Role of QoS in Web Services

Dr. Pallavi P. Jamsandekar [1] & Shabnam S. Mahat [2]

Abstract: **Web services are self-contained, modular business process applications that Web users or Web connected programs can access over a network. Services previously possible only with the older standardized service known as Electronic Data Interchange (EDI) are now likely to become Web services. Web services are designed to be published, discovered, and invoked dynamically in a distributed computing environment. By facilitating real-time programmatic interaction between applications over the Internet, Web Services may allow companies to more easily exchange information, leverage information resources, and integrate business processes. The functionalities that can be implemented by Web services have virtually no limits, ranging from major services like storage management and customer relationship management (CRM) down to much more limited services such as furnishing a stock quote and checking bids for an auction item. However, QoS is still the big issue for web services. Quality of service (QoS) is significant and necessary for web service applications quality assurance. Furthermore, web services quality has contributed to the successful implementation of Electronic Commerce (EC) applications. We wish this paper introduces an important issues and future trends of Web services and e-business. Moreover, there are many different perspectives and purposes of web services, and various techniques to describe QoS requirements. This paper proposes a Web services discovery model in which the functional and non-functional requirements (i.e. quality of services) are taken into account for the service discovery.**

Keywords: **Web services, e-commerce, e-business, QoS requirement, QoS model.**

---

1. Department of Computer Applications, Institute of Management and Rural Development Administration, Bharati Vidyapeeth Deemed University, Sangli.
Pallavi.jamsandekar@yahoo.com

2. Department of Computer Applications, Institute of Management and Rural Development Administration, Bharati Vidyapeeth Deemed University, Sangli.
Shabnam.k59@rediffmail.com

1. INTRODUCTION

Web services are software components that communicate using pervasive, standards-based Web technologies including HTTP and XML-based messaging. Web services are designed to be accessed by other applications and vary in complexity from simple operations, such as checking a banking account balance online, to complex processes running CRM (customer relationship management) or enterprise resource planning (ERP) systems. Since they are based on open standards such as HTTP and XML-based protocols including SOAP and WSDL, Web services are hardware, programming language, and operating system independent. This means that applications written in different programming languages and running on different platforms can seamlessly exchange data over intranets or the Internet using Web services [9]. Web services refer to a family of technologies that can universally standardize the communication of applications in order to connect systems, business partners, and customers cost-effectively through the World Wide Web. Web services will ease the constraints of time, cost, and space for discovering, negotiating, and conducting e-business transactions. As a result, Web services will change the way businesses design their applications as services, integrate with other business entities, manage business process workflows, and conduct e-business transactions.

2. WEB SERVICE STACK

The goal of the Web service technology is to get applications working together over standard Internet protocol; therefore the choice of some base protocols depends on the overlay communication system we use. Actually the transport layer in Internet is implemented with HTTP and TCP/IP therefore we need upper layers which can communicate with such implementation. As for SOA, the stack can be divided into layers: core layers, like transport and messaging techniques, and high-level layers, likeservice description, publication, composition and collaboration. Both core and high level layers have several and specific functionalities that are implemented by usually more than one solution. In Figure 1 we can see the overall picture of the Web service Stack.

like service description, publication, composition and collaboration. Both core and high level layers have several and specific functionalities that are implemented by usually more than one solution. In Figure 1 we can see the

overall picture of the Web service Stack.

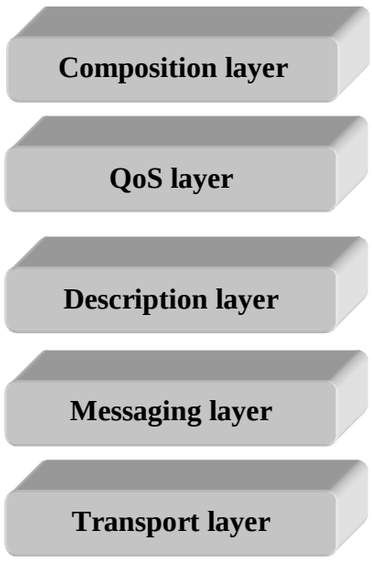

Fig 1: Web Service Stack

Here follows a list of layers and related implementations that can be identified in the Web service stack: [12].

i.Transport layer: core layer of the stack usually implemented with HTTP and TCP/IP. It can be also implemented with SMTP, FTP and other transport protocols according to the requirements. What is deemed important is that the upper layers do not have to care about the specific transport base.

ii.Messaging layer: SOAP, WS-Addressing, WS-Notification and WS-Eventing are examples of protocols that can implement these functionalities.

iii.Description layer: it is responsible of describing the public interface of the Web service so that the it can be published and invoked by the clients. WSDL and WS-Policy are examples of protocols that implement these functionalities.

4. WEB SERVICES STANDARDS

Web services rely on a set of standards to support interoperability among applications developed in different languages and running on different platforms or operating systems. One way to understand Web services is to understand Web services standards [11].
Web services standards including:
1) SOAP (Simple Object Access Protocol)

iv.Quality of service layer: this high-level layer embraces quality parameters like security, reliability and transactions. We can mention WS-Security, WS-Authorization, WS-Reliable Messaging, WS-Transaction and WS-Coordination as examples of implementation of these nonfunctional facilities.

v.Composition layer: this layer has several challenges and open issues and still many researches are focused on it. Compositions and collaborations are usually implemented using BPEL4WS and WS-CDL.

3. WEB SERVICE ARCHITECTURE

Web Services architecture requires three fundamental operations: publish, find, and bind. Service providers publish services to a service broker. Service requesters find required services using a service broker and bind to them.

Operations in Web Service Architecture:
i. Publish: Service descriptions need to be published in order for service requestor to find them
ii. Find: Service requestor retrieves a service description directly or queries the service registry for the service required
iii. Bind: Service requestor invokes or initiates an interaction with the service at runtime.

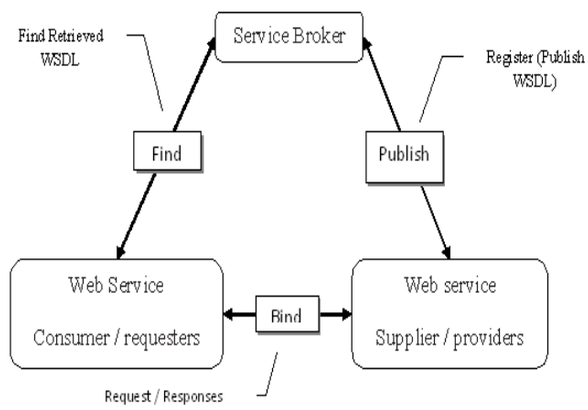

Fig 2: Web Service Architecture

2) WSDL (Web Services Description Language)
3) UDDI (Universal Description, Discovery, and Integration)

1) SOAP (Simple Object Access Protocol)

SOAP is an XML-based protocol from the W3C for exchanging data over HTTP. It provides a simple,

standards-based method for sending XML messages between applications. Web services use SOAP to send messages between a service and its client. SOAP messages are XML documents that contain some or all of the following elements [9]:

1. Envelope – specifies that the XML document is a SOAP message; encloses the message itself.
2. Header (optional) – contains information relevant to the message, e.g., the date the message was sent, authentication data, etc.
3. Body – includes the message payload.
4. Fault (optional) – carries information about a client or server error within a

2) WSDL (Web Services Description Language)

SOPE define a wired protocol which can be used for messaging, but it lacks a mechanism for describing which kind of messages should be transmitted and where to [10]. WSDL addresses this problem by providing a method for describing the communication in a structured way. WSDL file is an XML document that describes a Web service using six main elements [9]:

1. Port type – groups and describes the operations performed by the service through the defined interface.
2. Port – specifies an address for a binding, i.e., defines a communication port.
3. Message – describes the names and format of the messages supported by the service.
4. Types – defines the data types (as defined in an XML Schema) used by the service for sending messages between the client and server.
5. Binding – defines the communication protocols supported by the operations provided by the service.
6. Service – specifies the address (URL) for accessing the service.

3) UDDI (Universal Description Discovery and Integration)

UDDI is a standard sponsored by OASIS (Organization for the Advancement of Structured Information Standards). Often described as the yellow pages of Web services, UDDI is a specification for creating an XML-based registry that lists information about businesses and the Web services they offer. UDDI provides businesses a uniform way of listing their services and discovering services offered by other organizations. Though implementations vary, UDDI often describes services using WSDL and communicates via SOAP messaging. Registering a Web service in a UDDI registry is an optional step, and UDDI registries can be public or private (i.e. isolated behind a corporate firewall). To search for a Web service, a developer can query a UDDI registry to obtain the WSDL for the service he/she wishes to utilize. Developers can also design their Web services clients to receive automatic updates about any changes to a service from the UDDI registry.

5. NON-FUNCTIONAL REQUIREMENTS OF QOS :

QoS for web service applications is the ability of their services to provide added value to the best solution for requesters' enquiries, taking into account their specific requirements [2].

QoS assurance requirements include [8]:

1. **Service level agreement**. QoS should be represented by agreements which are established by negotiating between service requester and provider prior to service execution. Standard mechanisms should be provided to create and manage agreements.

2. **Service level attainment**. If the agreement requires attainment of Service Level, the resources used by the service should be adjusted so that the required QoS is maintained. Therefore, mechanisms for monitoring services quality, estimating resource utilization, and planning for and adjusting resource usage are required.

3. **Migration**. It should be possible to migrate executing services or applications to adjust workloads for performance or availability.

QoS is still the big issue for web services research and remains one of the main research questions that need to be explored. We believe that QoS should not only be measured but should also be predicted during the development and implementation stages. However, there are challenges and constraints to determine and choose QoS requirements for high quality web services. We emphasize that it is essential for developers and service providers to identify the QoS for their web services. However, the quality of a web service is not only measured by its functionality, but QoS also take into account the non-functional requirements. There are five essential QoS requirements as the main non-functionality that service providers must consider when developing their web service applications. These are as follows [7]:

1) Readiness contains the following characteristics [1][7].

i. Availability: refers to the presence of a web service for a client to connect to it

ii. Accessibility: refers to the capability of satisfying a web service request. Accessibility is the quality aspect of a service that represents the degree it is capable of serving a Web service request. It may be expressed as a probability measure denoting the success rate or chance of a successful service instantiation at a point in time. There could be situations when a Web service is available but not accessible. High accessibility of Web services can be

achieved by building highly scalable systems. Scalability refers to the ability to consistently serve the requests despite variations in the volume of requests.

2) Transaction relates to ACID property, which contains the following characteristics **[**1]
i. Atomicity: executes entire transactions or not at all.
ii.Consistency : maintains the data integrity and consistency in update transaction.
iii.Isolation : individual transactions run as if no other transactions are present.
iv.Durability : is the persistence of results.

3) Reliability refers to the capability of maintaining the service and service quality **[**5][4]. The number of failures per month or year represents a measure of reliability of a Web service. In another sense, reliability refers to the assured and ordered delivery for messages being sent and received by service requestors and service providers.
i.Completeness.
ii. Robustness: it specifies how the service reacts to invalid, incomplete input messages.

4) Speed refers to the service time and performance to improve service quality [6].
i.Service time is the length of time for services taken to provide a response to various types of requests.
ii. Performance: is the quality aspect of Web service, which is measured in terms of throughput and latency. Higher throughput and lower latency values represent good performance of a Web service. Throughput represents the number of Web service requests served at a given time period. Latency is the round-trip time between sending a request and receiving the response.

5) Security refers to authentication mechanisms, messages encryption and access control, confidentiality, non-repudiation and resilience to denial-of service attacks. Security has added importance because Web service invocation occurs over the public Internet. The service provider can have different approaches and levels of providing security depending on the service requestor.

i. Authentication

ii.Authorization

iii.Confidentiality

iv.Non repudiations

6) Regulatory: Regulatory is the quality aspect of the Web service in conformance with the rules, the law, compliance with standards, and the established service level agreement. Web services use a lot of standards such as SOAP, UDDI, and WSDL. Strict adherence to correct versions of standards (for example, SOAP version 1.2) by service providers is necessary for proper invocation of Web services by service requestors.

7) Integrity: Integrity is the quality aspect of how the Web service maintains the correctness of the interaction in respect to the source. Proper execution of Web service transactions will provide the correctness of interaction. A transaction refers to a sequence of activities to be treated as a single unit of work. All the activities have to be completed to make the transaction successful. When a transaction does not complete, all the changes made are rolled back.

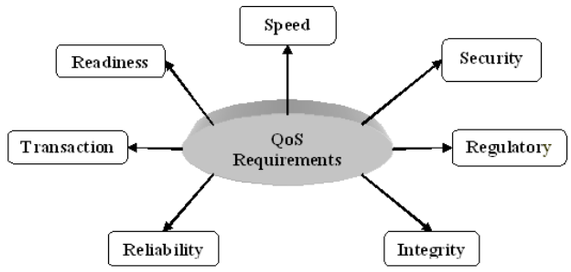

Fig 3: Components of the QoS model.

Whereas the regulated registries based on the new model can serve to the applications needing quality of service assurance. Actually, what we are looking for to discovery of web services is functional as well as non-functional requirements. In the proposed model, Web service provider needs to supply information about the company, the functional aspects of the provided service as requested by the current UDDI registry, as well as to supply quality of service information related to the proposed Web service. The claimed quality of service needs to be certified and registered in the repository. There are four rules in this model as follows:
i. **Web service supplier** offers Web Services by publishing the service into the registry like before.
ii.**The Web Service Consumer** needs the web service offered by the provider.
iii. **Web Service QoS Certifier:** Is responsible to verify the claims of quality of service for a web service before its registration.
iv. **The new UDDI registry** is a repository of registered Web Services with lookup facilities. It is different with the current model of UDDI by having information about the functional description of the web services as well as its associated quality of service registered in the repository.

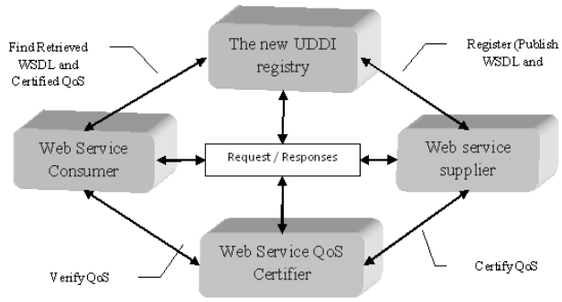

Fig 4: The new Web Services registration and discovery model.

6. CONCLUSION

Web services are the technology to bridge the gap between IT and business. Web services will ease the constraints of time, cost, and space for discovering, negotiating, and conducting e-business transactions. As a result, Web services will change the way businesses design their applications as services, integrate with other business entities, manage business process workflows, and conduct e-business transactions. The development of web service applications is an important stage, and yet, most of the current research concern with the implementation of high quality web services at the operational stage, predicting the QoS of web service applications is necessary for service providers and requesters. Therefore, service providers should provide good quality web service applications with QoS requirements being incorporated into their designs. We wish that this introductory article can shed some light for researchers to better understand importance of QoS in Web services and e-business.

7. REFERENCES

1. Mani A. and Nagarajan "A. Understanding Quality of Services for Web Services" Retrieved July 1, 2005, from: www.ibm.com/developerworks/library/wsquality.html

2. Wan Ab Rahman W.N. "A Quality of Service Prediction Model to Support Web Services Interoperability in Electronic Commerce", Interim Assessment Report, School of Computing, Science and Engineering, the University of Salford, Manchester, UK, 2008.

3. Cardoso J., Sheth A., Miller J., Arnold J. and Kochut K. Quality of Service for Workflows and Web Service Processes. Retrieved January 15, 2007 from: http://scholar.google.com/url?sa=U&q=http://chief.cs.uga.edu/~jam/home/theses/cardoso_dissert/CSM%2BQo S-WebSemantics.pdf

4. Silver G.A., Maduko A., Jafri R., Miller J.A. and Sheth A.P. Modeling and Simulation of Quality of Service for Composite Web Services. Retrieved December 24, 2006, from http://scholar.google.com/url?sa=U&q=http://www2.ac.edu/faculty/gsilver/UGA/publications/SCI.pdf.

5. Agarwal V., Dasgupta K., Karnik N., Kumar A., Kundu A., Mittal S. and Srivastava B. "A Service Creation Environment Based on End to End Composition of Web Services", Proceeding of the 14th International World Wide Web Conference, Japan, 2005, pp. 128-137, ISBN 1 59593 046 9. ACM, New York, USA.

6. Mathijssen S. "A Fair Model for Quality of Web Services". Retrieved December 6, 2006, from: http://scholar.google.com/url?sa=U&q=http://wwwhome.cs.utwente.nl/~referaat/documents/2005_03_BDATA_AND_APPLICATION_INTEGRATIO

7. "Challenges to Describe QoS Requirements for Web Services Quality Prediction to Support Web Services Interoperability in Electronic Commerce" Wan Nurhayati, WAN AB. RAHMAN and Dr Farid MEZIANE

8. "The Open Grid Services Architecture, Version 1.0" Foster, Argonne & U.Chicago, H. Kishimoto, Fujitsu, A. Savva, Fujitsu

9. "Web services: Benefits, challenges, and a unique, visual development solution" Written by Erin Cavanaugh Product Marketing Manager Altova

10. "Web Services White Paper" by Sonera Palza Ltd, June 121, 2002.

11. "The Implications and Impacts of Web Services to Electronic Commerce Research and Practices" Minder Chen, Andrew N. K. Chen, Benjamin B. M. Shao.

12. "A Meta-Model for QoS-Aware Service Compositions" Daniela Malfatti daniela.malfatti@studenti.unitn.it 19 dicembre 2007