

Sujet de Master M2 recherche

Développement logiciel : *the green touch*

Encadrant : Thomas Ledoux (Thomas.Ledoux@mines-nantes.fr), tel. : 02 51 85 82 19

Lieu : Ecole des Mines de Nantes

Stage gratifié ou rémunéré, possibilité de poursuivre en thèse Cifre

Mots clés : énergie, langages, green patterns, architectures logicielles, systèmes autonomes, Cloud computing

Contexte et problématique

En quelques années, le problème de la maîtrise énergétique est devenu un enjeu majeur de société. En informatique, les principaux travaux se sont concentrés sur des mécanismes permettant de maîtriser l'énergie au niveau du matériel [1] et des infrastructures serveurs (centre de données) [2]. Le renforcement du rôle de l'informatique dans notre société - de plus en plus numérique - conduit à traiter ces problèmes également au niveau du logiciel.

L'exemple de la société Facebook rapidement confrontée à ce trait d'union entre le logiciel, le matériel et l'énergie est révélateur de l'importance de traiter l'efficacité énergétique directement au sein du logiciel. Une estimation passée montrait que les 30.000 serveurs de Facebook consommaient 100 millions de kWh par an [3]. Facebook utilise maintenant le pré-compilateur HipHop for PHP qui transforme du code PHP en code C++ qui est ensuite compilé à l'aide de g++. 90% des pages de Facebook reposent aujourd'hui sur HipHop avec un gain moyen de 50%. Il faut donc 2 fois moins de serveurs à Facebook pour fonctionner [4] !

L'anecdote précédente montre combien il est important de s'intéresser à l'*éco-conception du logiciel*. L'écriture de programme, la conception de l'architecture logicielle peuvent avoir une incidence sur l'environnement. Le logiciel est aujourd'hui le principal facteur d'obsolescence des équipements informatiques qui implique alors toute une chaîne de remplacement très énergivore (la fabrication du matériel électronique concentre les principales nuisances environnementales) [5]. Il est donc important d'analyser les différentes pistes et de déterminer les divers leviers possibles pour participer à la génération de « code vert », à l'efficacité énergétique du logiciel.

Objectifs

La maîtrise énergétique du logiciel tout au long de son cycle de vie (de sa conception à son exécution) constitue l'objectif principal de ce stage. Pour faire suite à un *position paper* de l'équipe en 2010 [6], il s'agit de faire de réelles propositions et de les mettre en œuvre. Plusieurs axes de recherche sont possibles : (i) au niveau langage et compilation [7] [8] ; (ii) au niveau règles/bonnes

pratiques de programmation et *patterns* [9][10][11] ; (iii) au niveau infrastructures logicielles (client/serveur, SOA, Cloud computing) [12] [13].¹

Plan de travail

En fonction de l'état d'avancement et des affinités du candidat, le travail à réaliser s'effectuera autour deux parties indépendantes et complémentaires.

Partie 1

- Etat de l'art multi-axes

- Tester les outils de mesure qui permettent de mesurer les consommations des ressources (e.g. Energy Checker d'Intel [14]). Faire un bilan.
- Identifier et valider un certain nombre de règles d'éco-conception (recensées sur différentes architectures, dans différents langages).

- Conception/réalisation de « composant vert »

- Proposer un certain nombre de green patterns pour Java, pour le développement d'applications Web.
- Proposer des outils d'introspection de code, voire de refactoring de « code gourmand ». Par exemple, on voudrait savoir si telle portion de programme ne fait pas trop d'échanges avec le serveur ou n'appelle pas de bibliothèques énergivores (e.g. Flash).
- Définir des métriques permettant de noter l'efficacité énergétique d'une application.

Partie 2

Reconfiguration dynamique d'application « SaaS verte »

Le passage à l'échelle et la disponibilité d'une application logicielle est souvent assurée aujourd'hui par son exécution dans un Cloud. Or, notre équipe a récemment montré, grâce à un simulateur, qu'une application de type SaaS peut-être reconfigurée à la volée pour être plus *eco-compliant* en fonction du contexte d'exécution tout en garantissant une qualité de service (QoS) correcte [13]. L'objectif de cette partie du stage est de réaliser un *proof-of-concept* pour le Cloud montrant une application SaaS se reconfigurant dynamiquement pour réduire son empreinte énergétique et d'évaluer notre modèle théorique.

Positionnement

Ce stage s'inscrit au cœur des activités Green IT de l'équipe Ascola. Le stage est complémentaire aux travaux de l'équipe réalisés dans l'ANR MyCloud (<http://mycloud.inrialpes.fr>) et se place également au carrefour des sensibilités Langage et Systèmes distribués de l'équipe. **Possibilité de poursuite en thèse Cifre² avec une société nantaise en septembre 2012.**

Profil du candidat

Le candidat possèdera une certaine expérience dans le domaine du génie logiciel et les systèmes répartis. Un « appétit » pour les problématiques d'environnement durable sera nécessaire. Une

¹ Les axes d'étude systèmes d'exploitation, réseaux et télécoms seront exclus de ce stage.

² http://www.anrt.asso.fr/fr/espace_cifre/accueil.jsp?r=3&p=1

connaissance approfondie dans un ou plusieurs des domaines suivants sera appréciée : langages dédiés, systèmes autonomes, architectures orientées services, Cloud computing.

Références

- [1] Susanne Albers. Energy-efficient algorithms. Communications of the ACM Vol 53, n°5 (May 2010)
- [2] Hermenier, F., Lorca, X., Menaud, J.M., Muller, G., Lawall, J.: Entropy: a consolidation manager for clusters. In: VEE '09: Proceedings of the 2009 ACM SIGPLAN/SIGOPS international conference on Virtual execution environments, New York, NY, USA, ACM (2009)
- [3] <http://www.greenit.fr/article/energie/facebook-consomme-autant-quun-tgv>
- [4] <http://www.zdnet.fr/blogs/greenit/hiphop-facebook-divise-par-deux-le-temps-d-execution-de-php-39712771.htm>
- [5] <http://www.greenit.fr/article/logiciels/logiciel-la-cle-de-l-obsolence-programmee-du-materiel-informatique-2748>
- [6] Jean-Marc Menaud, Adrien Lèbre, Thomas Ledoux, Jacques Noyé, Pierre Cointe, Rémi Douence and Mario Südholt. Vers une réification de l'énergie dans le domaine du logiciel. In Journées du GDR Génie de la Programmation et du Logiciel (GDR GPL 2010), Pau, mars 2010.
- [7] Shlomo Zilberstein. Using anytime algorithms in intelligent systems. AI Magazine, 17(3), 1996.
- [8] Seo, C., Malek, S., Medvidovic, N.: Component-Level Energy Consumption Estimation for Distributed Java-Based Software Systems. In: Component-Based Software Engineering, Springer Berlin Heidelberg (2008)
- [9] <https://sites.google.com/a/octo.com/green-challenge>
- [10] <http://greencodelab.fr/>
- [11] Gregor Hohpe, Bobby Woolf. Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions, Addison-Wesley Professional, 2004
- [12] Mello Ferreira, A., Kritikos, K., Pernici, B.: Energy-aware design of service-based applications. In: ICSOC-ServiceWave '09: Proceedings of the 7th International Joint Conference on Service-Oriented Computing, Berlin, Heidelberg, Springer-Verlag (2009)
- [13] Frederico Alvares De Oliveira Junior and Thomas Ledoux. Self-management of applications QoS for energy optimization in datacenters, 2nd International Workshop on Green Computing Middleware, In conjunction with ACM/IFIP/USENIX 12th International Middleware Conference, 2011
- [14] <http://software.intel.com/en-us/articles/intel-energy-checker-sdk/>