

Un modèle de composant pour la reconfiguration dynamique de réseaux de capteurs sans fil

Natacha Hoang, Nicolas Belloir, Xavier Detant, Congduc Pham

*LIUPPA

BP 1155

64013 Pau CEDEX

(nom.prenom@univ-pau.fr)

Résumé. Dans ce papier, nous proposons une approche permettant la reconfiguration dynamique des réseaux de capteurs sans fil (WSNs). Cette proposition repose sur une architecture logicielle bâtie autour d'un nouveau système d'exploitation et d'un modèle de composant basé sur Fractal. Ce système d'exploitation, baptisé *Valentine*, permet de conserver à l'exécution la construction par composant, en opposition à une construction monolithique caractéristique des principales solutions existantes dans les WSNs. De plus, étant dédié aux WSNs, il est tout naturellement orienté événement. Il a été généré à partir du canevas logiciel Think. Le modèle de composant permet de réutiliser la flexibilité du modèle abstrait Fractal et d'implémenter le mécanisme de reconfiguration dynamique.

1 Introduction

Les avancées technologiques confirment la présence croissante de l'informatique dans les objets du monde réel : d'après Crnkovic (2006), 98 % du code développé pour les systèmes futurs sera embarqué. De plus en plus d'objets physiques se voient dotés de processeurs et de moyens de communication sans fil leur permettant de traiter de l'information mais également de la transmettre. Cette évolution prend place dans le contexte de l'informatique pervasive, mieux connue sous le nom d'*ubiquitous computing* (voir Weiser (1993)).

Les réseaux de capteurs sans fil¹ entrent dans ce contexte. Ils sont constitués de petits éléments électroniques, appelés capteurs, aux ressources fortement contraintes, mais néanmoins dotés de moyens de communication, de traitement et de sauvegarde de l'information. Ils se différencient des capteurs traditionnels généralement fixes ou peu mobiles par leur mobilité due à la non présence de câblage ainsi que par leur ressources plus réduites. En dépit de cet élément de distinction, ces capteurs sont néanmoins capables d'acquérir des informations sur leur environnement proche, de les traiter et de les distribuer. Cela permet d'envisager alors un large panel de nouvelles applications plus proches du monde réel.

Cependant, l'intégration de tels capteurs dans leur environnement cible n'est pas chose aisée. Les problèmes rencontrés sont généralement liés aux contraintes inhérentes induites par leurs ressources : par exemple la gestion drastique qui doit être faite de la batterie, qui par définition est à capacité fixe et non rechargeable, ou encore les problèmes de gestion de la mémoire. Un certain nombre de travaux ont porté sur le développement d'environnements spécifiques à ces capteurs, pour prendre en compte ces contraintes. Par exemple, Hill et al. (2000) ont proposé TinyOS, le système d'exploitation de référence des WSNs. Il gère au plus prêt la mémoire limitée grâce à de nombreuses optimisations. Cette optimisation a cependant un coût, notamment au niveau de l'architecture du système d'exploitation et du code qui sont généralement monolithiques et par de là même peu souples. Cela rend par exemple les capteurs difficiles à reconfigurer dynamiquement.

D'un autre côté, l'ingénierie logicielle basée composant (voir le livre de Szyperski et al. (2002))², est une approche maintenant reconnue permettant le développement d'applications à la fois flexibles et bien structurées, ayant souvent des capacités de reconfiguration ou d'administration à distance. Dans ce contexte, la reconfiguration dynamique consiste à remplacer un composant logiciel par un autre dans une application en exécution. Cette action peut être nécessaire pour plusieurs raisons. Il peut par exemple être nécessaire de substituer un composant

¹WSN pour *Wireless Sensor Networks*

²CBSE pour *Component-Based Software Engineering*