

Résumé de la thèse de **Maryam ESLAMICHALANDAR**

(dirigée par Kamel BARKAOUI)

Intitulée

Compatibilité de la composition des services Web : Adaptation suite à l'évolution des protocoles métier

Résumé

Avec l'utilisation croissante d'architectures logicielles indépendantes de la plateforme et du langage dans le paradigme de l'architecture orientée services (SOA), la technologie de services web permet l'interopérabilité dynamique et flexible des processus métiers aussi bien au niveau intra qu'inter-organisationnel.

Bien que la normalisation des services web permet de réduire l'hétérogénéité et rend plus facile leur interopérabilité, il y a toujours besoin de vérifier leur compatibilité en particulier dans le contexte inter-entreprises. Deux services sont compatibles si une collaboration entre eux est accomplie avec succès et que chacun puisse atteindre ses résultats attendus (états finaux).

L'approche typique devant permettre à des services incompatibles d'interagir correctement est l'adaptation du service. L'adaptation consiste dans ce contexte à faire face principalement aux discordances relevées au niveau des interfaces de service (incompatibilités entre signatures de services) ainsi qu'aux discordances qui ont lieu au niveau des protocoles métiers (incompatibilité dans l'ordre des messages échangés entre services).

On distingue deux principales techniques d'adaptation: modification de service ou synthèse d'un composant adaptateur. L'adaptation en termes de modification de service exige

l'application de certaines mesures d'optimisation pour supporter les spécifications du service partenaire. Dans le cas où l'adaptation traite de la création d'un adaptateur, un composant autonome modère les interactions entre les deux services de sorte que l'interopérabilité soit obtenue. En d'autres termes, l'adaptateur compense les différences entre interfaces de services par conversion de données (c'est-à-dire par transformation de message) et celles entre protocoles métiers en réorganisant les échanges de messages ou en générant un message manquant.

Nous nous concentrons ici sur le problème de la reconfiguration dynamique de l'adaptateur en présence d'évolution de protocoles métiers. Après avoir traité de la vérification d'un adaptateur en exploitant des techniques structurelles existantes développées dans le cadre de la théorie des réseaux de Petri, nous établissons une identification des patrons de mise à jour d'adaptateurs ainsi que la mise en correspondance de ces patrons avec les différents types d'évolutions possibles au niveau des protocoles métiers des services web.

Ce travail a abouti à la proposition d'un algorithme permettant, d'une part de détecter les patrons d'évolution adéquats suite à une évolution d'un des protocoles métier des services partenaires et, d'autre part et sous certaines conditions, la mise à jour à la volée de la spécification du nouvel adaptateur obtenu ainsi que sa vérification.

Enfin, les expérimentations réalisées sur un prototype montrent les avantages en termes de temps et de coût de l'approche dynamique proposée par rapport aux méthodes statiques conduisant systématiquement à la régénération complète de l'adaptateur.

Abstract

The advent of Web service technologies in the paradigm of *Service oriented architecture* (SOA) enables dynamic and flexible interoperation of distributed business processes within and across organization boundaries. One of the challenges in working with heterogeneous and autonomous Web services is the need to ensure their interoperability and compatibility. The typical approach for enabling incompatible services to interact is service *adaptation*. The need for adaptation in Web services comes from the heterogeneity at the levels of service *interface* and *business protocol*. The service interface incompatibilities include service *signature* mismatches (e.g., message and operation name, number; the type of input/output message parameters of operations; and the parameter value constraint). The mismatches at the business protocol (or service behavior) level arise from the order constraints that services

impose on messages exchanges (e.g., *deadlock* where both partner services are mutually waiting to receive some message from the other, and *unspecified reception* in which one service sends a message while the partner is not expecting it). In service interaction through adaptation, an *adapter* mediates the interactions between two services with potentially different interfaces and business protocols such that the interoperability is achieved, i.e., adapter compensates for the differences between their interfaces by data mappings, and between their business protocols by rearranging the messages exchanges or generating a missing message.

In this dissertation, we focus on how to cope with the dynamic evolution of business protocol P of a given service (i.e., P is changed to P') that is adapted by an adapter in the context of service interaction. Web service specifications constantly evolve. For variety of reasons, service providers may change their business protocols. Therefore, it is important to understand the potential impacts of the changes arising from the evolution of service business protocol on the adapter.

We present an approach to automatically detect the effects of business protocols evolution on the adapter and, if possible, to suggest fixes to update the specification of adapter on-the-fly. Besides, we propose a technique to verify the correctness of new adapter which is dynamically re-configured. Finally, we describe a prototype tool where experimentations show the benefits of proposed approach in terms of time and cost compared to the static methods aiming for complete regeneration of adapter or manual inspection and adaption of the adapter with respect to changes in the business protocols.