Université Paris-Sud

École Doctorale Informatique de Paris-Sud

Laboratoire de Recherche en Informatique

Résumé de Thèse

Une Approche Symbolique pour la Vérification et le Test des Chorégraphies de Services

soutenue le 31 Octobre 2013

par

Hữu Nghĩa NGUYỄN

pour l'obtention du Doctorat en Informatique

Composition du Jury:

Rapporteurs:	Manuel Núñez	– Universidad Complutense de Madrid, Spain
	Gwen Salaün	– Grenoble Inp, Inria, France
Examinateurs:	Philippe Dague	– Université Paris Sud , France
	Pascal Poizat	– Université Paris Ouest Nanterre la Défense, France
	Fatiha Zaïdi	– Université Paris Sud, France
	Gianluigi Zavattaro	– Università di Bologna, Italy

Superviseurs: Pascal Poizat et Fatiha Zaïdi (directrice)

Janvier 2010 – Octobre 2013

1 Problématique

Contexte. L'ingénierie orientées services est un nouveau paradigme pour développer des logiciels distribués et collaboratives. Un tel logiciel se compose de plusieurs entités, appelés *services*, chacun d'entre eux étant par exemple une application Web, un service Web, ou même un utilisateur. Les services peuvent être développés indépendamment et sont composées pour atteindre quelques exigences. Les *chorégraphies de services* définissent ces exigences avec une perspective globale, basée sur les interactions entre des participants qui sont implémentés en tant que services.

Une *description* de chorégraphie spécifie les interactions entre les *rôles* d'une collaboration. Une *implémentation* de chorégraphie est un ensemble des services qui réalisent les comportements des rôles. Dans cette thèse, nous supposons qu'un rôle peut être joué par un ou plusieurs services.

Les développements des chorégraphies de services peuvent suivre une descendante approche ou une ascendante approche. Dans les *descendantes approches*, on projette la description de la chorégraphie sur des rôles pour obtenir leur comportements qui sont ensuite implémentés en des services. Dans les *ascendantes approches*, on essaie de composer des services existes tel que leur comportements sont attendus par la description.

Il y a trois concepts concernant ces approches: réalisabilité, projection et conformité. La *réalisabilité* est utilisée dans descendante approche pour répondre la question: la spécification peut être réalisé ou pas. La *projection* génère des comportements de chaque rôle à partir d'une spécification. La *conformité* vérifie qu'une implémentation est satisfaite des exigences prévues dans la spécification ou pas. La réalisabilité et la projection sont réalisées aux niveaux de modèles, en se basant sur des modèles des spécification. La conformité est réalisée par soit la vérification au niveau modèle si on a modèle de l'implémentation soit par le test de l'implémentation réelle.

Problématique. Lorsque le modèle de la chorégraphie et sa mise en œuvre sont disponibles, certaines questions concernant le développement de la chorégraphie peuvent être effectuées au niveau du modèle. *Interactions* entre des partenaires d'une chorégraphie sont généralement réalisés par des échanges de messages, par exemple, SOAP (Simple Object Access Protocol) messages dans les services Web. En conséquence, c'est critique de *supporter des passages de valeurs à la modélisation et à l'analyse des chorégraphies.*

La plupart des approches existantes ne prennent pas adéquatement des passages de valeurs, c'est à dire, ne pas prendre en compte les données échangées par des interactions et de l'utiliser pour sélectionner des branchements. Elles abstraient des données. Cela peut générer des problèmes d'approximations, par exemple, des *faux négatifs* dans le processus de vérification. Par conséquent, la présence du passage de valeurs dans le modèle de la chorégraphie peut changer des verdicts de vérifications, par exemple, la réalisabilité.

Certaines approches supportent des passages de valeurs en travaillant sur les systèmes fermés, c'est à dire, les passages de valeurs sont des constantes. Dans un tel cas, cela pourrait conduire à des problèmes d'efficacité graves, *l'explosion combinatoire*, lors de l'analyse de la chorégraphie. Quelques approches évitent ce problème en analysant syntaxiquement une chorégraphie lors de la vérification réalisabilité. Cependant, ces approches manquent les cas de inaccessible et conditionnel.

Après qu'une chorégraphie a été mise en œuvre, la mise en œuvre doit être vérifiée pour se conformer à la spécification et pour s'assurer qu'elle fonctionne proprement. Lorsque le modèle de mise en œuvre n'est pas disponible, la conformité entre la mise en œuvre et sa spécification peut être garanti par des tests, aussi connu comme le test de conformité. Des testes consistent à effectuer des expériences sur des mise en œuvre sous test (IUT). Une mise en œuvre de la chorégraphie est un ensemble de services qui peuvent être développées indépendamment, et puis composées pour réaliser un objectif commun. Elle est généralement déployée sur un système distribué composé

des machines faiblement couplés qui sont physiquement distribués et ne partagent pas les ressources, mais plutôt sont connectés via des échanges de messages.

Dans un contexte distribué, il est difficile, voire impossible dans certains cas, pour contrôler l'ensemble de IUT, par exemple, le système qui fonctionne 24/7. Par conséquent, des tests ne doivent pas perturber leurs fonctionnements naturels, car ils pourraient produire des mauvais comportements. En d'autres termes, le testeur doit réaliser le test uniquement en observant le comportement de l'IUT, par exemple, des entrants et des sorties de IUT, par le point d'observations (OP).

Les observations pourraient ne pas être complète. C'est causée par des événements non-observables ou des occurrences partielles des observations, à savoir , le manque d' horloge commun de système distribué. Par conséquent, le testeur n'a qu'une vue partielle ou un ensemble de vues partielles de l'IUT.

2 Approches Proposées

Utilisation de la technique symbolique en modélisation et en vérification. Dans notre environnement intégré, les données sont pris en charge en utilisant la technique symbolique et un solveur SMT. Nous représentons des paramètres des messages et les variables par des valeurs symboliques au lieu de celles en concret. Celle-là nous permet de réduire les faux-négatifs et de contourner le problème d'explosion combinatoire de l'espace d'états, ces problèmes sont durs à l'abstraction et à l'énumération des valeurs pour les approches existantes basées données.

Utilisation du test passif boîte noire pour tester la conformité. Le test passif est une méthode de test qui vise à ne pas perturber l'exécution de IUT pendant le processus de test, et ne pas rendre indisponible l'implémentation pendant le processus de test. En plus, le test passif boîte noire n'ai pas besoin d'accéder au code source. Au contraire, des tests actifs nécessitent de contrôler IUT en envoyant des entrées à

IUT et en observant des sorties.

3 Contributions

Les contributions de la thèse sont multiples. Elles peuvent être regroupées en trois grandes catégories.

La première contribution est un modèle symbolique et un environnement intégré pour spécifier et analyser les chorégraphies de services. En particulier, nous avons défini un langage formel avec un modèle d'interaction qui permet à spécifier à la fois des points de vue globale (chorégraphie) et local (exigences de rôle ou description du service). Notre langue supporte l'échange d'informations et les constructions liées aux données (structures conditionnelles et boucles). Nous donnons également une sémantique symbolique à cette langue en utilisant une transformation de modèle vers graphes des transitions symboliques, évitant ainsi l'extraction des données et sur-approximation , la restriction par des domaines de données liés à la main, et la limitation de descriptions fermés niveau de mise en œuvre.

Nous avons proposé ensuite un environnement intégré symbolique dans lequel les variables sont manipulés en utilisant des symboles plutôt que leurs valeurs concrètes. Celui-ci permet de modéliser et d'analyser des chorégraphies en présence de données sans souffrir de l'explosion combinatoire de l'espace d'états et sans limite leurs domaines de données . L'environnement intégré est utilisé pour vérifier la réalisabilité, la conformité, et la projection.

Nous ne vérifions pas seulement la réalisabilité d'une chorégraphie, mais nous proposons aussi des solutions pour être réalisable. Pour ce faire, nous construisons une fonction de projection intelligente. Elle peut introduire certaines interactions supplémentaires quand la chorégraphie est irréalisable. En outre, les interactions supplémentaires sont minimales afin de minimiser la mise en œuvre et le trafic.

La projection prend en compte les deux modes de communications : synchrone et asynchrone.

La relation de conformité a été construite en se basant sur la bi-simulation faible. Elle permet de vérifier la conformité à la présence de la chorégraphie raffinement, c'est à dire, où de nouveaux services et / ou des interactions peuvent être ajoutés par rapport à la spécification. Allant plus loin que un résultat de conformité vrai ou faux, notre approche prend en charge la génération de la contrainte la plus générale sur les informations échangées pour atteindre la conformité.

La seconde contribution est liée au dépistage passif des chorégraphies de services. Notre travail a été commencé par les tests passifs des conformités des chorégraphies. Les traces d'exécutions de IUT seront examinées par rapport aux spécifications de chorégraphies pour émettre des verdicts. La conformité est vérifiée à la fois au niveau local et au niveau mondial. La conformité locale représente qu'une implémentation de service joue correctement ou non son rôle dans la chorégraphie. La conformité globale assure que les interactions entre les services respectent l'ordre de la chorégraphie ou si elles divergent de la collaboration envisagée.

Les limites de l'approche de test ci-dessus, par exemple, les tests hors-ligne, sans des passages de valeurs, et l'exigence d'une horloge globale pour synthétiser des traces locaux, ont été surmontés. Il est basé sur une approche de test passif orienté des propriétés,c'est à dire, seules traces d'exécution qui concernent les propriétés seront examinées. Une propriété peut exprimer un critique comportement, positive ou négative, à tester sur un service isolé (localement) ou sur un ensemble de services (globalement). Ce travail prend en charge la vérification en ligne, c'est à dire, les défauts sont détectés dès qu'ils sont générés, de ce genre de propriétés sont vérifiées par rapport à des traces de fonctionnement locales de chaque service dans un système distribué où aucune horloge mondiale est nécessaire .

La dernière contribution de cette thèse est la disponibilité des outils supportés. Les théories proposées sont totalement équipée par nos outils qui sont

disponibles en téléchargement ou à utiliser en ligne. Figure 1 représente l'outil SChorA, à l'adresse http://schora.lri.fr, qui permet de vérifier la conformité, la réalisabilité et la projection. L'utilisateur interagit directement avec SChorA grâce à une interface graphique de Web. Les outils sont implémentés en environ 34000 lignes de Java.

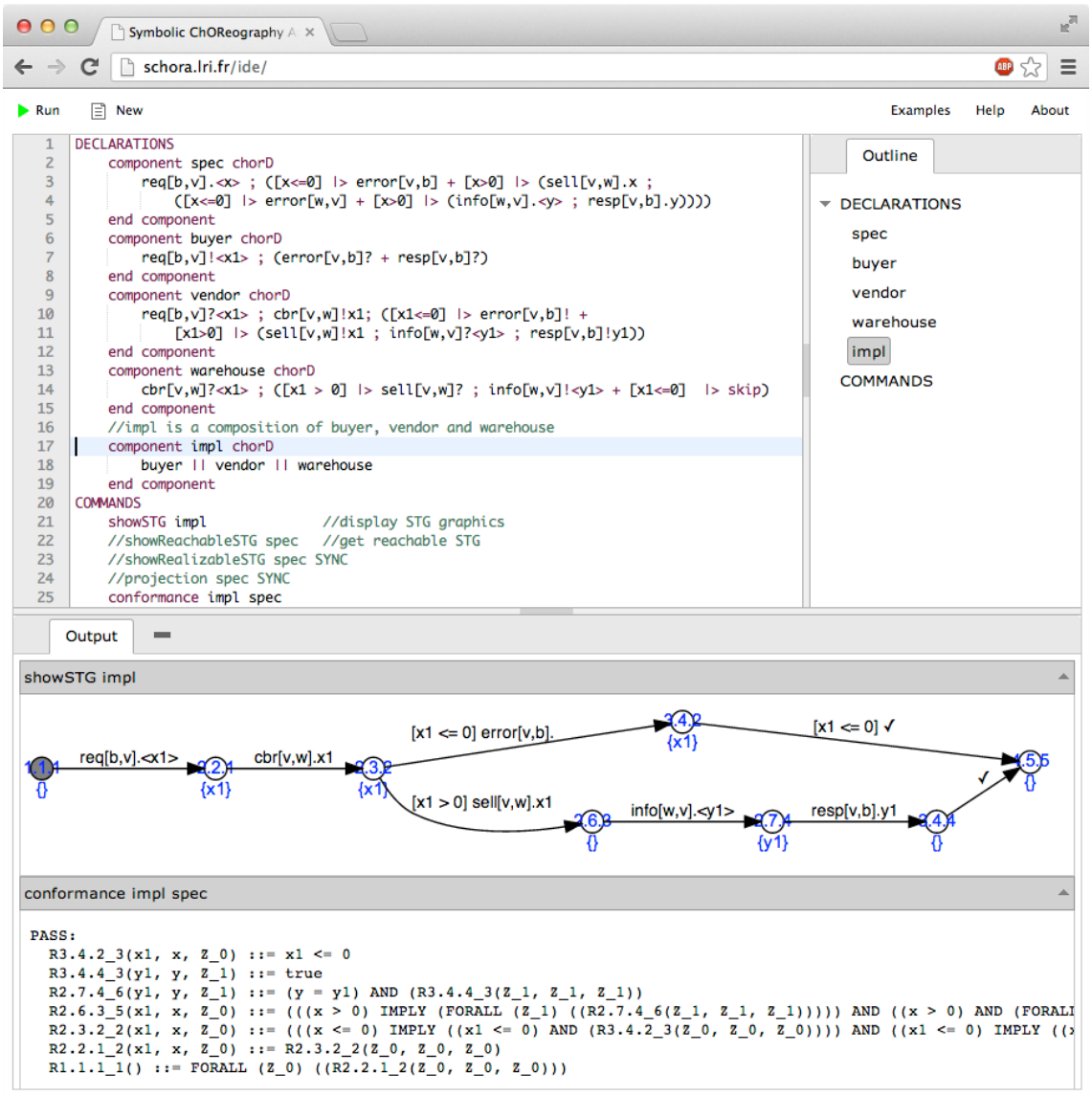

FIGURE 1 – Conformance Checking by SChorA tool

4 Plan du Manuscrit

Le manuscrit de la thèse est organisé en six chapitres comme ci-dessous :

Le chapitre 1 présente les problématiques et donne une d'ensemble du manuscrit.

Le chapitre 2 présente un état de l'art de la chorégraphie de services. Nous allons à partir des notions générales de chorégraphies de services à un aperçu des approches existantes pour modéliser et pour tester des chorégraphies.

Le chapitre 3 présente notre langue de spécification et notre modèle formel des chorégraphies à base d'interaction supportée des passages de valeurs. Les sémantiques symboliques de la langue et le modèle sont également introduites en les traduisant en des graphes des transitions des symboliques. Nous examinons ensuite les problèmes fondamentales de la chorégraphie: la conformité, la projection et la réalisabilité avec la présence des passages de valeurs dans le modèle. Enfin, nous présentons nos outils pour valider le modèle proposé.

Le chapitre 4 introduit nos deux approches de tests passives : une approche naïve et une autre approche liée aux propriétés, pour tester les implémentations réelles de chorégraphies de services. L'approche naïve teste tous les comportements observés de IUT tandis que l'approche liée aux propriétés se concentre uniquement sur certains comportements critiques de IUT. Il y a aussi des améliorations de la première approche par la seconde approche. Les évaluations de deux approches par outils supportés sont également présentées.

Le chapitre 5 présente une application de nos outils, qui sont présentés dans les chapitres ci-dessus, sur le processus de développement d'une étude de cas simple. Il se compose de la spécification et la modélisation de l'étude de cas par notre langue et notre modèle. Il est ensuite vérifié par l'outil SChorA qui analyse, en utilisant certains attributs tels que la possibilité de réalisation, l'accessibilité, la conformité, et la projection. Enfin, deux outils de tests passifs seront introduits pour garantir l'exactitude de la mise en œuvre.

Le chapitre 6 conclut cette thèse. Les limites de la thèse sont discutées et des travaux futurs sont également soulignés.

5 Publications

Les contributions principales de cette thèse ont déjà été publiés dans les actes de conférences internationales ainsi que présenté en jours de recherche nationaux.

Conférences Internationales.

1. Huu Nghia Nguyen, Pascal Poizat and Fatiha Zaïdi.*Automatic Skeleton Generation for Data-Aware Service Choreographies*. in ISSRE - IEEE International Symposium on Software Reliability Engineering, pages 320-329. November 2013.
2. Huu Nghia Nguyen, Pascal Poizat and Fatiha Zaïdi.*Online Verification of Value-Passing Choreographies through Property-Oriented Passive Tesing*. in HASE - IEEE International Symposium on High Assurance Systems Engineering, pages 106-113. October 2012.
3. Huu Nghia Nguyen, Pascal Poizat and Fatiha Zaïdi. *A Symbolic Framework for the Conformance Checking of Value-Passing Choreographies*. in ICSOC - International Conference on Service Oriented Computing, pages 525-532. November 2012.
4. Huu Nghia Nguyen, Pascal Poizat and Fatiha Zaïdi. *Passive Conformance Testing of Service Choreographies*. in SAC - ACM Symposium on Applied Computing, pages 1528-1535. March 2012.

Présentations.

1. *A Symbolic Framework for the Conformance Checking of Value-Passing Choreographies*. in Journées du GDR GPL, MTV2 track. April 2013.

2. *Vérification des chorégraphie implantant en BPEL.* in Journées du GDR GPL, COSMAL track. June 2011.

Poster.

1. *Symbolic Approach for the Verification and the Test of Service Choreographies.* in Journées du GDR GPL. April 2013.