

**The Journal of
Mathematics and Computer Science**

Available online at

<http://www.TJMCS.com>

The Journal of Mathematics and Computer Science Vol. 4 No.2 (2012) 153 - 158

**Presenting New Framework for Semi-automatic Composition
of Web Services**

Sajjad Baghernezhad^{1*}, Mohammad Eiman Jamnezhad², Saeedeh Baghernezhad³

¹Department of Computer, Darab Branch, Islamic Azad University, Darab, Iran,
Sbaghernezhad@yahoo.com

²Departments of Computer, Beyza Branch, Islamic Azad University, Beyza, Iran,
E.jamnezhad@gmail.com

³ Regional Information Center for Science & Technology, Shiraz, Iran,
Saeedehbaghernezhad@yahoo.com

Received: January 2012, Revised: April 2012

Online Publication: June 2012

ABSTRACT

Many recent studies regarding web focused on the web services composition. The main advantage of the web services over other distributed systems technologies is the possibility of automatic services composition with previous ones so it is possible to do a series of recommended operations. Many techniques have been presented to realize the idea, but it is still challengeable how to compose such services to be both sufficient and realizable structurally. In the study we present some techniques to be used to facilitate the discovery, choice and semi-automatic composition of incongruous web services. The framework has three main qualities distinct from other activities in the same field; first the algorithm makes possible the discovery and the composition of present services according to the user's needs in low error possibility. Meanwhile, it differentiates between different levels of workflows and finally the proposed framework permits the user to distinguish and scrutinize the workflow with high authority.

Key words: Composed web service, composition of web services, automatic integration of services.

* Corresponding Author

1. INTRODUCTION

It is possible to consider a web service as a part of a workflow including complicated functions to meet the clients' needs. Web services are so independent, descriptive and modifier of the approach that they can to be distributed and documented in the web. Nowadays more and more the factories and organizations do their main and related duties through internet. So it is an effective step to have a useful and effective choice and present different services to organizations through web during performance led to develop web service approaches. In fact, if no web service can meet individually the user's needs, it is necessary to have the possibility to compose present services to do it [1].

This process encourages considerable researches concerning web services in both universities and industries. Several innovations have been in the study concerning the web services to have platforms and languages to compose easily the incongruous systems. In fact, the notions such as SOAP[2], WSDL[3], UDDI[4] and DAML-S[†], define standard approaches to describe and present the request. Meanwhile, some other cases such as BPEL4WS[‡] focus on presenting composed services in which the processing and link between the services have priority. Contrary to all of these works it is still very difficult to integrate the web services and it is impossible to encounter all of these processes manually yet. Generally the complicity is composed of following cases: first the numbers of services available to web have increased considerably in recent years and one expects to search a vast place of web services; secondly the services can be created and updated accidentally. So it is necessary to update the composition system during execution and any decision should be taken on the basis of the daily information. Thirdly the web services can develop through different organizations benefiting from different notion models to describe the service. In fact, there is no unique language to describe and assess the web services in similar cases; so the web services integration by an automatic or semi-automatic device is developing yet; that is why several methods have been proposed in this regard [5].

We present a framework to automates the web services composition in this study. The technique proposed in the study has important qualities. First we differentiate between different levels of workflows by the technique and then a composition method adapted with our data (Including present sources) is used in the framework to find any fault.

Second section is to examine the basic notions concerning web service and their composition techniques. In third section we examine our proposed framework and its details. And finally in fourth section we present the conclusion and analyze our following activities.

2. PRIMARY CONCEPTS

[†] DARPA agent markup language for services

[‡] Business Process Execution Language for Web Services

Workflow

“Workflow” is the performance procedure of the different proceedings in different work fields executed as a set of serial functions to achieve the same goal.

Workflow is approximately a new subject used nowadays vastly. It is interesting that the performance of each activity is classified in a different level of a serial list of priorities [6].

The composition of web services

Many studies have been done in field of the composition of web services in two methods: workflow and artificial intelligence program [7]. In relation to workflow it can be said that there is an integrated service similar to some steps of activities set in many methods in which the description of an integrated service includes a set of indivisible services with “installation algorithm” and “stocks interaction” among the services. In fact, a workflow as it is clear by the name it defines the work processes flow [8].

Nowadays frequent techniques composing manually web services through workflows make the user encounter a low level view with details [9,10,11], but many automatic methods composing web services automate all step of the composition by using artificial intelligence method or another technology similar to it. The framework has no mechanism to correct related fault(s); for instance, if the necessary service is not executable on due time for any reason, the automatic composition process is damaged and subsequently no difference appears between the abstract and applied designs; that is why it is difficult to share the programs [12,13]. So many methods have been studied and used to facilitate the services composition automatization in order to correct the fault [14].

3. THE PROPOSE FRAMEWORK

The method proposed in the study uses the web services composition on the basis of workflow. The architecture of the framework is shown in Figure 1. By the mechanism each detail can be considered as a service. As you see the automatic composition framework can begin to work by receiving the user's request through the workflow management. The framework includes two general kernels and four partial services: abstract workflow generator, applied workflow generator, reasoning engine, assignment engine, abstract workflow set and rules bank.

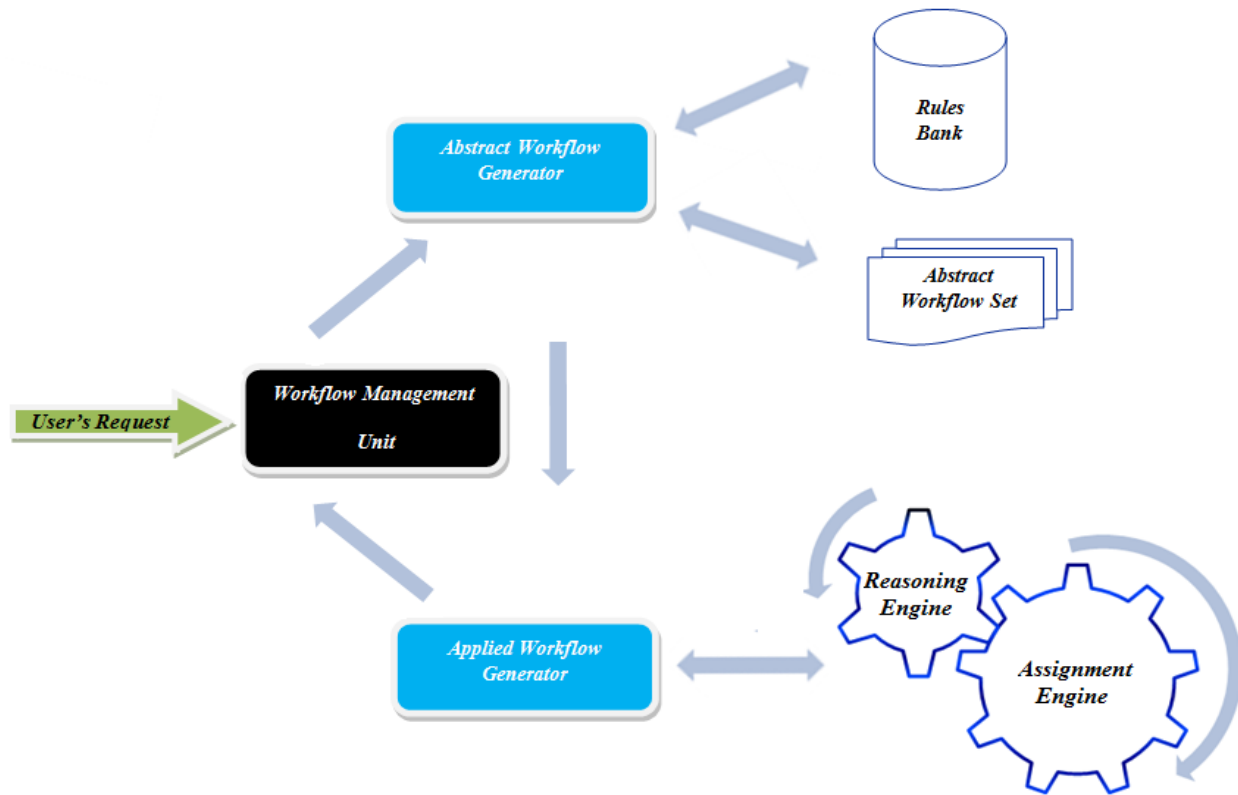


Fig 1: The architecture of the presented framework

The workflow management unit harmonizes all the process and directs the messages flow among the details. Abstract workflow generator accepts the entered request and creates an abstract workflow. Here all activities, inputs and outputs are used with their logic or real names. First abstract workflow generator searches the abstract workflow set to define if there is any previous similar request; if it has been proposed, proceeded and replied before, the found abstract workflow is returned to the management unit; otherwise, the request is presented to the reasoning engine to model a process by virtue of the present principles in the rules bank and prepared according to the request. If the rules bank does not include the new requested proceeding model, the workflow management unit tries to recover an ideal composition of the duties finally led to a set of activities based on inputs, outputs, prerequisites and later conditions and directly affects the rules bank so adapts the analyzed process with different users' tastes and needs; then (In any conditions), applied workflow generator service receives an abstract workflow and focuses its activities on the assignment of individual user's needs with some samples of concentrated web service and the samples installed on the samples on the same moment. The activity points the proposed framework benefiting from present web services and supports directly reasoning and assignment engines. If the applied web services is successfully assigned to abstract workflow, a performable sample has been produced and is returned to the management unit; otherwise, the reasoning and assignment engines try to gain a composition of present web services which is

performable easily even without considering the quality and other nonfunctional standards.

Sometimes it is not possible to complete the user's request because there is no performable web service; so applied workflow generator asks for help from the management unit and announces its need concerning a substitute abstract workflow; this process corrects mostly the fault.

The place of the abstract workflow set is a special service making the abstract workflows source available. This source reserves regularly the generated abstract workflows. If any similar requests proposed later, it can play an effective and useful role in the system in less time and hereby it promotes the efficiency and exploitation of the proposed technique.

The rules bank reserves the acceptance models as a series of general rules of web services integration along an abstract workflow formation.

Reasoning engine supports applied workflow generator. The service presents specially a traceable algorithm to produce a chain of services able to meet the output needs by the inputs.

The assignment engine presents an intelligent assignment mechanism whose main duty is adapt applied services with the abstract ones and convert them to an ideal applied workflow. In fact, the interaction between the assignment and reasoning engines creates an applied workflow for the applied workflow generator.

It is possible to consider all main details as the services benefiting from the Apache engine. It is possible to describe the services by using DAML-S. Meanwhile, the assignment and reasoning engines can benefit from the DQL/JTP servers to classify and perform their described duties. Also the abstract workflow set can include DAML-S profiles and proceeding models. DAML-S produces a type of abstract profile as the production proceeding model. The abstract workflow passes through the applied workflow generator to adapt with all present individual, necessary services. Finally BPEL4WS document including the addresses of assigned services is produced and then the main regulated program is performable.

4. CONCLUSION

In the paper we propose a framework to facilitate the semi-automatic composition of web services. We describe the framework and mention the list of their relations. Some of the important qualities of our mechanism are as follows: fault finding, adaptive engine, benefiting completely from the composition mechanism, differentiation between different levels of workflows in order to benefit again and high quality supervision on the work process by the user. These qualities assure that our framework is better than other automatic techniques in viewpoint of creating performable workflows. Meanwhile, it includes implicitly a device to distinguish specialist users and registers the gained

data. Our following activities focus on installing a real framework in the world out or theoretical subjects.

REFERENCE

- [1] B. Benatallah, M. Dumas, and Q. Z. Sheng, Facilitating the Rapid Development and Scalable Orchestration of Composite Web Services. Distributed and Parallel Databases, An International Journal, 2005.
- [2] Simple Object Access Protocol (SOAP). <http://www.w3.org/TR/SOAP>.
- [3] Web Services Description Language (WSDL). <http://msdn.microsoft.com/xml/general/wsdl.asp>.
- [4] OASIS Open. UDDI org. <http://uddi.xml.org/uddi-org>.
- [5] F. Casati, and M. Ch. Shan, Dynamic and Adaptive Composition of E-Services. Information Systems, 2001.
- [6] F. Leymann, and D. Roller, Production Workflow: Concepts and Techniques. Prentice Hall, Upper Saddle River, NJ, USA, 2000.
- [7] J. Rao, and X. Su, A Survey of Automated Web Service Composition Methods, in First International Workshop on Semantic Web Services and Web Process Composition, 2004.
- [8] S. Baghernezhad, M.H. Yektaei, and M.Abbasi, Review of Method to Reduce Response Time of Integrated Web service in the University. Iranian Student Conference on Information Technology, 2011.
- [9] E. Wu, E. Sirin, J. Hendler, D. Nau, and B. Parsia, Automatic Web Services Composition Using SHOP2. Twelfth World Wide Web Conference, 2003.
- [10] S. McIlraith, and T.C. Son, Adapting Golog for Composition of Semantic Web Services. In Proc. of the 8th International Conference on Knowledge Representation and Reasoning (KR '02), France, 2001.
- [11] E. Deelman, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, K. Vahi, K. Blackburn, A. Lazzarini, A. Arbree, R. Cavanaugh, and S. Koranda, Mapping Abstract Complex Workflows onto Grid Environments. Journal of Grid Computing. Vol. 1, 2002.
- [12] K. Sycara, M. Paolucci, A. Ankolekar, and N. Srinivasan, Automated Discovery, Interaction and Composition of Semantic Web Services. Journal of Web Semantics, Volume 1, Issue 1, December 2003.
- [13] E. Sirin, J. Hendler, and B. Parsia, Semi-automatic composition of web services using semantic descriptions. In Web Services: Modeling, Architecture and Infrastructure Workshop in conjunction with ICEIS, 2002.
- [14] S. Baghernezhad, M.H. Yektaei, and M.Abbasi, How to Optimize Efficiency in Integrating Web Services Dynamically. 3rd National Conference on Computer Engineering and Information Technology, 2011.