

## Systèmes de réécriture communicants pour les Web-Services

Equipe Distribcom, INRIA Rennes

### Contexte :

L'équipe DistribCom développe une ligne d'étude dans le domaine des services composés (ou orchestrations de services). Nos premières études ont porté sur le concept de document actif, un modèle proposé par Serge Abiteboul [1] dans son langage Active XML (AXML). Les documents actifs sont des documents structurés de type XML qui contiennent, en plus de données, des références à des services à appeler pour compléter l'information présente dans un document.

Le concept de document actif peut être facilement illustré par l'exemple suivant : supposons qu'un site fournisse à ses utilisateurs le temps actuel dans plusieurs villes. Les données renvoyées pour une ville donnée sont des documents XML contenant : le nom de la ville, le temps actuel, et les prévisions pour le lendemain. On peut noter que le service n'est pas décrit comme un service de prévisions météorologiques, et pourrait ne pas retourner la partie prédiction. Une autre possibilité est de retourner un nom de ville, le temps actuel, plus un pointeur vers un service à appeler (URL) pour connaître les prévisions. Les documents actifs fonctionnent de cette manière, en intégrant aux données des références à d'autres services qui permettent d'obtenir plus d'information. De plus, le concept de document actif permet de traiter des documents de manière paresseuse, c'est-à-dire de travailler sur un document sans expliciter systématiquement les parties du document accessibles par invocation d'un service.

Le concept de document actif est très puissant : lorsque les appels de service sont gardés (i.e. que leur exécution dépend du contenu du document), un document actif permet de décrire un véritable programme distribué, des enchaînements de tâches (workflows,...). Les modèles AXML peuvent ainsi simuler des machines de Turing.

Nous pensons que le concept de document actif est particulièrement adapté à la modélisation et à l'étude des workflows dans les Web-Services complexes, dans lesquels les aspects "contrôle" et les aspect "données" sont interdépendants. En effet, il est courant de rencontrer des situations dans lesquelles le flot de contrôle d'un service dépend des données ("si le montant de la commande dépasse 200 euros, alors souscrire une assurance"), ou inversement

des situations où les données dépendent du contrôle (« Si la commande est confirmée avant la date d'expiration, alors les frais de port sont offerts »).

Il existe de nombreuses possibilités pour modéliser et implémenter des systèmes distribués au dessus d'une architecture web. Dans l'équipe Distribcom, nous avons tout particulièrement étudié les documents actifs, et plus précisément AXML. Bien sûr, d'autres modèles sont possibles. Pour chaque type de modèle, il faut considérer son expressivité, mais aussi la possibilité d'analyser de manière automatique certaines propriétés d'un système. Dans le contexte des Web-Services, les questions qui se posent immédiatement sont :

**La compatibilité** : Deux services sont-ils compatibles, i.e. si un service A appelle un service B, les données envoyées par A lors de l'appel à B sont-elles conformes à l'utilisation prévue de B, et les résultats renvoyés par B sont-ils conformes à ceux attendus par A ?

**La terminaison** : Un appel à un service A s'achève-t-il toujours par le renvoi de données, ou l'appel peut-il ne pas se terminer ?

**La correction** : un système composé de plusieurs services interagissant est-il correct s'il utilise un sous-service A particulier implémentant une fonctionnalité  $f$  du système ? Est-il correct pour tout service satisfaisant un ensemble d'exigences (compatibilité des données, QoS, ...) associées à  $f$  ? Un système satisfait-il certaines propriétés logiques, décrites dans une logique donnée ?

Comme on peut s'y attendre pour, la plupart de ces questions sont indécidables sur des modèles suffisamment expressifs pour modéliser de vrais Web-Services. Il faut alors raisonner sur des abstractions, ou sur un sous-ensemble du langage pour que ces problèmes redeviennent décidables.

L'équipe Distribcom a étudié ces questions pour plusieurs modèles de documents actifs manipulant conjointement contrôle (workflow) et données. Parmi les modèles étudiés, on peut citer Active XML [1], les Tree Pattern Rewrite Systems [4,5], ou plus récemment les Docnets [7], une variante de réseaux de Petri. Il ressort de ces études, que les documents actifs peuvent être vus de manière assez générale comme des systèmes de réécriture distribués communicants.

Un premier objectif de ce post doc est de poursuivre l'étude du concept de document actif sous cet angle de systèmes de réécriture communicants (SRC). Les challenges à atteindre sont de définir les bonnes propriétés de tels systèmes permettant :

- De composer des SRC pour obtenir des systèmes plus grands
- D'assurer la décidabilité de la composition de SRCs
- De vérifier certaines propriétés logiques des systèmes modélisés par des SRC

Le deuxième objectif de ce post doctorat est de contribuer au développement de la plateforme Documents Actifs dans l'équipe Distribcom.

Nous avons développé une infrastructure au-dessus de REST, et mis en place une petite plateforme expérimentale de quatre machines qui manipulent conjointement des services REST et des documents actifs. Nous comptons ouvrir cette plateforme à l'utilisation de formalismes d'orchestration tels que le langage d'orchestration Orc, développé à Austin [3]. Nous comptons également intégrer dans cette plate-forme les résultats de nos études concernant la Qualité de Service et les contrats (SLA). Enfin, les exemples utilisés à ce jour sont de petits modèles jouets. Nous souhaitons développer des cas d'études exploitant les caractéristiques des documents actifs : distribution, mélange contrôle/données,...

#### Contacts :

Pour tout renseignement sur le poste, envoyez un courriel à Loïc Hélouët ([loic.helouet@inria.fr](mailto:loic.helouet@inria.fr))

#### Candidature:

Envoyer une lettre de motivation, un CV, et des références à [loic.helouet@inria.fr](mailto:loic.helouet@inria.fr) .

#### Profil recherché:

Le candidat doit posséder un doctorat, et avoir effectué sa thèse dans le domaine des Web-services, des méthodes formelles, ou des bases de données. Par ailleurs, le candidat doit avoir un certain goût pour les aspects théoriques de l'informatique distribuée.

Le langage de développement du projet est JAVA, des compétences dans ce domaine seront appréciées. Une expérience dans le domaine des Web-Services sera également bienvenue.

La connaissance de l'anglais est indispensable.

Ce poste est ouvert à tout candidat sans restriction de nationalité. La connaissance du français est un plus mais n'est pas obligatoire.

#### Description du poste :

Le poste proposé est un contrat d'un an au sein de l'INRIA Rennes, dans l'équipe DISTRIBCOM. Des interactions avec d'autres chercheurs, notamment des équipes S4, Vertecs de l'IRISA, ainsi qu'avec d'autres partenaires académiques et industriels sont également prévues. Les tâches prévues dans le cadre de ce poste sont :

- La contribution aux études effectuées dans Distribcom autour des Web-Services et des documents actifs.

- La contribution au développement de la plateforme Web-Services de Distribcom : intégration de nouvelles fonctionnalités, de nouveaux modèles, réalisation de cas d'étude, ....

Tous les développements se feront en JAVA, en utilisant REST comme infrastructure de service. Les prototypes étant destinés à être distribués, une attention particulière sera portée à la portabilité des solutions techniques envisagées, à la documentation du code et des logiciels produits.

#### Bibliographie :

[1] Serge Abiteboul, Luc Segoufin, Victor Vianu: Static analysis of active XML systems. PODS 2008: 221-230

[2] Loïc Hélouët, Albert Benveniste, Distributed active XML and service Interfaces, INRIA reserach report no 7082, 2009.

[3] David Kitchin, William R. Cook, Jayadev Misra: A Language for Task Orchestration and Its Semantic Properties. CONCUR 2006: 477-491.

[4] Blaise Genest, Anca Muscholl, Zhilin Wu. Verifying Recursive Active Documents with Positive Data Tree Rewriting, FSTTCS 2010.

[5] Blaise Genest, Anca Muscholl, Olivier Serre, Marc Zeitoun. Tree Pattern Rewrite Systems. ATVA 2008.

[6] Albert Benveniste, Loïc Hélouët, Document Based Modeling of Web Services Choreographies Using Active XML, ICWS 210, Miami, Florida, 2010.

[7] Benoît Masson, Loïc Hélouët, Albert Benveniste. Compatibility of Data-Centric Web Services, WS-FM 2011.