HAL
open science

Transformation de modèles de lignes de produits en environnement hétérogène

Farah Maamar

To cite this version:

Farah Maamar. Transformation de modèles de lignes de produits en environnement hétérogène. INFormatique des ORganisation et Systèmes d'Information de Décision, May 2014, LYON, France. http://www.irit.fr/ Guillaume.Cabanac/docs/fjc2014/fjc2014actes.pdf#page=45. hal-01001777

HAL Id: hal-01001777

https://paris1.hal.science/hal-01001777v1

Submitted on 4 Jun 2014

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.

Transformation de modèles de lignes de produits en environnement hétérogène

Farah Maamar

Centre de Recherche en Informatique, Université Paris 1 Panthéon-Sorbonne
90, rue de Tolbiac
75013 Paris

farah.maamar@malix.univ-paris1.fr

MOTS-CLÉS : *Familles de produits, Lignes de produits, interopérabilité, Transformation, l'ingénierie des exigences, le modèle pivot.*

KEYWORDS: *Software Product Lines, interoperability, Transformations, Requirement engineering, Pivot Model.*

ENCADREMENT : *Camille Salinesi (PR) et Raûl Mazo peña (MCF)*

1. Contexte

Le domaine de l'ingénierie de ligne de produits est en cours de développement, dans un monde où la concurrence est de plus en plus rude et où les entreprises sont de plus en plus amenées à diversifier leurs gammes de produits. Les entreprises les plus compétitives et innovantes se dirigent vers ce nouveau paradigme, afin d'augmenter leurs productivités et diminuer le temps de réalisation du produit. L'identification des points variables et communs d'une ligne de produits dans un modèle est une approche clé dans l'ingénierie des lignes de produits et plus particulièrement pour la gestion d'éléments commun et variable. Le processus de développement d'une ligne de produits passe par plusieurs phases, chaque phase est réalisée par un outil différent, si bien que des incohérences peuvent avoir lieu. Pour assurer la bonne collaboration et la cohérence de ces outils, il est nécessaire de faire face aux problèmes d'interopérabilité de ces outils. Dans cet article nous considérons que la transformation du modèle est l'une des solutions clés d'interopérabilité.

2. État de l'art

Dans la littérature, nous trouvons beaucoup de travaux en rapport avec l'interopérabilité. Nous pensons que dans le cadre de l'ingénierie dirigée par les modèles, la transformation de modèle peut être un moyen pour résoudre le problème

d'interopérabilité. L'IEEE Standard Computer Dictionary définit l'interopérabilité comme étant la capacité de deux ou plusieurs systèmes ou composants de pouvoir échanger de l'information et de pouvoir l'utiliser. Dans le contexte de transformation de modèles, nous soulignions que les systèmes qui nous intéressent sont des outils d'ingénierie de modèles de ligne de produits, et dans ce contexte, une définition précise pour l'interopérabilité a été proposée par (Jouault et al., 2007), l'interopérabilité est la capacité d'exécuter des programmes écrits dans un langage avec des outils conçus pour un autre langage, selon un consensus opérationnel et sémantique, par exemple: exécuter un code ATL avec l'outil QVT-compliant engine.

Dans (Mazo et al., 2011), un travail qui vise à soulever le problème de configuration du modèle de ligne de produits, ils ont pensé à automatiser le processus de raisonnement sur les caractéristiques de modèle. L'approche de transformation qu'ils proposent consiste à appliquer un ensemble de règles de transformation sur 60 modèles (FM) de 2000 caractéristiques, ils ont mis au point pour cela deux stratégies complètement automatisées dans leurs outils VARIAMOS.

(Levendovszky et al., 2002) définit le mapping comme étant un ensemble de règles de transformation de modèles permettant de traduire des instances d'un méta-modèle source en instance d'un méta-modèle cible, le travail de (Baîna et al., 2006) a confirmé qu'il existe une forte corrélation entre l'interopérabilité et le mapping de modèles et a proposé la définition suivante : Soit A et B deux méta-modèles, A et B sont dits interopérables si et seulement s'il existe un mapping bijectif de M_A vers M_B, i.e. qu'il est possible de créer une instance de modèle A à partir de l'instanciation de modèle B, et vice-versa. À partir de cette définition, il a classé trois niveaux de mapping et par conséquent trois niveaux d'interopérabilité.

3. Problématique

Il existe plusieurs outils de modélisation de MLPs et d'autres outils pour les analyser et les configurer, chaque outil a son propre formalisme dont lequel il présente le MPLs, après la modélisation, il est nécessaire d'analyser et configurer le MLPs, i.e. la gestion d'éléments commun et variable. C'est pour cette raison que c'est important qu'un formalisme de modèle donné soit convertible en d'autres formalismes utilisés par les outils d'analyses, i.e. comme les solveurs connus tels que GnuProlog, SAT, BDD et bien d'autres. Dans ce contexte de transformation de modèles, nous allons préparer un environnement propice pour la collaboration et l'interopérabilité entre ces différents outils d'ingénierie de ligne de produits. Certaines questions d'interopérabilité nécessitent l'étude et la vérification:

- Comment assurer l'interopérabilité entre les outils de modélisation et les outils de configurations ?

Il existe quelques approches d'interopérabilité dans le contexte de transformation de modèle qui sont citées dans la section d'état de l'art, et d'autres comme l'utilisation d'annotations sémantiques basées sur les ontologies, ce qui nous amène à poser la question suivante :

- Quelle démarche est la plus adéquate pour assurer l'interopérabilité dans le contexte de transformation des modèles ?

4. Actions réalisées

Dans le cadre de notre travail, il est difficile de se positionner dans les niveaux définis par (Baîna et al., 2006) dans la section 2, car d'un côté nous avons traité une transformation dans un seul sens et de l'autre côté tout concept du modèle **M**a a son équivalent dans le modèle **M**b, mais pas en isomorphisme, en effet, le concept du modèle pivot que nous employons pour la transformation, est un modèle général, qui doit contenir tous les éléments de modèle cible donc chaque concept du modèle source doit avoir son équivalent dans le modèle cible. Dans le contexte de notre travail, «Le modèle Pivot» est un méta-modèle du modèle cible, ce méta-modèle doit être le plus complet possible en contenant tous les éléments nécessaires de modèle cible. Pour but d'obtenir la description la plus exhaustive possible. Le modèle source qui va être transformé, doit être conforme au méta-modèle (modèle Pivot). Ceci implique donc la création d'une couche d'abstraction supplémentaire qui se traduit par l'ajout d'une autre étape de transformation en l'occurrence celle du modèle pivot vers le modèle cible. L'intérêt de ce modèle réside dans son indépendance de tout langage de modèle source. Le modèle pivot est le modèle qui est au centre de la transformation, de telle sorte que ce dernier permet de transformer tout formalisme du modèle source en modèle cible décrit dans le modèle pivot (méta-modèle).

Nous avons présenté dans cet article une solution d'interopérabilité par transformation de MLPs en utilisant le modèle Pivot, le cadre applicatif de notre approche de transformation de modèle peut être défini mathématiquement de la façon suivante :

$$\textbf{(1)}\begin{cases} (F_1\,(MS\mathbf{1}) = X\mathbf{1} \wedge P_{mmc}\,(X\mathbf{1}))=MC \\ (F_2\,(MS\mathbf{2}) = X\mathbf{2} \wedge P_{mmc}\,(X\mathbf{2}))=MC \\ (F_3\,(MS\mathbf{3}) = X\mathbf{3} \wedge P_{mmc}\,(X\mathbf{3}))=MC \end{cases} \Leftrightarrow (F_i(MS_i)=X_i \wedge P_{mmc}(X_i))=MC,\ i/i\in\mathbb{N}$$

Nous illustrons le modèle (**1**) par un schéma qui décrit la relation entre les fonctions et leurs enchainements comme suit :

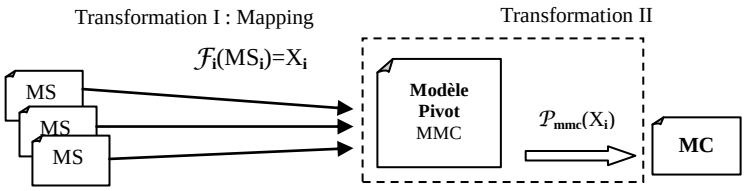

Figure 1 : processus de transformation de MLPs

Les différents formalismes de modèle source sont notés MSi/i∈ℕ, les différents types de modèle cible du même formalisme sont notés MC, les règles de première transformation sont notées F, les différents formalismes de MLPs sont notés i/i∈ℕ, le méta-modèle de modèle cible est noté mmc.

L'approche de transformation est constitué de deux étapes de transformation : une transformation du modèle source en modèle Pivot (transformation I) et du modèle Pivot vers le modèle cible (transformation II).L'application du modèle Pivot nécessite une seule transformation $\mathcal{P}_{\mathbf{mmc}}(X_{\mathbf{i}})$ en modèle cible MC et donc un seul ensemble de règles de transformations applicables pour toute transformation de $MS_{\mathbf{i/i\in N}}$, car le modèle Pivot joue le rôle du modèle intermédiaire vers le même formalisme de modèle cible MC. En effet, la nécessité d'avoir une interopérabilité plus flexible et générale nous a encouragés à penser à une solution plus ouverte,

5. Actions futures

Dans de futurs travaux, nous allons essayer de faire face à la même problématique de l'interopérabilité mais au niveau sémantique, pour cela, i) nous allons tester le mapping bijectif, ii) Nous allons proposer aussi un mécanisme pour détecter la perte d'information qu'elle soit sémantique ou syntaxique. Il est évident que le modèle pivot ainsi défini, ne permet pas de couvrir totalement la sémantique de chaque modèle. Donc il est nécessaire d'enrichir ce modèle pivot par d'autres formalismes de modèle cible. Tester l'application de l'approche proposée sur des différents formalismes de MLPs, ça nous permettra d'améliorer notre approche et façonnera les décisions que nous aurons à prendre.

Bibliographie

Mazo R., Salinesi C., Diaz D., Alberto L-M., " Transforming attribute and clone enabled feature modèles into constraint programs over finit domains ", September 2011, ENASE 2011: Beijing, China, p.188-199.

Levendovszky T., Karsai1 G., Maroti1 M., Ledeczi1 A., Charaf H., " Model Reuse with Metamodel-Based Transformations ", in Proceedings of the 7th International Conference on Software Reuse: Methods, Techniques , and Tools, p. 166 -178, 2002.

Baîna S., interopérabilité dirigé par les model, thèse, Université Henri Poincaré, Nancy I, 2006.

Jouault F., Kurtev I., "On the interoperability of model-to-model transformation languages", ATLAS Group, INRIA and LINA, University of Nantes, France, 14 May 2007.