

Sujet de master recherche « Architectures logicielles distribuées » 2006–2007

Extraction de Connecteurs pour l'Architecturation de Systèmes Orientés Objet

Encadrant principal : Tahar Khammaci
 courriel : Tahar.Khammaci@univ-nantes.fr
 tél. : 02-51-12-58-34

Co-encadrant(s) : Abdelhak-Djamel Seriai

Mots clés:

Génie logiciel, Architecture logicielle, Retro-ingénierie, Architecturation, Java

Objectif du stage

La modélisation et la représentation de la notion d'architecture logicielle est devenue aujourd'hui une phase importante du processus de développement des systèmes informatiques. L'architecture d'un système décrit sa structure à un haut niveau d'abstraction sous la forme d'un graphe bi-parti dont les composants et les connecteurs forment les noeuds. Ce niveau d'abstraction offre ainsi de nombreux avantages tout au long du cycle de vie du logiciel.

Cependant l'architecture du système n'est pas toujours disponible. Nous distinguons deux cas de figures. Le premier est celui des systèmes ne disposant pas d'une représentation de leurs architectures. Le deuxième cas de figure est dû aux écarts qui peuvent exister entre l'architecture d'un système et la représentation dont on dispose : c'est le phénomène d'érosion. Dans les deux cas, l'absence d'architecture ou au pire la présence d'une architecture erronée rendent la maintenance, l'évolution et la migration dangereuses.

Partant de ces constats, nous nous sommes intéressés, dans le cadre du projet ROMANTIC (Re-engineering of Object-oriented systems by Architecture extraction and migration to Component based ones.), au développement d'une approche permettant d'extraire l'architecture logicielle d'un système existant à partir de son implémentation objet. La figure 1 ci-dessous montre le processus d'architecturation.

Suivant ce processus, l'architecture d'un système orienté objet existant est obtenue par l'identification de ses éléments architecturaux : ses composants, ses connecteurs et sa configuration. Dans ce cadre, nous définissons un composant architectural comme le regroupement d'un ensemble d'éléments objets (classes objets, interfaces, etc.). Ces groupements sont identifiés grâce à la définition d'une partition de l'ensemble des classes du système, où chaque partie représentera un futur composant.

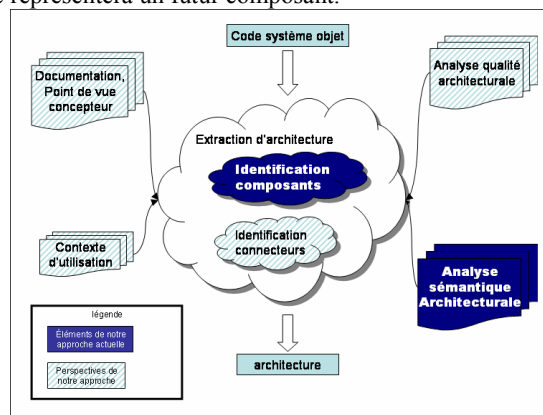


Figure : Phases et éléments de l'architecturation.

Travail à réaliser

Il s'agit de proposer une approche pour l'identification et l'extraction des connecteurs logiciels dans le code d'un système orienté objet. Pour cela, le stagiaire réalisera les étapes suivantes :

1. Etude des propriétés des connecteurs dans une architecture logicielle. Ce travail consiste à réaliser un état de l'art des différents types de connecteurs et de leurs rôles.
2. Définition d'un certain nombre de patterns réifiant les propriétés de quelques types de connecteurs étudiés dans l'état de l'art.
3. Définition d'un processus d'identification et d'extraction des pattern-connecteurs à partir d'un code Java.

Références

1. F. Lüders, I. Crnkovic, and A. Sjögren. Case study: Componentization of an industrial control system. In Proc. 26th COMPSAC 2002, Oxford, UK, Aug. 2002. IEEE Computer Society Press.
2. C. Szyperski. Component Software: Beyond Object-Oriented Programming. ACM, Press and Addison-Wesley, 1998.
3. M. Oussalah & al. Ingénierie des composants : Concepts, techniques et Outils, juin 2005, Vuibert.
4. C. Riva : Architecture Reconstruction in Practice. In: Proceedings of the 3rd Working IEEE/IFIP Conference on Software Architecture (WICSA), 2002.
5. C. Riva, Y. Yang : Generation of Architectural Documentation using XML. In: Proceedings of the 9th Working Conference on Reverse Engineering (WCRE), pp. 161-169, 2002.
6. K. De Hondt, A Novel Approach to Architectural Recovery of Object-Oriented Systems, PhD Theses, Vrije Universiteit Brussel, 1998.
7. M. Shaw and D. Garlan. Software Architecture. Perspectives on an Emerging Discipline. Prentice Hall, 1996.
8. T. A. Wiggerts, Using Clustering Algorithms in Legacy System Remodularization, Working Conference of Reverse Engineering (WCRE '97), pp. 33-43, 1997.
9. N. C. Mendonça, Software architecture recovery for distributed systems, PhD Thesis, Imperial College, Department of Computing, 1999.
10. R. Kazman and S. J. Carriere. Playing Detective: Reconstructing Software Architecture from Available Evidence, Automated Software Engineering, 6(2):107-138, Apr. 1999.
11. N. Medvidovic and V. Jakobac. Using software evolution to focus architectural recovery., Volume 13, Number 2 , April, 2006. Special Issue Section on Software Architecture Recovery. Automated Software Engineering Journal. Springer Netherlands.
12. A. Smeda, T. Khammaci et M. Oussalah. Software connectors in the COSA approach, Proceedings of the third AOSD Workshop on Aspects, Components, and Patterns for Infrastructure Software (ACP4IS), Lancaster, UK, 2004.
13. N. Medvidovic and R. Taylor. A Classification and Comparison Framework for Software Architecture Description Languages, IEEE Transaction on Software Engineering, 28(1):70-93, 2000.
14. N Mehta, N. Medvidovic and S. Phadke. Towards a Taxonomy of Software Connectors, Proceedings of the 33rd International Conference on Software Engineering (ICSE'00), Limerick, Ireland, May 2000.