

# Environnement non-invasif pour la parallélisation massive de tâches

Fabien Hermenier `fabien.hermenier@unice.fr`  
Vincent Kherbache `vincent.kherbache@inria.fr`

30 janvier 2015

**Nombre d'étudiants souhaités : 4**

## Description du sujet

BtrPlace<sup>1</sup> est un ordonnanceur de machines virtuelles pour cloud IaaS développé dans l'équipe Scale. Actuellement l'évaluation de ses performances nécessite la résolution de 2,500 instances dont la durée de chaque résolution peut varier de quelques millisecondes à plusieurs minutes. L'évaluation se fait sur la plateforme Grid'5000<sup>2</sup>. Les instances sont distribuées et exécutées en parallèle sur un ensemble de machines (entre 20 et 50 le plus souvent) pour minimiser le temps d'évaluation. Grid'5000 peut distribuer des tâches via son ordonnanceur mais le temps de déploiement d'une simple tâche est trop long par rapport à son temps de calcul effectif. Une solution pourrait consister à déployer et adapter des ordonnanceurs existants dans Grid'5000 mais le déploiement automatique d'un tel environnement pourrait prendre des mois et n'apporter qu'une faible amélioration des temps d'ordonnancement. La solution actuelle consiste alors à réserver des machines physiques sur Grid'5000 et à distribuer les instances de manière statique (et donc sous-optimale) via différents scripts. L'approche n'est pas souple, pas adapté aux tests rapide en mode interactif, et non tolérante aux pannes.

Cette situation est relativement fréquente lorsque l'on a énormément de petits calculs à réaliser. Nous souhaiterions revoir l'approche pour évaluer les performances de BtrPlace. Concrètement, nous souhaitons un ordonnanceur simple, installable en espace utilisateur et pouvant distribuer des calculs indépendants (instances) sur des machines esclaves. Pour simplifier l'approche les esclaves ne devront pas être génériques, il est acceptable de coder un esclave par type de calcul si la librairie utilisée pour communiquer avec l'ordonnanceur est suffisamment simple à embarquer et à utiliser.

Pour ce projet, il importe de proposer :

- un ordonnanceur générique simple à déployer en espace utilisateur.
- une architecture REST<sup>3</sup> pour le dialogue entre les composants et pour s'abstraire des problèmes de langages.

---

1. <http://www.btrplace.org>

2. <http://www.grid5000.fr>

3. [http://fr.wikipedia.org/wiki/Representational\\_State\\_Transfer](http://fr.wikipedia.org/wiki/Representational_State_Transfer)

- une bibliothèque minimale en Java reposant sur l'architecture REST pour faciliter la communication ordonnanceur/esclaves.
- une interface Web simple et intuitive pour superviser l'exécution des instances.

La validation du projet se fera par le développement d'esclaves factices ou d'un esclave adapté pour BtrPlace. Le passage à l'échelle et la réactivité de l'ordonnanceur seront vérifiés en déployant plusieurs centaines d'esclaves sur Grid'5000.

## Lieu

Inria

## Pré-requis

- Programmation Java
- HTML5, CSS, JavaScript, Bootstrap
- fondamentaux en programmation distribué et parallèle

## Informations complémentaires