

Sujet de master recherche « Architectures logicielles distribuées » 2006–2007**L'évolution logicielle par les modèles : application à l'environnement Eclipse**

Encadrant principal : Dalila Tamzalit – LINA

Courriel : Dalila.Tamzalit@univ-nantes.fr

tél. : 02 51 12 58 50

Co-encadrant(s) : Cédric Brun – Obeo

Courriel : cedric.brun@obeo.fr

Tél. : 02 51 13 51 42

Objectif du stage

L'évolution logicielle est une nécessité incontournable. Elle permet d'éviter que les architectures logicielles ne restent figées et soient obsolètes par rapport aux besoins en perpétuel changement. En effet, l'évolution permet non seulement d'allonger la durée de vie des systèmes mais également de prendre en compte de nouveaux besoins ou des fonctionnalités plus complexes. Une architecture doit donc pouvoir être modifiée pour rester utilisable, disponible et robuste auprès de ses utilisateurs, et cela tout au long de son cycle de vie.

La plupart des efforts de développement de systèmes logiciels sont investis dans leur évolution [1], [2]. Il n'est plus nécessaire d'arguer en faveur de la description architecturale de ces systèmes à différentes fins (compréhension, haut niveau d'abstraction, analyse et transformation). Cependant, les systèmes logiciels se complexifient tellement et sont tellement sujet à des adaptations, mises à jour et autres modifications, qu'il est primordial d'arriver à gérer correctement leur cycle d'évolution [3] [4]. Il est clair que les interventions ponctuelles leur font plus de tort, à terme, que de bien. L'évolution est une tâche ardue et complexe [5]. L'évolution doit être une discipline organisée et guidée. Elle est devenu, au demeurant, une discipline à part entière en Génie Logiciel, illustré par le nombre croissant d'ouvrages et de conférences qui lui sont entièrement dédiés [6].

Il est donc nécessaire de proposer une expertise de l'évolution, grâce à des moyens appropriés, notamment la nécessité de la représenter et de la mener à terme grâce à un haut niveau d'abstraction [7] et de réutilisation. Un des moyens les plus approprié est l'approche par les modèles [8].

Eclipse est une plate-forme Java open source dédiée pour le développement d'outils. La plate-forme est facilement étendue par de multiples plugs-ins astucieusement intégrés. Elle présente un certain nombre de concepts qui pourraient répondre à la problématique d'évolution abordée dans ce stage, et elle propose un moyen efficace d'utiliser l'approche par les modèles via EMF. La confrontation avec les moyens d'évolution apportés par Eclipse est nécessaire pour valider l'idée que la démarche s'installe à différents niveaux d'abstraction. L'utilisation d'une plateforme pragmatique consolidera également la base théorique des différentes stratégies d'évolution envisagées.

Contexte

Le contexte de ce stage s'inscrit dans un partenariat université (*équipe Modal*) -entreprise (*Obeo*) autour de la thématique de l'évolution logicielle dirigée par les modèles. Il prend en compte les spécificités et compétences de l'équipe Modal (<http://lina.atlanstic.net/fr/equipes/team8/index.html>) et d'Obeo, est une société nantaise spécialisée dans l'ingénierie des modèles appliquée à l'amélioration des techniques de conception logicielle et à la fiabilisation du cycle de vie des applications existantes (www.obeo.fr).

Travail à réaliser

Le travail représente une continuité d'un premier travail commun. L'objectif double est :

- d'« architecturer » l'évolution d'un système logiciel, c'est à dire de considérer l'évolution comme une architecture logicielle, avec ses concepts, ses opérations, son modèle.
- d'appliquer et d'utiliser ce modèle pour gérer l'évolution d'autres modèles/applications.

Ce travail sera validé sur l'évolution d'un méta-modèle avec la propagation et la gestion des impacts sur les modèles qui en sont issus et il sera implémenté sous Eclipse.

Mots-clés

Evolution – méta-modèle – modèle – architectures logicielles – IDM (Ingénierie Dirigées par les Modèles) – Eclipse

Références

- [1] I. Sommerville, *Software Engineering*. Boston, MA, USA: Addison- Wesley Longman Publishing Co., Inc., 1996.
- [2] S. Demeyer, S. Ducasse, and O. Nierstrasz, *Object-Oriented Reengineering Patterns*. Morgan Kaufmann, 2002.
- [3] M. Oussalah: génie objet, Analyse et conception de l'évolution, Hermes édition, 1999, ISBN 2-7462-0029-5.
- [4] N. Sadou, D. Tamzalit, and M. Oussalah. La problématique de l'évolution structurelle dans les «architectures logicielles a base de composants. » In 2 èmes journées sur l'Ingénierie Dirigée par les Modèles, pages 93–108, Lille, 2006.
- [5] S. Ajila. Software maintenance : An approach to impact analysis of objects change. *Software - Practice and Experience*, 25(10) :1155–1181, 1995.
- [6] I. Ivkovic and K. Kontogiannis. A framework for software architecture refactoring using model transformations and semantic annotations. In *Proceedings of the IEEE Conference on Software Maintenance and Reengineering*, 2006.
- [7] A. V. Deursen, C. Hofmeister, R. Koschke, L. Moonen, and C Riva. Symphony : View-driven software architecture reconstruction. In *WICSA*, pages 122–134, 2004.
- [8] T. Massoni, R. Gheyi, and P Borba. A model-driven approach to formal refactoring. In *OOPSLA '05 : Companion to the 20th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 124–125, New York, NY, USA, 2005. ACM Press.