Sindh Univ. Res. Jour. (Sci. Ser.) Vol.45 (4) 721-726 (2016)

SINDH UNIVERSITY RESEARCH JOURNAL (SCIENCE SERIES)

Framework for Automatically Composed Web Services Using Semantic Web

V. KHALIQUE[++], S. NIZAMANI, S. M. S. SHAH, Y. A. MALKANI[++]* B. BALOCH**

Department of Software Engineering, Mehran University of Engineering and Technology, Jamshoro.

Received 8th October 2015 and Revised 16th June 2016

Abstract: Over the past decades, switched to Web services. Web services offers flexibility and interoperability of independent functions that are platform independent software applications and perform a specific task. Private and public institutions in the developed nations are offering web services. These include service such as online appointment booking for patients, air, train, hotel bookings and online banking. Semantic Web services are a synergetic confluence of semantic Web and Web services. They have the potential to offer value-added services by automatically discovering and assembling Web services to accomplish a domain task.

Automated Web service composition is a key research area concerning Web services as it reduces human effort. In this regard, the challenge is service discovery and communication between independent services. In this paper, the framework of semantically automated/enhanced composite Web services is proposed. A scenario of planning a trip is taken into consideration and as a proof of concept. A trip planner application (TPA) is developed, based on the proposed framework. TPA is a trip planner capable of automatically identifying the best deals for the users. Based on certain limited input parameters, the trip planner service would automatically identify other related services and invoke them. For accomplishing the task of planning a trip, for instance flight-booking service would invoke hotel reservation service by using flight details, with least possible human effort and interaction with the service. This service would automate the process of invoking other related services.

Keywords: Automatically composed Web services, composite Web services, Web services, Semantic Web, Ontology

1. INTRODUCTION

Web services add a new level of functionality to the Web. They are part of the open environment of distributed applications (Kopecky, *et al.,* 2007). Web services are independent, interoperable and modular applications, which can be invoked across the Web. For accomplishment of complex tasks, a singular Web Service cannot offer a complete solution as typically Web services act independently. Such tasks would require services that can discover the required services and communicate with them. A number of solution have been proposed for creating such automated /composite services for example, composition of Web services through orchestration (Peltz, 2003). and Formal Methods for Web Service Composition (Pathak, *et al.,* 2007) one such solution is through the use of semantics.

The goal of Web Services have hiked from simple collaboration to intelligent teamwork. An important challenge is to develop methodologies and tools that enable (semi-) automatic composition of services by taking into account the functional, non-functional and behavioural requirements of the service developer (Pathak, *et al.,* 2007).

The Semantic Web works closely with the knowledge representation so that it structures data in a machine interpretable form. Comprising of information represented, stored, retrieved and processed intelligently, the Semantic Web is the extension of the existing Web. Increasing automation and accuracy on numerous facets such as search, information extraction and information integration, Semantic Web can enhance the understanding of computers (software) about the information they store and process.

In the course of creating the intelligent Web, researchers are striving for applications that are user friendly. They need to provide more functionality and facilitate users. There is a need of intercommunication, invocation and collaboration of Web services.

2. BACKGROUND

The advent of the World Wide Web (WWW) accelerated the exchange of goods and services in a very short span of time. The local markets restricted to their geographical limitations, sprouted to international markets and the exchange of products, known as commerce now extended to (Electronic Commerce). As a result, the level of goods and services increased to match international standards (Fawzy and Youssef 2003) With simple independent Web applications, businesses flourished and new promising opportunities on Web attracted many local and international businesses. resulted in more complex systems and soon intercommunication amongst independent Web

[++]Corresponding authors: Vijdan Khalique: vijdan.khalique@faculty.muet.edu.pk yasir.malkani@usindh.edu.pk
*Institute of Mathematics and Computer Science (IMCS), University of Sindh, Jamshoro.
**Sindh Agriculture, University, Tando Jam

applications became more of a necessity than a choice. There is a need for one Web application to communicate with the other. Web applications are developed on numerous platforms, the interoperability became a crucial issue. Web Services played their part to address these issues.

Web Services add a new level of functionality to the Web, a step toward an open environment of distributed applications. The Web services paradigm promises to enable rich, flexible, and dynamic interoperation of highly distributed and heterogeneous Web-hosted services (Hull, and Su. 2004). Computers must also be able to find and combine services on the Web to free users' hands and make the Web of services scale together with the Web of data (Hull, and Su. 2004) It is mainly concerned with two things: common formats for grouping data which is extracted from a range of varied sources; and language for describing how the data is connected to the real world objects (Dew, *et al.,* 2011) 1)

2)

UDDI (Universal Description, Discovery and Integration) is used for searching the Web Services. It is searched for the WSDL (Web Service Description 3) Language) with semantic annotations only. These 4) annotations provide a URI (universal resource identifier) for the ontology associated with it. In UDDI, 5) each Web Service has WSDL consisting of SAWSDL 6) (Semantic Annotations for Web Service Description Language). Only those Web Services are addressed which contain Semantic Annotations. When appropriate services are found, the Web Service client is developed and invoked dynamically.

Resource description framework (RDF) is a W3C standard for encoding knowledge. It is method of structuring and processing data for semantic Web. Currently, the Web consist decentralized presentation of data. However, it makes sense to human beings but for computers, it just simply a meaningless structure of mark-up language. This means that computer does not understand the real meaning of this data. RDF is an attempt to represent and structure data in such a way that it makes sense to computer and enables it to produce and infer knowledge out from it.

Ontology enables effective sharing of knowledge and its reuse over the Web and business logics are implemented by Semantic Web Services (SWS). *Ontology is a form of knowledge organization a specification of a conceptualization. (*Kopecky, *et al.,* 2007). It is a formal way of representing knowledge in a conceptual manner, where there exists a realistic relationship amongst various classes. The semantic Web services implement the business logics and ontologies are used for effective sharing of knowledge. In order to enable smart invocation of semantic Web services, ontologies can help to provide required knowledge.

SAWSDL is the World Wide Web Consortium's (W3C; www.w3.org) first step toward standardizing technologies for Semantic Web Services (SWSs). As a standard, it provides a common ground for the various ongoing efforts toward SWS frameworks, such as the Web Service WSMO (Domingue, *et al.,* 2005). the OWL-based Web Service Ontology OWL-S (Lara, *et al.,* 2004). SAWSDL are annotations that helps existing Web services to convert into Semantic Web services (SWS) (Kopecky, *et al.,* 2007).

3. MATERIALS AND METHODS

3.1. State-Of-The-Art In Web Service Composition

According to the research, three major challenges stand in the way of Web Service composition (Rao, *et al.,* 2005).

There are a huge number of Web Services, and the number is growing dramatically making the Web Services search difficult.

The Web Services are created and updated dynamically, therefore the up-to-date information about the Web Services is integral

A number of organizations are creating and publishing Web Services in their own specific concept model for web service description. There does not exist a unique language to define and evaluate the Web services in an identical means.

The process of automatic Web Service composition is fully or partially achieved using various methodologies and Web Service modelling languages**.** Current web service composition approaches range from practical languages aspiring to become standards (like BPEL, WS-CDL, OWL-S and WSMO) to theoretical models (like automata, Petri nets and process algebras). These approaches also include Artificial Intelligence Planning (Peer, 2005)., ASTRO approach (Marconi, *et al.,* 2008). model-based approach and workflow-based approach (Xiong, *et al.,* 2011).

3.2. Static and Dynamic Web Service Composition

The Web service composition based on how they are composed and they are divided in to two basic categories static and dynamic Web Services.

Static Web Service composition involves gathering all related Web Services, bounding all services together, implementation and deployment in a single system. Static composition works well in the scenarios where problems to be solved by a composite Web Service

does not change frequently or changes are minor in requirement. Static composition is not suitable in situations where functionality should change with every incoming requirement on the runtime.

In contrast with static Web Service Composition, Dynamic Web services composition requires the execution system to support automatic discovery, selection, and binding of service components (Sheng, *et al.,* 2014). The automated composition is very challenging as there are many critical issues to take care of for example time for composing and invoking the Web Services, the correctness of Web Services being composed, Quality of Service QoS etc. (Sheng, *et al.,* 2014). (Sapkota, 2010).

3.3. Manual and Automated Web Service Composition

Manual composition requires a modelling language like BPEL or OWL-S. The designer of composite Web Services manually takes care of all the aspects of Web Service composition. Manual composition is a time-consuming and error-prone procedure with no assurance that the execution result will satisfy the user's requirements (Sheng, *et al.,* 2014). In comparison with manual composition of Web Services, automated composition uses Semantic Web and Artificial Intelligence (AI) Planning techniques.

3.4 Proposed Framework and Trip Planner Scenario

This section proposes a testing scenario to exam the functionality and applicability of the proposed framework (in section 4). For understanding the context of proposed framework, one needs to understand where and how to apply and test the results of the framework. Traditionally, trip planning takes place with the help of travel agents who charge for their services. However, during the past decade the trend has shifted to the Internet. Where users can identify the best deals.

There are two category of travellers. First category is of those travellers who have difficulty in deciding the most feasible option according to their budget, time and desired level of luxury. They consult travel agents or refer to their acquaintances. The second category of travellers comprise of those who use online Website to plan their trip and have to surfs through multiple Websites to do so. For example, if he wants to book a flight, a hotel room and a taxi that takes him from airport to hotel. During this process of different reservations, he visits three different Websites for planning his trip. Still he is not sure whether he has managed to plan according to his budget. In case, the traveller wants to plan a trip within certain budget limitations, where he is looking for an economical trip, he might not be sure of if he has chosen those taxi, flight and hotel services, which are the cheapest. There are fair chances of the trip he thinks is the cheapest might not be the one as he failed to look for even cheaper options for hotel room, taxi and flight.

There are many reputed trip planner Websites such as http://www.expedia.co.in and http://www.makemytrip.com/, which provides similar facilities. These Websites do provide the functionality of booking flight and hotel together but the hotel and flight booking services they provide is not necessarily the cheapest ones as they have certain subscriptions of these service providers. The following figures give the main interface of www.expedia.co.in and www.makemytrip.com respectively:

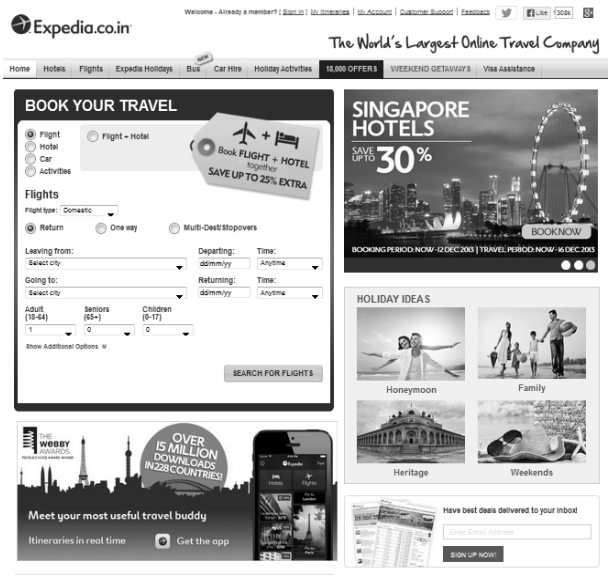

Fig. 1: Screenshot showing the main Webpage of www.expedia.co.in offering hotel and flight booking

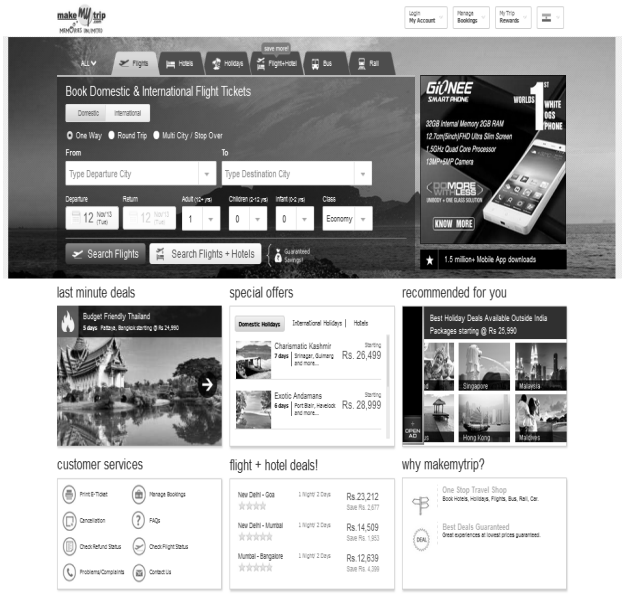

Fig.2: Screenshot showing the main Webpage of offering hotel and flight booking

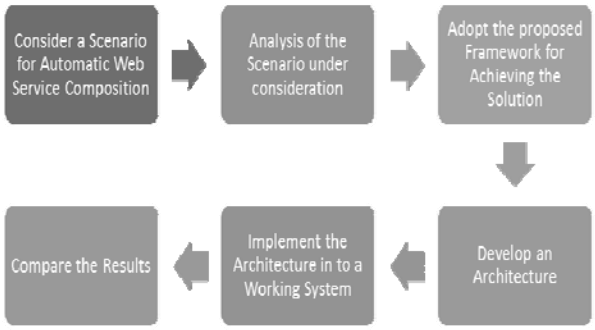

Fig.3: Methodology for perceiving a scenario and applying Web Service Composition Framework.

Consequently, there is a need of a system, which serves as a platform. This scenario suits the best to prove the concept of Creating Automatically Composed Web Services Using Semantics framework.

In this research, Trip Planner Application TPA is developed as a proof-of-concept to show that the proposed Automated

3.5 Proposed Framework

In this research, a framework is proposed which can be adopted by system designers, who are targeting a certain solution with the assistance of automatic composition of Web Services. The framework suggest some general steps as a methodology. This framework is then applied on the trip planning scenario (discussed in section 4). The purposed framework is named as Automatically Composed Web Services ACWS using Sematic Web.

3.5.1. Framework Adaptation Methodology

The flow diagram in figure 3, illustrates the methodology to implement the proposed framework of Web Services composition.

The first step in the methodology is choosing a scenario where Web Service composition is required. In this research, the scenario of trip planning is considered and the overview of the scenario is in section 4. The methodology start with identifying the problem where automated web service composition can serve to solve the problem.

The analysis takes place for identifying the workflow of the process. This analysis helps to identity he parameters associated with the service composition such as the quality of service, time, amount of human interaction with the system etc. The business flow, critical success factors and business logics are identified und understood. Those business rules, which require web service composition, are isolated and focused to proceed with the methodology. Based on the parameters identified in the previous steps, the proposed framework is adopted to reach the desired solution i.e. planning a trip.

The methodology is proceeded with the development of the architecture of the application, implementation and testing and comparing results to see if the automatic composition of Web Services is yielding better results.

3.5.2 Automatically Composed Web Services ACWS Using Semantic Web Framework

Composed of three independently working Web services; taxi booking, flight reservation and hotel reservation, the trip planner application can successfully perform the co-ordination required to plan a trip. The user provides the departure point, destination point, timings and budget for his trip. The application then invokes those services, which fits the user budget criteria and plans the user trip from the flight that would take him to the desired destination, the taxi that would pick user from airport and take him to the desired hotel where user wishes to stay, all under specified budget.

The ACWS frameworks is composed of four basic components as show in diagram 2.

1. **Search for Application Specific Web Services**: process of searching all Web Services related to the chosen scenario of Web Service composition leads to the search of flight, hotel and taxi booking Web Services for developing TPA. This initial phase of ACWS using Semantic Web framework is a manual process. The application designer/architect would have to look for all the available Web Services in the public/private service directories. This step can be automated using various Web Service discovery frameworks and solutions such as Agent based discovery considering QoS (Al-Masri, *et al.,* 2007). collaborative tagging system Chukmol, *et al.,* 2008).

2. **Implement Web Services**: TPA implements all web services clients and the data extracted from these clients is stored in the TPA's local storage. The Web Service clients are abstract representation of the actual web service. The client communicates with the Web Service via SOAP messages.

3. **Local Storage**: The trip planner application TPA, stores data locally. The data in the local storage takes advantage of RDF and OWL for structuring and representing it. The ontology stores the name of the web service client, the service provider and the detailed data returned by the service. Thus, the stored data in the local storage can be divided into two categories;

a. Data about flight, taxi and hotels details. For Example, name of flight, taxi and hotels, rooms available in a hotel, available taxi services and their costs, availability of flight to a source to destination etc.

b. Data about the Web Service providing the flight, taxi and hotel booking facility. For example, service provider is name, name of the implemented Web Service client etc.

4. **Web Service Composition Engine**: This involves selecting and invoking those web services automatically, that satisfies the goal. The Web service composition engine first looks/searches by querying the locally stored RDFs for relevant Web Services. RDFs are in the local storage of TPA server and Simple Protocol and RDF Query Language (SPARQL) queries extracts result from them. After locating appropriate Web Services, the final stage involves putting together the results automatically.

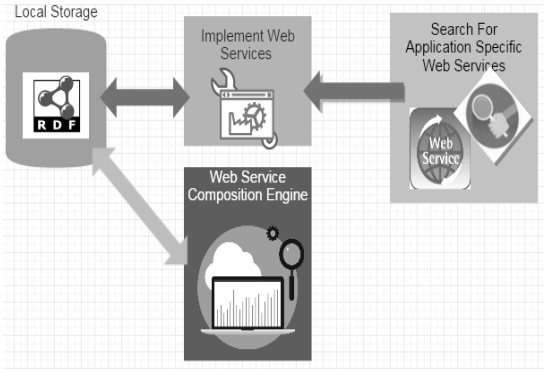

Fig. 4: Automatically Composed Web Services using Semantic Web Framework

4. RESULTS AND DISCUSSION

4.1 Overall Working

The ACWS using Semantic Web proposes a localised solution specially designed for addressing a particular scenario (e.g. trip planning, as considered in this research). The TPA system requires a complete knowledge about all existing Web Services regarding booking of hotels, flights and taxis. This is crucial for the system as all booking follows because of this information.

The process of Web Services composition starts with searching of all relevant Web Services specific to trip planning scenario (as proposed in this research). A UDDI (Universal Description, Discovery and Integration) look up takes place where the UDDI mentions the list of Web service descriptions (WSDL). Selection of relevant WSDLs follows with the implementation in the TPA's system. All implemented WSDLs become the part of TPA. The TPA business logic stores this information about flight, taxis and hotels returned upon the invocation of TPA's WSDL clients in the TPA's semantic local storage.

The foundation of TPA local storage is RDF to make the system intelligent. As the RDF are open XML-based data representations, the querying cannot be done using simple SQL. A more specialized querying mechanism is SPARQL.

4.2 Performance Evaluation and Analysis

The evaluation of the implementation of TPA that implements the proposed solution of ACWS using Semantic Web requires an experimental approach. Therefore, an experiment was set up to verify the system results in a practical scenario trip planning. The respondent used TPA and one of the famous trip-planning website; www.bookmytrip.com. The aim of the experiment is to compare the facility and ease delivered though TPA in comparison with other similar online trip planner that does not have the proposed ACWS using Semantic Web framework. In order to ensure uniformity and consistency of scenario for respondents, the experiment had only a single task of planning a trip.

The time consumed in planning a trip by respondents is mentioned in **(Table 1)**. This shows that majority of the user were able to complete the booking in first five minutes of their time. Although, a higher number of users found TPA more convenient in usage compared to book my trip.

Table 1: time consumed by number of respondents for planning a trip using TPA **(Fig. 5)** shows a graph where the results from the experiment comprehensively illustrate the difference in time taken for planning a trip.

	TPA	www.bookmytrip.com
time < 5	13	11
5 < time < 10	8	7
10 < time < 15	9	8
15 < time <20	5	9

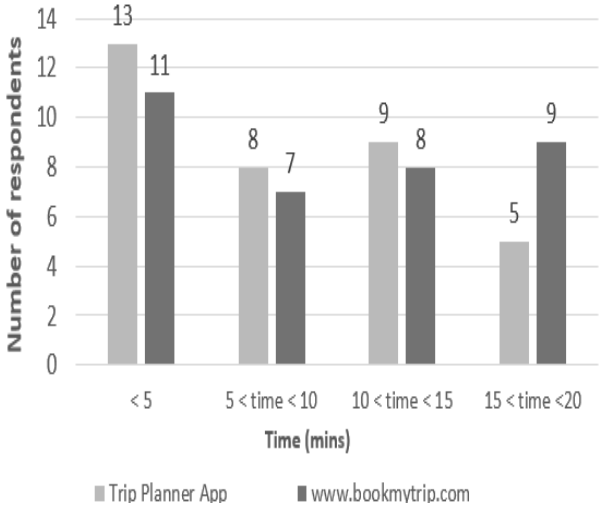

Fig. 5: Comparison of time consumed by respondents of the experiment, in planning a trip using TPA and www.bookmytrip.com

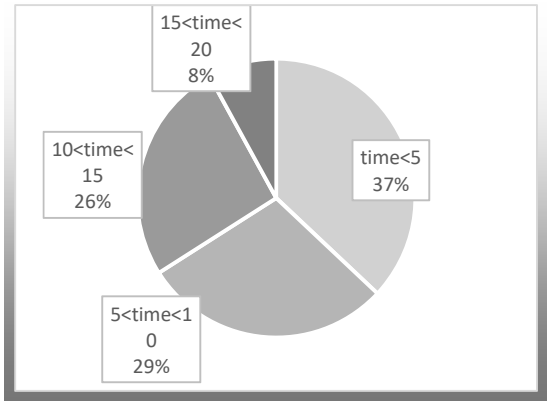

Fig.6: ratio of respondents with respect to time consumed in planning trip with TPA

5. CONCLUSION

TPA assists in reducing human effort. This research proposes a framework (ACWS using Semantic Web) and its implementation methodology. The research focuses on adoption of the ACSW using Semantic Web framework in a scenario (like trip planning) where there is a need to coordinate and compose various Web Services to achieve one goal. SPARQL queries help to efficiently query RDFs and fetching data from it since the data is represented using OWL's vocabulary.

In the future, the proposed ACWS using Semantic Web framework can be implemented in TPA to test the working and acquire the results. The research can be continued with working on more quality factors such as comfort level of the user and other preferences to produce more customized results. The performance of ACWS framework can be compared with other technologies and the performance lagging can be evaluated to improve its performance.

REFERENCES:

Al-Masri, E., and Q. H. Mahmoud, (2007). Qos-based discovery and ranking of web services. In Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on 529-534. IEEE. Held in Turtle Bay Resort, Honolulu, Hawaii, USA.

Chukmol, U., Aïcha-Nabila and Y, Amghar. (2008)."Enhancing web service discovery by using collaborative tagging system."Next Generation Web Services Practices, NWESP'08. 4th International Conference on. IEEE, 2008.

Dew, P., M. and S. Nizamani. (2011) "QA Com PS: a quality-aware federated computational semantic web service for computational modellers." URL http://weblidi.info.unlp.edu.ar/worldcomp-mirror/SWW3720. pdf.

Domingue, J, D. Roman, and M. Stollberg. (2005) "Web service modeling ontology (WSMO)- An ontology for semantic web services."Position paper at the W3C Workshop on Frameworks for Semantics in Web Services..

Fawzy S., and M. A. Youssef, (2003) "Internet-based and its impact on manufacturing and business operations", Data Systems, Vol. 103, 8, 546 – 552.

Hull, R., and J. Su. (2004) "Tools for design of composite web services." Proceedings of the ACM SIGMOD international conference on of data. ACM.

Kopecky, J., T. Vitvar, C. Bournez, and J. Farrell, (2007). Sawsdl: Semantic annotations for wsdl and xml schema. Internet Computing, IEEE, 11(6), 60-67.

Lara, R., (2004) "A conceptual comparison of WSMO and OWL-S." *Web services*. Springer Berlin Heidelberg, 254-269.

Marconi, A., M. Pistore, and P. Traverso, (2008). Automated Composition of Web Services: the ASTRO Approach. IEEE Data Eng. Bull., 31(3), 23-26.

Peltz, C. (2003). Web services orchestration and choreography. Computer,36(10), 46-52.

Pathak, J., S. Basu, and V. Honavar, (2007) Assembling composite web services from autonomous components. Emerging Artificial Intelligence Applications in Computer Engineering, 160, 394-405.

Peer, J. (2005). "Web service composition as AI Planning-A Survey." University of St. Gallen

Rao, J. and X. Su, (2005). A survey of automated web service composition methods. In Semantic Web Services and Web Process Composition 43-54. Springer Berlin Heidelberg.

Sheng, Q. Z., X. Qiao, A. V. Vasilakos, C. Szabo, S. Bourne, and X. Xu, (2014). Web services composition: A decade's overview. Information Sciences, 280, 218-238.

Sapkota, B. (2010). Architectures, Concepts and Technologies for Service Oriented Computing: proceedings of ACT4SOC , 4th International Workshop on Architectures, Concepts and Technologies for Service Oriented Computing in conjunction with ICSOFT 2010, Athens, Greece.

Xiong, Z. G., Z. L., Zhai, X. M., Zhang, and X. W. Xia, (2011). Grid workflow service composition based on colored petri net. JDCTA: International Journal of Digital Content Technology and its Applications, 5(5), 125-131.