

VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wānanga o te Ūpoko o te Ika a Māui



School of Engineering and Computer Science
Te Kura Mātai Pūkaha, Pūrorohiko

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341
Fax: +64 4 463 5045
Internet: office@ecs.vuw.ac.nz

The Title

Alexandre Sawczuk da Silva

Supervisors: Dr. Hui Ma, Prof. Mengjie Zhang

November 20, 2014

Submitted in partial fulfilment of the requirements for
PhD.

Abstract

A short description of the project goes here.

Contents

1	Introduction	1
1.1	Problem Statement	1
1.2	Motivations	1
1.3	Research Goals	1
1.4	Organisation of Proposal	3

Figures

Chapter 1

Introduction

1.1 Problem Statement

1.2 Motivations

1.3 Research Goals

- *How can EC approaches to automated Web service composition take user preferences into account?*

User constraints and preferences are a common requirement when performing Web service composition. For example, consider the case of a user who wishes to book transportation and accommodation for a trip, including transportation from the airport to the hotel, a room at the hotel for a given number of days, and transportation back to the airport [3]. In this scenario, the user is likely to have constraints on the attributes of a given service (e.g. the hotel booking service must either be that of a two-star or that of a three-star hotel), as well as conditional constraints (e.g. if the hotel booking service is for a two-star hotel, then the transportation to and from the airport should be arranged using a taxi service, otherwise a shuttle service should be used). Different techniques have been explored to achieve compositions that consider such constraints, including the use of AI planning with rules encoding user constraints [2] and the representation of the composition as a constraint satisfaction problem to be processed with a solver engine [5]. However, this territory has not been widely explored by applying evolutionary computation techniques (a GA approach for composition that considers inter-service relationships has been proposed [14], even though no actual examples of these relationships were presented). Due to the flexibility and efficiency of these techniques, it would be interesting to focus on the investigation of ways in which to extend them to apply these constraints.

The specification of user constraints is closely connected with the semantic representation of Web services. In the example presented earlier, where the user specifies constraints based on an attribute of a hotel service (the hotel's number of stars), the assumption is that this information is in fact available for that service. This type of information, which encodes details about the underlying domain of a service, is what is referred to as semantics. Though standards for semantically describing a Web service exist [8], semantic Web service composition is currently geared towards the utilisation of planning techniques and semantic reasoning engines [9]. Therefore, further research on evolutionary computation applied to semantic Web service composition would also contribute to enabling the specification of user constraints.

- *Is it possible to take user constraints into account while at the same time optimising potentially conflicting QoS preferences?*

One possible way of achieving optimisation of these different dimensions simultaneously is through the use of multi-objective techniques.

The following types of constraints and user preferences have been presented in literature:

- Conditional branch structures that reflect user preferences [11]. The paper covers two types of user preferences that require branching. The first type is when the user prefers one service instead of another according to a condition (i.e. the if-else construct – note that this construct eventually closes into a diamond, and only one output is produced). The second type is when the user specifies a list of services with similar functionalities ranked according to personal preferences. If the service with the highest priority fails, then the service with the second-highest priority will be executed (e.g. PayWithCard service with higher priority, PayInCash service with lower priority).
- Preferences specified using a logic language (based on linear temporal logic) [10]. For example, specify that you do not want to book a Hilton hotel, but you want a 3-star hotel paid using a credit card (this language requires relatively detailed semantic information about services). Another example: prefers not book air ticket until the hotel has been booked (order of service execution).
- Hard constraints on services: service properties (i.e. service must have a specific property with a specific value) and coreography details (essentially if-else constraints that close into a diamond) [3].
- Preferences specified using the Knowledge Interchange Format (KIF) language, as it provides a well-defined syntax and semantic that can be applied to constraints [4]. The types of preferences that can be defined are the same as those in [10].
- Temporal and causality constraints: Constraints on the flow structure of the composition (e.g. if service 1 executes, then service 2 must also execute) [5].
- Logical constraints: any logical expressions on integer or string values, e.g. Hotel should be in Barcelona and the cost should be under \$450 a day [5].
- If/then constraints: if a constraint X holds, then another constant Y must hold as well. E.g. If a hotel has less than 4 stars, then its cost a day must be under 100 dollars [5].
- Preferences specified using PDDL3 (Planning Domain Definition Language). In PDDL3, preferences are described using logical formulae (i.e. logical statements encoding constraints) and temporal preferences (i.e. how often a constraint must hold as the planning steps through its states) [6].
- Data flow constraints of the composition specified using a visual language. For example, if a room is available book it, otherwise cancel the entire transaction. The basic idea is to check if the data net of constraints is satisfied by the plan presented as the solution, with some heavy formalisation [7].
- Use of a representation that employs domain objects, allowing the specification of control flow requirements and also of logical constraints [1].

- Global constraints based on attributes of a single service (e.g. only one service that costs 50 may be invoked). Solved using ILP [13].
- Constraints on the resources consumed while creating the composition (note they are not part of the solution itself) [12].

1.4 Organisation of Proposal

Bibliography

- [1] BERTOLI, P., KAZHAMIKIN, R., PAOLUCCI, M., PISTORE, M., RAIK, H., AND WAGNER, M. Control flow requirements for automated service composition. In *Web Services, 2009. ICWS 2009. IEEE International Conference on* (2009), IEEE, pp. 17–24.
- [2] BOUSTIL, A., MAAMRI, R., AND SAHNOUN, Z. A semantic selection approach for composite web services using OWL-DL and rules. *Service Oriented Computing and Applications* 8, 3 (2014), 221–238.
- [3] BOUSTIL, A., SABOURET, N., AND MAAMRI, R. Web services composition handling user constraints: towards a semantic approach. In *Proceedings of the 12th International Conference on Information Integration and Web-based Applications & Services* (2010), ACM, pp. 913–916.
- [4] GAMHA, Y., BENNACER, N., NAQUET, G., AYEB, B., AND ROMDHANE, L. B. A framework for the semantic composition of web services handling user constraints. In *Web Services, 2008. ICWS'08. IEEE International Conference on* (2008), IEEE, pp. 228–237.
- [5] KARAKOC, E., AND SENKUL, P. Composing semantic web services under constraints. *Expert Systems with Applications* 36, 8 (2009), 11021–11029.
- [6] LIN, N., KUTER, U., AND SIRIN, E. *Web service composition with user preferences*. Springer, 2008.
- [7] MARCONI, A., PISTORE, M., AND TRAVERSO, P. Specifying data-flow requirements for the automated composition of web services. In *Software Engineering and Formal Methods, 2006. SEFM 2006. Fourth IEEE International Conference on* (2006), IEEE, pp. 147–156.
- [8] MARTIN, D., BURSTEIN, M., MCDERMOTT, D., MCILRAITH, S., PAOLUCCI, M., SYCARA, K., MCGUINNESS, D. L., SIRIN, E., AND SRINIVASAN, N. Bringing semantics to web services with owl-s. *World Wide Web* 10, 3 (2007), 243–277.
- [9] SABOOHI, H., AND KAREEM, S. A. World-altering semantic web services discovery and composition techniques-a survey. In *The 7th International Conference on Semantic Web and Web Services (SWWS)* (2011), pp. 91–95.
- [10] SOHRABI, S., PROKOSHYNA, N., AND MCILRAITH, S. A. Web service composition via the customization of golog programs with user preferences. In *Conceptual Modeling: Foundations and Applications*. Springer, 2009, pp. 319–334.
- [11] WANG, P., DING, Z., JIANG, C., AND ZHOU, M. Automated web service composition supporting conditional branch structures. *Enterprise Information Systems* 8, 1 (2014), 121–146.

- [12] XIANG, C., ZHAO, W., TIAN, C., NIE, J., AND ZHANG, J. Qos-aware, optimal and automated service composition with users' constraints. In *e-Business Engineering (ICEBE), 2011 IEEE 8th International Conference on* (2011), IEEE, pp. 223–228.
- [13] YOO, J. J.-W., KUMARA, S., LEE, D., AND OH, S.-C. A web service composition framework using integer programming with non-functional objectives and constraints. *algorithms* 1 (2008), 7.
- [14] ZHANG, Z., ZHENG, S., LI, W., TAN, Y., WU, Z., AND TAN, W. Genetic algorithm for context-aware service composition based on context space model. In *Web Services (ICWS), 2013 IEEE 20th International Conference on* (2013), IEEE, pp. 605–606.