Suitable QoS Parameters Survey for Standard Web Services & Web Applications to Understand their Cloud Deployability

S K V Jayakumar
Department of Computer Science
School of Engineering& Technology
Pondicherry University
Pondicherry, India.

Jayraj Singh
Department of Computer Science
School of Engineering& Technology
Pondicherry University
Pondicherry, India.

K Suresh Joseph
Department of Computer Science
School of Engineering& Technology
Pondicherry University
Pondicherry, India.

***Abstract*- The vital role of web plays in different application domains such as, distribution of information, business, education, industry, and entertainment, E-commerce, survey, online shopping, core banking and much more. In the recent years, in cloud environment there is huge demand for web services and Web applications portability or deployability because of the rapid and enormous requirement in the development of e-business solutions; cloud offers its services through using web services & web applications. These are chosen by any user on the basis of their properties which could be either functional or non-functional or both. In this paper, some standards, qualities & suitability of web services & web applications have been thoroughly surveyed and analyzed to understand whether they are cloud deployable or not. In precise, the required qualities and standards for a web services and web applications for their effective deployability are examined over cloud.**

Keywords- QoS, Web Services, Cloud computing, Web applications, Suitability, Standard & Specification, Security

I. INTRODUCTION

The web services and web applications are the software entities that have capabilities to define well- interface for performing specific task using internet. Some examples are like whether forecasting, news, entertainment or services altering the world business state, such as online shopping or booking services. Quality of services plays very important role in selecting the web services & application from UDDI registry which should be functionally matched with their services request. Traditional web services are lack of QoS description. So it is difficult to choose its offered services by only its functionality. These web services also are not suitable to deploy in the cloud environment [1]. Hence there is a need of effective mechanism that can retrieve or identify the most appropriate services. Now a days the cloud computing is very hot areas for research and playing an important role in web service environment to developing many distributed applications. There are many mechanisms are used, which finding out the effective web services on the basis of different quality parameters. By using these services, Cloud provides number of benefits to user using these services. For example, 1) Reduction of costs- instead of the online hosting of the applications, the cost for deploying the web services & web application in the cloud can be less due to lower hardware costs. 2) Universal access - cloud computing can allow employees which are remotely access the applications and can utilize the services of distributed application using internet. 3) Upgrades the software - in cloud environment, the service provider can easily update the software. 4) Choice of applications-in cloud computing, a cloud user get full flexibility to select the services according to their needs with a fast implementation time, for example pay only for those services whatever they using [2]. Web services fulfill these desired tasks. Hence effective web services is needed which can be discovered on the basis of some parameters. Here in this paper we identified some quality attributes, some special characteristics, standards & specification for web services and web application. By the help of these we can ensure high product quality services and can find the suitability of the web services for deploying in the cloud environment.

II. BEHAVIOR CHARACTERISTICS OF WEB SERVICES

In this section, we introduced the web services behavioral characteristics required for being part of the Cloud [3, 14].

A. XML-Based

XML is the platform independent framework, so it will support all genre of platform. Using XML the web services specify the transportation of data, and its representation. It also eliminates the binding of operating system, networking, or platform using XML, web services. So web services based applications are most powerful in the way of inter-operable application at their core level.

B. *Loosely Coupled*

In loosely coupled environments, Clients and server are not closely tied with another. A consumer is loosely tied for the use directly of the web services. Without compromising the consumer interaction or Client with web service, the web service interface can be change any time.

C. *Course-gained*

Course grained feature provides composite interface that allow to compose the applications and processes using services from different environments. By using course grained interfaces, control accessibility to the objects referenced by each service is achieved by a system of services. In Object-oriented technologies like java, using individual methods we expose the services.

D. *Ability to be synchronous or Asynchronous*

In web service environment the synchronicity means the binding of clients should be synchronous with the execution of web services. Client's responsibility is to block and wait for service to complete its remaining operation. That is he gets result when service has completed. But in asynchronous operations clients call to the services and can executes other function also [4].

E. *Supports Document exchange Files*

XML The advantage of using XML is not only for representing the data but we can also represent complex documents. For example when we represent a current address it will be simple while representing an entire book or RFQ, they can be complex. Web services support these documents for business integration.

F. *Supports of Remote Procedure Calls (RPCs)*

Web services allows for the clients to remotely access the objects, methods and invoke the operation by using XML-based protocol. By remote procedure call we have to expose the inputs and output parameters of a web services.

G. *Machine-to-machine interaction*

In cloud environment WS support to provide interoperability by machine to machine interaction. The web services describe an interface using WSDL, which is machine processable. By using machine to machine interaction; system can be used independently on different software and hardware platforms. It can also become programming language independent in which it is written [6].

H. *Interoperability*

The interoperability is the quality aspect of any system that shows the ability to work with other systems or product without doing any special effort. SOAP, WSDL and UDDI protocols play very important role to provide interoperability in web service environment. These protocols define the self- describing way to call and search procedures in a software application-regardless the platform dependency and location. Still web service interoperability problem suffers at the definition, discovery and request/response level [7].

I. *Platform and language-independence*

To be platform and language independent, a web Application & web services which are running on server must be installed and run on many different platforms and provide a common interface that is usable to define different language binding. Due to this web service technology is sometimes considered as a prevailing integration technology on Internet/Intranet [8, 9].

J. *Leverage the architecture of the WWW*

By leveraging the architecture of World Wide Web, Web services use an on demand platform for enterprise architecture, which allow the integration of different application rapidly. By doing this it will increase efficiencies and innovation, while at the same time it reduced costs also [10, 11].

III. QoS of Web Services and Web Application

From the functional & non-functional QoS, the web services usually have non-functional QoS. In cloud environment web services provides two types of quality aspects: technical and business aspects [5, 6, 8, 18].

A. *Technical Aspects*

1) *Availability*

Availability is the quality parameter of web service environment, where we check that web services and application are present or ready for the clients for immediate use.

2) *Accessibility*

It represents the degree in which it is capable of serving maximum service request.

3) *Integrity*

Integrity means how the web service maintains the correctness of the interaction. That is accuracy of the communication, It describes communication can only be accessed or modified by authentic parties.

4) *Performance*

By checking the performance in cloud environment in terms of throughput or latency, we can predict the quality of web services.

5) *Reliability*

Reliability is the quality parameter of web services that shows the degree of maintaining the services. Reliability provided by soap over HTTP protocol.

6) *Regulatory*

Regulatory of web services ensure conformance of rules, law, compliances with earlier established agreement and standard.

7) *Security*

Security of the web services is the quality aspect that provides confidentiality of information, integrity, authentication of the involved parties, non-repudiation ...etc. It also provides encryption and access control of messages [10].

B. *Business Aspects*

1) *Pricing*

Price is the ability to create profit by extracting the value of firm’s product to its consumer.

2) *Billing*

In a specific time the total cost for doing the business

3) *Warranty policies*

It is an assurance given by one party (seller) to the other party about some conditions and fact which will happen true.

IV. Quality Factors in ISO 9126

ISO/IEC 9126 consists of four parts for the Product quality. 1) Quality model. 2) Internal metrics. 3) External metrics. 4) Quality in use metrics. The quality factors and their sub characteristics are described by ISO 9126 for the web services shown by figure [7, 9].

V. General Security Aspects of Web Services

The security is most important concern in the web service environment. It is very critical to achieve. However, neither XML-RPC nor SOAP specifications make any authentication requirements or any explicit security [12, 46, 15]. The different security aspects are given as.

A. *Authorization*

It ensures that only authorized users or authentic user can access an application. It controls the level of accessibility that is read/write, accessibility to which data / resources.

B. *Integrity*

It verifies that data has not been changed by unauthorized third parties.

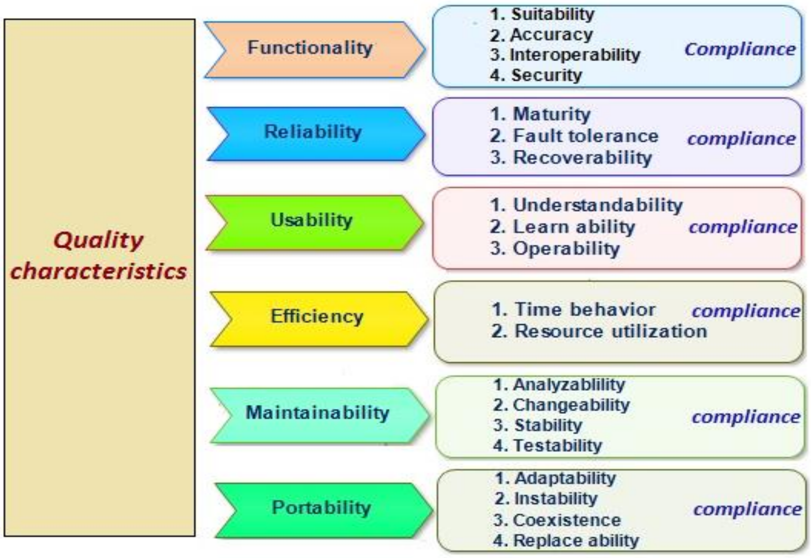

Figure 1: Representation of Quality Characteristics

C. *Authentication*

It ensures the identity of a client which connects to a web service, since, there is no clear strong authentication scheme is used in web service environment but still by using some protocols we provide authentication [12]. 1) HTTP supports some Basic authentication for HTML documents. 2) SOAP-Digital Signature (SOAP-DSIG) is used to provide public key cryptography for sign digitally the SOAP messages. By using this SOAP-DSIG validity of client and server is also checked.

D. *Non-repudiation /Auditing*

Non-repudiation means, a user should not deny the performed operations or initiating a transaction. The effective key for non-repudiation is auditing and logging. For example, in an e-business, by non-repudiation mechanism a consumer cannot deny, suppose if he ordered 10 copies of a particular book [17].

E. *Confidentiality*

It ensures to keep data secret for unauthorized parties like users, machines. In web service environment XML-RPC and SOAP protocol are used to ensure the confidentiality in the communication when a client sends a XML based request. These protocols work over HTTP which helps to provide encryption via Secure Sockets Layer (SSL).

VI. Bottlenecks in Performance of WS

Due to limitations of underlying message & transport protocol in the web service environment, the web service tends to create performance bottleneck. The HTTP protocol provides best delivery service as well as stateless data forwarding mechanism. Instead of this it creates some major problems like, 1) No bandwidth is available when amount of users and data over network increases as a result the packets frequently started to discard. 2) No guarantee of packet delivery to the destination.

VII. Standardization

The significant standardization authorities for web Services are 1) Internet Engineering Task Force (IETF), 2) W3C World Wide Web Consortium (W3C), 3) Organization for the Advancement of Structured Information Standards (OASIS) and 4) Web Services Interoperability Organization (WS-I) [17, 18].

A. *W3C standards for web services*

W3C approved the standard as ISO/IEC international standards for web service [12, 13].

1) *ISO/IEC 24824-2:2006 IT- Generic applications of ASN.1: Fast Web Services*

This specification is for ASN.1 SOAP messages which provide a fast web services. This specification also carries the semantics of W3C SOAP messages. The ASN.1 schema enables that WSDL content should be encapsulated efficiently by using some encoding technology in SOAP based envelope. Here changes in SOAP binding syntax are not needed [25].

2) ISO/IEC 25437:2006 IT-Telecommunications and information exchange between systems WS-Session - WSs for Application Session Services.

It specifies Application Session Services with SOAP protocol binding defined in ISO/IEC 22534 (ECMA-354). This Session Services allow creating and maintaining a relationship with Servers. This is also called application session [30].

3) ISO/IEC DIS 40210, IT – W3C SOAP Version 1.2 Part 1: Messaging Framework

SOAP protocol plays as a very light protocol in distributed environment for exchanging information. By using XML technologies it defines an extensible messaging framework to provide a message construct which can be exchanged over a variety of underlying protocols. This message framework has been designed to be programming model independent [31]. The SOAP messaging framework define by ISO/IEC 40210:2011 is consist of: 1) The SOAP processing model, 2) The SOAP Extensibility model, 3) The SOAP underlying protocol binding framework, 4) The SOAP message construct [31].

4) ISO/IEC DIS 40220, IT – W3CSOAP Version 1.2 Part 2: Adjuncts

ISO/IEC 42020:2011 describe a set of adjuncts for use with the Version 1.2 based SOAP messaging framework which is specified in ISO/IEC 42010:2011. That is depends on this ISO/IEC 42010:2011 [32].

5) ISO/IEC DIS 40230,IT – W3CSOAP Message Transmission Optimization Mechanism

This specification specifies the features for optimizing the transmission of base 64 encoded data and wire format of a SOAP message [33].

6) ISO/IEC DIS 40240, IT – W3C Web Services Addressing 1.0 – Core

The ISO/IEC 42040:2011 defines XML Infoset representation and a set of abstract properties of web services in order to provide end level addressing to support message transmission between processing nodes. These processing nodes may be firewalls and gateways, endpoint managers…etc., in a transport-neutral manner [34].

7) ISO/IEC DIS 40250,IT–W3C WS Addressing 1.0 – SOAP Binding

It defines the binding of the abstract properties like data processing, test model, information exchange, and cryptography to SOAP Messages these properties are defined in ISO/IEC 42040 [32].

8) ISO/IEC DIS 40260, IT – W3C Web Services Addressing 1.0 – Metadata

ISO/IEC 40260:2011 defines a WS-Policy that can be used to the support of WS-Addressing by a Web service. The inclusion of WSDL metadata in end level references as well as abstract properties by using WSDL is also defined by this specification ISO/IEC 40240 [31].

9) ISO/IEC DIS 40270, IT – W3C Web Services Policy 1.5 – Framework

This specification describes the policies that refer to domain –specific capabilities, characteristics and other requirements in a web application system.

10) ISO/IEC DIS 40280, IT – W3C Web Services Policy 1.5 – Attachment

ISO/IEC 40280:2011 defines two general mechanisms which describe how WSDL, UDDI are associated with their policies to which they apply.

B. *Security specific standards and recommendations by IETF*

1) *RFC1108 - U.S. Department of Defense Security Options for the Internet protocol*

This specification is used to specify the basic security option for the American department of defense. By using this RFC1108 the high-level description with the Internet Protocol for better extended security Option is specified [39].

2) *RFC2196 - Site Security Handbook*

This specification guides to develop the securities policies and procedures for the system on the internet. The idea behind this site security handbook is to provide an annotated version of the freely accessible RFC 2196 with the reference to the ISO/IEC 27001 for willing to implement security best practices [39].

3) *RFC 2222- Simple Authentication and security Layer*

RFC 2222 describes a mechanism for adding identities of users and security of data in support to connection-based protocols. RFC 2222 allows decoupling authentication mechanisms from application protocols. This Authentication mechanism can also support proxy authorization. The RFC 2222 also describes some commands for identifying and authentication of user with a server to protection of protocol interaction [12].

4) *RFC2323 - IETF Identification and Security Guidelines*

The RFC 2323 represent a guideline for identification and security protocols, the RFC 2323 provides some guidelines to attention to a web service sub-group within IETF: “facial hairius extremis”. By these guidelines the IETF conferences may run more efficiently.

5) *RFC2401 - Security Architecture for the Internet Protocol*

This specification provides a base architecture for IPsec complaint systems which helps to give the various security services for traffic at IP layer. For example it provides a mechanism to manage manual and automatic key exchange as well as some algorithms for authentication and encryption. Authentication header and encapsulating security payload is used by security protocols which is applicable in both IPv4 and IPv6 environment.

6) *RFC2411 - IP Security Document Roadmap*

The IP Sec protocol suit define in this specification allows high privacy and authentication services at IP layer. How encapsulated security payload protocol and Authentication header protocol are implemented is also described. That is, an explanation of including new Encryption Algorithm and Authentication Algorithm are described.

7) *RFC2504 - Users' Security Handbook*

This specification provides information which is needed for the users to keep their system secure in the network. The network and system administrator wish to use these specific guidelines to keep their communication private and to their system and network secure.

8) *RFC2828 - Internet Security Glossary*

This RFC 2828 provides explanations, abbreviations, and recommendations on the based on information system for use of security technology. The main intension is to improve the Comprehensibility of particularly Internet Standards documents (ISDs) and Internet security.

9) *RFC3365 - Strong Security Requirements for IETF Standard Protocols.*

There is no any central administrator for the internet network. So that it is managed directly by independent hosts and networks. To provide the security across the network some protocols as well as security parameters are needed. The RFC 3365 tells about the IETF consensus on different security requirements and protocols. The role of this specification is to ensure that security protocols should have features to give security for the application used in the communication network.

10) *RFC3414 - User-based Security Model (USM) for SMNP version 3*

This specification describes the architecture of SNMP protocol by using user based security model (USM). The RFC 3414 defines several elements for providing SNMP message level security. In this model it includes management information base (MIB) for managing and monitoring the configuration parameters.

11) RFC3631 - Security Mechanisms for the Internet

It does not specify an Internet standard of any kind. It only describes the information for the Internet community.

12) RFC3871 - Operational Security Requirements for Large Internet Service Provider (ISP) IP Network Infrastructure.

It defines a framework which is a collection of operational security requirements applicable to certain network topology contexts to internet service provider in a large infrastructure network.

13) RFC4033 - DNS Security Introduction and Requirements

It introduces the DNS Security Extensions protocol (DNSSEC). These extensions do not provide confidentiality. The DNS security extensions provide origin authentication, public key distribution and integrity protection for DNS data.

14) RFC4251 – The Architecture of Secure Shell (SSH) Protocol

RFC 4251 not only specifies the architecture of secure shell (SSH) protocol but it also helps to describe the notation and terminology used in the protocol .With the help of this protocol we achieve secure remote login and secure network services in an insecure network environment. This protocol also provides a mechanism that ensures the authentication of clients to server; confidentiality and integrity.

C. OASIS standards for different information formats

OASIS is an international nonprofit consortium that promotes the adaption of product of open standards for information society. The information format may in the form of Standard Generalized Markup Language (SGML), XML, and HTML [22-24].

1) OASIS Standard 1.0 for Web Services Security

OASIS Web Services Security Technical Committee (WSS-TC) use core SOAP Message Security 1.0 specification by creating additional token profiles including the Web Services Security. The Web Services Security specification set includes:

2) Web Services Security: SOAP Message Security V1.0

The enhancements in SOAP protocol message is described in this specification. It helps to provide message confidentiality and integrity. Some general-mechanism that helps to associate security tokens with message content is also described. It supports multiple security token formats [42].

3) Web Services Security: Username Token Profile V1.0

This is a SOAP message security specification which describes how to use the User name Token with the Web Services that is how a consumer of web service identifies the requestor by “username”. And using a password or secret keys how it authenticates the identity of producer in web service environment [23, 24].

4) Web Services Security: X.509 Token Profile V1.0

This specification describes how to use X.509 certificate for the SOAP message security in web services system. The main goal of using X.509 certificate is to check the validity of public key which used for authenticate the encrypted SOAP message. It also provides the binding between public key and some attribute like issuer name, serial number, validity interval [19].

5) Web Services Security: SAML Token Profile V1.0

This is a SOAP message security specification that defines extension of SOAP protocol which is implemented for message authentication and encryption. It defines a security Assertion markup language (SAML); an XML based standard data format language for exchanging the authentication and authorization of data between parties involve in the communication.

6) OASIS Web Services Distributed Management

This WSDM specification is used for managing the web service by exposing some managing functionality in a reusable way through two specifications. 1) Management using web services (MUWS) that describe a framework to define how to access the manageable interface. 2) Management of web service (MOWS) is used to define how a web service is used as a resource. It also allows some management tools with the help of WSDL interfaces that provides management events and metrics to be exposed queried.

VIII. WEB SERVICE SPECIFICATION

The WS specifications are referred to as “WS -*”. Here “WS-” is a prefix used to show the specification related with corresponding Web Services. The existing WS* standards in the web service environments are WS-Discovery, WS-Addressing, WS-Federation, WS-Security, WS-Policy and WS –Trust [18].

A. XML Specification

1) XML (eXtensible Markup Language)

XML is a markup language that defines rules for encoding documents which is readable to both machine and human. XML provides textual data format which support Unicode for the languages. By using XML we can emphasize the simplicity, generality and usability over the internet [20].

2) XML Namespaces

XML documents have different elements. Each element has its own element type name and also has different attributes. For giving unique names for elements and their attributes, we use XML namespace.

When different objects or elements have same names in XML documents, we resolve the ambiguity by XML namespaces or we can say that for avoiding the confliction between element names [30].

3) XML Schema

An XML schema is a brief description in terms of constraints of an XML document. It tells about structure of xml document. The XML schema is used to define how the elements and their attributes are appeared in the document. It also described their child elements and their numbers and order .This XML-based schema is more powerful so that it is an alternative to Document Type Definition language (DTD) [24, 34, 35].

4) Xpath

XPath is used as language in XML document for finding information. It contains a library of standard functions. This is an element in XSLT function. XSLT documents cannot be creating Without XPath knowledge. XPath includes over 100 built-in functions like time and date comparison, numeric values, string values, node and sequence manipulation, Boolean values , Qname manipulation etc.,

5) XQuery and Xpointer

By using XPath expressions ,XQuery and XPointer are both built. XQuery is an XML query language like SQL in database which is used to extract the XML data. The XML query contains the sequence of XML atomic values or fragments. After processing this, it also returns XML fragments or atomic values. XPointer is the extension of XPath which allow the hyperlinks to specific part in xml documents.

6) XML Information Set

It describes the abstract data set which provides consistent information in a well- formed xml document. This XML information set is also called “Infoset” and is created by different methods. There is no requirement for the validation of XML documents if we have XML information set.

7) Xinclude

It provides a mechanism to merge the XML documents. By this mechanism we incorporate the different content of XML documents and Non-XML documents.

B. Message Specification

1) SOAP (Simple Object Access Protocol)

SOAP protocol is a communication protocol used to exchange the structure information between applications in web service environment. The SOAP protocol is XML based so it provide platform as well as language independency. This protocol works over HTTP or SMTP for negotiation and transmission of the messages [19, 25].

2) SOAP-over-UDP

Many application protocols do not require the delivery guarantees, while others make use of multicast transmission. In order to allow Web services for supporting these patterns we need a way to map SOAP envelopes to user datagram. By using WS-Discovery the support is needed for web services, where the multicast forwarding and need for low connection overhead makes UDP a natural choice. It provides One-Way and Request-Response message patterns [28, 37].

3) SOAP Message Transmission Optimization Mechanism

In canonical lexical representation, there is no standard way to indicate data, so there is no guarantee of optimization to be preserved if multiple SOAP nodes involved. This MTOM mechanism provides an optimization of transmission of encoded data such as images, files, along with Web service request and wire format of SOAP messages. It also enables binding with SOAP protocols using XOP to provide optimized MIME multipart serialization of SOAP messages [22, 23].

4) WS-Notifications

It is a set of standards that provides the web services to interact using “notifications” or “Events” WS-notification provides interfaces with other standards like WS-addressing for high availability, workload Management and WS-reliable messaging for reliable transmission. By providing this interface it enables web services to use publish and subscribe the message pattern. The WS-notification approach allow to web services to expand the information to one another without having to have prior knowledge of each other Web Services [22].

5) *WS-Base Notification*

It is a standard business approach for notification or events. It enables the web service application to participate in the publish and subscribe messaging pattern. It defines a web services interfaces for two of the important roles in the notification pattern, namely the notification producer and notification consumer roles.

6) *WS-Topics*

WS-Topics specification describes how a notification producer application can associate a topic with the notification messages that it produces.

7) *WS-Brokered Notification*

WS-Brokered Notification is to standardize the exchanges messages that are involved in subscribing of a message broker and web services published. The overall requirements of WS-Notification are define in WS-Base notification.

8) *WS-Addressing*

This specification defines XML 1.0, XML namespaces element that identifies WS endpoints and also provides secure end level identification in messages. It is a transport-neutral mechanism that addresses the WS & messages. This Specification support message transmission in between processing nodes through Internet network. For example firewalls, gateways [31].

9) *WS-Eventing*

The WS-Eventing provides a protocol that helps to subscribe or accept subscriptions for event notification messages for web services. When events occur in the web service and web application then some web service often interested to receive messages. WS-Eventing provides a protocol for one web service called “subscriber” to register interest called “subscription” with another web service called an “event source” and receiving message called “event messages”. The subscriber has responsibility to manage the subscription with a WS, and how event message should be delivered. This subscription manager can delete and create the event subscriptions [44].

10) *WS-Enumeration*

This specification for enumeration is defined a simple SOAP-based protocol that provide a session that allow the data source called an enumeration context. By using this enumeration context consumer request for XML element information.

11) *WS-Make Connection*

This specification is used to define a protocol which allows messages to be transferred between WS-Make Connection-aware nodes. It uses transport-specific back-channel. In this specification SOAP binding is defined to support interoperability of web services. It can be implemented using different technologies.

IX. WEB SERVICE PROTOCOL STACK

The web services protocol stack has four main layers but still it is evolving, if we add some additional technologies, we may add some additional layers in the stack.

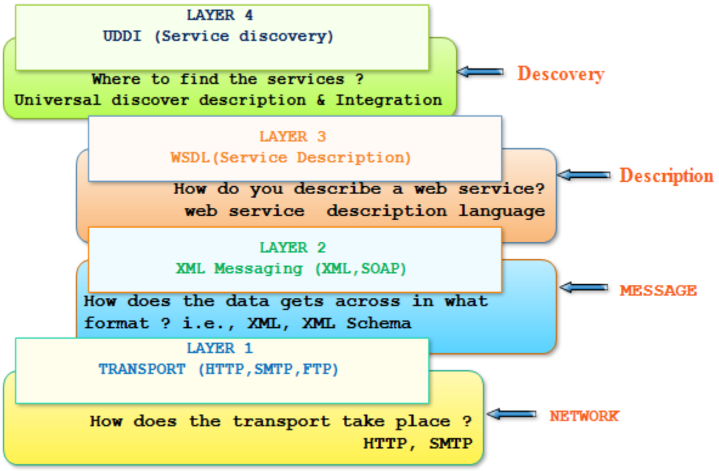
LAYER 4
UDDI (Service discovery)
Where to find the services ?
Universal discover description & Integration
Descovery
LAYER 3
WSDL(Service Description)
How do you describe a web service?
web service description language
Description
LAYER 2
XML Messaging (XML,SOAP)
How does the data gets across in what format ? i.e., XML, XML Schema
MESSAGE
LAYER 1
TRANSPORT (HTTP,SMTP,FTP)
How does the transport take place ?
HTTP, SMTP
NETWORK

Figure 2: Layered diagram for web service protocol stack

A. *Service transport*

This layer is responsible for "how does the transmission can be taking place". In this layer some transporting protocols such as HTTP, SMTP, FTP and some newer protocol for example Blocks extensible Exchange protocol (BEEP) are used these help to transfer the message between applications.

B. *XML messaging*

This layer describes "In what format the data should be exchange". That is transferring of data will be in XML format. This layer can allow XML-RPC and SOAP protocols for communication purpose. In this layer the messages are encoded in XML format before transferring so that it can be understood easily at another end. XML-RPC and SOAP are also included in this layer

C. *Service description*

This layer is mainly to describe the public interface for a specific web service to be communicating. That is how a web service is described. Currently Web Service Description Language (WSDL) is responsible to handle the service description.

D. *Service discovery*

This layer describes the registry in which web services resides. By using this common registry a client gets easy publishing and finding functionality. Currently, this registry task is handled by Universal Description Discovery and Integration (UDDI).

X. WS-Protocol Stack for Security Specifcation Analysis

The web services protocol stack has four main layers but is still evolving. Additional layers would be further added in the stack as and when new technologies are added. Some of the web services specifications have been analyzed in this section and shown here. This analysis gives the ideas and concept for developing progress, and also helps to find the problems which are not addressed in the existing web based systems. The major changes from previous system to new web services-based systems are also identified. Table 1 of this section summarizes the web services protocol security specifications.

Table I. WS-Protocol Security Specifications

Security Specification	**Reliable message Specification**	**WS Interoperability Specification**	**Privacy**
WS-Security	WS-Reliable message	Ws-I Basic Profile	P3P
XML Signature	WS-Reliable	Ws-I Basic security Profile	
XML Encryption	WS-RM Policy Assertion	Simple SOAP Binding Profile	
	Resource Specification		
WS-Secure Conversation	WS-resource Framework		
WS-Trust	Ws-Transfer		
XML-Security Policy	Ws-Fragment		
XML-Federation	Resource representation SOAP Header Block		
Service Protocol Specification for Business and message Communications			
Metadata Specification	**Business Process Specification**		**Draft Specification**
Ws-Policy	Ws-BPEL		Device profile for WS

JSON-WSP	Ws-CDL	EbXML
WS-Policy Assertions	Ws Choreography interface	
WS-Policy Attachment	Ws Choreography	
WS-Discovery	XML Process definition L(G)	
WS-Inspection	Transaction Specification	
WS-Metadata Exchange	Management specification	
Universal Description Discovery & integration		
WSDL-2.0 Core		
WSDL 2.0 SOAP Binding		

Table I is describing the different aspects of WS Protocol specification which are used on different layers of the WS protocol stack.

A. *Security Specifications [11, 16, 46]*

1) *WS-Security*

It defines the security protocols for web services. WSS provide an extension to SOAP protocol to apply security. This specification was published by OASIS. It is a member of the WS-* family of web service specifications. The security specification provides how integrity and confidentiality can be enforced on messages and allows the communication of various security token formats, such as Kerberos, SAML, and X.509. Its ultimate goal is to provide end to end security by focusing on the use of XML Signature and XML Encryption [36].

2) *XML Signature*

XML signatures can be used to sign data, typically XML documents, but we can also sign anything that is accessible via a URL. An XML signature is also used as detached signature by sign a resource outside its containing XML document. And as envelope signature by sign some part of its containing document. XML Signatures are also written as XML-Dsig, XMLDSig, and XML-Sig [45].

3) *XML Encryption*

XML Encryption provides secure exchange of structured data by end-to-end security for applications. This most popular XML technology provides security and structuring the data, therefore by this natural way, we handle complex requirements for security in data interchange applications.

4) *XML Key Management (XKMS)*

XKMS defines the protocols for distributing and registering public keys, it also helps in suitable use in conjunction with the standard for XML Signatures defined by W3C and IETF. XKMS comprises two parts - XML Key Registration Service Specification (X-KRSS) and XML Key Information Service Specification (X-KISS).

5) *WS-Secure Conversation*

This WS-Secure Conversation specification is created by IBM and others. The defined protocol under this specification is used for securing the sequences of messages (WS-Trust encrypts single messages with a token) /entire conversations [33].

6) *WS-Trust*

It defines the protocol that are responsible for establishing the trust relationship using tokens among web service end points like service provider, consumers [32].

7) *WS-Security Policy*

The security policy assertions are defined under this specification such as capabilities, requirements, and preferences with regard to security [31].

8) *WS-Federation*

It allows different security realms (different organizations) to share security information on identities. It identifies federation for web services;

9) *WS-federation Active /Passive requester Profile*

This WS federation specification is used to define how the security mechanisms like identity, authentication and authorization schemes work across trust zones. Mainly this specification deals with SOAP-enabled applications (called active requestor) that, how they make requests using these mechanisms.

While in the context of passive requester profile it describe how a web browsers (called passive requestor), make requests using these security mechanisms [46].

10) Web Services Security Kerberos Binding

This specification deals with the use of Kerberos tokens in web service security. Specifically, it describes how to add signatures and encryption to the SOAP messages by attaching encoded Kerberos tickets [38].

11) Web Single Sign-On Interoperability Profile

This specification is used to ensure that how the service determines the protocols supported by the client's identity. It specify the mechanism that allows us to interact with a service using either WS-Federation based Identity Providers or Liberty Identity Federation for an interoperability profile of single sign-on metadata exchange..

12) Web Single Sign-On Metadata Exchange Protocol

This specification describes the identity processing protocol suits supported by WS provider. It specifies how to increase the service's ability to successful and efficient communication with the provider and how a service can query to an identity provider for metadata.

13) Secure Assertion Markup language (SAML) XACML

SAML is an XML-based standard data format language that provides a framework for exchanging security information like authentication and authorization data between business parties.

B. Metadata Exchange Specification

1) WS-Policy

WS-Policy specification allows to web service consumers to specify their policy requirements and for web services to use XML to advertise their policies on security, QoS based etc. The flexible and extensible grammar is also provided for expressing the capabilities, general characteristics of Web Service entities and requirements [32].

2) JavaScript Object Notation Web-Service Protocol JSON-WSP

JSON-WS protocol is used for service description, requests and responses. JSON-WSP is very much inspired from JSON-RPC. The description format of JSON-RPC has the same purpose as WSDL has for SOAP or IDL for CORBA. The HTTP protocol is used to communicate between clients and a JSON-WSP server [31].

3) WS-Policy Assertions

A policy assertion is a machine readable metadata expression that identifies behaviors required for Web services capability and interactions. A policy is composed of multiple policy assertions.

4) WS-Policy Attachment

The two general purpose mechanisms are defined under this specification for associating the policies. These policies may be defined as part of existing metadata about the subject or defined independently and associated through an external binding to the subject [31].

5) WS-Discovery

It the process of finding a suitable web service for a given task. Under this specification a multicast discovery protocol locate the web services. The primary mode of discovery is a client searching for one or more target services [27].

6) WS-Inspection

This Web service specification is for "discovery documents" developed in a joint effort by Microsoft and IBM. The service description language is not defined in this specification. But It Provide a aggregating method for different types of service descriptions. XML format is used for listing references to existing service descriptions. A set of conventions are defined in the specification so that it is easy to locate WS-Inspection documents [40].

7) WS-Metadata Exchange

The Specification defines how the metadata can be embedded in end point references such as Policies, WSDL file from the server. And how the metadata associates with the web service endpoint for example WS transfer resources or HTTP resources. By using this specification, it can be advertised that how metadata associated with implicit features could be retrieve from a metadata resource [41].

8) *Universal Description Discovery and Integration*

UDDI is an extensible markup language based registry which is platform-independent, that helps to register and locate web service applications. The WSDL documents are access very easily by enquiring the SOAP messages. In the UDDI registry the different protocols binding and different message format of the WS which is required to interact with different web services are also listed [27, 47].

9) *WSDL 2.0 Core*

The core language which describes Web services based on an abstract model of what the service offers are defines in the specification. WSDL enables one to separate the description of the abstract functionality offered by a service from service description details such as "how" and "where" that functionality is offered.

10) *WSDL 2.0 SOAP Binding*

This binding is intended to ease the migration from WSDL 1.1 to WSDL 2.0 for implementers describing services that use SOAP 1.1 protocol. And, this binding allows users to continue using SOAP 1.1 protocol. This specification depends on WSDL 2.0 Core and WSDL 2.0 Adjuncts [28].

11) *Web Services Semantics (WSDL-S)*

WSDL-S is the extension version of WSDL standard that includes some semantic elements. By facilitating the improving discovery enabling the integration of legacy software, composition of services with a Web Services framework, we can improve the reusability of web services [27, 29].

C. *Business processes Specification*

1) *Web Services Business Process Execution Language (WS-BPEL)*

WS-BPEL is an OASIS standard. This is also an executable language that specifies the actions within business processes in web services environment. It is an OASIS standard. Processes involve in BPEL import and export the information by using WS interfaces exclusively.

2) *Web service choreography Description Language (WS-CDL).*

WS-CDL is an XML-based language which is responsible to peer to peer collaborations of participant’s web services. The participant’s web services are defined from their common and complementary observable behavior, global viewpoint; and by ordered message exchanges, we get a common business goal.

3) *Web Service Choreography Interface*

This is an XML-based interface description language. The flow of messages is described between the exchanging between the Web Services participating in choreographed interactions with other services. The dynamic interfaces of the Web Service participating in a given message are also specifying.

4) *WS-Choreography*

This W3C specification defines an XML- based business process modeling language which help to define collaboration protocols for web service participants, in which services act as peers , and interaction may be stateful or long-lived [21].

5) *XML Process Definition Language (XPDL)*

XPDL is a format standardized by the Workflow Management Coalition (WfMC) to interchange business process definitions between different workflow products, i.e. In between different management suits and different modeling tools. XML schema are described in XPDL for specifying the declarative part of business process / workflow.

D. Transaction Specifications

1) *WS-Business Activity*

This specification defines the protocols that have the ability to compensate action if an error occurs in heterogeneous web services environment. These protocols also provide the web services application to participate in loosely coupled business processes. For example, an application that sends an email cannot retrieve if email following a failure in the business task. However that application provides a handler that compensates at business level and sent another email which advertised the circumstances. However, the application can provide a business-level compensation handler that sends another email advising of the new circumstances.

2) *WS-Atomic Transaction*

This OSIS standard worked over all-or-nothing property for a group of services, that is a set of services and three protocols (completion, volatile two-phase commit and durable two-phase commit) are defined.

These protocols ensure registration, propagation, automatic activation, and atomic termination of Web Services.

3) *WS-Coordination*

An extensible framework is described by this protocol for providing the coordination of actions in distributed applications. These coordination protocols support a number of applications. This framework allows existing transaction processing, workflow, and other systems for coordination to operate in a heterogeneous environment or to hide their proprietary protocols [48].

4) *WS-CAF*

Web Services Composite Application Framework (WS-CAF) is an open framework which is also developed by OASIS. WS-CAF provides a generic and open framework for those applications which contains multiple services used together, sometimes referred as composite applications. Ease of implementation, interoperability, and easily usable are the main characteristics of WS-CAF.

5) *WS-Transaction*

This WS-Transaction activity specification is given by web service Interoperability Organization (WS-I Organization) which is an industry-wide effort to know and provide the standardizing how Web services are requested and delivered.

6) *WS-Context*

This specification is a part of the WS-CAF suite. The purpose of web context is to provide a reference a shared context which is the interactions between web services. The WS context provides the details of the application-specific execution of the web services. It is easily included in the header of SOAP message. Context may be passed by value, or by reference. WS context makes easy for end point implementation for a web service to access message context and security information [26].

7) *WS-TXM*

This is a very useful fault-tolerance technique, especially used when multiple remote resources are involved. Here in presence of failures atomic transactions guarantied consistency. These ACID properties in the atomic transaction specification ensure that consistency of state should be preserve in complex business applications despite concurrent accesses and failures.

E. Management Specification

1) *WS-management*

It is a SOAP-based protocol that manages the servers, devices, applications and various Web services. It also helps in accessing and exchanging the managing information across the IT infrastructure [43].

2) *WS-Management Catalog*

The web service management catalog is a list or a set of resource metadata documents available from a WS-Management services. It describes the default format of metadata used in WS-Management protocol [43].

3) *WS-Resource Transfer*

This specification is very important core component of a resource access protocol in the web service environment. The intended goal of this specification is to meet the several requirements for example, 1) To support a various encoding methods including both SOAP version 1.1&1.2 envelopes. 2) It defines standard techniques for accessing resources and controlling them using commands like "get", "put", "create", and "delete".

4) *Web Services Distributed Management*

The specification WSDM is used for managing and monitoring the status of other web services with the help of well define network protocol in the cloud that is WSDM-compliant.

5) *WS-Transfer*

It defines the transfer of XML-based from a server to another server.

F. Presentation Oriented Specification

1) Web Services for Remote Portlets

This specification allows to "plug and play" visual user facing web services with portals. It helps us to communicate with remote portlets. This WSRP specification also forms a repository of web services to consume the different applications.

G. WS-Reliable messaging

This specification defines a protocol which is used to transfer the reliable messages over distributed application instead of network or system failures. That is, it allow to the applications to send and receive messages with reliable and efficiently. A SOAP binding is defined in this specification to provide interoperable of web services.

H. WS-Reliability

This WS-Reliability specification provides a SOAP-based protocol which is responsible for exchanging the ordered SOAP messages with guaranteed transmission, the binding of SOAP protocol with HTTP is also define.

I. WS-RM Policy Assertion

WS-RM describes a specific domain assertion for providing reliable messaging. To configure reliable messaging which is defined in WS-policy framework, we need to group all its policy assertion together such as add its child element. The WS* specifications are composed with each other to provide a web service environment. We also used XML, SOAP and WSDL extensibility models in this scenario.

J. Resource Specifications

A resource specification defines the methods and numbers for choosing resources as well as the amount of effort required for a given service within a group. Each service has exactly one resource specification but many services may share the same resource specification.

K. Web Services Resource Framework

The WSRF define the relationship between web services and stateful resources. It is a set of five technical specifications that describes WS- resource in terms of web service message exchange. This specification allows programmer to declare and implement the relationship between one or more stateful resources and Web services.

L. WS-Resource

The WS-resources keep the state information of the web service. Each resource has unique key by which we can instruct the resource of a particular web service to be use during the interaction with another WS.

1) WS-Base Faults

This specification provides a standard mechanism against faults in web service transaction. That is it reports the faults when any error introduce at the time of WS-Service invocation.

2) WS-Service Group

The WS-Service group is a form of heterogeneous collection of web services or web resources. The members comes under this group are specified using components called entries. A web service entries is a WS-resource. This WS-group also defines how exactly the operation should be perform over this group such as 'add new service, remove service, find the service' in the group.

3) WS-Resource Properties

This specification describes a message exchange standard which provides to clients requester to query and updates the implied resource properties. A resource may have zero or more properties for example -Filename, Size.

4) WS-Resource Lifetime

Resources have non-trivial lifecycles. This life cycle can be destroyed and can be created anytime. This is the time period between creation and destruction. To manage the life cycle, The WS-resource lifetime provides some mechanisms.

M. WS-Transfer

The mechanism for acquiring XML-based representations of entities using the Web service infrastructure is defined under this specification. It defines two types of entities.1) Resources, these entities are addressable by an endpoint reference that provides an XML representation. 2) Resources factories, these are web services that can create a new resource from an XML representation.

N. WS-Fragment

This specification is used to enable resource without need to include the entire XML representation in a message exchange. WS-fragment specification extends the WS-transfer specification to enable clients to retrieve and manipulate parts or fragments of a WS-Transfer.

O. Resource Representation SOAP Header Block

The specification specifies the serialization and semantics of a SOAP header block for carrying resource representations in SOAP messages.

P. Web Services Interoperability Specification

This WS-I specifications provides additional information to improve interoperability between vendor implementations [15].

1) WS-I Basic Profile

This specification provides the guidance for core web services specifications such as SOAP, WSDL and UDDI. This profile uses WSDL to enable description of services as sets of endpoints operating on message.

2) WS-I Basic Security Profile

It defines the WS-I Basic Security Profile 1.0 based on a set of non-proprietary along with clarifications and amendments to those specifications which promote interoperability.

3) Simple Soap Binding Profile

This profile defines the way WSDL documents are to bind operations to a specific transport protocol SOAP.

Q. Draft Specification

This specification describes the Schemas and APIs which is necessary to facilitate interoperability between provisioning systems in a consistent manner using Web services. Some other specifications are:

1) Devices Profile for web services (DPWS)

The DPWS define a minimal set of implementation constraints that enable secure Web Service messaging, description, discovery and eventing on resource-constrained devices.

2) EbXML

Electronic Business XML (EbXML) is a Global Standard for electronic business. This specification provides the ability to conduct business over internet. It gives businesses of any size. EbXML relies on some of Internet's existing standards like, HTTP, SMTP, FTP, TCP/IP, MIME, UML, and XML. This is very easy to implemented and deployed virtually on any computing platform.

R. Privacy

1) Platform for Privacy Preferences Project (P3P)

The P3P protocol enables Websites to express their privacy practices in a standard format. due to this users have more control on their personal information at browsing time. That is users need not read the privacy policies at every site when they visit or browse.

XI. CONCLUSION

From the discussions of this paper, it is clearly visible that the web services and web applications have a lot of quality parameters, standards & specifications. It is also understood from this detailed examination that there have been no guidelines proposed for the acceptable standard or values for the quality attributes of the web services and web applications. Such proposal on the quality attributes would help to find the best of web services and web applications for their suitability or deploy ability in the cloud. The functional and non-functional QoS parameters of web services & web applications with their standards and specifications have been discussed in details which further needs an in-depth analysis. This would facilitate the researchers to easily find the suitability of web services and web applications on the basis of such analysis on quality attribute for engaging them in the cloud.

ACKNOWLEDGEMENT

This work has been funded by the University Grants Commission (UGC) of India. The authors would like to thank the sponsors for their support, under the project titled “Web Services Suitability Assessment for Cloud Computing”, File No – 41-627/2012.

REFERENCES

[1] E. Preeti, J. Dustin, Rashka, and D. McDiarmid, “Quality Web Systems: Performance, Security, and Usability,” Addison- Wesley, Reading, Mass, 2001.

[2] Jain, Anurag Punde, “A Survey- use of Software Quality Attributes for Web Based Software Applications,” The International Journal of Engineering Sciences & Research Technology, 2010.

[3] M.S. Nanda Kishore, S.K.V. Jayakumar1, G. Satya Reddy, P. Dhavachelvan, D. Chandramohan, N.P. Soumya Reddy, “Web Service Suitability Assessment for Cloud Computing,” Trends in Networks and communication-Springer, 2011.

[4] P. Jithin, S.K.V. Jayakumar, “Performance Comperison of web service in Iaas cloud and standard deployable model, ” International journal of computer trends and technology, 4, 2013.

[5] G. Breiter, M. Behrendt, “ Life cycle and characteristics of services in the world of cloud computing,” IBM Journal of Research and Development, Internet and Enterprise Scale Data Centers, 53(4), 2009.

[6] L.L. Constantine and L.A.D. Lockwood, “Software for Use: A Practical Guide to the Models and Methods of Usage Centered Design,” ACM Press, New York, 2000.

[7] Scharl, “Evolutionary Web Development,” Springer- Verlag, Berlin,2000.

[8] Luis Olsina, Gustavo Rossi, “Measuring Web Application Quality with WebQEM ,” IEEE Trans,2002.

[9] Luis Olsina, Gustavo Rossi, “A Quantitative Method for Quality Evaluation of Web Sites and Application,” IEEE Trans, 2002.

[10] S. Murugesan and Y. Deshpande, “Web Engineering: A New Discipline for Development of Web-Based Systems,” Web Engineering, Lecture Notes in Computer Science 2016, Springer-Verlag, Berlin, pp. 3–13, 2001.

[11] Pierluigi Plebani, “Quality of Web services,” Summer School on Service Oriented Architectures 2006.

[12] Jeff Offutt, “Quality Attributes of Web Software Applications,” IEEE Software 2002.

[13] www.ibm.com/developerworks/webservices/tutorials/ws-understand-web-services1.

[14] W3C Standards Approved as, “ ISO/IEC International Standards documents,” http://www.w3.org/2011/07/wspas-pr.html.

[15] David Booth, Hugo Haas, Francis McCabe, Eric Newcomer, MichaelChampion, Chris Ferris, David Orchard, “Web Services Architecture,” Editors of World Wide Web Consortium, 2004.

[16] http://www.w3.org/TR/2004/NOTE-ws-arch-20040211.

[17] R. Chinnici, J.J. Moreau, A. Ryman, S. Weerawarana, “WSDL Version 2.0 Part 1: Core Language,” Editors of World Wide Web Consortium, http://www.w3.org/TR/2007/REC-wsdl20-20070626, 2007.

[18] http://www.w3.org/TR/wsdl20.

[19] ISO/IEC 9126-1:2001, “Software engineering -Product quality - Part 1: Quality model”.

[20] J. D. Meier, Alex Mackman, Michael Dunner, Srinath Vasireddy, Ray Escamilla, “Improving Web Application Security: Threats and Countermeasures,” Microsoft patterns & practices published 2003 http://msdn.microsoft.com/enus/library/aa302370.aspx.

[21] N. Mendelsohn, M. Nottingham, and H. Ruellan, “SOAP MTOM,” Editors of World Wide Web Consortium, W3C Recommendation, 2005.

[22] http://www.w3.org/TR/2005/REC-soap12-mtom-20050125/.

[23] http://www.w3.org/TR/soap12-mtom/.

[24] T. Bray, J. Paoli, C.M. Sperberg-McQueen, E. Maler, and F. Yergeau, “Extensible Markup Language (XML) 1.0 (Fourth Edition)”, Editors of World Wide Web Consortium, 1998.

[25] http://www.w3.org/TR/2006/REC-xml-20060816.

[26] http://www.w3.org/TR/REC-xml.

[27] Semantic Business Processes Management Working Group .http://www.sbpm.org.

[28] Oasis Standard, E. Maler, P.Mishra, and R. Philpott (Editors), “Bindings and Profiles for the OASIS Security Assertion Markup Language (SAML) V1.1,” 2003.

[29] Oasis Standard, A. Nadalin, C.Kaler, P. Hallem-Baker, R.Monzillo (Editors), “Web Services Security: SOAP Message Security 1.0 (WSSecurity)”.

[30] Web Services Security: SOAP Message Security 1.0 (WS Security 2004) 2013.

[31] W3C Note, Simple Object Access Protocol (SOAP) 1.1, Don Box, David Ehnebuske, Gopal Kakivaya, Andrew Layman, Noah Mendelsohn, Henrik Nielsen, Satish Thatte, Dave Winer, W3C Note, 8, (http://www.w3.org/TR/2000/NOTE-SOAP-20000508), 2000.

[32] M. Little, E. Newcomer, G. Pavlik, (Eds) Web Services Context Specification Version 1.0 OASIS, 2007 http://docs.oasis-open.org/ws-caf/ws-context/v1.0/wsctx.htmlretrieved 2008.

[33] Erik Christensen, Francisco Curbera, Greg Meredith, Sanjiva Weerawarana, W3C Note, Web Services Descriptive Language 1.115, (See http://www.w3.org/TR/2001/NOTE-wsdl-20010315), 2001

[34] Keith Ballinger, David Ehnebuske, Martin Gudgin,Mark Nottingham and Prasad Yendluri, Editors. “The Web Services-Interoperability Organization,” Final Material 2004.

[35] R. Akkiraju, J. Farrell, J.A Miller, M. Nagarajan, M.T Schmidt, A. Sheth, K. Verma, “Web Service Semantics - WSDL-S,” Technical Note,Version 1.0, 2005.

[36] http://lsdis.cs.uga.edu/Projects/METEOR-S/WSDL-S.

[37] http://www.alphaworks.ibm.com/g/g.nsf/img/semanticsdocs/$file/wssemantic_annotation.pdf.

[38] T. Bray, et al, “Namespaces in XML 1.1,” 2004.

[39] (See http://www.w3.org/TR/2004/REC-xml-names11-20040204/).

[40] D. Box, et al, “Web Services Addressing (WS-Addressing),” http://www.w3.org/Submission/2004/SUBMws-addressing-20040810/).

[41] S. Anderson, et al, “Web Services Trust Language (WS-Trust),” (See http://schemas.xmlsoap.org/ws/2005/02/trust).

[42] Anderson, et al, “Web Services Secure Conversation Language (WS-Secure Conversation),” (See http://schemas.xmlsoap.org/ws/2005/02/sc).

[43] H. Thompson, et al, “XML Schema Part 1: Structures,”(See http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/), 2001.

[44] P Biron, et al, “XML Schema Part 2: Data types,” (See http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/), 2001.

[45] A. Nadalin, et al, “Web Services Security: SOAP Message Security,” http://docs.oasisopen.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf.

[46] H. Combs, et al, "SOAP-over-UDP," September 2004. http://schemas.xmlsoap.org/ws/2004/09/soap-over-udp.

[47] S. Bellovin, and M. Merritt, "Limitations of the Kerberos Authentication System," Computer Communications Review, October 1990.

[48] S. Kent, and R. Atkinson, “Security Architecture for the Internet Protocol," RFC 2401,1998.

[49] K. Ballinger, et al, “WS-I Basic Profile Version 1.1,” 2006.

[50] (See http://www.ws-i.org/Profiles/BasicProfile-1.1.html).

[51] K. Ballinger, et al, "Web Services Metadata Exchange (WS-Metadata Exchange),” 2006.

[52] K. Ballinger, et al, “WS-I Basic Profile Version 1.1,”April 2006. http://www.ws-i.org/Profiles/BasicProfile-1.1.html.

[53] R. McCollum, et al., http://msdn.microsoft.com/library/en-us/dnglobspec/html/ws-management.pdf.

[54] D. Box et al, “Web Services Eventing (WS-Eventing),” 2004.

[55] D. Eastlake, J. Reagle, and D. Solo, Editors, “The Internet Society & World Wide Web Consortium,” 2002.

[56] http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/.

[57] J. Schwarz, B. Hartman, A. Nadalin, C. Kaler, M. Davis, M. Hirsch, & K.S. Morrison, Security Challenges, Threats and Countermeasures Version 1.0, WS-I, Microsoft, IBM, Oracle, Data Power, Sarvega, Nokia Corporation,Layer7,2005.

[58] http://www.ws-i.org/Profiles/BasicSecurity/SecurityChallenges-1.0.pdf.

[59] R Rajmohan, N. Padmapriya and S.K.V Jayakumar, “Article: A Survey on Problems in Distributed UDDI. International”, Journal of Computer Applications 36(3), 2011.

[60] OASIS Standard, Web Services Coordination (WS-Coordination) 1.2, http://docs.oasis-open.org/wscoor/wstx-wscoor-1.2-spec-os.doc, 2009.