

COMPOSANTS ADAPTABLES ET MOBILES

Anne-Marie Dery-Pinna et Jérémy Fierstone

Equipe Rainbow, I3S
930, route des Colles, 06903 Sophia-Antipolis cedex

Résumé — Ce document présente brièvement notre modèle de composants qui comprend la définition d'interactions logicielles. Les composants décrits dans ce modèle peuvent être assemblés dynamiquement par pose d'interactions, adaptés à différents environnements d'exécution. Cette spécification des interactions et la capacité d'adaptabilité inhérente à notre modèle à composants permettent d'envisager une migration des composants.

1. Objectif

Une nouvelle génération d'applications voit le jour avec l'essor de la programmation par composants logiciels. De nombreuses plates-formes permettent d'assembler des composants afin de construire l'application visée. Seulement, ces solutions n'abordent pas le délicat problème sous-jacent de l'assemblage dynamique des composants non préparés à interagir, de leur adaptation en continu aux changements de l'environnement ainsi que de leur mobilité. L'objectif de nos recherches est de proposer un modèle de composants à base d'interactions permettant de résoudre ces problèmes.

2. Modèle de composants

Le modèle de composants visé doit avoir les mêmes propriétés que le modèle de composants des EJBs mais autorise une plus grande adaptabilité à la composition grâce aux interactions logicielles [1]. Les interactions logicielles permettent de spécifier de manière externe, abstraite et dynamique, dans le langage ISL [1], les dépendances entre composants, mais également de modifier le comportement d'un composant (c'est-à-dire l'exécution des méthodes). Certaines dépendances d'un composant avec d'autres composants et services systèmes (composants particuliers) sont décrites en ISL dans son descripteur de déploiement et stockées sous forme de méta informations accessibles à l'exécution. Par exemple les dépendances d'un composant avec les services de persistance et d'adaptation sont connues à cette étape.

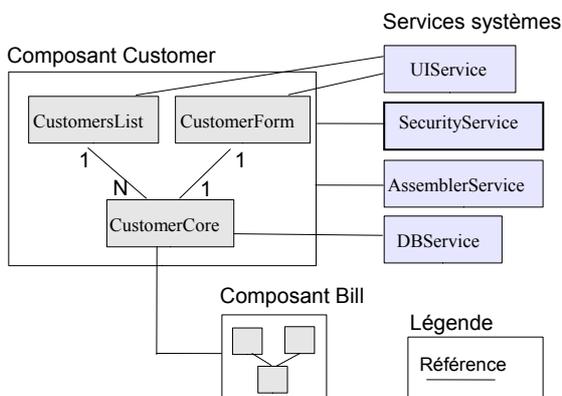


Figure 1 — Vue d'ensemble de l'architecture

La figure 1 illustre l'architecture visée par le modèle qui garantit que les services systèmes sont dissociés du composant *Customer*.

3. Vers des composants assemblables, adaptables et mobiles

1. Assemblage et adaptation des composants

Parmi les services systèmes fournis par la plate-forme d'implémentation du modèle, le service "AssemblerService" utilise le serveur d'interactions Noah [2] afin de poser ou de retirer des interactions logicielles entre composants ou services pour les assembler. Le service "UIService" permettrait d'interpréter un fichier SUNML [3], langage d'IHM adaptable et composable basé sur XML, d'assembler les IHMs par fusion des éléments redondants tout en projetant l'IHM décrite de manière abstraite vers le bon type de périphérique de sortie (voix, braille, texte, graphique).

2. Mobilité des composants

Le service "MobileService" permet quant à lui de sérialiser des composants afin de conserver leurs états respectifs (y compris leurs méta informations), puis de les restaurer par "repose" des interactions. Chaque environnement d'exécution peut inclure ses propres services systèmes car certains sont spécifiques au support matériel (par exemple, sur un PDA, le service de base de données "DBService" sauvegarde les données dans des fichiers). Ainsi les interactions doivent être typées afin de traiter différemment les dépendances d'un composant vers des services systèmes de celles vers d'autres composants. De la sorte, nous devrions pouvoir modifier dynamiquement les interactions entre un composant et un service local.

Bibliographie

- [1] Laurent Berger, «Mise en oeuvre des interactions en environnements distribués, compilés et fortement typés: le modèle MICADO », Thèse de l'Université de Nice, octobre 2001
- [2] <http://rainbow.essi.fr/noah>
- [3] <http://rainbow.essi.fr>