WEB SERVICES CHOREOGRAPHY TESTING USING SEMANTIC SERVICE DESCRIPTION

HAZLIFAH BINTI MOHD RUSLI

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Doctor of Philosophy (Computer Science)

Faculty of Computing
Universiti Teknologi Malaysia

FEBRUARY 2017

DEDICATION

To my husband Khairil Anwar and
children Nur Khairina, Hazman and Hafiz

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to my principle supervisor, Prof. Dr. Suhaimi Ibrahim and co-supervisor Associate Prof. Dr. Mazidah Puteh for their continued support and invaluable constructive comments which have helped me in completing this research. I am also indebted to Dr. Sayed Gholam Hassan Tabatabaei who inspired me in selecting this research area, and Dr. Mohd Naz'ri Mahrin for his vivacious discussions that kept my focus on the research problems at hand.

I also would like to thank my two special colleagues Dr. Kanmani, Dr. Noorihan and Dr. Usama El-Hagari for their stimulating conversations and taking interest in my research work. I am also thankful to Universiti Teknologi MARA and Ministry of Higher Education for funding my studies for this PhD programme. My special gratitude goes to my parents, my aunt Siti, parents-in-law, and brothers for their encouragement at all times. My special thanks also go to all the staff at AIS, co- researchers at UTM and colleagues at Universiti Teknologi MARA Dungun and Jasin, and my 'UiTM ISE sisters' who have helped me in many ways to complete this research.

At last and most importantly, this programme would not have been possible without the support of my husband, Encik Khairil Anwar, and my mother Puan Haziyah Hashim. I am thankful for their endless prayer, patience, love, consistent support and motivation for me to be where I am at present.

ABSTRACT

Web services have become popular due to their ability to integrate with and to interoperate heterogeneous applications. Several web services can be combined into a single application to meet the needs of users. In the course of web services selection, a web candidate service needs to conform to the behaviour of its client, and one way of ensuring this conformity is by testing the interaction between the web service and its user. The existing web services test approaches mainly focus on syntax-based web services description, whilst the semantic-based solutions mostly address composite process flow testing. The aim of this research is to provide an automated testing approach to support service selection during automatic web services composition using Web Service Modeling Ontology (WSMO). The research work began with understanding and analysing the existing test generation approaches for web services. Second, the weaknesses of the existing approaches were identified and addressed by utilizing the choreography transition rules of WSMO in an effort to generate a Finite State Machine (FSM). The FSM was then used to generate the working test cases. Third, a technique to generate an FSM from Abstract State Machine (ASM) was adapted to be used with WSMO. This thesis finally proposed a new testing model called the *Choreography to Finite State Machine* (C2FSM) to support the service selection of an automatic web service composition. It proposed new algorithms to automatically generate the test cases from the semantic description (WSMO choreography description). The proposed approach was then evaluated using the Amazon E-Commerce Web Service WSMO description. The quality of the test cases generated using the proposed approach was measured by assessing their mutation adequacy score. A total of 115 mutants were created based on 7 mutant operators. A mutation adequacy score of 0.713 was obtained. The experimental validation demonstrated a significant result in the sense that C2FSM provided an efficient and feasible solution. The result of this research could assist the service consumer agents in verifying the behaviour of the Web service in selecting appropriate services for web service composition.

ABSTRAK

Perkhidmatan web telah menjadi semakin digemari disebabkan oleh kemampuan mereka untuk berintegrasi dan mengoperasikan pelbagai aplikasi yang berbeza. Beberapa perkhidmatan web boleh digabungkan ke dalam satu aplikasi untuk memenuhi keperluan pengguna. Dalam proses pemilihan perkhidmatan web, calon perkhidmatan web perlu mematuhi tingkah laku perkhidmatan web pelanggan, dan salah satu cara untuk memastikan pematuhan ini adalah dengan menguji interaksi antara perkhidmatan web dan penggunanya. Kebanyakan teknik ujian perkhidmatan web yang sedia ada hanya memberi tumpuan kepada penerangan perkhidmatan web berdasarkan sintaks, manakala kebanyakan teknik berasaskan semantik hanya mengambil kira ujian aliran kerja rencam. Matlamat penyelidikan ini adalah untuk menyediakan pendekatan generasi ujian automatik yang dapat menyokong pemilihan perkhidmatan yang sesuai semasa penyepaduan perkhidmatan web secara automatik menggunakan Ontologi Pemodelan Perkhidmatan Web (WSMO). Penyelidikan ini dimulakan dengan memahami dan menganalisa pendekatan generasi ujian automatik yang sedia ada dalam perkhidmatan web. Kedua, kelemahan pendekatan yang sedia ada telah dikenal pasti dan ditangani dengan menggunakan peraturan peralihan koreografi WSMO dalam usaha untuk menjana sebuah Mesin Keadaan Terhingga (FSM). FSM ini seterusnya digunakan untuk menjana kes-kes ujian kerja. Ketiga, teknik untuk menjana FSM dari Mesin Keadaan Niskala (ASM) disesuaikan untuk digunakan bersama WSMO. Tesis ini akhirnya mencadangkan satu model pengujian baru yang dikenali sebagai Koreografi kepada Mesin Keadaan Terhingga (C2FSM) untuk menyokong pemilihan perkhidmatan sewaktu penyepaduan perkhidmatan web automatik. Algoritma-algoritma baru dicadangkan untuk menjana kes-kes ujian secara automatik dari perihal semantik (perihal koreografi WSMO). Pendekatan yang disyorkan kemudiannya dinilai dengan menggunakan perihal WSMO Perkhidmatan Web E-Dagang Amazon. Kualiti kes-kes ujian yang dihasilkan menggunakan pendekatan yang dicadangkan itu diukur dengan menilai skor kecukupan mutasinya. Sebanyak 115 mutan dicipta berdasarkan 7 pengendali mutan. Sebanyak 0.713 skor kecukupan mutasi diperolehi. Pengesahan eksperimen menunjukkan hasil yang bermakna dalam erti kata bahawa C2FSM mampu menyediakan penyelesaian yang berkesan dan boleh dilaksanakan. Hasil kajian ini boleh membantu ejen perkhidmatan pengguna dalam mengesahkan kelakuan perkhidmatan web dalam memilih perkhidmatan yang sesuai untuk komposisi perkhidmatan web.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ASM	-	Abstract State Machine
C2FSM		Choreography to Finite State Machine
IOPE	-	Input, Output, Precondition and Effect
OWL	-	Web Ontology Language
RDF	-	Resource Definition Framework
RDFS	-	Resource Definition Framework (Scheme)
SAWSDL	-	Semantic Annotation for WSDL
SOAP	-	Simple Object Access Protocol
SWRL	-	Semantic Web Rule Language
SWS	-	Semantic Web Service
SWST	-	Semantic Web Services Testing
UDDI	-	Universal Description, Discovery, and Integration
W3C	-	World Wide Web Consortium
WS	-	Web Service
WSDL	-	Web Services Description Language
WSMF	-	Web Service Modeling Framework
WSML	-	Web Service Modeling Language
WSMO	-	Web Service Modeling Ontology
WSMX	-	Web Service Execution Environment
WWW	-	World Wide Web

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

This chapter begins with a brief introduction on the subject of the research, i.e. web services choreography testing using semantic behavioural description. Firstly, the background of the problem is described and statement of the problem is defined. This is then followed by the objectives and scope of this research. The final section contains the significance of this research and brief descriptions of some important terms that are used in this research.

1.1 Background of the Problem

Web services are Internet-based, modular application that uses the Simple Object Access Protocol (SOAP) for communication and transfers data in XML through the Internet [1]. It is another distributed computing technology (like CORBA, RMI, and EJB) that allows the creation of client/server application. However unlike CORBA, RMI, and EJB, web services allow easy integration and interoperability between heterogeneous applications. Web services are platform and language independent, meaning that a client program written in C running on Windows can communicate with a web service application programmed in Java running on Linux. This ensures that a company's application can talk to its business partner's application, even though they are written in different languages and run on different platforms. Throughout this thesis, the terms web service and service are used interchangeably.

In general, most web service architecture are based on the requester, provider and registry components. Whenever a requester or a client requires a particular web service, the client will search for the web service by querying the registry. The registry will then respond by providing the necessary service descriptions. The requester then selects the most suitable web service and binds to its provider by invoking it. The web service description or the Web Service Description Language (WSDL), contains information that allows the requester to invoke the web service. The basic web service usage process consists of the provider publishing its services, requester discovering these services and selecting them, requester and provider negotiating Service Level Agreements (SLAs) and finally requester invoking the services. If no suitable services are found, the requester needs to compose several services by discovering, selecting and determining the order of invocation of those services [2]. Composing web services can be done manually, semi-automatically or automatically.

Manual Web service composition process is where a human designer creates an abstract composite process, searches for and selects suitable services, and finally binds them to the abstract process The tasks involved in manually composing a Web service is time-consuming and error-prone. Automatic web services composition automatically generates a composite service specification by leveraging on the semantic Web and artificial intelligence (AI) planning techniques [3]. The idea is to utilise the semantic description of the web services during the discovery and selection process and to generate a plan using AI planning based on requester's needs.

Research on web services have often focused on issues such as web service discovery, selection, composition and mediation. Of late, research in the area of testing web services have also gained much interest. Web services testing is an important aspect of web services as it provides a way to ensure the trustworthiness of a web service-based systems, in the sense that the web service does what it says it will do in its service description or whether the created composite service behaves as what is required. If a client is not able to trust a service from a service provider, the

client will not use it. Testing is also necessary to ensure that the service chosen best satisfies the user's needs in terms of functionalities and quality.

Unlike traditional software systems testing, there exists several issues that set web service testing apart from traditional testing. First, traditional testing is often performed statically and offline, whilst web service testing must be done dynamically and in real time [4]. Secondly, web services have no graphical user interfaces. They are only accessible via interfaces published in web services specification (e.g., WSDL), and via network protocols (e.g., SOAP) [5]. Thirdly, only the web service specification is available to describe a web service. No source code is available [1]. All these make it difficult to perform testing tasks.

1.2 Problem Statement

Designing and creating good quality test cases highly depend on information that is available regarding the web services under test, and most often than not, it requires the knowledge, instinct and skill of the tester on how the web services behave and interact. Several WSDL-based test case generation approaches have been proposed such as generating test data from XML schema [6, 7], generating test data from formal model of XML schema [8, 9], generating test operations [7], generating SOAP messages [10], and generating test data using data pertubation technique based on WSDL [11].

Unfortunately, WSDL lacks behavioural description, i.e. the definition of preconditions and effects of each web service operation as well as the implied sequencing of these operations. Behavioural specifications are useful in cases where web services assuming an interaction protocol (stateful web services) and web services operating on persistent data [12]. There have been efforts by the testing research community to complement the WSDL file with additional information to support the automatic generation of test cases. Bertolino *et al.* [13] augmented WSDL with Protocol State Machine to verify the interaction between web services to

be registered with already registered services. Keum *et al.* [14] augmented WSDL with Extended Finite State Machine (EFSM) to describe the temporal ordering information of the service behaviour. A proposal was made by Tsai *et al.* [15] to extend WSDL with information such as input-output dependency, invocation sequences, hierarchical functional description and concurrent sequence specifications.

However, these additional information are mostly syntax based, not semantic-based descriptions. Although, current syntax-based approaches allows automation of test case generation, it does not support the understanding of the information in the web services description that can allow intelligent agents to discover these services, test them, and finally select the most suitable service to be used in the composition. Having a semantic-based description allows semantic testing agents to read and reason with the descriptions to allow automatic test generation, execution and analysis, thus, support automatic composition of web services. Therefore these semantic descriptions not only allow the automation of web service discovery, selection and composition, they also can be used to support test automation during automatic web services composition.

Current research on testing of semantic web services focuses on issues such as mutation testing [16-18], test case evaluation [19], test selection [20], test prioritization [21], test reduction [22] and test case generation [23-40]. Research work done on semantic-based test case generation dealt with using different semantic service description and testing different aspects of the web services. The test approaches and techniques were proposed to test the functional behaviour of a single operation within a service [30-34], the flow of a composite process [23, 25, 26, 28, 29, 36, 38-40] and the functional conformance of a stateful service [24, 27, 35, 37].

The existing semantic web service test case generation approaches were based on OWL-S, WSMO, WSDL-S and IOPE descriptions. OWL-S are upper ontology for services written using OWL, WSMO stands for Web Service Modeling Ontology, WSDL-S stands for Web Service Description Language Semantics whilst IOPE stands for Input, Output, Precondition and Effect. Most of the test case

generation approaches were derived from OWL-S specification [25, 26, 28, 36, 38, 39] or extended versions of OWL-S specification [23, 40] of the services to be tested, followed by WSDL-S [24, 30, 34, 37, 38], WSMO [31-33] and IOPE descriptions [24, 35].

As mentioned earlier in the problem background, automatic composition of web services involves generating a plan using AI planning. This plan acts as the composite service specification and the software agent needs to automatically discover and select suitable web services to be included in the planner. Most often than not, the candidate services are selected based on the functional capability of the services. However, the candidate services also needs to be tested to determine whether the candidate service is also suitable in terms of its interaction behaviour. Most existing testing approaches that have been mentioned earlier tested the flow of service composite process using OWL-S process model [23, 25, 26, 28, 29, 36, 38-40]. Although OWL-S process model can be used to describe the interaction protocol between a web service and its client, the testing approaches generated test sequences for the different paths according to the OWL-S process model control construct. The functional testing approach [30-34] is only suitable for testing the behaviour of a single operation of a service. Although there exists testing approaches that tests the operation sequence of a service, the approaches are IOPE-based [24, 27, 35, 37]. An advantage of IOPE-based test approaches is that it is not restricted to only one particular semantic service description. However IOPE-based test approaches lack the framework support on web service discovery, selection, composition and mediation that is offered by WSMO through Web Service Execution Environment (WSMX). Similar to OWL-S process model, WSMO choreography can also be used to describe the interaction protocol. Although WSMO were used in existing test approach, it was used for testing the functional capability of a single operation of a web service [31-33]. It was not used to test the interaction behaviour of the web service.

This research aims to provide an approach for testing web services described using WSMO choreography transition rules and ontology that is able to support service selection in an automatic composition of web services. The approach is called

Semantic Web Services Testing (SWST). SWST makes use of existing Abstract State Machine (ASM) testing technique combined with WSMO reasoner to generate test cases from WSMO web service descriptions.

The following are the challenges of testing the web service interaction behaviour to support automatic web service composition using semantic behavioural description:-

(a) Extraction of relevant information from WSMO web service and WSMO ontology to support testing
(b) Transformation of the web service choreography into a state model.
(c) Generation of test sequences from the state model.

1.3 Research Questions

The ultimate goal of this research is to provide an automated testing approach to support service selection during automatic web services composition using WSMO elements. The output of this research is expected to increase the confidence of the client or the user of the service in other SWS tasks such as discovery, selection and composition. The general research question is as follows:

How can WSMO be used to support test automation for testing the suitability of the behaviour of the web service to support service selection in automatic web service composition?

The following research questions are formulated to address the stated general research question and the discussed problems in this research area:

(a) **RQ1**: Why are the existing approaches still not able to answer current issues in web service test case generation?
(b) **RQ2**: How can the interaction between the client and the candidate service be tested?

(c) **RQ3**: How can the semantic information be used in testing the suitability of a candidate service?

(d) **RQ4**: How can test cases be automatically generated using the proposed approach?

(e) **RQ5**: How to evaluate the quality of test cases generated using the proposed approach to support web service testing?

1.4 Objectives of the Study

The objectives of this study have been derived from the problem statement above. The objectives of this research are to:

(a) To investigate and evaluate the state of the art in web service test case generation approaches.

(b) To formulate and design a new technique for automated web services test case generation approach based on semantic web service behavioural description.

(c) To evaluate the quality of the test cases generated using the proposed approach based on specific and acceptable benchmarks.

1.5 Scope of the Study

The underlined research covers several areas that include web services testing, semantic web services, abstract state machine testing and web services composition. In order to achieve the objectives of the study, the research directions are limited to the following scope of study:

First of all, the research is mainly focused on web services testing, specifically testing the behaviour of the service. This involves understanding current approaches to test case generation, test execution and test analysis relating to web

services testing. This research only focuses on web services testing that use UDDI, WSDL and SOAP technology and not RESTful web services. The research also only covers interaction testing of web services. Mutation testing is employed to evaluate the generated test cases based on benchmarks such as the number of mutants detected by the test cases.

The second direction of this research is related to semantic web services. The basis of semantic web services used in this research is WSMO. Web services are annotated with WSMO web services descriptions. To describe the behaviour of the web service, the transition rules and ontologies are used. These transition rules and ontologies will be used later in generating the test cases. WSMO reasoner is used to reason on the ontologies during transition rules execution to create state models used later in test generation.

The third direction of this research is abstract state machine (ASM) testing. As the WSMO choreography is based on ASM, current approaches to ASM testing is studied and understood. There are several approaches relating to ASM-based testing such as leveraging on state models and model checkers. This research adopts the state model approach, where the specification is transformed to another state machine before test cases are generated.

Finally, web services composition is one area that needs to be understood in order to understand the different testing approaches applied to the composition of web services. The main interest of this research is on testing automatic web services composition approach using artificial intelligence (AI) planning and WSMO. However, the creation of the planner is not part of the scope of this research. The research only demonstrates the testing approach to test web services described using WSMO before it is selected to be put into the planner.

1.6 Significance of the Study

There have been several surveys done on web services testing between the year 2006 and 2013 [41-51]. This shows a growing interest in web services testing research. Out of these surveys, only one survey was on testing of semantic web services with 34 identified studies. The low number of research articles on semantic web service testing could be due to the fact that the semantic web service research only emerged in 2001 [52, 53]. The idea of adding semantic notifications to a web service description was to facilitate the automatic discovery, selection and composition of web services. Thus, it is only natural for these research areas to have been addressed first by numerous studies. However, the importance of testing has made the testing research community to study on how the semantic notations can be used to facilitate testing as well.

As mentioned earlier in the chapter, testing of web services is important as it provides confidence to the services user to use the service and helps them to decide on the selection of the services. Testing can also be used to ensure that a composed service behaves as it is intended. Related work on web service composition focuses on the orchestration of the composed services, whilst less work is done on testing the behaviour of the service to be used in the composition. As for semantic web services testing, there have been approaches to support them using WSMO [19], OWL-S [36, 39], and WSDL-S [34]. However, most of these approaches attempt to address testing of the business process of the composition. Furthermore, it is not clearly stated how the ontologies are used or mapped to the formalism adopted in their testing approaches. In this research, WSMO is used as the basis of the research to improve the automation of test generation in order to test service behaviour to support selection during automatic service composition.

1.7 Glossary

This section explains some of the terms that have been used in this research. The detailed explanation for each of these terms is provided in the Literature Review section.

- **Web Service** – a software system or technology which describes the services using the XML and these services can be accessed by other software systems using the XML based messages via web. It consists of three important components which are the WSDL, the UDDI and the SOAP.
- **Semantic Web Service** – a new paradigm that brings semantic descriptions to data and behaviour of web services. It has evolved from the integration of the semantic web and web service technologies.
- **Web Service Modeling Ontology (WSMO)** – a formal SWS framework that provides semantic descriptions to all the related aspects of the web service. The WSMO consists of four core elements namely the *Goal*, the *Web Service*, the *Mediator* and the *Ontology*.
- **Web Service Modeling Toolkit (WSMT)** – is an ontology engineering toolkit for the WSMO framework. It is an integrated development environment (IDE) for semantic web services developed for the Eclipse framework. The IDE supports the engineering of WSMO descriptions, creation of mediation mappings and interfacing with Semantic Execution Environment (SEE) and external systems.
- **Web Service Execution Environment (WSMX) -** is the reference implementation of WSMO. It is an execution environment for business application integration where enhanced web services are integrated for various business applications.
- **Web Service Modeling Language (WSML)** – a concrete formal language of the WSMO framework that is used to describe the *Goal*, the *Web Service*, the *Mediator* and the *Ontology* elements.
- **Ontology** - Ontology refers to a formalization of the knowledge in the domain. It is able to interweave human and computer understanding

of symbols. Basic building blocks of ontology design include: classes or concepts, properties of each concept describing various features and attributes of the concept such as restrictions or axioms, instances and relationships.

- **Abstract State Machine –** a state machine that in each step computes a set of updates of the machine's variables. The *states* of ASMs are arbitrary structures in the standard sense they are used in mathematical sciences, i.e. domains of objects with functions and predicates defined on them. The basic *operations* of ASMs are guarded destructive assignments of values to functions at given arguments which are called *guarded transition rules*.
- **Finite State Machine (FSM)** – a mathematical model that can be used to design computer programs. An FSM is composed of states, transition and actions. A state stores information about the past. A transition indicates a state change and is described by a condition. An action is an activity that needs to be performed. An FSM can only be in one state out of a finite number of states. A state moves from one state to another when initiated by a triggering event or condition, which is called transition.
- **Stateful Web Service** – a web service may have more than one operation. A stateful web service is a web service whose operation's response is determined by the internal state of the web service.
- **WSMO Choreography** – describes the expected/requested behaviour of web services, which is in fact a formalization of their public business processes.

1.8 Thesis Outline

This research discusses some specific issues of testing in the SWS approaches during automatic web service composition. It also highlights the limitation of the existing approaches in semantic web services composition testing. It describes a proposed testing approach that enhances automation of the existing approaches. The

proposed solution uses the WSMO ontology and the WSMO transition rules to describe the behaviour of the web service. Both ontologies and transition rules are used in several algorithms within the proposed solution to generate a sequence of test cases. This thesis is organised as follows:-

Chapter 2: This chapter discusses the literature relevant to the research work. It begins with a description on the concepts of web services, web services composition, and semantic web services. This is followed by a discussion on state-of-the-art web services testing issues and solution, web service test generation approaches, state-based web services testing and semantic web services testing. As WSMO choreography is based on abstract state machine, concept of abstract state machine is described and approaches in abstract state machine testing is discussed. A survey and comparative evaluation of the web services testing is presented from three main perspectives which are state-of-the-art web services testing issues, state-based web services testing and semantic web services. The comparative evolution from these three perspectives is necessary in order to understand the strengths and weaknesses of current approaches to web services testing that utilizes semantic description to test interaction behaviours of services during a composition. The outcome of this survey highlights the need for further research in utilizing the semantic description of the web service behaviour to support web service composition testing.

Chapter 3: This chapter describes the research design, procedure and activities which are used in this research. It also discusses on the research instruments, the evaluation criteria, assumptions and limitations that have been adopted and observed in this research.

Chapter 4: This chapter explains the modeling of the SWST. Both architectural and algorithmic aspect of SWST is described in detail. The chapter begins with the motivation of the research; summarises the limitations of the existing approaches; and an analysis on the required elements to overcome the limitations. This is followed by a detailed discussion on the proposed testing approach, SWST. The algorithmic aspect of SWST called C2FSM (Choreography to Finite State

Machine) is the heart of the SWST approach. A description of the C2FSM terms and definitions is provided. This chapter also explains how the algorithms are developed.

Chapter 5: This chapter describes in detail the implementation of the C2FSM approach. It begins with a description on how the semantic description is prepared and validated. A detail explanation on the design and implementation of the proposed algorithm is then provided. Finally, the chapter explains how Chinese Postman Problem algorithm is used to generate test sequences from the generated state machine. The proposed C2FSM algorithms and Chinese Postman Problem algorithm is implemented in a prototype tool called Choreography-Based Test Generator (CBTG).

Chapter 6: This chapter explains the evaluation on the test case generated by CBTG. Firstly a case study is selected. Secondly, mutation testing is applied to measure the quality of the test cases generated. Results of the mutation testing is shown and discussed at the end of the chapter.

Chapter 7: This chapter concludes this thesis by describing the research achievements and contributions. This is followed by the research summary and suggestions for research future works.

REFERENCES

1. Alonso, G., Casati, F., Kuno, H. and Machiraju, V. *Web Services: Concepts, Architectures and Applications*. Springer-Verlag Berlin Heidelberg. 2004
2. Fensel, D., Facca, F. M., Simperl, E. and Toma, I. *Semantic Web Services*. New York: Springer. 2011
3. Sheng, Q. Z., Qiao, X. Q., Vasilakos, A. V., Szabo, C., Bourne, S. and Xu, X. F. Web services composition: A decade's overview. *Information Sciences*. 2014. 280: 218-238.
4. Akkiraju, R., Farrell, J., Miller, J., Nagarajan, M., Schmidt, M.-T., Sheth, A. and Verma, K. *Web Service Semantics - WSDL-S*. White Paper, 2005.
5. IEEE. *IEEE Standard Glossary of Software Engineering Terminology*. IEEE Std 610.12-1990. 1990
6. Bartolini, C., Bertolino, A., Marchetti, E. and Polini, A. WS-TAXI: A WSDL-Based Testing Tool for Web Services. *Proceedings of the International Conference on Software Testing Verification and Validation*. 1-4 April, 2009. Denver, Colarado, USA: IEEE Computer Society. 2009. 326 - 335.
7. Bai, X., Dong, W., Tsai, W.-T. and Chen, Y. WSDL-Based Automatic Test Case Generation for Web Services Testing. *Proceedings of the 2005 IEEE International Workshop on Service-Oriented System Engineering*. 20-21 October, 2005. Beijing, China: IEEE Computer Society. 2005. 215-220.
8. Ma, C., Du, C., Zhang, T., Hu, F. and Cai, X. WSDL-Based Automated Test Data Generation for Web Service. *Proceedings of the International Conference on Computer Science and Software Engineering*. 12-14 December, 2008. Wuhan, China: IEEE Computer Society. 2008. 731-737.
9. Hanna, S. and Munro, M. An Approach for Specification-based Test Case Generation for Web Services. *Proceedings of the IEEE/ACS International*

Conference on Computer Systems and Applications. 13-16 May, 2007. Amman, Jordan: IEEE. 2007. 16-23.

10. Sneed, H. M. and Huang, S. The design and use of WSDL-Test: a tool for testing Web services. *Journal of Software Maintenance and Evolution: Research and Practice*. 2007. 19(5): 297-314.
11. Xu, W., Offutt, J. and Luo, J. Testing Web Services by XML Perturbation. *Proceedings of the 16th IEEE International Symposium on Software Reliability Engineering*. 8-11 November, 2005. Chicago, USA: IEEE Computer Society. 2005. 257-266.
12. Ramollari, E., Dranidis, D. and Simons, A. J. H. Reliable Web Service Publication and Discovery through Model-Based Testing and Verification. *Proceedings of the 3rd Annual South East European Doctoral Student Conference: Infusing Research and Knowledge in South East Europe*. 26-27 June, 2008. Thessaloniki, Greece: SEERC. 2008. 302-314.
13. Bertolino, A., Frantzen, L., Polini, A. and Tretmans, J. Audition of Web Services for testing conformance to open specified protocols. In: Reussner, R. H., Stafford, J. A. and Szyperski, C. A. ed. *Architecting Systems with Trustworthy Components.* Springer. 1-25; 2006.
14. Keum, C., Kang, S., Ko, I.-Y., Baik, J. and Choi, Y.-I. Generating Test Cases for Web Services Using Extended Finite State Machine. *Proceedings of the The 18th IFIP International Conference on Testing Communicating Systems*. 16-18 May, 2006. New York, USA: Springer Berlin Heidelberg. 2006. 103-117.
15. Tsai, W. T., Paul, R., Wang, Y., Fan, C. and Wang, D. Extending WSDL to Facilitate Web Services Testing. *Proceedings of the 7th IEEE International Symposium on High Assurance Systems Engineering*. 23–25 October, 2002. Tokyo, Japan: IEEE Computer Society. 2002. 171-172.
16. Wang, R. and Huang, N. Requirement Model-Based Mutation Testing for Web Service. *Proceedings of the 4th International Conference on Next Generation Web Services Practices*. 20-22 October, 2008. Seoul, South Korea: IEEE Computer Society. 2008. 71-76.
17. Lee, S., Bai, X. and Chen, Y. Automatic Mutation Testing and Simulation on OWL-S Specified Web Services. *Proceedings of the 41st Annual Simulation*

Symposium. 3-16 April, 2008. Ottawa, Ontario, Canada: IEEE Computer Society. 2008. 149-156.

18. Wang, X., Huang, N. and Wang, R. Mutation Test Based on OWL-S Requirement Model. *Proceedings of the IEEE International Conference on Web Services*. 6-10 July, 2009. Beijing, China: IEEE Computer Society. 2009. 1006-1007.
19. Jokhio, M. S., Dobbie, G., Tianming, H. and Jing, S. Automated Mutation-Based Test Case Evaluation for Semantic Web Services. *Proceedings of the 23rd Australian Software Engineering Conference* 7-10 April, 2014. Sydney, Australia: IEEE Computer Society. 2014. 210-219.
20. Bai, X., Kenett, R. S. and Yu, W. Risk Assessment and Adaptive Group Testing Of Semantic Web Services. *International Journal of Software Engineering and Knowledge Engineering*. 2012. 22(5): 595-620.
21. Askarunisa, A., Punitha, K. A. J. and Abirami, A. M. Black Box Test Case Prioritization Techniques for Semantic Based Composite Web Services Using OWL-S. *Proceedings of the International Conference on Recent Trends in Information Technology*. 3-5 June, 2011. Chennai, India: 2011. 1215-1220.
22. Askarunisa, A., Abirami, A. M. and MadhanMohan, S. A Test Case Reduction Method for Semantic Based Web Services. *Proceedings of the International Conference on Computing Communication and Networking Technologies*. 29-31 July, 2010. Karur, India: IEEE. 2010. 1-7.
23. Huang, H., Tsai, W.-T., Paul, R. and Chen, Y. Automated Model Checking and Testing for Composite Web Services. *Proceedings of the Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing*. 18-20 May, 2005. Seattle, WA, USA: IEEE Computer Society. 2005. 300-307.
24. Sinha, A. and Paradkar, A. Model-Based Functional Conformance Testing of Web Services Operating on Persistent Data. *Proceedings of the Workshop on Testing, Analysis, and Verification of Web services and Applications*. 17 July, 2006. Portland, Maine, USA: ACM. 2006. 17-22.
25. Bai, X., Wang, Y., Dai, G., Tsai, W.-T. and Chen, Y. A Framework for Contract-Based Collaborative Verification and Validation of Web Services. *Proceedings of the 10th International ACM SIGSOFT Symposium on*

Component-Based Software Engineering. 9-11 July, 2007. Medford, MA, USA: 2007. 258-273.

26. Dai, G., Bai, X., Wang, Y. and Dai, F. Contract-Based Testing for Web Services. *Proceedings of the 31st Annual International Computer Software and Applications Conference*. 24-27 July, 2007. Beijing, China: IEEE Computer Society. 2007. 517-526.
27. Paradkar, A. M., Sinha, A., Williams, C., Johnson, R. D., Outterson, S., Shriver, C. and Liang, C. Automated Functional Conformance Test Generation for Semantic Web Services. *Proceedings of the IEEE International Conference on Web Services*. 9-13 July, 2007. Salt Lake City, UT, USA: IEEE Computer Society. 2007. 110-117.
28. Wang, Y., Bai, X., Li, J. and Huang, R. Ontology-based test case generation for testing Web Services. *Proceedings of the Eighth International Symposium on Autonomous Decentralized Systems*. 21-23 March, 2007. Sedona, Arizona, USA: IEEE Computer Society. 2007. 43-50.
29. Bai, X., Lee, S., Tsai, W.-T. and Chen, Y. Ontology-Based Test Modeling and Partition Testing of Web Services. *Proceedings of the IEEE International Conference on Web Services*. 23-26 September, 2008 Beijing, China: IEEE. 2008. 465-472.
30. Noikajana, S. and Suwannasart, T. Web Service Test Case Generation Based on Decision Table. *Proceedings of the The Eighth International Conference on Quality Software*. 12-13 August, 2008. Oxford, United Kingdom: IEEE Computer Society. 2008. 321-326.
31. Jokhio, M. S. Goal-Based Testing of Semantic Web Services. *Proceedings of the 24th IEEE/ACM International Conference on Automated Software Engineering*. 16-20 November, 2009. Auckland, New Zealand: IEEE Computer Society. 2009. 707-711.
32. Jokhio, M. S., Dobbie, G. and Sun, J. Towards Specification Based Testing for Semantic Web Services. *Proceedings of the Australian Software Engineering Conference*. 14-17 April, 2009. Gold Coast, QLD, Australia: IEEE Computer Society. 2009. 54-63.
33. Jokhio, M. S., Dobbie, G. and Sun, J. A Framework for Testing Semantic Web Services Using Model Checking. *Proceedings of the Fourth South-East*

European Workshop on Formal Methods. 4-5 December, 2009. Thessaloniki, Greece: IEEE Computer Society. 2009. 17-24.

34. Noikajana, S. and Suwannasart, T. An Improved Test Case Generation Method for Web Service Testing from WSDL-S and OCL with Pair-Wise Testing Technique. *Proceedings of the 33rd Annual IEEE International Computer Software and Applications*. 20-24 July, 2009 Seattle, WA, USA: IEEE Computer Society. 2009. 115-123.
35. Ramollari, E., Kourtesis, D., Dranidis, D. and Simons, A. Leveraging Semantic Web Service Descriptions for Validation by Automated Functional Testing. *Proceedings of the 6th European Semantic Web Conference*. 31 May - 4 June, 2009. Heraklion, Greece: Springer. 2009. 593-607.
36. Li, B., Ji, S., Qiu, D. and Cai, J. Generating Test Cases of Composite Services Based On OWL-S And EH-CPN. *International Journal of Software Engineering and Knowledge Engineering*. 2010. 20(07): 921-941.
37. Ma, C., Wu, J., Zhang, T. and Hu, F. Web Services Testing Based On Stream X-Machine. *Proceedings of the 10th International Conference on Quality Software*. 14-15 July, 2010. Zhangjiajie, China: IEEE. 2010. 232-239.
38. Yu, B. and Li, D. Generating Test Data Based on XML Schema. *Proceedings of the 2010 International Conference on Computer Application and System Modeling*. 22-24 October. 2010. Taiyuan, China: IEEE. 2010. V15-344-V15-348.
39. Liu, C.-H., Chen, S.-L., Kuo, J. Y. and Huang, T.-Y. A Flow Graph-Based Test Model for OWL-S Web Services. *Proceedings of the International Conference on Machine Learning and Cybernetics*. 10-13 July, 2011. Guilin, Guangxi, China: IEEE Computer Society. 2011. 897-902.
40. Xu, X. and Shafin, S. H. Automated Testing of Web Services System Based on OWL-S. *International Journal of Computer Information Systems and Industrial Management Applications*. 2013. 6: 206-216.
41. Tahir, A., Tosi, D. and Morasca, S. A systematic review on the functional testing of semantic web services. *Journal of Systems and Software*. 2013. 86(11): 2877-2889.
42. Palacios, M., García-Fanjul, J. and Tuya, J. Testing in Service Oriented Architectures with dynamic binding: A mapping study. *Information and Software Technology*. 2011. 53(3): 171-189.

43. Mohd Rusli, H., Ibrahim, S., Puteh, M. and Tabatabaei, S. G. H. A Comparative Evaluation of State-of-the-Art Web Service Composition Testing Approaches. *Proceedings of the 6th International Workshop on Automation of Software Test*. 23-24 Mei, 2011. Honolulu, USA: ACM. 2011. 29-35.
44. Ladan, M. I. Web Services Testing Approaches: A Survey and a Classification. *Proceedings of the Networked Digital Technologies*. 7-9 July, 2010. Praque, Czech Republic: Springer Berlin Heidelberg. 2010. 70-79.
45. Endo, A. T. and Simao, A. d. S. *Formal Testing Approaches for Service-Oriented Architectures and Web Services: A Systematic Review*. 2010.
46. Endo, A. and Simao, A. A Systematic Review on Formal Testing Approaches for Web Services. *Proceedings of the Brazilian Workshop on Systematic and Automated Software Testing*. 11 November, 2010. Brazil: 2010. 89-98.
47. Bozkurt, M., Harman, M. and Hassoun, Y. 2010. *Testing Web Services: A Survey.* Technical Report King's College London.
48. Zakaria, Z., Atan, R., Ghani, A. A. A. and Sani, N. F. M. Unit Testing Approaches for BPEL: A Systematic Review. *Proceedings of the Asia-Pacific Software Engineering Conference*. 1-3 December, 2009. Penang, Malaysia: IEEE Computer Society. 2009. 316-322.
49. Canfora, G. and Di Penta, M. Service-Oriented Architectures Testing: A Survey. In: De Lucia, A. and Ferrucci, F. ed. *Software Engineering.* Salerno, Italy: Springer Berlin Heidelberg. 78-105; 2009.
50. Barbir, A., Hobbs, C., Bertino, E., Hirsch, F. and Martino, L. Challenges of Testing Web Services and Security in SOA Implementations. In: Baresi, L. and Nitto, E. D. ed. *Test and Analysis of Web Services.* Springer Berlin Heidelberg. 395-440; 2007.
51. Canfora, G. and Di Penta, M. Testing Services and Service-Centric Systems: Challenges and Opportunities. *IT Professional*. 2006. 8(2): 10-17.
52. Fensel, D., Lausen, H., Polleres, A., Bruijin, J. d., Stollberg, M., Roman, D. and Dominique, J. *Enabling Semantic Web Services*. Springer-Verlag Berlin Heidelberg. 2007
53. McIlraith, S. A., Son, T. C. and Honglei, Z. Semantic Web Services. *IEEE Intelligent Systems*. 2001. 16(2): 46-53.

54. Ching-Seh, W. and Chi-Hsin, H. The Web Services Composition Testing Based on Extended Finite State Machine and UML Model. *Proceedings of the Fifth International Conference on Service Science and Innovation*. 29-31 May, 2013. Kaohsiung, Taiwan: IEEE Computer Society. 2013. 215-222.
55. W3C Working Group. *WSDL v2.0*. 2007 [cited 2010 11 June]; Available from: http://www.w3.org/TR/wsdl20/.
56. W3C Working Group. *SOAP v1.2*. 2007 [cited 2010 11 June]; Available from: http://www.w3.org/TR/soap12-part1/.
57. Offutt, J. and Xu, W. Generating Test Cases for Web Services Using Data Perturbation. *ACM SIGSOFT Software Engineering Notes*. 2004. 29(5): 1-10.
58. OASIS. *UDDI v 3.0.2*. 2004 [cited 2010 11 June]; Available from: http://www.uddi.org/pubs/uddi_v3.htm.
59. Peltz, C. Web Services Orchestration and Choreography. *Computer*. 2003. 36(10): 46-52.
60. Yu, L. *Introduction to the Semantic Web and Semantic Web Services*. Chapman and Hall/CRC. 2007
61. Berners-Lee, T., Hendler, J. and Lassila, O. The Semantic Web. *Scientific American*. 2001. 284(5): 34-43.
62. W3C Semantic Web. *W3C Semantic Web Activity*. 2010 [cited 2010 29 June]; Available from: http://www.w3.org/2001/sw/.
63. W3C Recommendation. *Resource Description Framework (RDF): Concepts and Abstract Syntax*. 2004 [cited 2010 1 July]; Available from: http://www.w3.org/TR/rdf-concepts/.
64. W3C Working Group. *Comparing Semantic Web Services Execution Environments*. 2007.
65. W3C Working Group. *Web Service Modeling Ontology (WSMO)*. 2005 27 October 2011]; Available from: http://www.w3.org/Submission/WSMO/.
66. Fensel, D., Kerrigan, M. and Zaremba, M. *Implementing Semantic Web Services: The SESA Framework*. Springer-Verlag Berlin Heidelberg. 2008
67. Nagarajan, M. Semantic Annotations in Web Services. In: Cardoso, J. and Sheth, A. P. ed. *Semantic Web Services, Processes and Applications.* Springer US. 35-61; 2006.
68. W3C Working Group. *Web Service Semantics - WSDL-S*. 2005 27 October 2011]; Available from: http://www.w3.org/Submission/WSDL-S/.

69. Kopecky, J., Vitvar, T., Bournez, C. and Farrell, J. SAWSDL: Semantic Annotations for WSDL and XML Schema. *IEEE Internet Computing*. 2007. 11(6): 60-67.

70. Feier, C. and Domingue, J. *D3.1v0.2 WSMO Primer*. 2005 [cited 2011 15 March]; Available from: http://www.wsmo.org/TR/d3/d3.1/v0.2/20050401/.

71. Cimpian, E., Mocan, A., Bussler, C. and Haller, A. WSMX Process Mediation Based on Choreographies. In: Bussler, C. J. and Haller, A. ed. *Business Process Management Workshops.* Springer Berlin Heidelberg. 130-143; 2006.

72. Cimpian, E. and Mocan, A. *D13.7 v0.2 Process Mediation in WSMX*. 2005 [cited 2012 1 October]; Available from: http://www.wsmo.org/TR/d13/d13.7/v0.2/.

73. Haller, A., Scicluna, J. and Haselwanter, T. *D13.9v0.1 WSMX Choreography*. 2005.

74. Gurevich, Y. Evolving Algebras 1993: Lipari Guide. In: Borger, E. ed. *Specification and Validation Methods.* New York, USA: Oxford Universty Press. 9-36; 1995.

75. Roman, D., Scicluna, J., Nitzsche, J., Fensel, D., Polleres, A., Bruijn, J. d. and Heymans, S. *D14v1.0. Ontology-based Choreography*. 2007.

76. Bertolino, A. and Polini, A. The Audition Framework for Testing Web Services Interoperability. *Proceedings of the 31st EUROMICRO Conference on Software Engineering and Advanced Applications*. 30 August -3 September, 2005. Porto, Portugal: IEEE Computer Society. 2005. 134-142.

77. Looker, N., Munro, M. and Xu, J. Testing Web Services. *The 16th IFIP International Conference on Testing of Communicating Systems, Student Forum.* Oxford, UK. 2004.

78. El Ioini, N. S-Test: A Framework for Services Testing. *Proceedings of the On the Move to Meaningful Internet Systems*. 26-30 October, 2015. Rhodes, Greece: Springer International Publishing. 2015. 612-619.

79. Sneed, H. M. and Verhoef, C. Measuring Test Coverage of SoA Services. *Proceedings of the IEEE 9th International Symposium on the Maintenance and Evolution of Service-Oriented and Cloud-Based Environments*. 2 October, 2015. Bremen, Germany: IEEE Computer Society. 2015. 59-66.

80. Kiran, M. and Simons, A. J. H. Model-Based Testing for Composite Web Services in Cloud Brokerage Scenarios. In: Ortiz, G. and Tran, C. ed. *Advances in Service-Oriented and Cloud Computing.* Springer International Publishing. 190-205; 2015.

81. Martens, A. and Tappenden, A. F. A Benchmark for Automated Web Service Testing. *1st Consortium for Software Engineering Research Meeting.* Edmonton, Canada. 2014.

82. Sneed, H. M. Web Service Test Evolution. *Proceedings of the 8th International Conference in Software Quality*. 18-21 January, 2016. Vienna, Austria: Springer International Publishing. 2016. 171-185.

83. Belli, F., Endo, A. T., Linschulte, M. and Simao, A. A Holistic Approach to Model-Based Testing Of Web Service Compositions. *Software: Practice and Experience*. 2014. 44(2): 201-234.

84. López, M., Ferreiro, H., Francisco, M. and Castro, L. Automatic Generation of Test Models for Web Services Using WSDL and OCL. *Proceedings of the 11th International Conference on Service-Oriented Computing*. 2-5 December, 2013. Berlin, Germany: Springer Berlin Heidelberg. 2013. 483-490.

85. Kumar, A., Pandey, A. K. and Zade, F. R. Automatic Test Data Generation of Web Services through XML Schema. *Asian Journal of Convergence in Technology*. 2015. 2(2).

86. Kumar, A. A Novel Testing Model for SOA based Services. *International Journal of Modern Education and Computer Science*. 2015. 7(1): 31.

87. Kaschner, K. Conformance Testing for Asynchronously Communicating Services. *Proceedings of the 9th International Conference on Service-Oriented Computing*. 5-8 December, 2011 Paphos, Cyprus: Springer Berlin Heidelberg. 2011. 108-124.

88. Wang, H., Xing, J., Yang, Q., Han, D. and Zhang, X. Modification Impact Analysis Based Test Case Prioritization for Regression Testing of Service-Oriented Workflow Applications. *Proceedings of the 39th Annual IEEE Computer Software and Applications Conference Workshops* 1-5 July, 2015. Taichung, Taiwan: IEEE. 2015. 288-297.

89. Bozkurt, M. and Harman, M. Cost-Aware Test Suite Minimisation for Service-Centric Systems. *Proceedings of the 4th Symposium on Search-Based Software Engineering*. 28 - 30 September, 2012. Trento, Italy: 2012. 13-18.
90. Mei, L. Preemptive Regression Testing of Workflow-Based Web Services. *IEEE Transactions on Services Computing*. 2014. 8(5): 740 - 754.
91. Petrova-Antonova, D., Ilieva, S. and Manova, D. TASSA: Testing Framework for Web Service Orchestrations. *Proceedings of the IEEE/ACM 10th International Workshop on Automation of Software Test*. 23-24 May, 2015. Florence/Firenze, Italy: IEEE/ACM. 2015. 8-12.
92. Zhu, H. and Zhang, Y. A Test Automation Framework for Collaborative Testing of Web Service Dynamic Compositions. In: Bouguettaya, A., Sheng, Q. Z. and Daniel, F. ed. *Advanced Web Services.* Springer New York. 171-197; 2014.
93. Besson, F., Moura, P., Kon, F. and Milojicic, D. Bringing Test-Driven Development to Web Service Choreographies. *Journal of Systems and Software*. 2014. 99: 135-154.
94. de Melo, A. C. V. and Silveira, P. Improving Data Perturbation Testing Techniques for Web Services. *Information Sciences*. 2010. 181(3): 600-619.
95. Estero-Botaro, A., García-Domínguez, A., Domínguez-Jiménez, J., Palomo-Lozano, F. and Medina-Bulo, I. A Framework for Genetic Test-Case Generation for WS-BPEL Compositions. *Proceedings of the International Conference on Testing Software and Systems*. 23-25 September, 2014. Madrid, Spain: Springer Berlin Heidelberg. 2014. 1-16.
96. Bentakouk, L., Poizat, P. and Zaïdi, F. Checking the Behavioral Conformance of Web Services with Symbolic Testing and an SMT Solver. *Proceedings of the 5th International Conference on Tests & Proofs*. 30 June - 1 July, 2011. Zurich, Switzerland: Springer Berlin Heidelberg. 2011. 33-50.
97. Mao, C. Towards a Hierarchical Testing and Evaluation Strategy for Web Services System. *Proceedings of the 7th ACIS International Conference on Software Engineering Research, Management and Applications*. 2-4 December, 2009. Haikou, China: IEEE. 2009. 245-252.
98. Mao, C. A Specification-Based Testing Framework for Web Service-Based Software *Proceedings of the IEEE International Conference on Granular*

Computing. 22 September, 2009. Nanchang, China: IEEE Computer Society. 2009.

99. Hou, J. and Xu, L. DFTT4CWS: A Testing Tool for Composite Web Services Based on Data-Flow. *Proceedings of the Sixth Web Information Systems and Applications Conference*. 18-20 September, 2009. Xuzhou, China: IEEE Computer Society. 2009. 62-67.
100. Ma, C., Wu, J., Zhang, T., Zhang, Y. and Cai, X. Testing BPEL with Stream X-Machine. *Proceedings of the International Symposium on Information Science and Engineering*. 20-22 December, 2008. Shanghai, China: IEEE Computer Society. 2008. 578-582.
101. Lallali, M., Zaidi, F. and Cavalli, A. Transforming BPEL into Intermediate Format Language for Web Services Composition Testing. *Proceedings of the 4th International Conference on Next Generation Web Services Practices*. 20-22 Oct. 2008. Seoul, Korea: 2008. 191-197.
102. Zheng, Y., Zhou, J. and Krause, P. A Model Checking based Test Case Generation Framework for Web Services. *Proceedings of the International Conference on Information Technology*. IEEE Computer Society. 2007.
103. Garcia-Fanjul, J., de la Riva, C. and Tuya, J. Generation of Conformance Test Suites for Compositions of Web Services Using Model Checking. *Proceedings of the Testing: Academic and Industrial Conference - Practice And Research Techniques*. 29-31 Aug. 2006. 2006. 127-130.
104. Yan, J., Li, Z., Yuan, Y., Sun, W. and Zhang, J. BPEL4WS Unit Testing: Test Case Generation Using a Concurrent Path Analysis Approach. *Proceedings of the 17th International Symposium on Software Reliability Engineering*. 7-10 November, 2006. Raleigh, North Carolina, USA: IEEE Computer Society. 2006. 75-84.
105. Yuan, Y., Li, Z. and Sun, W. A Graph-Search Based Approach to BPEL4WS Test Generation. *Proceedings of the International Conference on Software Engineering Advances*. October 29-November, 2003. Tahiti, French Polynesia: IEEE Computer Society. 2006. 14-22.
106. Zhou, L., Xu, L., Xu, B. and Yang, H. Generating Test Cases for Composite Web Services by Parsing XML Documents and Solving Constraints. *Proceedings of the 39th Annual Computer Software and Applications Conference*. 1-5 July, 2015. Taichung, Taiwan: IEEE. 2015. 304-309.

107. Cao, T.-D., Felix, P., Castanet, R. and Berrada, I. Testing Web Services Composition Using the TGSE Tool. *Proceedings of the World Conference on Services-I*. 6-10 July 2009. Los Angeles, California, USA: 2009. 187-194.

108. Kourtesis, D., Ramollari, E., Dranidis, D. and Paraskakis, I. Increased Reliability in SOA Environments Through Registry-Based Conformance Testing of Web Services. *Production Planning & Control: The Management of Operations*. 2010. 21(2): 130-144.

109. Miao, W. and Liu, S. A Formal Engineering Framework for Service-Based Software Modeling. *IEEE Transactions on Services Computing*. 2013. 6(4): 536-550.

110. Endo, A. T. and Simao, A. Model-Based Testing of Service-Oriented Applications via State Models. *Proceedings of the IEEE International Conference on Services Computing*. 4-9 July, 2011. Washington, DC, USA: IEEE. 2011.

111. Cao, T.-D., Felix, P., Castanet, R. and Berrada, I. Online Testing Framework for Web Services. *Proceedings of the Third International Conference on Software Testing, Verification and Validation*. 6-10 April, 2010. Paris, France: IEEE. 2010. 363-372.

112. Hou, S.-S., Zhang, L., Lan, Q., Mei, H. and Sun, J.-S. Generating Effective Test Sequences for BPEL Testing. *Proceedings of the 9th International Conference on Quality Software*. 24-25 August, 2009. Jeju, Korea: IEEE Computer Society. 2009. 331-340.

113. Lallali, M., Zaidi, F., Cavalli, A. and Hwang, I. Automatic Timed Test Case Generation for Web Services Composition. *Proceedings of the 6th IEEE European Conference on Web Services*. 12-14 November, 2008. Dublin, Ireland: IEEE Computer Society. 2008. 53-62.

114. Zheng, Y., Zhou, J. and Krause, P. An Automatic Test Case Generation Framework for Web Services. *Journal of Software*. 2007. 2(3): 64-77.

115. Garcia-Fanjul, J., Tuya, J. and de la Riva, C. Generating Test Cases Specifications for BPEL Compositions of Web Services Using SPIN. *Proceedings of the International Workshop on Web Services Modeling and Testing*. 9 June, 2006. Palermo, Italy: 2006. 83-94.

116. Dranidis, D., Kourtesis, D. and Ramollari, E. Formal Verification of Web Service Behavioural Conformance through Testing. *Proceedings of the 3rd*

South-East European Workshop on Formal Methods. 30 November – 1 December, 2007. Thessaloniki, Greece: SEERC. 2007. 112–125.

117. Yu, J., Han, J., Gunarso, S. and Versteeg, S. A Business Protocol Unit Testing Framework for Web Service Composition. *Proceedings of the 25th International Conference on Advanced Information Systems Engineering*. 17-21 June, 2013. Springer Berlin Heidelberg. 2013. 17-34.
118. Belli, F. and Linschulte, M. Event-Driven Modeling and Testing of Real-Time Web Services. *Service Oriented Computing and Applications*. 2010. 4(1): 3-15.
119. Miao, W. and Shaoying, L. A Formal Specification-Based Testing Approach to Accurate Web Service Selection. *Proceedings of the IEEE Asia-Pacific Services Computing Conference*. 12-15 December, 2011. Jeju Island, South Korea: IEEE. 2011. 259-266.
120. Fraser, G., Wotawa, F. and Ammann, P. E. Testing with Model Checkers: A Survey. *Software Testing, Verification and Reliability*. 2009. 19(3): 215-261.
121. Bentakouk, L., Poizat, P. and Zaïdi, F. A Formal Framework for Service Orchestration Testing Based on Symbolic Transition Systems. *Proceedings of the 9th International Workshop on Formal Approaches to Testing of Software*. 2-3 November, 2009. Eindhoven, The Netherlands: Springer Berlin Heidelberg. 2009. 16-32.
122. Youngki, P., Woosung, J., Byungjeong, L. and Chisu, W. Automatic Discovery of Web Services Based on Dynamic Black-Box Testing. *Proceedings of the 33rd Annual IEEE International Computer Software and Applications Conference*. 20-24 July, 2009. Washington, USA: IEEE. 2009. 107-114.
123. Bartolini, C., Bertolino, A., Marchetti, E. and Polini, A. Towards Automated WSDL-Based Testing of Web Services. *Proceedings of the 6th International Conference on Service-Oriented Computing*. 1-5 December, 2008. Sydney, Australia: Springer Berlin Heidelberg. 2008. 524-529.
124. Liu, C.-H., Chen, S.-L. and Li, X.-Y. A WS-BPEL Based Structural Testing Approach for Web Service Compositions. *Proceedings of the 4th IEEE International Symposium on Service-Oriented System Engineering* 18-19 December, 2008. Jhongli, Taiwan: IEEE Computer Society. 2008. 135-141.

125. Gargantini, A. and Riccobene, E. ASM-based Testing: Coverage Criteria and Automatic Test Sequence Generation. *Journal of Universal Computer Science*. 2001. 7(11): 1050-1067.

126. Grieskamp, W., Gurevich, Y., Schulte, W. and Veanes, M. Conformance Testing with Abstract State Machines. In: Moreno-Diaz, R., Buchberger, B. and Freire, J.-L. ed. *Computer Aided Systems Theory - EUROCAST 2001*. Las Palmas de Gran Canaria, Spain: Springer Berlin Heidelberg. 2001.

127. Gargantini, A., Riccobene, E., Rinzivillo, S. and Börger, E. Using Spin to Generate Tests from ASM Specifications. *Proceedings of the 10th International Workshop on ASM Advances in Theory and Practice* 3-7 March, 2003. Taormina, Italy: Springer Berlin Heidelberg. 2003. 263-277.

128. Grieskamp, W., Lepper, M., Schulte, W. and Tillmann, N. Testable Use Cases in The Abstract State Machine Language. *Proceedings of the Second Asia-Pacific Conference on Quality Software*. 3-7 March, 2001. Taormina, Italy: Springer Berlin Heidelberg. 2001. 167-172.

129. Grieskamp, W., Gurevich, Y., Schulte, W. and Veanes, M. Generating Finite State Machines from Abstract State Machines. *ACM SIGSOFT Software Engineering Notes*. 2002. 27(4): 112-122.

130. Barnett, M., Grieskamp, W., Nachmanson, L., Schulte, W., Tillmann, N. and Veanes, M. Model-Based Testing with AsmL .NET. *Proceedings of the 1st European Conference on Model-Driven Software Engineering*. 11 December, 2003. Nuremberg, Germany: 2003. 12-19.

131. Barnett, M., Grieskamp, W., Schulte, W., Tillmann, N. and Veanes, M. Validating Use-Cases With The AsmL Test Tool. *Proceedings of the Third International Conference on Quality Software*. 6-7 November, 2003. Washington, DC, USA: IEEE Computer Society 2003. 238-246.

132. Barnett, M., Grieskamp, W., Nachmanson, L., Schulte, W., Tillmann, N., Veanes, M., Petrenko, A. and Ulrich, A. Towards a Tool Environment for Model-Based Testing with AsmL :Formal Approaches to Software Testing. In: Petrenko, A. and Ulrich, A. ed. *Formal Approaches to Software Testing.* Springer Berlin Heidelberg. 1102-1103; 2004.

133. Cavarra, A., Börger, E., Butler, M., Bowen, J. and Boca, P. Data Flow Analysis and Testing of Abstract State Machines. *Proceedings of the First*

International Conference on Abstract State Machines, B and Z. 16-18 September, 2008 London, UK: Springer Berlin Heidelberg. 2008. 85-97.

134. Cavarra, A. A Data-Flow Approach to Test Multi-Agent ASMs. *Formal Aspects of Computing*. 2011. 23(1): 21-41.

135. Lázaro, M. and Marcos, E. Research in Software Engineering: Paradigms and Methods. *Proceedings of the 17th International Conference Advanced Information Systems Engineering Workshop*. 13-17 June, 2005. Porto, Portugal: 2008.

136. Fabbri, S. C. P. F., Delamaro, M. E., Maldonado, J. C. and Masiero, P. C. Mutation Analysis Testing For Finite State Machines. *Proceedings of the 5th IEEE International Symposium on Software Reliability Engineering* 6-9 November, 1994. Monterey, California, USA: IEEE. 1994. 220-229.

137. Utting, M. and Legeard, B. *Practical Model-Based Testing: A Tools Approach*. Morgan Kaufmann. 2007

138. Jia, Y. and Harman, M. An Analysis and Survey of the Development of Mutation Testing. *IEEE Transactions on Software Engineering*. 2011. 37(5): 649-678.

139. DeMillo, R. A., Lipton, R. J. and Sayward, F. G. Hints on Test Data Selection: Help for the Practicing Programmer. *Computer*. 1978. 11(4): 34-41.

140. Offutt, A. J. and Untch, R. H. Mutation 2000: Uniting the Orthogonal. In: Wong, W. E. ed. *Mutation Testing for the New Century*. Boston, MA: Springer US. 34-44; 2001.

141. Burnstein, I. *Practical Software Testing: A Process-Oriented Approach*. Springer Publishing Company, Incorporated. 2010

142. Ammann, P. and Offutt, J. *Introduction to Software Testing*. Cambridge University Press. 2008

143. Belli, F., Budnik, C. J., Hollmann, A., Tuglular, T. and Wong, W. E. Model-based mutation testing—Approach and Case Studies. *Science of Computer Programming*. 2016. 120: 25-48.

144. Mei, H. and Zhang, L. A Framework for Testing Web Services And Its Supporting Tool. *Proceedings of the IEEE International Workshop on Service-Oriented System Engineering*. 20-21 October, 2005. Beijing, China: IEEE Computer Society. 2005. 199-206.

145. Kleczek, D., Losada, S., Bas, J. L., Bellido, S., Contreras, J. and Montes, M. M. *D10.8 WSMO Descriptions of Application 2 WP 10: Case Study eBanking*. 2005.

146. Bruijn, J. d., Fensel, D., Kerrigan, M., Keller, U., Lausen, H. and Scicluna, J. *Modeling Semantic Web Services: The Web Service Modeling Language*. Springer Publishing Company, Incorporated. 2008

147. Kopecky, J., Roman, D., Scicluna, J. and Fensel, D. *D3.4v0.2. WSMO Use Case: Amazon E-commerce Service*. 2006.

148. Nitzsche, J., Lessen, T. v., Karastoyanova, D. and Leymann, F. Composing Services on the Grid Using BPEL4SWS. *Multiagent and Grid Systems*. 2009. 5(3): 287-309.

149. Nitzsche, J., Van Lessen, T., Karastoyanova, D. and Leymann, F. WSMO/X in the Context of Business Processes: Improvement Recommendations. *International Journal of Web Information Systems*. 2007. 3(1/2): 89-103.

150. Hernandez, R. L. *A Flexible Model for the Semi-automatic Location of Services*. Doctor of Philosophy. Universidad Autonoma de Madrid; 2007

151. Tabatabaei, S. G. H., Wan M. N. Wan Kadir and Ibrahim, S. AIMO: An Effective Approach to Support SemanticWeb Service Discovery and Composition. *International Journal of Computational Science*. 2009. 3(2): 133-150.

152. Tabatabaei, S. G. H., Kadir, W. M. N. W., Ibrahim, S. and Dastjerdi, A. V. AIMO Translator: Bridging the Gap between Semantic Web Service Discovery and Composition. *Proceedings of the Fifth International Conference on Internet and Web Applications and Services*. 9-15 May, 2010. Barcelona, Spain: IEEE Computer Society. 2010. 268-273.

153. Tabatabaei, S. G. H., Kadir, W. M. N. W., Ibrahim, S. and Dastjerdi, A. V. Integrating Discovery and Composition of Semantic Web Services Based on Description Logic. *Journal of Computer Science, Informatics & Electrical Engineering*. 2009. 3(1).

154. Tabatabaei, S. G. H., Kadir, W. M. N. and Ibrahim, S. Semantic Web Service Discovery and Composition Based on AI Planning and Web Service Modeling Ontology. *Proceedings of the IEEE Asia-Pacific Services Computing Conference*. 9-12 December, 2008. Yilan, Taiwan: IEEE Computer Society. 2008. 397-403.

155. Bertolino, A., De Angelis, G., Frantzen, L. and Polini, A. The PLASTIC Framework and Tools for Testing Service-Oriented Applications. In: Andrea, L. and Filomena, F. ed. *Software Engineering.* Springer Berlin Heidelberg. 106-139; 2009.

156. WSMX. *Web Services Execution Environment*. 2013; Available from: http://sourceforge.net/projects/wsmx/.

157. Dimitrov, M., Momtchev, V., Simov, A., Ognyanof, D. and Konstantinov, M. *wsmo4j Programmers Guide*. 2006.

158. Bruijn, J. d., Lausen, H., Krummenacher, R., Polleres, A., Predoiu, L., Kifer, M. and Fensel, D. *D16.1v0.21 The Web Service Modeling Language WSML*. 2005.

159. Roman, D., Scicluna, J., Feier, C., Stollberg, M. and Fensel, D. *D14v0.1. Ontology-based Choreography and Orchestration of WSMO Services*. 2005.

160. Steinmetz, N. and Toma, I. *D16.1v1.0 WSML Language Reference*. 2008.

161. Stephan, G., Pascal, H. and Andreas, A. Knowledge Representation and Ontologies. In: Studer, R., Grimm, S. and Abecker, A. ed. *Semantic Web Services: Concepts, Technologies, and Applications.* Springer Berlin Heidelberg. 51-105; 2007.

162. Thimbleby, H. The directed Chinese Postman Problem. *Software: Practice and Experience*. 2003. 33(11): 1081-1096.

163. Everett, G. D. and McLeod, R. *Software Testing: Testing Across the Entire Software Development Life Cycle*. Wiley-IEEE Computer Society Press. 2007

164. Salva, S. and Rabhi, I. Stateful Web Service Robustness. *Proceedings of the Fifth International Conference on Internet and Web Applications and Services*. 9-15 May, 2010. Barcelona, Spain: IEEE. 2010. 167-173.

165. *Product Advertising API Developer Guide API Version 2013-08-01*. 2013.

166. Fabbri, S. C. P. F., Maldonado, J. C. and Delamaro, M. E. Proteum/FSM: A Tool to Support Finite State Machine Validation Based On Mutation Testing. *Proceedings of the XIX International Conference of the Chilean Computer Science Society*. 3-13 November, 1999. Talca, Chile: IEEE Computer Society. 1999. 96-104.