

# Expression qualitative de politiques d'adaptation pour les composants Fractal

Franck Chauvel<sup>\*,\*\*</sup>, Olivier Barais<sup>\*</sup>, Noël Plouzeau<sup>\*</sup>, Isabelle Borne<sup>\*\*</sup>, Jean-Marc Jézéquel<sup>\*</sup>

<sup>\*</sup> IRISA (INRIA & Université de Rennes 1)  
Campus de Beaulieu  
F-35042 RENNES (FRANCE)  
prénom.nom@irisa.fr

<sup>\*\*</sup> Laboratoire VALORIA (Université de Bretagne Sud)  
Centre de Recherche Yves Coppens  
Campus de Tohannic  
56017 VANNES CEDEX  
prénom.nom@univ-ubs.fr

**Résumé.** Les plates-formes d'exécution récentes telles que Fractal ou OpenCOM offrent de nombreuses facilités pour assurer la prise en compte de propriétés extra-fonctionnelles (introspection, sondes, chargement dynamique, etc). Cependant, l'intégration de politiques d'adaptation reste délicate car elle nécessite de corrélérer la configuration du système avec l'évolution de son environnement. Le travail présenté dans cet article propose une description qualitative des évolutions de l'environnement et une interprétation possible basée sur de la logique floue. L'article présente également une extension de la plate-forme Fractal implémentant les mécanismes nécessaires à l'exécution de ces politiques d'adaptation de haut niveau. L'approche est illustrée à l'aide d'un serveur HTTP qui modifie sa configuration (architecturale et locale) en fonction de plusieurs paramètres extra-fonctionnels tels que la charge du serveur et la dispersion des requêtes.

## 1 Introduction

Lors du développement de nouveaux systèmes logiciels, l'aspect fonctionnel n'est plus l'unique critère de satisfaction de l'utilisateur : les problématiques extra-fonctionnelles sont au cœur des efforts de recherche actuels. Il est, par exemple, nécessaire de maintenir un niveau correct pour des propriétés liées à la qualité de service (QoS pour "Quality of Service") telles que la fiabilité, la disponibilité, le temps de réponse, l'ergonomie, *etc.*

Pour maintenir la qualité de service d'un système, deux approches sont possibles. On peut d'une part, vérifier *a priori*, (lors de la conception) que les contraintes sur les propriétés de qualité de service ne sont pas violées à l'exécution. L'utilisation de techniques formelles comme les réseaux de Petri par exemple permet d'obtenir des prévisions sur les consommations de