

Un ADL pour les Architectures Distribuées à Composants Hétérogènes

Guillaume Dufrêne*, Lionel Seinturier*

*USTL-LIFL CNRS UMR 8022, équipe projet INRIA ADAM
Parc scientifique de la Haute-Borne
40, Avenue Halley
59650 Villeneuve d'Ascq Cedex - FRANCE
{guillaume.dufrenel, lionel.seinturier}@lifl.fr

Résumé. Dans cet article, nous présentons un ADL pour les architectures à composants hétérogènes et distribuées, et son utilisation au moment du déploiement. Actuellement, il n'existe pas de solution générique pour déployer une architecture distribuée basée sur différents intergiciels. Pour répondre à cette problématique, nous proposons dans cet article une approche pour décrire une telle architecture et un support pour le déploiement. Notre solution s'appuie sur un langage de description d'architecture possédant des notions de dépendances verticales et horizontales. Un exemple simple est présenté pour illustrer notre langage et pour valider nos contributions.

1 Introduction

Problématique. Les composants logiciels sont un thème majeur aussi bien dans le monde académique que dans l'industrie informatique. Ils sont le sujet de nombreuses études dans différentes communautés telles que l'ingénierie logicielle ou les intergiciels (middleware). Ces dernières années, de nombreux Canevas de Composants (CC) ont été proposés comme Arch-Java (Aldrich et al., 2002), PECOS (Genssler, 2002), K-Component (Dowling et Cahill, 2001), OpenCOM (Clarke et al., 2001), OSGi (OSGi Alliance, 2004) ou Fractal (Bruneton et al., 2006). Une étude plus détaillée montre qu'il existe en fait plus de 20 canevas différents. Certains de ces CC se veulent généralistes alors que d'autres visent des domaines d'application bien précis comme les systèmes temps réels ou les environnements dynamiques. Dans le domaine des intergiciels, la notion de composant logiciel prend une ampleur importante aussi bien dans le monde professionnel avec des CC comme EJB (Sun Microsystems, 1997), .NET/COM+ (Lantim, 2003), CCM (Merle et al., 2002), SCA (OSOA, 2007) mais aussi dans le monde académique avec des CC tel que QuA (Eliassen et al., 2004). Les composants se présentent comme une solution bien plus adaptée que les objets, choisis il y a une dizaine d'années, pour implémenter des intergiciels à large échelle et adaptables. Les concepts de haut niveau de ces CC sont similaires. Ils partagent l'idée qu'*un composant logiciel est une unité de composition possédant des interfaces définies contractuellement et des dépendances au contexte explicites. Un composant logiciel peut être déployé indépendamment et il est sujet à la composition par un*