)
Pr. Raul MAZO Professeur à l'ENSTA Bretagne, France.	Co-directeur (France)
Pr. Henda BEN GHEZALA Professeure à l'ENSI, Université de la Manouba, Tunisie.	Directeur (Tunisie)
MCF. Inès GAM Maître de conférences à l'ISI, Université de Tunis Manar, Tunisie.	Co-directeur (Tunisie)

[Cette page est intentionnellement laissée en blanc]

[Cette page est intentionnellement laissée en blanc]

RÉSUMÉ

Les lignes de produits présentent une stratégie de production qui offre aux compagnies un avantage concurrentiel en améliorant la productivité et en favorisant la personnalisation. Cependant, les avantages potentiels des lignes de produits peuvent être perdus lorsque leurs modèles de variabilité sont larges et complexes impliquant des centaines de variantes conduisant, parfois, à des milliards de combinaisons possibles. En effet, le processus de configuration devient une tâche sujette aux erreurs vu la pléthore des exigences et des contraintes à satisfaire en même temps. Par conséquent, guider les parties prenantes lors des processus de configuration complexes et recommander les meilleures alternatives de configuration de manière à conduire une expérience satisfaisante devient un défi. Ce projet de doctorat se concentre sur le guidage des processus de configuration de lignes de produits à base des traces de processus. Pour ce faire, les actions des parties prenantes sont tracées, fouillées et analysées. Ensuite, des solutions de guidage sont produites et orchestrées tout au long du processus de configuration. Ce projet de doctorat comprend les principales contributions suivantes: (i) Un état de l'art des insuffisances de la configuration des lignes de produits et des solutions existantes, (ii) Une méthode outillée pour le guidage de configuration de ligne de produits baptisée ConfiLog; la méthode comprend l'extraction de journaux d'événements, des traitements de fouille de processus évalués par une méthode "Goal-Question-Metric" et un moteur de recommandation, (iii) Une preuve de concept pour démontrer que le guidage de ConfiLog est générique supportant toute ligne de produits, (iv) Des instruments d'outillage avec lesquels opère la méthode ConfiLog, (v) Une évaluation de la méthode ConfiLog avec une étude empirique avec des étudiants en utilisant une ligne de vélo réelle.

La présente recherche est un work package qui fait partie d'un projet de recherche entre le Centre de Recherche en Informatique de l'Université de Paris 1 Panthéon-Sorbonne en France et le laboratoire RIADI de l'Université de Manouba en Tunisie. Cette coopération interuniversitaire numéro 16G / 1416 est un Partenariat Hubert Curien (PHC) qui vise à proposer des solutions pour l'ingénierie des Lignes de Produits.

Mots clés : Lignes de Produits; Processus de Configuration; Insuffisances de la configuration; Fouille de processus; Goal-Question-Metric; Recommandation, Guidage.

ABSTRACT

Product lines present a production strategy that provides organizations a competitive edge by improving productivity and fostering customization. However, the potential benefits of product lines can be missed when dealing with large variability models implying hundreds of variants leading, sometimes, to billions of possible configurations. Indeed, the configuration process becomes an error prone task due to the plethora of requirements and constraints to satisfy at the same time. Consequently, guiding the stakeholders during such complex configuration processes and recommending the best configuration alternatives until leading them to a satisfying experience becomes a challenge. This phd project focuses on enhancing product line configuration processes based on process traces. Therefore, stakeholder actions of previous product line configurations are logged, mined and analyzed. Afterwards, guidance solutions are devised and orchestrated throughout the configuration process. This phd project includes the following main contributions: (i) A state of the art of product line configuration shortcomings and existing solutions, (ii) A tooling method for product line configuration guidance baptized ConfiLog; the method includes event logs extraction, process mining treatments evaluated by a Goal-Question-Metric method and a recommendation engine, (iii) A Proof of concept to demonstrate that ConfiLog guidance is generic and can monitor all product lines, (iv) Tooling instruments that support the ConfiLog method, (v) An evaluation of the ConfiLog method with an Empirical Study With Students (ESWS) using a real bike line.

This research is a work package of a research project between the Centre de Recherche en Informatique of the University of Paris 1 Panthéon-Sorbonne in France and the RIADI laboratory of the University of Manouba in Tunisia. This inter-university cooperation number 16G/1416 is a Hubert Curien Partnership (PHC) that aims to propose solutions for Product Line Engineering.

Keywords : Product Lines; Configuration Process; Configuration Shortcomings; Process Mining; Goal-Question-Metric; Recommendation, Guidance.

[Cette page est intentionnellement laissée en blanc]

REMERCIEMENTS

Au Cap Bon, berceau de mon enfance, les cultivateurs se réunissent, pendant le printemps, autour de leurs vergers pour la récolte des fleurs d'oranger. Après la cueillette, de grands sacs en toile renfermant des kilos de fleurs seront destinés plus tard pour la distillation. Spéculativement, la récolte ne serait honorable uniquement si l'arrosage et l'élagage des branches ainsi que le bêchage de la terre étaient au gré de la passion, de la patience et de la persévérance.

En passant de l'autre côté de la Méditerranée, Paris, également connu pour la fragrance de ses parfums, m'a fait comprendre que les chercheurs sont comme les cultivateurs, ils ne peuvent pas empêcher leur béatitude en voyant la récolte venir et les charmilles du savoir se couvrir. J'ai encore compris que comme la distillation, par la technique traditionnelle d'autrefois, la recherche est aussi laborieuse mais enrichissante. En effet, pour produire un litre de néroli, il faut une tonne de fleurs de bigaradier. En poussant le processus jusqu'à la distillation, on obtient une huile essentielle. Tout cela renvoie à des années de recherches fournies qui sont l'aboutissement de ce travail.

Et savez-vous comment on appelle l'eau de fleurs d'oranger? « Zhar » qui veut dire littéralement «chance». Moi, j'ai eu la chance d'être "ramassé" par une connaissance, Docteur Inès Gam de l'université Tunis Manar, à qui je dois une immense reconnaissance. Je remercie Ines pour les effluves émanant de ses méthodes de travail.

Les vents violents pendant l'éclosion font que les fleurs tombent ou que l'arbre dérive quand il n'est pas suffisamment développé. Professeur Camille Salinesi de l'Université Paris 1 Panthéon Sorbonne, avait confiance en moi pendant ces temps difficiles, il m'était un haubanage qui solidifie et protège. Je tiens à remercier Camille pour le modèle qu'il a tracé en moi.

Je suis également reconnaissant envers la professeure Henda Ben Ghezala de l'Université de la Manouba, pour son soutien précieux, son labour incontournable, et pour nos entretiens vertueux qui m'ont été d'une utilité indéniable.

Je tiens à exprimer ma sincère gratitude envers le professeur Raul Mazo de l'ENSTA Bretagne, pour les soins fructueux qu'il m'a prodigués pour aboutir à ce présent hydrolat.

Pour évaluer la délicatesse olfactive, généralement les parfumeurs font appel à des aromaticiens scientifiques. Professeur Naoufel Kraiem de l'Université Tunis Manar,

professeur Saïd Assar de l'Institut Mines-Télécom France, professeure Olfa Belkahla-Driss de l'Université de la Manouba et professeure Bénédicte Le Grand de l'Université Paris 1 Panthéon Sorbonne, m'ont fait le grand honneur d'accepter de faire partie du jury évaluant cette thèse.

Tout comme une photosynthèse, mes collègues m'étaient un rayon de soleil auprès duquel je tirais régulièrement mon énergie. Je tiens à remercier mes collègues du Centre de Recherche en Informatique et le laboratoire RIADI; particulièrement David, Danillo, les deux Elena, Ali, Floriane, Nourhene, Camillo, Luisa, Angela, Sabrina, Inès et Astrid. Je n'oublie surtout pas Valérie Monfort, docent de qualité incontestable à l'UFR 27, qui m'a épaulé et transmis son expertise pendant mes expériences d'ATER. Merci du fond du cœur Valérie. Je tiens à remercier également les pompiers et les agents de propreté du Centre Pierre Mendès France et particulièrement Koma.

Avant de finir, l'arbre a des racines, les miennes sont ma famille qui m'a légué cet héritage de passion et d'amour de l'enseignement et de la recherche. Je vous remercie pour votre soutien et votre bienveillance. Je remercie la femme de ma vie pour sa patience et sa surveillance quand l'alambic se bouche.

Inspirés du printemps 2020, à Hammamet

[Cette page est intentionnellement laissée en blanc]

TABLE DES MATIÈRES

Chapitre I : Introduction

I.1. Contexte	1
I.2. Problèmes observés	3
I.3. Questions de recherche	5
I.4. Méthode de recherche	6
I.5. Contributions	8
I.6. Environnement de recherche	10
I.7. Plan de mémoire	11

PARTIE A : EXISTANT

Chapitre II : Background

II.1. Introduction	15
II.2. La recommandation : Principes	15
II.3. Typologie des paradigmes classiques de recommandation	16
II.3.1. Le filtrage collaboratif	16
II.3.1.1. <i>La techniques des k-plus proches voisins</i>	17
II.3.1.2. <i>La techniques probabiliste</i>	18
II.3.2. La recommandation basée sur le contenu	19
II.3.3. La recommandation basée sur la connaissance	21
II.3.4. La recommandation heuristique	22
II.3.5. La recommandation hybride	24
II.4. La recommandation basée sur la fouille de processus	25
II.4.1. La découverte, la mesure de conformité et l'amélioration de processus	26
II.4.1.1. <i>La découverte de processus</i>	26
II.4.1.2. <i>La mesure de conformité de processus</i>	27
II.4.1.3. <i>L'amélioration de processus</i>	28
II.4.2. Les algorithmes de fouille de processus	28
II.4.2.1. <i>Le Heuristic Miner</i>	29
II.4.2.2. <i>Le Fuzzy Miner</i>	30
II.5. Motivations pour la configuration des lignes de produits	32
II.6. Conclusion	33

Chapitre III : Etat de l'art

III.1. Introduction	37
III.2. Méthode de recherche adoptée	37
III.3. Mise en oeuvre de la méthode	38
III.3.1. Définition des mots clés de recherche	38
III.3.2. Définition des sources	38
III.3.3. Sélection et quantification des articles	39
III.4. Résultats	40
III.4.1. Problèmes du processus de configuration des lignes de produits	40

III.4.1.1. <i>La complexité de la décision</i>	40
III.4.1.2. <i>Le manque de la performance</i>	42
III.4.1.3. <i>Le besoin de guidage</i>	43
III.4.2. Solutions du processus de configuration des lignes de produits	44
III.4.2.1. <i>Les méthodes de configuration interactive</i>	45
III.4.2.2. <i>Les méthodes de configuration basées sur la recommandation</i>	45
III.4.2.3. <i>Les méthodes de configuration combinées</i>	46
III.4.2.4. <i>Les méthodes de configuration heuristiques</i>	47
III.4.2.5. <i>Les méthodes de configuration basées sur la fouille</i>	47
III.5. Discussions et Positionnement	48
III.6. Conclusion	51

PARTIE B : MÉTHODE

Chapitre IV : Aperçu de la méthode ConfiLog

IV.1. Introduction	56
IV.2. Cadre méthodologique	56
IV.3. Vue globale de la méthode ConfiLog	58
IV.3.1. La vue fonctionnelle de la méthode ConfiLog	58
IV.2.2. La vue dynamique de la méthode ConfiLog	60
IV.3. Exemple d'application	62
IV.5. Caractéristiques de la méthode ConfiLog	64
IV.6. Conclusion	67

Chapitre V : La méthode ConfiLog

V.1. Introduction	70
V.1. Orchestration des buts des questions et des métriques	70
V.1.1. But 1 : Réduire le temps requis pour atteindre une configuration valide du point de vue du processus	71
V.1.2. But 2 : Maximiser la flexibilité et l'efficacité de la configuration du point de vue du processus	72
V.1.3. But 3 : Maximiser la personnalisation de la configuration du point de vue du processus	73
V.1.4. But 4 : Promouvoir la recommandation de la configuration du point de vue des intérêts de l'entreprise	75
V.2. La fouille de processus de configuration	76
V.2.1. L'extraction des traces de processus de configuration	76
V.2.2. La reconstitution des modèles de processus de configuration	77
V.2.3. Le filtrage et l'interprétation des modèles de processus de configuration	79
V.3. Calcul de la recommandation	82
V.3.1. Le paramétrage du processus de configuration	83
V.3.2. L'abstraction et le filtrage des réponses de fouille	83
V.3.3. La pondération des réponses de fouille	84
V.3.4. Le calcul de l'impact de décision	85
V.3.5. Démonstration du moteur de recommandation ConfiLog avec la ligne de sites web	86
V.3.5.1. <i>Etape n du processus de configuration de la ligne de sites web</i>	87
V.3.5.2. <i>Etape n+1 du processus de configuration de la ligne de sites web</i>	90
V.4. Conclusion	91

PARTIE C : APPLICATION

Chapitre V : Outillage

VI.1. Introduction	96
---------------------------	-----------

VI.2. ConfiLog : Les choix d'ingénierie	96
VI.2.1. Application Internet Riche	96
VI.2.3. Architecture en couches	97
VI.2.4. Choix d'implémentation	98
VI.2. Architecture logicielle de ConfiLog	99
VI.2.1. Côté client	100
VI.2.1.1. Modélisateur de lignes de produits	100
VI.2.1.2. Configureur de lignes de produits	103
VI.2.2. Côté serveur : La couche de configuration	106
VI.2.2.1. Moteur d'interfaces	106
VI.2.2.2. Gestionnaire de configuration	106
VI.2.3. Côté serveur : La couche de fouille	107
VI.2.2.1. Chargeur (ETL)	107
VI.2.2.2. Fouilleur de processus	109
VI.2.4. Coté serveur : La couche de recommandation	112
VI.2.3.1. Moteur de recommandation	113
VI.2.3.2. Conducteur de recommandation	115
VI.6. Conclusion	118

Chapitre VI: Evaluation

VII.1. Introduction	121
VII.2. Paramétrage de l'étude empirique avec les étudiants	121
VII.2.1. Planification	122
VII.2.1.1. Groupes d'étudiants participants	122
VII.2.1.2. Motifs de la configuration sans guidage (P1)	122
VII.2.1.3. Motifs de la configuration avec guidage (P2)	123
VII.2.1.4. Hypothèse et questions de recherche	123
VII.2.1.3. Protocole d'expérimentation	124
VII.2.2. Environnement	126
VII.2.3. Questionnaires	126
VII.2.4. Etude de cas : Ligne de vélos	127
VII.3. Résultats de l'expérimentation	130
VII.3.1. Background et attentes des étudiants	130
VII.3.2. Comparaison analytique des deux processus de l'expérimentation	131
VII.3.3. Retour d'expérience des étudiants	134
VII.3.4. Temps de réponse	135
VII.3.5. Passage à l'échelle	137
VII.4. Menaces à la validité	138
VII.5. Conclusion	140

Chapitre VII : Conclusion et Perspectives

VIII.1. Résumé des contributions	144
VIII.2. Perspectives de recherche et Discussions ouvertes	146

Références bibliographiques	150
------------------------------------	------------

Annexes	163
----------------	------------

LISTE DES FIGURES

Figure (1) : Aperçu de la méthode de guidage envisagée	9
Figure (2) : Fondements de la recommandation basée sur le filtrage collaboratif	19
Figure (3) : Fondements de la recommandation basée sur le contenu	20
Figure (4) : Fondements de la recommandation basée sur la connaissance	22
Figure (5) : Fondements de la recommandation basée sur les heuristiques	23
Figure (6) : Fondements de la recommandation hybride	24
Figure (7) : Techniques de fouille de processus : Découverte, conformité et amélioration	26
Figure (8) : Instanciation de la maquette de (DSR)	57
Figure (9) : Vue fonctionnelle de la méthode ConfiLog	60
Figure (10) : Récapitulatif des interactions effectuées entre la partie prenante et ConfiLog	61
Figure (11) : Modèle adapté d'une ligne de produits de sites Web	63
Figure (12) : Structure du Modèle de la méthode (GQM)	70
Figure (13) : Exemple de reconstitution d'un modèle de processus complet	78
Figure (14) : Exemple de modèle de processus filtré	81
Figure (15) : Processus de recommandation ConfiLog	82
Figure (16) : Etat de configuration de la Ligne de produits de Site Web	86
Figure (17) : L'architecture logicielle de ConfiLog	99
Figure (18) : Capture d'écran d'accueil du Configurateur ConfiLog	101
Figure (19) : Capture d'écran de la sélection du modèle et du but de guidage	104
Figure (20) : Capture d'écran de la conduite du processus de configuration	105
Figure (21) : Partie du modèle de processus de configuration des sites Web	111
Figure (22) : Interface de débogage de la pondération et des calculs d'impact de décisions	117
Figure (23) : Protocole d'expérimentation de l'étude empirique avec les étudiants	125
Figure (24) : Ligne de vélos en notation OVM	128
Figure (25) : Comparaison des temps de réponse requis pour (P1) et (P2)	136
Figure (26) : Temps de réponse après la configuration du premier point de variabilité	138

LISTES DES TABLEAUX

Tableau (1) : Distribution des publications sélectionnées en fonction des sources	38
Tableau (2) : Complexité de la décision durant le processus de configuration des lignes de produits	40
Tableau (3) : Manque de performance durant le processus de configuration des lignes de produits	42
Tableau (4) : Besoin du guidage durant le processus de configuration des lignes de produits	43
Tableau (5) : Evaluation des solutions existantes pour le guidage de configuration des lignes de produits VERSUS la méthode envisagée	50
Tableau (6) : (GQM) pour le but “Réduire le temps requis pour atteindre une configuration valide du point de vue du processus”	72
Tableau (7) : (GQM) pour le but “Maximiser la flexibilité et l'efficacité de la configuration du point de vue du processus”	73
Tableau (8) : (GQM) pour le but “Maximiser la personnalisation de la configuration du point de vue du processus”	74
Tableau (9) : (GQM) pour le but “Promouvoir la recommandation de la configuration du point de vue des intérêts de l'entreprise”	75
Tableau (10) : Attributs capturés dans le journal d'événements	76
Tableau (11) : Calcul des prédicats pour étape 1	88
Tableau (12) : Calcul des prédicats pour étape 2	90
Tableau (13) : Résultats analytiques de la configuration avec et sans guidage	131

[Cette page est intentionnellement laissée en blanc]

Chapitre I : Introduction

I.1. Contexte	1
I.2. Problèmes observés	3
I.3. Questions de recherche	5
I.4. Méthode de recherche	6
I.5. Contributions	8
I.6. Environnement de recherche	10
I.7. Plan de mémoire	11

Chapitre I

Introduction

I.1. Contexte

Les lignes de produits présentent un timbre de réutilisation qui prône une vision d'économie logicielle. La finalité envisagée par cette vision est de développer, capitaliser et réutiliser un ensemble de produits en fonction des facteurs de variation plutôt que de développer chaque produit unique, massivement et séparément, le cas d'économie d'échelle. Le Software Engineering Institute (SEI) définit une ligne de produits comme étant « un ensemble de systèmes qui partagent un ensemble de caractéristiques communes, répondant aux besoins spécifiques d'un segment de marché particulier ou d'une mission particulière et qui sont développés à partir d'un ensemble commun d'assets de base de manière prescrite »¹.

En ingénierie des lignes de produits, [Clement et al., 2002] [Pohl et al., 2005] distinguent deux processus, notamment, l'ingénierie de domaine et l'ingénierie d'application.

L'ingénierie de domaine consiste à spécifier les assets réutilisables et leur contexte de réutilisation en définissant les éléments communs et variables de la ligne de produits. Les éléments communs et variables sont généralement représentés à travers des modèles, appelés modèles de lignes de produits.

Quant à l'ingénierie d'application, il s'agit de développer des produits par réutilisation du domaine. En effet, ce processus permet de générer des applications spécifiques en exploitant la variabilité de la ligne de produits.

Le processus de configuration constitue une tâche fondamentale dans l'ingénierie des lignes de produits reflétant ainsi le passage du domaine à l'application. Il comprend l'ensemble des activités qui spécifient un produit valide à partir d'un modèle d'une ligne de produits conformément aux exigences des utilisateurs vis-à-vis de ce produit [Hubaux

¹ Lien : <http://www.sei.cmu.edu>

et al., 2010] [Deelstra et al., 2003]. Par ailleurs, la conduite du processus de configuration considère deux niveaux analogues, notamment, les variantes du modèle à configurer et l'ordre de configuration de ces variantes. Généralement, une configuration est conduite en fonction des actions de la partie prenante. La partie prenante est considérée comme étant un ou plusieurs acteurs concerné(s) par la configuration. Tout dépend du contexte, la partie prenante peut correspondre à un internaute s'il s'agit d'un site de e-commerce par exemple et d'un architecte de système s'il s'agit d'une configuration de système à partir d'une ligne de produits logiciels, etc.

Avant d'aborder une catégorisation des processus de configuration, il convient de souligner qu'une configuration représente le produit issu d'un processus de configuration de la ligne de produits. Les éléments contenus dans la configuration doivent inclure tous les éléments obligatoires et certains des éléments variables de la ligne. Ces derniers doivent respecter les règles d'interdépendance entre eux (par exemple, un choix unique d'un élément parmi un groupe d'éléments strictement alternatifs).

Deux grandes catégories de processus de configuration peuvent être distinguées, notamment, les processus de configuration sans guidage et les processus de configuration avec guidage.

Dans les processus de configuration non guidés, les parties prenantes configurent le modèle de la ligne de produits et c'est à posteriori, qu'ils vérifient la cohérence des choix vis-à-vis des contraintes du domaine [Djebbi, 2011]. De par sa nature, ce processus aboutit à la construction d'un produit de la ligne. Néanmoins, ces méthodes présentent plusieurs limites en relation avec la cohérence de la configuration, particulièrement dans les larges modèles de lignes de produits. Il devient impossible de configurer des millions voire des milliards de combinaisons de choix possibles et vérifier, à posteriori, leur cohérence par rapport au modèle de la ligne de produits. Par exemple, chez le fabricant automobile Renault, le processus de configuration peut conduire à 10^{21} configurations pour la ligne de produits "Traffic" [Astesana et al., 2010]. En plus de cela, dans certains outils de configuration tels que les configureurs en ligne des ordinateurs ou des voitures opérant sous des modèles contenant un large éventail de variantes, il est important de garantir à l'internaute des choix flexibles exécutés dans des temps de réponse courts. En l'occurrence, il s'avère important aussi de proposer un support de configuration à plusieurs utilisateurs faisant des choix ensemble.

Pour pallier aux limites des processus de configuration non guidés, plusieurs approches préconisent un support simplifiant la tâche de configuration par la propagation des choix de la partie prenante [Rosenmuller et al., 2010] [White et al., 2007]. Pour ce faire, ces approches ont eu recours à la formalisation des langages de modélisation de la variabilité afin de pouvoir automatiquement analyser les exigences d'une ligne de produits en termes de configuration. La littérature distingue trois types de formalisation des exigences d'une ligne de produits, notamment : (i) *les approches dirigées par les modèles graphiques* comme FODA [Kang et al., 1990], OVM [Pohl et al., 2005] et DOPLER [Dhungana et al., 2007], (ii) *les approches dirigées par la logique* [Batory, 2005] et (iii) *les approches dirigées par la programmation de contraintes* [Salinesi et al., 2011] [Benavides et al., 2007].

Les approches dirigées par la logique et la programmation des contraintes visent à transformer les modèles de variabilité en un langage compréhensible par une machine afin de pouvoir raisonner sur les solutions de configuration possibles. En effet, les approches dirigées par la logique transforment le modèle de la ligne de produits sous forme de relations logiques et d'utiliser des solveurs de logique pour amener les parties prenantes à des configurations cohérentes. Cependant, d'une manière générale, la finalité de ces approches n'est pas la configuration en elle-même, mais la mesure de conformité des modèles.

I.2. Problèmes observés

Un problème particulier, qui constitue en soi un défi important pour l'industrie, est celui du guidage du processus de configuration à partir de larges modèles. *Le passage à l'échelle* transforme le processus de configuration en une tâche ardue qui consomme énormément de temps et risque fréquemment l'erreur. L'activité de configuration devient rapidement lourde vu le nombre de décisions à prendre (avec restriction d'ordre parfois) et le temps de réponse requis. Vraisemblablement, les configureurs industriels ou commerciaux comme les boutiques en ligne de vélos² ou encore de voitures³ adhèrent, entre autres, à ces problèmes de performance. Par ailleurs, la littérature distingue trois types de problèmes en relation avec le processus de configuration notamment : (i) *la complexité de la décision* [Noorian et al., 2017][Zhan et al., 2018] [Hayashi et al., 2017], (ii) *le manque de performance* [Ochoa et al., 2017][Sinnhofer et al., 2016][Zhang et al.,

² Lien : <http://www.bikeconfig.com/>

³ Lien : <https://www.renault.fr/vehicules/vehicules-particuliers/talisman/configurateur.html>

2013] et (iii) *l'appel explicite au guidage* [Dumitrescu et al., 2013][Urli et al., 2012][Chastel et al., 2015] tout au long du processus de configuration.

Ces anomalies ont été rapportées par la communauté des lignes de produits, qui, à son tour, a proposé, des méthodes de configuration interactive [Benavides et al., 2010] [Hadzic et al., 2004], des méthodes de recommandation [Jannack et al., 2011], une combinaison de ces deux méthodes [Triki et al., 2014] et des méthodes heuristiques [Mazo et al., 2014], entre autres.

Bien que les travaux de recherche cités ci-dessus ont contribué à résoudre, en partie, les problèmes de passage à l'échelle dans le processus de configuration, des carences en rapport avec l'ordre du processus et les différents buts du guidage des parties prenantes, résident encore, et notamment, parce-que :

- ***L'ordre de configuration est statique*** : Peu de contributions se sont focalisées sur l'ordre du processus de configuration, ce qui constitue un manque dans la littérature. Toutefois, les méthodes heuristiques prétendent que cet ordre est calé sur la hiérarchie du modèle ou les contraintes des variantes. Par ailleurs, la nature de l'ordre est considérée statique vu que ce dernier est défini à l'avance, comme dans la suite d'outils Dopler [Dhungana et al., 2011] ou dans le workflow de [Hubaux et al., 2009]. Cet ordre ne présente aucune adaptabilité en fonction des particularités du processus de configuration en question, du fait qu'il est fermement lié au modèle. Les autres méthodes citées ci-dessus, fournissent des prédictions sur les valeurs des variantes qui peuvent aller avec une configuration partielle donnée, et non une série dynamique des choix à présenter. Par exemple, chez Renault, pour certains acheteurs potentiels, présenter, en premier, les points de variabilité en rapport avec le côté esthétique de la voiture, semble plus intéressant que présenter les points de variabilité en rapport avec le côté mécanique.
- ***Les buts de guidage sont ignorés*** : d'un point de vue marketing, il n'est pas toujours acceptable de fournir le même guidage à des parties prenantes ayant des préférences et/ou des préoccupations différentes. En effet, même si le guidage "standard" permet d'épargner des efforts et du temps de révision des choix, il est dégradé dès que la partie prenante ne se sent pas visée par l'un des choix qui lui ont été fournis. En outre, la prise en compte des comportements des parties prenantes pendant la configuration est un axe ignoré. Par exemple, chez Renault, il semble intéressant pour certains acheteurs potentiels de choisir un processus de

configuration détaillé, s'ils sont connaisseurs du monde de l'automobile. Toutefois, ils peuvent choisir un processus de configuration avec moins de détails, s'ils utilisent le configurateur juste pour faire des essais et des combinaisons de choix.

Les limites révélées des approches de guidage déjà proposées motivent l'idée de se poser des questions autour des valeurs des variantes, de l'ordre du processus de configuration et du guidage dirigé par des buts. Pour pallier ces insuffisances, des techniques logicielles comme les systèmes de recommandation [Jannach et al., 2010], en vue de guidage, ont vu le jour pour nous faciliter la vie. Néanmoins, la littérature est prodigue en solutions de recommandation pas forcément adaptées au contexte de configuration de lignes de produits.

I.3. Questions de recherche

Cette thèse s'intéresse à la question de recherche suivante :

Comment mieux guider les parties prenantes lors du processus de configuration d'une ligne de produits ?

De cette grande question, d'autres questions de recherche en découlent :

Q1. Quels sont les défis du guidage de la configuration de lignes de produits face aux solutions existantes?

La littérature admet que le guidage du processus de configuration repose sur plusieurs considérations en rapport avec le modèle de la ligne de produits et les acteurs de configuration. Cependant, différentes controverses se suscitent face à des modèles syntaxiquement différents, d'un côté, et des acteurs ayant différents buts et différentes préoccupations de l'autre côté. Afin d'apporter des éléments de réponses à ces incompréhensions, cette question de recherche vise à identifier les défis du guidage envisagé.

Q2. A quels niveaux du processus de configuration d'une ligne de produits, le guidage s'applique ?

Plusieurs solutions de guidage ont vu le jour, néanmoins, ces dernières ne conduisent pas la configuration de la même manière. En effet, la recommandation des variantes du modèle a été considérée dans maintes études, cependant, la recommandation d'un ordre enchaîné pour la configuration de ces variantes a été ignorée. D'un point de vue

technique, l'ordre de configuration des variantes s'avère capital pour la production du guidage, particulièrement dans les cas où les recommandations sont calculées en fonction d'une configuration partielle.

Q3. Quelles seraient les solutions de guidage à apporter pour guider le processus de configuration d'une ligne de produits ?

Afin d'apporter une solution de guidage adéquate et à la hauteur des attentes des industriels, cette question de recherche cherche à se focaliser sur les besoins concrets des configurateurs. Par ailleurs, elle s'intéresse aux moyens techniques de recommandation qui permettront de guider le processus de configuration de lignes de produits. Généralement, les connaissances déduites à partir du traçage des expériences antérieures servent à mieux guider les expériences futures. Cette hypothèse mène à des réflexions en rapport avec les particularités du contexte en question, à savoir les contextes contraints par une variabilité interdépendante comme la configuration des lignes de produits.

Cette thèse s'intéresse aux hypothèses de recherche suivantes :

H1. Une méthode de guidage basée sur les traces de processus permettra de mieux guider les parties prenantes lors du processus de configuration d'une ligne de produits.

H2. Une méthode de guidage, prenant en compte les différents niveaux du processus de configuration et les différentes préoccupations des parties prenantes, permettra de mieux guider les parties prenantes lors du processus de configuration d'une ligne de produits.

H3. Une implémentation bienséante de la méthode susmentionnée permettra de répondre aux défis relevés.

I.4. Méthode de recherche

Le projet rapporté dans cette thèse a été mené en utilisant une méthodologie de recherche de type "Design Science" [March et Smith, 1995] [Hevner et al., 2004] définie comme étant la conception et la validation des propositions de solution à des problèmes pratiques.

La présente recherche a été conduite conformément aux lignes directives citées ci-dessous [Hevner et al., 2004] :

- La conception d'un artefact,
- La pertinence du problème,

- L'évaluation de la conception,
- La contribution de la recherche,
- La mesure de rigueur de la recherche et
- La conception du processus de recherche.

Conjointement à la méthode "Design Science", la présente recherche fait recours à la méthode "Goal-Question-Metric" (GQM) [Basili et al., 1994] pour la rétroaction des lignes directives citées ci-dessus. La (GQM) est défini comme étant une méthode pour la définition des buts quantifiables et l'interprétation des données de mesure logicielle par le biais des questions et des métriques. En effet, la (GQM) adoptée par la méthode proposée dans cette thèse opère sous trois niveaux, à savoir :

- Le niveau conceptuel [Buts] : Les buts sont définis en vue d'améliorer le processus de configuration des lignes de produits.
- Le niveau opérationnel [Questions] : Les questions sont posées en vue d'apporter des réponses du guidage permettant d'atteindre les buts.
- Le niveau quantitatif [Métriques] : Les métriques sont utilisées pour évaluer les réponses aux questions de guidage de manière quantitative et qualitative.

Ainsi, la (GQM) présente la méthode qui a permis de "forger" les buts, les questions de guidage et les critères d'évaluation des résultats obtenus dans la méthode de guidage proposée dans cette thèse.

Par ailleurs, la présente recherche suit le modèle de présentation de la recherche sur les systèmes d'information [Peppers et al., 2007] vu qu'elle vise à valider les hypothèses de recherche au moyen de prototypes et une étude de cas. Le processus suivant a été mis en oeuvre :

1. Un état de l'art sur les problèmes et solutions du processus de la configuration des lignes de produits a été effectué.
2. La problématique de recherche a été explorée à partir de l'état de l'art.
3. Une analyse critique des approches existantes pour le guidage du processus de configuration a été réalisée en se référant aux questions de recherche.
4. Une méthode de guidage de configuration des lignes de produits basée sur la fouille de processus a été proposée.
5. Un outillage qui supporte la méthode ConfiLog a été implémenté.
6. Une évaluation de la méthode en fonction d'une expérimentation conduite avec des étudiants a été menée.

I.5. Contributions

Cette thèse présente quatre contributions principales :

1. Un état de l'art des problèmes de configuration de lignes de produits et des solutions existantes

La revue des problèmes et des solutions rapportés dans la littérature sur le guidage de processus de configuration des lignes de produits montre qu'aucune considération des buts des acteurs de la configuration n'est mentionnée. En outre, aucun guidage sur l'ordre des étapes pendant le processus de configuration n'est pris en compte. Bien que la littérature offre l'étude de diverses approches de guidage, à notre connaissance, aucune étude n'avait auparavant exploré la fouille de processus de configuration pour l'analyse des données événementielles de processus.

2. Une méthode pour le guidage du processus de configuration de lignes de produits

Face à des modèles de lignes de produits de plus en plus variés et des parties prenantes de plus en plus différentes, le guidage à produire doit tenir compte des buts. Dans la méthode baptisée "ConfiLog", la partie prenante a le choix d'exprimer son but de guidage (par exemple : un processus de configuration rapide) pour qu'elle soit guidée tout au long du processus de configuration.

La figure (1) schématise les grandes lignes de la méthode de ConfiLog. En effet, tout au long du processus de configuration, les actions des parties prenantes sont tracées pour capturer les décisions de sélection, de refus, ou encore d'abandon, l'horodatage des décisions, le but du guidage sélectionné etc. Les traces de processus sont capitalisées dans une base de traces et fouillées plus tard afin de reconstituer des modèles de processus de configuration. Pour chaque but de guidage sélectionné par la partie prenante, la méthode ConfiLog orchestre, en arrière plan, des questions de guidage et des métriques d'évaluation capitalisées dans une base (GQM). Les réponses aux questions de guidage et leurs évaluations se font par le biais d'un processus de filtrage et d'interprétation des modèles des processus reconstitués.

Les instances du processus qui vont avec (i) la configuration en question et (ii) le but de guidage sélectionné par la partie prenante représentent l'essence du guidage. Par ailleurs, ces instances de processus sont réutilisées par un moteur de recommandation qui calcule à chaque étape de la configuration en cours, l'ordre des décisions et les instances de décisions soit le refus ou la sélection d'une variante donnée en expliquant à

la partie prenante pourquoi le guidage en question est suggéré. Étant générique, la méthode proposée pourra, par conséquent, s'appliquer à toute ligne de produits.

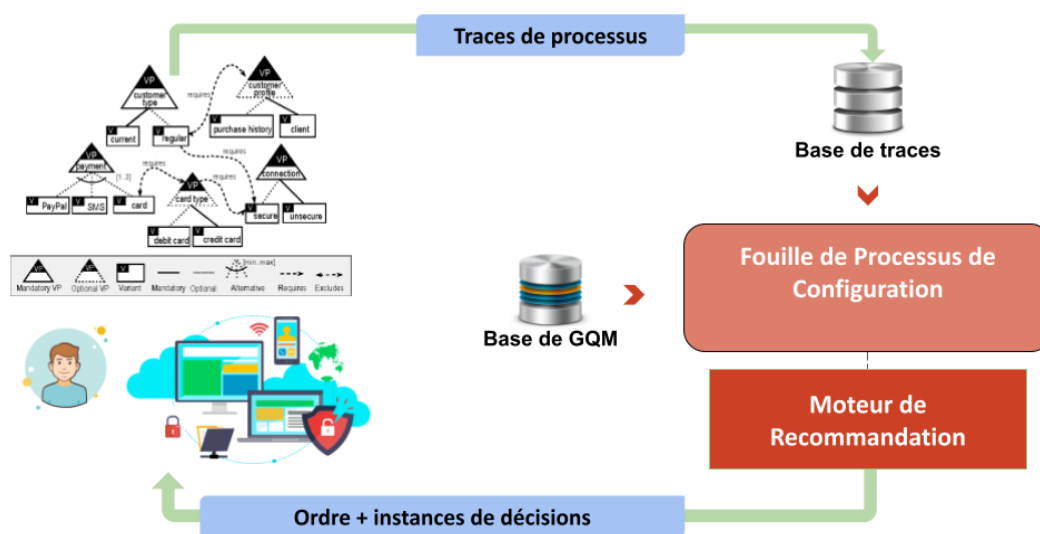


Figure (1) : Aperçu de la méthode de guidage envisagée

3. Un outillage qui supporte la méthode ConfiLog

Afin de démontrer la faisabilité de la méthode, un configurateur a été développé. Le configurateur englobe cinq grandes fonctionnalités, respectivement : la modélisation de la ligne, le calcul de sa cohérence, la configuration, la capitalisation des traces et la production du guidage fondé sur la fouille de processus. Ce configurateur a été implémenté en utilisant les principales technologies suivantes : HTML, Java EE, Javascript et le moteur d'interfaces Freemarker.

4. Une évaluation de la méthode ConfiLog

La méthode ConfiLog a été évaluée par une étude empirique menée avec des groupes d'étudiants. Pour réaliser cela, des étudiants en France et en Tunisie ont été invités à configurer un vélo, dans un premier temps, sans utiliser de méthode particulière de guidage, puis dans un deuxième temps, en utilisant le guidage de la méthode ConfiLog. Les attentes, les traces de configuration et les retours d'expériences ont été confrontés et évalués.

I.6. Environnement de recherche

Cette thèse s'inscrit dans le cadre d'un projet de collaboration entre le laboratoire RIADI de l'Université de la Manouba en Tunisie et le Centre de Recherche en

Informatique (CRI) de l'Université Paris 1 Panthéon-Sorbonne en France. Cette coopération inter-universitaire franco-tunisienne s'inscrit sous la forme d'un Partenariat Hubert Curien (PHC) numéro 16G/1416 intitulé CONFIGURE.

L'objectif de CONFIGURE est de développer une méthode pour la recommandation et la configuration des lignes de produits. Ainsi, la structure du projet en termes de work packages (WP) est comme suit (1 work package de gestion et 3 work packages scientifiques)

- WP0: Gestion du projet
- WP1: Spécification des lignes des produits
- WP2 : Constitution de l'ontologie pour les lignes de produits
- WP3: Elaboration du système de recommandation

Ce manuscrit touche principalement à WP3 et légèrement à WP1. Les autres WPs ont été confiés à d'autres membres du projet.

La recherche a été réalisée conjointement, au sein du laboratoire RIADI (Tunisie) et du laboratoire de recherche CRI (France) dans le cadre d'une thèse en cotutelle entre l'Université Paris 1 Panthéon Sorbonne et l'Université de la Manouba. Plusieurs déplacements entre la France et la Tunisie ont été effectués par le doctorant, les directeurs de thèse et les co-encadrants. Également, des visioconférences entre les membres de l'équipe ont été organisées régulièrement pour le suivi des travaux de recherche.

Le RIADI et le CRI sont des références scientifiques de renommée en ingénierie des lignes de produits [Mazo et al., 2014] [Salinesi et al., 2012] [Ouali et al., 2011a] [Ouali et al., 2011b], en systèmes de recommandation [Epure et al., 2017a] [Epure, al., 2017b] [Yanes et al., 2010] [Gorrab et al., 2017] et en fouille de processus [Khodabandelou et al., 2013] [Deneckere et al., 2014] [Thabet et al., 2018] [Thabet et al., 2017]. CONFIGURE justifie une continuité de collaborations, du RIADI et du CRI, sur lesquelles ils veillent depuis plusieurs années. L'intégration de cette équipe a aidé à mieux fixer les défis auxquels ce projet de thèse fait face.

Cette implication a permis d'identifier la problématique, de mener la démarche de recherche de type "design-science" et d'élaborer une étude de cas, la validant.

La problématique définie dans la thèse est centrale aux partenaires industriels de CONFIGURE, du fait qu'ils travaillent aussi sur des configurateurs industriels et sont confrontés à des besoins de guidage lors du processus de configuration. Néanmoins,

“ConfiLog” est générique et adaptable à d’autres contextes adoptant l’ingénierie des lignes de produits. À travers cette thèse, une méthodologie globale et outillée de configuration des lignes de produits guidée par le biais de la fouille de processus, est mise en place.

I.7. Plan de mémoire

Dans ce chapitre d’introduction, La motivation du sujet a été démontrée, les principales contributions ont été listées, un aperçu de la méthode est fourni, et l’environnement dans lequel la recherche a été développée est présenté.

La suite de ce mémoire est composée des chapitres suivants :

- **Chapitre II : Background.** Ce chapitre rappelle l’ensemble des paradigmes de recommandation pour positionner cette recherche par rapport à la communauté scientifique de fouille de processus et de systèmes de recommandation.
- **Chapitre III : Etat de l’art.** Ce chapitre identifie les problèmes de configuration de lignes de produits et les insuffisances des méthodes proposées.
- **Chapitre IV : Aperçu de la méthode ConfiLog.** Ce chapitre présente la méthodologie adoptée pour mener à bien ce projet de recherche. En outre, il présente un aperçu général de la méthode ConfiLog.
- **Chapitre V : La méthode ConfiLog.** Ensuite, il explique en détail les fonctions du guidage (i) dirigé par des buts, (ii) basé sur la fouille de processus et (iii) produit via un moteur de recommandation.
- **Chapitre VI : Outillage.** Ce chapitre décrit la suite des outils développés et réutilisés pour la production et l’orchestration du guidage.
- **Chapitre VII : Évaluation.** Ce chapitre illustre la méthode ConfiLog, par une étude empirique avec des étudiants, et ses aboutissements. Une évaluation du guidage fourni aux étudiants a été conduite ainsi.
- **Chapitre VIII : Conclusion et Perspectives.** Ce chapitre fournit un récapitulatif des résultats scientifiques apportés par cette thèse et ouvre des perspectives de recherche en rapport avec des problèmes connexes.

[Cette page est intentionnellement laissée en blanc]

PARTIE A : EXISTANT

Chapitre II : Background

Chapitre III : Etat de l'art

Chapitre II : Background

II.1. Introduction	15
II.2. La recommandation : Principes	15
II.3. Typologie des paradigmes classiques de recommandation	16
II.3.1. Le filtrage collaboratif	16
II.3.1.1. <i>La techniques des k-plus proches voisins</i>	17
II.3.1.2. <i>La techniques probabiliste</i>	18
II.3.2. La recommandation basée sur le contenu	19
II.3.3. La recommandation basée sur la connaissance	21
II.3.4. La recommandation heuristique	22
II.3.5. La recommandation hybride	24
II.4. La recommandation basée sur la fouille de processus	25
II.4.1. La découverte, la mesure de conformité et l'amélioration de processus	26
II.4.1.1. <i>La découverte de processus</i>	26
II.4.1.2. <i>La mesure de conformité de processus</i>	27
II.4.1.3. <i>L'amélioration de processus</i>	28
II.4.2. Les algorithmes de fouille de processus	28
II.4.2.1. <i>Le Heuristic Miner</i>	29
II.4.2.2. <i>Le Fuzzy Miner</i>	30
II.5. Motivations pour la configuration des lignes de produits	32
II.6. Conclusion	33

Chapitre II

Background

II.1. Introduction

Ce chapitre présente un background à l'intersection de deux axes de recherche, notamment, les systèmes de recommandation dans un contexte de configuration et la fouille de processus. La Section 2 expose les enjeux de la recommandation dans le contexte de la configuration de lignes de produits. La Section 3 met en exergue les différents paradigmes de recommandation. La Section 4 distingue la recommandation par le biais de la fouille de processus et présente ses algorithmes courants.

II.2. La recommandation : Principes

Vu la pléthore des choix confrontés, le processus de prise de décisions devient rapidement une activité de longue haleine, consistant à sélectionner ou refuser un choix parmi une large panoplie. Dans le processus de configuration des lignes de produits, la décision de sélectionner ou refuser une variante repose sur des contraintes, une analyse du contexte et un ensemble d'étapes intermédiaires qui conduisent à la configuration finale. D'une manière plus objective, une prise de décision, sur un point de variabilité, consiste à exploiter un ensemble d'informations afin que la décision soit appropriée avec les besoins et/ou les attentes de la partie prenante. Devant les larges lignes de produits, la partie prenante se retrouve envahie par un grand nombre de décisions complexes et risque de ne pas savoir quelles décisions prendre ni dans quel ordre.

Pour faire face à ce phénomène, des techniques informatiques ont vu le jour pour faciliter le processus de prise de décisions. En effet, les systèmes de recommandation basés, entre autres, sur le profil de la partie prenante, le contenu du modèle à configurer, et/ou des heuristiques ont été proposés. Par ailleurs, des techniques de recommandation basées sur la fouille des données [Hand et al., 2014], la fouille de texte [Berry, 2004] ou la fouille de processus [Aalst, 2011] sont de plus en plus utilisées. Dans un contexte de recommandation basée sur la fouille de processus, le potentiel système

extrait des connaissances à partir de lourds volumes de données de processus afin de trouver des informations de recommandation supportant les critères fixés au préalable, et en interpréter des solutions.

Avant d'aborder une typologie des paradigmes classiques de recommandation ainsi que la recommandation basée sur la fouille de processus, il convient de définir la recommandation elle-même. En effet, la recommandation est définie comme étant une stratégie de prise de décisions pour les utilisateurs dans des environnements d'information complexes [Rashid et al., 2002]. De surcroît, la recommandation est définie comme étant un moyen d'aider les parties prenantes à faire des choix lorsqu'il n'y a pas suffisamment de connaissances ou d'expérience par rapport aux alternatives [Resnick et al., 1997].

II.3. Typologie des paradigmes classiques de recommandation

Différents paradigmes de recommandation existent dans la littérature, notamment, le filtrage collaboratif [Resnick et al., 1994], la recommandation basée sur le contenu [Pazzani et al., 1997], la recommandation basée sur la connaissance [Burke, 2000], la recommandation heuristique [Mazo et al., 2014] et la recommandation hybride [Burke, 2007]. Les sous sections qui suivent, expliquent le principe de chaque paradigme par des définitions formelles et des illustrations.

II.3.1. Le filtrage collaboratif

Comme son nom l'indique, le filtrage collaboratif regroupe des techniques de recommandation qui visent à opérer une sélection sur les éléments à configurer (filtrage) en se basant sur le comportement et les préférences exprimées par d'autres parties prenantes (collaboration). Pour ce faire, le filtrage collaboratif permet de recommander l'utilité d'une décision donnée pour une ou plusieurs parties prenantes. Les recommandations de filtrage collaboratif se basent fondamentalement sur des évaluations de cette décision par d'autres parties prenantes.

Formellement, l'utilité $u(p, d)$ d'une décision d pour une partie prenante p est calculée en fonction des utilités des décisions $u(p_j, d)$ estimées par les parties prenantes qui lui sont similaires [Adomavicius et al., 2005]. En effet, j présente les parties prenantes similaires avec $p_j \in P$.

La littérature distingue deux types de techniques de filtrage collaboratif, notamment, les techniques basées sur la mémoire et les techniques basées sur les modèles. La

technique des k-plus proches voisins [Sarwar et al., 2001] et la technique probabiliste [Cöster et al., 2002] sont deux formes de filtrage collaboratif basées sur la mémoire. Chaque technique de filtrage collaboratif revêt ses propres principes.

II.3.1.1. La techniques des k-plus proches voisins

La technique des k-plus proches voisins est parmi les techniques les plus adoptées dans un contexte de filtrage collaboratif. En se basant sur la notion de voisinage, cette approche permet de former une collection des parties prenantes qui sont similaires à une partie prenante p à partir de laquelle une suite d'informations quantifiées (ex. évaluations) relatives aux décisions est extraite. Ces données permettent de calculer la prédiction d'évaluation d'une décision d par la partie prenante p .

La méthode de k-plus proches voisins permet de prédire l'évaluation $P_{p,d}$, d'une décision d pour la partie prenante p . La prédiction est basée sur les évaluations précédentes V données à la décision d par les k-plus proches parties prenantes. La présente technique [Keller et al., 1985] considère un calcul de la moyenne des évaluations en fonction du nombre n des parties prenantes. Également, un calcul de score de similarité entre les parties prenantes $sim(p, p_i)$ est calculé.

$$P_{p,d} = \bar{v}_d + k \sum_{i=1}^n sim(p, p_i) (v_{i,d} - \bar{v}_i)$$

Avec :

- n : Le nombre de parties prenantes considérées
- $v_{i,d}$: L'évaluation de la décision d par la partie prenante p_i
- \bar{v}_i, \bar{v}_d : la moyenne des évaluations données par la partie prenante p_i

$$\bar{v}_i = \frac{1}{|D_i|} \sum_{j \in D_i} v_{i,j}$$

- $sim(p, p_i)$: Le score de similarité entre la partie prenante p et la p_i
- k : un coefficient de normalisation telle que la somme des mesures de similarité soit égale à 1.

Le score de similarité $sim(p, p_i)$ est une mesure de distance entre les évaluations des décisions faites par les deux parties prenantes p et p_i . Plusieurs méthodes peuvent être utilisées pour calculer $sim(p, p_i)$. Les plus populaires parmi celles-ci sont la méthode du cosinus [Breese et al., 1998] [Sarwar et al., 2001] et celle du coefficient de Pearson [Resnick et al., 1994].

II.3.1.2. La techniques probabiliste

La technique probabiliste est basée sur l'apprentissage de modèle en liant les parties prenantes p , les décisions d et leurs évaluations. Tenant compte du profil de la partie prenante et de ses anciennes évaluations, de nouvelles évaluations $p_{p,d}$ sont prédites en utilisant une approche probabiliste de recommandation collaborative [Breese et al., 1998] :

$$p_{p,d} = E(V_{p,d}) = \sum_{i=0}^n i * Pr(V_{p,d} = i | p_{p,d}, d' \in D_p)$$

Cette expression de probabilité, consiste à calculer la probabilité qu'une partie prenante p attribue une évaluation à une décision d à partir de ses évaluations précédentes. Les valeurs des évaluations sont des entiers compris entre 0 et n . Pour ce faire, plusieurs études [Breese et al., 1998] proposent l'utilisation de deux modèles, notamment, le modèle de clusters [Chickering et al., 1997] et le modèle à base de réseaux bayésiens [Oufaida et Nouali, 2009].

Les techniques de filtrage collaboratif sont largement utilisés dans les algorithmes de Netflix¹, Amazon² et les configurateurs de voitures³. Par exemple, pour recommander une décision d en rapport avec la puissance de voiture à un internaute p , un système à base de filtrage collaboratif cherche les utilités $u(pj, d)$ estimées par les internautes ayant des préférences similaires pour les voitures. Les puissances de voitures les plus appréciées par ces internautes sont recommandées à p . Afin de prédire l'évaluation $P_{p,d}$ de la puissance de voiture d pour l'internaute p , la méthode de k -proches voisins se base sur la moyenne des évaluations précédentes V données à la puissance de voiture d par les k -plus proches internautes et un score de similarité entre les internautes $sim(p, p_i)$ en tenant en compte d'autres décisions en rapport avec la couleur, l'énergie et la tapisserie... Quant à la méthode probabiliste, elle calcule la probabilité qu'un internaute p attribue une évaluation à la puissance de voiture d à partir de ses propres évaluations précédentes sur d'autres décisions via des modèles probabilistes.

La figure (2) récapitule les fondements de la recommandation basée sur le filtrage collaboratif. En effet, pour produire une liste de recommandation basée sur le filtrage

¹ Lien: www.netflix.com

² Lien : www.amazon.com

³ Lien : www.renault.fr/

collaboratif, le système se focalise principalement sur le profil de la partie prenante, les paramètres de contexte et les données de la communauté.

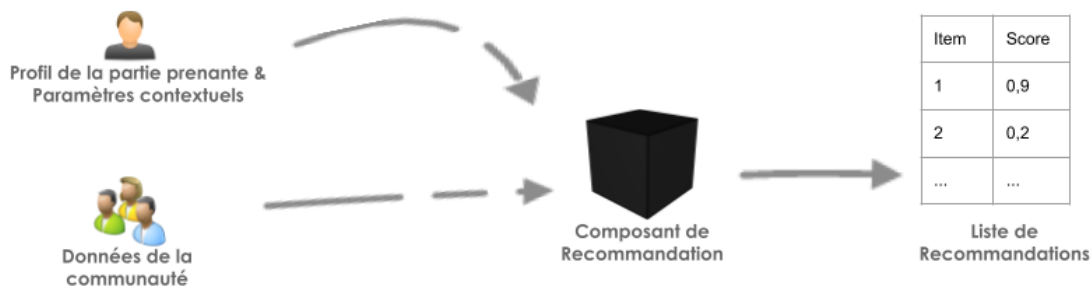


Figure (2) : Fondements de la recommandation basée sur le filtrage collaboratif

Ce paradigme de recommandation répond principalement à la question suivante: “*Qu’est ce qui est populaire parmi les choix des parties prenantes similaires ?*”

II.3.2. La recommandation basée sur le contenu

La recommandation basée sur le contenu est une dérivée du filtrage d’information [Salton, 1989]. Elle permet de recommander des décisions similaires aux décisions déjà appréciées par la partie prenante.

Formellement, elle permet de prédire l’utilité $u(p, d)$ d’une décision en se basant sur les utilités $u(p, di)$ assignées par la partie prenante p aux décisions similaires à d [Adomavicius et al., 2005]. Cette mesure de similarité est calculée à partir des descriptions, des décisions et du profil de la partie prenante en termes de caractéristiques appelées « métadonnées ». La recommandation basée sur le contenu est alors composée de :

- Descriptions des décisions à recommander en fonction de leurs variantes.
- Une représentation du profil de la partie prenante en fonction de ses préférences exprimées avec les variantes des décisions, et de son historique de décisions appréciées [Pazzani et Billsus, 2007].
- Une méthode pour comparer le profil de la partie prenante avec les décisions, et pour prédire l’utilité u .

Afin de présenter une décision d’une manière structurée, des poids sont attribués à ses différentes variantes. Soit V_i^d la variante ayant le numéro i dans la décision d . Pour chaque variante V_i^d , un poids w_i^d est associé indiquant son degré d’importance dans d . Ainsi, un vecteur de poids noté w^d est associé à chaque décision. Cette suite de vecteurs

génère un espace vectoriel associé à la décision. La “Term Frequency-Inverse Document Frequency” présente une mesure parmi les plus répandues dans la recommandation basée sur le contenu permettant de calculer la valeur des poids [Salton, 1989].

$$w_i^d = Tf_{i,d} * \log\left(\frac{n}{DF_i}\right)$$

Avec :

- n : Le nombre de décisions
- $Tf_{i,d}$: Le nombre d'occurrences de la variante V_i dans la décision d
- DF_i : Le nombre de décisions contenant au moins une fois la variante V_i

La décision est définie comme suit :

$$\text{Décision } (D) = \{(V_i^d, w_i^d)\}, i = 1..n$$

Le profil de la partie prenante peut être construit de manière explicite à partir des questionnaires sur les variantes qui l'intéresse ou pas, ou encore implicitement à partir des évaluations ou d'actions précédentes sur les décisions.

Reprenons l'exemple précédent concernant le configurateur des voitures. Un configurateur reposant sur la recommandation basée sur le contenu, analyse les variantes communes des voitures telles que la couleur et la tapisserie, appréciées par l'internaute pour lui recommander des décisions en rapport avec la puissance de voiture qui sont similaires à ses préférences. Pour ce faire, une pondération des décisions est calculée pour identifier l'importance de chacune. La pondération tient en compte le nombre d'occurrences des variantes communes, $Tf_{i,s}$ et combien de fois ces variantes étaient présentes dans d'autres décisions DF_i .

La figure (3) récapitule les fondements de la recommandation basée sur le contenu. En effet, pour produire une liste de recommandation basée sur le contenu, le système se focalise principalement sur le profil de la partie prenante, les paramètres de contexte et les caractéristiques du produit sur lequel la recommandation est envisagée.

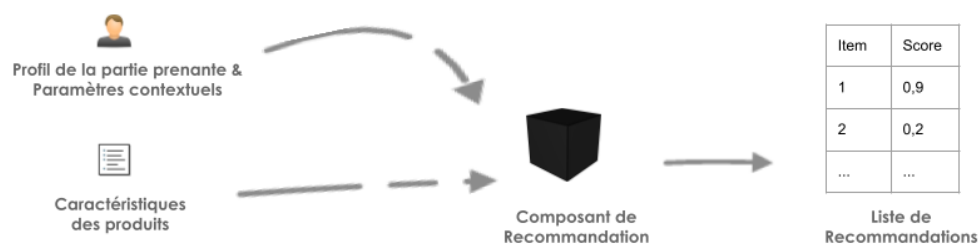


Figure (3) : Fondements de la recommandation basée sur le contenu

Ce paradigme de recommandation répond principalement à la question suivante: *“Qu’est ce qui est pertinent selon les choix que la partie prenante a déjà appréciés?”*

II.3.3. La recommandation basée sur la connaissance

L'idée clé de la recommandation basée sur la connaissance, repose sur le principe de recommander ce qui correspond le mieux en fonction des besoins de la partie prenante [Wieringa et Morali, 2012].

En utilisant différents modèles définis à partir des variantes de décisions, un configurateur reposant sur la recommandation basée sur la connaissance propose ce qui pourrait convenir à la partie prenante [Rao et Talwar, 2008]. Ce type de recommandation met en relief les besoins des parties prenantes et la manière de les satisfaire [Felfernig et Burke, 2008]. Les approches de recommandation basée sur la connaissance peuvent être classées en deux types : la recommandation basée sur les cas [Mirzadeh et al., 2005] et la recommandation basée sur les contraintes [Felfernig et al., 2007]. Dans les deux cas cités ci-avant, pour produire une recommandation, les besoins de la partie prenante sont spécifiés. Toutefois, la partie prenante est invitée à revoir ses besoins au cas où aucune solution ne serait proposée.

La recommandation basée sur les cas utilise un raisonnement par réutilisation de cas passés similaires au cas actuel pour trouver des solutions [Mirzadeh et al., 2005].

Quant à la recommandation basée sur les contraintes, elle produit des recommandations en respectant un ensemble de contraintes ou « règles de recommandation » [Felfering et al., 2014] spécifiées explicitement par la partie prenante. Les exigences, les propriétés des décisions et les contraintes forment « une base de connaissances du système de recommandation ».

Une activité de recommandation peut être considérée comme un Problème de Satisfaction des Contraintes CSP ($P, D, CR \cup COMP \cup FILT \cup DEC$) avec :

- P : un ensemble fini de variables représentant les exigences potentielles de la partie prenante.
- V : un ensemble de variables définissant les propriétés de base des décisions.
- CR : un ensemble d'exigences potentielles.
- $COMP$: un ensemble de contraintes de (in)compatibilité limitant l'ensemble d'exigences possibles.
- $FILT$: un ensemble de contraintes de filtrage définissant les relations entre les exigences de la partie prenante et les décisions.

- DEC : un ensemble de décisions proposées sous forme de contraintes de décisions limitant les instanciations possibles dans \mathcal{D} .

Reprenons l'exemple précédent concernant le configurateur des voitures. Un configurateur reposant sur la recommandation basée sur la connaissance, transforme le modèle d'une voiture sous forme d'un CSP à partir duquel un internaute effectue ses choix. Les variantes v telles que la couleur, la tapisserie, la puissance de voiture sont servies à l'internaute avec des contraintes du domaine FILTR qui réduisent l'espace de solutions. La finalité d'opter pour un CSP est de recommander à l'internaute que des choix cohérents. Par exemple, une voiture de faible puissance n'aura jamais des roues avec des dimensions larges.

La figure (4) récapitule les fondements de la recommandation basée sur la connaissance. En effet, pour produire une liste de recommandation basée sur la connaissance, le système se focalise principalement sur le profil de la partie prenante, les paramètres de contexte, les caractéristiques du produit sur lequel la recommandation est envisagée et encore le modèle de variabilité du produit.

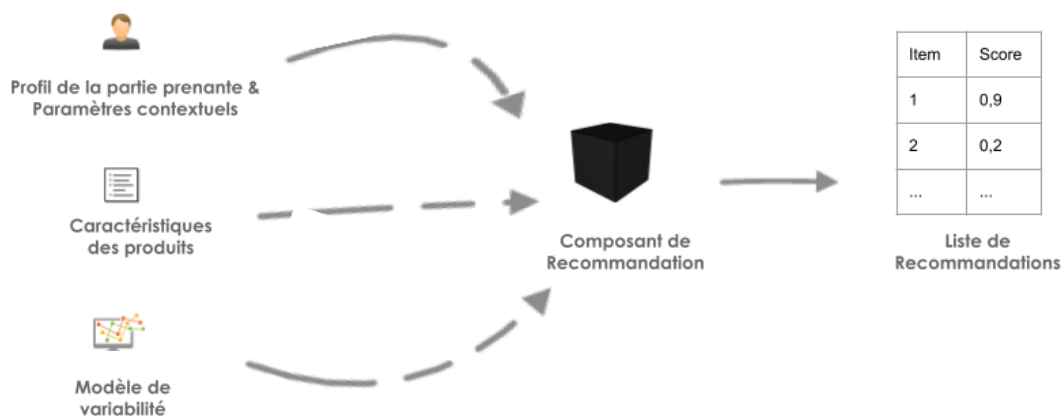


Figure (4) : Fondements de la recommandation basée sur la connaissance

Ce paradigme de recommandation répond principalement à la question suivante: “*Qu’est ce qui répond le mieux aux besoins de la partie prenante?*”

II.3.4. La recommandation heuristique

La recommandation heuristique permet de recommander des décisions de processus en se basant sur un ensemble d'heuristiques.

Dans ce contexte, six heuristiques ont été proposées [Mazo et al., 2014] afin de gérer au mieux l'interactivité de la partie prenante lors de ses prises de décisions. En particulier,

ces heuristiques sont destinées à éviter des prises de décisions inutiles ou inefficaces. Chaque méthode heuristique présente des principes, des avantages et des inconvénients. Les heuristiques sont les suivantes :

- Heuristique 1 : Les variantes avec le plus petit domaine sont traitées en premier.
- Heuristique 2 : Les variantes les plus contraignantes sont traitées en premier.
- Heuristique 3 : Les variantes obligatoires dans tous les produits sont choisies en premier.
- Heuristique 4 : Le traitement des variantes se fait automatiquement lorsqu'un seul choix est possible.
- Heuristique 5 : Les variantes requises par la dernière variante sont traitées en premier.
- Heuristique 6 : Les variantes divisant l'espace de problèmes en deux sont traitées en premier.

Contrairement à la recommandation basée sur le contenu et le filtrage collaboratif, ce paradigme de recommandation ne consiste pas à recommander une décision mais plutôt les variantes d'un produit en utilisant les heuristiques sans considérer la similarité avec les configurations passées. En effet, dans un configurateur de voiture basé sur la recommandation heuristique, l'internaute peut être appelé à configurer la puissance de la voiture en premier vu que sa décision va impliquer plusieurs propagations automatiques de choix, suivant Heuristique 2, en rapport avec la dimension des roues, le modèle de motorisation et la puissance de la climatisation.

La figure (5) récapitule les fondements de la recommandation basée sur les heuristiques. En effet, pour produire une liste de recommandation basée sur les heuristiques, le système se focalise principalement sur des heuristiques conçus au préalable et le modèle de variabilité du produit.

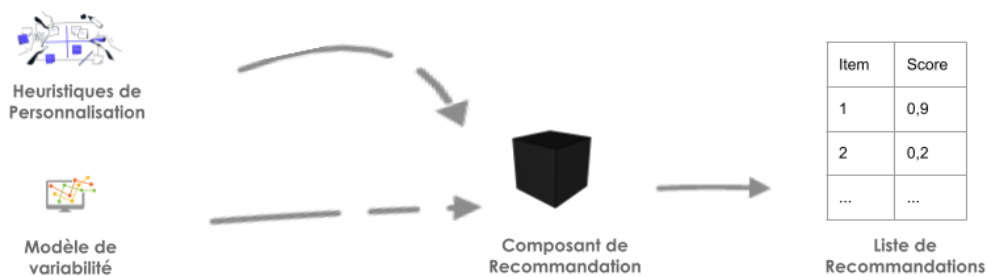


Figure (5) : Fondements de la recommandation basée sur les heuristiques

Ce paradigme de recommandation répond principalement à la question suivante: “*Quelle heuristique répond le mieux aux besoins de la partie prenante?*”

II.3.5. La recommandation hybride

La recommandation hybride combine deux ou plusieurs paradigmes de recommandation pour fournir une meilleure performance de la recommandation et surmonter les limites spécifiques de chaque paradigme [Burke, 2002].

A titre d'exemple, la combinaison du filtrage collaboratif et la recommandation basée sur le contenu est très fréquente dans la littérature. Différentes méthodes d'hybridation sont présentées [Burke, 2007]; ci-dessous les méthodes les plus répandues :

- La recommandation pondérée : Il s'agit de générer des recommandations en faisant une union ou une intersection des résultats fournis par chaque paradigme. Les résultats de chaque paradigme sont pondérés.
- La recommandation mixte : Il s'agit de combiner différentes solutions générées simultanément par plusieurs systèmes de recommandation.
- La recommandation en cascade : Il s'agit de traiter les recommandations de chaque paradigme dans une méthode hiérarchique. Les solutions générées par un paradigme de recommandation sont affinées pour être utilisées par le paradigme de recommandation suivant.

La figure (6) récapitule les fondements de la recommandation hybride.

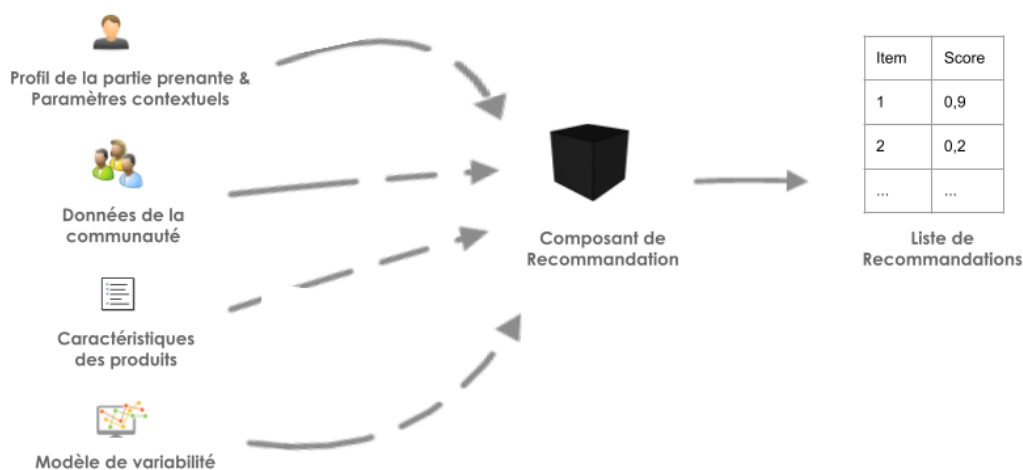


Figure (6) : Fondements de la recommandation hybride

En effet, pour produire une liste de recommandation hybride, le système se focalise sur une combinaison totale ou partielle des requis des autres paradigmes de recommandation à savoir, le profil de la partie prenante, les paramètres de contexte, les

données de la communauté, les caractéristiques du produit sur lequel la recommandation est envisagée et encore le modèle de variabilité du produit.

Ce paradigme de recommandation répond principalement à la question suivante: *“Quelle combinaison de recommandation répond le mieux aux besoins de la partie prenante?”*

Dans cette section, un panorama des paradigmes de recommandation a été présenté. La description détaillée des paradigmes de recommandation citées ci-dessus avec des exemples est illustrée dans la littérature [Jannach et al., 2010], [Mazo et al., 2014]. La section suivante met en relief la recommandation basée sur la fouille de processus et ses algorithmes.

II.4. La recommandation basée sur la fouille de processus

La fouille de processus est un axe de recherche largement emprunté, récemment, pour des fins de prédiction [Pourbafrani et al., 2019] et de recommandation [Epure et al., 2017], particulièrement dans les systèmes d'information supportant des processus lourds qui induisent en erreur. En effet, la fouille de processus s'intéresse à l'analyse de processus métiers sur la base des journaux d'événements⁴ que l'on peut extraire à partir des systèmes d'information comme les boutiques en ligne. Le but ultime de la fouille de processus est l'amélioration de la performance d'un système en offrant une figure plus complète des instances de processus.

Grâce aux progrès des systèmes et services du TIC⁵, l'évolution de la fouille de processus a été motivée par l'énorme volume de données de processus capturées dans les journaux des événements. Les journaux d'événements reflètent les faits et décrivent les traces d'exécution de chaque mise en œuvre de processus. Les traces d'exécution peuvent stocker des informations sur les événements comme l'horodatage⁶ et la ressource⁷. Les journaux des événements est une collection d'événements utilisés en entrée pour la fouille de processus. Chaque événement fait référence à une activité (c'est-à-dire, une étape bien définie dans un processus) qui est liée à un cas particulier (c'est-à-dire, une instance de processus). Le défi consiste à exploiter les traces de processus, d'une manière significative, pour produire une valeur ajoutée. En effet, en fonction de la fouille de processus, il est possible de déduire un aperçu des activités, d'identifier les goulots

⁴ Des fichiers qui enregistrent les événements significatifs selon une structure prédéfinie.

⁵ Technologies de l'Information et de la Communication

⁶ L'horodatage (appelé aussi timestamping) est un mécanisme informatique qui consiste à associer une date et une heure à un événement du processus.

⁷ La ressource est une personne ou un dispositif exécutant l'activité.

d'étranglement, d'anticiper les problèmes, de recommander des instances de processus ou de rationaliser les processus.

II.4.1. La découverte, la mesure de conformité et l'amélioration de processus

Les journaux d'événements présentent l'essence qui opérationnalise toute technique de fouille de processus. La littérature [Aalst, 2011] distingue trois techniques de fouille de processus, notamment :

- La découverte des modèles de processus,
- La mesure de conformité des processus et
- L'amélioration de processus.

La Figure (7) illustre les techniques clés de la fouille de processus.

De manière proactive, les journaux des événements enregistrent des traces de processus reflétant des interactions homme-machine.

A partir des journaux d'événements, des modèles de processus peuvent être découverts pour être vérifiés ou améliorés après. Sur la base des modèles de processus, le système d'information évolue.

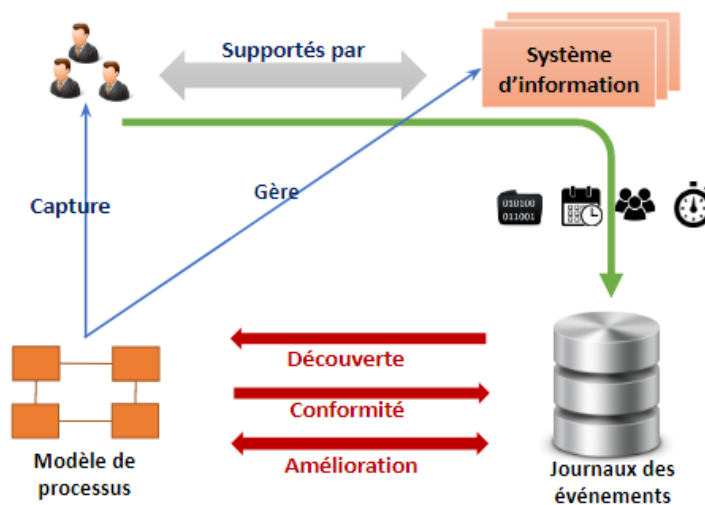


Figure (7) : Techniques de fouille de processus : Découverte, conformité et amélioration [Aalst, 2011]

II.4.1.1. La découverte de processus

Dans plusieurs processus métiers, les organisations utilisent des procédures pour gérer les cas possibles. Cependant, dans la plupart des cas, les procédures sont informelles et non documentées. De plus, même lorsque des procédures sont documentées, la réalité peut être très différente. Par conséquent, il est important de

découvrir les processus réels en utilisant les données d'événements. Les modèles de processus découverts peuvent être utilisés pour :

- Discuter des problèmes entre les parties prenantes afin de parvenir à un consensus.
- Analyser les goulots d'étranglements.
- Générer des idées d'amélioration de processus.

La découverte de processus est la première technique de fouille de processus. En outre, cette technique s'avère une tâche très complexe vu qu'elle est censée refléter la réalité dans un modèle de processus encore inconnu basé uniquement sur des journaux d'événements et n'utilisant aucune information préalable. La découverte de processus se concentre sur la récupération du flux de contrôle d'un processus basé sur les traces d'exécution sans utiliser un modèle a priori. Le défi consiste à découvrir un modèle de processus qui représente étroitement le comportement observé dans les journaux des événements, mais qui ne permet pas des comportements supplémentaires.

II.4.1.2. La mesure de conformité de processus

Au cours des dernières années, de puissantes techniques de fouille de processus ont été développées pour construire automatiquement un modèle de processus approprié à partir des journaux d'événements. Alors la découverte de processus, construit un modèle sans aucune information a priori (autre que celles venant des journaux d'événements). Cependant, la mesure de conformité utilise un modèle de processus et des journaux d'événements en entrée afin de comparer le comportement modélisé (c'est-à-dire le modèle de processus) et le comportement observé (c'est-à-dire les journaux des événements).

La finalité de la mesure de conformité est de vérifier si la réalité telle qu'elle est enregistrée dans les journaux des événements est conforme au modèle métier et vice versa. Par ailleurs, il est possible de quantifier les différences (par exemple, 80% des cas observés sont possibles selon le modèle) et diagnostiquer les écarts (par exemple, une décision prise bien que le modèle ne la permette pas). La mesure de conformité entre le comportement observé dans les journaux des événements et le comportement modélisé peut être utilisée pour :

- Vérifier la qualité de processus documentés (en évaluant s'ils décrivent la réalité avec précision),
- Identifier les cas déviants et comprendre ce qu'ils ont en commun,

- Identifier les fragments de processus où se produisent la plupart des écarts, ou
- Vérifier la qualité d'un modèle de processus découvert.

Par ailleurs, la mesure de conformité peut être utilisée pour diverses raisons, allant de l'évaluation d'un algorithme de découverte, jusqu'à l'audit des processus.

II.4.1.3. L'amélioration de processus

L'idée de l'amélioration de processus repose sur le potentiel d'étendre ou d'améliorer un modèle de processus. Cette amélioration se base sur des informations du processus réel découverts à partir des journaux d'événements. La mesure de conformité se focalise sur l'alignement entre le modèle et la réalité alors que l'amélioration de processus vise à modifier ou à étendre le modèle a priori.

Un type d'amélioration de processus est la réparation, c'est-à-dire la modification du modèle pour mieux refléter la réalité. Par exemple, si deux activités sont modélisées de manière séquentielle, mais que, dans la réalité, elles peuvent se produire dans n'importe quel ordre, le modèle peut alors être corrigé pour refléter cela.

Un autre type d'amélioration est l'extension, c'est-à-dire l'ajout d'une nouvelle perspective au modèle de processus en le corrélant avec les journaux. Un exemple est l'extension d'un modèle de processus avec des données de performance, comme l'horodatage.

II.4.2. Les algorithmes de fouille de processus

Dans un contexte de recommandation, les trois techniques de fouille de processus, à savoir la découverte de processus, la mesure de conformité et l'amélioration des processus, s'avèrent importantes pour produire des connaissances. En littérature, il existe de nombreuses méthodes associées à la production des connaissances venant de la découverte, la mesure de conformité et l'amélioration de processus [Chemingui et al., 2019]. Les méthodes locales examinent les relations locales (comme l'algorithme "Heuristic Miner" [Weijters et al., 2006]) entre les activités dans les journaux des événements, tandis que les méthodes globales (comme l'algorithme "Fuzzy Miner" [Günther et al., 2007]) construisent et affinent un modèle de processus basé sur la totalité des journaux d'événements.

De plus, chaque algorithme de fouille de processus opère par ses propres traitements. Par exemple, le "Heuristic Miner" utilise les fréquences des activités pour gérer les

activités inopinées; tandis que le “Genetic Miner” [Bratosin et al., 2010] extrait des journaux complexes et bruyants et déduit, par une approche mathématique, les problèmes et/ou les imprévus. D’autres algorithmes plus récents, se concentrent sur la gestion de modèles complexes du monde réel ou de journaux bruyants à l’aide du clustering et de l’abstraction.

Dans le reste de cette section, des définitions formelles des deux algorithmes les plus répandues pour la recommandation [Aalst, 2011] sont présentées, notamment : Le “Heuristic Miner” et le “Fuzzy Miner”.

II.4.2.1. Le Heuristic Miner

Le “Heuristic Miner” suppose qu’un journal d’événements peut contenir des informations manquantes, incorrectes ou aberrantes par rapport au contexte. Le but de cette technique est d’en déduire les comportements observés en excluant le reste des données appelées données “bruit” [Whelan et al., 2016]. L’entrée de l’algorithme “Heuristic Miner” est un journal d’événements L contenant plusieurs traces de processus.

Dans un processus, les événements sont supposés classés par ordre chronologique et, dans ce cas, les horodatages ne sont pas nécessaires. Cependant, si les événements ne sont pas classifiés, les horodatages sont nécessaires pour le tri. L’aboutissement de “Heuristic Miner” est un graphe de dépendances dans lequel les nœuds représentent les activités et les arrêts représentent les relations de dépendance entre les activités. Dériver le graphe de dépendance revient à calculer la certitude d’existence d’une relation de dépendance entre deux activités a et b identifiées dans le journal d’évènements L . Pour chaque paire d’activité, la certitude ($a \Rightarrow L^b$) est calculée en prenant en compte des séquences directes entre (a et b) et (b et a) comme suit :

$$a \Rightarrow L^b = \frac{\text{Count}(a > L^b) - \text{Count}(b > L^a)}{\text{Count}(a > L^b) + \text{Count}(b > L^a) + 1}$$

La construction d’un graphe de dépendances considère les deux principes suivants :

- Chaque activité non initiale doit avoir au moins une autre activité en tant que cause.
- Chaque activité non dernière doit avoir au moins une autre activité en tant que dépendance.

Ensuite, une sélection des meilleurs candidats des causes et des dépendances est conduite. La sélection se fait en fonction des scores de dépendance les plus élevés, calculés pour les paires qui incluent cette activité cible. Ainsi, une première version du graphe de dépendance est construite. Par ailleurs, des versions plus précises sont construites en fonction des seuils de dépendances paramétrables. Par le biais des seuils restrictifs, le “Heuristic Miner” découvre le comportement le plus représentatif. Cependant, des seuils plus élevés pourraient également conduire à la découverte de comportements moins fréquents et exceptionnels.

De la même manière, le “Heuristic Miner” est capable de détecter la concurrence entre deux activités, les boucles d’exécution, les séquences directes et indirectes, etc.

II.4.2.2. Le Fuzzy Miner

Le “Fuzzy Miner” est un algorithme de fouille de processus recommandé pour la découverte des processus qui présentent un comportement non structuré ou contradictoire. Il applique diverses techniques, telles que la suppression de tronçons sans importance, le clustering de nœuds fortement corrélés dans un seul nœud et la suppression des clusters de nœuds isolés [Gunther et al., 2007]. L’utilisation du “Fuzzy Miner” vise à résoudre des problèmes de processus non structurés qui présentent un grand nombre de comportements conflictuels et susceptibles d’influencer négativement le processus entier.

Tout comme le “Heuristic Miner”, le comportement de l’algorithme “Fuzzy Miner” est toujours représenté dans le modèle de processus qui peut être un graphe de dépendance. Les comportements moins significatifs pourraient être exclus de la représentation en fonction des seuils paramétrables. Contrairement à “Heuristic Miner”, la visualisation est personnalisée et plus expressive. Elle met en relief les activités et les relations les plus significatives.

La découverte d’un modèle de processus à travers l’algorithme “Fuzzy Miner” passe par une suite d’étapes. En effet, une première version “brute” d’un graphe de dépendances est construite. Les nœuds du graphe représentent les activités et les arrêts représentent les séquences observées dans le journal d’événements. Ensuite, des méthodes de transformation sont appliquées pour simplifier le graphe.

La première transformation correspond à la résolution des conflits de séquence d’activités. Pour chaque paire d’activités, a et b , avec des séquences directes observées dans les deux sens, $a > L^b$ et $b > L^a$, une transformation est requise. La résolution des

conflits de séquence d'activités montre, souvent, la présence de concurrence, des boucles ou d'une exception à une relation de dépendance. Pour résoudre ces conflits, une métrique d'importance relative (*rel*) d'activités est définie [Gunther et al., 2007] :

$$rel(a, b) = \frac{1}{2} * \frac{sig(a,b)}{\sum_{c \in A} sig(a,c)} + \frac{1}{2} * \frac{sig(a,b)}{\sum_{c \in A} sig(c,b)}$$

Avec :

$sig : Ax A \rightarrow \mathbb{R} \geq 0$ est l'importance d'une séquence directe

La métrique d'importance relative (*rel*) est appliquée pour chaque paire d'activités, dans les deux sens ($rel(a, b)$ et $rel(b, a)$) afin d'identifier l'importance d'une séquence dans le journal d'événements.

Pour ce faire, un seuil est défini pour marquer quelle est la valeur la plus faible de l'importance relative.

Après la résolution des conflits de séquence d'activités, une seconde méthode de transformation à savoir le filtrage des arrêts est requise. En fonction des paramètres définis de manière interactive par la partie prenante, seuls les arrêts entrants et sortants de chaque activité avec des valeurs d'importance supérieures au seuil défini sont conservés dans le graphe.

Enfin, les outils implémentant le "Fuzzy Miner" comme ProM⁸ et Disco⁹ proposent une configuration par défaut de paramètres en fonction des données d'entrée. Les paramètres de ces outils permettent de mettre en valeur ou encore faire abstraction à un aspect donné en fonction des préférences de l'utilisateur. L'adaptation des paramètres permet de suivre l'exhaustivité d'un comportement observé dans le journal, d'une part, et de fouiller plusieurs représentativités détaillées et abstraites des processus, d'autre part.

En guise de comparaison, le "Heuristic Miner" semble plus approprié pour les processus structurés ou pour les processus avec un faible nombre d'activités. Cependant, le "Fuzzy Miner" permet une analyse interactive et exploratoire pour les processus non structurés et incomplets incluant une multitude d'activités à savoir les activités de configuration de lignes de produits. Les deux algorithmes, le "Heuristic Miner" et le "Fuzzy Miner" permettent de découvrir des modèles de processus à partir des journaux d'événements. La principale différence réside dans la mesure d'importance/certitude des nœuds et des

⁸ Lien : www.promtools.org/doku.php

⁹ Lien : www.fluxicon.com

arrêts. En effet, le “Fuzzy Miner” permet d’avoir une représentation graphique et dynamique des étapes du processus de configuration. Par ailleurs, il met en valeur l’ordre suivi par l’acteur de la configuration et permet d’en tirer des comportements habituels. En outre, le “Fuzzy Miner” permet de filtrer le journal d’évènements et d’instancier des modèles de processus répondant à des questions précises (par exemple, le modèle de processus représentant l’ordre et les variantes des processus complets exécutés pendant la dernière semaine).

La description détaillée de l’algorithme “Heuristic Miner” et “Fuzzy Miner” avec des exemples est illustrée dans la littérature [Whelan et al., 2016].

II.5. Motivations pour la configuration des lignes de produits

La configuration des lignes de produits fait intervenir plusieurs préoccupations principalement la prise de décisions sur des variantes interdépendantes et le contrôle de la cohérence de ces décisions vis-à-vis d’un modèle de variabilité. En effet, dans les larges modèles de variabilité des problèmes de passage à l’échelle, engendrant des performances souvent médiocres, se posent.

Par analogie, les paradigmes classiques de recommandation, principalement, la recommandation collaborative et la recommandation basée sur le contenu, présentent des potentiels pour remédier aux problèmes susmentionnés. En effet, ces paradigmes de recommandation sont largement adoptés dans les boutiques en ligne présentant, parfois, des préoccupations analogues à celles du processus de configuration des lignes de produits. Néanmoins, les solutions de guidage du processus de configuration envisagées nécessitent, entre autres, un raisonnement significatif sur les processus plutôt qu’un raisonnement sur les données. Généralement, la configuration finale d’une ligne de produits passe par un nombre important de décisions intermédiaires. Plus le modèle de la ligne de produits est important, plus le nombre de décisions à prendre est multiplié. Par ailleurs, la suggestion des instances de décisions et de leur ordre dans l’ensemble du processus de configuration, s’avère fructueuse.

Par ailleurs, ayant des avantages adéquats aux problèmes observés dans la configuration des lignes de produits (cf. chapitre I), l’applicabilité de la fouille de processus est motivée par ses aboutissements suivants :

- La présentation de la chronologie et de la fréquence des décisions au sein des processus de configuration antérieurs.

- La mise en évidence de comment les décisions de configuration sont reliées les unes aux autres.
- L'analyse et la comparaison des décisions de configuration sur une période donnée et/ou pour des parties prenantes données.
- La catégorisation des décisions de configuration en fonction de leurs propriétés selon des données tracées, par exemple classifier les processus de configuration antérieurs en fonction de leur similitude, ou de leur temps de finalisation.
- La simulation permettant de tirer des conclusions sur des modèles de processus de configuration découverts, les tendances et les obstacles dans le déroulement du processus. Il est ainsi possible d'identifier des engorgements dans le processus de configuration.

Étant approprié à des processus incluant une pléthore de décisions et n'ayant pas toujours le même ordre, l'utilisation de l'algorithme de fouille de processus, le "Fuzzy Miner" est motivante. Le chapitre III analyse les solutions existantes pour le guidage du processus de configuration des lignes de produits et détaille leurs insuffisances. Par ailleurs, ces insuffisances doivent motiver davantage l'appel à une stratégie de fouille de processus de configuration des lignes de produits.

II.6. Conclusion

Dans ce chapitre, un background des systèmes de recommandation et de la fouille de processus a été présenté.

La littérature distingue plusieurs paradigmes de recommandation à savoir :

- La recommandation collaborative : Elle regroupe des méthodes qui visent à recommander des variantes en utilisant leurs évaluations par les parties prenantes.
- La recommandation basée sur le contenu: Elle regroupe des méthodes qui visent à recommander des variantes semblables à celles que la partie prenante a préférées dans le passé.
- La recommandation basée sur la connaissance : Elle regroupe des méthodes qui visent à collecter le maximum d'informations sur les exigences de la partie prenante pour pouvoir ensuite lui recommander des variantes cohérentes.
- La recommandation heuristique : Elle consiste à recommander à la partie prenante des choix en fonction des heuristiques prédéfinies pour l'accélération et/ou la facilitation des décisions.

- La recommandation hybride : Elle consiste à fusionner deux ou plusieurs paradigmes dans le but d'optimiser la recommandation.

Bien que la fouille de processus ait pour objet majeur la détection des anomalies et la mesure de conformité, de récentes recherches l'adoptent pour des fins de prédiction et de recommandation. Pour ce faire, les traces des parties prenantes sont capturées et enregistrées dans un journal d'évènements à partir duquel des modèles de processus sont découverts. La découverte des modèles de processus suit des algorithmes de fouille à savoir le "Fuzzy Miner" et le "Heuristic Miner". Ces algorithmes permettent d'abstraire le journal d'évènements et générer une représentation graphique à partir de laquelle des mesures de correction et/ou amélioration de processus peuvent être interprétées. Par ailleurs, le "Fuzzy Miner" effectue une abstraction d'évènements qui est adaptée aux processus mal structurés et incompréhensibles. En effet, les interprétations des modèles de processus de "Fuzzy Miner" revêt deux dimensions, notamment : une amélioration de l'ordre de processus des décisions et une idée plus claire sur les instances de décisions. Dans le reste de ce manuscrit, la recommandation à base de la fouille de processus est considérée comme étant un paradigme de recommandation en soi.

Le chapitre suivant détaille les problèmes de configuration de lignes de produits et les solutions existantes.

[Cette page est intentionnellement laissée en blanc]

Chapitre III : Etat de l'art

III.1. Introduction	37
III.2. Méthode de recherche adoptée	37
III.3. Mise en oeuvre de la méthode	38
III.3.1. Définition des mots clés de recherche	38
III.3.2. Définition des sources	38
III.3.3. Sélection et quantification des articles	39
III.4. Résultats	40
III.4.1. Problèmes du processus de configuration des lignes de produits	40
<i>III.4.1.1. La complexité de la décision</i>	<i>40</i>
<i>III.4.1.2. Le manque de la performance</i>	<i>42</i>
<i>III.4.1.3. Le besoin de guidage</i>	<i>43</i>
III.4.2. Solutions du processus de configuration des lignes de produits	44
<i>III.4.2.1. Les méthodes de configuration interactive</i>	<i>45</i>
<i>III.4.2.2. Les méthodes de configuration basées sur la recommandation</i>	<i>45</i>
<i>III.4.2.3. Les méthodes de configuration combinées</i>	<i>46</i>
<i>III.4.2.4. Les méthodes de configuration heuristiques</i>	<i>47</i>
<i>III.4.2.5. Les méthodes de configuration basées sur la fouille</i>	<i>47</i>
III.5. Discussions et Positionnement	48
III.6. Conclusion	51

Chapitre III

Etat de l'art

III.1. Introduction

Ce chapitre présente une revue de la littérature [Kitchenham et al., 2009] des problèmes de configuration de lignes de produits et de solutions existantes. La Section 2 présente la méthode de recherche adoptée pour conduire cette revue de littérature. La Section 3 détaille les problèmes du processus de configuration évoqués dans la littérature ainsi que les solutions existantes. La Section 4 présente une discussion des insuffisances des solutions et définit la portée de la méthode envisagée. Quant à la Section 5, elle conclut le chapitre.

III.2. Méthode de recherche adoptée

Le nombre de travaux traitant les processus de configuration de lignes de produits est en hausse importante depuis les années 2000 [Pohl et al., 2005]. La présente revue de littérature se focalise sur un aspect particulier notamment les problèmes confrontés par les parties prenantes pendant le processus de configuration. Pour ce faire, la méthode de recherche adoptée repose sur les deux questions de recherche *LQR* suivantes :

LQR1. "Quels sont les problèmes du processus de configuration des lignes de produits?"

LQR2. "Quelles sont les solutions existantes pour guider le processus de configuration des lignes de produits?"

A cet effet, la revue de littérature est conduite suivant les étapes listées ci-dessous :

1. La définition des questions de recherche,
2. La définition des mots clés de recherche (cf. section 3.1),
3. La définition des sources (cf. section 3.2),
4. La sélection et quantification des documents (cf. section 3.3) et
5. L'extraction des données et des synthèses (cf. section 4 et 5).

III.3. Mise en oeuvre de la méthode

La mise en oeuvre de la méthode de recherche inclut la définition des mots clés de recherche, la définition des sources, la sélection et la quantification des papiers.

III.3.1. Définition des mots clés de recherche

Le choix des mots clés a été fait en se référant aux titres et aux résumés des papiers. Une liste de références, de journaux et de conférences les plus importants a été sélectionnée. La revue inclut des publications scientifiques en rapport avec les problèmes du processus de configuration et les solutions apportées par la communauté scientifique. Pour le balayage et la construction d'un répertoire de publications scientifiques, les mots suivants ont été utilisés comme chaînes de recherche : "product line configuration problems", "product line configuration shortcomings", "product line configuration issues", "product line configuration difficulties", "product line configuration solutions", et "recommendation for product configuration".

III.3.2. Définition des sources

La sélection des articles s'est focalisée sur la période allant de l'année 2000 jusqu'à l'année 2020. En utilisant Google Scholar et la rubrique de recherche gratuite de DBLP, les chaînes de recherche ont été introduites. Par ailleurs, Google Scholar a été utilisé pour rechercher des publications qui ont cité les papiers déjà sélectionnés.

De ce fait, des papiers supplémentaires ont été collectés. Le balayage des articles a permis la sélection de 54 publications liées à des journaux, des conférences et des ateliers de recherche. La fréquence et la distribution de ces publications sont détaillées dans le Tableau (1).

Tableau (1) : Distribution des publications sélectionnées en fonction des sources

Source : Nom du Journal/Conférence/Atelier de Recherche	Papiers retenus
International Systems and Software Product Line Conference	11
International Working Conference on Variability Modelling of Software-Intensive Systems	3
International Conference on Research Challenges in Information Science	3
Advances in Intelligent Systems and Computing Journal	1
Computer Standards & Interfaces Journal	1
IEEE Access Journal	1
IEEE Internet Computing Journal	1
Information and Software Technology Journal	1

Information Systems Journal	1
International Production Research Journal	1
Transactions on Software Engineering and Methodology Journal	1
Internet computing Journal	1
Journal of ACM Transactions on Information Systems	1
Journal of User Modeling and User-Adapted Interaction	1
Journal of Software: Evolution and Process	1
Marketing Science Journal	1
Science of Computer Programming Journal	1
AAAI Conference on Artificial Intelligence	1
Asia Pacific Software Engineering Conference	1
Business-Process and Information Systems Modeling Conference	1
Complex Systems Design & Management Conference	1
Digital Libraries Conference	1
International Generative Programming: Concepts and Experiences Conference	1
International Principles and Practice of Constraint Programming Conference	1
International Requirements Engineering Conference	1
International Software Engineering and Knowledge Engineering Conference	1
International Software Testing, Verification and Validation Conference	1
Recommendation Systems in Software Engineering Edition	1
System Sciences Annual Hawaii Conference	1
Journées des lignes de produits	2
Applied computing Symposium	1
Record for the Artificial Intelligence community Workshop	1
Record for Sigmod community Workshop	1
International Business Modeling and Software Design Symposium	1
International Combining Modelling and Search-Based Software Engineering Workshop	1
Symposium on Software Components, Architectures, and Reuse	1
Visualisation in Software Product Line Engineering (ViS-PLE) Workshop	1

III.3.3. Sélection et quantification des articles

Le processus de sélection de pertinence a permis d'identifier 52 articles jugés fortement liés au contexte présent. La sélection a reposé sur des directives d'inclusion et d'exclusion, notamment :

- L'inclusion des études qui ont eu lieu entre l'année 2000 et l'année 2020,
- L'inclusion des études analytiques et expérimentales des problèmes de configuration,

- L'exclusion des papiers qui ne sont pas disponibles en format électronique,
- L'exclusion des papiers écrits en langue, autre que l'anglais ou le français: Par exemple, les papiers écrits en espagnol ou en allemand ont été exclus.

III.4. Résultats

La présente section concerne l'étape extraction des données. En premier lieu, elle souligne les problèmes du processus de configuration confrontés dans la littérature. Ensuite, elle identifie les solutions existantes pour faire face à ces problèmes.

III.4.1. Problèmes du processus de configuration des lignes de produits

Une grande similitude est retrouvée entre les problèmes signalés dans les articles recensés, entre autres, l'incompréhension des choix, la pléthore des variantes, la lourdeur d'exécution des décisions, etc. En effet, certains articles utilisent des mots différents pour exposer le même problème. Par conséquent, trois classes de problèmes ont été identifiés, notamment :

- La complexité de la décision,
- Le manque de la performance et
- Le besoin de guidage.

III.4.1.1. La complexité de la décision

La complexité de la décision durant le processus de configuration présente une classe de problèmes largement évoquée dans la littérature. Plusieurs problèmes en découlent, à savoir :

- L'explosion combinatoire des variantes entraînant des décisions conflictuelles,
- La qualité de la configuration : inefficacité, incomplétude et manque de flexibilité
- L'impact des décisions interdépendantes

Le Tableau (2) référence les problèmes de complexité de la décision, durant le processus de configuration des lignes de produits, identifiés dans les articles recensés.

Tableau (2) : Complexité de la décision durant le processus de configuration des lignes de produits

Problème	Références
Classe 1 : La complexité de la décision	
Explosion combinatoire des variantes entraînant des décisions	[Afzal et al., 2020], [Yao et al., 2019], [Xiang et al., 2018], [Noorian et al., 2017], [Ochoa et al., 2017], [Hayashi et al., 2017], [Zhan et al., 2017], [Pereira et al., 2017], , [Afzal et al., 2016], [Lu et al., 2016], [Sion et al., 2016],

conflictuelles	[Sinnhofer et al., 2016], [Bagheri et al., 2014], [Noorian et al., 2014], [Mazo et al., 2014], [Zhang, 2013], [Guana et al., 2013], [Dumitrescu et al., 2013b], [Zhang et al., 2010], [Deneckere et al., 2010], [Elsner et al., 2010], [Mendonca et al., 2008], [Mendonca et al., 2007], [Botterweck et al., 2007], [Guo, 2006]
La qualité de la configuration : inefficacité, incomplétude et manque de flexibilité	[Afzal et al., 2020], [Horcas et al., 2017], [Noorian et al., 2017], [Ochoa et al., 2017], [Hayashi et al., 2017], [Afzal et al., 2016], [Lu et al., 2016], [Sion et al., 2016], [Sinnhofer et al., 2016], [Chastel et al., 2015], [Zhang, 2013], [Dumitrescu et al., 2013a], [Urli et al., 2012], [Elsner et al., 2010], [Botterweck et al., 2007], [Guo, 2006]
L'impact des décisions interdépendantes	[Sion et al., 2016], [Mazo et al., 2014], [Triki et al., 2014], [Mendonca et al., 2007]

La présente revue de littérature montre que plusieurs travaux, s'intéressant aux problèmes de la configuration des lignes de produits, évoquent *l'explosion combinatoire de variantes* entraînant des processus sujets aux erreurs et des décisions incompatibles (cf. ligne 3, tableau 2). Les grands modèles de lignes de produits peuvent avoir plusieurs centaines de variantes, et l'espace de la solution peut avoir plusieurs milliers ou plus de constituants. Ainsi, il devient difficile de comprendre les multiples décisions à prendre parmi un large éventail de variantes. La partie prenante ne sait plus par où commencer et où finir. Le deuxième problème le plus souvent mentionné, en rapport avec la complexité des décisions, concerne *la qualité de la configuration*, notamment: l'inefficacité, la non exhaustivité et le manque de flexibilité (cf. ligne 4, tableau 2). Les parties prenantes sont confrontées à des modèles de lignes de produits et n'en déduisent pas exactement ce qu'elles veulent lors de la configuration. Par conséquent, ils quittent le processus avant d'achever toutes les décisions à prendre car les choix proposés ne sont pas flexibles et se retournent parfois vers des concurrents. Le troisième problème, concernant la complexité des décisions, est *la nécessité de révéler l'impact des décisions* puisque la configuration est basée sur un modèle de lignes de produits contenant des variantes très contraignantes (cf. ligne 5, tableau 2). À chaque étape de la configuration, les choix peuvent entraîner la propagation automatique de la sélection ou le rejet d'autres variantes, une propagation qui peut avoir un impact négatif sur les décisions en aval. Il est donc crucial de notifier la partie prenante des impacts de ses actions avant d'être confrontée à un espace de solutions réduit et qui ne correspond ni à ses intentions, ni à ses exigences.

III.4.1.2. Le manque de la performance

Le manque de performance des outils de configuration présente encore une classe de problèmes signalés non seulement en littérature mais aussi par les industriels. Le manque de performance revient généralement à deux problèmes, à savoir :

- L'annulation et la répétition des choix et
- La lourdeur du temps d'exécution des décisions.

Le Tableau (3) référence les problèmes de manque de performance, durant le processus de configuration des lignes de produits, identifiés dans les articles recensés.

Tableau (3) : Manque de performance durant le processus de configuration des lignes de produits

Problème	Références
Classe 2 : Le manque de la performance	
L'annulation et la répétition des choix	[Mazo et al., 2014], [Sayyad et al., 2013], [Elsner et al., 2010]
La lourdeur du temps d'exécution des décisions	[Horcas et al., 2017], [Noorian et al., 2017], [Ochoa et al., 2017], [Hayashi et al., 2017], [Afzal et al., 2016], [Sinnhofer et al., 2016], [Bagheri et al., 2014], , [Mazo et al., 2014], [Triki et al., 2014], [Dumitrescu et al., 2013a], [Sayyad et al., 2013], [Guana et al., 2013], [Elsner et al., 2010]

Généralement, les parties prenantes sont amenées à *annuler des choix et à répéter des cycles provoquant des retours en arrière* (cf. ligne 3, tableau 3). En effet, cela revient à une incompréhension des choix ou une mauvaise présentation des variantes. Techniquement, l'annulation de choix accorde une charge énorme sur le configurateur, particulièrement dans les premières étapes de configuration. Par conséquent, la performance est intensément réduite. Plusieurs recherches (cf. ligne 4, tableau 3) ont révélé que *prendre trop de temps*, pour proposer des combinaisons valides, peut constituer un obstacle important au processus de configuration. D'un point de vue marketing, il est inacceptable de faire attendre la partie prenante pendant plusieurs secondes pour savoir si ses exigences sont correctes ou non en termes de configuration. Dans la plupart des cas, les parties prenantes ont besoin de configurations valides avec des retours rapides, ce qui réduit les délais. Ce problème est fréquent, en particulier dans les grandes lignes de produits qui nécessitent des calculs complexes pour atteindre une configuration valide. Le fait que la recherche de configuration complète soit valide est un problème NP-difficile [Astesana et al. 2010], le calcul d'une telle configuration croît quadratiquement. Ces

chercheurs estiment qu'il est essentiel de réfléchir aux solveurs élastiques, qui peuvent évoluer quelle que soit la taille du modèle de la ligne.

III.4.1.3. Le besoin de guidage

Le besoin de guidage durant le processus de configuration présente encore une classe de problèmes très soulignée dans les articles recensés entre 2010 et 2020 vu :

- Le manque de support de la partie prenante,
- Le besoin de prédiction et de recommandation,
- Le besoin d'automatisation des choix,
- Le besoin de guidage interactif et visuel,
- Le manque d'idées et des informations des impacts de décisions,
- Le manque de personnalisation,
- Le besoin d'élicitation des buts, des besoins et des préférences,
- Le besoin de mapper les buts avec les variantes du modèle de la ligne.

Le Tableau (4) référence les problèmes derrière le besoin de guidage, durant le processus de configuration des lignes de produits, identifiés dans les articles recensés.

Tableau (4) : Besoin du guidage durant le processus de configuration des lignes de produits

Problème	Références
Classe 3 : Le besoin de guidage	
Le manque de support de la partie prenante	[Feichtinger et al., 2020], [Pereira et al., 2017], [Sion et al., 2016], [Sinnhofer et al., 2016], [Chastel et al., 2015], [Bagheri et al., 2014], [Mazo et al., 2014], [Triki et al., 2014], [Guana et al., 2013], [Deneckere et al., 2010], [Mendonca et al., 2008], [Dumitrescu et al., 2013b], [Urli et al., 2012]
Le besoin de prédiction et de recommandation	[Feichtinger et al., 2020], [Pereira et al., 2017], [Noorian et al., 2014], [Mazo et al., 2014], [Triki et al., 2014], [Zhang, 2013], [Chastel et al., 2015]
Le besoin d'automatisation des choix	[Afzal et al., 2020], [Noorian et al., 2017], [Zhan et al., 2017], [Guana et al., 2013], [Mazo et al., 2014]
Le besoin de guidage interactif et visuel	[Pereira et al., 2017], [Lu et al., 2016], [Sion et al., 2016], [Triki et al., 2014], [Urli et al., 2012], [Botterweck et al., 2007], [Cirilo et al., 2007], [Schmid et al., 2006]
Le manque d'idées et des informations des impacts de décisions	[Sion et al., 2016], [Triki et al., 2014], [Mendonca et al., 2007]

Le manque de personnalisation	[Horcas et al., 2017], [Pereira et al., 2017], [Triki et al., 2014], [Elsner et al., 2010], [Deneckere et al., 2010]
Le besoin d'élicitation des buts, des besoins et des préférences	[Noorian et al., 2017], [Afzal et al., 2016], [Bagheri et al., 2014], [Noorian et al., 2014], [Sinnema et al., 2004]
Le besoin de mapper les buts avec les variantes du modèle de la ligne	[Pereira et al., 2017], [Sinnhofer et al., 2016], [Bagheri et al., 2014], [Noorian et al., 2014], [Dumitrescu et al., 2013b]

Les recherches référencées dans le tableau (4), sont allées plus loin, au-delà de l'appel explicite de solutions de guidage pertinentes dans les processus de configuration d'une ligne de produits. En effet, ces études (cf. ligne 3, tableau 4) ont mentionné le *manque d'assistance fournie aux parties prenantes* lorsqu'elles configurent un produit. Pour pallier ces insuffisances, certaines recherches (cf. ligne 4, tableau 4) ont mentionné la nécessité d'utiliser des techniques de *prévision et de recommandation* lors du processus de configuration (par exemple, recommander des valeurs de variantes en fonction de prévisions basées sur des mesures de similarité). D'autres recherches (cf. ligne 5, tableau 4) ont mentionné la nécessité de *propagation de quelques décisions automatiques*, en particulier dans des situations où il n'y a pas de choix. Une autre manière d'améliorer le processus de configuration (cf. ligne 6, tableau 4) consiste à fournir aux parties prenantes des *interfaces ergonomiques avec un support visuel interactif*. Pour révéler l'impact des décisions, certaines études (cf. ligne 7, tableau 4) soulignent la nécessité de fournir aux parties prenantes *des informations et des messages sur les conséquences de leurs actions*. La personnalisation, constitue le motif principal d'adoption d'une méthode d'ingénierie de lignes de produits. Toutefois, proposer aux parties prenantes des processus sans guidage les amène à *des choix qui manquent de personnalisation* (cf. ligne 8, tableau 4). D'autres recherches (cf. ligne 9, tableau 4) incitent à *élucider les exigences, les préférences et les buts de chaque partie prenante* pour améliorer le processus de configuration. En outre, d'autres recherches (cf. ligne 10, tableau 4) soulignent *la nécessité de "mapper" les exigences, préférences et buts avec les variantes du modèle de la ligne de produits correspondant*.

III.4.2. Solutions du processus de configuration des lignes de produits

Les lacunes du processus de configuration de lignes de produits ont été abordées par la communauté du génie logiciel qui a proposé une panoplie de solutions telles que, des méthodes de configuration interactives [Benavides et al., 2010], [Hadzic et al., 2004], des

méthodes basées sur des recommandations [Jannach et al., 2011], une combinaison de ces deux méthodes [Triki et al., 2014], des méthodes heuristiques [Mazo et al., 2014] et des méthodes basées sur la fouille [Varela-Vaca et al., 2019].

III.4.2.1. Les méthodes de configuration interactive

Comme leur nom l'indique, les méthodes de configuration interactive proposent un support à partir duquel la partie prenante interroge un modèle de la ligne de produits. Dans ce genre de méthodes, les spécifications des lignes de produits sont mappées dans un langage exécutable [Krueger et al., 2013] [Cirilo et al., 2007], par exemple : des programmes de contraintes de domaine fini [Salinesi et al., 2011] et des formules propositionnelles [Batory et al., 2005] [Acher et al., 2013]. Le but de cette transformation est de propager automatiquement les décisions de la partie prenante et de fournir une solution valide. D'autres études, [Dumitrescu et al., 2013 a] [Dumitrescu et al., 2013 b] ont révélé que les méthodes interactives sont efficaces pour aider les parties prenantes à identifier les décisions incohérentes, mais que la probabilité de décisions incohérentes était si élevée et que cette approche est pratiquement impossible à utiliser lorsque les parties prenantes n'ont pas déjà une connaissance approfondie des systèmes à configurer.

III.4.2.2. Les méthodes de configuration basées sur la recommandation

Dans les méthodes de configuration basées sur la recommandation, les configurations reposent principalement sur une sélection de variantes basée sur un ou plusieurs paradigmes de recommandation. Souvent, ces paradigmes de recommandation permettent de résoudre le problème d'explosion combinatoire confronté dans le cas d'une configuration sans guidage. Dans ce contexte, plusieurs méthodes de sélection de variantes ont été proposées dans la littérature, à savoir : la sélection basée sur les scores [Falkner et al., 2011], la sélection collaborative [Herlocker et al., 2004] [Linden et al., 2003], la sélection inclusive ou exclusive [Falkner et al., 2011], la sélection basée sur la fréquence [Mirzadeh et Ricci, 2005] [Mooney et al., 2000], la sélection basée sur l'entropie [Witten et Frank, 2002] et la sélection basée sur une combinaison de la popularité et de l'entropie [Mirzadeh et Ricci, 2005].

- La sélection inclusive ou inclusive : Dès que la partie prenante sélectionne une variante du modèle de la ligne, le système inclut ou exclut toutes les variantes ayant des contraintes de domaine avec cette variante. En outre, toutes les

variantes obligatoires seront sélectionnées automatiquement, également pour les choix exclusifs là où une seule variante est permise.

- La sélection basée sur les scores : Des scores d'importance sont affectés aux variantes du modèle de la ligne de produits selon leur importance pour la partie prenante en vue de favoriser la configuration, en premier, des variantes les plus pertinentes.
- La sélection collaborative : Les parties prenantes laissent des évaluations sur les variantes de la ligne de produits. Une mesure de similarité entre les parties prenantes est conduite. En fonction de ces évaluations, le système recommande les variantes bien évaluées par des parties prenantes similaires.
- La sélection basée sur la fréquence : Une métrique de fréquence de la sélection des variantes du modèle dans les autres configurations est calculée. Généralement, le calcul de la fréquence s'effectue en fonction du nombre de sélection de cette variante et du nombre de sélections des autres variantes. Les variantes étant les plus sélectionnées sont les premières à être recommandées.
- La sélection basée sur l'entropie : L'entropie représente l'information donnée par une variante du modèle. Elle permet de quantifier les informations fournies par cette variante afin d'identifier le produit final à configurer. La variante ayant l'entropie la plus élevée est recommandée en premier.
- La sélection basée sur une combinaison de la fréquence et de l'entropie : Cette méthode permet de surmonter les limites de la sélection basée sur l'entropie, notamment, la pertinence des variantes recommandées. Ce type de sélection basée, recommande les variantes ayant le plus d'informations disponibles et qui sont les plus sélectionnées.

III.4.2.3. Les méthodes de configuration combinées

Comme leur nom l'indique, les méthodes combinées regroupent les méthodes interactives et les méthodes basées sur la recommandation. Plusieurs travaux [Triki et al., 2014] [Tran et Cohen., 2000] [Burke, 2002] proposent une configuration interactive en organisant le processus de configuration en une série d'étapes. Les variantes à configurer à chaque étape sont recommandées en fonction d'une similitude entre les décisions antérieures et tous les autres processus de configuration effectués auparavant. Ce genre d'approche présente deux inconvénients importants. Premièrement, ces méthodes souffrent d'un problème de démarrage à froid, qui est typique dans toutes les stratégies de

filtrage collaboratif. Deuxièmement, cela crée un renforcement cognitif : les parties prenantes reçoivent des recommandations standardisées qui ne correspondent pas forcément à leurs attentes.

III.4.2.4. Les méthodes de configuration heuristiques

Dans les méthodes heuristiques, la partie prenante est guidée tout au long de la configuration en utilisant des heuristiques visant à réduire le nombre d'actions ou à minimiser l'impact des décisions [Mazo et al., 2014]. En outre, les heuristiques visent à aider la partie prenante à spécifier pas à pas les variantes en fonction de leurs besoins (cf. chapitre II, section 3). L'intérêt principal de la recommandation basée sur les heuristiques est d'améliorer les performances de la configuration en évitant les décisions inefficaces. Cependant, d'autres problèmes de configuration tels que la personnalisation du guidage semblent conscients mais pas surmontés. Cette approche ne recommande pas de processus mais elle recommande des variantes en fonction de plusieurs heuristiques.

III.4.2.5. Les méthodes de configuration basées sur la fouille

Les méthodes de configuration basées sur la fouille regroupent les méthodes basées sur la fouille de données et les méthodes basées sur la fouille de processus. Véritablement, la fouille de processus est au croisement de deux approches à savoir (i) l'approche donnée incluant la structuration, la centralisation et la visualisation et (ii) l'approche processus incluant la modélisation et l'exécution des enchaînements de séquences d'événements.

Les méthodes de configuration basées sur la fouille de données [Chastel et al., 2015] [Zhang, 2013] [Witten et Frank, 2002] utilisent la fouille de données pour l'amélioration des prochains processus de configuration. Généralement, ce genre de méthodes capture les traces de configuration afin de les exploiter, ultérieurement, pour identifier des corrélations de variantes complexes et des règles d'association capables de produire des recommandations. Ces méthodes suggèrent aux parties prenantes l'action suivante lorsqu'ils construisent une configuration en utilisant des algorithmes d'intelligence artificielle. Pour ce faire, ils capturent les traces de produits de configuration et les classent à l'aide des mesures de clustering afin de déterminer quel est le cluster le plus proche. Cependant, ces méthodes fournissent également aux parties prenantes des suggestions sur une seule action à effectuer et non sur une série d'actions à suivre. De plus, l'action suggérée est basée sur sa fréquence dans les traces et ne prend en compte aucun objectif de guidage.

Par ailleurs, une méthode de configuration des lignes de produits basée sur la fouille de processus [Varela-Vaca et al., 2019] est identifiée dans la présente revue de littérature. Cette méthode se focalise sur la recommandation d'un "workflow" de configuration, en d'autres termes, l'ordre et la séquence dans lesquels les différents éléments de configuration sont présentés aux parties prenantes. Pour ce faire, les actions de configuration sont capturées et exploitées, ultérieurement, pour déterminer le workflow de configuration qui correspond le mieux aux traces capitalisées. Cependant, le "workflow" proposé dans les prochaines configurations ne considère que la fréquence des variantes du modèle de la ligne ce qui fait que la séquence d'ordre à proposer est standard et ne tient pas en compte les différences des parties prenantes.

D'autres méthodes [Hubaux et al., 2009] proposent d'assister les configurations complexes en fournissant des processus de configuration intelligents, les notions de vues et des "workflows" ont été définies ainsi. Ce type de guidage est fourni en particulier pour gérer les configurations collaboratives qui impliquent différentes parties prenantes avec des préoccupations différentes. Néanmoins, cette méthode présente des limites, car elle ne tient pas compte des intentions de la partie prenante et ne se concentre pas sur les comportements d'exécution des processus. Les solutions de guidage proposées sont standards. En effet, celles-ci ne s'adaptent pas à la variété des besoins de guidage et des situations auxquelles sont confrontées les collaborateurs de configuration. De plus, elles offrent des choix limités qui font que les parties prenantes soient confrontées à des processus inflexibles; cela peut suggérer des valeurs de variantes mais jamais un chemin et un ordre de processus de configuration.

III.5. Discussions et Positionnement

La présente section concerne l'étape d'extraction des synthèses.

Les limites révélées pour chaque méthode de recommandation motivent l'idée d'adopter une stratégie de fouille de processus qui vise à analyser l'ensemble du processus de configuration. Par ailleurs, le but est de guider les futures configurations en suggérant, à chaque étape, des valeurs pour un élément du modèle d'une ligne de produits donnée d'une part, et des chemins de configuration ou une séquence d'activités, dans un ordre prédéfini, d'autre part. De plus, la solution de guidage envisagée doit être dirigée par des buts; qu'elle prenne en compte un but de guidage, un état de processus et une situation donnée.

Le tableau (5) compare les solutions existantes et souligne la solution envisagée dans ce manuscrit tout en utilisant les critères suivants :

- *Cohérence du guidage* : Même dans les modèles larges et complexes contenant des variantes interdépendantes, la manière de percevoir le processus de configuration doit garantir la cohérence du guidage. En fait, les décisions doivent être propagées et ne conduiront jamais à des recommandations qui violent les contraintes du domaine. Deux types de vérification de la cohérence peuvent être retenus à savoir : la vérification à *priori* (avant d'entamer la configuration) et la vérification à *posteriori* (une fois que la partie prenante finalise ses choix)
- *Flexibilité de la décision* : Tout au long du processus de configuration, la partie prenante se sent libre d'annuler ses choix et de modifier les décisions antérieures déjà prises. Dans le cas où la configuration est rigide et aucun retour sur les choix n'est autorisé, la décision est considérée *stricte*. Dans le cas contraire, la décision est considérée *flexible*.
- *Dimension du guidage* : Optimalement, le guidage du processus de configuration doit tenir compte de la recommandation d'un ordre du processus de configuration, d'une part, et de la recommandation des valeurs des variantes à chaque étape de la configuration, d'autre part. Dans le cas où le guidage est calculé à chaque étape de configuration, ce dernier est considéré comme *dynamique*. Cependant, dans le cas où le guidage est calculé une seule fois au début de la configuration, ce dernier est considéré comme *statique*.
- *Insensibilité du guidage* : Dans certains cas, la partie prenante est implicitement guidée par la hiérarchie conceptuelle du modèle de ligne de produits telle que la notation FODA [Kang et al., 1990]. En effet, les variantes positionnées au niveau supérieur du modèle (racines) sont configurées en premier. Ensuite, l'intervenant procède à la configuration des variantes positionnées au niveau inférieur (fils jusqu'aux feuilles). D'autres formes implicites de guidage peuvent conduire la partie prenante à configurer branche par branche, la racine et ses feuilles, puis passer à la racine suivante et ses feuilles. Le guidage produit doit ignorer la hiérarchie du modèle de ligne de produits et tenir compte de la variabilité orthogonale telle que la notation OVM [Pohl et al., 2005]. Dans le cas contraire, le guidage est considéré sensible à la *syntaxe du modèle*. Par ailleurs, le guidage produit peut être fortement influencé par la technique de recommandation derrière par exemple recommander toutes les variantes valides dans un programme de contraintes. Dans ce cas, le

guidage est considéré sensible au *raisonnement systématique*. Dans d'autres cas de figures, la recommandation peut être influencée par les traitements de fouille derrière particulièrement dans les recommandations basées sur des *analyses de fouille de processus et de fouille de données*. Le guidage devient standard et dépend uniquement des traces capitalisées sans aucune prise en compte de la différence des parties prenantes.

- *Intention de la partie prenante* : Face à une panoplie d'intentions des parties prenantes, différentes manières de guider l'acteur de la configuration s'avèrent importantes. La distinction des intentions des parties prenantes conduit à une meilleure personnalisation du guidage. Les travaux sélectionnés sont mitigés entre la *considération* de l'intention et son *ignorance*.
- *Portabilité du guidage* : D'autres logiciels configurables tels que les ERPs et les composantes COTS travaillent à améliorer leur processus de configuration. Les solutions de guidage *génériques* qui peuvent être transposées sont fortement appréciées à comparer avec les solutions de guidage avec une portabilité *limitée*.

Tableau (5) : Evaluation des solutions existantes pour le guidage de configuration des lignes de produits VERSUS la méthode envisagée

Méthodes connexes	Critères de comparaison					
	Cohérence	Flexibilité	Dimension	Insensibilité	Intention	Portabilité
<i>o</i> -SPLIT [Afzal et al., 2020]	Priori	Stricte	Statique	Syntaxe du modèle	Ignorée	Limitée
COLOSSI [Varela-Vaca et al., 2019]	Néant	Stricte	Dynamique	Analyses de fouille de processus	Ignorée	Générique
WeaFQAs [Horcas et al., 2017]	Priori	Stricte	Statique	Syntaxe du modèle	Ignorée	Générique
SATVaEA [Xiang et al., 2018]	Priori	Stricte	Statique	Raisonnement systématique	Ignorée	Limitée
QcCP [Noorian et al., 2017]	Priori	Flexible	Statique	Syntaxe du modèle	Considérée	Limitée
Agile Tames Product Line Variability [Hayashi et al., 2017]	Priori	Stricte	Dynamique	Néant	Ignorée	Générique
Extended FeatureIDE [Pereira et al., 2017]	Priori-Posteriori	Flexible	Dynamique	Raisonnement systématique	Considérée	Générique
ZEN-FIX [Lu et al., 2016]	Priori	Stricte	Statique	Raisonnement systématique	Ignorée	Limitée
Extended pure::variants [Sinnhofer et al., 2016]	Posteriori	Flexible	Dynamique	Néant	Considérée	Générique
Extended VariaMos [Mazo et al., 2014]	Priori	Flexible	Statique	Syntaxe du modèle	Ignorée	Limitée
Recodyn [Triki et al., 2014]	Posteriori	Stricte	Dynamique	Raisonnement	Ignorée	Limitée

				systematique		
VITAL [Zhang 2013]	Posteriori	Stricte	Statique	Analyses de corrélation	Ignorée	Limitée
IBEA Configurations [Sayyad et al., 2013]	Priori	Stricte	Statique	Syntaxe du modèle	Ignorée	Limitée
SPAF [Guana et al., 2013]	Posteriori	Stricte	Statique	Raisonnement systématique	Ignorée	Limitée
COVM [Dumitrescu et al., 2013a]	Posteriori	Flexible	Dynamique	Syntaxe du modèle	Ignorée	Limitée
FAMILIAR [Acher et al., 2013]	Posteriori	Stricte	Statique	Raisonnement systématique	Ignorée	Limitée
GEARS [Krueger et al., 2013]	Priori	Stricte	Statique	Raisonnement systématique	Ignorée	Limitée
SpineFM [Urli et al., 2012][Urli et al., 2014][Chastel et al., 2015]	Posteriori	Flexible	Statique	Analyses de fouille de données	Ignorée	Limitée
PLiC [Elsner et al., 2010]	Priori	Stricte	Statique	Syntaxe du modèle	Ignorée	Limitée
SPLOT [Mendonca et al., 2008]	Priori	Stricte	Statique	Syntaxe du modèle	Ignorée	Limitée
Genarch [Cirilo et al., 2007]	Posteriori	Stricte	Statique	Raisonnement systématique	Ignorée	Limitée
VISIT-FC [Botterweck et al., 2007]	Posteriori	Stricte	Statique	Syntaxe du modèle	Ignorée	Générique
ReMAP' [Schmid et al., 2006]	Posteriori	Stricte	Statique	Syntaxe du modèle	Ignorée	Générique
COVAMOF [Sinnema et al., 2004]	Priori	Stricte	Vues	Syntaxe du modèle	Ignorée	Générique
La méthode envisagée: CONFILOG	Temps réel	Flexible	Dynamique + valeurs de variantes	Indépendante	Dirigée par les buts	Tout logiciel configurable

III.6. Conclusion

Dans ce chapitre un état de l'art des problèmes de configuration des lignes de produits et des insuffisances des solutions existantes est dressé. Il est à noter que, dans cet état de l'art, les articles qui exposent les problèmes de configuration ne proposent pas forcément des solutions. Vice-versa, les articles qui proposent des solutions de configuration ne reviennent pas forcément sur les problèmes de configuration. En effet, les problèmes de configuration de lignes de produits identifiés sont catégorisés en 3 classes à savoir :

- La complexité de la décision,
- Le manque de performance et
- Le besoin de guidage.

Face à ces problèmes, plusieurs méthodes ont vu le jour en vue d'améliorer le processus de configuration à savoir :

- Les méthodes de configuration interactive,
- Les méthodes de configuration basées sur la recommandation,
- Les méthodes de configuration combinées,
- Les méthodes de configuration heuristiques et
- Les méthodes de configuration basées sur la fouille.

Les solutions existantes pallient les problèmes de configuration en une partie. Néanmoins, ces solutions s'avèrent caduques face à des parties prenantes de plus en plus exigeantes et différentes. En effet, plusieurs méthodes produisent du guidage rigide, standard et parfois dépendant au modèle de la ligne. De plus, la considération des buts et des différences de la partie prenante demeure omise. Ensuite, toutes les méthodes recensées recommandent des variantes ou des valeurs de variantes mais pas les deux avec un ordre de processus. Les problèmes identifiés et le besoin de recommander des valeurs et un ordre de processus en même temps correspondent parfaitement aux apports de la fouille de processus (cf. Chapitre II).

[Cette page est intentionnellement laissée en blanc]

PARTIE B : MÉTHODE

Chapitre IV : Aperçu de la méthode ConfiLog

Chapitre V : La méthode ConfiLog

Chapitre IV : Aperçu de la méthode ConfiLog

IV.1. Introduction	56
IV.2. Cadre méthodologique	56
IV.3. Vue globale de la méthode ConfiLog	58
IV.3.1. La vue fonctionnelle de la méthode ConfiLog	58
IV.2.2. La vue dynamique de la méthode ConfiLog	60
IV.3. Exemple d'application	62
IV.5. Caractéristiques de la méthode ConfiLog	64
IV.6. Conclusion	67

Chapitre IV

La méthode ConfiLog

IV.1. Introduction

Ce chapitre présente un aperçu de la méthode ConfiLog avant de passer à la détailler. Cette méthode est baptisée ConfiLog vu que la recommandation fournie lors de la CONFIGuration est basée principalement sur des LOGs. Le cœur de cette méthode réside en un algorithme de recommandation qui permet de configurer des lignes de produits dont la complexité est due à la large variabilité de leurs modèles. ConfiLog génère des recommandations en interagissant avec la partie prenante qui a la possibilité de choisir un but de guidage. Le processus de configuration est interactif et incrémental de façon à configurer les variantes de la ligne au fur et à mesure des étapes du processus.

IV.2. Cadre méthodologique

ConfiLog a été construite et évaluée par une méthode de recherche de type *Design Science Research* (DSR). Le (DSR) est connu comme étant un paradigme pour conduire et communiquer de la recherche appliquée comme la recherche en Systèmes d'Information ou encore en Génie Logiciel [Engström et al.,2020]. Par ailleurs, le but de (DSR) est de produire des connaissances pour les praticiens d'une discipline et de partager des idées empiriques tirées des investigations et des prescriptions effectuées [Aken, 2004]. Ce type de recherche a pour objectif d'exposer les problèmes, de suggérer des solutions et de motiver leur utilité.

Pour l'exécution d'une méthodologie (DSR) typique, cette recherche a été guidée par les étapes suivantes [Hevner et al., 2004]:

- Identifier les problèmes du processus de configuration (cf. Chapitre III)
- Proposer une solution pour les problèmes rencontrés (cf. Chapitre V)

- Concevoir une architecture qui supporte les solutions (cf. Chapitre VI)
- Evaluer la solution (cf. Chapitre VII) et
- Souligner les valeurs ajoutées et les limites (cf. Chapitre VIII).

Ainsi, cette recherche vise à proposer une méthode qualitative à caractère exploratoire qui souligne le rôle de la fouille des processus pour remédier aux problèmes du processus de configuration des lignes de produits. Cependant, les processus de formalisation et de vérification d'ingénierie de domaine des lignes de produits [Pohl et al., 2005] ne sont pas pris en compte dans ce manuscrit.

Ce projet de recherche a été fondé en se basant sur une maquette visuelle pour la conduire d'un (DSR) [Storey et al., 2007] instanciée dans la figure (8). La maquette résume les fondamentaux de la méthodologie et justifie le choix de (DSR), par ailleurs, elle met en évidence les aspects traités, comment ils sont validés, comment la compréhension du problème est réalisée et quelles solutions sont envisagées. De surcroît, l'instanciation de la maquette incite à réfléchir, non seulement, à la manière dont la méthode proposée ajoute de nouvelles connaissances, mais aussi, à la relation entre la règle générale et son instanciation.

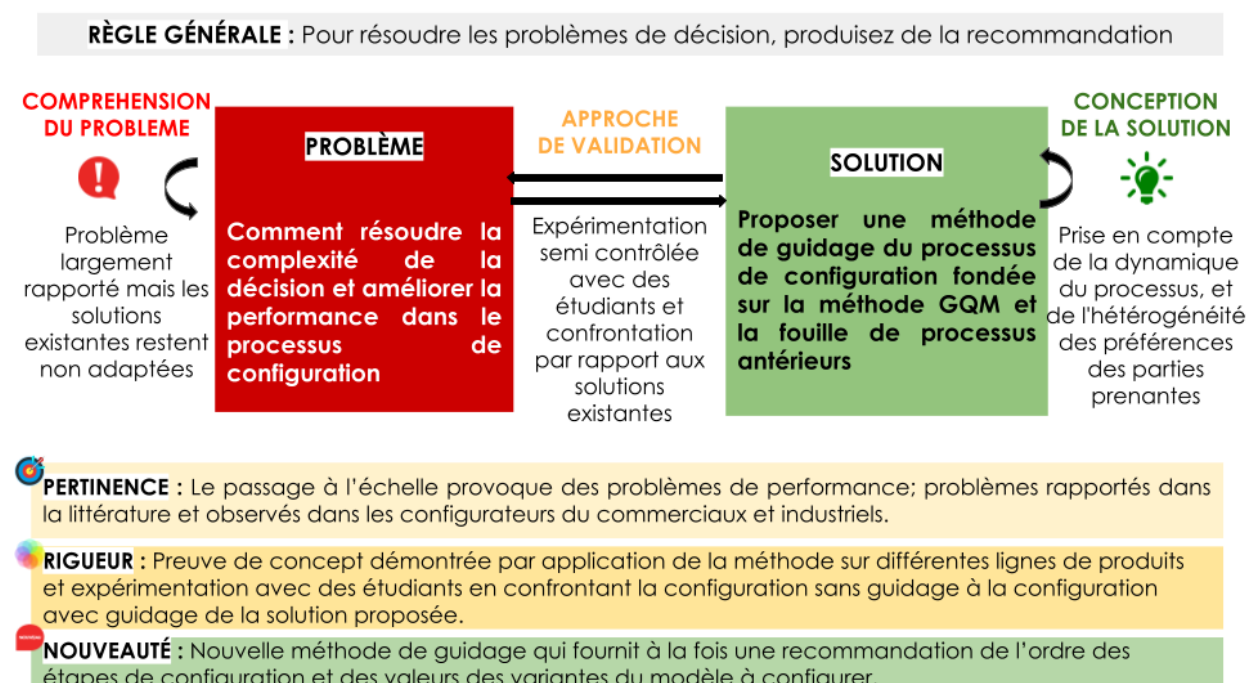


Figure (8) : Instanciation de la maquette de (DSR) [Storey et al., 2007]

La méthode ConfiLog doit répondre aux problèmes identifiés dans la littérature à savoir (i) la complexité de la décision, (ii) le manque de performance et (iii) le besoin de guidage. Ces problèmes de configuration sont rapportés à la fois par les chercheurs et par les praticiens. En effet, une solution de guidage du processus de configuration est envisagée; cette dernière est (i) fondée sur la fouille de processus, (ii) orchestrée par des buts, des questions et des métriques d'évaluation définis par la méthode (GQM) et (iii) validée par des expérimentations semi contrôlées avec les étudiants.

La concentration bi dimensionnelle sur la dynamique du processus et les instances de décisions à la fois est la ligne directrice de la solution envisagée. La pertinence, la rigueur et la nouveauté constituent des considérations de premier ordre. La section suivante détaille comment la méthode ConfiLog accomplit ces considérations.

IV.3. Vue globale de la méthode ConfiLog

La méthode ConfiLog prend comme entrées (i) un modèle de ligne de produits et (ii) un but de guidage exprimé par la partie prenante. En contrepartie, elle produit du guidage sur deux niveaux à savoir (i) des valeurs de variantes de la ligne et (ii) un ensemble d'étapes de processus de configuration à suivre. Cette section se focalise sur la vue fonctionnelle et la vue dynamique de la méthode ConfiLog.

IV.3.1. La vue fonctionnelle de la méthode ConfiLog

La Figure (9) donne une vue fonctionnelle de la méthode ConfiLog considérant deux étapes à savoir :

- La conduite du processus de configuration et
- La production du guidage.

L'étape de la conduite du processus de configuration se concentre sur l'exécution de la configuration même en fonction des actions des parties prenantes. Dès qu'une partie prenante est confrontée à une configuration d'une ligne de produits, elle peut sélectionner un but de guidage souhaité, par exemple un processus de configuration rapide. Les buts de guidage sélectionnés par les parties prenantes ne peuvent pas être autres que les buts répertoriés dans une base de la méthode (GQM). Étant basée sur des actions de la partie prenante, la configuration est systématiquement propagée et sa cohérence conceptuelle est vérifiée afin de s'assurer que les prochaines étapes du processus sont basées uniquement sur une configuration partielle correcte. En boucle

pour chaque étape de configuration, la cohérence des décisions est vérifiée jusqu'à atteindre une configuration jugée complète par la partie prenante. A chaque étape de configuration, la partie prenante peut annuler des choix déjà configurés en faisant un retour en arrière. ConfiLog réagit dynamiquement en mettant à jour une requête de recommandation indiquant (i) la configuration partielle, (ii) les variantes qui restent compatibles et les variantes qui le sont plus et (iii) le but de guidage exprimé par la partie prenante.

L'étape de la production du guidage comprend (i) une orchestration des buts, des questions et des métriques, (ii) une fouille de processus de configuration et (iii) un calcul de la recommandation.

Les buts, les questions et les métriques sont structurés dans une base appelée base GQM. Chaque but de guidage répertorié dans la base GQM et sélectionné plus tard par une partie prenante lors de la configuration, est atteint si la méthode réussit à répondre à ses questions de guidage. La réponse aux questions de guidage est évaluée en fonction des métriques.

Pour répondre à ladite requête, la mécanique de la méthode ConfiLog se base sur deux fonctions majeures, notamment, la fouille de processus de configuration et le moteur de recommandation. La fouille de processus de configuration est opérée sur la base de traces de processus enrichie après chaque fin de processus de configuration. La méthode capture tout au long de la configuration plusieurs attributs pour tracer le processus à savoir l'horodatage des décisions et leur nature (choix manuels ou optionnels). La fouille de processus de configuration vise ainsi à reconstituer des modèles de processus bruts et les filtrer/interpréter suivant la méthode (GQM). Les modèles de processus filtrés correspondent à des réponses de guidage qui sont interrogées, tout au long du processus de recommandation, par un moteur de recommandation.

En fonction de la configuration en question, des prédicats de support sont paramétrés afin de garder uniquement des instances de processus pertinentes. Les instances de processus jugées pertinentes sont pondérées. Un calcul de l'impact de la sélection ou le refus de chaque décision est calculé ainsi. Calculer l'impact des décisions à venir revient à calculer à quel point une potentielle suggestion est favorable de répondre à un but de configuration donné dans un état de configuration donnée. En fonction des calculs d'impact, des recommandations en rapport avec l'ordre des décisions de configuration et

les instances de ces décisions sont fournies à la partie prenante. Seules les variantes en instance de configuration sont recommandées. Cependant, les variantes incluses dans la configuration partielle sont exclues. En plus de guider la partie prenante pour affecter des valeurs aux variantes, l'algorithme se charge de définir un ordre de processus à suivre. Le système met en premier plan la valeur recommandée de la variante sans l'imposer à la partie prenante, assurément. Pareillement, l'adoption de la recommandation de l'ordre des étapes de configuration reste facultative. La partie prenante est toujours libre d'adapter une recommandation ou de changer de but de guidage à tout moment de la configuration s'il le souhaite.

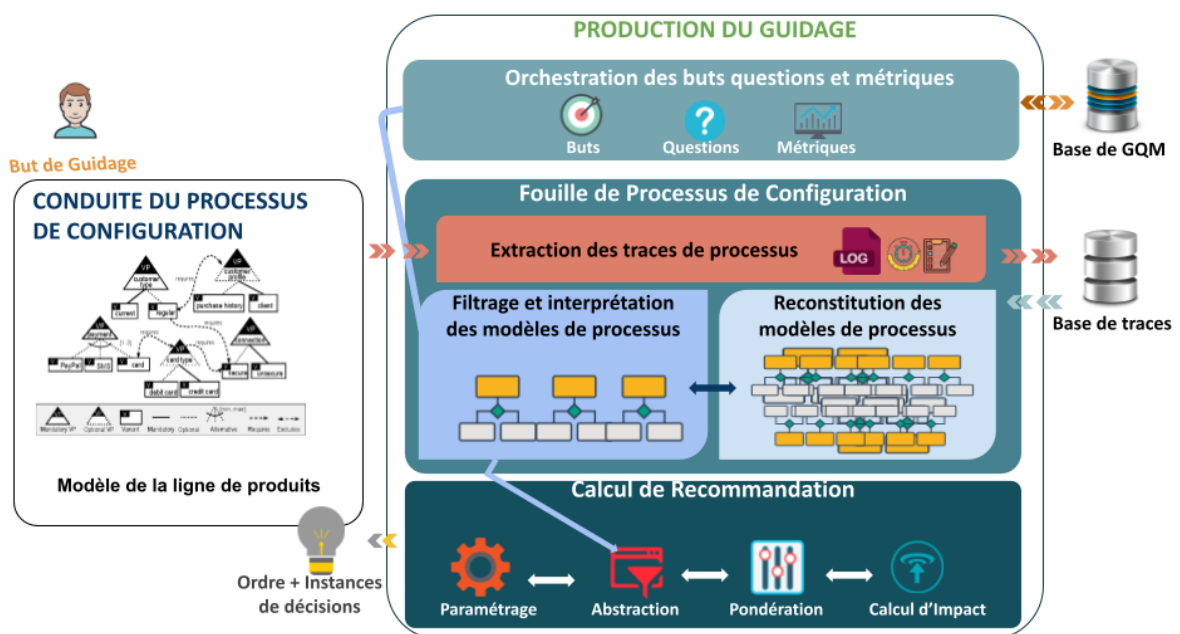


Figure (9) : Vue fonctionnelle de la méthode ConfiLog

IV.2.2. La vue dynamique de la méthode ConfiLog

La Figure (10) présente un récapitulatif des interactions effectuées entre une partie prenante qui s'appelle Alice et le système ConfiLog sous forme de diagramme de séquence. La séquence montre un but de guidage exprimé par la partie prenante avant toute manœuvre dans le système. Ensuite, Alice choisit une variante à configurer. Les choix effectués sur cette variante peuvent provoquer une propagation des contraintes du domaine. Les variantes du modèle configurées manuellement et par propagation de contraintes sont enregistrés dans un paquet appelé configuration partielle. En fonction du paquet de la configuration partielle, une fouille de processus est conduite. La fouille

de processus permet la reconstitution des modèles de processus qui représentent l'essence de la recommandation. Ainsi, le configurateur retourne une recommandation à Alice. Ces interactions se font en boucle tout au long du processus de configuration. Une fois Alice valide la configuration, les choix et les étapes sont stockés dans la base de journaux d'événements.

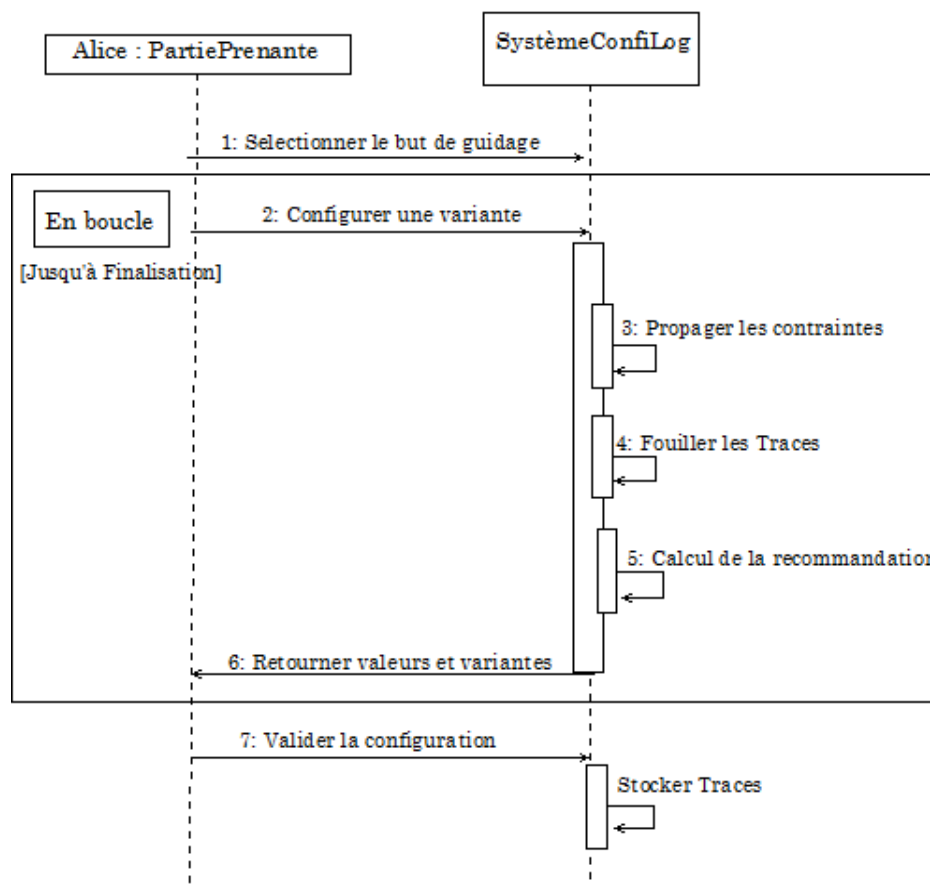


Figure (10) : Récapitulatif des interactions effectuées entre la partie prenante et ConfiLog

Relativement au diagramme de séquence de la figure (10) et de point de vue instance, ConfiLog distingue trois couches (cf. Chapitre V), notamment : la configuration, la fouille de processus de configuration et le calcul de la recommandation. Ces couches collaborent via une mécanique de système de manière à assurer la bonne assistance de la partie prenante tout au long du processus de configuration.

- La configuration : Cette couche est responsable de la réception des actions des parties prenantes, de calcul de cohérence de la configuration et de la propagation des contraintes de domaine à chaque itération de la boucle (cf. Figure 10).
- La fouille des processus de configuration : Chaque processus de configuration complet ou incomplet donne une trace de configuration. Les traces de configuration

sont stockées dans une base contenant des journaux d'évènements. La base de journaux d'évènements sert ultérieurement à des fins de fouille de processus et particulièrement la reconstitution des modèles de processus reflétant différents contextes.

- Le calcul de recommandation : Une fois, des modèles de processus sont reconstitués, des recommandations sont calculées suivant une requête spécifique contenant un but de guidage et une configuration partielle.

IV.3. Exemple d'application

L'exemple de la ligne de produits *Site Web* est un exemple basique qui a été largement utilisé en ingénierie de domaine et ingénierie d'application des lignes de produits [Lackner et al., 2014][Rincon et al., 2015][Farahani et al., 2019]. Cet exemple est repris pour illustrer l'application de la méthode ConfiLog et ses grandes fonctions. Le modèle de ligne de produits présenté dans la Figure (11) est spécifié au moyen de la notation FODA¹ [Kang et al., 1990]. De plus, il a été choisi pour sa simplicité.

Chaque *site web* à configurer à partir de la ligne de produits contient obligatoirement des mesures de Performance et un *Serveur web*. D'autres *Services Additionnels* sont optionnels. La Performance est mesurée en Minutes, Secondes ou Millisecondes. Le *Serveur web* a obligatoirement un contenu et optionnellement un Protocole et un service de Traçage. Le Contenu est Statique mais il peut être Dynamique en même temps. Les contenus dynamiques sont implémentés en ASP, PHP et/ou Thymeleaf . Il y a deux types de protocoles, notamment : le HTTPS et/ou le FTP. Le service de Traçage capitalise les données dans des Bases de données ou des Fichiers. Les Services additionnels contiennent obligatoirement des Rapports et des Bannières publicitaires en Images. Des bannières en HTML5 sont optionnelles. Les statistiques et les menus de Recherche sont des Services Additionnels optionnels.

La ligne de sites web présente un ensemble de contraintes de domaine de type *exige* et *exclut* qui sont listées ci-dessous :

Contraintes Exige: [\neg Recherche \vee Secondes, \neg Statistiques \vee Secondes, \neg Fichier \vee FTP, \neg Services additionnels \vee Performance, \neg Rapports \vee Fichier, \neg Contenu \vee Protocoles, \neg Dynamique \vee HTTPS] Contraintes Exclut: [\neg HTTPS \wedge \neg Millisecondes]

¹ FODA (Feature Oriented Domain Analysis) est une notation utilisée pour l'analyse de domaine et axée sur les caractéristiques des systèmes. L'objectif de FODA est d'identifier les caractéristiques importantes décrivant les aspects communs et variables d'un système à l'autre dans une famille de produits.

En guise d'exemple, les sites web avec services de Recherche et de Statistiques doivent avoir obligatoirement une Performance mesurée en Secondes. Les Protocoles de type HTTPS excluent que la mesure de Performance soit mesurée en Millisecondes.

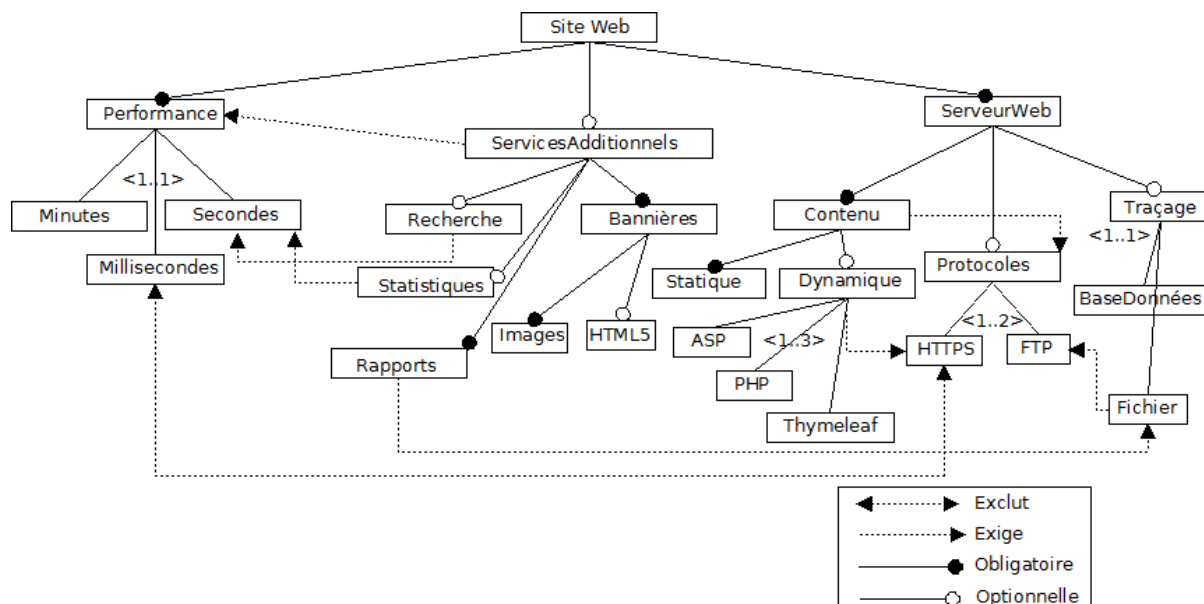


Figure (11) : Modèle adapté d'une ligne de produits de sites Web [Rincon et al., 2015]

Pour appliquer la méthode ConfiLog à la ligne de sites web, il faut que la partie prenante sélectionne un but de guidage parmi ceux proposés. La base de la méthode (GQM) englobe trois buts de guidage majeurs, notamment :

- Un processus de configuration rapide,
- Un processus de configuration flexible ou
- Un processus de configuration personnalisé.

Ensuite, la partie prenante introduit ses exigences qui sont capturées dans un paquet appelé configuration partielle. Relativement à l'exemple (cf. figure 11), la partie prenante sélectionne la variante Services Additionnels. En contrepartie, le système va propager automatiquement les contraintes du domaine et mettre à jour le paquet de configuration partielle en incluant la variante fraîchement configurée et les contraintes qu'elle implique.

Paquet de Configuration partielle : [Performance: Selected, Services Additionnels: Selected, Rapports: Selected, Bannières: Selected, Images: Selected, Contenu: Selected, et Statique: Selected].

En fonction de la configuration partielle et du but de guidage exprimé, l'algorithme va calculer la recommandation à partir des traces rapides qui ont déjà eu lieu avant. Le système va fouiller les modèles de processus reconstitués et les filtrer de façon à garder uniquement les instances de processus répondant aux deux critères suivants :

- Les instances de processus jugées rapides après traitement de fouille de processus.
- Les instances de processus qui garantissent la présence des variantes capturées dans le paquet de configuration partielle.

A partir du modèle de processus filtré, l'algorithme calcule l'impact de temps sur le reste du processus, si une variante à recommander est sélectionnée ou rejetée. En fonction de ces calculs, des recommandations sont fournies aux parties prenantes sous forme des valeurs de variantes (sélectionner ou refuser) et/ou un ordre d'étapes de configuration respectant les contraintes. A chaque étape de configuration, les valeurs et l'ordre sont recalculés de façon cohérente permettant de combiner les valeurs de la configuration partielle avec celles de la recommandation tout en préservant des temps de réponse minimales.

IV.5. Caractéristiques de la méthode ConfiLog

La méthode ConfiLog rassemble certaines caractéristiques spécifiques, notamment :

- **Une cohérence des décisions** : Une particularité de la méthode ConfiLog réside dans la manière de percevoir le processus de configuration. Même dans les modèles larges et complexes avec des variantes interdépendantes, la garantie de conformité et de cohérence de la configuration est assurée. Au fur et à mesure du processus de la configuration, une propagation des choix de façon à ce que la partie prenante ne soit jamais confrontée à des configurations qui violent les contraintes du domaine de la ligne. La configuration est calculée par le biais des équations mathématiques [Salinesi et al., 2011] qui projettent les choix de la partie prenante sur le modèle de la ligne en garantissant une marge d'erreur nulle. De même, étant donné que les traces sur lesquelles est basée la fouille sont cohérentes, le service de recommandation ConfiLog ne fournira jamais des recommandations qui ne respectent pas les contraintes du domaine.
- **Une meilleure liberté de la partie prenante** : Tout au long du processus de configuration, la prise de décision est assouplie du fait que la modification des choix déjà effectués est possible. A chaque annulation de choix, la méthode

ConfiLog recalcule la cohérence de la configuration et la pertinence de la recommandation. La partie prenante est notifiée si son annulation de choix aurait un impact sur d'autres variantes du modèle de la ligne.

- **Un guidage en deux dimensions** : Outre que son calcul dynamique, à chaque étape de configuration, le guidage de la méthode ConfiLog s'attèle à la recommandation d'un ordre de processus de configuration d'un côté et la recommandation des valeurs de variantes d'un autre côté. A chaque étape de configuration, ConfiLog calcule un nouvel ordre du reste des variantes à configurer. Cette dynamique d'ordonnement des variantes prend en compte les variantes déjà configurées dans le processus en cours et le but de guidage choisi par la partie prenante. En sus de l'ordre de guidage, la méthode ConfiLog renforce le guidage en fournissant des recommandations des valeurs de variantes. L'algorithme récupère, en outre, des informations permettant de recommander la sélection, le rejet ou l'ignorance d'une variante donnée.
- **Un guidage indépendant de l'arborescence du modèle de la ligne** : Dans certaines solutions évoquées dans l'état de l'art (cf. Chapitre III, section 4), la partie prenante est guidée implicitement par la hiérarchie conceptuelle du modèle de la ligne. En effet, les variantes se positionnant dans la partie supérieure du modèle, appelées nœuds, sont configurées en premier. Ensuite, la partie prenante passe à configurer les variantes, se positionnant dans la partie inférieure du modèle, appelées feuilles. D'autres formes implicites de guidage peuvent conduire la partie prenante à configurer le nœud et ses feuilles et passer, par la suite, au nœud suivant ainsi que ses feuilles. Ces formes d'assistance de configuration, parfois inconscientes, sont prises en compte par la méthode ConfiLog de façon à ce que la recommandation de l'ordre des variantes soit orthogonal et insensible à l'arborescence du modèle.
- **Une prise en compte de l'intention de la partie prenante** : Face à une panoplie d'intentions des parties prenantes, différentes manières de guider le processus de configuration s'avèrent importantes. ConfiLog distingue trois buts majeurs de guidage, notamment : la performance, la flexibilité et la personnalisation du processus de configuration.

La performance est un facteur très important auquel doivent répondre les configureurs compétitifs. Pour optimiser la performance, ConfiLog veille à recommander des valeurs et un ordre de variantes qui accélère le processus de

configuration. Par exemple, les traces durant lesquelles il n'y a pas eu d'allers/retours, causant des backtracks au configurateur, peuvent être exploitées dans ce but de guidage.

La flexibilité doit être assurée dans un contexte de configuration. En effet, face à un large éventail de choix, il est tout à fait normal que la partie prenante soit confrontée à des processus de longue haleine. Pour y remédier, les recommandations évitent d'exploiter des traces là où il y avait des problèmes. En outre, le processus est animé par plusieurs pistes de configuration, par exemple : recommander des produits qui sont à la mode ou qu'on peut atteindre leurs constructions avec un nombre restreint d'étapes.

La personnalisation constitue un objet majeur de l'adoption de l'ingénierie des lignes de produits. Pour renforcer cet aspect, la méthode ConfiLog donne le moyen à ses parties prenantes de pouvoir personnaliser la configuration afin que celle-ci devienne la plus unique possible. Pour ce faire, ConfiLog implique davantage les choix manuels de la partie prenante plutôt que proposer des décisions prédéfinies venant des configurations similaires.

Pour chaque but de guidage, des questions de guidage sont posées. La réponse à ses questions est l'objet de l'adoption d'une stratégie de fouille de processus.

- **Une Incrémentalité et une complétude de la configuration** : La méthode ConfiLog permet de générer des configurations complètes aussi bien que des configurations partielles. Cette dernière option analyse les solutions alternatives les unes après les autres, et non pas d'un seul coup. Ou encore, étant donnée, une configuration partielle, il devient possible d'induire les prochains choix à faire et les variantes nécessitant encore un arbitrage. Par un processus itératif, les nouveaux choix sont pris en compte de manière interactive et de nouveaux résultats sont fournis au fur et à mesure des choix effectués.
- **Portabilité du guidage** : L'extraction des traces de processus de configuration et leur fouille ouvre des champs entiers d'application. Cette applicabilité est analogue à l'applicabilité ubiquitaire de l'ingénierie des lignes de produits dans le milieu industriel et commercial. La méthode ConfiLog est applicable sur tout modèle de variabilité, modèles avec variabilité arborescente ou orthogonale inclus.

IV.6. Conclusion

Ce chapitre a donné une vue globale de la méthode ConfiLog construite suivant les recommandations d'une méthode (DSR) et évaluée en fonction d'une méthode (GQM). En effet, pour la production du guidage, la méthode ConfiLog utilise les éléments suivants :

- Un modèle de ligne de produits,
- Des journaux d'évènements contenant les traces de processus de configuration antérieurs,
- Un but exprimé par la partie prenante et
- Une base de la méthode (GQM).

En contrepartie, ConfiLog produit du guidage qui considère :

- L'ordre des décisions à prendre dans le processus de configuration et
- Des instances de décisions à prendre dans le processus de configuration.

Pour parvenir à de tels résultats, ConfiLog reconstitue des modèles de processus à partir des journaux d'évènements. Les modèles de processus sont filtrés et interprétés suivant une base de la méthode (GQM) . À partir des connaissances capitalisées des modèles de processus reconstitués, ConfiLog calcule des recommandations dynamiques, en d'autres termes, les recommandations recalculées à chaque étape de configuration. Un exemple d'application d'une ligne de sites web a été illustré pour expliquer les manipulations et les scénarios fondamentaux de la méthode ConfiLog.

L'intérêt de la méthode ConfiLog, à comparer avec les méthodes existantes, réside dans la complémentarité de ses caractéristiques principalement la cohérence, la souplesse et la personnalisation du guidage. Un exemple d'application d'une ligne de sites web a été illustré pour expliquer les manipulations et les scénarios fondamentaux de la méthode ConfiLog. Par ailleurs, l'algorithme de recommandation de ConfiLog et les traitements de fouille de processus sont détaillés dans le chapitre V.

[Cette page est intentionnellement laissée en blanc]

Chapitre V : La méthode ConfiLog

V.1. Introduction	70
V.1. Orchestration des buts des questions et des métriques	70
V.1.1. But 1 : Réduire le temps requis pour atteindre une configuration valide du point de vue du processus	71
V.1.2. But 2 : Maximiser la flexibilité et l'efficacité de la configuration du point de vue du processus	72
V.1.3. But 3 : Maximiser la personnalisation de la configuration du point de vue du processus	73
V.1.4. But 4 : Promouvoir la recommandation de la configuration du point de vue des intérêts de l'entreprise	75
V.2. La fouille de processus de configuration	76
V.2.1. L'extraction des traces de processus de configuration	76
V.2.2. La reconstitution des modèles de processus de configuration	77
V.2.3. Le filtrage et l'interprétation des modèles de processus de configuration	79
V.3. Calcul de la recommandation	82
V.3.1. Le paramétrage du processus de configuration	83
V.3.2. L'abstraction et le filtrage des réponses de fouille	83
V.3.3. La pondération des réponses de fouille	84
V.3.4. Le calcul de l'impact de décision	85
V.3.5. Démonstration du moteur de recommandation ConfiLog avec la ligne de sites web	86
V.3.5.1. Etape <i>n</i> du processus de configuration de la ligne de sites web	87
V.3.5.2. Etape <i>n+1</i> du processus de configuration de la ligne de sites web	90
V.4. Conclusion	91

Chapitre V

La méthode ConfiLog

V.1. Introduction

Ce chapitre détaille les fondements de la méthode ConfiLog à savoir (i) l'orchestration des buts, des questions et des métriques, (ii) la fouille de processus permettant la reconstitution des modèles de processus de configuration à partir de la base des journaux d'événements et leurs interprétations finalement (iii) calcul de recommandation permettant de guider les parties prenantes et conduire les futurs processus de configuration. La preuve de concept [Morgan, 2013] de cette méthode est ensuite menée sur la ligne de produits de sites web.

V.1. Orchestration des buts des questions et des métriques

Face aux différentes préoccupations des parties prenantes impliquées dans la configuration, la méthode ConfiLog identifie pour chaque processus de configuration les traitements de fouille associés pour produire du guidage. Dans ce sens, la méthode (GQM) proposée par Basili et al. (1994) comme étant un cadre de définition et d'interprétation des mesures logicielles a été adoptée. En effet, la méthode (GQM) est le fruit de nombreuses années d'expérience pratique et de recherche universitaire. La Figure (12) montre les niveaux de hiérarchie du modèle de la méthode (GQM) : buts, questions et métriques.

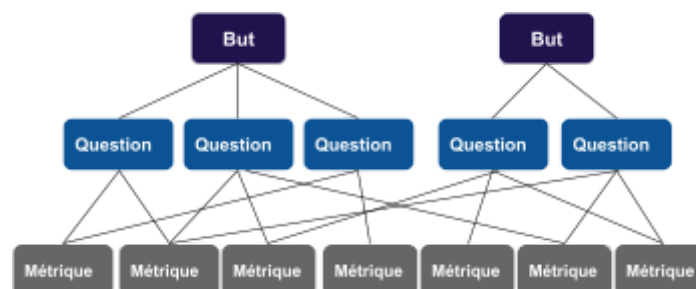


Figure (12) : Structure du Modèle de la méthode (GQM) [Basili et al., 1994]

Le niveau supérieur présente les buts à définir. Chaque but doit être opérationnalisé en générant des questions qui aident à comprendre comment l'atteindre. Les questions tentent d'amener des réponses satisfaisant le but en question. Le niveau inférieur présente les métriques qui évaluent les réponses aux questions.

Pour procéder à la (GQM), quatre buts de guidage ont été identifiés dans la première étape de la (GQM) appliquée à la méthode ConfiLog. Pour une cohérence des buts à répertorier dans la base de la méthode (GQM), la syntaxe suivante a été adoptée : *[But, Problème, Objet et Point de vue]*.

- *But 1*: Réduire le temps requis pour atteindre une configuration valide du point de vue du processus.
- *But 2*: Maximiser la flexibilité et l'efficacité de la configuration du point de vue du processus.
- *But 3*: Maximiser la personnalisation de la configuration du point de vue du processus.
- *But 4*: Promouvoir la recommandation de la configuration du point de vue des intérêts de l'entreprise¹.

Dans la deuxième étape, des questions permettant d'atteindre les buts de guidage ont été identifiées. La réponse à ces questions se fait à travers une interprétation des modèles de processus reconstitués. Dans une troisième étape, les métriques qualitatives et/ou quantitatives permettant d'évaluer les réponses à ces questions ont été identifiées. Pour chaque but, au moins une question de guidage est posée. De plus, pour chaque question, la réponse est évaluée avec au moins une métrique. Pour ce faire, atteindre un but i revient à interroger les modèles de processus reconstitués à partir du journal d'événements et apporter des réponses aux $q_{i.n}$ questions. Les réponses sont évaluées, ultérieurement, en utilisant les $M_{i.n.m}$ métriques.

V.1.1. But 1 : Réduire le temps requis pour atteindre une configuration valide du point de vue du processus

Le Tableau (6) décrit le modèle de la méthode (GQM) pour le but suivant "Réduire le temps requis pour atteindre une configuration valide du point de vue du processus". Le temps requis est considéré comme étant une priorité dans un processus de configuration. Afin de réduire le temps requis d'une configuration donnée, ConfiLog apporte des

¹ Le but numéro 4 nécessite des informations contextuelles en rapport avec l'activité de l'entreprise

réponses, à travers les modèles de processus de configuration, en rapport avec les variantes qui doivent être configurées en premier (Q1.1), le temps moyen requis pour finaliser la configuration (Q1.2), les décisions qui obliquent le processus de configuration en engendrant des retours sur les choix (Q1.3) et les processus présentant des configurations rapides ou lentes (Q1.4).

Tableau (6) : (GQM) pour le but “Réduire le temps requis pour atteindre une configuration valide du point de vue du processus”

But 1	Objectif	Réduire
	Problème	Le temps requis pour atteindre
	Objet	Une configuration valide
	Point de Vue	Du point de vue du processus
Questions / Metriques		<p>Q1.1. Quelles sont les variantes qui doivent être configurées en premier ? <i>M1.1.1. Nombre de processus qui commencent avec cette variante</i> <i>M1.1.2. Nombre de variantes les plus sélectionnées</i></p> <p>Q1.2. Quel est le temps moyen requis pour finaliser le processus ? <i>M1.2.1. Nombre d'interactions avec la partie prenante</i> <i>M1.2.2. Temps requis pour finaliser n étapes</i></p> <p>Q1.3. Quels sont les processus qui n'ont pas impliqué des retours sur les choix ? <i>M1.3.1. Nombre des variantes impliquées dans le processus de configuration</i> <i>M1.3.2. Nombre des interactions de la partie prenante avec chaque variante</i></p> <p>Q1.4. Quels sont les processus qui ont duré au-dessus / au-dessous du temps moyen ? <i>M1.4.1. Temps requis pour finaliser n étapes</i> <i>M1.4.2. Tolérance de temps au-dessus/ au-dessous du temps moyen</i></p>

V.1.2. But 2 : Maximiser la flexibilité et l'efficacité de la configuration du point de vue du processus

Le Tableau (7) décrit le modèle de la méthode (GQM) pour le but suivant : “Maximiser la flexibilité et l'efficacité des alternatives de configuration du point de vue du processus”. La flexibilité du processus de configuration fournit un confort pour la partie prenante qui lui permettra de finaliser sa configuration. Pour amener des éléments de réponses en rapport avec la flexibilité, ConfiLog identifie quelles sont les configurations représentant des parties prenantes qui ont eu du mal à atteindre un produit final (Q2.1)

et quelles variantes ont causé ces difficultés (Q2.2). En outre, elle cherche à identifier les processus exécutés avec le plus / le moins des interactions avec les parties prenantes (Q2.3). Par ailleurs, des chemins de configuration sûrs / douteux sont identifiés (Q2.4).

Tableau (7) : (GQM) pour le but “Maximiser la flexibilité et l'efficacité de la configuration du point de vue du processus”

But 2	Objectif	Maximiser
	Problème	La flexibilité et l'efficacité
	Objet	De la configuration
	Point de vue	Du point de vue processus
Questions / Metriques		<p>Q2.1. Est ce que la partie prenante s'est perdue pendant le processus ? <i>M2.1.1. Temps requis pour configurer n étapes</i> <i>M2.1.2. Nombre de retours sur les choix</i></p> <p>Q2.2. Quelles sont les variantes qui causent les backtracks ? <i>M2.2.1. Aboutissement des retours sur les choix. Enum About {réussie, échouée}</i> <i>M2.2.2. Affectation de la valeur d'une variante. Enum Affect {Manuelle, automatique}</i> <i>M2.2.3. Nombre des variantes impliquées dans un backtrack</i></p> <p>Q2.3. Quels sont les processus qui ont été exécutés avec un minimum/maximum de choix manuels / propagés? <i>M2.3.1. Type de variante. Enum Type {Optionnelle, Obligatoire, Regroupée}</i> <i>M2.3.2. Nombre des allers/retours sur les choix</i></p> <p>Q2.4. Quels sont les chemins les plus sûrs/douteux dans un processus ? <i>M2.4.1. Fréquence de passage d'une variante à une autre</i> <i>M2.4.2. Nombre des allers/retours sur les choix</i></p>

V.1.3. But 3 : Maximiser la personnalisation de la configuration du point de vue du processus

Le Tableau (8) décrit le modèle de la méthode (GQM) pour le but suivant “Maximiser la personnalisation de la configuration du point de vue du processus”. La personnalisation du processus de configuration constitue un objectif majeur d'une stratégie de ligne de produits. Sauf dans les larges modèles de lignes de produits, il s'avère difficile de personnaliser le guidage. Pour exécuter cela, ConfiLog identifie les processus là où la configuration était minutieuse (Q3.1), l'historique des choix d'une partie prenante donnée (Q3.2), les tendances et les intentions similaires (Q3.3), les

“best/worst sellers” des différentes parties prenantes (Q3.4) et différentes périodes (Q3.5) et finalement les configuration ayant sélectionné et/ou rejeté le maximum/minimum de choix.

Tableau (8) : (GQM) pour le but “Maximiser la personnalisation de la configuration du point de vue du processus”

But 3	Objectif	Maximiser
	Problème	La personnalisation
	Objet	De la configuration
	Point de vue	Du point de vue processus
Questions / Métriques	<p>Q3.1. Quels sont les processus qui ont été exécutés minutieusement ? <i>M3.1.1. Temps moyen requis pour finaliser un processus</i> <i>M3.1.2. Temps requis pour le passage d’une variante à une autre</i> <i>M3.1.3. Nombre des variantes sélectionnés et/ou refusés</i> <i>M3.1.4. Type de variante. Enum Type {Optionnelle, Obligatoire, Regroupée}</i></p> <p>Q3.2. Quelle est la trace d’une partie prenante donnée ? <i>M3.2.1. Décisions sur les variantes de la partie prenante. Enum Dec {accepter, refuser}</i> <i>M3.2.2. Temps requis pour finaliser la configuration</i></p> <p>Q3.3. Quelles sont les parties prenantes qui ont des intentions similaires <i>M3.3.1. Décisions sur les variantes de la partie prenante. Enum Dec {accepter, refuser}</i> <i>M3.3.2. Décisions sur les variantes des autres parties prenantes. Enum DecAutres {accepter, refuser}</i> <i>M3.3.3. Type de variante. Enum Type {Optionnelle, Obligatoire, Regroupée}</i> <i>M3.3.4. Temps requis pour finaliser la configuration</i></p> <p>Q3.4. Quels sont les processus qui incluent des best-sellers / worst-sellers pour chaque catégorie des parties prenantes (Sexe, Âge, Pays, etc..) ou pour une période donnée (été, vacances, etc..) ? <i>M3.4.1. Décisions sur les variantes des autres parties prenantes. Enum DecAutres {accepter, refuser}</i> <i>M3.4.3. Métriques contextuelles (Période, événements, localisation, etc...)</i></p> <p>Q3.5. Quels sont les processus qui incluent les configurations à la mode ? <i>M3.5.1. Décisions sur les variantes des autres parties prenantes. Enum DecAutres {accepter, refuser}</i> <i>M3.5.2. Métrique contextuelle (Intervalle de temps de mode)</i></p> <p>Q3.6. Quels sont les processus qui incluent les configurations avec le maximum/minimum des variantes sélectionnées ? <i>M1.6.1. Type de variante. Enum Type {Optionnelle, Obligatoire, Regroupée}</i> <i>M1.6.2. Nombre de variantes de la ligne</i></p>	

* : Question qui demande des métriques contextuelles pour que sa réponse soit évaluée

V.1.4. But 4 : Promouvoir la recommandation de la configuration du point de vue des intérêts de l'entreprise

Le Tableau (9) décrit le modèle de la méthode (GQM) pour le but suivant “Promouvoir la recommandation de la configuration du point de vue des intérêts de l'entreprise”. Dans certaines situations, l'entreprise cherche à favoriser la vente des produits qui ne lui coûtent pas cher (Q4.1, Q4.3) ou liquider des produits parce qu'elle en a assez en stock par exemple (Q4.2). Néanmoins, les réponses à ces questions nécessitent des données autre que la méthode ConfiLog trace dans un journal d'évènements.

Tableau (9) : (GQM) pour le but “Promouvoir la recommandation de la configuration du point de vue des intérêts de l'entreprise”

But 4	Objectif	Promouvoir
	Problème	La recommandation
	Objet	De la configuration
	Point de vue	Du point de vue des intérêts de l'entreprise
Questions / Métriques	<p>Q4.1 * Quels sont les processus qui amènent à réduire le coût de la production ? <i>M4.1.1. Temps requis pour finaliser n étapes</i> <i>M4.1.3. Métrique contextuelle (Coût de chaque variante)</i></p> <p>Q4.2. * Quels sont les processus qui amènent à liquider le stock ? <i>M4.2.1. Temps requis pour finaliser n étape</i> <i>M4.2.3. Métrique contextuelle (Stock de chaque variante)</i></p> <p>Q4.3. * Quels sont les processus qui amènent à minimiser la charge des ressources humaines ? <i>M4.3.1. Temps requis pour finaliser n étape</i> <i>M4.3.3. Métrique contextuelle (Ressources humaines demandées pour la production de chaque variante)</i></p>	

* : Question qui demande des métriques contextuelles pour que sa réponse soit évaluée

V.2. La fouille de processus de configuration

L'approche adoptée par la méthode ConfiLog est fondée, entre autres, sur la reconstitution des modèles de processus de configuration antérieurs. En effet, les

modèles de processus reconstitués sont analysés pour en tirer des interprétations servant de guidage. Pour ce faire, ConfiLog utilise les techniques de la fouille de processus vu que ces dernières permettent d'appréhender les actions des parties prenantes et les étapes de configuration telles qu'elles se déroulent réellement.

Les sous-sections suivantes présentent la méthode conçue pour transformer les processus de configuration en un journal d'événements, les types de modèles de processus découverts par la fouille de processus et la manière avec laquelle les connaissances requises doivent être interprétées.

V.2.1. L'extraction des traces de processus de configuration

Le choix d'événements à enregistrer dans le journal d'événements a un impact fondamental sur la structure du modèle de processus reconstitué et son interprétation. Dans le travail actuel, l'objectif est de révéler des indicateurs de performance, de flexibilité et de personnalisation en rapport avec les questions de guidage (cf. Section 2). Par conséquent, l'activité de sélection ou de refus d'une variante donnée constitue l'événement le plus approprié pour des fins d'analyse de processus. Pour chaque événement, la méthode ConfiLog, capture les attributs listés dans le Tableau (10).

Tableau (10) : Attributs capturés dans le journal d'événements

Attribut	Identifiant et Description
ConfigID	A1: L'identifiant unique du processus de configuration
Originator	A2: La partie prenante qui opère la configuration
PLname	A3 : L'intitulé de la ligne de produits
VariantID	A4: L'identifiant unique de la variante
VariantName	A5 : L'intitulé de la variante
VariantType	A6 : Le type de la variante. Enum Type {Optionnelle, Obligatoire, Regroupée}
VariantValue	A7 : La valeur affectée à la variante
VariantDecisionType	A8 : Le type de décision. Enum Décision {Manuelle, propagée}
VariantDecisionStep	A9 : L'ordre consécutif de décision dans le processus
Timestamp	A10 : L'horodatage de décision

Chaque processus de configuration devrait avoir un identifiant unique `configID`, un auteur `originator` et une ligne de produits `PLName` à laquelle il fait référence. Pour chaque évènement l'identifiant du processus, l'intitulé `variantName` et le type `variantType` de la variante sur laquelle la configuration a eu lieu sont capturés. Le traçage d'évènements considère trois types de variantes, notamment : les variantes obligatoires, les variantes optionnelles et les variantes regroupées. Les variantes obligatoires doivent figurer dans tous les processus de configuration. Les variantes optionnelles sont des variantes facultatives que la partie prenante peut sélectionner ou pas. Les variantes regroupées sont des variantes de cardinalité de [1..3] par exemple. Également, pour chaque évènement le traçage capture si la variante était sélectionnée ou refusée par l'attribut `variantValue`. Le type de la décision avec laquelle une variante a été configurée est capturé par l'attribut `variantDecisionType`. Pour chaque configuration d'une variante, l'ordre dans l'ensemble du processus `VariantDecisionStep` et son horodatage `Timestamp` sont capturés.

V.2.2. La reconstitution des modèles de processus de configuration

La configuration des lignes de produits représentée en tant que processus montre très souvent des irrégularités. Ces irrégularités, induisant une non structuration des étapes de processus, peuvent revenir à plusieurs facteurs à savoir :

- L'abandon d'un processus de configuration avant sa finalisation,
- Les retours multiples sur les choix de configuration,
- Les décisions de configuration qui violent les contraintes du domaine, etc.

Compte tenu de cela, le "Fuzzy Miner" [Gunther et al., 2007] implémenté dans l'outil Disco² est sélectionné comme étant l'algorithme principal pour la reconstitution des modèles de processus dans la méthode ConfiLog (cf. chapitre 2).

La reconstitution des modèles de processus est opérée à partir du journal d'évènements traçant les attributs listés ci-dessus (cf. tableau 10). Le journal d'évènements est alimenté au fur et à mesure que des actions des parties prenantes sont produites. L'algorithme "Fuzzy Miner" permet de générer automatiquement un modèle de processus "brut" comme celui présenté en exemple dans la figure (13).

² Lien : www.fluxicon.com/disco/

Dans le présent modèle de processus, les nœuds sont paramétrés pour représenter les activités de configuration. Une activité de configuration revient à une sélection ou un rejet d'une variante du modèle de la ligne de produits. Par contre, les arcs sont paramétrés pour représenter le passage d'une activité de configuration à une autre. Généralement, les arcs sont découverts en fonction de l'ordre des activités de configuration et de leurs horodatages respectifs. Par ailleurs, les modèles de processus de configuration reconstitués sont accompagnés par des informations de fréquence et de performance, entre autres :

- Le nombre de processus de configuration,
- Le nombre des activités de configuration,
- La durée moyenne, médiane, maximale, minimale d'un processus de configuration,
- L'horodatage du début et fin des processus de configuration,
- Les processus de configuration similaires,
- Les parties prenantes avec des choix similaires, etc.

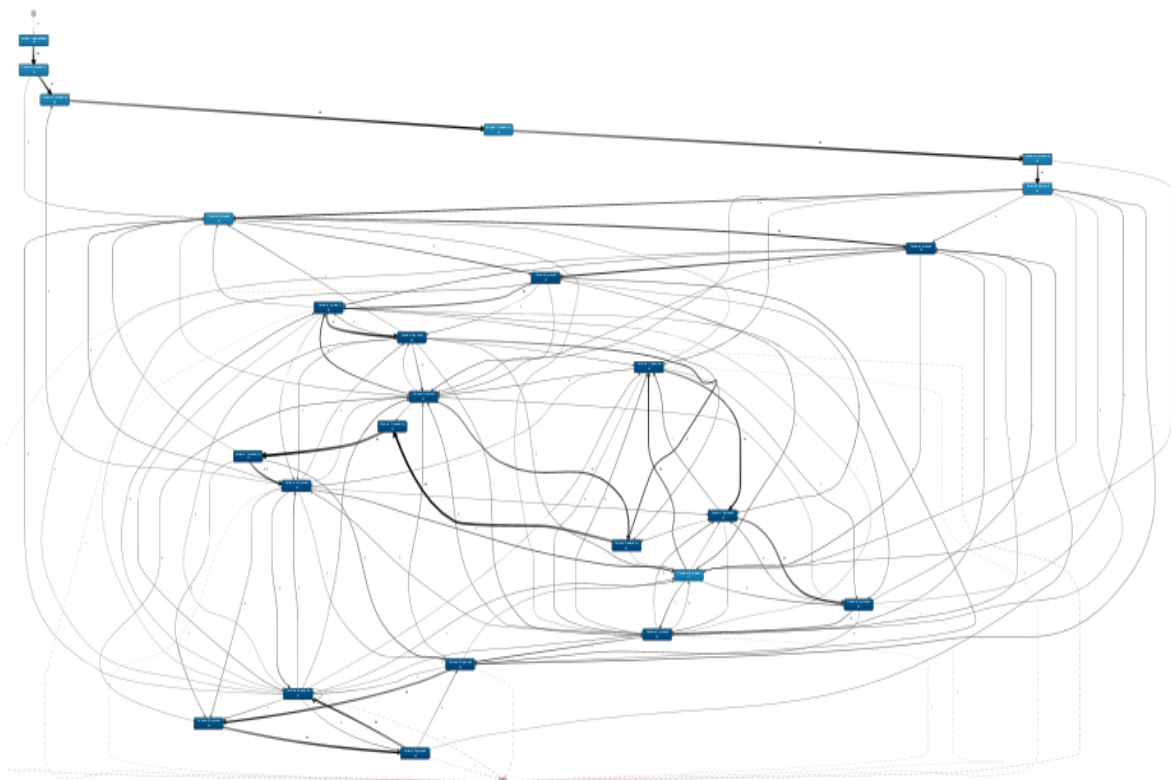


Figure (13) : Exemple de reconstitution d'un modèle de processus complet

A première vue, le modèle de processus reconstitué suite à des configurations d'une ligne de produits semble incompréhensible, d'où l'appellation de forme de spaghetti dans la littérature [Günther et al., 2007]. Ceci s'explique surtout par le fait que les données sont issues d'un cas réel mais pas encore filtré.

En réalité, les processus de configuration ne sont pas totalement linéaires impliquant souvent des activités de configuration qui ne produisent que très rarement et donc n'ayant pas une importance primordiale pour les différentes analyses. Pour y pallier, plusieurs possibilités sont disponibles, notamment, le filtrage et l'interprétation détaillés dans la section suivante.

V.2.3. Le filtrage et l'interprétation des modèles de processus de configuration

A la base, le modèle de processus reconstitué montre la totalité des activités de configuration. Certaines activités de configuration ne sont sollicitées qu'une ou deux fois sur des milliers de cas; ce qui est négligeable. Pour mieux tirer profit des modèles de processus et des informations calculées à partir des journaux d'événements, plusieurs filtres peuvent être appliqués, par exemple, prendre en compte que les traces contenant l'exécution d'une activité de configuration donnée. Par ailleurs, le modèle peut être adapté et ciblé à des situations particulières et des réponses particulières aux questions de guidage. La configurabilité adéquate de la taille du modèle de processus revient à paramétrer les fréquences, les performances, l'événement de début, l'événement de fin et/ou les filtres d'activité.

Techniquement, les traitements de fouille de la méthode ConfiLog se focalisent sur des indicateurs de corrélation, d'un point de vue processus plutôt que des données statiques, pour identifier les comportements similaires, simplifier le modèle de processus à différents niveaux d'abstraction souhaités et appliquer des filtres sur des activités qui se croisent. Dans certains cas, ConfiLog filtre les modèles de processus pour en extraire des parties, par exemple, pour surveiller une partie du processus dans différentes situations. De même, le filtrage des modèles de processus peut être très utile pour sélectionner uniquement les instances de processus qui commencent et/ou se clôturent par des activités particulières ou dans un laps de temps particulier.

Pour répondre aux questions de guidage, ConfiLog interroge le modèle de processus en utilisant les directives citées ci-dessous. Ces directives sont plus significatives s'il y a des milliers d'événements différents et la réponse à une question de guidage donnée se concentre uniquement sur des aspects particuliers.

- Mappage du modèle de processus de configuration : Cette directive combine plusieurs activités de configuration en une seule. Cependant, elle permet de détailler un ensemble d'activités et leur séquence.

- Filtrage des activités de configuration en double : Cette directive supprime les répétitions des activités de configuration contenus dans un modèle de processus.
- Filtrage des valeurs d'attribut: cette directive ne conserve que les événements qui ont une valeur d'attribut donnée. Par exemple, filtrer le modèle de processus de façon à garder uniquement les instances de processus qui ont commencé par la sélection d'une variante donnée.
- Filtrage de longueur d'instance de processus: cette directive spécifie les instances de processus à conserver en fonction d'un seuil paramétrable du nombre d'événements. Par exemple, filtrer le modèle de processus de façon à garder uniquement les instances de processus qui n'ont pas dépassé une dizaine d'activités de configuration.
- Filtrage de fréquence d'instance de processus: cette directive spécifie le nombre d'instances de processus. Ce traitement permet de se concentrer uniquement sur les instances de processus les plus fréquentes. Par exemple, filtrer le modèle de processus de façon à garder uniquement les instances de processus qui se sont répétés autant de fois en fonction d'un pourcentage de fréquence. Ce filtrage est plus significatif s'il y a des milliers d'événements différents pour se focaliser uniquement sur les activités qui se produisent fréquemment ou dans un laps de temps bien déterminé.

Si par exemple les experts souhaitent recommander aux parties prenantes des processus de configuration rapides, ils ont tendance à filtrer le modèle de processus complet (cf. Figure 13) de manière à garder les processus qui ont abouti à une configuration complète et qui ont une durée au-dessous de la durée moyenne de l'ensemble des processus de configuration. Le modèle de processus filtré par rapport à la figure (13) est repris dans la Figure (14). Ce filtrage est opéré via les directives citées ci dessus.

Quelque soit l'analyse, cette dernière doit répondre aux questions de guidage posées au préalable et ceci est la résultante d'une compréhension des besoins de la recommandation en question et de ses divers aboutissants répertoriés dans la base de la méthode (GQM).

Les résultats produits après filtrage et interprétation du modèle de processus sont repris sous forme d'un script informatique qui est enregistré dans des datasets de guidage. Ces datasets de guidage représentent l'essence du moteur de recommandation de ConfiLog.

Les experts mettent à jour les datasets de guidage, périodiquement, une fois que de nouveaux évènements se sont produits et d'autres interprétations sont envisagées.

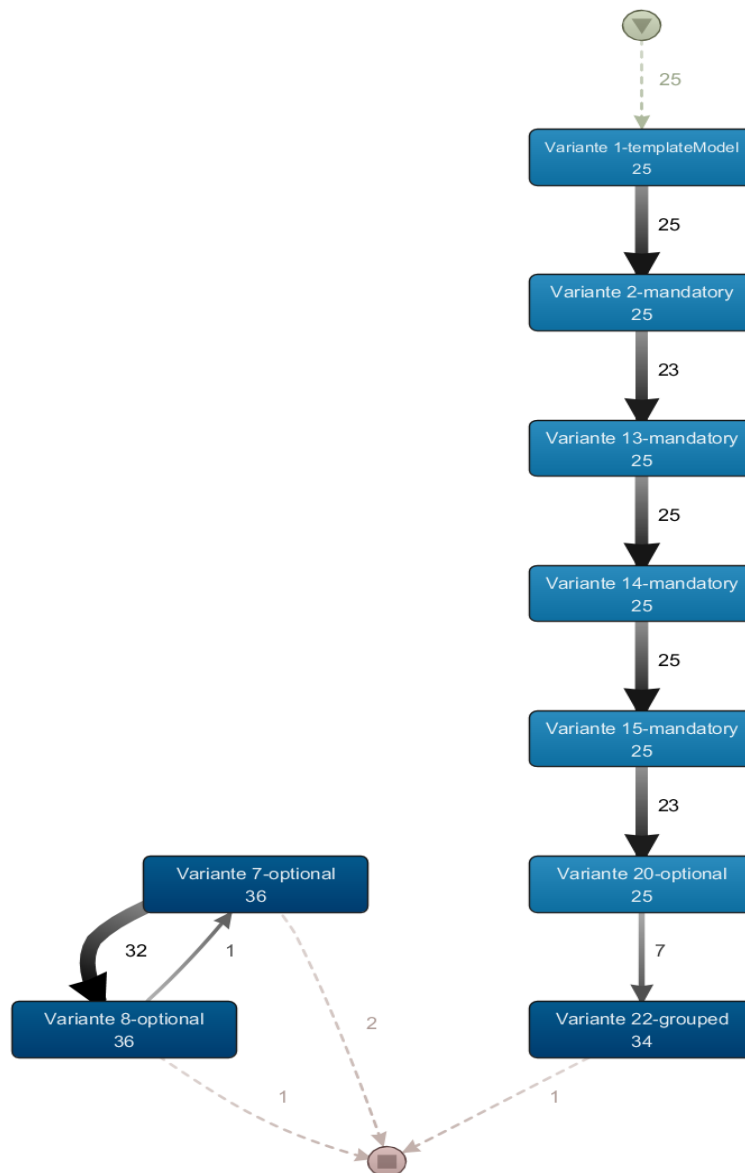


Figure (14) : Exemple de modèle de processus filtré

V.3. Calcul de la recommandation

Le guidage produit par la méthode ConfiLog est un ensemble de recommandations dynamiques allant sur deux dimensions, notamment:

- (i) L'ordre des décisions à prendre dans le processus de configuration et
- (ii) Les instances de décisions à prendre dans le processus de configuration.

Le guidage dit dynamique est proposé à chaque étape de configuration. Par ailleurs, le calcul de ce guidage est fondé sur :

- (i) Un but de guidage donné par la partie prenante (Rapidité, Flexibilité ou personnalisation) et
- (ii) Une configuration partielle, qui correspond à une affectation de valeurs aux variantes de modèle par la partie prenante.

La Figure (15) détaille les étapes du processus de recommandation de la méthode ConfiLog à savoir :

- Le paramétrage du processus de configuration,
- L'abstraction et le filtrage des réponses de fouille,
- La pondération des réponses de fouille et
- Le calcul d'impact de la décision.

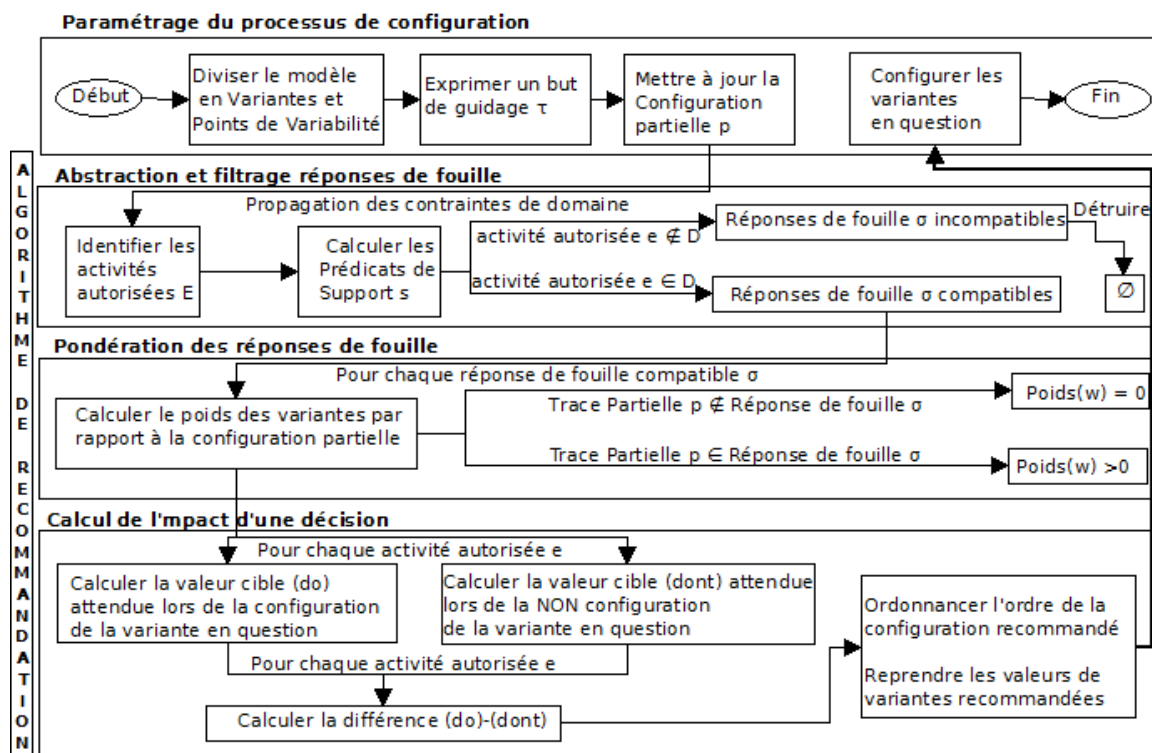


Figure (15) : Processus de recommandation ConfiLog

V.3.1. Le paramétrage du processus de configuration

La méthode ConfiLog divise le modèle de la ligne de produits à configurer, par les parties prenantes, en points de variabilité. Chaque point de variabilité couvre des variantes sur lesquelles la partie prenante effectue des activités de configuration. Une

activité de configuration a correspond au fait de sélectionner ou rejeter une variante donnée.

Avant d'entamer la configuration, la partie prenante exprime son but de guidage. Le but de guidage τ est paramétrable par la partie prenante, $\text{Enum } \tau \{ \text{Rapidité, Flexibilité, Personnalisation} \}$.

Dès que la partie prenante effectue une activité de configuration a , ses choix sont pris en compte afin de mettre à jour la configuration partielle $\rho \in A^*$. En parallèle, les contraintes du domaine sont propagées automatiquement. Par conséquent, des nouvelles activités autorisées E apparaissent pour initier de nouvelles recommandations. Une recommandation est initiée par une requête de recommandation r . Formellement, une requête de recommandation r est définie par un tuple $r(\tau, \rho, E)$ avec :

- Un but de guidage τ ,
- Une trace partielle $\rho \in A^*$ et
- Un ensemble d'activités autorisées (le reste des variantes du modèle à configurer qui vont avec la trace partielle) $E \subseteq A$.

Les activités autorisées E constituent le paramètre d'abstraction et de filtrage des réponses de guidage.

V.3.2. L'abstraction et le filtrage des réponses de fouille

Au fur et à mesure que le processus de configuration progresse, des activités deviennent autorisées E et d'autres deviennent censurées vu qu'elles violent les contraintes du domaine. Les recommandations sont calculées uniquement pour les activités autorisées E en considérant des réponses de fouille.

Une réponse de fouille $\sigma \in A^*$ est une séquence finie d'activités A , donnée par une fonction $\sigma: \{1, \dots, n\} \rightarrow A$ et $\sigma': \{1, \dots, m\} \rightarrow B$. Les réponses de fouille répondent à un but de guidage donné avec un apport $\tau(\sigma)$ comme les exemples ci-après. Les apports des réponses de fouille sont des indicateurs identifiés grâce à la fouille de processus de configuration et stockés dans un dataset de guidage.

- $\text{Rapidité}(x)=500$: La réponse de fouille (x) a un apport de Rapidité de 500 secondes.
- $\text{Flexibilité}(y)=15$: La réponse de fouille (y) a un apport de Flexibilité de 15.
- $\text{Personnalisation}(z)=5$: La réponse de fouille (z) a un apport de Personnalisation de 5.

Un Dataset de guidage $D \in \beta(A^*)$ est un multi-ensemble des réponses de fouille se référant aux activités de A . Pour chaque but de guidage, le dataset contient des réponses de fouille sélectionnées pour répondre aux questions de guidage et représenter les instances de processus rapides, flexibles et personnalisés.

Seules les réponses de fouille jugées compatibles avec la recommandation en question doivent être retenues. La pertinence d'une réponse de fouille pour produire une recommandation est déterminée à base de support de cette réponse à la recommandation envisagée. Les réponses de fouille qui ne supportent pas (i) le but de guidage τ , (ii) la trace partielle ρ et (iii) les activités autorisées ϵ sont éradiquées. Pour ce faire, un prédicat $s(\tau, \rho, \sigma, \epsilon)$ est appliqué jusqu'à filtrage de toutes les réponses inutiles. L'opérateur membre (\in) identifie l'appartenance d'une activité dans une réponse de fouille donnée: $a \in \sigma \iff \exists 1 \leq i \leq n \text{ ai} = a$. Le dataset filtré est défini comme suit : $D_s(\tau, \rho, \epsilon) = [\sigma \in D \mid s(\tau, \rho, \sigma, \epsilon)]$.

Pour classer le degré de pertinence des réponses de fouille retenues pour la recommandation, ces dernières sont pondérées. Les réponses de fouille présentant un degré élevé de corrélation avec la trace partielle doivent être pondérées plus élevées que celles avec une corrélation faible ou nulle.

V.3.3. La pondération des réponses de fouille

Les réponses de fouille σ retenues supportant une activité de configuration a n'ont pas la même importance, c'est-à-dire que certaines réponses de fouille peuvent concorder avec la trace partielle ρ mieux que d'autres réponses. Par conséquent, des fonctions de pondération (ω) sont définies en affectant un poids pour chaque réponse de fouille. Une fonction de pondération (ω) est définie pour exprimer l'importance de chaque réponse de fouille σ pour la recommandation d'une activité autorisée ϵ . Le poids d'une réponse de fouille peut être compris entre 1 et 0 :

- La valeur 1 indique que la réponse de fouille σ et la trace partielle ρ correspondent entièrement.
- La valeur 0 indique que la réponse de fouille σ et la trace partielle ρ ne correspondent pas du tout.

Par ailleurs, le poids de la réponse de fouille $\omega(\tau, \rho, \sigma)$ est calculé en fonction d'une division DIV du nombre des activités de la trace partielle incluses \cap dans la réponse de fouille par le nombre des activités de la trace partielle ρ .

$\omega(\tau, \rho, \sigma) = |\rho \cap \sigma| \text{ DIV } |\rho|$ est une pondération de réponse de fouille.

Les pondérations de réponses de fouille sont exploitées pour le calcul de l'impact de chaque décision de configuration à venir pour la partie prenante.

V.3.4. Le calcul de l'impact de décision

Pour chaque activité autorisée e , l'algorithme de la méthode ConfiLog calcule son impact de décision en fonction de deux facteurs :

- Une valeur cible attendue lors de la configuration (**do**) de e et
- Une valeur cible attendue lors de la NON configuration (**dont**) de e .

La valeur cible attendue lors de la configuration (**do**) de l'activité autorisée e est calculée en fonction de la sommation \sum des poids des réponses de fouille compatibles $\omega(\tau, \rho, \sigma)$ multipliés par leur apport $\tau(\sigma)$; le tout divisé par la sommation \sum des poids des réponses de fouille $\omega(\tau, \rho, \sigma)$:

$$\text{do}(e, \tau, \rho, D) = \frac{\sum_{\sigma \in Ds(\tau, \rho, e)} [\omega(\tau, \rho, \sigma) * \tau(\sigma)]}{\sum_{\sigma \in Ds(\tau, \rho, e)} \omega(\tau, \rho, \sigma)}$$

De même, la valeur cible attendue lors de la NON configuration (**dont**) de l'activité autorisée e est calculée de la même manière de la valeur (**do**), néanmoins, la (**dont**) considère plutôt les réponses de fouille qui ne supportent pas l'activité autorisée e en question.

$$\text{dont}(e, \tau, \rho, D) = \frac{\sum_{e' \in E \setminus \{e\}} \sum_{\sigma \in Ds(\tau, \rho, e) \setminus Ds(\tau, \rho, e')} [\omega(\tau, \rho, \sigma) * \tau(\sigma)]}{\sum_{e' \in E \setminus \{e\}} \sum_{\sigma \in Ds(\tau, \rho, e') \setminus Ds(\tau, \rho, e)} \omega(\tau, \rho, \sigma)}$$

Les calculs de la valeur (**do**) et (**dont**) sont itératifs et doivent être recalculés à chaque étape de configuration c'est-à-dire à chaque fois que la configuration partielle ρ est mise à jour. Pour chaque étape de configuration, l'algorithme ConfiLog ordonne les activités à recommander en fonction de la soustraction des valeurs [**do** - **dont**]. En effet, les activités ayant le plus grand résultat de soustraction sont recommandés en premier si l'algorithme cherche à *minimiser* le temps. Cependant, si l'algorithme cherche à *maximiser* la flexibilité ou la personnalisation, les activités ayant le plus faible résultat de soustraction sont recommandées en premier.

V.3.5. Démonstration du moteur de recommandation ConfiLog avec la ligne de sites web

Pour des raisons de simplification, la présente section reprend la ligne de Siteweb (cf. chapitre IV) à un état de configuration là où les branches de performance et les branche de ServeurWeb sont toutes configurées. La démonstration de l'algorithme de recommandation de ConfiLog avec une deuxième ligne de produits du monde de la téléphonie est reprise en annexe A.

A l'état, le système est invité à produire du guidage à la partie prenante en rapport avec la branche Services additionnels. L'état de la configuration au moment de la production du guidage est repris dans la figure (16). Les variantes qui sont retenues sont cochées, les variantes qui sont refusées sont barrées, ce qu'en restent sont en attente de configuration.

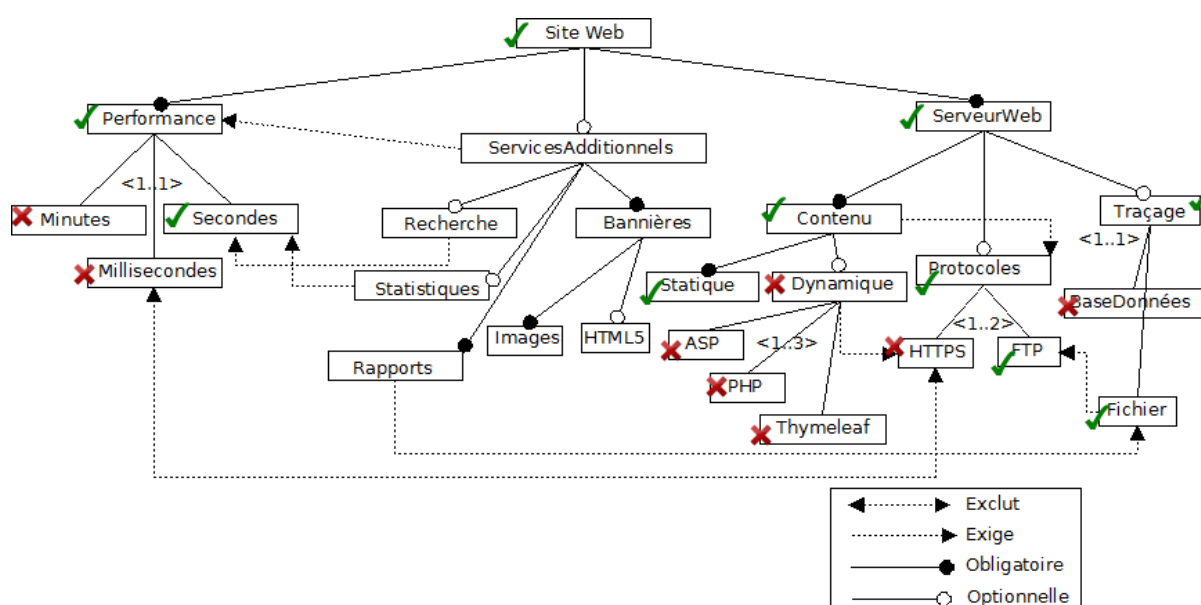


Figure (16) :Etat de configuration de la Ligne de produits de Site Web

A l'instant, la partie prenante a déjà effectué des choix de sélection ou de refus sur les variantes [SiteWeb, Performance, Minutes, Millisecondes, Secondes, ServeurWeb, Contenu, Statique, Dynamique, ASP, PHP, Thymeleaf, Protocoles, HTTPS, FTP, Trçage, BaseDonnées, Fichier] et laisse le choix à la partie prenante pour la sélection ou le refus des variantes suivantes : [ServicesAdditionnels, Recherche, Statistiques, Rapports, Bannières, Images, HTML5].

La suite de cette section considère le scénario où la partie prenante introduit son but de guidage, qu'il ait un processus de configuration rapide τ = Rapidité.

Après fouille de processus, le dataset de guidage ρ est alimenté par un modèle de processus sélectionnant uniquement les instances de processus qui ont été configurées rapidement :

- Rapidité(1)=100 secondes: [SiteWeb-selected, Performance-selected, Minutes-deselected, Millisecondes-deselected, Secondes-selected, ServeurWeb-selected, Contenu-selected, Statique-selected, Dynamique-deselected, ASP-deselected, PHP-deselected, Thymeleaf-deselected, Protocoles-selected, HTTPS-deselected, FTP-selected, Traçage-selected, BaseDonnées-deselected, Fichier-selected, ServicesAdditionnels-selected, Recherche-selected, Statistiques-deselected, HTML5-selected]
- Rapidité(2)=200 secondes: [SiteWeb-selected, Performance-selected, Minutes-deselected, Millisecondes-deselected, Secondes-selected, ServeurWeb-selected, Contenu-selected, Statique-selected, Dynamique-deselected, ASP-deselected, PHP-deselected, Thymeleaf-deselected, Protocoles-selected, HTTPS-deselected, FTP-selected, Traçage-selected, BaseDonnées-deselected, Fichier-selected, ServicesAdditionnels-selected, Recherche-deselected, Statistiques-selected]
- Rapidité(3)=150 secondes: [SiteWeb-selected, Performance-selected, Minutes-deselected, Millisecondes-deselected, Secondes-selected, ServeurWeb-selected, Contenu-selected, Statique-selected, Dynamique-deselected, ASP-deselected, PHP-deselected, Thymeleaf-deselected, Protocoles-selected, HTTPS-deselected, FTP-selected, Traçage-selected, BaseDonnées-deselected, Fichier-selected, ServicesAdditionnels-selected, Recherche-selected, Statistiques-selected, HTML5-deselected]

La partie prenante continue à suivre un ensemble d'étapes de processus de configuration jusqu'à finir la configuration de son site web souhaité.

V.3.5.1. Etape n du processus de configuration de la ligne de sites web

La partie prenante sélectionne la variante `ServicesAdditionnels`. Alors, le système propage automatiquement la sélection de `Bannières` et `Images` étant donné que ce sont deux variantes obligatoires dans les `ServicesAdditionnels`. La variable configuration partielle ρ et la variable activités autorisées E sont mis à jour comme suit: $\rho = [\text{SiteWeb-selected, Performance-selected, Minutes-deselected, Millisecondes-deselected, Secondes-selected, ServeurWeb-selected, Contenu-selected, Statique-selected, Dynamique-deselected, ASP-deselected, PHP-deselected, Thymeleaf-deselected, Protocoles-selected, HTTPS-deselected, FTP-selected, Traçage-selected, BaseDonnées-deselected, Fichier-selected, Servicesadditionnels-selected}]$; $E = [\text{Recherche, Statistiques, HTML5}]$.

Le calcul des prédicats en fonction des activités autorisées est détaillé dans le tableau (11). En effet, le tableau contient trois colonnes, une colonne pour chaque activité autorisée et trois lignes, une ligne pour chaque réponse de fouille enregistrée dans le dataset de guidage. Les "τ" signifient qu'il y a eu, effectivement, une décision de sélection ou de refus de cette activité autorisée dans la réponse de fouille en question. Les "F"

signifient qu'aucune décision sur l'activité autorisée n'a eu lieu dans la réponse de fouille en question.

Tableau (11) : Calcul des prédicats pour étape 1

	Recherche	Statistiques	HTML5
Rapidité(1)=100 secondes	T	T	T
Rapidité(2)=200 secondes	T	T	F
Rapidité(3)=150 secondes	T	T	T

Dans ce cas, tous les processus sont à pondérer parce qu'ils contiennent au moins une activité autorisée τ .

Le calcul de pondération des processus s'effectue comme suit :

$$\text{Pondération Rapidité(1):} = \frac{19}{19} = 1$$

$$\text{Pondération Rapidité (2):} = \frac{19}{19} = 1$$

$$\text{Pondération Rapidité (3):} = \frac{19}{19} = 1$$

Le dividende correspond au nombre des variantes qui sont en commun entre la trace partielle et la trace de fouille sur laquelle va se baser le guidage. Le diviseur correspond au nombre des variantes qui sont dans la configuration partielle.

La valeur cible attendue lors de la configuration de chaque activité autorisée e s'effectue comme suit : Do ((Recherche, Statistique, HTML5), Rapidité, (p), D).

Pour la variante HTML5, l'algorithme ne considère pas la réponse de fouille Rapidité (2), parce qu'elle ne figure pas parmi ses variantes (cf. tableau 11).

Configuration de la variante Recherche :

$$\text{Do (Recherche)} = (1 * 100 \text{ secondes}) + (1 * 200 \text{ secondes}) + (1 * 150 \text{ secondes}) = 450$$

Configuration de la variante Statistique :

Do (Statistique) = (1 * 100 secondes) + (1 * 200 secondes) + (1 * 150 secondes) = 450

Configuration de la variante HTML5 :

Do (HTML5) = (1 * 100 secondes) + (1 * 150 secondes) = 250

La valeur cible attendue lors de la NON configuration de chaque activité autorisée s'effectue comme suit : Dont ((Recherche, Statistique, HTML5), Rapidité, (p) ,D). En effet, l'algorithme de ConfiLog ne considère que les prédicats F pour chaque réponse de fouille. Par exemple, pour le calcul de la NON configuration de la variante HTML5, seule la réponse de fouille Rapidité(2) sera considérée (cf. tableau 11).

NON configuration de la variante Recherche :

Dont (Recherche) = 0

NON configuration de la variante Statistique :

Dont (Statistique) = 0

NON configuration de la variante HTML5 :

Dont(HTML5) = (1*100 secondes) = (1* 200 secondes) = 200

Le calcul de la différence entre les fonctions do et dont s'effectue comme suit :

La différence do et dont pour la variante Recherche :

(Recherche, 450, 0) = 450

La différence do et dont pour la variante Statistique :

(Statistique, 450, 0) = 450

La différence do et dont pour la variante HTML5 :

(HTML5, 250, 200) = 200

Pour l'étape 1, l'algorithme ConfiLog recommande les variantes ayant le résultat de soustraction [do - dont] le plus élevé en premier. En effet, l'ordre de processus recommandé pour cette étape de configuration est le suivant : (Select: Recherche ≤ Deselect: Statistiques < Select: HTML5)

V.3.5.2. Etape n+1 du processus de configuration de la ligne de sites web

La partie prenante ne prend pas en compte le guidage fourni par ConfiLog et rejette la variante Recherche. Alors la variable configuration partielle ρ et la variable activités autorisées E sont mis à jour comme suit : $\rho = [\text{SiteWeb-selected}, \text{Performance-selected}, \text{Minutes-deselected}, \text{Millisecondes-deselected}, \text{Secondes-selected}, \text{ServeurWeb-selected}, \text{Contenu-selected}, \text{Statique-selected}, \text{Dynamique-deselected}, \text{ASP-deselected}, \text{PHP-deselected}, \text{Thymeleaf-deselected}, \text{Protocoles-selected}, \text{HTTPS-deselected}, \text{FTP-selected}, \text{Traçage-selected}, \text{BaseDonnées-deselected}, \text{Fichier-selected}, \text{ServicesAdditionnels-selected}, \text{Recherche-deselected}] ;$
 $E = [\text{Statistiques}, \text{HTML5}]$.

Le calcul des prédicats en fonction des activités autorisées est détaillé dans tableau (12). En effet, le tableau contient une colonne de moins qui présente la variante Recherche qui vient d'être rejetée par la partie prenante. Le reste des valeurs du tableau restent inchangées par rapport au tableau (11).

Tableau (12) : Calcul des prédicats pour étape 2

	Statistique	HTML5
Rapidité(1)=100 secondes	T	T
Rapidité(2)=200 secondes	T	F
Rapidité(3)=150 secondes	T	T

Dans ce cas, les trois réponses de fouille sont à tenir en compte parce qu'elles contiennent au moins une activité autorisée.

Le calcul de pondération des processus s'effectue comme suit :

$$\text{Pondération Rapidité(1)} : \frac{19}{20} = 0,95$$

$$\text{Pondération Rapidité(2)} : \frac{19}{19} = 1$$

$$\text{Pondération Rapidité(3)} : \frac{19}{20} = 0,95$$

La valeur cible attendue lors de la configuration de chaque activité autorisée s'effectue comme suit : Do ((Statistiques, HTML5), Rapidité, (p), D)

Configuration de la variante Statistiques :

$$\text{Do (Statistiques)} = (0,95 * 100 \text{ secondes}) + (1 * 200 \text{ secondes}) + (0,95 * 150 \text{ secondes}) = 437,5$$

Configuration de la variante HTML5 :

$$\text{do(HTML5)} = (0,95 * 200 \text{ secondes}) + (0,95 * 150 \text{ secondes}) = 332,5$$

La valeur cible attendue lors de la NON configuration de chaque activité autorisée s'effectue comme suit : Dont ((Statistiques, HTML5), Rapidité, (p), D)

NON configuration de la variante Statistiques =

$$\text{Dont (Statistiques)} = 0$$

NON configuration de la variante HTML5 =

$$\text{Dont (HTML5)} = (1 * 200 \text{ secondes}) = 200$$

Le calcul de la différence entre la fonction do et dont s'effectue comme suit :

La différence do et dont pour la variante Statistiques :

$$\text{(Statistiques, 437,5, 0)} = 437,5$$

La différence do et dont pour la variante HTML5 :

$$\text{(HTML5, 332,5, 200)} = 132,5$$

Pour l'étape 2, l'algorithme ConfiLog recommande les variantes ayant le résultat de soustraction [do - dont]. En effet, l'ordre de processus recommandé pour cette étape de configuration est le suivant : (Deselect: Statistiques < Select: HTML5)

La partie prenante fait son choix sur la variante statistiques, alors pour l'étape 3 de la configuration il ne reste qu'une variante à recommander. La partie prenante la configure et finalise le processus de configuration.

V.4. Conclusion

La recommandation calculée par ConfiLog opère autour des buts de guidage suivants :

- Réduire le temps requis pour atteindre une configuration valide du point de vue du processus,
- Maximiser la flexibilité et l'efficacité de la configuration du point de vue du processus,
- Maximiser la personnalisation de la configuration du point de vue du processus,

- Promouvoir la recommandation de la configuration du point de vue des intérêts de l'entreprise.

Afin d'atteindre ces buts exprimés par la partie prenante, des questions de guidage permettant d'interroger les modèles de processus reconstitués, sont posées. Les réponses à ces questions de guidage apportées via des directives de fouille de processus sont évaluées en utilisant des métriques.

Par ailleurs, un processus de recommandation est conduit afin de réutiliser les réponses de guidage et les orchestrer tout au long de processus de guidage. Pour ce faire, ConfiLog suit 4 étapes principales, à savoir :

- Le paramétrage du processus de configuration,
- L'abstraction et le filtrage des réponses de fouille,
- La pondération des réponses de fouille et
- Le calcul d'impact de la décision.

Les décisions qui auront un impact en faveur de l'atteinte du but de guidage en question seront recommandées. Cette méthode de guidage pour les lignes de produits est générique vu qu'elle est applicable sur tout modèle de variabilité spécifiant une ligne de produits et/ou un logiciel configurable. L'outillage de ConfiLog et son évaluation sont étudiés respectivement dans les deux chapitres suivants.

[Cette page est intentionnellement laissée en blanc]

PARTIE C : APPLICATION

Chapitre VI : Outillage

Chapitre VII : Evaluation

Chapitre VI : Outillage

VI.1. Introduction	96
VI.2. ConfiLog : Les choix d'ingénierie	96
VI.2.1. Application Internet Riche	96
VI.2.3. Architecture en couches	97
VI.2.4. Choix d'implémentation	98
VI.2. Architecture logicielle de ConfiLog	99
VI.2.1. Côté client	100
<i>VI.2.1.1. Modélisateur de lignes de produits</i>	<i>100</i>
<i>VI.2.1.2. Configureur de lignes de produits</i>	<i>103</i>
VI.2.2. Côté serveur : La couche de configuration	106
<i>VI.2.2.1. Moteur d'interfaces</i>	<i>106</i>
<i>VI.2.2.2. Gestionnaire de configuration</i>	<i>106</i>
VI.2.3. Côté serveur : La couche de fouille	107
<i>VI.2.2.1. Chargeur (ETL)</i>	<i>107</i>
<i>VI.2.2.2. Fouilleur de processus</i>	<i>109</i>
VI.2.4. Côté serveur : La couche de recommandation	112
<i>VI.2.3.1. Moteur de recommandation</i>	<i>113</i>
<i>VI.2.3.2. Conducteur de recommandation</i>	<i>115</i>
VI.6. Conclusion	118

Chapitre VI

Outillage

VI.1. Introduction

Ce chapitre décrit la mise en oeuvre de l'outillage de la méthode ConfiLog^{1/2} afin d'opérationnaliser le guidage de la configuration. La première section est une argumentation des choix d'ingénierie en rapport avec le type d'application implémentée, l'architecture logicielle adoptée et les implémentations. Alors que, la section 2 détaille l'architecture logicielle de ConfiLog en se basant sur les modules fonctionnels de chaque couche. Des interfaces graphiques et des lignes de code du configurateur de ConfiLog sont illustrées, en application d'une ligne de sites web.

VI.2. ConfiLog : Les choix d'ingénierie

Lors de l'implémentation de la méthode ConfiLog, plusieurs solutions d'ingénierie étaient envisagées. Tout bien considéré, il était décidé d'implémenter une Application de type Internet Riche, supportant une architecture en couche et réutilisant des outils open source.

VI.2.1. Application Internet Riche

Pour répondre au mieux aux besoins d'une configuration guidée faisant intervenir, entre autres, des personnes physiquement éloignées, ConfiLog a été développée comme étant une application web. En considérant la complexité de la potentielle application Web, il a fallu choisir entre le développement d'une Application Internet Riche (RIA : Rich Internet Application) [Fraternali et al., 2010] et le développement d'une Application Client Riche (RCA: Rich Client Application) [Anbuselvan, 2015]. En définitive, il a été décidé de développer une (RIA) vu que ce type d'application est plus adapté pour les applications qui proposent une expérience utilisateur riche engendrant beaucoup de calculs et de charge pour l'application. En effet, l'interface de configuration

¹ Lien : www.youtube.com/watch?v=Ds69w86GCzQ&t=43s

² Lien : www.github.com/HoussemCHEMINGUI/ConfigLogFFV

de ConfiLog doit permettre une forte interactivité indépendamment du matériel et des ressources du côté client. De plus, les (RIA) sont simplement adaptables à des architectures en couches qui incluent des couches de présentation, de données et de métiers. Néanmoins, ces considérations privent d'utiliser amplement les ressources côté client dans l'exécution du processus de configuration à comparer avec une (RCA).

VI.2.3. Architecture en couches

L'organisation en couches s'avère importante face à différentes considérations du potentiel outillage de la méthode ConfiLog à savoir la conduite du processus de configuration, l'extraction des traces de processus, le filtrage et la transformation des journaux d'événements, les traitements de fouille de processus et le calcul de la recommandation. En outre, l'architecture en couches permet à l'outillage de ConfiLog de mieux répondre aux besoins fonctionnels à savoir la maintenabilité et l'évolutivité, la performance, la modularité de développement et finalement la testabilité :

- **Maintenabilité et Évolutivité** : Les couches de ConfiLog regroupent des traitements de nature commune ce qui fait que rarement qu'une modification particulière induit un impact sur plus d'une couche. Les évolutions d'une couche donnée n'impactent pas forcément les couches voisines. En séparant ConfiLog en plusieurs niveaux, des modifications et des ajouts de traitements de fouille n'ont pas amené à retravailler l'ensemble de l'application. L'évolutivité de ConfiLog pour répondre à de nouveaux besoins et/ou changements est souple vu que les fonctions ne sont pas monolithiques.
- **Performance** : Généralement la fouille de processus impliquant des milliers voir des millions d'événements présente une charge énorme sur le système. Pareillement, le calcul de la cohérence de la configuration et de la recommandation, à partir d'un grand espace de solutions, présente un obstacle de performance. Par conséquent, la charge de traitement des requêtes des parties prenantes étant divisée sur plusieurs couches permet une réactivité performante sans ralentir pour autant l'exécution. En effet, la division en couches permet l'allégement de l'interface de configuration en affectant les calculs lourds entre différentes couches revenant à différents serveurs.
- **Modularité de développement** : La cohésion des modules de chaque couche permet de diviser le développement sur plusieurs intervenants, ayant différentes connaissances et en différentes périodes. Le développement de ConfiLog a été

conduit sur plusieurs étapes au fur et à mesure que des connaissances supplémentaires sur les modules de la méthode ont été acquises. Chaque période de développement avait une responsabilité définie par la couche.

- Testabilité : L'organisation en couches permet à ConfiLog la réalisation de tests indépendants, chaque couche à part. L'isolation des couches permet d'identifier rapidement les sources des obstacles et les bogues qui entravent le processus de recommandation dans son évolution.

VI.2.4. Choix d'implémentation

Il était question de choisir un langage de programmation hautement structuré pour supporter l'architecture en couches et les potentielles contraintes à savoir les adaptations et les éventuels enrichissements de chaque couche. En effet, cela a fait pencher cette recherche entre deux choix : développer un outil à partir de rien ou bien étendre un open source déjà existant. Après plusieurs itérations dans les deux sens, la réutilisation de l'open source Splot³ semblait intéressante dans le sens qu'elle va permettre d'épargner énormément de temps et d'effort. En outre, Splot propose une licence de type MIT [Lin et al., 2006] donnant permission à réutiliser librement l'intégralité de son code à moins qu'elle soit citée dans les potentielles extensions.

Dans le contexte de ConfiLog, cette réutilisation permet d'exploiter des lignes de code en rapport avec la modélisation des lignes de produits et de calcul de cohérence des modèles. Néanmoins, l'extension ConfiLog envisageait aussi à enrichir ce code en instaurant les fonctionnalités suivantes :

- Considérer une structure orthogonale plutôt que hiérarchique pour la modélisation des lignes de produits;
- Tracer le profil et les actions de la partie prenante tout au long du processus de configuration;
- Sauvegarder et exporter les traces de processus de configuration;
- Fouiller les traces de processus de configuration et découvrir des modèles de processus;
- Calculer le guidage à travers un moteur de recommandation orchestré par la méthode (GQM); et
- Produire des interfaces dynamiques pour servir le guidage aux parties prenantes.

³ Lien : www.github.com/marcilio/splot

Pour implémenter ces extensions, le code de ConfiLog a été écrit en trois langages de programmation, notamment : HTML, Java, et Javascript associés à un moteur de templates Freemarker⁴. En effet, Freemarker extrait les données des objets et les intègre dans le code HTML selon une maquette définie à l'avance sans faire recours à un langage de script comme PHP. L'utilisation de Freemarker remplace, pareillement, l'utilisation des Pages Serveur Java (JSP) devenues obsolètes pour la séparation des considérations et des vues.

VI.2. Architecture logicielle de ConfiLog

Face aux différentes considérations d'ingénierie, une architecture logicielle client - serveur en couches a été préconisée pour concrétiser la méthode ConfiLog.

La Figure (17) détaille l'architecture logicielle de la méthode ConfiLog.

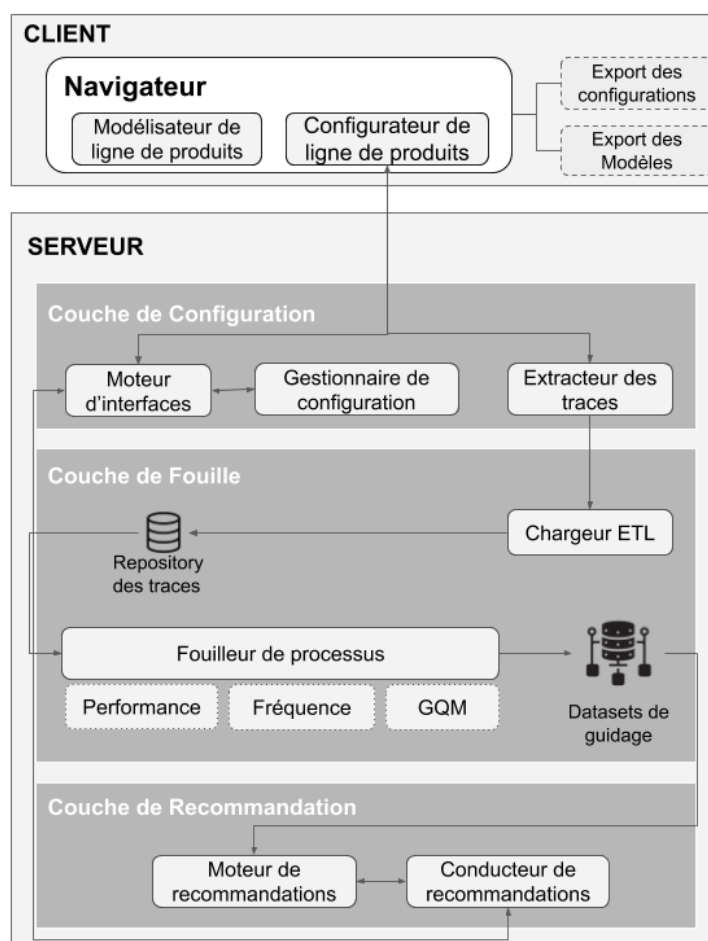


Figure (17) : L'architecture logicielle de ConfiLog

⁴ Lien : <https://freemarker.apache.org/>

Du côté Client, les parties prenantes utilisent le système via un navigateur qui leur propose des outils pour construire des modèles de variabilité de lignes de produits. En outre, le système propose des interfaces de configuration afin d'effectuer des choix à partir des variantes du modèle de la ligne de produits.

Du côté serveur, les traitements de configuration sont enchaînés par plusieurs modules se référant aux couches de l'architecture de ConfiLog mentionnées ci dessous :

- La couche de Configuration,
- La couche de Fouille et
- La couche de Recommandation.

Généralement, les architectures des configurateurs logiciels, basées sur les paradigmes de recommandation classiques ou encore sur la simple propagation des contraintes, considèrent une couche de configuration et une couche de recommandation. Cependant, l'architecture ConfiLog rajoute une couche de fouille de processus qui récupère des traces de processus de la couche configuration en fonction desquelles des modèles de processus, servant de recommandation, sont reconstitués.

VI.2.1. Côté client

Le côté client de l'architecture de ConfiLog distingue deux modules principaux à savoir :

- Le modélisateur de lignes de produits et
- Le configurateur de lignes de produits.

La Figure (18) illustre l'écran d'accueil affiché aux utilisateurs de ConfiLog, afin de modéliser des lignes de produits et les configurer plus tard.

VI.2.1.1. Modélisateur de lignes de produits

Il existe de nombreux logiciels, aussi bien payants que gratuits, pour modéliser des lignes de produits et les configurer ensuite tels que VariaMos⁵, FeatureIDE⁶, pure::variants⁷ et FAMA⁸. Leurs fonctionnements sont relativement simples : il suffit de modéliser soigneusement les variantes de la ligne de produits et les contraintes pour générer une interface de configuration. Ces derniers diffèrent dans le traitement de mêmes concepts. Par exemple, une variante est appelée "feature" dans FeatureIDE et appelée "Variable" dans VariaMos.

⁵ Lien : <https://variarnos.com/home/>

⁶ Lien : <http://www.featureide.com/>

⁷ Lien : <https://www.pure-svstems.com/>

⁸ Lien : https://www.isa.us.es/fama/?FaMa_Framework

CONFILOG
Mining process LOGs to enhance product line CONFIGurations

Houssem CHEMINGUI, Ines GAM, Raul MAZO, Camille SALINESI, Henda BEN GHEZALA
Université Panthéon Sorbonne Paris 1, France
Université de la Manouba, Tunisie

Janvier 2019

Home Product Line Model Editor Product Line Configuration Product Line Model Repository Contact Us

Edit Product Line Models Configure Product Line Models Consult Product Line Models

What's New?

Jan 2019 ConfiLog is now under construction in the [CRI Laboratory!](#)
Dec 2018 Discussion about the ConfiLog future functions between the tunisian and french teams.

Welcome to ConfiLog

Welcome to the ConfiLog project's website. The primary goal of ConfiLog is to *put Software Product Lines research into practice* through the guidance of practitioners in the field during the configuration process. With ConfiLog you can also edit, share and download product line models instantly.

ConfiLog is working on the following Internet browsers (Windows): **Safari, Mozilla, Google Chrome** and **Microsoft IE**. We value your [feedback](#) a lot!

Figure (18) : Capture d'écran d'accueil du Configurateur ConfiLog

ConfiLog permet de modéliser non seulement la variabilité hiérarchique telle que dans FODA mais aussi la variabilité orthogonale telle que dans OVM. Pour configurer un modèle, il faut, d'abord, commencer par saisir les différentes variables du problème dans l'onglet "Edit Product Line Models", ainsi que définir les contraintes du domaine. Lors de la modélisation de la ligne des sites web dans ConfiLog, les variables et les contraintes créées par la partie prenante sont formulées automatiquement en un problème SAT [Cai et al., 2011]. Un problème SAT est appelé aussi problème de satisfaisabilité booléenne qui, en fonction d'une formule de logique propositionnelle, détermine s'il existe une affectation des variables propositionnelles qui rend la formule vraie. Le modèle de la ligne de sites web construit par des entrées de la partie prenante est formulé comme suit:

```

/* Nom de la ligne de produits */
(1) <PLM_name="Website PL">
/* Métadonnées de la ligne de produits incluant une description, le nom de la partie prenante qui la
modélisé, son adresse, son email, son numéro de téléphone, son site Web et son organisation */
(2) <meta>
(3) <data name="description">An academic example of a product line websites</data>
(4) <data name="creator">Sam Legrand</data>
(5) <data name="address"></data>
(6) <data name="email">Sam.legrand@etu.univ-paris1.fr</data>
(7) <data name="phone">0767921291</data>
(8) <data name="website"> https://www.pantheonsorbonne/</data>
(9) <data name="organization">Université Paris 1 Panthéon Sorbonne</data>
(10) </meta>
/* Variantes du modèle de la ligne de produits [Type NomVraiante(Identifiant) Cardinalité]
Enum Type (:r = racine, :m = Obligatoire, :g = regroupée avec cardinalité, :o = optionnelle) */
(11) <variants>
(12) :r SiteWeb(_r)
(13)     :m Performance(_r_1)
(14)         :g (_r_1_4) [1,1]
(15)             : Minutes(_r_1_4_5)
(16)             : Millisecondes(_r_1_4_6)
(17)             : Secondes(_r_1_4_7)
(18)     :o ServicesAdditionnels(_r_2)
(19)         :o Recherche(_r_2_9)
(20)         :o Statistiques(_r_2_10)
(21)         :m Rapports(_r_2_12)
(22)         :m Bannières(_r_2_13)
(23)             :m Images(_r_2_13_14)
(24)             :o HTML5(_r_2_13_15)
(25)     :m ServeurWeb(_r_3)
(26)         :m Contenu(_r_3_16)
(27)             :m Statique(_r_3_16_17)
(28)             :o Dynamique(_r_3_16_18)
(29)                 :g (_r_3_16_18_19) [1,*]
(30)                     : ASP(_r_3_16_18_19_20)
(31)                     : PHP(_r_3_16_18_19_21)
(32)                     : Thymeleaf(_r_3_16_18_19_22)
(33)     :o Protocoles(_r_3_23)
(34)         :g (_r_3_23_24) [1,*]
(35)             : HTTPS(_r_3_23_24_25)
(36)             : FTP(_r_3_23_24_26)
(37)     :o Traçage(_r_3_27)
(38)         :g (_r_3_27_28) [1,1]
(39)             : BaseDonnées(_r_3_27_28_29)
(40)             : Fichier(_r_3_27_28_31)
(41) </variants>

```

```

/* Contraintes du domaine de la ligne de produits */
(42) <constraints>
/* Variante 1 Exige Variante 2 = [~IdentifiantVariante1 or IdentifiantVariante2]
Variante 1 Exclut Variante 2 = [~IdentifiantVariante1 or ~IdentifiantVariante2] */
(43) constraint_1:~_r_3_16 or _r_3_23
(44) constraint_2:~_r_3_16_18 or _r_3_23_24_25
(45) constraint_3:_r_3_23_24_26 or ~_r_3_27_28_31
(46) constraint_4:~_r_1_4_6 or ~_r_3_23_24_25
(47) constraint_5:_r_1 or ~_r_2
(48) constraint_6:_r_1_4_7 or ~_r_2_10
(49) constraint_7:_r_1_4_7 or ~_r_2_9
(50) constraint_8:~_r_2_12 or _r_3_27_28_31
(51) </constraints>
(51) </PLM>


```

La première ligne correspond au nom du modèle de la ligne. Les lignes entre (2) et (10) incluses sont consacrées aux méta données en rapport avec la partie prenante qui modélise la ligne. Les lignes entre (11) et (41) incluses correspondent aux variantes de la ligne. Chaque variante est définie par un identifiant, un libellé et un type. Les types :g, :m et :p correspond respectivement aux variantes regroupées par des cardinalités “grouped”, obligatoires “mandatory” et optionnelles “optional”. L’arborescence du script reflète les points de variabilité et les variantes. Les lignes entre (42) et (51) incluses correspondent aux contraintes du domaine. Les contraintes d’inclusion entre deux variables sont définis avec l’écriture suivante `~identifiant_variable1 or identifiant_variable2`. Les contraintes d’exclusion entre deux variables sont définis avec l’écriture suivante `~identifiant_variable1 or ~identifiant_variable2`.

VI.2.1.2. Configureur de lignes de produits

La configuration de de la ligne modélisée est un processus de décision qui peut impliquer un enchaînement d’étapes conduisant à la configuration finale. Une fois l’onglet “Product Line Configuration” est initialisé, la liste de tous les modèles de lignes de produits répertoriés s’affiche. Comme illustré dans la Figure (19), la partie prenante sélectionne le modèle à configurer et choisit un but de guidage.

Une fois la partie prenante sélectionne un modèle de ligne de produits et un but de guidage, une interface contenant toute la variabilité du modèle s’initialise comme l’indique la figure (20). En fonction du but de guidage sélectionné, l’interface structure les variantes du modèle en séparant les points de variabilité et leur variantes respectives. Tout au long du processus de configuration, le gestionnaire de configuration confronte la spécification du modèle à l’ensemble des décisions prises par la partie prenante dans le but de contrôler la cohérence des choix.



Mining process LOGs to enhance product line CONFIGurations

Houssem CHEMINGUI, Ines GAM, Raul MAZO, Camille SALINESI, Henda BEN GHEZALA
 Université Panthéon Sorbonne Paris 1, France
 Université de la Manouba, Tunisie

Janvier 2019

Home
Product Line Model Editor
Product Line Configuration
Product Line Model Repository
Contact Us


Select a product line model from the table below and [Click Here](#)

Type of guidance: No guidance ▼

#	Name	Guidance	Variants	Creator	Creation Date
1.	<input checked="" type="radio"/> Eshop	Customization	10	HoussemCHEMINGUI	
2.	<input type="radio"/> Test		11	cf	
3.	<input type="radio"/> Bike		12	Mohamed Housseem	
4.	<input type="radio"/> Aircraft PL		13	Houss chemi	Jan 2019
5.	<input type="radio"/> Arcade Game PL Feature Model		61	Edson Oliveira Junior	Aug 2009
6.	<input type="radio"/> HousseemBikeV6		116	Housseem CHEMINGUI	

CENTRE DE RECHERCHE EN INFORMATIQUE, UNIVERSITE PANTHEON SORBONNE PARIS 1 / LABORATOIRE DE RECHERCHE RIADI, UNIVERSITE DE LA MANNOUBA

Figure (19) : Capture d'écran de la sélection du modèle de la ligne de produits et le but de guidage



Mining process LOGs to enhance product line CONFIgurations

Houssem CHEMINGUI, Ines GAM, Raul MAZO, Camille SALINESI, Henda BEN GHEZALA
 Université Panthéon Sorbonne Paris 1, France
 Université de la Manouba, Tunisie

Janvier 2019

Home
Product Line Model Editor
Product Line Configuration
Product Line Model Repository
Contact Us

| WebsitePL (25 variants)

Originator Name

- Website
 - Performance
 - ^ [1..1]
 - Minutes
 - Mellisecondes
 - Secondes
 - AdditionalServices
 - Searches
 - Statistics
 - Reports
 - Banners
 - Images
 - Flash
 - Web Server
 - Contents
 - Static
 - Dynamic
 - ^ [1..3]
 - ASP
 - PHP
 - JSP

Configuration Steps [reset]

20%

Step	Decision	#Decisions (cummulative)	#Propagations (at step)	#SAT checks (at step)	SAT time (at step)
1	Website	5 (20,0%)	4	8	13 ms

Done Export configuration: CSV file | XML

Auto-completion: [Less Features](#) | [More Features](#)

Figure (20) : Capture d'écran de la conduite du processus de configuration

VI.2.2. Côté serveur : La couche de configuration

La couche de configuration de l'architecture ConfiLog distingue trois modules à savoir:

- Le moteur d'interfaces,
- Le gestionnaire de la configuration et
- L'extracteur de traces.

VI.2.2.1. Moteur d'interfaces

Les décisions à prendre sur les variantes du modèle de la ligne de produits sont affichées dans une interface de configuration (cf. figure 20). Néanmoins, l'ordre de prise de ces décisions est recommandé en fonction de la configuration partielle et le but de guidage.

Le moteur de recommandation calcule pour chaque prise de décision quelle décision à suivre et quelle valeur à lui recommander. En fonction de ces paramètres, le moteur d'interfaces ordonne les étapes du processus de configuration et recommande de sélectionner ou refuser les variantes en question. La partie du code qui correspond au appel des paramètres de recommandation afin d'ordonner les étapes de configuration est illustré ci-dessous.

```
(1) let orderList = '<ol type="1">';
(2) FinalOrder.forEach(function(item){
/* item.activity = étape de Configuration && betterChoice = Accepter ou refuser */
(3) orderList += '<li style="color:gray;">' + item.activity + ' - <span style="color:' +
(4) (item.betterChoice == 'select' ? 'green' : 'red') + ';'>' + item.betterChoice + '</span></li>';
(5) });
(6) orderList += '</ol>';
(7) document.getElementById('recommendedOrder').innerHTML = orderList;
```

VI.2.2.2. Gestionnaire de configuration

Le configurateur transforme le problème de décision SAT en un ensemble de choix de configuration à effectuer. Au fur et à mesure que des décisions sont prises par la partie prenante, le gestionnaire de la configuration calcule la cohérence des choix, met à jour les paquets de configuration partielle et adapte les requêtes de guidage.

Le gestionnaire de configuration assure qu'aucune violation des contraintes du domaine se produise. Au cours du processus de la configuration, la partie prenante peut

sélectionner, refuser ou ignorer des variantes. La Figure (20) montre un tableau à droite présente un récapitulatif des choix effectués avec une barre de pourcentage représentant l'état d'avancement de la configuration par rapport au processus complet.

VI.2.2.2. *Extracteur de traces*

En parallèle de calcul de cohérence, des traces de processus de configuration sont capturées. Ces traces reflètent les comportements des parties prenantes, leur choix et leur manipulation de configuration, néanmoins elles nécessitent une transformation considérable pour être fouillées plus tard. De son côté, l'utilisateur de ConfiLog est capable d'exporter sa configuration, même si elle n'est pas finalisée. La partie du code qui correspond à la capture des traces de processus est illustrée ci-dessous.

```
/* Structure de la trace */
(1) <b><br/>ID, Model Name, Originator Name, Feature ID, Feature Name, Feature Type, Feature Value,
(2) Feature Decision Type, Feature Decision Step, TimeStamp, Recommendation Type</b>
(3) <textarea id="textbox"><#list features as feature>
/* Appel des attributs */
(4) ${id}, ${modelName}, ${originatorName}, ${feature.id}, ${feature.name}, ${feature.type},
(5) ${feature.value}, ${feature.decisionType}, ${feature.decisionStep}, ${feature.timeStamp},
(6) ${recommendationType}
(7) </#list></textarea>
/* Bouton export des traces pour les utilisateurs */
(8)<button id="create">SAVE TRACES TO FILE</button>
(9 <a download="${originatorName}.csv" id="downloadlink" style="display: none"><button
(10) onclick="fonctionreset()">DOWNLOAD TRACES</button></a>
(11)<br>
```

VI.2.3. Côté serveur : La couche de fouille

La couche de fouille distingue deux modules à savoir :

- Le chargeur (ETL) et
- Le fouilleur de processus.

VI.2.2.1. *Chargeur (ETL)*

Afin de capitaliser les traces de configuration, un processus extracto-chargeur (ETL) [Vassiliadis et al., 2002] est conduit afin de donner une vue consolidée des traces permettant une meilleure fouille. Il s'agit d'un processus permettant d'effectuer des synchronisations massives des traces de processus vers des journaux d'évènements. Ces

synchronisations permettent d'enregistrer les traces dans une structure permettant la reconstitution des modèles de processus, leurs filtrages et leurs interprétations.

La structure des journaux d'événements regroupe des informations en rapport avec le processus de configuration, les variantes du modèle de la ligne de produits, l'ordre des étapes et l'horodatage. D'un autre côté, la structure des journaux d'événements prend en compte les éléments nécessaires pour répondre aux questions de guidage par la fouille de processus. Le repository des traces présente des enregistrements des traces de la configuration qui sont déjà filtrés et transformés suivant un processus (ETL).

Durant le processus de configuration, l'outil ConfiLog capitalise les traces de la partie prenante conformément à la structure définie précédemment (cf. chapitre V). Seules les traces référant à des configurations cohérentes, par rapport aux contraintes du domaine, sont considérées. Ce filtrage se fait à travers des processus (ETL). Un exemple de trace de processus de configuration du modèle de la ligne de sites web est illustré ci-dessous.

```
/* Identifiant unique du processus de configuration */
(1) ID : b62377
/* Nom du modèle de la ligne sur lequel est opérée le processus de configuration */
(2) Model Name : WebsitePL
/* Nom de la partie prenante effectuant le processus de configuration */
(3) Originator Name : Emanuel Koch
/* A chaque étape de configuration, identifiant de la variante, nom de la variante, type de la
variante, valeur affectée à la variante, type de décision de la variante, ordre de configuration de
la variante, l'horodatage et le type de recommandation */
(4) Feature ID, Feature Name, Feature Type, Feature Value, Feature Decision Type, Feature Decision
Step, TimeStamp, Recommendation Type

(5) _r_1_2_3, Minutes, grouped, 1, manual, 2, 13/11/2019 15:44:49:756, NoRecommendation
(6) _r_1_2_4, Milliseconds, grouped, 0, propagated, 2, 13/11/2019 15:44:49:756, NoRecommendation
(7) _r_1_2_5, Secondes, grouped, 0, propagated, 2, 13/11/2019 15:44:49:756, NoRecommendation
(8) _r_6, ServicesAdditionnels, optional, 0, manual, 3, 13/11/2019 15:44:51:356, NoRecommendation
(9) _r_6_8, Recherche, optional, 0, propagated, 3, 13/11/2019 15:44:51:356, NoRecommendation
(10) _r_6_9, Statistiques, optional, 0, propagated, 3, 13/11/2019 15:44:51:356, NoRecommendation
(11) _r_6_10, Rapports, mandatory, 0, propagated, 3, 13/11/2019 15:44:51:356, NoRecommendation
(12) _r_6_11, Bannières, mandatory, 0, propagated, 3, 13/11/2019 15:44:51:356, NoRecommendation
(13) _r_6_11_12, Images, mandatory, 0, propagated, 3, 13/11/2019 15:44:51:356, NoRecommendation
(14) _r_6_11_13, HTML5, optional, 0, propagated, 3, 13/11/2019 15:44:51:356, NoRecommendation
(15) _r_7_14_24, Dynamique, optional, 1, manual, 4, 13/11/2019 15:44:54:20, NoRecommendation
(16) _r_7_14_24_25_26, ASP, grouped, 1, manual, 5, 13/11/2019 15:44:55:105, NoRecommendation
(17) _r_7_14_24_25_27, PHP, grouped, 1, manual, 6, 13/11/2019 15:44:56:60, NoRecommendation
(18) _r_7_14_24_25_28, Thymeleaf, grouped, 0, manual, 7, 13/11/2019 15:44:57:325, NoRecommendation
(19) _r_7_14_24_25_28, Thymeleaf, grouped, 1, manual, 7, 13/11/2019 15:45:0:994, NoRecommendation
```

```
(20) _r_7_14_24_25_28, Thymeleaf, grouped, 0, manual, 7, 13/11/2019 15:45:3:926, NoRecommendation
(21) _r_7_14_24_25_26, ASP, grouped, 0, manual, 5, 13/11/2019 15:45:7:264, NoRecommendation
(22) _r_7_15, Protocoles, optional, 1, manual, 8, 13/11/2019 15:45:11:186, NoRecommendation
(23) _r_7_15_20_21, Https, grouped, 0, manual, 9, 13/11/2019 15:45:13:567, NoRecommendation
(24) _r_7_15_20_22, Ftp, grouped, 1, propagated, 9, 13/11/2019 15:45:13:567, NoRecommendation
(25) _r_7_16, Traçage, optional, 0, manual, 10, 13/11/2019 15:45:16:220, NoRecommendation
(26) _r_7_16_17_18, BaseDonnées, grouped, 0, propagated, 10, 13/11/2019 15:45:16:220,
NoRecommendation
(27) _r_7_16_17_19, Fichier, grouped, 0, propagated, 10, 13/11/2019 15:45:16:220, NoRecommendation
```

La ligne (1) correspond à un identifiant unique affecté à chaque trace de processus de configuration, la ligne (2) correspond au nom du modèle de la ligne configurée. La ligne (3) correspond au nom de la partie prenant effectuant la configuration. Le reste des lignes hormis la ligne (4) présentent des valeurs aux attributs capturés pendant la configuration à savoir l'ordre, l'horodatage, le type de variantes, le type de décision etc. (cf. chapitre V). Dans la présente trace de processus la partie prenante est passée par 22 étapes de configuration. Chaque étape de configuration traite une variante particulière.

Les traces de processus des différentes parties prenantes sont capitalisées dans un repository de traces permettant d'alimenter le fouilleur de traces avec des événements de configuration à fouiller. Dans la méthode ConfiLog la fouille de processus est une stratégie qui se fait d'une manière régulière à chaque fois que le repository des traces est enrichi. Plus le repository est riche, hétérogène et couvrant une longue période, plus la prestation de la fouille est significative.

VI.2.2.2. Fouilleur de processus

Le fouilleur de processus encapsule trois activités, notamment, l'analyse de la performance et l'analyse de la fréquence des processus ainsi que le requêtage en fonction de la méthode (GQM).

Le fouilleur de processus fournit des réponses aux questions de guidage en interrogeant des modèles de processus reconstitués. Les traitements de fouille de processus dépendent des objectifs du guidage requis. Les interprétations des modèles de processus sont effectuées en fonction des indicateurs de guidage définis. Les réponses aux questions de guidage sont stockées dans des datasets de guidage qui présentent l'essence de l'algorithme de recommandation.

Il existe de nombreux outils académiques et commerciaux pour fouiller des processus métiers comme Celonis⁹, Process Street¹⁰, Prom¹¹ et Disco¹². Leur fonctionnement dépend principalement de la taille des journaux d'événements à fouiller, des traitements de fouille envisagés, des algorithmes à actionner et la nature des modèles à manipuler. Par exemple, pour la découverte des modèles de processus par l'algorithme "Fuzzy Miner", il est nécessaire de charger les journaux d'événements dans l'outil, de paramétrer des filtres sur ces journaux et de générer des modèles de processus.

ConfiLog intègre l'outil Disco pour la fouille des processus de configuration et l'évaluation des réponses de guidage via les métriques. Ce choix de réutilisation est motivé par les motifs suivants :

- Disco implémente déjà l'algorithme "Fuzzy Miner" pour l'analyse des modèles de processus métiers,
- Les modèles de processus générés par Disco sont interactifs. Par exemple, ils peuvent être interrogés pour mettre en valeur des vues différentes comme la performance ou bien la fréquence des tâches,
- Il est possible d'exporter les modèles de processus sous forme graphique (image ou PDF) ou scripturale (CSV ou XML),
- L'utilisation de Disco est relativement simple avec des interfaces ergonomiques distinguant les différentes analyses requises et
- Étant chercheurs de l'université Paris 1, la structure Fluxicon¹³ a fourni pour ce travail de recherche une licence académique pouvant supporter jusqu'à 5 millions d'événements de traces.

Une partie du modèle de processus reconstitué à partir des journaux d'événements de processus de configuration de la ligne de sites web est illustrée dans la Figure (21). Cette partie montre uniquement les activités les plus fréquentes dans le repository de traces. Les nœuds se réfèrent aux activités du processus et les arcs se réfèrent aux transitions entre les activités. Les chiffres montrent la fréquence d'une activité ou transition donnée dans le repository de traces. Comme, dans l'exemple en question, il y a eu 46 processus de configuration de sites web. Dans les 46 processus, la variante "Performance" a été sélectionnée. Cependant, les variantes "Search", "Statistics" et "FTP" ont été

⁹ Lien : <https://www.celonis.com/>

¹⁰ Lien : <https://www.process.st/product/>

¹¹ Lien : <http://www.promtools.org/doku.php>

¹² Lien : <https://www.fluxicon.com/>

¹³ Lien : <https://www.fluxicon.com/team/>

sélectionnées 60, 60 et 56 fois, respectivement. Cette récurrence peut être due à une boucle de choix contraignants dans laquelle le guidage s'impose.

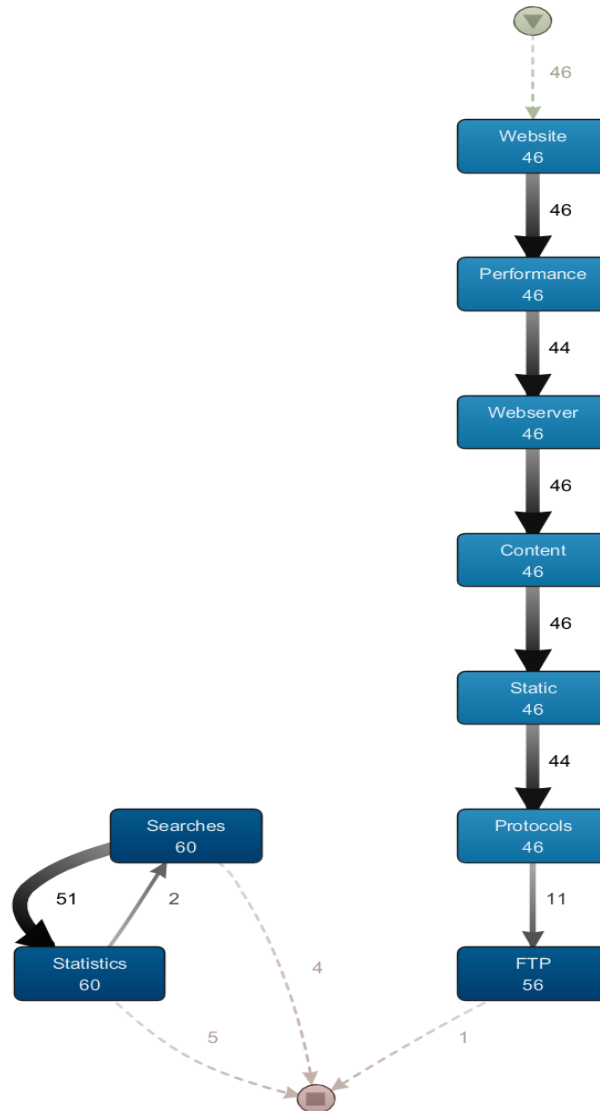


Figure (21) : Partie du modèle de processus de configuration des sites Web

Ce modèle de processus, pouvant être filtré pour répondre aux questions de guidage, est converti dans le script xml ci-dessous :

```

(1) <?xml version="1.0" encoding="UTF-8"?>
/* Modèle de processus de configuration */
(2) <ProcessMap numNodes="11">
/* 11 noeuds se référant à 11 activités de configuration */
(3)   <Nodes size="11">
/* Pour chaque activité de configuration : la fréquence, durée et répétition*/
  
```



```

(4)         <Node index="0" activity="_r-Website-templateModel">
(5)             <Frequency total="46" case="46" start="46" end="0" maxRepetitions="1"/>
(6)             <Duration total="1" min="0" max="1" mean="1" median="1"/>
(7)         </Node>
(8)         <Node index="1" activity="_r_1-Performance-mandatory">
(9)             <Frequency total="46" case="46" start="0" end="0" maxRepetitions="1"/>
(10)            <Duration total="0" min="0" max="0" mean="0" median="0"/>
(11)            ...
(12)     </Nodes>
/* 12 arcs se référant à 11 passages d'une activité de configuration à une autre */
(13)     <Edges size="12">
/* Pour chaque passage : la fréquence, durée et répétition*/
(14)         <Edge sourceIndex="0" targetIndex="1" type="observed">
(15)             <Frequency total="46" case="46" maxRepetitions="1"/>
(16)             <Duration total="0" min="0" max="0" mean="0" median="0"/>
(17)         </Edge>
(18)         <Edge sourceIndex="3" targetIndex="4" type="observed">
(19)             <Frequency total="46" case="46" maxRepetitions="1"/>
(20)             <Duration total="0" min="0" max="0" mean="0" median="0"/>
(21)         </Edge>
(22)         ...
(23)     </Edges>
(24) </ProcessMap>

```

Mis à part les filtres à appliquer pour interroger un modèle de processus, Disco permet de fournir des statistiques différentes, notamment :

- Le nombre d'événements dans le repository;
- Le nombre de processus exécutés;
- Le nombre moyen, minimal et maximal d'événements par processus; et
- La durée moyenne, minimale et maximale d'évènements par processus.

Ces statistiques étoffent généralement le modèle de processus pour répondre au mieux aux questions de guidage orchestrées par les buts. Pour chaque réponse apportée par la fouille, des filtres différents peuvent être appliqués sur le même modèle de processus. Par exemple, pour identifier les activités qui sont derrière les backtracks du système, ConfiLog applique un filtre permettant de repérer les évènements avec une fréquence supérieure à 1 dans le même processus.

Les modèles de processus contenant les réponses aux questions de guidage sont exportables sous forme d'un script à réutiliser par le recommandeur ConfiLog.

VI.2.4. Coté serveur : La couche de recommandation

La couche de recommandation distingue deux modules à savoir :

- Le moteur de recommandation et
- Le conducteur des recommandations produits.

VI.2.3.1. Moteur de recommandation

Le moteur de recommandation génère des recommandations. Pour ce faire, il prend en entrée des datasets de guidage calculés par le fouilleur de processus, effectue des calculs de pondération et d'impact de décisions et il produit un ordre à recommander avec des instances de valeur de variantes.

L'implémentation du moteur de recommandation ConfiLog (cf. Chapitre V) considère les fonctions listées ci-dessous pour chaque étape de configuration :

- La prise en compte du but de guidage;
- La prise en compte d'une configuration partielle;
- La pondération de la pertinence de chaque processus dans le modèle;
- Le calcul de l'impact de la (non) configuration d'une variante donnée; et
- La recommandation des valeurs de variantes, d'un côté, et l'ordre dynamique des étapes de configuration de l'autre côté.

Afin d'éviter une surcharge d'information et d'aider les développeurs à comprendre ConfiLog, ce manuscrit reprend, ci-dessous, les parties du code source qui correspondent au moteur de recommandation.

Pour effectuer la pondération des processus l'algorithme mesure l'intersection entre deux ensembles de variantes à travers la fonction `getInterSection(set1, set2)`.

```
/* Calcul de l'intersection entre deux ensembles de variantes */
(1) function getInterSection(set1, set2){
(2)     let ret = 0;
(3)     set1.forEach((item1) => {
(4)         set2.forEach((item2) => {
(5)             ret += (item1.optionName == item2.optionName &&
(6) item1.optionSelected == item2.optionSelected);
(7)         });
(8)     });
(9)     return ret;
(10) }
```

Afin d'identifier l'intersection entre la configuration partielle `Partial` et le dataset de guidage `file.data`, la fonction `getInterSection` est paramétrée comme suit `getInterSection(file.data, Partial)`. Les poids `weight` d'un processus du dataset de guidage correspond à la division de la valeur d'intersection par le nombre de variantes dans la configuration partielle.

```
(1) let weights = [];
(2)     filesData.forEach((file) => {
(3)
(4)         let interSection = getInterSection(file.data, Partial);
/* Division de l'intersection par la longueur de la configuration partielle */
```

```

(5)             let weight = interSection / Partial.length;
(6)             weights.push({
(7)               'fileName': file.fileName,
(8)               'weight': weight,
(9)               'guidance': file.guidance,
(10)            });
(11)          });

```

Il est à souligner que chaque processus inclus dans les dataset de guidage, dispose un indice de guidage `file.guidance`. L'indice de guidage reflète à quel point le processus en question répond à un but de guidage particulier. Pour les datasets de guidage se référant à des configurations flexibles et/ou personnalisées l'indice de guidage est mesuré à travers les métriques d'évaluation. Néanmoins, pour les datasets de guidage de configurations rapides, l'indice de guidage représente la durée du processus.

La valeur cible attendue après l'exécution d'une décision donnée `doValue` est calculée en multipliant les poids des processus supportant cette décision par leurs indices de guidage.

```

(1) filesData.forEach((file) => {
/* Calcul d'impact de la configuration d'une variante manuellement */
(2)   for(let item of file.data) {
(3)     if (item.optionMode == 'manual') {
(4)       for(let weight of weights) {
(5)         if (weight.fileName == file.fileName){
(6)           if (doValue == INF) doValue = 0;
/* Multiplication des poids par l'indice de guidage des datasets de guidage (cf. chapitre V) */
(7)           doValue += (weight.weight * file.guidance);
(8)         } } }
(9)       });

```

La valeur cible attendue après la non exécution d'une décision donnée `dontValue` est calculée en multipliant les poids de processus supportant d'autres décisions autorisées par leurs indices de guidage.

```

(1) filesData.forEach((file) => {
/* Calcul d'impact de la NON configuration d'une variante manuellement */
(2)   var found = false;
(3)   for(let item of file.data) {
(4)     if (item.optionMode == 'manual') {
(5)       found = true;
(6)     }
(7)   }
(8)   if (!found) {
(9)     for(let weight of weights) {
(10)      if (weight.fileName == file.fileName) {
(11)        if (dontValue == INF) dontValue = 0;
/* Multiplication des poids par l'indice de guidage des datasets de guidage (cf. chapitre V) */
(12)        dontValue += (weight.weight * file.guidance);
(13)      } } }
(14)     });

```

VI.2.3.2. Conducteur de recommandation

Le conducteur de recommandation permet d'injecter les recommandations dans le processus de configuration. Les recommandations sont conduites au moteur d'interface, qui est chargé, à son tour, de les présenter aux parties prenantes et que ce soit d'une manière ergonomique en cohérence avec leurs choix antérieurs.

La différence entre la `doValue` et la `dontValue` est calculée à travers la variable `valueABS`. Dans le cas de recommandation des configurations flexibles ou personnalisées, `guidanceType == "flexibility" || guidanceType == "customization"` les variantes impliquées dans les `valueABS` les plus petites sont recommandées en premier. Néanmoins, pour les recommandations des configurations rapides, les variantes impliquées dans les `valueABS` les plus importantes sont recommandées en premier.

```
/* Soustraction entre l'Impact de décision lors de la configuration (dovalue) et l'impact de
décision lors de la NON configuration (dontValue) */
(1) 'valueABS': Math.abs(doValue-dontValue),
(2) FinalOrder.sort((a, b) => {
/* Comparer la différence pour pour deux activités autorisées */
(3)   if (a.valueABS > b.valueABS) {
/* Si on cherche à MAXIMISER la flexibilité (flexibility) ou la personnalisation (customization) =>
on recommande l'activité ayant la différence (valueABS) la plus petite */
/* Si on cherche à MINIMISER le temps de réponse =>on recommande l'activité ayant la différence
(valueABS) la plus importante */
(4)       if (guidanceType == "flexibility" || guidanceType == "customization")
(5)           return 1;
(6)       else
(7)           return -1;
(8)   } else if (a.valueABS < b.valueABS) {
(9)       if (guidanceType == "flexibility" || guidanceType == "customization")
(10)          return -1;
(11)      else
(12)          return 1;
(13)    }
```

Afin de garantir que la configuration garde son esprit ludique et souple, ConfiLog propose qu'à tout moment la partie prenante peut revenir sur ses choix, les exporter ou modifier son but de guidage carrément. Le débogage de l'algorithme de recommandation pour un guidage de processus de configuration d'une ligne basique de sites web est repris ci-dessous.

Tout d'abord l'algorithme effectue une sélection des processus pertinents pour le guidage et calcule leur poids (degré de supporter la configuration partielle et le but de guidage configuration rapide).

Dans le cas où la partie prenante exprime vouloir une configuration rapide et sélectionne, ensuite, la variante `Paiement_TransfertBancaire`. L'algorithme interroge l'ensemble des processus enregistrés dans le dataset de guidage des processus rapides. Il s'avère que le processus numéro 1, en ligne (1), ne supporte pas la configuration partielle `weight: 0` cependant le reste des processus, en ligne (2) et (3) la supportent amplement `weight: 1`.

```
/* 3 datasets de guidage : chaque dataset a un nom (fileName), une pondération (weight) et un apport de guidage (Guidance) */
(1) {fileName: "10guidance.xml", weight: 0, guidance: "10"}
(2) {fileName: "13guidance.xml", weight: 1, guidance: "13"}
(3) {fileName: "15guidance.xml", weight: 1, guidance: "15"}
```

L'algorithme calcule la valeur `doValue`, la valeur `dontValue` et la valeur `valueABS` pour toutes les variantes autorisées soit `Paiement_CreditBancaire`, `Sécurité_Standard`, `Menu_Recherche`, `Traçage_Achat` et `Sécurité_Elevée`. Ensuite, il récupère pareillement la meilleure décision `betterChoice` soit de sélectionner ou de refuser la variante en question.

```
/* 5 activités autorisées : Chaque activité a un impact de décision lors de sa configuration (DoValue), un impact de décision lors de sa NON configuration (dontValue), l'intitulé de la variante à configurer (activity), une valeur de soustraction des deux impacts (valueABS) et une instance de décision à recommander pour sélectionner ou refuser la variante (betterChoice) */
(1) {doValue: 28, dontValue: 0, activity: "Sécurité_Elevée", valueABS: 28, betterChoice: (2)"select"}
(2)"select"}
(3) {doValue: 0, dontValue: 28, activity: "Sécurité_Standard", valueABS: 28, betterChoice: (4)"deselect"}
(4)"deselect"}
(5) {doValue: 28, dontValue: 0, activity: "Menu_Recherche", valueABS: 28, betterChoice: "select"}
(6) {doValue: 28, dontValue: 0, activity: "Traçage_Achat", valueABS: 28, betterChoice: "deselect"}
(7) {doValue: 15, dontValue: 13, activity: "Paiement_CreditBancaire", valueABS: 2, betterChoice: (8)"select"}
(8)"select"}
```

A chaque étape de configuration, le moteur de recommandation reçoit une requête de recommandation. La réponse à cette requête est conduite vers le moteur d'interfaces de la couche configuration. A son tour, le moteur d'interfaces reformule la réponse en une structure compréhensible par la partie prenante. Pour ce faire, il agence l'ordre des interfaces et favorise la sélection d'une variante recommandée. L'interface de recommandation détaillant le débogage de la pondération et des calculs d'impacts est reprise dans la figure (22). L'ordre de la liste des variantes est mis à jour de façon à mettre en premier la variante recommandée soit `Paiement_CreditBancaire`. La perspective à droite de la liste des variantes montre l'instance de décision recommandée soit `select`.

The screenshot displays the CONFILOG web application interface. The main content area shows 'Eshop (10 variants)' with a tree view of configuration options: Security (High, Standard), Search, Logging, Payment (BankTransfer, CreditBank), and Catalogue. A 'Guidance type: Performance' section provides a 'recommended order' list: 1. High - select, 2. Standard - deselect, 3. Search - select, 4. Logging - deselect, 5. CreditBank - select. Below this is a 'Configuration Steps' table showing the progress of decisions.

Step	Decision	#Decisions (cumulative)	#Propagation (at step)
1	✓ Eshop	4 (40,0%)	3
2	✓ BankTransfer	5 (50,0%)	0

The browser's developer tools are open, showing the DOM tree and console logs. The console displays a list of configuration steps with their respective values and activities.

```

final order:
  SplotConfigurationSe...ype=performance:390
  SplotConfigurationSe...ype=performance:391
  (5) [{"doValue": 0, "donotValue": 28, "activity": "High", "valueABS: 28, betterChoise: "select"}
  1: {"doValue": 0, "donotValue": 28, "activity": "Standard", "valueABS: 28, betterChoise: "desele...
  2: {"doValue": 0, "donotValue": 28, "activity": "Search", "valueABS: 28, betterChoise: "select"}
  3: {"doValue": 0, "donotValue": 28, "activity": "Logging", "valueABS: 28, betterChoise: "deselec...
  4: {"doValue": 13, "donotValue": 15, "activity": "CreditBank", "valueABS: 2, betterChoise: "sele...
  length: 5
  __proto__: Array(0)
    
```

Figure (22) : Interface de recommandation et débogage de la pondération des datasets suivie des calculs d'impact de décisions

VI.6. Conclusion

La méthode ConfiLog a été implémentée sous forme d'une application Internet riche (RIA) vu que ce type d'application est adapté à une expérience utilisateur engendrant des calculs lourds. La conduite des décisions de configuration cohérentes implique une grande charge pour l'application, particulièrement dans les larges modèles de lignes de produits contenant une pléthore de variantes. Par ailleurs, plus le modèle de ligne de produits est large, plus il laisse trace à des événements qui reflètent les étapes du processus de configuration. Idem, la fouille de processus devient coûteuse en termes de ressources. Pour ce faire, une architecture logicielle client et serveur a été adoptée. Face aux différentes considérations, trois couches de côté serveur réagissent afin de produire le guidage, notamment : *la couche configuration*, *la couche fouille* et *la couche recommandation*. Le choix d'une architecture en couches garantit la maintenabilité et l'évolutivité, la performance, la modularité de développement et la testabilité.

La couche de configuration de la méthode ConfiLog distingue trois modules à savoir le moteur d'interfaces, le gestionnaire de la configuration et l'extracteur de traces. Quant à la couche de fouille, elle distingue deux modules à savoir le chargeur (ETL) et le fouilleur de processus. Au final, la couche de recommandation distingue un moteur de recommandation et un conducteur des recommandations produits.

Le moteur de recommandation mesure l'intersection entre deux ensembles de variantes afin de calculer la pondération des processus en fonction d'une trace partielle. Par ailleurs, il calcule l'impact d'une décision en respect avec le but de guidage exprimé par la partie prenante. Le conducteur de guidage fournit au moteur d'interfaces des éléments de recommandation sur deux niveaux : un ordre des étapes du processus de configuration et des valeurs de variantes. Le moteur d'interface les adapte en agençant dynamiquement les étapes de configuration et en favorisant les valeurs de variantes recommandées. A chaque étape de configuration, la recommandation est recalculée.

Le chapitre suivant présente une évaluation de la méthode ConfiLog.

[Cette page est intentionnellement laissée en blanc]

Chapitre VII : Evaluation

VII.1. Introduction	121
VII.2. Paramétrage de l'étude empirique avec les étudiants	121
VII.2.1. Planification	122
<i>VII.2.1.1. Groupes d'étudiants participants</i>	<i>122</i>
<i>VII.2.1.2. Motifs de la configuration sans guidage (P1)</i>	<i>122</i>
<i>VII.2.1.3. Motifs de la configuration avec guidage (P2)</i>	<i>123</i>
<i>VII.2.1.4. Hypothèse et questions de recherche</i>	<i>123</i>
<i>VII.2.1.3. Protocole d'expérimentation</i>	<i>124</i>
VII.2.2. Environnement	126
VII.2.3. Questionnaires	126
VII.2.4. Etude de cas : Ligne de vélos	127
VII.3. Résultats de l'expérimentation	130
VII.3.1. Background et attentes des étudiants	130
VII.3.2. Comparaison analytique des deux processus de l'expérimentation	131
VII.3.3. Retour d'expérience des étudiants	134
VII.3.4. Temps de réponse	135
VII.3.5. Passage à l'échelle	137
VII.4. Menaces à la validité	138
VII.5. Conclusion	140

Chapitre VII

Evaluation

VII.1. Introduction

L'évaluation est un élément indispensable pour continuer à s'aligner avec la logique de Design Science Research, suivie dans le présent manuscrit. L'évaluation de la méthode ConfiLog repose sur son outillage développé (cf. Chapitre VI). Étant générique, ConfiLog est applicable pour le guidage de processus de configuration de toute ligne de produits. En effet, une preuve de concept a été conduite par le biais d'un exemple basique d'une ligne de sites web (cf. Chapitre V), en outre, un débogage du moteur de recommandation lors d'un processus de configuration d'une ligne de sites Web a été détaillé (cf. Chapitre VI).

Ce chapitre détaille une expérimentation menée avec des étudiants in vivo. L'utilisation des études empiriques avec des étudiants (ESWS : Empirical Studies With Students) aide les chercheurs à avoir un aperçu sur les nouvelles ou anciennes techniques et les méthodes proposées [Carver et al., 2010]. En effet, les (ESWS) sont utiles aux industriels et aux communautés de chercheurs vu qu'elles sont conduites de manière adéquate et qu'elles ne surestiment pas, particulièrement, les résultats.

L'expérimentation confronte la configuration avec guidage de ConfiLog à la configuration sans guidage. Par ailleurs, ce chapitre détaille les résultats de l'expérimentation en termes de qualité et de performance de guidage. A bon escient, des menaces à la validité de l'expérimentation sont soulignées.

VII.2. Paramétrage de l'étude empirique avec les étudiants

Pour l'étude empirique conduite avec les étudiants, la présente section détaille la planification de l'expérimentation, l'environnement de la configuration, les questionnaires pré et post-expérimentation ainsi que la ligne de vélos à configurer.

VII.2.1. Planification

Les (ESWS) étant souvent utilisées pour obtenir des preuves confirmant ou infirmant une hypothèse de recherche, cette expérimentation revêt deux **Processus** à savoir :

- **(P1)** : La collecte et l'analyse (i) des attentes, (ii) des traces de configuration d'une ligne de vélos et (iii) des retours d'expérience *des étudiants n'ayant pas utilisé une méthode particulière pour le guidage du processus de configuration.*
- **(P2)** : La collecte et l'analyse (i) des attentes, (ii) des traces de configuration d'une ligne de vélos et (iii) des retours d'expérience *des étudiants ayant utilisé la méthode ConfiLog pour le guidage du processus de configuration.*

VII.2.1.1. Groupes d'étudiants participants

Pour parvenir à exécuter les deux processus sans guidage (P1) et avec guidage (P2), la présente expérimentation a fait recours à des étudiants de deuxième année licence miashs à l'université Panthéon Sorbonne Paris 1 et des étudiants de la première année licence appliquée en informatique à l'ISET Nabeul. Par ailleurs, ces étudiants ont été répartis en groupes de binôme. Tous les binômes, ayant contribué à cette expérimentation, ont participé à (P1) et également à (P2). Chaque binôme représente un étudiant "Baseline" et un étudiant "Connaisseur".

- **Étudiant "Baseline"** : Cet étudiant configure le vélo qui va avec ses préférences sans se servir de méthode de guidage particulière (P1), en un premier temps, et en se servant du guidage de ConfiLog (P2), en un deuxième lieu.
- **Étudiant "Connaisseur"** : Cet étudiant explique à son collègue les choix particulièrement techniques (par exemple l'unité inch, les différents types de guidons, la différence entre aluminium et carbone, etc...) lors de (P1) et également lors de (P2).

L'identification des étudiants "Baseline" et des étudiants "Connaisseurs" a été faite au moyen d'un pré-questionnaire (cf. Annexe B). La répartition des étudiants en binôme vise à faire travailler les étudiants ensemble sur une configuration avec des tâches complémentaires détaillées dans le protocole d'expérimentation ci-dessous.

VII.2.1.2. Motifs de la configuration sans guidage (P1)

Les traces de configuration capitalisées lors de (P1) sont censées prouver que le processus de configuration souffre vraiment des lacunes et que les étudiants étaient

confrontés à des gênes au cas où aucun guidage n'est servi. Ces lacunes rejoignent les problèmes du processus de configuration identifiées en état de l'art (cf. chapitre III) à savoir la complexité de la décision, le manque de performance et le besoin de guidage.

Outre que l'analyse des rendus de la configuration sans guidage (attentes, traces et retours d'expérience), les principales intentions derrière (P1) sont les suivantes:

- Alimenter la base de journaux d'événements de ConfiLog (cf. chapitre V) avec des traces de processus de configuration servant de fouille et de calcul de recommandation.
- Pallier au problème de démarrage à froid de la recommandation.

La problème de démarrage à froid [Schein et al., 2002] est inévitable vu que le moteur de recommandation de ConfiLog ne peut pas tirer d'inférence pour des décisions de configuration sur lesquelles il n'a pas encore recueilli des connaissances (tirés de la fouille des traces de configurations antérieures).

VII.2.1.3. Motifs de la configuration avec guidage (P2)

La configuration avec guidage (P2) invite les étudiants à configurer la même ligne de vélos via le configurateur de ConfiLog mais cette fois-ci avec le guidage produit à partir des traces de processus de (P1).

Outre que l'analyse des rendus de la configuration avec guidage (attentes, traces et retours d'expérience), les principales intentions derrière (P2) sont les suivantes:

- Mesurer le niveau de satisfaction des étudiants vis à vis du guidage fourni et
- Évaluer les qualités, les défaillances et les carences du guidage fourni.

VII.2.1.4. Hypothèse et questions de recherche

Les résultats finaux confrontent l'hypothèse si le configurateur a vraiment réussi à orchestrer un processus de configuration adéquat avec une valeur ajoutée de guidage. Pour cela les deux questions de recherche ci-dessous EQR1 et EQR2 ont été soulevées selon la structure suivante [Booth et al. 2003] : *“Je suis en train d'étudier <le sujet x> parce que <cause x> afin de <motivation x>”*.

- *EQR1. Je suis en train d'étudier les traces des étudiants tout au long de la configuration d'une ligne de vélos, parce que j'essaie de montrer comment le processus de configuration est sujet à*

l'erreur,

afin de mieux comprendre l'utilité de guider la configuration.

- *EQR2. Je suis en train d'étudier les traces des étudiants tout au long de la configuration d'une ligne de vélos, parce que j'essaye de montrer qu'il est possible de fouiller et d'analyser les insuffisances de la configuration sans guidage, afin de mieux comprendre l'utilité de la fouille de processus de configuration.*

VII.2.1.3. Protocole d'expérimentation

La Figure (23) décrit le protocole d'expérimentation et les phases conduites durant la présente expérimentation. Les superviseurs avaient à exécuter les tâches suivantes :

1. La mise en contexte : Expliquer aux différents étudiants le déroulement de l'expérimentation,
2. La création des groupes des étudiants en fonction des rendus de pré-questionnaire,
3. La collecte des traces de configuration de vélos,
4. L'application de la méthode ConfiLog uniquement pour la configuration avec guidage (P2) et
5. L'évaluation des rendus de la configuration avec guidage (P1) et la configuration sans guidage (P2). Les rendus des configurations présentent les attentes des étudiants, leurs traces de configurations et leurs retours d'expérience.

Cependant, les étudiants étaient invités à effectuer les tâches suivantes :

1. La réponse au pré-questionnaire permettant d'identifier leur savoir par rapport aux vélos et d'éclaircir leur attentes. En fonction des rendus de pré-questionnaire les superviseurs de l'expérimentation ont formé les groupes en binôme à savoir : Un étudiant "Baseline" et un étudiant "Connaisseur".
2. La configuration en binôme d'un vélo. L'étudiant "Baseline" prend des décisions de configuration qui sont surveillées et encadrées par l'étudiant "Connaisseur".
3. La réponse au post-questionnaire une fois que la configuration est finalisée. Les rendus des questionnaires et les actions des étudiants tout au long de la configuration du vélo sont capitalisés pour évaluer la méthode ConfiLog plus tard.

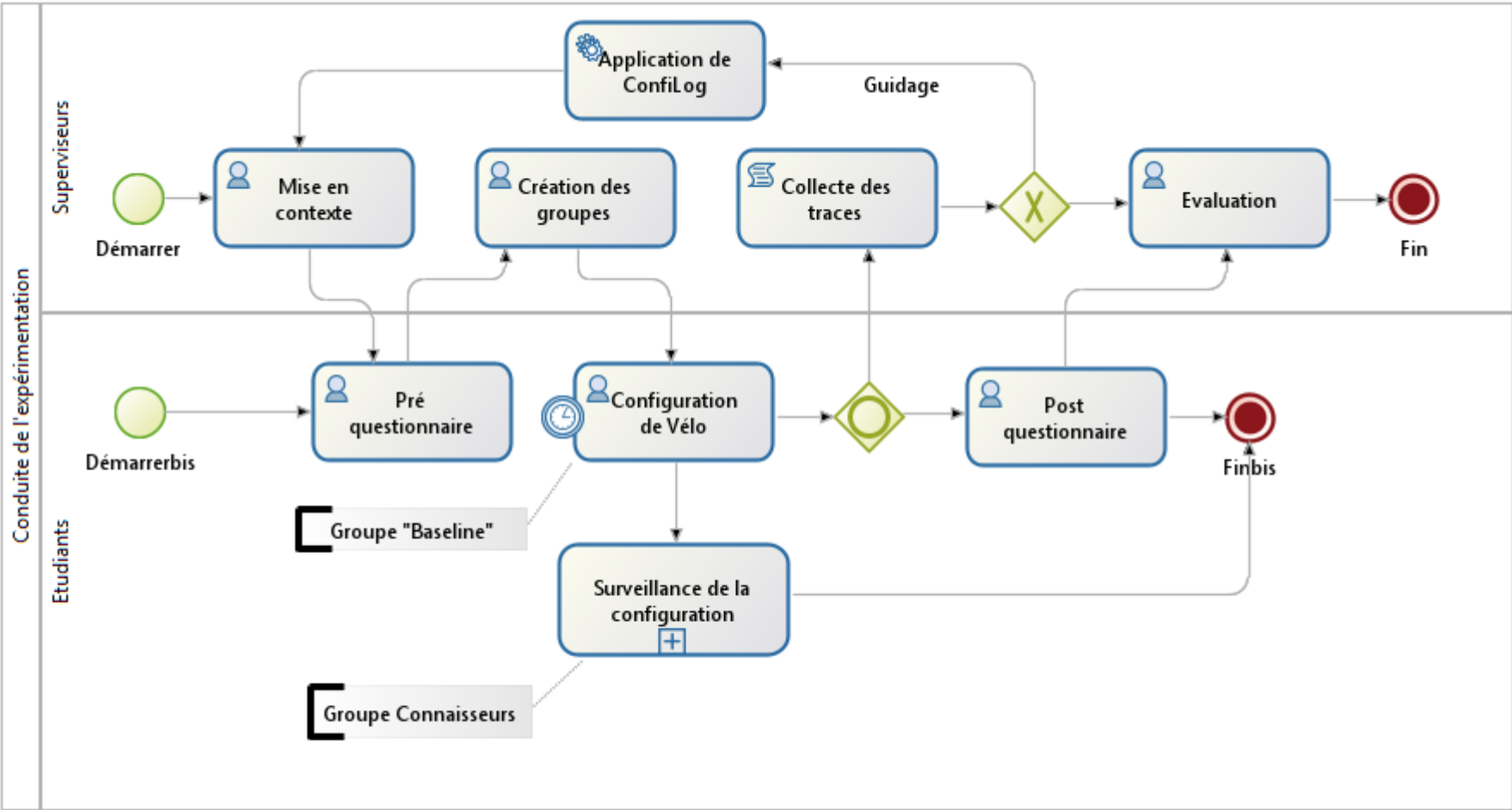


Figure (23) : Protocole d'expérimentation de l'étude empirique avec les étudiants

VII.2.2. Environnement

L'expérience a été menée dans un environnement familier pour les étudiants, dans le but de minimiser les facteurs environnementaux externes susceptibles de menacer la validité des résultats. En effet, cette expérimentation s'est déroulée en deux étapes: une étape en présentiel dans les salles de cours et une partie en ligne via des outils de collaboration à distance.

Pour imposer le travail en groupe, les étudiants du même binôme ont été regroupés afin de collaborer ensemble. Par ailleurs, les étudiants étaient appelés à limiter leur accès à Internet durant l'expérimentation afin de minimiser les distractions (messagerie instantanée, courrier électronique, etc.) tout en gardant accès aux ressources nécessaires pour mener à bien l'expérience. Ces ressources comprennent le configurateur (cf. Chapitre VI) et une documentation.

Les étudiants étaient avertis qu'ils doivent se sentir libres de revenir sur leurs choix de configuration déjà effectués, cependant, ils ne savaient pas la finalité de la présente expérimentation.

VII.2.3. Questionnaires

La première phase de l'expérimentation, pour les étudiants, consiste à répondre à un pré-questionnaire. Les questionnaires ont été conçus à l'aide d'une échelle de Likert [Likert, 1932]. Cette méthode de mise à l'échelle psychométrique contient un ensemble d'éléments à évaluer, mesurant le niveau d'accord de l'étudiant par rapport à une déclaration donnée. En effet, les éléments de Likert considèrent un format en cinq points : (1) totalement en désaccord, (2) plutôt en désaccord, (3) ni en accord ni en désaccord, (4) peu d'accord, et (5) fortement d'accord.

Le pré-questionnaire est utilisé pour déterminer le background et le profil général des étudiants afin de cerner les éventuelles différences entre eux concernant leurs compétences de base.¹ Cela a permis de confirmer leurs connaissances de la configuration et leur familiarisation avec la structure d'un vélo et ses accessoires. Tous les étudiants ont répondu au pré-questionnaire, dont les réponses sont analysées pour la création des groupes et l'évaluation du guidage plus tard.

A la fin de la configuration, les étudiants étaient invités à répondre à un post-questionnaire mesurant le niveau de satisfaction par rapport à la configuration

¹ Lien : <https://docs.google.com/forms/d/e/1FAIpQLSdQBS2Q5rJuBa5Nt--i6rqKbJbN-d7rUOzhXjJ6APkx0jA/viewform>

menée². Le post-questionnaire (cf. Annexe C) adopte pareillement l'échelle de Likert pour évaluer l'expérience des étudiants qui ont utilisé le configurateur de vélo.

VII.2.4. Etude de cas : Ligne de vélos

Les étudiants "Baseline" étaient invités à configurer, chacun, un vélo selon les instructions de la mise en contexte. Le modèle de vélos à configurer est détaillé dans la Figure (24) ci-dessous.

La spécification de la ligne impose que l'étudiant fasse des choix aux variantes associées à des points de variabilité qui peuvent avoir des cardinalités différentes. Par exemple, pour le point de variabilité cadre du vélo `Frame Design` de cardinalité `[1..1]`, l'étudiant est invité à sélectionner une seule variante parmi plusieurs, notamment, le cadre en diamant `Diamond`, le cadre en trapèze `Step-Through`, ou le cadre `Triathlon`. Encore, pour le point de variabilité guidon `Handlebar`, l'étudiant est invité à sélectionner une seule variante parmi plusieurs, notamment, le guidon route `Drop`, courbé `Pursuit` ou plat `Riser`. Ces choix sont exclusifs dans le sens où une seule variante est permise. Cependant, pour d'autres points de variabilité de cardinalité `[0..*]`, l'étudiant a la possibilité de sélectionner aucune variante ou plusieurs. Par exemple pour le point de variabilité accessoires `Accessories` l'étudiant peut choisir ou non des gardes-boue `Mudguards`, un casque `Kask Road`, une pompe `Air Pump`, une housse de vélo `Bike cover`, une gourde `Water bottle`, un grand panier `Big panier` ou un petit panier `Small panier`. En effet, le modèle de la ligne de vélos, distingue alors des points de variabilité qui doivent obligatoirement être configurés et des points de variabilité qui sont optionnels. Les points de variabilité obligatoires sont dans des triangles de ligne continue, comme la fourche du vélo `Fork`, par contre, les points de variabilité optionnelle sont dans des triangles de ligne pointillée comme les préventions du vol `Theft Preventions`.

Enfin, les variantes du modèle de la ligne de vélos sont interdépendantes en fonction des contraintes du domaine. Par exemple, un étudiant qui a un `Budget` moins que 500 unités de monnaie `Less 500` lui prive d'un vélo avec des accessoires. Cette contrainte est spécifiée via des contraintes de type `Excludes`. Outre cela, un étudiant qui a choisi un vélo avec lumière `Light` doit obligatoirement choisir une source d'énergie `Energy Source` qu'elle peut être dynamo électrique `Dynamo`, une batterie `Battery` ou encore un panneau solaire `solar panel`. Cette contrainte est spécifiée via des contraintes de type `Requires`.

² Lien:

https://docs.google.com/forms/d/e/1FAIpQLSeobHu9Ba_geeinieSGvcSynYDZPtZfUPSVmyRRhe4CGQdo9Q/viewform

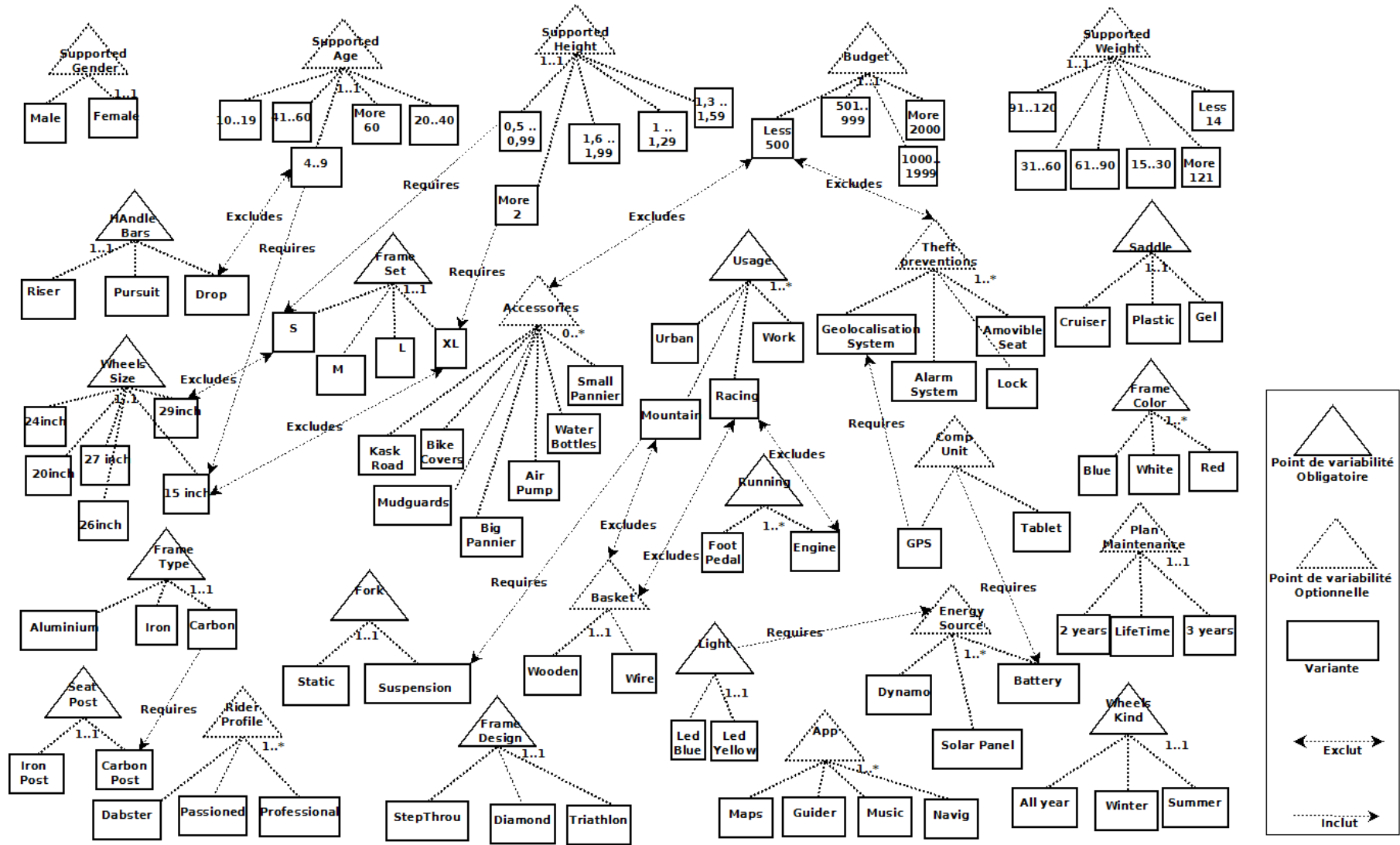


Figure (24) : Ligne de vélos en notation OVM

La spécification complète du vélo et sa modélisation dans le configurateur ConfiLog est détaillée dans les annexes (cf. Annexe D).

Le modèle de la ligne de vélos est conçu avec la notation OVM [Pohl et al., 2005]. En effet, la ligne compte 116 variables dont 17 variables obligatoires, 14 variables qui sont optionnelles et 85 variables qui sont groupées avec des cardinalités différentes. En plus, 16 contraintes de domaine sont spécifiées dont 8 contraintes d'exclusion et 8 contraintes d'inclusion. Le nombre de variantes à configurer n'est pas dévoilé aux étudiants mais il est égal au nombre des variantes optionnelles et groupées en cardinalité soit 99. Les étudiants étaient appelés uniquement à construire progressivement une configuration d'un vélo, qui va avec leur préférence, en un ensemble d'itérations.

La multiplication des solutions possibles pour chaque point de variabilité est équivalente à l'ensemble des solutions possibles à partir du modèle.

Le nombre de combinaisons possibles c des variantes n appartenant à un point de variabilité donné PV , est donné par la relation ci-dessous. Cependant, seules des variantes discernables ont été considérées. Par ailleurs, la répétition des variantes sélectionnées, leurs permutations et leur ordre de sélection ne sont pas pris en compte.

$$\prod_{PVn}^{PV1} \sum_{Min}^{Max} C_n^p = \frac{n!}{p!(n-p)!}$$

Avec :

- PV1 : Premier point de variabilité
- PVn : Dernier point de variabilité
- Min : Cardinalité minimale
- Max : Cardinalité maximale
- C : Combinaisons possibles
- p : nombre des variantes à sélectionner
- n : Nombre des variantes du PV

En effet, le modèle de la ligne de vélos mène à plus d'un milliard de configurations possibles sauf que celles-ci ne sont pas toutes valides d'où le besoin de guidage.

Il est à rappeler que la configuration sans guidage ne fait pas des propagations automatiques de choix liés à une contrainte donnée au fur et à mesure qu'une décision est prise. Ceci revient à configurer la ligne entière et vérifier la cohérence de la configuration à posteriori. Néanmoins, effectuer des configurations valides dans un espace de solutions qui mène à plus d'un milliard de configurations possibles semble

absurde. Tout bien considéré, il a été décidé d'aider les étudiants qui ont participé à l'expérimentation en leur proposant des configurations basées sur la connaissance. En effet, le configurateur ConfiLog propageait automatiquement les choix des étudiants au cas où une contrainte d'exclusion ou d'inclusion est envisagée. En effet, la propagation des choix réduit énormément l'espace de solutions et permet d'effectuer des choix cohérents permettant la production du guidage ultérieurement.

VII.3. Résultats de l'expérimentation

A partir des traces collectées dans (P1) et des fouilles opérées par la méthode ConfiLog et son moteur de recommandation, du guidage a été produit pour guider les étudiants dans (i) l'enchaînement du processus de configuration, (ii) l'affectation des valeurs aux variables du modèle en fonction de leurs buts respectifs.

Cette section détaille les backgrounds des étudiants et leurs attentes vis-à-vis de la configuration d'un vélo. Ensuite, les résultats des deux processus de l'expérimentation sont comparés et analysés afin de souligner la valeur ajoutée du guidage de la méthode ConfiLog. Enfin, les retours d'expérience des étudiants sont évalués ainsi que des mesures de temps de réponse et de passage à l'échelle.

VII.3.1. Background et attentes des étudiants

Les étudiants, qui ont participé à la présente expérimentation, ont été interrogés sur leurs connaissances de vélos ainsi que sur leurs connaissances informatiques. Par ailleurs, des questions par rapport à leurs attentes (Annexe B bis) lui ont été posées.

Pour commencer, 58,3% des participants étaient des étudiants contre 41,7% des étudiantes. Sexes confondus, les étudiants interrogés ont des connaissances hétérogènes par rapport au monde du cyclisme. En effet, 61,1% prétendent qu'ils connaissent bien les particularités, les choix inclusifs et exclusifs que peut imposer le vélo; par exemple, l'énergie est indispensable pour l'éclairage à vélo et que plusieurs sources d'énergie peuvent le faire telles que les batteries, les dynamos, les panneaux solaires, etc. Le reste des étudiants avouent ne pas connaître grand chose la dessus.

Des questions en rapport avec leurs facultés de manipuler des systèmes configurables lui ont été également posées. En effet, 27,3% ont exprimé leur familiarisation avec le

processus de configuration, 12,2% étaient neutres et 60,4% pensent qu'ils ont des connaissances minimales de comment ça se passe en réalité.

A leur tour, les attentes des étudiants par rapport à la faculté de la configuration étaient mitigées. En effet, 59,7% s'attendent à ce que la configuration d'un vélo soit facile contre 40,3% qui s'attendent à une configuration perplexe impliquant des difficultés. Cependant, les attentes des étudiants, en rapport avec (i) la rapidité de la configuration, (ii) la flexibilité des choix et (iii) leur assistance à prendre des décisions simples et cohérentes, étaient fermes et déterminantes. En effet, 62,5% des étudiants s'attendent à ce que le processus de configuration du vélo ne dépasse pas les 10 minutes. 93,5%, de surcroît, s'attendent à pouvoir prendre des décisions et les modifier plus tard. En outre, 51,1% pensent que le configurateur va assister leurs décisions en vue de ne pas se perdre. Néanmoins, 77% des étudiants affirment avoir déjà une idée sur le vélo souhaité.

Les attentes des étudiants soutiennent l'hypothèse de besoin de guidage assumé par la méthode ConfiLog, par ailleurs, leurs backgrounds ont été exploités pour la formation des groupes.

VII.3.2. Comparaison analytique des deux processus de l'expérimentation

En tout, 66 binômes d'étudiants ont participé à la configuration sans guidage de ConfiLog (P1) et 113 binômes ont participé à la configuration avec guidage de ConfiLog (P2). Le Tableau (13) confronte les résultats analytiques des deux processus de l'expérimentation (P1) et (P2).

Tableau (13) : Résultats analytiques de la configuration avec et sans guidage

	(P1) Sans Guidage ³	(P2) Avec Guidage ⁴			
		But 1: Rapide	But 2 : Flexible	But 3 : Personnalisé	Tous buts confondus
Nombre de configurations	66	81	21	11	113
Évènements produits	19118	7946	2087	1094	11107
Configuration correctes	45%	99%	96%	98%	97%
Durée Minimale	1 min et 18 sec	14 sec	14 sec	14 sec	14 sec
Durée moyenne	21,9 min	107,1	5,4 min	4,1 min	2,7 min

³ Lien : <https://docs.google.com/spreadsheets/d/1Zp3Sy1eZfa9gemq4sw2B8BzQ2Fqp2gdp3ae5AISPwvg/edit?usp=sharing>

⁴ Lien : <https://docs.google.com/spreadsheets/d/1BiZcQPz7usYe0mFJqv4O3btDMkfMgIeGdOvAokrcdDU/edit?usp=sharing>

		sec			
Durée maximale	1 h et 27 min	9 min et 36 sec	9 min et 36 sec	9 min et 36 sec	9 min et 36 sec
Durée médiane	255 sec	95,6 sec	5,5 min	4,6 minutes	115 sec
Décisions différentes	251	220	210	184	246
Nombre minimal des évènements par processus	12	20	20	20	20
Nombre requis des évènements par processus	99	99	99	99	99
Nombre maximal des évènements par processus	971	104	104	104	104
Fréquence moyenne d'une décision	76,17	36,12	9,94	5,95	45,15
Fréquence médiane d'une décision	66	28	9	6	32
Variables regroupées	85,42%	85,87%	85,91%	85,74%	85,86%
Variantes optionnelles	14,58%	14,13%	14,09%	14,26%	14,14%
Variantes refusées	74,12%	73,55%	61,09%	69,84%	70%
Variantes retenues	25,88%	26,45%	38,91%	30,16%	30%
Décisions propagées	69,63%	73,41%	64,93%	72,39%	71,7%
Décisions manuelles	30,37%	26,59%	35,07%	27,61%	28,3%

En examinant les chiffres du Tableau (13), il s'avère que dans la configuration non guidée (P1), pour configurer 66 vélos, il a fallu 19118 évènements pour aboutir uniquement à 36 configurations de vélos complètes. Cependant, dans la configuration guidée (P2), pour configurer 113 vélos, il a fallu 11107 évènements pour aboutir à 110 des configurations complètes. Avec le guidage fourni aux étudiants, la durée moyenne d'un processus de configuration est réduite de 81% soit 2,7 minutes au lieu de 21,9 minutes. Également, la durée médiane d'un processus de configuration est passé de 255 secondes à 115 secondes soit une réduction de 45%.

Par ailleurs, il s'avère que le guidage encadre les décisions à effectuer. En effet, 251 décisions différentes ont eu lieu pour produire 66 configuration soit une moyenne de 3,80 décisions qui n'ont jamais été prises au processus d'avant. Par contre, 246 décisions différentes ont eu lieu pour produire 113 configurations soit une moyenne de 2,19 décisions qui n'ont jamais été prises au processus d'avant. Le nombre minimal des

événements permet d'identifier les configurations complètes et les allers retours. Le nombre maximal des allers retours, provoqués suite à des décisions manuelles, dans la configuration guidée est de 5, cependant, dans la configuration non guidée il a eu 872 allers retours.

Dans les configurations sans guidage, la même décision revient en moyenne 76,17 dans l'ensemble du journal des événements. Cependant, dans les configurations guidées; la même décision revient en moyenne 45,15 des décisions dans l'ensemble du journal des événements. La baisse de ce pourcentage reflète que le guidage fourni est volatil et ne réutilise pas des stéréotypes de guidage standard. La fréquence médiane confirme l'adaptabilité et la diversité du guidage fourni.

Les pourcentages en rapport avec les variables de la ligne qui ont été sélectionnés ou refusés; propagés ou manuels, groupés ou optionnels, montrent que le guidage de ConfiLog ne prive pas les étudiants d'effectuer les choix qui les intéressent ou bien qu'ils orientent le guidage vers des propagations de contraintes ou une sélection des variantes bien données.

Il est à souligner que volontairement, et pour une présentation globale, il n'y a pas eu de filtrage du journal en fonction de complétude des processus, même les processus inachevés et/ou perdus sont considérés dans les statistiques.

Par ailleurs, le guidage proposé aux 113 étudiants était en fonction des buts. 71% des étudiants ont choisi d'effectuer des configurations rapides. Généralement, les configurations rapides se concentrent sur un minimum de choix manuels des étudiants et qui prennent un minimum de temps pour atteindre une configuration finie et valide. 19% des étudiants ont choisi d'effectuer des configurations flexibles. Généralement, les configurations flexibles se concentrent sur les choix manuels des étudiants et qui n'induisent pas à des allers et des retours pour atteindre une configuration finie et valide. Enfin, 10% des étudiants ont choisi d'effectuer des configurations personnalisées. Généralement les configurations personnalisées se concentrent, en premier, sur les variables du modèle en rapport avec le profil de l'étudiant et en deuxième lieu sur l'historique de l'étudiant s'il en a un.

Les modèles de processus de configuration avec guidage et sans guidage sont joints dans les liens en bas de page.⁵

⁵ Lien : <https://www.dropbox.com/home/Map%20models%20evaluation>

VII.3.3. Retour d'expérience des étudiants

Après la configuration du vélo, les étudiants étaient invités à un post-questionnaire récapitulant leur retour d'expérience (cf. Annexe C bis). Les retours d'expérience des étudiants visent à tirer les enseignements positifs et négatifs du guidage fourni. En effet, la présente sous section confronte les retours des étudiants lors de la configuration sans guidage (P1) et avec le guidage de ConfiLog (P2).

11,54% des étudiants confirment que le configurateur a planté au moins une fois pendant la configuration sans guidage (P1), cependant, lors de la configuration avec guidage (P2) aucun étudiant parmi les 113 ne s'est plaint d'un éventuel blocage. Techniquement, cette performance se traduit par la baisse du nombre d'allers/retours sur les choix. Par conséquent, le configurateur prend beaucoup moins de temps pour la conduite et la propagation des choix des étudiants. Le retour d'expérience montre que 26,92% des étudiants de (P1) n'ont pas rencontré de difficultés pour assembler des choix qui sont cohérents. En contrepartie, 73,45% des étudiants de (P2) n'ont trouvé aucun problème là dessus.

Le guidage est jugé utile par 92,92% des étudiants qui ont participé à (P1) et (P2). Les étudiants de (P2) trouvent que le guidage était pertinent à 93,8% pour répondre à leur but de guidage.

Le post-questionnaire comprend aussi des questions en rapport avec le ressenti et la motivation des étudiants tout au long de la configuration du vélo. En effet 84,62% des étudiants de (P1) trouvent que les prises de décisions étaient hyper monotones. Cependant, 43,36% des étudiants de (P2) trouvent que le processus de configuration de vélos était attractif. L'optimisation de ce volet nécessite d'amples considérations d'interactivité et d'ergonomie des interfaces de configuration. Des conducteurs d'Interface-Homme-Machine [Wang et al., 2018] peuvent être adoptés pour mieux animer le processus de prise de décisions.

Étant avantageuse mais critiquable, la méthode ConfiLog nécessite encore des révisions surtout une fois incorporée dans un cadre commercial, là où les exigences fonctionnelles et non fonctionnelles des configurateurs sont doublées. En effet, 18,58% des étudiants de (P2) s'attendent à encore plus de guidage.

VII.3.4. Temps de réponse

A l'issue des premières exécutions de configurateur de ConfiLog, un goulot d'étranglement applicatif en rapport avec le temps nécessaire pour le calcul de guidage, a été identifié. Partant de ce fait, un test de charge avec des scénarios de configuration sur mesure, a été appliqué. L'analyse du problème a révélé que même les variantes qui sont ignorées par la partie prenante ou qui violent la configuration en question sont considérées quand même par le moteur de recommandations sans être recommandées plus tard. Par conséquent, à chaque étape de configuration, le moteur de recommandation continue le calcul des prédicats, le calcul du poids et le calcul de l'impact de la décision de toutes les variantes du modèle.

De façon générale, les variantes éliminées manuellement ou par propagation de choix ne doivent pas figurer dans la configuration finale si, et seulement si, il a eu un retour sur les choix par la partie prenante. A cet égard, le moteur de recommandation de ConfiLog réduit progressivement les datasets de guidage et l'espace de solutions des configurations au fur et à mesure que des décisions sont prises par la partie prenante.

Pour améliorer les performances de l'outil, une technique Balise-Entité ETag [Hayton et al., 2016] a été adoptée afin d'éviter le calcul du guidage pour une variante déjà configurée. Tout bien pesé, le moteur de recommandation ignore le dataset violant la configuration en question et donc réduit le nombre de requêtes faisant analyse des différents datasets. Les performances de ConfiLog en termes de temps de réponse se sont nettement améliorées à l'issue de cette adaptation.

La figure (25) confronte graphiquement deux courbes de temps requis pour l'exécution des mêmes décisions de configuration. La courbe inférieure présente les temps requis lors d'une configuration sans le guidage de ConfiLog (P1), néanmoins, la courbe supérieure présente les temps requis lors d'une configuration avec le guidage de ConfiLog (P2).

L'environnement dans lequel sont effectuées ces mesures est un ordinateur portable avec la configuration suivante :

- Windows 7 Professionnel de 64 bits,
- Un processeur Intel(R) Core™ i5-4200M 2,50 GHz et
- Une mémoire RAM de 8,00 GB.

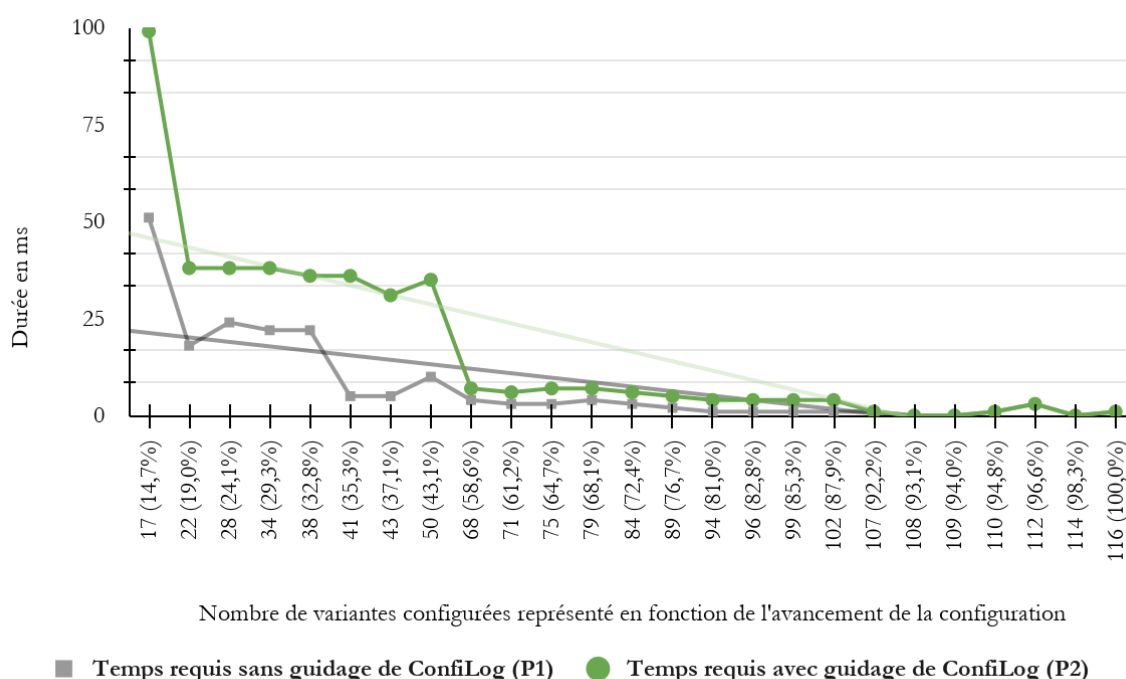


Figure (25) : Comparaison des temps de réponse requis pour les mêmes décisions dans (P1) et (P2)

De prime abord, le graphique montre que les temps de réponse requis lors de (P1) et ceux requis lors de (P2) sont analogues, néanmoins, (P2) prend plus de temps. En effet, ce temps de réponse supplémentaire s’explique par le temps requis pour le calcul de la recommandation. Au fur et à mesure que des décisions sont prises, le moteur de recommandation de ConfiLog fait un balayage des datasets, des pondérations des instances de processus et des calculs d’impact de chaque décision en respect avec le but de guidage exprimé par la partie prenante.

A l’issue de la courbe descendante et fluctuante de (P2), plusieurs interprétations sont effectuées, notamment :

- Le temps de réponse le plus important correspond à la première étape de configuration. Techniquement, la première étape de configuration conduit à des calculs de cohérence en rapport avec tout le reste des variantes du modèle. Le temps de réponse de l’étape 1 jugé le plus important de la configuration n'a pas dépassé 99 millisecondes donc reste raisonnable.
- La courbe est descendante; le temps de réponse pour la première étape était de 99 millisecondes, néanmoins, le temps de réponse pour la dernière étape de configuration était une milliseconde. Ceci s’explique par le fait qu’au fur et à

mesure que le processus de configuration progresse : (i) la configuration partielle est mise à jour, (ii) l'espace de solutions est réduit et (iii) les datasets violant la configuration partielle sont ignorés. C'est ainsi que les opérations de calcul de guidage et la vérification des contraintes, en rapport avec la configuration partielle, seront évitées dans les prochaines étapes. La conduite progressive de la configuration ainsi que la recommandation, favorise une diminution du temps de réponse en fonction de l'avancement du processus.

- La courbe est fluctuante; Une chute libre de la courbe est observée à l'étape 2 vu que le configurateur est passé de 17 à 4 propagations de contraintes. Une légère augmentation, au-dessus de la courbe de tendance, du temps de réponse est constatée à l'étape 8. Cette augmentation représente des décisions impliquant plus de propagations de contraintes que l'étape 2.

VII.3.5. Passage à l'échelle

La présente évaluation s'intéresse à vérifier si le configurateur maintient sa performance en termes de latence et de robustesse lors de la montée en charge des données [Andrey et al., 2006] à gérer. En effet, les configurateurs performants doivent gérer de larges modèles tout en préservant un temps de réponse minime. Par exemple, dans un contexte de boutique en ligne, atteindre le pire cas de temps de réponse [Wilhelm et al., 2008] mène les internautes à se diriger vers d'autres configurateurs plus compétitifs et plus concurrentiels.

Étant donné que ConfiLog repose à la fois sur une recommandation basée sur la connaissance et sur la fouille de processus, le temps de réponse dépend principalement de deux critères, notamment, la taille de la ligne de produits et la taille de repository des traces. Néanmoins pour la performance de ConfiLog au passage à l'échelle, la présente expérimentation ne possède pas des traces de processus pour les différents modèles comparés. La Figure (26) présente l'évolution du temps de réponse nécessaire pour l'exécution des décisions des parties prenantes, en fonction de l'augmentation graduelle de la taille de la ligne.

L'étape 1 est choisie pour effectuer l'évaluation puisqu'elle a été identifiée comme étant l'étape la plus coûteuse en termes de temps de réponse. Cette lourdeur de temps de réponse revient au calcul de cohérence des décisions à partir de tout l'espace de solutions possibles, et le chargement de tous les datasets de guidage au moment où aucun choix

n'est encore introduit. Au fur et à mesure que des décisions sur les choix sont prises, l'espace de solutions ainsi que le nombre de datasets considérés par le guidage sont réduits.

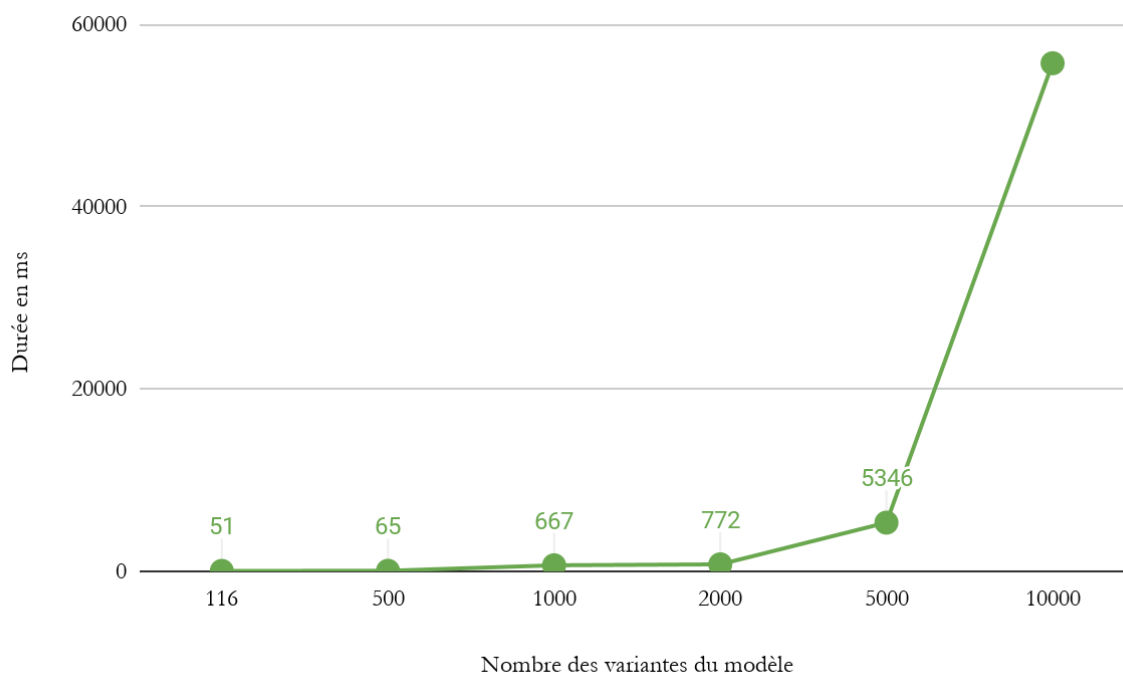


Figure (26): Temps de réponse après la configuration du premier point de variabilité

La figure (26) montre que le temps de réponse reste raisonnable, soit inférieur à une seconde, même pour des modèles de variabilité contenant 3000 variantes. Cependant, le temps de réponse augmente d'une manière exponentielle à partir de 5000 variantes. En dépassant le seuil de modèles contenant 3000 variantes, la performance commence à se dégrader.

VII.4. Menaces à la validité

A l'issue des résultats de l'expérimentation empirique détaillée ci-avant, il est primordial de s'interroger sur la justesse avec laquelle les mesures effectuées mènent à des résultats valides et conformes à différents contextes, pour autant, des menaces à la validité, s'il y'en a, doivent être soulignées.

Il existe différentes classifications des types de menaces à la validité d'une expérimentation. La présente section se concentre sur une classification qui distingue quatre types de validité [Cook et al, 1979] à savoir (i) la validité de la conclusion, (ii) la

validité interne, (iii) la validité conceptuelle et (iv) la validité externe. En effet, la validité de la conclusion reflète la confiance statistique dans l'exactitude des résultats et des conclusions tirées. La validité interne estime le degré de confiance que l'on peut avoir dans les résultats d'une expérimentation. La validité conceptuelle garantit que les mesures utilisées pendant l'expérimentation sont correctement conçues et que les concepts modélisant les traitements reflètent correctement les relations de cause à effet. Finalement, la validité externe estime le degré de confiance que l'on peut avoir dans les conclusions tirées à partir des résultats d'une expérimentation.

En se servant d'une liste de contrôle détaillée dans une référence d'expérimentation logicielle de renommée [Wohlin et al., 2012], les suivantes menaces à la validité de l'expérimentation de ConfiLog sont identifiées :

- *Validité de la conclusion de mesure* : Le passage à l'échelle a été mesuré en termes de la taille du modèle de la ligne de produits en question. C'est le critère le plus important pour les temps de calcul, mais il serait judicieux de mesurer aussi ce critère en termes de taille du repository des traces en fonction duquel le guidage est calculé. L'évaluation du passage à l'échelle doit considérer, également, le nombre et la nature des contraintes ainsi que la métrique des utilisateurs simultanés. Techniquement, la taille de repository de traces peut affecter la performance des traitements de fouille, et par ailleurs, la production du guidage. Néanmoins, cela reste maîtrisable vu que la fouille peut être opérée dans un serveur à part par le biais des interfaces de programmation (API) qu'offrent les outils de fouille de processus et de big data. La gestion des traces de volume colossal et les aspects d'interopérabilité d'outils sont à définir au cas par cas pour chaque entité souhaitant mettre en place la méthode ConfiLog.
- *Validité interne* : Les actions des parties prenantes, stockées dans le repository des traces, sont collectées auprès des étudiants. Cela n'affecte pas négativement les résultats de ConfiLog, mais ne reflète pas les résultats sur des achats réels dans une boutique en ligne de vélos.
- *Validité conceptuelle* : Mise à part la preuve de concept effectuée sur une ligne de sites web et une ligne de téléphonie basique, la méthode ConfiLog a été appliquée que sur un seul cas réel avec des vraies traces de configuration. Pour pousser encore la généralisabilité, cette recherche devrait s'appliquer à d'autres lignes de produits dans d'autres domaines comme l'industrie des automobiles ou la vente en

ligne. Cela permettra d'encore mieux analyser le comportement de ConfiLog face à des complexités et des éventuelles évolutions de lignes de produits.

- *Validité externe* : Dans la présente expérimentation, l'évaluation de la méthode ConfiLog s'est basée principalement sur la valeur ajoutée du guidage, le temps de réponse et le passage à l'échelle. Cependant, une fois ConfiLog est transposée dans un milieu industriel, d'autres critères d'évaluation pourraient se rajouter à savoir la configurabilité de niveau de guidage, l'adaptabilité du modèle des traces et sa faculté de supporter des traces de produits plutôt que des traces de processus. Pareillement, dans un cadre commercial, l'ergonomie et la facilité d'utilisation peuvent être analysées.

VII.5. Conclusion

Ce chapitre a présenté une évaluation de la méthode ConfiLog. Pour ce faire, une étude empirique avec les étudiants pour la configuration d'un vélo a été menée. Le protocole d'expérimentation en fonction duquel cette étude a été conduite, considère deux processus :

- (P1) : Une expérimentation sans se servir d'une méthode particulière de guidage.
- (P2) : Une expérimentation en utilisant le guidage produit par la méthode ConfiLog.

Pour exécuter le protocole d'expérimentation, les étudiants ont été appelés pour participer à (P1) et à (P2) en effectuant les tâches suivantes :

- Répondre à un pré questionnaire : Les réponses au pré-questionnaire ont servi pour mettre en évidence les backgrounds et les attentes des étudiants, d'un côté, et pour répartir les groupes des étudiants d'un autre côté.
- Configurer, en binôme, un vélo en fonction de leurs préférences : Chaque binôme d'étudiants est composé d'un étudiant "Baseline" et d'un étudiant "Connaisseur". Dans (P1), cette tâche a servi pour collecter des traces de processus de configuration. En fonction de ces traces un guidage a été produit pour (P2). Dans (P2), cette tâche a servi pour évaluer le guidage fourni et sa valeur ajoutée.
- Répondre à un post-questionnaire : Le post-questionnaire a servi pour collecter les retours d'expérience des étudiants lors de (P1) et lors de (P2).

Ensuite, une analyse des résultats expérimentaux a été effectuée pour confronter les aboutissements de (P1) à ceux de (P2). En effet, l'application de la méthode ConfiLog sur la ligne de vélos a montré son avantage à réduire le nombre des décisions et la durée du processus de configuration tout en restant flexible et adapté à des parties prenantes différentes. Par ailleurs, elle augmente la motivation des parties prenantes et évite les abandons du processus de configuration.

De surcroît, la méthode ConfiLog prend des temps de réponse minimales pour produire le guidage et exécuter les décisions prises par les parties prenantes. Les temps de réponses de ConfiLog dépassent légèrement les temps de réponses d'un configurateur sans guidage avec quelques millisecondes de plus, néanmoins, ils restent tolérables pour des modèles contenant jusqu'à 3000 variantes.

[Cette page est intentionnellement laissée en blanc]

Chapitre VIII : Conclusion et Perspectives

VIII.1. Résumé des contributions	144
VIII.2. Perspectives de recherche et Discussions ouvertes	146

Chapitre VIII

Conclusion et Perspectives

VIII.1. Résumé des contributions

Dans cette thèse, une nouvelle méthode baptisée ConfiLog est proposée pour le guidage du processus de configuration des lignes de produits. ConfiLog permet d'accompagner la partie prenante à prendre des décisions pendant le processus de configuration et de lui proposer des recommandations correspondant à son but de guidage tout en respectant les contraintes du modèle de la ligne de produits. Le guidage fourni à la partie prenante est dynamique au fur et à mesure que des décisions sont prises. En effet, le guidage va sur deux dimensions, notamment, l'ordre des décisions du processus de configuration et les valeurs à affecter aux variantes du modèle à chaque étape de configuration.

Cette thèse vise alors à répondre à la question de recherche suivante :

“Comment mieux guider les parties prenantes lors du processus de configuration d’une ligne de produits ?”

Pour ce faire, les contributions suivantes ont été réalisées dans cette thèse :

1. *Un état de l'art sur les problèmes de configuration des lignes de produits et les solutions existantes* (cf. Chapitre III). Cette étude a montré que :

- Les problèmes de configuration s'articulent autour de trois phénomènes récurrents dans la littérature, notamment, (i) la complexité de la décision, (ii) le manque de performance et (iii) le besoin de guidage.
- Les solutions de configuration existantes sont classées en quatre catégories, notamment, (i) les méthodes de configuration interactive, (ii) les méthodes basées sur la recommandation, (iii) les méthodes combinées et (iv) les méthodes heuristiques. Cependant, ces solutions demeurent insuffisantes pour accompagner des parties prenantes ayant des préoccupations et des buts différents, à configurer un produit complexe qui s'adapte avec leurs préférences.

- Une forme complémentaire de guidage, qui trace le processus de configuration et les différentes interactions des parties prenantes est à envisager.

2. Une méthode de configuration *ConfiLog* a été proposée pour la configuration guidée des lignes de produits à base des traces de processus (cf. Chapitre IV et chapitre V). En interactivité avec la partie prenante, *ConfiLog* génère des recommandations au fur et à mesure des étapes du processus de configuration en combinant la recommandation basée sur la connaissance et la recommandation basée sur la fouille des traces de processus (cf. Chapitre II). Son principe est de raisonner sur un sous-ensemble de variantes (appelé configuration partielle) et un but de guidage exprimé par la partie prenante. Le guidage est calculé par une fouille des traces de processus de configuration antérieurs de la ligne et un algorithme de recommandation permettant l'orchestration interactive du guidage. En utilisant la spécification de la ligne de produits, une vérification des contraintes est effectuée de manière à ce que le moteur de recommandation ne fournisse jamais un guidage qui viole les contraintes du domaine. De plus, le guidage fourni est dynamique et adapté vu que ce dernier change d'une configuration à une autre et encore d'une étape de configuration à une autre (cf. Chapitre V).

La méthode *ConfiLog* se distingue des solutions existantes par la prise en compte de plusieurs aspects en une seule solution, notamment (i) la cohérence de la solution, (ii) la liberté de la partie prenante, (iii) la double dimension du guidage (ordre du processus et valeurs des variantes), (iv) l'indépendance du guidage vis à vis de l'arborescence du modèle de la ligne, (v) la prise en compte des buts de la partie prenante, (vi) l'incrémentalité et complétude de la solution et (vii) la portabilité du guidage.

3. Une ligne de téléphonie (cf. Chapitre V) et une ligne de site web (cf. Chapitre VI) ont été étudiées dans le cadre d'une preuve de concept. Cette étude a été, en premier lieu, conduite pour des fins exploratoires. La preuve de concept a permis, par ailleurs, de reformuler les objectifs du projet de recherche, de soulever puis de résoudre un certain nombre de verrous tels que les attributs à tracer dans le journal des événements, et les questions de guidage à se poser.

4. Un outillage qui supporte la méthode *ConfiLog* a été développé ^{1/2} (cf. Chapitre VI). Étant générique, l'outillage implémenté englobe cinq grandes fonctionnalités, respectivement : (i) la modélisation de la variabilité d'un système, (ii) la vérification de

¹ Lien : www.youtube.com/watch?v=Ds69w86GCzQ&t=43s

² Lien : www.github.com/HousseM/CHEMINGUI/ConfigLogFFV

sa cohérence, (iii) la configuration guidée en fonction des buts, (iv) la capitalisation des traces et (v) la production du guidage fondé sur la fouille de processus. Ce prototype a été implémenté en étendant l'open source Splot avec la méthode ConfiLog.

5. Une évaluation de la méthode ConfiLog a été réalisée en fonction d'une étude empirique avec les étudiants de l'université Paris 1 Panthéon Sorbonne divisés en deux groupes (cf. Chapitre VII). Le premier groupe d'étudiants était invité à configurer une ligne de vélos sans le guidage de ConfiLog, cependant, le deuxième groupe d'étudiants était invité à configurer une ligne de vélos avec le guidage de ConfiLog. L'évaluation effectuée est basée particulièrement sur une confrontation des traces de configuration d'étudiants du premier groupe et des traces de configuration d'étudiants du deuxième groupe. De plus, des retours d'expérience des étudiants après l'expérimentation ont été considérés.

VIII.2. Perspectives de recherche et Discussions ouvertes

Cette section présente des perspectives de recherche en relation avec les limites de la méthode de configuration ConfiLog.

Compte tenu des contributions de cette thèse, plusieurs questions scientifiques se posent concernant les conséquences et les inconvénients de ConfiLog pour les praticiens. A cet égard, il s'avère que les modèles de variabilité de lignes de produits évoluent dans le temps [Dhungana et al., 2010]. Généralement, l'évolution d'un modèle de variabilité revient à une évolution des exigences, des implémentations, des tests, etc. En effet, des nouvelles variantes et/ou contraintes de domaine peuvent être incorporées dans le nouveau modèle. Cependant, dans les lignes de produits dynamiquement adaptables [Hallsteinsen et al., 2008] [Alferez et al., 2014], le produit s'auto adapte dynamiquement pour répondre aux nouvelles évolutions. Par conséquent, les directives de guidage de ConfiLog ne pourraient pas éviter la divergence entre ce qui est capitalisé dans les traces et ce que contient le nouveau produit.

Techniquement, le guidage basé sur la version précédente de la ligne devient incompatible avec les nouvelles spécifications du modèle de la ligne. Cette réflexion amène à se poser des questions : Quels artefacts peuvent être exploités et ré-utilisés indépendamment des évolutions de la ligne? Encore plus loin, quelles dépendances et corrélations existent entre le processus de configuration et le modèle de la ligne? Quel est le degré de dépendance entre le modèle de la ligne et le guidage produit? Cette évolution de la ligne remet-elle en question encore une fois le phénomène de démarrage

à froid? En effet, dans un contexte de fouille de processus l'efficacité et la précision du guidage sont renforcées particulièrement si la base des traces est assez riche.

Il est à souligner, également, que le guidage produit par la méthode ConfiLog est toujours lié à un seul but exprimé par la partie prenante avant d'entamer le processus de configuration. Néanmoins, cette conception prive la partie prenante de changer d'avis pendant la configuration. Par exemple, un internaute qui va sur une boutique en ligne de voitures et qui souhaite procéder à une configuration flexible. A force de lui suggérer beaucoup de choix, il décide de changer de but de guidage afin de configurer rapidement les variantes qui restent. De surcroît, il s'avère que des fois la partie prenante ne sait pas exactement quel but de guidage qui va le mieux avec ses exigences ou qu'une combinaison des buts lui semble intéressante. En perspective, la considération des méthodes situationnelles [Kraiem et al., 2010], des méthodes intentionnelles [Rolland et al., 2010] et/ou des méthodes contextuelles [Frederiksen et al., 2008] dans la méthode ConfiLog semble capable d'adapter le guidage produit aux exigences alternantes et parfois floues de la partie prenante. Par ailleurs, l'intégration de ces méthodes augmente la liberté des parties prenantes. En effet, ces dernières ne se sentent plus accablées par un seul but de guidage. C'est d'autant plus avantageux de muter d'un but de guidage à un autre selon les spécificités du processus de configuration en question. Les approches citées ci-dessus doivent permettre la standardisation des activités, mais elles doivent, cependant, rester suffisamment flexibles et contrôlées pour mieux connaître les intentions de la partie prenante, le contexte en question et/ou la situation rencontrée.

Le contexte de processus de configuration amène à se poser encore des questions sur une perspective en rapport avec les méthodes de configuration collaboratives [Edded et al., 2019] là où plusieurs parties prenantes sont impliquées dans le même processus de configuration. Souvent, les configurations collaboratives mènent à des décisions contradictoires et/ou conflictuelles vu que ces dernières viennent de plusieurs acteurs différents à la fois. Mis à part le fait que les méthodes collaboratives nécessitent d'amples réflexions par rapport à la résolution des conflits, du guidage semble prometteur mais complexe. En effet, la potentielle solution doit prendre en compte plusieurs buts de guidage à la fois et gérer différentes priorités de configuration. Dans ce cas de figure, l'application de la méthode ConfiLog préserve-t-elle la liberté de la partie prenante et la flexibilité des décisions?

Par analogie aux lignes de produits, les autres systèmes configurables tels que les (ERP) et les (COTS) sont aussi contraints par le grand nombre d'exigences volatiles et de contraintes à satisfaire simultanément. La configuration devient rapidement sujette à l'erreur et confronte les parties prenantes à des problèmes de rigueur et de passage à l'échelle lors de la configuration de logiciels : les méthodes employées pour spécifier les systèmes à configurer ne permettent pas de maîtriser de manière formelle et systématique un nombre important de décisions complexes. Cependant, des approches visant à surmonter ces verrous existent et ont été publiées, mais dans certains domaines; peuvent-elles être adaptées pour toute sorte de logiciels configurables? Les défis scientifiques de la configuration logicielle peuvent-ils ainsi être transposés? [Chemingui et al., 2020]

D'un point de vue organisationnel, une recherche collaborative peut mieux conduire l'enquête sur les défis réels des logiciels configurables. Des méthodes telles que la recherche participative [Chevalier et al., 2019] et la recherche ethnographique [Holland et al., 2004] peuvent être recommandées pour mieux comprendre la conception de la configuration et les potentiels intérêts en commun des praticiens. Trop souvent, la valorisation de la recherche permet de motiver les fondements technologiques réels et leur utilité à surmonter les problèmes métiers. Sauf que véritablement, il existe une fracture entre les besoins de l'industrie et sur quoi se concentre la recherche universitaire au moins dans le contexte de configuration logicielle générique. Ceci s'explique par le fait que habituellement, l'interaction entre les acteurs industriels et les chercheurs se heurte à plusieurs barrières vu que les entreprises s'enferment sur eux même et gardent en secret leurs activités stratégiques. Par conséquent, la crainte de ne pas respecter leur confidentialité impacte significativement la valorisation de la recherche. En effet, le manque de collaboration entre les chercheurs et les praticiens crée des lacunes de fiabilité rendant plus complexe la valorisation des résultats de la recherche et la connexion des meilleures pratiques. Néanmoins, plus d'ouverture des chercheurs sur l'industrie optimise effectivement les méthodes et les outils issus de la recherche universitaire et épargne les incompatibilités non anticipées et parfois imprévues.

[Cette page est intentionnellement laissée en blanc]

REFERENCES BIBLIOGRAPHIQUES

[Aalst, 2011] Aalst, W. (2011). Discovery, Conformance and Enhancement of Business Processes.

[Aalst, 2012] Aalst, W. (2012). Process mining. *Communications of the ACM*, 55(8), 76-83.

[Acher et al., 2013] Acher, M., Collet, P., Lahire, P., & France, R. B. (2013). Familiar: A domain-specific language for large scale management of feature models. *Science of Computer Programming*, 78(6), 657-681.

[Adomavicius et al., 2005] Adomavicius, G., Tuzhilin, A. (2005). "Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions". *IEEE Trans. Knowl. Data Eng.* 17(6), 734.

[Afzal et al., 2016] Afzal, U., Mahmood, T., & Shaikh, Z. (2016). Intelligent software product line configurations: A literature review. *Computer Standards & Interfaces*, 48, 30-48.

[Afzal et al., 2020] Afzal, U., Mahmood, T., Khan, A. H., Jan, S., Rasool, R. U., Qamar, A. M., & Khan, R. U. (2020). Feature Selection Optimization in Software Product Lines. *IEEE Access*, 8, 160231-160250.

[Aken, 2004] Aken, J. E. V. (2004). Management research based on the paradigm of the design sciences: the quest for field-tested and grounded technological rules. *Journal of management studies*, 41(2), 219-246.

[Alfárez et al., 2014] Alfárez, G. H., Pelechano, V., Mazo, R., Salinesi, C., & Diaz, D. (2014). Dynamic adaptation of service compositions with variability models. *Journal of Systems and Software*, 91, 24-47.

[Anbuselvan, 2015] Anbuselvan, A. (2015). U.S. Patent No. 8,996,658. Washington, DC: U.S. Patent and Trademark Office.

[Andrey et al., 2006] Andrey, L., Festor, O., & Lahmadi, A. (2006, October). Evaluation du passage à l'échelle des systèmes de gestion: métriques et modèles.

[Astesana et al. 2010] Astesana, J. M., Cosserat, L., & Fargier, H. (2010). Constraint-based Modeling and Exploitation of a Vehicle Range at Renault's: Requirement analysis and complexity study. In *Workshop on Configuration* (Vol. 33).

[Astesana et al., 2010] Astesana, J.M., Cosserat, L., Fargier, H.: Constraint-based vehicle configuration: A case study. In: 2010 22nd IEEE International Conference on Tools with Artificial Intelligence (ICTAI), vol. 1, pp. 68 –75 (2010). DOI 10.1109/ICTAI.2010.19

[Bagheri et al., 2014] Bagheri, E., & Ensan, F. (2014). Dynamic decision models for staged software product line configuration. *Requirements Engineering*, 19(2), 187-212.

[Basili et al., 1994] Basili, V. R., Caldiera, G., & Rombach, H. D. (1994). The Goal Question Metric Approach, Chapter in *Encyclopedia of Software Engineering*. Wiley.

[Batory, 2005] D. Batory. Feature models, grammars, and propositional formulas. In *Software Product Lines Conference*, volume 3714 of *Lecture Notes in Computer Sciences*, pages 7–20. Springer–Verlag, 2005.

[Benavides et al., 2010] Benavides, D., Segura, S., & Ruiz-Cortés, A. (2010). Automated analysis of feature models 20 years later: A literature review. *Information systems*, 35(6), 615-636.

[Benavides et al., 2007] D. Benavides, S. Segura, P. Trinidad, and A. Ruiz-Cortés. FAMA: Tooling a framework for the automated analysis of feature models. In *Proceeding of the First International Workshop on Variability Modelling of Software-intensive Systems (VAMOS)*, pages 129–134, 2007.

[Berry, 2004] Berry, M. W. (2004). Survey of text mining. *Computing Reviews*, 45(9), 548.

[Booth et al., 2003] Booth, W. C., Booth, W. C., Colomb, G. G., Colomb, G. G., Williams, J. M., & Williams, J. M. (2003). *The craft of research*. University of Chicago press.

[Botterweck et al., 2007] Botterweck, G., Nestor, D., PreuBner, A., Cawley, C., & Thiel, S. (2007). Towards supporting feature configuration by interactive visualisation.

[Bratosin et al., 2010] Bratosin, C., Sidorova, N., & van der Aalst, W. (2010, July). Distributed genetic process mining. In *IEEE Congress on Evolutionary Computation* (pp. 1-8). IEEE.

[Breese et al., 1998] Breese, J. S., Heckerman, D., Kadie, C. (1998). “Empirical analysis of predictive algorithms for collaborative filtering”. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Madison, WI.

[Burke, 2007] Burke, R. (2007). “Hybrid Web Recommender Systems”. *The Adaptive Web 2007*, pages 377-408.

[Burke, 2002] Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4), 331-370.

[Burke, 2000] Burke, R. (2000). “Knowledge-based recommender systems”. *Library and Information Systems*, 69(32), pages180–200.

[Cai et al, 2011] Cai, S., & Su, K. (2011, November). Local search with configuration checking for SAT. In *2011 IEEE 23rd international conference on tools with artificial intelligence* (pp. 59-66). IEEE.

[Carver et al., 2010] Carver, J. C., Jaccheri, L., Morasca, S., & Shull, F. (2010). A checklist for integrating student empirical studies with research and teaching goals. *Empirical Software Engineering*, 15(1), 35-59.

[Chastel et al., 2015] Chastel, F., Blay-Fornarino, M., Pereira, C. D. C., & Urli, S. (2015). Apprendre des actions utilisateur pour guider le processus de configuration dans les lignes de produits logiciels.

[Chemingui et al., 2020] Chemingui, H., Gam, I., Mazo, R., Ghezala, H. B., & Salinesi, C. (2020). A Unified Vision of Configurable Software. In *INFORSID* (pp. 93-98).

[Chemingui et al., 2019] Chemingui, H., Gam, I., Mazo, R., Salinesi, C., & Ghezala, H. B. (2019). Product Line Configuration Meets Process Mining. *Procedia Computer Science*, 164, 199-210.

[Chevalier et al., 2019] Chevalier, J. M., & Buckles, D. J. (2019). *Participatory action research: Theory and methods for engaged inquiry*. Routledge.

[Chickering et al., 1997] Chickering, D., Heckerman, D., Meek, C. (1997). "A Bayesian approach to learning Bayesian networks with local structure". In *Proceedings of Thirteenth Conference on Uncertainty in Artificial Intelligence*, Providence, RI. Morgan Kaufmann.

[Cirilo et al., 2007] Cirilo, E., Kulesza, U., & de Lucena, C. J. P. (2007, August). GenArch-A Model-Based Product Derivation Tool. In *SBCARS* (pp. 31-46).

[Clement et al., 2002] Clement, P., & Northrop, L. (2002). *Software Product Lines, Practices and Pattern*.

[Cook et al., 1979] Cook, T. D., Campbell, D. T., & Day, A. (1979). *Quasi-experimentation: Design & analysis issues for field settings* (Vol. 351). Boston: Houghton Mifflin.

[Cöster et al., 2002] Coester, R., Gustavsson, A., Olsson, R. Rudstroem, A. (2002). "Enhancing web-based configuration with recommendations and cluster-based help". In *AH'02 Worksh. On Recommendation and Personalization in EComm.*, Malaga, Spain.

[Deelstra et al., 2003] Deelstra, S., Sinnema, M., & Bosch, J. (2003, November). A product derivation framework for software product families. In *International Workshop on Software Product-Family Engineering* (pp. 473-484). Springer, Berlin, Heidelberg.

[Deneckere et al., 2014] Deneckere, R., Hug, C., Khodabandelou, G., & Salinesi, C. (2014). Intentional process mining: discovering and modeling the goals behind processes using supervised learning. *International Journal of Information System Modeling and Design (IJISMD)*, 5(4), 22-47.

[Deneckere et al., 2010] Deneckère, R., & Kornysheva, E. (2010). Process line configuration: An indicator-based guidance of the intentional model map. In *Enterprise, Business-Process and Information Systems Modeling* (pp. 327-339). Springer, Berlin, Heidelberg.

[Dhungana et al., 2011] D. Dhungana, P. Grunbacher, and R. Rabiser, "The DOPLER meta- tool for decision-oriented variability modeling: A multiple case study," *Automated Software Engineering*, vol. 18, no. 1, pp. 77–114, 2011.

[Dhungana et al., 2010] Dhungana, D., Grünbacher, P., Rabiser, R., & Neumayer, T. (2010). Structuring the modeling space and supporting evolution in software product line engineering. *Journal of Systems and Software*, 83(7), 1108-1122.

[Dhungana et al., 2007] Dhungana, D., Rabiser, R., Grünbacher, P., & Neumayer, T. (2007, November). Integrated tool support for software product line engineering. In *Proceedings of*

the twenty-second IEEE/ACM international conference on Automated software engineering (pp. 533-534).

[Djebbi, 2011] Djebbi, O. (2011). L'ingénierie des exigences par et pour les lignes de produits (Doctoral dissertation, Paris 1).

[Dumitrescu et al., 2013 a] Dumitrescu, C., Mazo, R., Salinesi, C., & Dauron, A. (2013, August). Bridging the gap between product lines and systems engineering: an experience in variability management for automotive model based systems engineering. In Proceedings of the 17th International Software Product Line Conference (pp. 254-263).

[Dumitrescu et al., 2013 b] Dumitrescu, C., Tessier, P., Salinesi, C., Gérard, S., & Dauron, A. (2013). Flexible product line derivation applied to a model based systems engineering process. In Complex Systems Design & Management (pp. 227-239). Springer, Berlin, Heidelberg.

[Edded et al., 2019] Edded, S., Sassi, S. B., Mazo, R., Salinesi, C., & Ghezala, H. B. (2019). Collaborative configuration approaches in software product lines engineering: A systematic mapping study. *Journal of Systems and Software*, 158, 110422.

[Elsner et al., 2010] Elsner, C., Ulbrich, P., Lohmann, D., & Schröder-Preikschat, W. (2010, September). Consistent product line configuration across file type and product line boundaries. In International Conference on Software Product Lines (pp. 181-195). Springer, Berlin, Heidelberg.

[Engström et al., 2020] Engström, E., Storey, M. A., Runeson, P., Höst, M., & Baldassarre, M. T. (2020). How software engineering research aligns with design science: a review. *Empirical Software Engineering*, 25(4), 2630-2660.

[Epure et al., 2017a] Epure, E. V., Deneckere, R., Salinesi, C., Kille, B., & Ingvaldsen, J. (2017, May). Devising News Recommendation Strategies with Process Mining Support.

[Epure, al., 2017b] Epure, E. V., Kille, B., Ingvaldsen, J. E., Deneckere, R., Salinesi, C., & Albayrak, S. (2017, August). Recommending personalized news in short user sessions. In Proceedings of the Eleventh ACM Conference on Recommender Systems (pp. 121-129). ACM.

[Falkner et al., 2011] Falkner, A., Felfernig, A., & Haag, A. (2011). Recommendation technologies for configurable products. *Ai Magazine*, 32(3), 99-108.

[Farahani et al., 2019] Farahani, E. D., & Habibi, J. (2019). Feature model configuration based on two-layer modeling in Software Product Lines. *International Journal of Electrical & Computer Engineering* (2088-8708), 9.

[Feichtinger et al., 2020] Feichtinger, K., & Rabiser, R. (2020, October). Towards Transforming Variability Models: Usage Scenarios, Required Capabilities and Challenges. In Proceedings of the 24th ACM International Systems and Software Product Line Conference-Volume B (pp. 44-51).

[Felfernig et al., 2014] Felfernig, A. (2014). “Biases in decision making”. In International workshop on decision making and recommender systems, volume 1278, pages 32-37. CEUR Proceedings.

[Felfernig et al., 2008] Felfernig, A., Burke, R. (2008). “Constraint-based recommender systems: technologies and research issues”. In Proceedings of the 10th international conference on electronic commerce (ISEC08).

[Felfernig et al., 2007] Felfernig, A., Isak, K., Szabo, K., Zachar, P. (2007). “The VITA Financial Services Sales Support Environment”. AAAI/IAAI, pages 1692-1699, Vancouver, Canada.

[Fraternali et al., 2010] Fraternali, P., Rossi, G., & Sánchez-Figueroa, F. (2010). Rich internet applications. *IEEE Internet Computing*, 14(3), 9-12.

[Frederiksen et al., 2008] Frederiksen, H. D., & Mathiassen, L. (2008). A contextual approach to improving software metrics practices. *IEEE Transactions on engineering management*, 55(4), 602-616.

[Gorrab et al., 2017] Gorrab, A., Koubi, F., Le Grand, B., & Ghezala, H. B. (2017, October). New hashtags’ weighting schemes for Hashtag and User Recommendation on Twitter. In 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA) (pp. 564-570). IEEE.

[Guana et al., 2013] Guana, V., & Correal, D. (2013). Improving software product line configuration: A quality attribute-driven approach. *Information and Software Technology*, 55(3), 541-562.

[Günther et al., 2007] Günther, C. W., & Van Der Aalst, W. M. (2007, September). Fuzzy mining—adaptive process simplification based on multi-perspective metrics. In International conference on business process management (pp. 328-343). Springer, Berlin, Heidelberg.

[Guo, 2006] Guo, L. (2006). Consumption flexibility, product configuration, and market competition. *Marketing Science*, 25(2), 116-130.

[Hadzic et al., 2004] Hadzic T., Andersen H. R. An introduction to solving interactive configuration problems. Technical Report TR-2004-49, The IT University of Copenhagen, 2004.

[Hadzic et al., 2004] Hadzic, T., Subbarayan, S., Jensen, R. M., Andersen, H. R., Møller, J., & Hulgaard, H. (2004). Fast backtrack-free product configuration using a precompiled solution space representation. *small*, 10(1), 3.

[Hallsteinsen et al., 2008] Hallsteinsen, S., Hinchey, M., Park, S., & Schmid, K. (2008). Dynamic software product lines. *Computer*, 41(4), 93-95.

[Hand et al., 2014] Hand, D. J., & Adams, N. M. (2014). Data Mining. *Wiley StatsRef: Statistics Reference Online*, 1-7.

[Hayashi et al., 2017] Hayashi, K., Aoyama, M., & Kobata, K. (2017, September). Agile Tames Product Line Variability: An Agile Development Method for Multiple Product Lines of

Automotive Software Systems. In Proceedings of the 21st International Systems and Software Product Line Conference-Volume A (pp. 180-189). ACM.

[Hayton et al., 2016] Hayton, S. J., Jones, D. R., Lobo, A. R., Nicholson, R. B., & Vetterlein, G. P. (2016). U.S. Patent No. 9,253,278. Washington, DC: U.S. Patent and Trademark Office.

[Herlocker et al., 2004] Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1), 5-53.

[Hevner et al., 2004] Hevner, A.R., March, S.T., Park, J., Ram, S. (2004). "Design science in information system research". *MIS Quarterly*, 28(1), pages 75-105.

[Holland et al., 2004] Holland, D., & Leander, K. (2004). Ethnographic studies of positioning and subjectivity: An introduction. *Ethos*, 32(2), 127-139.

[Horcas et al., 2017] Horcas, J. M., Pinto, M., & Fuentes, L. (2017, September). Green configurations of functional quality attributes. In Proceedings of the 21st International Systems and Software Product Line Conference-Volume A (pp. 79-83). ACM.

[Hubaux et al., 2010] Hubaux, A., Heymans, P., Schobbens, P. Y., & Deridder, D. (2010, June). Towards multi-view feature-based configuration. In International Working Conference on Requirements Engineering: Foundation for Software Quality (pp. 106-112). Springer, Berlin, Heidelberg.

[Hubaux et al., 2009] Hubaux, A., Classen, A., Heymans, P.: Formal modelling of feature configuration workflow. In: SPLC'09, San Francisco, CA, USA (2009)

[Jannack et al., 2011] Jannack, D., Zanker, M., Felfering A., & Friedrich G. (2011) Recommender Systems, an Introduction. Cambridge University Press. New York, USA, 335 pages.

[Kang et al., 1990] Kang, K. C., Cohen, S. G., Hess, J. A., Novak, W. E., & Peterson, A. S. (1990). Feature-oriented domain analysis (FODA) feasibility study (No. CMU/SEI-90-TR-21). Carnegie-Mellon Univ Pittsburgh Pa Software Engineering Inst.

[Keller et al., 1985] Keller, J. M., Gray, M. R., & Givens, J. A. (1985). A fuzzy k-nearest neighbor algorithm. *IEEE transactions on systems, man, and cybernetics*, (4), 580-585.

[Khodabandelou et al., 2013] Khodabandelou, G., Hug, C., Deneckere, R., & Salinesi, C. (2013). Process mining versus intention mining. In *Enterprise, Business-Process and Information Systems Modeling* (pp. 466-480). Springer, Berlin, Heidelberg.

[Krueger et al., 2013] Krueger, C., & Clements, P. (2013, August). Systems and software product line engineering with BigLever software gears. In Proceedings of the 17th International Software Product Line Conference co-located workshops (pp. 136-140).

[Kitchenham et al., 2009] Kitchenham, B., Brereton, O. P., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering—a systematic literature review. *Information and software technology*, 51(1), 7-15.

[Kraiem et al., 2010] Kraiem, N., Selmi, S., & Ghezala, H. H. B. (2010). A situational approach for web applications design. *IJCSI*, 37.

[Lackner et al., 2014] Lackner, H., Thomas, M., Wartenberg, F., & Weibleder, S. (2014, March). Model-based test design of product lines: Raising test design to the product line level. In *2014 IEEE Seventh International Conference on Software Testing, Verification and Validation* (pp. 51-60). IEEE.

[Likert, 1932] Likert, R. (1932). A technique for the measurement of attitudes. *Archives of psychology*.

[Lin et al., 2006] Lin, Y. H., Ko, T. M., Chuang, T. R., & Lin, K. J. (2006). Open source licenses and the creative commons framework: License selection and comparison. *Journal of information science and engineering*, 22(1), 1-17.

[Linden et al., 2003] Linden, G., Smith, B., & York, J. (2003). Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7(1), 76-80.

[Lu et al., 2016] Lu, H., Yue, T., Ali, S., & Zhang, L. (2016, April). Nonconformity resolving recommendations for product line configuration. In *2016 IEEE International Conference on Software Testing, Verification and Validation (ICST)* (pp. 57-68). IEEE.

[March et Smith, 1995] March, A.T., Smith, G.F. (1995). "Design and natural science research on information technology". *Decision Support Systems*, 15(4), pages 251-266.

[Mazo et al., 2014] Mazo, R., Dumitrescu, C., Salinesi, C., Diaz, D. (2014). "Recommendation Heuristics for Improving Product Line Configuration Processes". In *Recommendation Systems in Software Engineering*. Robillard M, Maalej W., Walker R. and Zimmermann T. (Eds.), Springer-Verlag Berlin Heidelberg.

[Mendonca et al., 2008] Mendonça, M., Bartolomei, T. T., & Cowan, D. (2008, March). Decision-making coordination in collaborative product configuration. In *Proceedings of the 2008 ACM symposium on Applied computing* (pp. 108-113). ACM.

[Mendonca et al., 2007] Mendonca, M., Cowan, D., & Oliveira, T. (2007, January). A process-centric approach for coordinating product configuration decisions. In *2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07)* (pp. 283a-283a). IEEE.

[Mirzadeh et al., 2005] Mirzadeh, N., Ricci, F., Bansal, M. (2005). « Feature Selection Methods for Conversational Recommender Systems ». In *IEEE International Conference on e-Technology, e-Commerce, and e-Service (EEE'05)*, Hongkong, pages 772-777.

[Mooney et al., 2000] Mooney, R. J., & Roy, L. (2000, June). Content-based book recommending using learning for text categorization. In *Proceedings of the fifth ACM conference on Digital libraries* (pp. 195-204).

[Morgan, 2013] Morgan, A. M. (2013). 'Proof of concept': Beginning to use design-based research to improve science literacies for middle years learners. *Australian Journal of Language and Literacy*, The, 36(1), 3.

[Noorian et al., 2017] Noorian, M., Bagheri, E., & Du, W. (2017). Toward automated quality-centric product line configuration using intentional variability. *Journal of Software: Evolution and Process*, 29(9), e1870.

[Noorian et al., 2014] Noorian, M., Bagheri, E., & Du, W. (2014). From Intentions to Decisions: Understanding Stakeholders' Objectives in Software Product Line Configuration. In *SEKE* (pp. 671-677).

[Ochoa et al., 2017] Ochoa, L., Pereira, J. A., González-Rojas, O., Castro, H., & Saake, G. (2017, February). A survey on scalability and performance concerns in extended product lines configuration. In *Proceedings of the Eleventh International Workshop on Variability Modelling of Software-intensive Systems* (pp. 5-12). ACM.

[Ouali et al. 2011a] Ouali, S., Kraiem, N., & Ghezala, H. B. (2011, January). Toward a Comprehension View of Software Product Line. In *International Conference on Computer Science and Information Technology* (pp. 439-451). Springer, Berlin, Heidelberg.

[Ouali et al. 2011b] Ouali, S., Kraiem, N., & Ghezala, H. B. (2011). Framework for evolving software Product line. *International Journal of Software Engineering & Applications*, 2(2), 34-51.

[Oufaida et al, 2008] Oufaida, H., & Nouali, O. (2008). Le filtrage collaboratif et le web 2.0. *Document numérique*, 11(1), 13-35.

[Pazzani et al, 2007] Pazzani, M., Billsus, D. (2007). "Content-Based Recommendation Systems". In *The Adaptive Web: Methods and Strategies of Web Personalization*, Lecture Notes in Computer Science, Vol. 4321. Springer-Verlag, Berlin Heidelberg New York.

[Pazzani et al., 1997] Pazzani, M. and D. Billsus. (1997). "Learning and revising user profiles: The identification of interesting web sites". *Machine Learning*, Vol. 27, pages 313-331.

[Peppers et al., 2007] Peppers, K., Tuunanen, T., Chatterjee, M.A, Rothenberger, S. (2007). "A design science research methodology for information systems research". *Journal of Management Information Systems* 24(3), pages 45--77.

[Pereira et al., 2017] Pereira, J. A., Matuszyk, P., Krieter, S., Spiliopoulou, M., & Saake, G. (2017). A feature-based personalized recommender system for product-line configuration. *ACM SIGPLAN Notices*, 52(3), 120-131.

[Pohl et al., 2005] Pohl, K., Böckle, G., & van Der Linden, F. J. (2005). *Software product line engineering: foundations, principles and techniques*. Springer Science & Business Media.

[Pourbafrani et al., 2019] Pourbafrani, M., van Zelst, S. J., & van der Aalst, W. M. (2019, October). Scenario-Based Prediction of Business Processes Using System Dynamics. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"* (pp. 422-439). Springer, Cham

- [Rao et al., 2008] Rao, N.K., Talwar, V.G. (2008). “Application domain and functional classification of recommender systems a survey”. In *Desidoc journal of library and information technology*, 28(3), pages 17-36.
- [Rashid et al., 2002] Rashid, A. M., Albert, I., Cosley, D., Lam, S. K., McNee, S. M., Konstan, J. A., & Riedl, J. (2002, January). Getting to know you: learning new user preferences in recommender systems. In *Proceedings of the 7th international conference on Intelligent user interfaces* (pp. 127-134).
- [Resnick et al., 1997] Resnick, P., & Varian, H. R. (1997). Recommender systems. *Communications of the ACM*, 40(3), 56-58.
- [Resnick et al., 1994] Resnick, P., Iakovou, N., Sushak, M., Bergstrom, P., Riedl, J. (1994). “GroupLens: An open architecture for collaborative filtering of netnews”. In *Proceedings of the Computer Supported Cooperative Work Conference*.
- [Rincon et al., 2015] Rincón, L., Giraldo, G., Mazo, R., Salinesi, C., & Diaz, D. (2015). Method to identify corrections of defects on product line models. *Electronic notes in theoretical computer science*, 314, 61-81.
- [Rolland et al., 2010] Rolland, C., Kirsch-Pinheiro, M., & Souveyet, C. (2010). An intentional approach to service engineering. *IEEE Transactions on Services Computing*, 3(4), 292-305.
- [Rosenmuller et al., 2010] Rosenmüller, M., & Siegmund, N. (2010). Automating the Configuration of Multi Software Product Lines. *VaMoS*, 10, 123-130.
- [Salinesi et al., 2012] Salinesi, C., Triki, R., & Mazo, R. (2012). Combining configuration and recommendation to define an interactive product line configuration approach. *arXiv preprint arXiv:1206.2520*.
- [Salinesi et al., 2011] Salinesi, C., Mazo, R., Djebbi, O., Diaz, D., & Lora-Michiels, A. (2011, May). Constraints: The core of product line engineering. In *2011 fifth international conference on research challenges science* (pp 1-10). IEEE
- [Salton, 1989] “Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer”. Addison-Wesley, Reading, MA.
- [Sarwar et al., 2001] Sarwar, B., Karypis, G., Konstan, J., Reidl, J. (2001). “Itembased collaborative filtering recommendation algorithms”. In *Proceedings of the 10th international conference on World Wide Web*, pages 285-295.
- [Sayyad et al., 2013] Sayyad, A. S., Ingram, J., Menzies, T., & Ammar, H. (2013, May). Optimum feature selection in software product lines: Let your model and values guide your search. In *Proceedings of the 1st International Workshop on Combining Modelling and Search-Based Software Engineering* (pp. 22-27). IEEE Press.
- [Schein et al., 2002] Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M. (2002, August). Methods and metrics for cold-start recommendations. In *Proceedings of the 25th*

annual international ACM SIGIR conference on Research and development in information retrieval (pp. 253-260).

[Schmid et al., 2006] Schmid, K., Krennrich, K., & Eisenbarth, M. (2006, August). Requirements management for product lines: extending professional tools. In 10th International Software Product Line Conference (SPLC'06) (pp. 10-pp). IEEE.

[Sinnema et al., 2004] Sinnema, M., Deelstra, S., Nijhuis, J., & Bosch, J. (2004, August). Covamof: A framework for modeling variability in software product families. In International Conference on Software Product Lines (pp. 197-213). Springer, Berlin, Heidelberg.

[Sinnhofer et al., 2016] Sinnhofer, A. D., Pühringer, P., Potzmader, K., Orthacker, C., Steger, C., & Kreiner, C. J. (2016, June). A framework for process driven software configuration. In BMSD 2016 (pp. 1-8).

[Sion et al., 2016] Sion, L., Van Landuyt, D., Joosen, W., & de Jong, G. (2016, September). Systematic quality trade-off support in the software product-line configuration process. In Proceedings of the 20th International Systems and Software Product Line Conference (pp. 164-173). ACM.

[Thabet et al., 2018] Thabet, D., Ghannouchi, S. A., & Ghezala, H. H. B. (2018, September). A Process Mining-Based Solution for Business Process Model Extension with Cost Perspective Context-Based Cost Data Analysis and Case Study. In IFIP International Conference on Computer Information Systems and Industrial Management (pp. 434-446). Springer, Cham.

[Thabet et al., 2017] Thabet, D., Ayachi Ghannouchi, S., & Hajjami Ben Ghézala, H. (2014, April). Towards Business Process Model Extension with Cost Perspective Based on Process Mining. In Proceedings of the 16th International Conference on Enterprise Information Systems-Volume 3 (pp. 335-342). SCITEPRESS-Science and Technology Publications, Lda.

[Tran et al, 2000]: Tran, T., Cohen, R. (2000). "Hybrid Recommender Systems for Electronic Commerce". In Knowledge-Based Electronic Markets, Papers from the AAAI Workshop, Technical Report WS-00-04, AAAI Press.

[Triki et al., 2014] Triki, R., Mazo, R., Salinesi, C. (2014). "Combining configuration and recommendation to enable an interactive guidance of product line configuration". In Gérald KEMBELLEC, Ghislaine CHARTRON and Imad SALEH (Ed.). Recommender Systems. pages 135-154. Paris-France: Wiley-ISTE.

[Trinidad et al., 2008] P. Trinidad, D. Benavides, and A. Ruiz-Cortés. A first step detecting inconsistencies in feature models. In CAiSE Short Paper Proceedings, 2006.

[Urli et al., 2012] Urli, S., Perez, G., Zitoun, H., Blay-Fornarino, M., Collet, P., & Renevier-Gonin, P. (2012, November). Vers des interfaces graphiques flexibles de configuration. In Journée Lignes de Produits 2012.

[Varela-Vaca et al., 2019] Varela-Vaca, Á. J., Galindo, J. A., Ramos-Gutiérrez, B., Gómez-López, M. T., & Benavides, D. (2019, September). Process mining to unleash

variability management: discovering configuration workflows using logs. In Proceedings of the 23rd International Systems and Software Product Line Conference-Volume A (pp. 265-276).

[Vassiliadis et al., 2002] Vassiliadis, P., Simitsis, A., & Skiadopoulou, S. (2002, November). Conceptual modeling for ETL processes. In Proceedings of the 5th ACM international workshop on Data Warehousing and OLAP (pp. 14-21).

[Wang et al., 2018] Wang, J., Lin, M. F., Park, S., & Lee, P. S. (2018). Deformable conductors for human-machine interface. *Materials Today*, 21(5), 508-526.

[Weijters et al., 2006] Weijters, A. J. M. M., van Der Aalst, W. M., & De Medeiros, A. A. (2006). Process mining with the heuristics miner-algorithm. Technische Universiteit Eindhoven, Tech. Rep. WP, 166, 1-34.

[Whelan et al., 2016] E. WHELAN, R. TEIGLAND, E. VAAST, AND B. BUTLER, Expanding the horizons of digital social networks: Mixing big trace datasets with qualitative approaches, *Information and Organization*, 26 (2016), pp. 1–12.

[White et al., 2007] White, J., Schmidt, D. C., Wuchner, E., & Nechypurenko, A. (2007, September). Automating product-line variant selection for mobile devices. In 11th International Software Product Line Conference (SPLC 2007) (pp. 129-140). IEEE.

[Wieringa et Morali, 2012] Wieringa, R., Morali, A. (2012). “Technical action research as a validation method in information systems design science”. *Design science research in information systems. Advances in theory and practice*, 7286, pages 220-238. Springer Berlin Heidelberg.

[Wilhelm et al., 2008] Wilhelm, R., Engblom, J., Ermedahl, A., Holsti, N., Thesing, S., Whalley, D., ... & Stenström, P. (2008). The worst-case execution-time problem—overview of methods and survey of tools. *ACM Transactions on Embedded Computing Systems (TECS)*, 7(3), 1-53.

[Witten et Frank, 2002] Witten, I. H., & Frank, E. (2002). Data mining: practical machine learning tools and techniques with Java implementations. *Acm Sigmod Record*, 31(1), 76-77.

[Xiang et al., 2018] Xiang, Y., Zhou, Y., Zheng, Z., & Li, M. (2018). Configuring software product lines by combining many-objective optimization and SAT solvers. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 26(4), 14.

[Wohlin et al., 2012] Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., & Wesslén, A. (2012). *Experimentation in software engineering*. Springer Science & Business Media.

[Yanes et al., 2010] Yanes, N., Sassi, S. B., & Ghézala, H. H. B. (2010). Software Components Search Approaches in the Context of COTS-based Development. In *SEKE* (pp. 675-679).

[Yao et al., 2019] Yao, X., & Askin, R. (2019). Review of supply chain configuration and design decision-making for new products. *International Journal of Production Research*, 57(7), 2226-2246.

[Zhan et al., 2018] Zhan Z., Zhan Y., Huang M., Liu Y. (2018) Product Configuration based on Feature Model. In: Xhafa F., Patnaik S., Zomaya A. (eds) *Advances in Intelligent Systems and Interactive Applications. IISA 2017. Advances in Intelligent Systems and Computing*, vol 686. Springer, Cham.

[Zhang, 2013] Zhang, B., & Becker, M. (2013, January). Mining complex feature correlations from software product line configurations. In *Proceedings of the Seventh International Workshop on Variability Modelling of Software-intensive Systems* (pp. 1-7).

[Zhang et al., 2010] Zhang, G., Ye, H., & Lin, Y. (2010, November). Quality attributes assessment for feature-based product configuration in software product line. In *2010 Asia Pacific Software Engineering Conference* (pp. 137-146). IEEE.

[Cette page est intentionnellement laissée en blanc]

ANNEXES

Annexe A : Démonstration du moteur de recommandation ConfiLog avec une ligne de téléphonie basique

Une ligne de téléphonie basique peut être simplifiée comme le montre la figure ci-dessous. En effet, chaque téléphone contient obligatoirement une batterie, un Ecran, un CapteurRéseau et un LecteurSIM. L'EcranEnCouleurs, la CaméraHD, le lecteur DoubleSIM et la RésistanceEau sont optionnels. Les téléphones contenant une CaméraHD doivent avoir obligatoirement un EcranEnCouleurs.

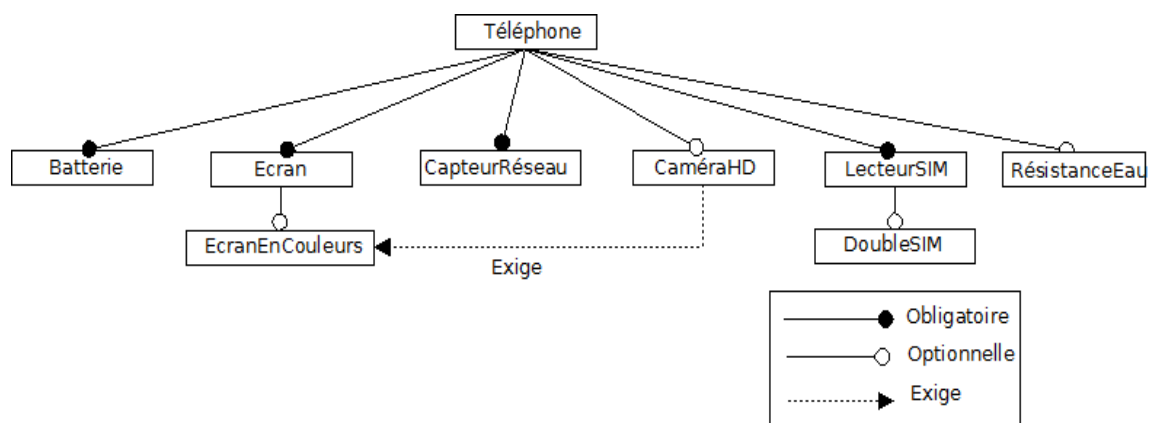


Figure (A): Annexes - Ligne de produits de téléphone basique

Pour la configuration d'un téléphone à partir du modèle ci-dessus, la méthode ConfiLog propage automatiquement les variantes obligatoires [Batterie, Ecran, CapteurRéseau, LecteurSIM] et laisse le choix à la partie prenante pour la sélection ou le refus des variantes suivantes : [EcranEnCouleurs, CameraHD, DoubleSIM, RésistanceEau].

La suite de cette section considère le scénario où la partie prenante introduit son but de guidage, qu'il aie un processus de configuration rapide $\tau = \text{Rapidité}$.

Après fouille de processus, le dataset de guidage \mathcal{D} est alimenté par un modèle de processus sélectionnant uniquement les instances de processus qui ont été configurées rapidement :

- Rapidité(1)=100 secondes: [EcranEnCouleurs-selected, CameraHD-deselected, RésistanceEau-selected]
- Rapidité(2)=200 secondes: [EcranEnCouleurs-deselected, CameraHD-selected, RésistanceEau-selected, DoubleSIM-deselected]

- Rapidité(3)=150 secondes: [EcranEnCouleurs-selected, RésistanceEau-deselected, DoubleSIM-selected]

La partie prenante suit un ensemble d' étapes de processus de configuration jusqu'à finir la configuration de son téléphone souhaité.

1. Etape 1 du processus de configuration de la ligne de téléphonie

La partie prenante sélectionne la variante EcranEnCouleurs. Alors la variable configuration partielle ρ et la variable activités autorisées ε sont mis à jour comme suit: ρ = EcranEnCouleurs-selected ; E = CameraHD, DoubleSIM, RésistanceEau.

Le calcul des prédicats en fonction des activités autorisées est détaillé dans le tableau (A). En effet, le tableau contient trois colonnes, une colonne pour chaque activité autorisée et trois lignes, une ligne pour chaque réponse de fouille enregistrée dans le dataset de guidage. Les "τ" signifient qu'il y a eu, effectivement, une décision de sélection ou de refus de cette activité autorisée dans la réponse de fouille en question. Les "F" signifient qu'aucune décision sur l'activité autorisée n'a eu lieu dans la réponse de fouille en question.

Tableau (A) : Annexes - Calcul des prédicats pour étape 1

	CameraHD	RésistanceEau	DoubleSIM
Rapidité(1)=100 secondes	T	T	F
Rapidité(2)=200 secondes	T	T	T
Rapidité(3)=150 secondes	F	T	T

Dans ce cas, tous les processus sont à pondérer parce qu'ils contiennent au moins une activité autorisée τ.

Le calcul de pondération des processus s'effectue comme suit :

Pondération Rapidité(1):

$$\frac{|[EcranEnCouleurs-selected] \cap [EcranEnCouleurs-selected, CameraHD-deselected, RésistanceEau-selected]|}{|[EcranEnCouleurs-selected]|}$$

$$= \frac{|[EcranEnCouleurs-selected]|}{|[EcranEnCouleurs-selected]|} = \frac{1}{1} = 1$$

Pondération Rapidité (2):

$$\frac{|[EcranEnCouleurs-selected] \cap [EcranEnCouleurs-deselected, CameraHD-selected, RésistanceEau-selected, DoubleSIM-deselected]|}{|[EcranEnCouleurs-selected]|}$$

$$= \frac{||}{|[EcranEnCouleurs-selected]|} = \frac{0}{1} = 0$$

Pondération Rapidité (3):

$$\frac{|[EcranEnCouleurs-selected] \cap [EcranEnCouleurs-selected, RésistanceEau-deselected, DoubleSIM-selected]|}{|[EcranEnCouleurs-selected]|}$$

$$= \frac{|[EcranEnCouleurs-selected]|}{|[EcranEnCouleurs-selected]|} = \frac{1}{1} = 1$$

La valeur cible attendue lors de la configuration de chaque activité autorisée e s'effectue comme suit : Do ((CameraHD, RésistanceEau, DoubleSIM), Rapidité, (EcranEnCouleurs-selected), Dataset). Pour la variante CameraHD, l'algorithme ne considère pas la réponse de fouille Rapidité (3), parce qu'elle ne figure pas parmi ses variantes (cf. tableau 11). Pareillement, pour la variante DoubleSIM, l'algorithme ne considère pas la réponse de fouille Rapidité (1), parce qu'elle ne figure pas parmi ses variantes.

Configuration de la variante CameraHD :

$$\text{Do (CameraHD)} = (1 * 100 \text{ secondes}) + (0 * 200 \text{ secondes}) = 100$$

Configuration de la variante RésistanceEau :

$$\text{Do (RésistanceEau)} = (1*100 \text{ secondes}) + (0 * 200 \text{ secondes}) + (1 * 150 \text{ secondes})= 250$$

Configuration de la variante DoubleSIM :

$$\text{Do (DoubleSIM)}= (0 * 200 \text{ secondes}) + (1 * 150 \text{ secondes}) = 150$$

La valeur cible attendue lors de la NON configuration de chaque activité autorisée s'effectue comme suit : Dont ((CameraHD, RésistanceEau, DoubleSIM), Rapidité, (EcranEnCouleurs-selected) ,D). En effet, l'algorithme de ConfiLog ne considère que les prédicats F pour chaque réponse de fouille. Par exemple, pour le calcul de la NON configuration de la variante CameraHD, seule la réponse de fouille Rapidité(3) sera considérée (cf. tableau 11).

NON configuration de la variante CameraHD :

$$\text{Dont (CameraHD)}= (1*150 \text{ secondes}) = 150$$

NON configuration de la variante RésistanceEau :
 Dont (RésistanceEau) = 0

NON configuration de la variante DoubleSIM :
 Dont(DoubleSIM) = (1*100 secondes) = 100

Le calcul de la différence entre les fonctions do et dont s'effectue comme suit :

La différence do et dont pour la variante CameraHD :
 (CameraHD, 100, 150) = 50

La différence do et dont pour la variante RésistanceEau :
 (RésistanceEau, 250, 0) = 250

La différence do et dont pour la variante DoubleSIM :
 (DoubleSIM, 150, 100) = 50

Pour l'étape 1, l'algorithme ConfiLog recommande les variantes ayant le résultat de soustraction [do - dont] le plus élevé en premier. En effet, l'ordre de processus recommandé pour cette étape de configuration est le suivant : (Select: RésistanceEau < Deselect: CameraHD ≤ Select: DoubleSIM)

V.3.5.2. Etape 2 du processus de configuration de la ligne de téléphonie

La partie prenante ne prend pas en compte le guidage fourni par ConfiLog et rejette la variante cameraHD. Alors la variable configuration partielle p et la variable activités autorisées E sont mis à jour comme suit : p = EcranEnCouleurs-selected, CameraHD-deselected ; E = RésistanceEau, DoubleSIM.

Le calcul des prédicats en fonction des activités autorisées est détaillé dans tableau (B). En effet, le tableau contient une colonne de moins qui présente la variante cameraHD qui vient d'être rejetée par la partie prenante. Le reste des valeurs du tableau restent inchangées par rapport au tableau (A).

Tableau (B) : Annexes - Calcul des prédicats pour étape 2

	RésistanceEau	DoubleSIM
Rapidité(1)=100 secondes	T	F

Rapidité(2)=200 secondes	T	T
Rapidité(3)=150 secondes	T	T

Dans ce cas, les trois réponses de fouille sont à tenir en compte parce qu'elles contiennent au moins une activité autorisée.

Le calcul de pondération des processus s'effectue comme suit :

Pondération Rapidité(1) :

$$\frac{|[EcranEnCouleurs-selected, CameraHD-deselected] \cap [EcranEnCouleurs-selected, CameraHD-deselected, RésistanceEau-selected]|}{|[EcranEnCouleurs-selected, CameraHD-deselected]|} = \frac{|[EcranEnCouleurs-selected, CameraHD-deselected]|}{|[EcranEnCouleurs-selected, CameraHD-deselected]|} = \frac{2}{2} = 1$$

Pondération Rapidité(2) :

$$\frac{|[EcranEnCouleurs-selected, CameraHD-deselected] \cap [EcranEnCouleurs-deselected, CameraHD-selected, RésistanceEau-selected, DoubleSIM-deselected]|}{|[EcranEnCouleurs-selected, CameraHD-deselected]|} = \frac{|[EcranEnCouleurs-deselected, CameraHD-selected, RésistanceEau-selected, DoubleSIM-deselected]|}{|[EcranEnCouleurs-selected, CameraHD-deselected]|} = \frac{0}{2} = 0$$

Pondération Rapidité(3) :

$$\frac{|[EcranEnCouleurs-selected, CameraHD-deselected] \cap [EcranEnCouleurs-selected, RésistanceEau-deselected, DoubleSIM-deselected]|}{|[EcranEnCouleurs-selected, CameraHD-deselected]|} = \frac{|[EcranEnCouleurs-selected]|}{|[EcranEnCouleurs-selected, CameraHD-deselected]|} = \frac{1}{2} = 0.5$$

La valeur cible attendue lors de la configuration de chaque activité autorisée s'effectue comme suit : Do ((RésistanceEau, DoubleSIM), Rapidité, (EcranEnCouleurs-selected, CameraHD-deselected), D)

Configuration de la variante RésistanceEau :

$$\text{Do (RésistanceEau)} = (1 * 100 \text{ secondes}) + (0 * 200 \text{ secondes}) + (0.5 * 150 \text{ secondes}) = 175$$

Configuration de la variante DoubleSIM :

$$\text{do(DoubleSIM)} = (0 * 200 \text{ secondes}) + (0.5 * 150 \text{ secondes}) = 75$$

La valeur cible attendue lors de la NON configuration de chaque activité autorisée s'effectue comme suit : Dont (RésistanceEau, DoubleSIM), Rapidité, (EcranEnCouleurs-selected, CameraHD-deselected), D)

NON configuration de la variante RésistanceEau =

$$\text{Dont (RésistanceEau)} = 0$$

NON configuration de la variante DoubleSIM =
Dont (DoubleSIM) = (1 * 100 secondes) = 100

Le calcul de la différence entre la fonction d_o et d_{ont} s'effectue comme suit :

La différence d_o et d_{ont} pour la variante RésistanceEau :
(RésistanceEau, 175, 0) = 175

La différence d_o et d_{ont} pour la variante DoubleSIM :
(DoubleSIM, 75, 100) = 25

Pour l'étape 2, l'algorithme ConfiLog recommande les variantes ayant le résultat de soustraction [$d_o - d_{ont}$]. En effet, l'ordre de processus recommandé pour cette étape de configuration est le suivant : (Select: RésistanceEau < Select: DoubleSIM)

La partie prenante fait son choix sur la variante RésistanceEau, alors pour l'étape 3 de la configuration il ne reste qu'une variante à recommander. La partie prenante la configure et finalise le processus de configuration.

Annexe B : Pré-Questionnaire

Questionnaire de pré-expérimentation

Avant de démarrer l'expérimentation, nous vous invitons à consacrer quelques instants pour répondre à ce questionnaire et nous fournir des informations concernant votre background, attentes et profil. Les informations peuvent être interprétées et analysées après.

Le questionnaire est divisé en sections. Chaque section comporte des questions de la même nature. Chaque question a également un identifiant (pour repérer facilement). Chaque réponse est évaluée selon l'échelle de Likert : 1 (Fortement désaccord), 2 (Partiellement désaccord), 3 (Ni d'accord ni désaccord), 4 (Partiellement d'accord), 5 (Fortement d'accord).

Pseudonyme: _____

Questionnaire

Background. J'ai une expérience considérable dans...

	1	2	3	4	5
... le monde du cyclisme.					
... les boutiques en ligne des vélos					
... le monde des lignes de produits.					
... le processus de configuration des lignes de produits.					
... l'instanciation des modèles interdépendants.					

Attente. Je m'attends à ce que...

	1	2	3	4	5
... le processus de configuration soit facile.					
... le processus de configuration ne dépasse pas les 10 minutes.					
... le processus de configuration soit flexible.					
... le processus de configuration soit guidé.					

J'ai une idée sur mon vélo souhaité					
Sexe : _____ Age : _____					
Merci pour votre participation.					

Annexe B bis: Rendus des étudiants pour le Pré-Questionnaire

Q1:J'ai une expérience considérable dans le monde de cyclisme				
Note: 1/5	Note: 2/5	Note: 3/5	Note: 4/5	Note: 5/5
10,1%	9,4%	19,4%	48,2%	12,9%

Q2:J'ai une expérience considérable dans les boutiques en ligne de vélos				
Note: 1/5	Note: 2/5	Note: 3/5	Note: 4/5	Note: 5/5
17,3%	20,1%	30,2%	28,8%	3,6%

Q3:J'ai une expérience considérable dans le monde des lignes de produits.				
Note: 1/5	Note: 2/5	Note: 3/5	Note: 4/5	Note: 5/5
33,8%	25,2%	13,7%	22,3%	5%

Q4:J'ai une expérience considérable dans le processus de configuration				
Note: 1/5	Note: 2/5	Note: 3/5	Note: 4/5	Note: 5/5
34,5%	25,9%	12,2%	23%	4,3%

Q5:J'ai une expérience considérable dans l'instanciation des modèles indépendants				
Note: 1/5	Note: 2/5	Note: 3/5	Note: 4/5	Note: 5/5
43,2%	23%	9,4%	18%	6,5%

Q6:Je m'attends à ce que le processus de configuration soit facile				
Note: 1/5	Note: 2/5	Note: 3/5	Note: 4/5	Note: 5/5
1,4%	1,4%	7,2%	30,2%	59,7%

Q7:Je m'attends à ce que le processus de configuration ne dépasse pas les 10 minutes				
Note: 1/5	Note: 2/5	Note: 3/5	Note: 4/5	Note: 5/5
3,6%	1,4%	32,4%	38,8%	23,7%

Q8:Je m'attends à ce que le processus de configuration soit flexible				
Note: 1/5	Note: 2/5	Note: 3/5	Note: 4/5	Note: 5/5
1,4%	0%	5%	28,8%	64,7%

Q9:Je m'attends à ce que le processus de configuration soit guidé				
Note: 1/5	Note: 2/5	Note: 3/5	Note: 4/5	Note: 5/5
0%	0,7%	3,6%	44,6%	51,1%

Q10:J'ai une idée de mon vélo souhaité				
Note: 1/5	Note: 2/5	Note: 3/5	Note: 4/5	Note: 5/5
6,5%	3,6%	12,9%	18%	59%

Sexe		
Homme	Femme	Autre
58,3%	41,7%	0%

Annexe C : Post-Questionnaire

Questionnaire de post-expérimentation

Merci pour votre participation dans l'expérimentation. Nous vous invitons à prendre une petite pause et répondre après à notre bref questionnaire qui va vous prendre quelques instants. Chaque question est liée à des problèmes concernant votre perception pendant l'expérimentation.

Le questionnaire est divisé en sections. Chaque section comporte des questions de la même nature. Chaque réponse est évaluée selon l'échelle de Likert: 1 (Fortement désaccord), 2 (Partiellement désaccord), 3 (Ni d'accord ni désaccord), 4 (Partiellement d'accord), 5 (Fortement d'accord).

Pseudonyme: _____

Questionnaire

Facteurs externes et menaces à la validité

	1	2	3	4	5
J'ai été assisté par un ami durant le processus de configuration.					
Le configurateur a planté au moins une fois.					
Le configurateur prend beaucoup de temps pour affecter les choix.					
Je n'étais pas prêt pour participer à l'expérimentation.					

Mesure de satisfaction

	1	2	3	4	5
Je trouve que le processus de configuration était monotone.					
Je me suis senti motivé durant le processus de configuration.					
Je vais recommander cette expérimentation à des amis.					

Qualité du guidage	1	2	3	4	5
Il m'était facile de tomber sur des choix de configuration qui sont cohérents.					
Les contraintes entre les choix ont rendu le processus de configuration sujet à l'erreur.					
Je m'attendais à plus de guidage à fournir par le configurateur.					
J'ai suivi le guidage fourni par le configurateur.					
J'ai trouvé pertinent le guidage fourni par le configurateur.					

Avez-vous des suggestions à nous communiquer pour améliorer la qualité du guidage de notre configurateur? :

Merci pour votre participation.

Annexe C bis: Rendus des étudiants pour le Post-Questionnaire

Q1:J'ai été assisté par un ami durant la configuration[Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
84.62%	71.68%	7.69%	18.58%	3.85%	2.65%	3.85%	2.65%	0%	4.42%

Q2:Le configurateur a planté au moins une fois pendant la configuration[Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
0%	69.46%	0%	3.54%	15.38%	0%	73.08%	0%	11.54%	0%

Q3:Le configurateur prend beaucoup de temps pour affecter les choix [Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
3.85%	80.53%	0%	14.16%	0%	1.77%	50%	2.65%	46.15%	0.88%

Q4:Je n'étais pas prêt pour participer à l'expérimentation[Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
73.08%	44.25%	7.69%	4.42%	0%	36.28%	7.69%	7.96%	11.54%	7.08%

Q5:Je trouve que la configuration était monotone[Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
0%	43.36%	0%	17.7%	0%	21.24%	15.38%	15.04%	84.62%	2.65%

Q6:Je me suis senti motivé durant la configuration [Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
57.69%	0%	26.92%	1.77%	7.69%	5.31%	0%	53.98%	7.69%	38.94%

Q7:Je vais recommander cette configuration a des amis [Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
61.54%	0.88%	26.92%	2.65%	0%	14.16%	3.85%	50.44%	7.69%	31.86%

Q8:Il m'était facile de tomber sur des choix qui sont cohérents [Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
26.92%	0.88%	11.54%	0%	23.08%	4.42%	11.54%	21.24%	26.92%	73.45%

Q9:Les contraintes entre les choix induisent des erreurs [Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
11.64%	83.19%	7.69%	6.19%	3.85%	3.54%	38.46%	3.54%	38.46%	3.54%

Q10:Je m'attendais à plus de guidage par le configurateur [Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
0%	14.16%	0%	14.16%	0%	29.02%	3.85%	23.89%	96.15%	18.58%

Q11:J'ai suivi le guidage fourni par le configurateur[Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
--	--	--	--	--	--	--	--	--	--

Annexes

Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
73.08%	1.77%	19.23%	1.77%	3.85%	3.54%	0%	53.1%	3.85%	39.82%

Q12:J'ai trouvé pertinent le guidage fourni par le configurateur [Guidage basé sur la connaissance Guidage basé sur la fouille de processus]									
Note: 1/5		Note: 2/5		Note: 3/5		Note: 4/5		Note: 5/5	
76.92%	0.88%	7.69%	0%	0%	5.31%	0%	34.51%	15.38%	59.29%

Annexe D : Modélisation de la ligne de Vélos dans ConfiLog

La modélisation de la ligne de vélos est présentée ci-dessous. La spécification de cette ligne a été discutée plusieurs fois avec des étudiants.

```
<ProductLineModel name="Conf">
<meta>
<data name="description">Bike model</data>
<data name="creator">Housseem CHEMINGUI</data>
<data name="address"></data>
<data name="email">houss.chemingui@gmail.com</data>
<data name="phone"></data>
<data name="website"></data>
<data name="organization">ENSI Manouba</data>
<data name="department"></data>
<data name="date"></data>
<data name="reference"></data>
</meta>
<Variant_tree>
:r Bike(_r)
  :m SupportedAge(_r_1)
    :g (_r_1_118) [1,1]
      : From4To9(_r_1_118_119)
      : From10To19(_r_1_118_121)
      : From20To40(_r_1_118_122)
      : From41To60(_r_1_118_123)
      : MoreThan60(_r_1_118_124)
  :m SupportedGender(_r_9)
    :g (_r_9_10) [1,1]
      : Male(_r_9_10_11)
      : Female(_r_9_10_12)
  :o SupportedHeight(_r_14)
    :g (_r_14_15) [1,1]
      : LessThan100cm(_r_14_15_16)
      : From100cmTo130cm(_r_14_15_17)
      : From130cmTo160cm(_r_14_15_18)
      : From160cmTo199cm(_r_14_15_19)
      : MoreThan200cm(_r_14_15_20)
  :o SupportedWeight(_r_54)
    :g (_r_54_55) [1,1]
      : LessThan14Kg(_r_54_55_56)
      : From15KgTo30Kg(_r_54_55_57)
      : From31KgTo60Kg(_r_54_55_58)
      : From61KgTo90Kg(_r_54_55_59)
      : From91KgTo120Kg(_r_54_55_60)
      : MoreThan121KG(_r_54_55_61)
```

```
:m Budget(_r_21)
  :g (_r_21_22) [1,1]
    : LessThan500Euros(_r_21_22_23)
    : From501EurosTo999Euros(_r_21_22_24)
    : From1000EurosTo1999Euros(_r_21_22_25)
    : MoreThan2000(_r_21_22_26)
:m RiderPorfile(_r_27)
  :g (_r_27_28) [1,1]
    : Dabster(_r_27_28_29)
    : Professional(_r_27_28_30)
    : Passionned(_r_27_28_31)
:m Usage(_r_109)
  :g (_r_109_110) [1,1]
    : Work(_r_109_110_111)
    : Urban(_r_109_110_112)
    : Racing(_r_109_110_114)
    : Mountain(_r_109_110_115)
:m FrameDesign(_r_81)
  :g (_r_81_82) [1,1]
    : Triathlon(_r_81_82_83)
    : StepThrough(_r_81_82_84)
    : Diamond(_r_81_82_85)
:m FrameSet(_r_91)
  :g (_r_91_92) [1,1]
    : XLarge(_r_91_92_93)
    : Large(_r_91_92_94)
    : Medium(_r_91_92_95)
    : Small(_r_91_92_96)
:m FrameType(_r_97)
  :g (_r_97_100) [1,1]
    : Iron(_r_97_100_101)
    : Aluminium(_r_97_100_102)
    : Carbon(_r_97_100_103)
:m FrameColor(_r_104)
  :g (_r_104_105) [1,*]
    : White(_r_104_105_106)
    : Blue(_r_104_105_107)
    : Red(_r_104_105_108)
:m Fork(_r_120)
  :g (_r_120_121) [1,1]
    : Suspension(_r_120_121_122)
    : Static(_r_120_121_123)
:m Saddle(_r_116)
  :g (_r_116_117) [1,1]
    : Plastic(_r_116_117_118)
    : Gel(_r_116_117_119)
```

```
:m SeatPost(_r_124)
  :g (_r_124_125) [1,1]
    : CarbonPost(_r_124_125_126)
    : IronPost(_r_124_125_127)
:m HandleBars(_r_128)
  :g (_r_128_129) [1,1]
    : Drop(_r_128_129_130)
    : Riser(_r_128_129_131)
    : Pursuit(_r_128_129_132)
:m WheelsSize(_r_32)
  :g (_r_32_33) [1,1]
    : 24Inch(_r_32_33_34)
    : 26Inch(_r_32_33_35)
    : 27Inch(_r_32_33_36)
    : 29Inch(_r_32_33_37)
    : 20Inch(_r_32_33_38)
    : 15Inch(_r_32_33_39)
:o WheelsKind(_r_44)
  :g (_r_44_45) [1,1]
    : AllYear(_r_44_45_46)
    : Summer(_r_44_45_47)
    : Winter(_r_44_45_48)
:o Light(_r_40)
  :g (_r_40_41) [1,1]
    : LedBlue(_r_40_41_42)
    : LedYellow(_r_40_41_43)
:o EnergySource(_r_62)
  :g (_r_62_63) [1,1]
    : SolarPanel(_r_62_63_64)
    : Dynamo(_r_62_63_65)
    : Battery(_r_62_63_66)
:m Running(_r_71)
  :g (_r_71_74) [1,*]
    : FootPedal(_r_71_74_75)
    : Engine(_r_71_74_76)
:o Application(_r_49)
  :o GoogleMaps(_r_49_50)
  :o Navigator(_r_49_51)
  :o Guider(_r_49_52)
  :o Music(_r_49_53)
:o CompUnit(_r_77)
  :g (_r_77_78) [1,1]
    : GPS(_r_77_78_79)
    : Tablet(_r_77_78_80)
:o Basket(_r_67)
  :g (_r_67_68) [1,1]
    : Wooden(_r_67_68_69)
```

```
        : SteelWires(_r_67_68_70)
:m MaintenancePlan(_r_86)
    :g (_r_86_87) [1,1]
        : 2years(_r_86_87_88)
        : 3years(_r_86_87_89)
        : LifeTime(_r_86_87_90)
:o Accessories(_r_139)
    :g (_r_139_140) [1,*]
        : Mudguards(_r_139_140_141)
        : AirPump(_r_139_140_142)
        : KaskRoad(_r_139_140_143)
        : BikeCovers(_r_139_140_144)
        : WaterBottles(_r_139_140_145)
        : SmallPannier(_r_139_140_146)
        : BigPannier(_r_139_140_147)
:o TheftPreventions(_r_133)
    :g (_r_133_134) [1,*]
        : AmovableSeat(_r_133_134_135)
        : Lock(_r_133_134_136)
        : AlarmSystem(_r_133_134_137)
        : GeolocalisationSystem(_r_133_134_138)
</Variant_tree>
<constraints>
constraint_1:~_r_77_78_79 or _r_133_134_138
constraint_8:_r_62_63_66 or ~_r_77
constraint_9:~_r_32_33_39 or ~_r_91_92_93
constraint_10:~_r_32_33_37 or ~_r_91_92_96
constraint_6:~_r_67 or ~_r_109_110_114
constraint_11:~_r_21_22_23 or ~_r_133
constraint_7:~_r_71_74_76 or ~_r_109_110_114
constraint_12:~_r_21_22_23 or ~_r_139
constraint_13:~_r_1_118_119 or ~_r_128_129_130
constraint_4:~_r_40 or _r_62
constraint_14:~_r_1_118_119 or _r_32_33_39
constraint_5:~_r_67 or ~_r_109_110_115
constraint_15:~_r_14_15_20 or _r_91_92_93
constraint_2:~_r_109_110_115 or _r_120_121_122
constraint_16:~_r_14_15_16 or _r_91_92_96
constraint_3:_r_97_100_103 or ~_r_124_125_126
</constraints>
</ProductLineModel>
```

[Cette page est intentionnellement laissée en blanc]

[Cette page est intentionnellement laissée en blanc]