

Sujet de master recherche « Architectures logicielles distribuées » 2005-2006

Composition et wrapping des connecteurs logiciels

Encadrant principal : Mourad OUSSALAH
Courriel : Mourad.Oussalah@univ-nantes.fr
Tel : 02-51-12-58-47
Co-encadrants : Tahar KHAMMACI
Courriel : Tahar.Khammaci@univ-nantes.fr
Tél : 02-51-12-58-34

Objectif du projet

Actuellement, un grand intérêt est porté au domaine de l'architecture logicielle. Cet intérêt est motivé principalement par la réduction des coûts et des délais de développement des applications informatiques. L'architecture logicielle modélise un système logiciel en termes de composants et d'interactions entre composants. Elle joue le rôle de passerelle entre l'étape d'expression des besoins et l'étape de codage du système logiciel. Elle permet d'exposer de manière compréhensible et synthétique la complexité d'un système logiciel et de faciliter l'assemblage des composants logiciels. L'assemblage des composants logiciels n'est pas toujours évident dans la mesure où les composants disponibles ne sont pas toujours compatibles. Ainsi, la technologie des connecteurs logiciels est proposée pour adapter les composants hétérogènes afin de faciliter leur assemblage. Les connecteurs logiciels sont des entités architecturales de communication qui modélisent de manière explicite les interactions (transfert de contrôle et de données) entre les composants. Ils contiennent des informations concernant les règles d'interaction entre les composants. Plusieurs propositions de classification des connecteurs ont été proposées. Nehta et al. [Nehta'00] ont classé les services d'interaction offerts par les connecteurs en quatre types. Chaque type de connecteur offre un ou plusieurs service (s) d'interaction : service de communication, service de coordination, service de conversion et service de facilitation. Oussalah et al. [Oussalah'04] ont classé les connecteurs en trois groupes : connecteurs implicites, ensemble énuméré de connecteurs prédéfinis et connecteurs dont les sémantiques sont définies par les utilisateurs. Nous nous intéressons à deux mécanismes sur les connecteurs : composition et « wrapping » que nous explicitons dans la suite.

Composition de connecteurs

La composition de connecteurs consiste à définir de nouveaux connecteurs à partir de connecteurs déjà existants en utilisant la relation de composition. Les applications logicielles complexes sont construites par une collection de composants logiciels indépendants et des mécanismes qui permettent l'interaction de ces composants logiciels. Dans certaines situations, des formes particulières d'interaction sont nécessaires pour relier des composants qui ne sont pas compatibles entre eux. De telles situations rendent très critique la conception et l'implémentation des mécanismes d'interaction. Par ailleurs, il est difficile de créer des connecteurs riches sémantiquement dont les systèmes logiciels en ont besoin. Actuellement, les architectes disposent de deux choix lors de la sélection des types de connecteurs pour la conception de systèmes logiciels : 1) utiliser un connecteur « sur étagères » ou 2) utiliser un connecteur spécialisé sans qu'aucune implémentation ne soit disponible. Aucun des deux choix ne semble faire l'unanimité. En effet, d'une part, il n'est pas toujours possible de trouver un connecteur déjà existant qui réponde exactement aux besoins du système logiciel, et d'autre part, la création d'un nouveau mécanisme de connexion est souvent une tâche difficile et très coûteuse. Il est donc nécessaire de combiner des possibilités multiples d'interactions dans un connecteur. Ainsi, il sera possible de construire de nouveaux types de connecteurs de manière systématique et à moindre coût par composition de connecteurs déjà existants.

Wrapping de connecteurs

Le « wrapping » est défini comme l'emballage d'une application dans une nouvelle interface, pour la faire exécuter dans un contexte différent. Le « wrapper » est un programme enveloppant l'exécution d'un autre programme pour lui préparer un environnement particulier. Il est utilisé par exemple pour sécuriser le fonctionnement de certaines applications en contrôlant très précisément leur interface avec le reste de l'univers connu.

Lors de l'assemblage d'un système logiciel à partir de composants développés de manière indépendante et potentiellement incompatibles, le mécanisme de « wrapping » peut être utilisée pour résoudre les problèmes d'incompatibilités ainsi que pour remédier à certaines déficiences des propriétés non fonctionnelles du système logiciel. Malheureusement, les méthodes actuelles de création et d'utilisation du mécanisme de « wrapping » sont souvent réalisées de manière ad hoc. Ceci a pour conséquence de rendre difficile la réutilisation ou la composition de ces « wrappers ». Le principal problème consiste à déterminer avec exactitude les critères de base pour la création, la compréhension et l'application des « wrappers ». En se focalisant sur une classe de connecteurs logiciels (wrappers qui sont dédiés à la communication et à la compatibilité par exemple), il est alors possible de donner une sémantique pour la caractérisation des connecteurs enveloppés tels que les protocoles de transformation, leur modularité et le raisonnement sur leurs propriétés.

Travail à réaliser

L'approche COSA (Component-Object based Software Architecture) permet de représenter l'architecture logicielle d'un système en termes de composants, connecteurs et configurations [Oussalah 04]. Il s'agit d'étudier les deux mécanismes de composition et de wrapping des connecteurs et de proposer des solutions pour leur prise en compte dans le cas des connecteurs COSA.

Références

- 1- Mehta N.R, Medvidovic N., Phadke S., "Towards a Taxonomy of Software Connectors", Proceedings of the 22nd International Conference on Software Engineering (ICSE'00), Limerick, Ireland, June 2000.
- 2- Oussalah M., Smeda A., Khammaci T., "An Explicit Definition of Connectors for Component-Based Software Architecture", Proceedings of the IEEE International Conference on the Engineering of Computer Based Systems (ECBS 2004), Brno, Czech Republic, May 2004.
- 3- Oussalah M. Khammaci T. Smeda A, "Expliciter les connecteurs dans les architectures logicielles", Actes des Journées OCM, Lille, Mars 2004.
- 4- B. Spitznagel and D. Garlan, "A Compositional Approach for Constructing Connectors", Proceedings of the Working IEEE.IFIP Conference on Software Architecture (WICSA'01), Royal Netherlands Academy of Arts and Sciences Amsterdam , The Netherlands, August 2001.
- 5- B. Spitznagel and D. Garlan, "A Compositional Formalization of Connector Wrappers", Proceedings of the 2003 International Conference on Software Engineering (ICSE'03), Portland, Oregon, USA, May 2003.