# A Review on Web Service Selection for Web Service Composition

**Kartik N. Shah , Shantanu Santoki, Himanshu Ghetia and RA. K. Saravanaguru**
School of Computer Science and Engineering, VIT University, Vellore, India

**Abstract**

**Since there are lot of Web services to perform same functionality, we need to select a proper Web Service based on QoS requirements. In this paper, we will study different Web Service selection techniques which are available in market. We will compare these web service selection methodologies using different parameters. Then we can recommend a particular web service for web service composition.**

**Index Terms**—*Selection of Web Services, Quality of Service, Web Service composition*

## I. INTRODUCTION

TECHNOLOGY of Web Service plays an important role in constructing application via Internet. The Web services composition is gaining a considerable area of an approach to the effective integration of heterogeneous, autonomous and distributed applications to build more sophisticated and Value Added Services (VAS). We need Algorithms to select the best services for the tasks of the composition of Web Services. QoS (Quality of Service) is one of the important criterions to select the best service out of available service for the tasks of composition. The main objective of selecting an algorithm is to maximize the QoS of the composition.

### A. Motivating Example

Consider the real time example of IRCTC web site. Web site is having main functionality of online rail ticket reservation. As one person want to go from 1 place to another place, he can use many options available with him like flight, cab etc. If the journey is too much long distance then sometimes we prefer to use first flight then train and then cab. Sometimes it is possible that flight is not going to particular area then we need to go for nearest airport and then we need to go by train. Sometimes it is possible that we are not having train till our destination, than we require a cab facility for that. So, the goal of reaching from our source to destination takes too much time. Sometimes we need to take a halt of 1 night in a hotel. So, in this case we require lot many reservations. In that case, consider the following figure.


Figure1: Motivating Example

As Figure 1 shows, we can say that IRCTC maintains all the basic need things to go from 1 place to another into 1 website. This example motivates that if we take all the services into 1 service then we can get everything at one place. Another example can be Flipkart.com which also best example for service composition in which, we can buy all the things category wise.

### B. Web Service Composition

From past few years, the WS selection problem has been extensively studied. Older works was focused on optimization of the WS for a single task, while the latest ones focus on the selection of WSs based on satisfying the QoS requirements of a workflow (or composite WS). For optimizing the selection for the entire process, computation of an optimal set of WSs for each possible execution path based on a weighted combination of QoS measures in the process [13]. When we want to decide a WS for a given activity t, the most popular execution path on which t appears is selected, and the WS assigned to t in that path is to be taken. For speeding up the computation, techniques of integer programming are employed. However, there can be many number of possible execution paths, especially when loop is present. Besides that, the failure of WSs at runtime requires the re-computationfor the set of new integer programs, which is practically not feasible. Another scheme for optimizing the end-to-end QoS various flow structures is proposed [14]. They have used a utility function derived from the QoS and formulated the problem of optimization as a general flow problem. Some tech- niques are concerned with only single QoS measure. Menasce proposed an estimation scheme to estimate the throughput of a composite WS from those of its constituent WSs. Then it uses throughput as a basis for selecting a WS [15]. Grassi and Patella proposed a new framework to recursively aggregate the reliability of a composite WS based upon those of its constituent WSs [16]. Many more works have been done to the derivation of other QoS measures, such as cost, response time, availability, fidelity, of a composite WS out of those of its constituent WSs [15].

The composition of Web services can be viewed as a three step process: 1) Composite Web service specification, 2) Select the component of Web services, and 3) Execute the composite Web services. During the first step, the user submits the goal he/she wants to have in the composite service achieves, along with some preferences and constraints that needs to be satisfied [12]. Workflows also can be used for modeling the composite Web service specification. In the second step, component of Web services that fulfill the user's goal are selected among the set of available services. This WS selection can also be done manually (steps 1 and 2 are integrated) or it can be automatically decided by the system. When the component of WSs are selected at design time, in the third step of the composition process, we can execute the selected component WS. While at runtime, selection and execution of component WS are integrated. The selection is described as dynamic in this case.

### C. Web Service Selection

Many works have been done for Web service selection. Design of a composite Web service to ensure correct and reliable execution but also optimal QoS remains an important challenge [1]. But, WSs composition based on transactional properties ensures us a reliable execution; however, it may not be an optimal QoS composite Web service is guaranteed. Moreover, the composite optimal QoS Web services may not guarantee a reliable execution of the resulting composite Web service. So, transactional -aware and QoS-aware should be integrated. However, the problem is generally addressed with the QoS side or with the transactional side separately. The conventional QoS-aware composition approaches [2], [3], [4], [5], [6] does not consider the transactional constraints at the time of composition process, likewise transactional-aware ones [7], [8], [9], [10], [11] do not consider QoS. QoS-oriented SOA for WS selection: Because the existing UDDI (Universal description, discovery and integration of Web services) model does not support the searching and the publishing of the WS QoS, many approaches have been proposed to improve the existing UDDI model to support the publishing and the searching of the WS QoS [1]. Based on the improved UDDI model, the WS selection can be made as this mode: the user searches the required WS with his QoS constraints, and the register center supplies a list of Web services to meet his QoS preferences [2-4]. However, the user does not have experiences on how much the WS QoS parameter values are suitable or unsuitable in most cases.

Figure 2 shows the high-level framework of selection model
of the multiple-aspects web services selection model [18].

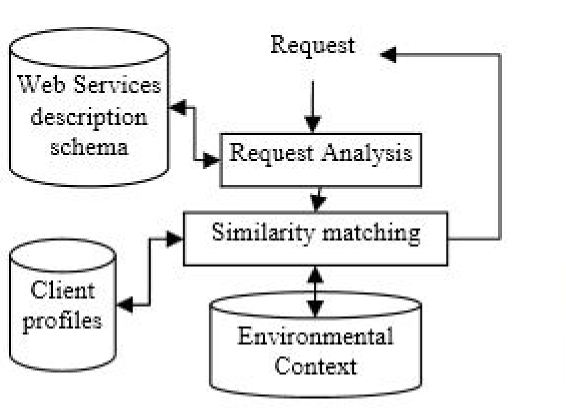


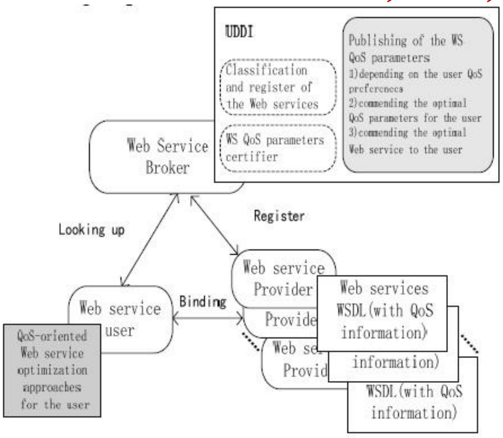

Figure2: Web Service selection model

### D. Web Service Reliability

The reliability of a WS can be defined as the probability of executing the WS successfully. In dynamic environment, too many works have been proposed in the literature to derive the reliability of a Web Service. A methodology to test the reliability of a WS effectively was proposed by generating test cases of different categories [17]. The reliability of a WS may change over time, to overcome this, an architecture-based reli- ability model has been proposed for incrementally updating the reliability of an atomic WS by contently generating test cases and aggregating the test results using a voting scheme[26]. The reliability of a composite WS can be derived by aggregating the reliabilities of individual WSs based upon the occurrence rate of each flow pattern.

## II. EXISTING WORK

QoS oriented Web Service architecture can be shown by following figure3 [19].




### A. Composition Requirement Tree Approach:

A composition requirement tree is a Weighted AND-OR tree. The leaf node contains the QoS property or service offer. The composition operator AND/OR refers to internal node. The label WXY, is used to represents the preference for the sub-tree rooted at the node Y while traversing from root to leaf on the edge between any two nodes X and Y, i.e. the edge label represents the preference to either simple or composite requirement. The leaf node represents a simple requirement (SR) and any sub-tree rooted at internal node represents the composite requirement (CR) [20].

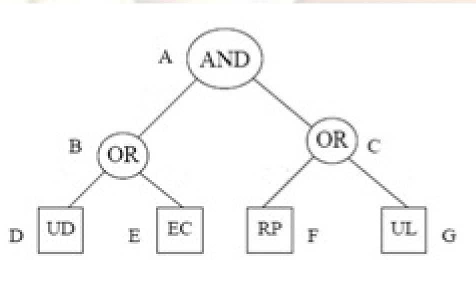



Figure4: CRT example

At the leaf nodes, the selection mechanism performs the following actions:

(1) Scaling and

(2) Ranking.

At the internal node, the selection mechanism performs two actions:

(1) Filtering and

(2) Ranking.

which are dependent on the type of the internal node (AND/OR).

### B. Selection process based on mediator:

Factors needs to consider are: time, cost, reliability and fidelity. Web services are integrated into composite services. Every business logic is expressed as a process model. The business process model of a composite service is to specify the structure combining component services and the interaction between them[22].

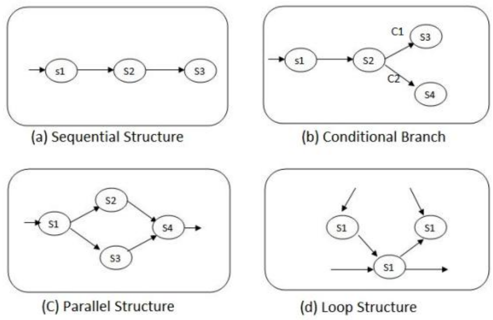



Figure5:Web Service Composition

Two Web services can be composed in one of the four ways sequence, conditional branch, loop structure and parallel. If (a) in Figure 5 is the process model of the composite Web service then two available execution paths may exist as shown in (b) and (c) of Figure 6 [22].

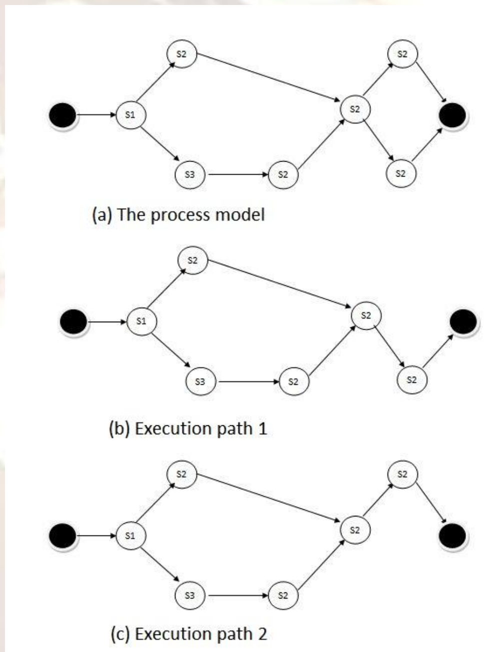



Figure6: Available Execution paths

Suppose that the set of execution plans p = {P1, P2, …
, Pn} for the given execution path. Then, for each P ∈ p,
P contains Web services which are selected to execute the business process according to the path. We consider 4 generic quality criteria to evaluate the QoS of Web services [22]:

(1) Execution cost: For a service 's', $q_{cost}$ is the amount of cost (money) requester has to pay to invoke the service 's'

$$Q_{cost}(P) = \sum q_{cost}(s_i)$$

(2) Response time: For a service 's', $q_{time}$ is the

time between request sent and response received and 'n' is number of execution paths.

$Q_{time}(P) = (\sum q_{time}(s_i))/n$

(3) Reliability: $q_{reliability}$ is the probability of responding the request in expected time for service 's'

$Q_{reliability}(P) = \exp(1/n \sum \ln(q_{reliability}(s_i))$

(4) Availability: $q_{availability}$ is the probability of service
's' is accessible for at particular time.

$Q_{availability}(P) = \exp(1/n \sum \ln(q_{availability}(s_i))$

**C. Selection based upon QoS value prediction:**

For a system of selection of web services, let M be training users and N be Web Service items. The relation between M
and N, which is denoted by M × N is known as user-item
Matrix.
The method of Recommendation can be shown by following
Figure7 [21].

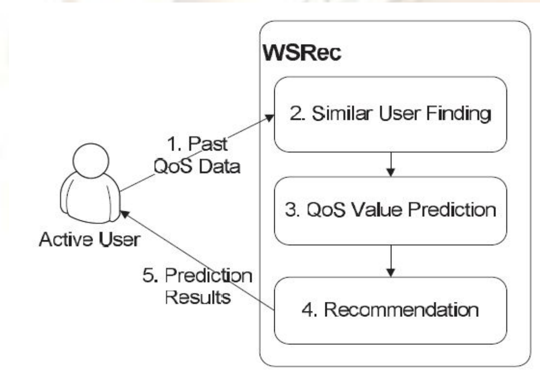



Figure7: QoS prediction procedure

This method first finds the similar neighbors and then it predict the missing values and then it goes for selection of web service. It can be carried out by following ways [21].

- For the web services which are functionally equivalent, based on QoS performance and prediction, optimal is selected.
- For the web services which are not functionally equiva- lent, top k web services, which are good, are suggested for selection.
- The top k active users having good QoS predicted values can suggest to find potential customers.

**D. A Tool framework for composite Web services:**

This tool framework automatically realizes orchestration and verification of composite Web Services based upon web services which satisfies user requirements. The framework is as follows[23].

➔ According to functional requirements, the developer de- signs the control flow structure of composite Web Service and creates ACWS (Abstract Composite Web Service).

➔ Composition verifier will verify the structural soundness of ACWS. If structure is sound then it is sent to service selector. Else it should be return back to developer to modify it.

➔ Service selector will query to UDDI and QoS Broker to get the candidate service and then select the best one and invoke the service tester. If component service which can not passed by tester, it is sent back to service selector.

➔ ACWS will be transformed into concrete composite web service by composite web service transformer and send to composite web service tester.

Composite web service will be tested by the composite web service tester in its engine. If the composite web service does not meet the QoS requirements then it is sent bask to developer.

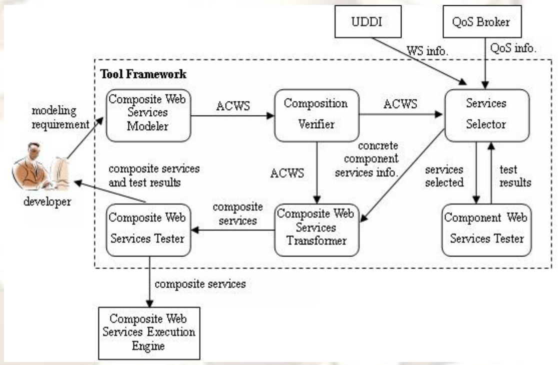



Figure8: Tool framework [23]

## III. SUMMARY

Table1 shows the comparison of 4 approaches which we have discussed in this paper. We can easily see that Tool approach is better than all the other approaches as it uses UDDI and QoS broker as database for Web service selection. Also we can note that QoS predication approach easily deals with web services, which has missing QoS values based upon QoS prediction model. Mediator approach considers Time, Cost, Reliability and Fidelity for making prediction. So, it covers almost all basic factors. So, QoS value can be more accurate. QoS Prediction and Tool approach, both can work for functionally equivalent and functionally non equivalent services.
We can summarize above mentioned selection techniques as follows:

| Measures / Approach | QoS based upon | Works for functionally equivalent or non functionally equivalent | Reliable | Work for missing QoS values |
|---|---|---|---|---|
| CRT approach [20] | Ranking | Functionally equivalent | Up to some extent | No |
| Mediator [5],[22] approach | Time, cost, reliability, fidelity | Functionally equivalent | Yes | No |
| QoS prediction approach [15], [24] | User-item matrix | Both | No | Yes |
| Tool approach [23] | UDDI and QoS Broker | Both | Yes | Yes |

Table1 : Comparison of methods

## IV. ACKNOWLEDGEMENT

The authors would like to thank the School of Computer Science and Engineering, VIT University, for giving them the opportunity to carry out this research.

## REFERENCES

[1] A. Liu, L. Huang, and Q. Li, "QoS-Aware Web Ser- vices Composition Using Transactional Composition Opera- tor," Proc. Seventh Int'l Conf. Advances in Web-Age Infor- mation Management (WAIM '06), pp. 217-228, June 2006. [2] M.C. Jaeger, G. Roec-Goldmann, and G. Muehl, "QoS Aggregation for Web Service Composition Using Workflow Patterns," Proc. Eighth IEEE Int'l Enterprise Distributed Ob- ject Computing Conf. (EDOC '04), pp. 149-159, 2004.

[3] D. Menasce, "Composing Web Services: AQoS View," IEEE Internet Computing, vol. 6, no. 8, pp. 88-90, Dec. 2004. [4] B. Wu, C.H. Chi, and S. Xu, "Service Selection Model Based on QoS Reference Vector," Proc. IEEE Congress Ser- vices (Services '07), pp. 270-277, 2007.

[5] L. Zeng, A.N.B. Benatallah, M. Dumas, J. Kalagnanam, and H. Chang, "QoS-Aware Middleware for Web Services Composition," IEEE Trans. Software Eng., vol. 30, no. 5, pp. 311-327, May 2004.

[6] W. Zhang, Y. Yang, S. Tang, and L. Fang, "QoS-Driven Service Selection Optimization Model and Algorithms for Composite Web Services," Proc. 31st Ann. Int'l Computer Software and Applications Conf. (COMPSAC '07), vol. 2, pp. 425-431, 2007.

[7] S. Bhiri, O. Perrin, and C. Godart, "Ensuring Required Failure Atomicity of Composite Web Services," Proc. 14th Int'l Conf. World Wide Web (WWW), pp. 138-147, 2005.

[8] S. Bhiri, O. Perrin, and C. Godart, "Extending Workflow Patterns with Transactional Dependencies to Define Reli- able Composite Web Services," Proc. Advanced Int'l Conf. Telecomm. and Int'l Conf. Internet and Web Applications and Services (AICT ICIW), p. 145, 2006.

[9] L. Li, C. Liu, and J. Wang, "Deriving Transactional Properties of Composite Web Services," Proc. IEEE Int'l Conf. Web Services (ICWS '07), pp. 631-638, July 2007.

[10] F. Montagut, R. Molva, and S.T. Golega, "Automating the Composition of Transactional Web Services," Int'l J. Web Services Research, vol. 5, no. 1, pp. 24-41, 2008.

[11] A. Portilla, G. Vargas-Solar, C. Collet, J.-L. Zechinelli- Martini, and L. Garcı´a-Ban˜ uelos, "Contract Based Behavior Model for Services Coordination," Proc. Third Int'l Conf. Web Information Systems and Technologies (WEBIST '07), pp. 109-123, Mar. 2007.

[12] Q. Yu, X. Liu, B. Athman, and B. Medjahed, "Deploying and Managing Web Services: Issues, Solutions, and Direc- tions," The VLDB J., vol. 17, no. 3, pp. 537-572, 2008.

[13] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, and Q.Z. Sheng, "Quality Driven Web Services Composition," Proc. 12th Int'l Conf. World Wide Web (WWW '03), pp. 411- 421, 2003.

[14] S.Y. Hwang, E.P. Lim, C.H. Lee, and C.H. Chen, "On Composing a Reliable Composite Web Service: A Study of Dynamic Web Service Selection," Proc. IEEE Int'l Conf. Web Services (ICWS '07), pp. 184-191, 2007.

[15] S.Y. Hwang, H. Wang, J. Tang, and J. Srivastava, "A Probabilistic Approach to Modeling and Estimating the QoS of Web-Services-Based Workflows," Information Sciences, vol. 177, no. 23, pp. 5484-5503, 2007.

[16] L.W. Johnson, R.D. Riess, and J.T. Arnold, Introduction to Linear Algebra. Addison Wesley Longman, 1998.

[17] J. Zhang and L.J. Zhang, "Criteria Analysis and Vali- dation of the Reliability of Web Services-Oriented Systems," Proc. IEEE Int'l Conf. Web Services (ICWS '05), pp. 621- 628, 2005.

[18] HuiNa Chua, S.M.F.D Syed Mustapha, "Web Services Selection based on Multiple Aspect Similarity Function," Proc. IEEE.WIC.ACM International Conference. Web Intel- ligence and Intelligent Agent Technology Workshops (WI- IATW'06), IEEE Computer Society Press, Dec. 2006, pp. 605- 609, doi:10.1109/WIIATW. 2006.144.

[19] Xiaopeng Deng and Chunxiao Xing, "A QoS-oriented Optimization Model for Web Service Group" 2009 Eigth IEEE/ACIS International Conference on Computer and

Infor- mation Science. pp. 903

[20] Demian Antony D'Mello and V. S. Ananthanarayana, "A Tree Structure for Web Service Compositions", ACM Compute 2009, Jan 9, 10, Bangalore, Karnataka, India.

[21] Zibin Zheng, Hao Ma, Michael R. Lyu and Irwin King, "QoS - Aware Web Service Recommendation by Collaborative Filtering",IEEE Transaction on service computing , VOL 4, No. 2, April-June 2011.

[22] Yukyong Kim and Kyung-Goo Doh, "A Trust Type Based Model for Managing QoS in Web service Composition", 2007 International Conference on Convergence Information Technology, pp. 438.

[23] Aijun Jiang, Xiaoyong Mei, Shixian Li, Fudan Zheng, "A QoS-based Tool Framework for Developing Composite Web Services", 2008 International Symposium on Information Science and Engieering,pp.663.

[24] Dr. IIavarasan Egambaram, G. Vadivelou and S. Prasath Sivasubramanian, "QoS Based Web Service Selection" ,www.ubicc.org.