

# Une lignes de produits logiciels de systèmes auto-adaptatifs

Philippe Collet  
Philippe.Collet@unice.fr

15 février 2014

**Nombre d'étudiants souhaités : 3 à 4**

## Description du sujet

De plus en plus, les systèmes logiciels doivent s'auto-adapter pour tenir compte de nouveaux besoins utilisateurs ou de changements de leur environnement qui surviennent après leur déploiement. En raison de leur grande complexité, les programmes adaptatifs sont difficiles à spécifier, à concevoir, à vérifier et à valider. Le plus souvent, ces programmes sont construits autour de boucles de rétro-action (*sense, compute, act*).

L'équipe Modalis de l'I3S développe ACTRESS, un environnement qui permet de concevoir et d'intégrer des mécanismes d'adaptation sous formes de boucles de rétro-action. Il repose sur un langage dédié, xFCDL, qui permet de définir et composer plus simplement les éléments des boucles à un niveau abstrait, tout en s'assurant automatiquement des propriétés de consistance et en permettant la génération de code efficace (support d'exécution performant à base d'acteurs Akka). Le langage fournit des mécanismes de composition, de distribution et est réflexif, ce qui permet de coordonner et composer des boucles multiples et distribuées.

Malgré ces avancées, il reste une grande part de complexité dans le déploiement de ces boucles de rétro-action, car elles doivent être très précisément configurées et customisées pour fournir une auto-adaptation de qualité. On retrouve ainsi de nombreux points de variation entre des éléments communs, à différents niveaux, des éléments de boucle (un senseur de CPU), des boucles (une auto-adaptation de charge sur un serveur web) ou de patrons de boucle similaires aux classiques patrons de conception. Ceci est un cadre de travail typique de lignes de produits logiciels (LPL). S'inspirant des techniques utilisées dans l'automobile ou l'aéronautique, le paradigme des LPL permet de gérer une partie de cette complexité en décrivant la variabilité d'un domaine de production, puis en appliquant des techniques de génération automatique.

L'équipe Modalis de l'I3S développe aussi FAMILIAR, un langage textuel pour la manipulation à grande échelle des feature models (langage étudié dans l'UE de génie logiciel). Actuellement développé en collaboration avec l'Université de Rennes 1 et la Colorado State University aux Etats-Unis, FAMILIAR est un langage dédié, reposant sur une implémentation en Java. Il dispose aussi d'une version en Scala facilitant la construction de lignes de produit.

L'objectif de ce TER est d'étudier et de concevoir un ensemble de patrons de conception variables pour la construction des systèmes auto-adaptatifs. Il s'agira d'utiliser FAMILIAR pour décrire la variabilité de patrons définis comme des squelettes de code ACTRESS. Quelques exemples de patron d'auto-adaptation existent déjà et serviront de points de départ. De même,

il existe une application de référence implémentée en ACTRESS qui servira à valider que les patrons variables peuvent être appliqués.

Le TER comportera les étapes suivantes à organiser selon un processus agile :

- étude et apprentissage de l’environnement ACTRESS,
- étude des patrons d’auto-adaptation existants,
- étude d’une solution technique pour définir des patrons variables à l’aide de la version Scala de FAMILIAR.
- implémentation en intégration continue de cette solution technique validée par une application de référence.

Ce TER est une excellente opportunité d’utiliser des langages (Scala, langages dédiés) et concepts (modèles d’acteur Akka, systèmes auto-adaptatifs) avancés tout en développant de façon agile.

## Lieu

Laboratoire I3S, site des templiers, Sophia Antipolis

## Prérequis

Bonne maîtrise du langage Java.

## Informations complémentaires

- ACTRESS : <http://fikovnik.net/Actress/>
- FAMILIAR : <http://familiar-project.github.io/>
- Scala : <http://www.scala-lang.org/>
- Akka : <http://akka.io/>