

## Sujet de Thèse

### Adaptations Dynamiques aux Contextes et Utilisateurs d'Applications à Architectures Variables

Directeur : **Michel Riveill, Professeur des Universités**  
Co-encadreur : **Stéphane Laviotte, Maître de Conférences**

#### Préambule

Ce sujet de thèse s'inscrit dans la thématique générale de l'Informatique Ambiante (aussi dénommée « ubiquitaire » ou « pervasive ») et dans un projet labellisé (CONTINUUM) au sein du pôle SCS (Solutions Communicantes Sécurisées).

Ce sujet s'inscrit au cœur des travaux de l'équipe Rainbow de l'I3S (Université de Nice – Sophia Antipolis / CNRS) sur l'adaptation dynamique des applications à l'environnement d'exécution, en particulier dans le cadre des applications dynamiques construites par assemblage de composants logiciels et de composants matériels. Il fait suite à une première thèse financée par une Bourse BDI CNRS-Entreprise, avec une collaboration entre l'équipe Rainbow de l'I3S et le CSTB (Centre Scientifique et Technique du Bâtiment). La soutenance de cette thèse est prévue pour le premier semestre 2008. Une seconde thèse financée par une bourse CIFRE est en cours de réalisation avec une collaboration entre l'I3S et la société MobileGov.

#### Introduction

Les applications logicielles, maintenant distribuées, voire mobiles, nécessitent toujours plus de capacités d'adaptation face à une multitude de contextes d'utilisation (multi-dispositifs, multi-applications, multi-utilisateurs, dans un environnement physique variable). Ce constat est par exemple à l'origine de coûts additionnels importants pour le portage des applications dans le domaine des applications qui se développent pour l'habitat. Cela ne concerne pourtant que le problème de l'adaptation statique des applications à des configurations de cibles variables (installation d'une application domotique dans une habitation en fonction des différents matériels utilisables). Que dire alors de l'enjeu proposé par une informatique qui voudrait adapter dynamiquement une application logicielle à un environnement d'exécution découvert dynamiquement, évoluant tout aussi dynamiquement, et partiellement connue a priori.

Un des principaux enjeux, donc, dans le développement logiciel pour les applications en informatique mobile et ambiante repose sur l'*adaptation dynamique logicielle d'une application à son contexte*. Mais ce sujet est vaste tant dans la définition de contexte (Laviotte S. *et al.*, 2005) que dans les mécanismes d'adaptation logicielle aujourd'hui utilisables pour modifier le comportement d'une application en cours d'exécution.

#### Mécanismes pour les applications de l'informatique ambiante et l'adaptation logicielle

De nombreux systèmes ont été définis pour répondre aux problématiques de l'information ambiante. Ils reposent sur les paradigmes suivants :

- **orienté service** (Robinson J. *et al.*, 2007): favorisant la robustesse et une approche décentralisée de collaboration entre services indépendants,
- **orienté composants** (Szyperki C., 1999) : apportant la dynamique aux systèmes en autorisant le remplacement de parties du logiciels (ajout, retrait, mise à jour des composants, ...) lors de l'exécution,
- **orienté aspect** (Kiczales G., 1997): augmentant la modularité, apportant la séparation des préoccupations et la capacité à ajouter des fonctionnalités non prévues à l'origine, de manière transverse,
- et **à base de communications par événements** (Carzaniga A. *et al.*, 1998) : permettant la réactivité des systèmes, même si pour de gros systèmes, cela s'avère plus difficile à maîtriser.

Mais les systèmes actuellement mis en œuvre pour répondre aux problématiques de l'informatique ambiante n'intègrent pas l'ensemble de ces paradigmes. Nos travaux précédents dans ce domaine ont conduit à l'étude et au développement d'un modèle auto-adaptatif pour ce type d'applications : WComp (Cheung-Foo-Wo D. *et al.*, 2006).

Les travaux que nous avons réalisés à ce jour portent sur la définition des mécanismes permettant l'adaptation d'une application. Dans notre périmètre d'étude, nous définissons une application logicielle en tant qu'un ensemble de composants ou services indépendants (briques logicielles ou mixtes matériel/logiciel) et une collection d'aspects d'assemblages. La notion d'aspect d'assemblage (AA) (Cheung-Foo-Wo D. *et al*, 2007) est en relation directe avec le paradigme de programmation par aspects, mais appliquée aux applications développées à base de composants et de services logiciels, de manière non intrusive dans les composants et services. Les AA, créés sans connaissance a priori des assemblages de composants sur lesquels ils peuvent agir (principes d'indépendance et de séparation des domaines de préoccupation), ont la capacité d'apporter des modifications *transversales* à des assemblages de composants existants.

Les travaux actuels portent sur l'étude d'un modèle générique pour la composition/fusion des aspects d'assemblages et les outils de définition des règles de composition spécifiques à d'autres domaines d'application.

## Objectifs et description des travaux de la thèse proposée

### *Dimension contextuelle des Aspects d'Assemblages*

Le formalisme des aspects d'assemblage permet d'exprimer de manière générique et indépendante d'une application :

- l'endroit (point de jonction) où doit être appliqué la modification sur l'assemblage (point de coupe) (le « où »)
- la modification (qui peut être un changement comportementale, l'ajout d'une fonctionnalité, l'observation de nouveaux critères, etc.) qui doit être apportée à l'assemblage, spécifiée sous la forme d'un greffon exprimé sous la forme d'un sous-assemblage de composant (le « quoi »).

Le problème complémentaire à traiter, au cœur de la thèse proposée, est la définition et l'étude des modèles et mécanismes pour le déclenchement de ces adaptations. Les mécanismes de déclenchement des aspects d'assemblage ont été pour le moment traités par déclenchement externe (Tigli J.-Y., *et al*, 2006).

Toutefois dans le cadre des adaptations dynamiques, il est primordial de définir les conditions contextuelles (le « quand ») permettant de définir les conditions de déclenchement des adaptations. L'étude de cette dimension temporelle est totalement justifiée par la dynamique des applications ciblées (apparition et disparition de composants ou services dans l'environnement de l'habitat, changement du comportement de l'utilisateur). Ces conditions peuvent être de deux ordres : soit dirigées par le contexte donc issues de l'observation de l'environnement (habitation par exemple), ou bien déclenchées volontairement par l'utilisateur. Il conviendra de détecter, d'étudier les conflits d'assemblage issus des deux points de vue et les mécanismes pour leur résolution.

*Cette thèse a donc pour premier objectif d'étudier les différentes approches permettant d'augmenter l'expressivité des AA et d'enrichir le mécanisme de composition/fusion pour y intégrer notamment les informations qui permettront à l'AA de s'activer sous certaines conditions contextuelles. Une attention toute particulière sera alors portée sur les choix qui pourraient influencer les performances de l'application notamment son comportement temporelle au combien important pour les systèmes interactifs.*

### *Dimension « End-User Programming »*

La programmation orientée aspect, introduite par (Kiczales *et al*, 1997), est utilisée pour adapter des applications à leurs environnements pervasifs (Rashid, *et al*, 2004). Cette approche est particulièrement adaptée à la modification transversale des propriétés et du comportement d'une application. En ce sens il peut s'agir d'une approche efficace pour l'adaptation d'une application, mais celle-ci demeure basée sur un langage de programmation et reste destinée à des développeurs avertis.

La majorité des personnes exploite les moyens informatiques à leur disposition en tant qu'utilisateurs de programmes « prêt à l'emploi ». Malheureusement les auteurs de ces programmes ne peuvent connaître l'ensemble des objectifs visés pour un utilisateur. Essayant de satisfaire au mieux les divers besoins, ils enrichissent leurs programmes de centaines de fonctionnalités dont la plupart ne seront pas utilisées ou inadaptées. Les programmes seraient beaucoup plus simples si chaque utilisateur pouvait ajouter lui-même ou adapter les fonctionnalités souhaitées. L'approche de la programmation par aspect présente de nombreux avantages pour cette problématique car les greffons appliqués au programme initial respectent bien l'indépendance et la séparation des préoccupations.

La seconde partie de cette thèse étudiera les possibilités de rendre accessible, à des utilisateurs non spécialistes de l'informatique les mécanismes d'adaptation et de paramétrisation des applications pour les transformer suivant les besoins spécifiques des utilisateurs.

## Phase Expérimentale

Cette étude s'achèvera par l'évaluation de prototypes de validation sur la plateforme WComp pour les contributions proposées dans le cadre d'usages suggérés par le CSTB avec une mise en œuvre possible sur la plate-forme GerHome<sup>1</sup> et dans l'Ubiquarium<sup>2</sup> Informatique (Hourdin et al., 2006).

## Références

- Carzaniga A., Rosenblum D. S., Wolf A. L., « Design of a Scalable Event Notification Service: Interface and Architecture », Tech. Rep. CU-CS-863-98, Department of Computer Science, Univ. of Colorado, september. 1998
- Cheung-Foo-Wo D., Tigli J.-Y., Lavirotte S. et Riveill M.. « Self-adaptation of event-driven component-oriented Middleware using Aspects of Assembly ». Dans 5th International Workshop on Middleware for Pervasive and Ad-Hoc Computing (MPAC), California, USA, novembre 2007.
- Cheung-Foo-Wo D., Tigli J.-Y., Lavirotte S. et Riveill M.. « WComp: a Multi-Design Approach for Prototyping Applications using Heterogeneous Resources ». Dans Proceedings of the 17th IEEE International Workshop on Rapid System Prototyping (RSP), pages 119-125, Chania, Crete, juin 2006. IEEE Computer Society Press.
- Hourdin V., Cheung-Foo-Wo D., Lavirotte S. et Tigli J.-Y.. « Ubiquarium Informatique: Une plate-forme pour l'étude des équipements informatiques mobiles en environnement simulé ». Dans 3ème Journées Francophones Mobilité et Ubiquité (UbiMob), Paris, septembre 2006.
- Kiczales G. Lamping J., Menhdhekar A., Maeda C., Lopes C., Loingtier J.-M., and Irwin J., « Aspect-oriented programming », *Proc. Euro. Conf. OOP 1997, LNCS 1241, Springer Verlag*, 1997.
- Lavirotte S., Lingrand D., Tigli J.-Y., « Définition du contexte et méthodes de sélection », Actes des Secondes Journées Francophones: Mobilité et Ubiquité (UbiMob), 2005, p. 9-12.
- Rashid A., Kortuem G., « Adaptation as an aspect in pervasive computing », *Symposium on Principles of Distributed Computing*, 2004.
- Robinson J., Wakeman I., and Chalmers D. « Composing software services in the pervasive computing environment: Languages or APIs? ». *Journal of Pervasive and Mobile Computing*, Apr. 2007.
- Szyperski C. « Component Software - Beyond Object-Oriented Programming », Addison-Wesley, 1999.
- Tigli J.-Y., Cheung-Foo-Wo D., Lavirotte S. et Riveill M.. « Adaptation au contexte par tissage d'aspects d'assemblage de composants déclenchés par des conditions contextuelles ». RSTI Série ISI - Adaptation et Gestion du Contexte, volume 11, numéro 5, pages 89-114, 2006. ISBN 2-7462-1672-8.

<sup>1</sup> Services numériques de maintien à domicile des personnes âgées.

<sup>2</sup> Du latin *Ubique*, en toute chose est tout être, avec le suffixe *rium* signifiant lieu ou structure. Donc *Ubiquarium* Informatique : « lieu ou structure dans laquelle l'informatique est en toute chose et en tout être ».