http://dx.doi.org/10.14257/ijsacs.2015.3.2.02

Various Factors Affecting Performance of Web Services

Mandeep Kumar[1]

[1]*Department of Computer Science Engineering, RBGI,Hoshiarpur, Punjab, India*
thakurr.mandeep@gmail.com

Abstract

With the increased use of Internet, delivering Quality of Service has become an utmost priority for the entire web service provider. In this paper, the various factors impacting the performance of the web services are analyzed with the help of performance monitoring tool. The different web services are monitored in random and regular fashion to identify the factors that make difference in providing the service to the large number of users. The load time of the web service is taken as a performance evaluating parameter. The variation in the load times for the same web service, when monitored at different times, is carefully analyzed to find out the factors that lead to this variation and affect the performance of any web service.

Keywords: *- Web Service, Performance monitoring, load time, QoS, Pingdom tool*

1. Introduction

A web service is a software system identified by a URI, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then interact with the Web service in a manner prescribed by its definition, using XML-based messages conveyed by Internet protocols [17]. It is an application that exposes a function which is accessible using standard Web technology and that adheres to Web services standards. Web services can vary in function from simple requests (*e.g.*, credit checking and authorization, pricing enquiries, inventory status checking, or a weather report) to complete business applications that access and combine information from multiple sources, such as an insurance brokering system, an insurance liability computation, a automated travel planner, or a package tracking system. [18].

The web service model consists of three entities, the service provider, the service registry and the service consumer. Figure 1.1 shows a graphical representation of the traditional web service model:

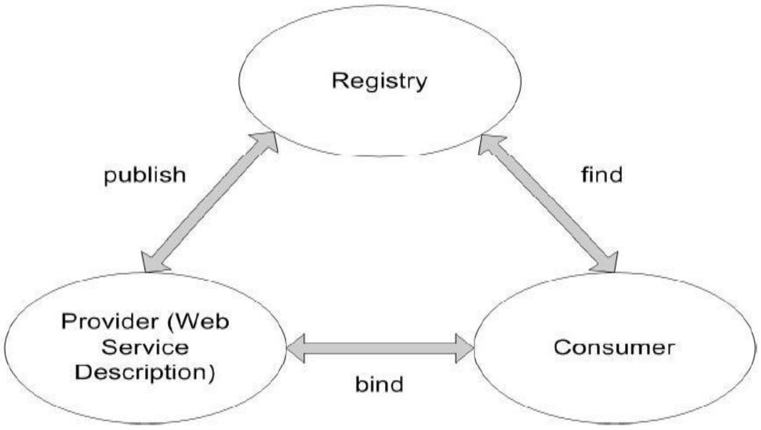


Figure 1. The Web Service Model

The service provider creates or simply offers the web service. The service provider needs to describe the web service in a standard format, which in turn is XML and publish it in a central Service Registry. The service registry contains additional information about the service provider, such as address and contact of the providing company, and technical details about the service. The Service Consumer retrieves the information from the registry and uses the service description obtained to bind to and invoke the web service [2].

Today, QoS has becomes an important criterion that determines the service usability and utility, both of which influence the popularity of a particular Web service, and an important selling and differentiating point between Web services providers. The QoS refers to the ability of the Web service to respond to expected invocations and to perform them at the level commensurate with the mutual expectations of both its provider and its customers. In the Web services' context, QoS can be viewed as providing assurance on a set of quantitative characteristics. These can be defined on the basis of important functional and non-functional service quality properties that include implementation and deployment issues as well as other important service characteristics such as service metering and cost, performance metrics (*e.g.*, response time), security requirements, integrity, reliability, scalability, and availability[18]. The monitoring of the web services performance is an upcoming area that needs attention.

2. Related Work

Much work has been carried out on the web services performance metrics to provide the customers satisfaction. Eyhab Al-Masri *et al.*, proposes the Web Service Relevancy Function (WsRF) used for measuring the relevancy ranking of a particular Web service based on client's preferences, and QoS metrics [1]. Mohamad Ibrahim Ladan *et al.*, proposes the classification of the existing metrics for Web Services along with their usage and benefits [2]. Natee Artaiam *et al.*, reviews a QoS model which covered various dimensions of service quality (*i.e.*, availability, accessibility, performance, reliability, security, and regulatory) and proposed metrics to enhance QoS measurement on the service side [3]. Amna Eleyan *et al.*, proposes a quality criteria classification that organizes web services qualities into four groups: performance, failure probability, trustworthiness and cost [4]. M. Shaban Jokhio *et al.*, presents a novel approach for testing semantic web services based on user goal specifications [5]. Dr. IIavarasan Egambaram *et al.*, evaluates some criteria to identify the levels of dynamism and automatization in service compositions. Furthermore, taking into account a model driven approach, proposes a strategy where different techniques can be used to make compositions more dynamic and automatic to handle interoperability demands [6]. Netra Patil *et al.*, suggests a quality based assessment and selection of web services [7]. Various quality parameters for discovering web services are identified and the quality assessment process is automated through Web Request Broker. Sha JingIn *et al.*, presents an approach to give QoS of service composition based on a decomposing algorithm and the numerical analysis of stochastic well-formed workflow (SWWF) models of web service composition [8]. Youngkon Lee *et al.*, presents the method describing the quality factors of Web services and SOA normally and quantitatively [9]. Julio Cezar Estrella *et al.*, depicts performance evaluation of a prototype architecture named WSARCH (Web Services Architecture) relating quality of service attributes [10]. It also evaluated performance of Web services based on a contract model and improved existing service discovery solution though the incorporation of quality-of-service issues and related metrics for ranking matches. Narayan Debnath *et al.*, proposes a Quantitative measurement method that evaluated QoS of web services, and provided a value that allowed the comparison and selection of web service that best meets the desired user characteristics [12]. Valentina Casola *et al.*, presents MAWeS, an autonomic framework that makes it possible to auto-configure and to auto-tune the composition of services, guaranteeing optimal performance and the fulfillment of given security requirements [13]. S. Maheswari et al. proposes a framework to provide QoS based Semantic Web Service Selection. The even QoS parameters execution time, response time, throughput, scalability, reputation, accessibility and availability are considered to select the best service supporting user needs [16]. Brahmananda Sapkota *et al.*, present a distributed Web service discovery architecture that is designed to be reliable, flexible and scalable [15].

Rest of the paper is organized as follows. Section 3 discusses the system environment. The result and analysis along with the simulation parameters is discussed in the Section 4. The entire work is concluded in Section 5.

3. System Environment

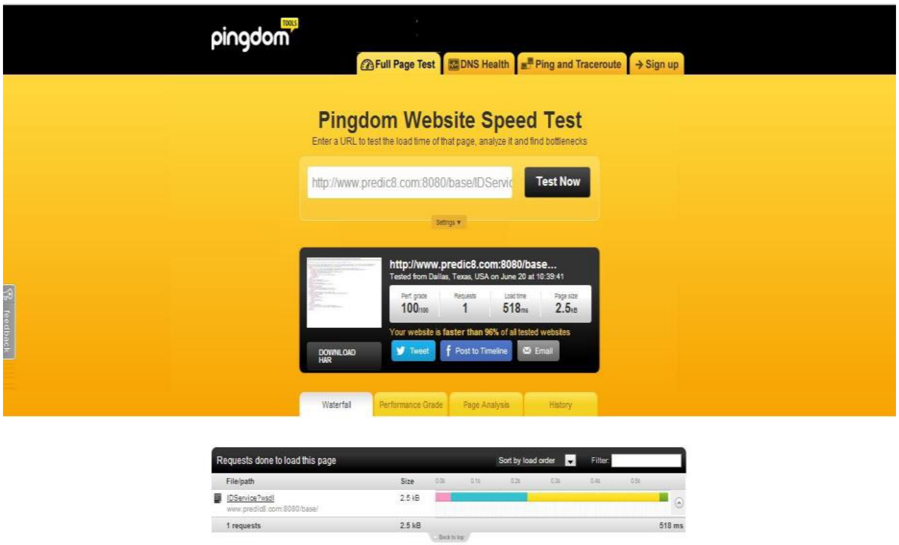
3.1. Methodology

To start with, the different Web Services from xmethod.com and servicerepository.com are selected. These selected web services are then evaluated by monitoring them on regular basis and random basis using the pingdom monitoring tool. In randomized monitoring, any web service is monitored at any moment of time using the pingdom tool. The load time of web services, DNS name and the distribution of load time is the outcome of the pingdom tool. The variation in the load time of the same web service when monitored at different times can be attributed to the various factors. Then the different factors that contributed in the variation of the load time of the web service are identified. The detailed analysis of the factors on the basis of the load time is carried out.

3.2. Monitoring Tools Used

1. **Pingdom:** The tool used for the analysis of the performance of the web services is pingdom. It examines all parts of a web page – View file sizes, load times, and other details about every single element of a web page (HTML, JavaScript and CSS files, images, *etc.*,). These elements can be sorted and filtered in different ways to identify performance bottlenecks. It automatically gives performance overview by putting together plenty of performance-related statistics based on the test result. These performance overviews can be traced over a period of time to analyze the variations in the performance of the web service. It allows testing the web service from different

parts of the world. The load time, the different components of the load time, DNS of the web service is found using this tool.



Pingdom Website Speed Test

Enter a URL to test the load time of that page, analyze it and find bottlenecks

http://www.predic8.com:8080/base/IDServic

http://www.predic8.com:8080/base...

Tested from Dallas, Texas, USA on June 20 at 10:39:41

Perf. grade	Requests	Load time	Page size
100/100	1	518ms	2.5kB

Your website is faster than 96% of all tested websites

Requests done to load this page

File/path	Size	0.0s 0.1s 0.2s 0.3s 0.4s 0.5s
IDService?wsdl www.predic8.com:8080/base/	2.5 kB	
1 requests	2.5 kB	518 ms

Figure 2. Page Analysis of the Server Response Code using Pingdom

2. **Nslookup and PING Commands:** These commands are used to retrieve the IP address of the Web service along with the Total Round trip time. The IP address is needed to find the IP Location of the web service which is further needed to calculate the distance between the client and server's position. The total number of packets sent and received can be also analyzed using these commands.

3. **IPlocation Finder:** IPloaction.net is used to find the location of any web service using its IP address. This location is needed to find the distance between the server and client's location.

4. **Distance Finder:** distancefromto.net is used to calculate the actual physical distance between the client and server's location. The distance is given in kms. It is use to check the affect of the distance on the performance of the web service based on the total load time.

4. Results and Discussions

The different web services are monitored regularly and randomly to check their performance and to work out the issues affecting their performance. The various factors that influence the performance of web services are identified and analyzed to study their impact on the load time.

4.1. Performance Evaluation Parameters

The load time is used as the performance evaluation parameter. It is defined as the time period to send a request and receive a response to that request. It is calculated in: milliseconds. The load time constitutes of the time spent in different states while retrieving the response for the request. It is the sum of the time period spent in establishing connection, waiting time, the time spent in connecting Domain Name Server, receiving time.

4.2. Result and Analysis

In this section, the non-functional parameters that affect the performance of the monitored web services are identified and evaluated based on their load time. The web services are monitored on regular and random basis. In randomized monitoring, the web services are monitored at different intervals randomly. In the randomized monitoring, the performance of the web services is analyzed by critically examining the load time of these web services taken at different intervals. While monitoring the performance, the web service host position (from which the service request is made) is also changed. Each web site is monitored 10 times with the help of Performance monitoring tool pingdom tool. The averaged value of load time is taken for each web service. This load time is critically analyzed to find out the various factors that caused variation in the load time every time when it is monitored. Similarly, in case of regular monitoring the web services are monitored at fixed regular intervals. The web services are monitored 5 times to check the load time variations in each case. After critically analyzing the load time, the below discussed factors were identified to affect the performance of the web services.

a) Distance: The actual geographical distance between the host (from which request is originated) and server (from which requested web service is retrieved) is the major factor that leads to variation in the value of the load time. Each time when the web service is monitored, the web service host position is changed using monitoring tool to analyze the impact of the distance on the load time value. The change in distance between the host and the server affects the load time of the web services as the time to establish the connection between them increases considerably. This leads to variation in the load time of the web services when monitored for the different servers residing at different geographical distances. For analyzing the distance effect, the web services which show standard deviation more than 300 ms in load time are considered for evaluation. The reason for considering these web services is that we want to find out the factors that have lead to more standard deviation in load time.

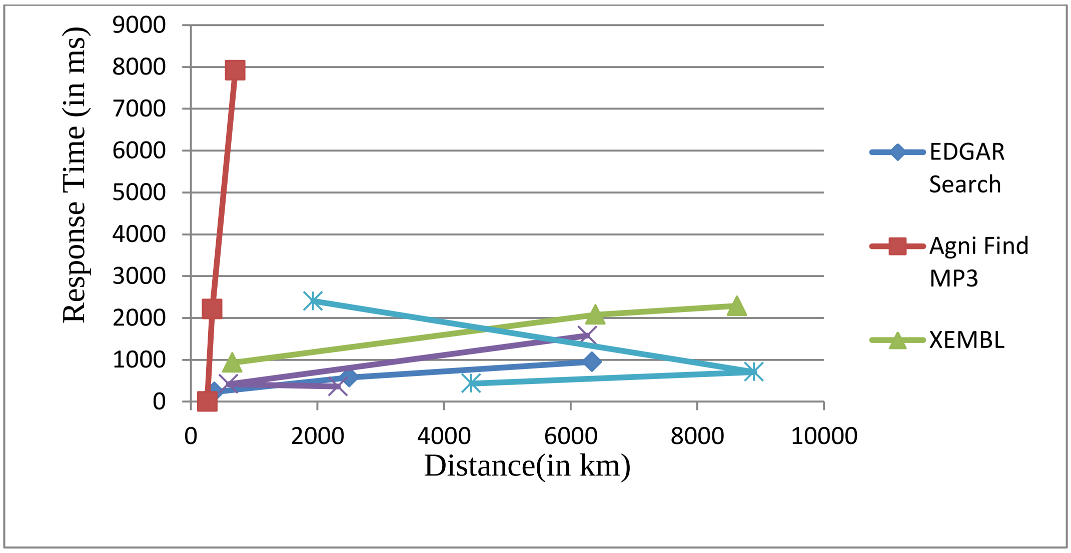


Figure 3. Effect of Distance on Load Time (Random)

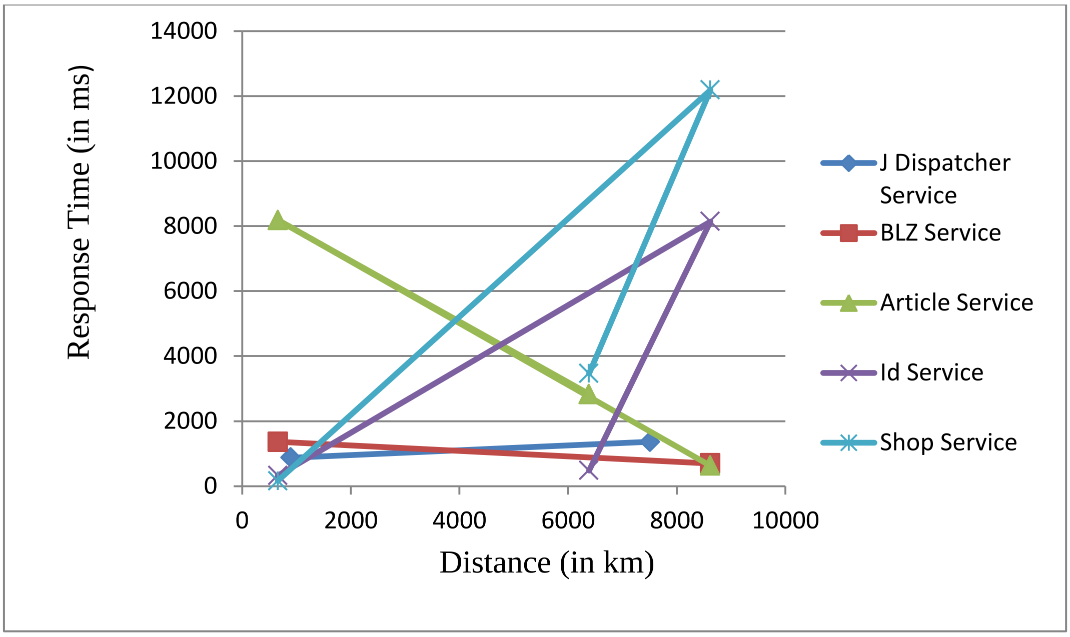


Figure 4. Effect of Distance on Load Time (Regular)

The Figures 3-4 shows the affect of distance on web services monitored on random and regular basis. It is analyzed that as the distance between the host and the web server increases, the load time in most of the cases also increases. In some cases, the load time has not varied in accordance to the distance; some other reason can be attributed to this variation.

b) Size: the size of the web page which is requested by the host also impacts the load time of that web service. The heavier the page, the more is the load time. When analyzing the variations in the load time values for the same web service over 10 times, the size of the requested web page also turns out to be a major factor. The web page consists of many hyperlinks to many other pages, images, varied size videos *etc*. The size of these individual components may greatly affect the size of the web page. The change in the design of the web service, addition of some features, deletion of some features and the change in the size of the components of the web services may lead to change in the size of that web page.

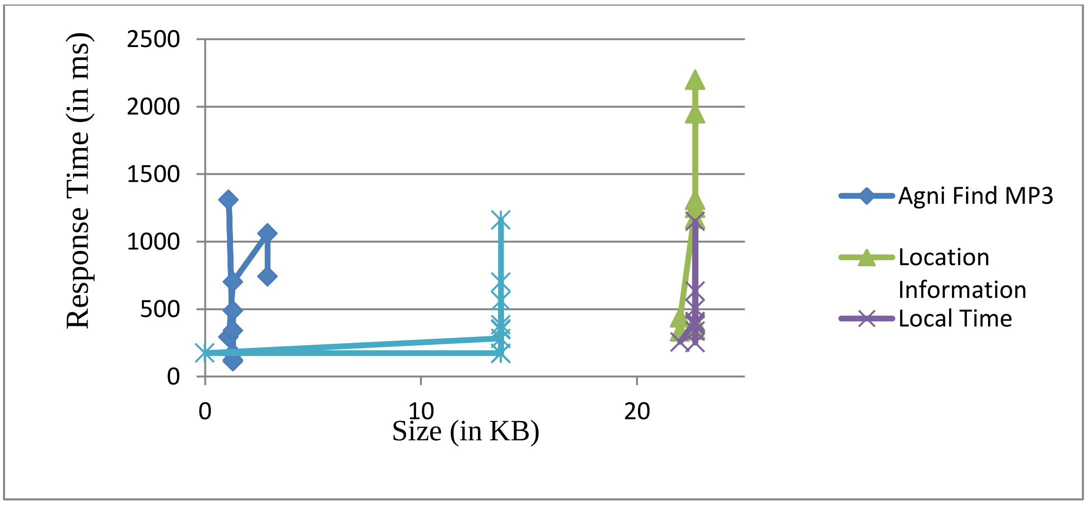


Figure 5. Effect of Size on Load Time(Random)

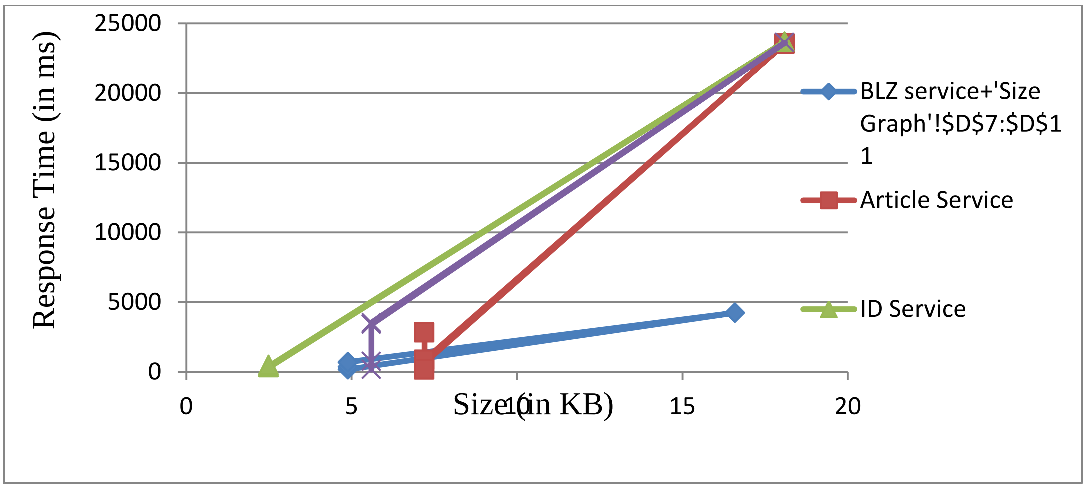


Figure 6. Effect of Size on Load Time(Regular)

From the above Figures 5-6, it is evident that the size of the web service page is also an important aspect which needs to be considered for analyzing the performance of the web page. The size of the web page can be altered by removing the heavy embedded audios and videos, by reducing the use of images and adding light weight components altogether. Thus the performance by any web service can be improved by varying the size of the web page.

c) Time: The time session of the day during which monitoring of the web services is done also affects the performance of web service, when monitored at different intervals. The reason for the variation in load time, when web services monitored over different time sessions of the day, is the variation in the load on the server during different time intervals. The load on the server depends upon the number of the host requests to be served by the server. These requests may vary invariably over the different time session of the day.

TABLE WITH FEW ENTRIES FOR RANDOM AND REGULAR BASIS

Table 1. Effect of Session of the Day on Load Time (Regular)

S.No.	Name of Web Service	Average Value of Morning	Average Value of Afternoon	Average Value of Evening
1	J Dispatcher Service	1395.0	0.0	1015.3
2	BLZ Service	360.0	1701.7	679.0
3	Article Service	510.0	630.0	13190.0
4	ID Service	332.0	366.0	12085.0
5	Shop Service	751.0	3465.0	11895.5
6	Customer Service	654.0	474.0	11888.0
7	DeFactoSF1	0.0	1665.0	965.5
8	Datos Alumno	23620.0	1920.0	1596.7
9	People Ask - Google Questions	316.0	1540.0	1592.0

Table 2. Effect of Session of the Day on Load Time (Random)

S.No.	Name of Web Service	Average Value of Morning	Average Value of Afternoon	Average Value of Evening
1	**EDGAR Search**	537.5	290.0	537.8
2	**Alan Bush Compositions**	438.0	298.0	728.3
3	**Agni Find MP3**	387.0	1060.0	586.5
4	**Xara 3D graphics generator**	524.7	956.5	442.4
5	**XEMBL**	1996.7	2290.0	1648.2
6	**Fox Central**	828.3	935.5	863.0
7	**Web Service Documentation Service**	616.5	450.5	697.0
8	**Location Information**	785.5	1265.3	887.6
9	**Place Finder**	0	534.8	2859.4

Thus, the geographical distance between the host and server, size of the web services and time session of the day are identified as the three main non functional parameters that greatly affected the performance of the web services. Thus, these factors can be considered while evaluating the performance of web services. QoS of the web services can be assured by taking the corrective action in accordance to the factor impacting the performance. Thus the basic motive of providing QoS to the customers can be satisfied.

5. Conclusion

In this paper, the performance of the web services is monitored using monitoring tool to identify the various factors that affect the performance of the web services. These web services are divided into two different categories based on the frequency of the performance monitoring. The load time which is defined as the time period to send a request and receive a response to that request, is used as the performance evaluation parameter. The close and detailed analysis of the load time for the same web service, reveals that distance between the host and server, size of web page and time session of the greatly affects the performance of any web service.

References

[1] E. Al-Masri and Q. H. Mahmoud, “QoS-based Discovery and Ranking of Web Services”, 16th International Conference on Computer Communications and Networks IEEE, **(2007)** August 13, pp. 529-534.

[2] M. Ibrahim Ladan, “Web Services Metrics: a Survey and a Classification”, International Conference on Network and Electronics Engineering, vol. 11, **(2011)**, pp. 93-98.

[3] N. Artaiam and T. Senivongse, “Enhancing Service-Side QoS Monitoring for Web Services”, 9th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, **(2008)**, pp. 765-770.

[4] A. Eleyan and L. Zhao, “Extending WSDL and UUDI with Quality Service Selection Criteria”, 3rd International Symposium on Web Services, **(2010)** April 21.

[5] M. Shaban Jokhio, “Goal-Based Testing of Semantic Web Services”, International Conference on Automated Software Engineering IEEE Computer Society, **(2009)**, pp. 707-711.

[6] Dr. I. Egambaram, G. Vadivelou and S. Prasath Sivasubramanian, “Dynamic Web Service Composition: Challenges and techniques”, International Conference on Intelligent Agent & Multi-Agent Systems, **(2009)** July 22, pp. 1-8.

[7] N. Patil and A. Gopal, “QuantifyingWeb Services on Quality Parameter s for Best-fit Web-service Selection”, International Journal of Computer Science and Application, **(2010)**, pp. 105-108.

[8] S. Jing and D. Yu-yue, “Performance Analysis of Web Service Composition based on Stochastic Well-formed Workflow”, 6th International Conference on Networked Computing (INC), **(2010)** May 13, pp. 1-5.

[9] Y. Lee and G. Yeom, “A Quality Chain Modeling Methodology for Ternary Web Services Quality View”, 5th ACIS International Conference on Software Engineering Research, Management & Application, IEEE Computer Society, **(2007)**, pp. 91-97.

[10] J. Cezar Estrella, R. Kenji T. Toyohara, B. Tardiole Kuehne, T. Caproni Tavares, R. H. C. Santana, M. J. Santana and S. Mazzini Bruschi, "A Performance Evaluation for a QoS-Aware Service Oriented Architecture", 6th World Congress on Services IEEE Computer Society, **(2010)**, pp. 260-267.

[11] A. Al-Moayed and B. Hollunder, "Quality of Service Attributes in Web Services", ICSEA, Fifth International Conference on Software Engineering Advances IEEE Computer Society, **(2010)** August 22, pp. 367-372.

[12] N. Debnath, P. Martellotto, M. Daniele Daniel Riesco and G. Montejano, "A method to evaluate QoS of web services required by a workflow", 11th International Conference on ITS Telecommunications (ITST), **(2011)** August 23, pp. 640-645.

[13] V. Casola, E. P. Mancini, N. Mazzocca, M. Rak and U. Villano, "Self-optimization of secure web services", Computer Communications journal, vol. 31, no. 18, **(2008)** December, pp. 4312-4323.

[14] S. Ran, "A Model for Web Services Discovery with QoS", CSIRO Mathematical and Information Sciences, Spring, vol. 4, no. 1, **(2003)**, pp. 1-10.

[15] B. Sapkota, D. Roman and D. Fensel, "Distributed Web Service Discovery Architecture", Advanced Int'l Conference on Telecommunications and Int'l Conference on Internet and Web Applications and Services IEEE Computer Society, **(2006)**, pp.136-141.

[16] S. Maheswari and G. R. Karpagam, "QoS Based Efficient Web Service Selection", European Journal of Scientific Research, vol. 66, **(2011)**, pp. 428-440.

[17] L. Zeng, H. Lei and H. Chang, "Monitoring the QoS for Web Services", Procs of ICSOC 2007, Springer Publications, Vienna, Austria, **(2007)** September, pp. 132-144.

[18] M. Papazoglou, "Web Services: principles and Technology", Pearson Publications, **(2008)**.

[19] U. Wahli, O. Burroughs, O. Cline, A. Go and L. Tung, "Web Services Handbook for Web Sphere Application Server Version 6.1", IBM Red Books Publication, **(2006)**.

[20] D. Bianculli and C. Ghezzi, "Monitoring Conversational Web Services", Procs of IW-SOSWE, Dubrovnik, Croatia, **(2007)** September, pp. 15-21.

Author

Mandeep Kumar has done MTECH. in Software System from Guru Nanank Dev University, Amritsar. Currently, he is working as Assistant Professor at Rayat Bahra and Institutes, Hoshiarpur. His areas of interest include Software Engineering and Network Security.