

Visualisation de composition de Modèles

Sujet de stage de Master 2 Recherche spécialité Informatique
au Centre de Recherche Public - Gabriel Lippmann
Année universitaire 2012-2013

Contact:

Nicolas Biri

Centre de Recherche Public - Gabriel Lippmann
Département Informatique, Systèmes et Collaboration (ISC)
41, rue du Brill, L-4422 Belvaux, Luxembourg
<http://www.lippmann.lu> – biri@lippmann.lu

1. Présentation générale du sujet

L'ingénierie logicielle dirigée par les modèles (Model Driven software Development - MDSD) a pour but de produire tout (ou partie) d'un logiciel sur la base de modèles. Avec comme objectif de faciliter la maintenance, l'évolution et la production de logiciels. Toutefois, la complexité des applications actuelles mène à des modèles complexes et sont donc difficiles à maintenir. Afin de réduire cette complexité, une approche souvent utilisée est de séparer les préoccupations afin de définir des modèles plus petits et donc plus simples à maintenir. Le modèle final est alors obtenu par composition de ces modèles initiaux. Il existe toutefois plusieurs possibilités de compositions pour des modèles (par exemple, deux modèles de processus peuvent être mis en séquence ou en parallèle). Afin de choisir celle qui correspond au but de l'utilisateur et de valider le modèle final, il est donc utile de fournir des outils pour visualiser les différents résultats possibles pour la composition de différents modèles.

Le but de cette proposition de stage est d'étendre des outils de visualisation de graphes existants pour fournir un outil permettant de visualiser la composition de modèles. Il s'agit plus précisément d'adapter l'outil de réécriture de graphes Porgy [1] qui est une extension du logiciel Tulip [2] afin de pouvoir l'utiliser dans le contexte de la composition de modèles.

2. Travail demandé

- Représenter tout type de modèle dont le méta-modèle est défini selon le modèle EMOF [3] afin de le représenter sous forme de graphes. Cette phase, relativement simple, consiste à analyser un document XML pour construire un graphe représentant ce modèle.
- Proposer une extension de Tulip afin de pouvoir représenter des types de modèles particuliers sous leur notation standard à l'intérieur du logiciel. Ainsi, par exemple, un graphe représentant un diagramme de classe devra être visualisable dans Tulip sous forme d'un diagramme de classe et non d'un simple graphe.
- Implémenter des règles de composition de modèles (à la manière de celles décrites dans [4]) afin de pouvoir les visualiser, à la manière de ce qui est fait dans Porgy.

3. Références bibliographiques

[1] Porgy – <http://tulip.labri.fr/TulipDrupal/?q=porgy>

[2] Tulip – <http://tulip.labri.fr/TulipDrupal/>

[3] MOF Specification – <http://www.omg.org/spec/MOF/2.4.1/PDF/>

[4] Morin, B., Klein, J., Barais, O., Jézéquel, J.M. 2008. A generic weaver for supporting product lines. Proceedings of the 13th international workshop on Early Aspects, EA'08, 11-18.