Web Service Composition: A Survey on the Various Methods used for Web Service Composition

Libin Thomas
Department of Computer Science
Christ University
Bangalore, India

Ashok Immanuel
Department of Computer Science
Christ University
Bangalore, India

Abstract: This paper is a survey about the constantly evolving web services, their composition methods and the various problems and challenges that they faced. The paper aims to provide an overview of numerous Web Service technologies like WSDL, UDDI, SOAP etc. The paper also provides insights about composition/processes like WS-CDL (Web Services Choreography Description Language), BPEL etc. The intention of this survey is to help the reader to understand about the different problems that were faced by the different composition methods while also providing information regarding the various aspects of Web Services.

Keywords: Web Services, SOA, WSDL, BPEL4WS, Web Service Composition, Orchestration, RESTful services, Composition Methods.

INTRODUCTION

Web Service is a virtual interface that defines a collection of operations that is made available all over the internet and a standard XML messaging system is used for the same. Web Service Composition is a method of combining these web services so that they perform website specific or requirement specific actions. These Web Service Composition methods face a lot of problems. The solutions to these composition methods were researched upon by various scholars and these research ideologies based on problems that are part of the automatic web composition methods have been surveyed upon in this paper.

This paper will cover Web Services and related concepts, Web Service composition life cycle, Web Service composition methods, Web Service composition problems that was brought into picture in the last two decades.

WEB SERVICES

Web Services are a group of open protocols and standards that are perused in sharing information amongst applications or systems that are accessed by people in different parts of the world. Web Services can be used to exchange data over networks for software applications and could be reproduced in a variety of languages and be run on numerous platforms. Web Services are described to be self-contained, distributed, dynamic and modularized software or web solutions that can be published, described, identified based on location or called over the network to create products that satisfy certain user needs, supply chains and processes. These applications are local, web-based or distributed. HTML, XML, Java, HTTP, TCP/IP are some of the open standards upon which Web Services are built on. On the basis of the type of services given and the wants of the users, Web Services can be analyzed beneath the framework of engineering of software components. Certain characteristics that a Web Service is expected to have are a description about the Web Service that permits other interacting Web Services to admit its range of operations and how it can be used, the permission for other Web Services to

access this service at all times or whenever required, and to be invoked easily. The above attributes and the perusal of XML based technologies actively encourage it to be used again and the interoperability of these Web Services. As of today, we have two types of Web Services that are available:

- SOAP – based Web Services which is based on SOAP and WSDL which stand for Simple Object Access Protocol and Web Service Description Language, respectively.

- RESTful web applications or services that are based on the REST based architectural principles.

Web Services should rather be called "Services of Internet" as it would be more appropriate than using the term "Web Services". Web refers to Hyper Text Transfer Protocol (HTTP) and the World Wide Web. The internet basically refers to a larger network of computers that make use of a multiple number of protocols. A web service doesn't necessarily have to use HTTP to pass a message. Instead it could use any of the myriad protocols available for sending a message [1].

A. SOAP BASED WEB SERVICES

SOAP (Simple Object Access Protocol) is a messaging standard that allows communication using HTTP and XML, for processes which execute on essentially different operating systems (such as Windows and Linux)

SOAP is stateless. It is a one-way message passing or exchanging channel between SOAP nodes, that is, between two communicators [1]. SOAP could be made use of, to establish intricate communications methods like request/response etc., by merging single sided exchanges that has aspects given by the transport protocol that lies beneath. (Don Box et al, 2000).

SOAP is brought up from testing systems such as the HTTP and XML for text mark-up and therefore, it is a lightweight

standard which is not dependent on the type of operating system, movement and is not dependent on the platform. [1]

SOAP consists of three parts:

- Envelope Construct: It defines an overall framework for expressing what is in a message. It specifies if it is optional or mandatory and also who should deal with it.

- Encoding Rules: It defines a mechanism for serialization which can be used to exchange application-defined datatypes created instances.

- RPC Representation: It defines a convention that can be used to represent remote procedure calls and responses.(Don Box et al,2000)

A SOAP message contains the following:
- The Envelope is the top element of the XML document representing the message.

- The Header can be considered as a generic mechanism that can be used for adding features to a SOAP message, in a decentralized manner without any sort of agreement made prior, between communicating parties. SOAP specifies if it is optional or mandatory and also who should deal with it.

- The Body contains information which is mandatory and is intended for the final recipient of the message. The Fault Element used for reporting errors is defined by SOAP.(Don Box,2000)

There are two types of SOAP requests. The first is the Remote Procedure Call (RPC). This is synchronous method. A message is propagated by the sender and he usually expects a reply or a message which documents errors from the receiver, for which he waits. Document request is yet another category of a SOAP request. A full XML document is passed within a SOAP message to and from the client.

B. RPC

SOAP-RPC is implemented using Remote Procedure Call (RPC). Using RPC, programs or functions that reside on a system that is located in another location can be accessed as we would a function that resides locally. All of the marshalling and un marshalling of data is controlled using SOAP and XML. RPC-style web services are known to be coupled tightly and are interface-driven. By sending parameters and receiving return values, clients invoke the web services. [1]

DOCUMENT

The client and/or server passes as the body of the information or message in the place of parameters, an XML document with a document style message. Document style messaging has other winning features over a remote procedure call. A contract between an application and a service can be broken by

an RPC interface, with any changes. The rules for document messaging are not as rigid as with RPC and numerous updates can be applied without destroying the requesting application, to the XML schema. This is because of the fact that an XML document in place of an expected structured return. [1]

C. B2B SERVICES

BPL (Business Process Languages) directly address behavioral aspects; i.e., not only which operations are made available but what is their effect on processes. Business process is a long running state where the state should be persistent and bursty i.e, sleeps most of the time but responds to triggered events.

D. B2C SERVICES

Business to Consumer Web Services. Using B2C services, there is a lot more communication with the end users of the services. Using SOAP, we can provide user (consumers) access to specific data by creating specialized functions.

Advantages of SOAP
- Working environments can be heterogeneous
- Based on XML
- Not dependent on platform.
- Not the dependent on the type of transportation.
- It makes use of PTP
- The ability to operate with other devices.
- Just In Time Discovery
- It is robust.

Disadvantages of SOAP
- Packet Sizes
- Implementation Issues
- Security Issues
- Versioning Issues
- Message Path
- Latency
- Ontology
- Statelessness

E. RESTFul SERVICES

REpresentational State Transfer or RESTful services, do not require as much bandwidth as SOAP. The architecture of REST is based on the standards of the web and it uses HTTP for message passing and communicating. RESTs working is dependent on resources where it treats every part of the system as a resource and each of these resources can be accessed by an interface which is common for all, by making use of the HTTP standards. Roy Fielding coined the term 'REST' in 2000.
Within a REST architecture, authorization to access the resources is given by a REST Server while the REST client gains access and showcases the utilities. URIs or global IDs are used to identify each resource. JSON and XML some of the resources Text, JSON and XML are some of the representations used by REST to substitute a resource with

JSON being the most popular method of representation these days.

Web Services are a group of standards and rules (protocols) which are used for communicational purposes by an application or between machines. Web Services can be used by applications which are developed using various languages and are running on numerous platforms to pass between themselves, data over a network, just like how the Internet works, in largely similar IPC on one system. Open standards enable the processes to operate alongside other processes of various programming language or platform (e.g., between C and Java, or Macintosh and Windows applications)

RESTful web services are those services that are based on the principle and architecture of REST. HTTP is commonly used by these services to include the basics of REST architecture. A RESTful web service usually defines a URI. It gives the stand in for resources such as JSON and set of HTTP Methods.
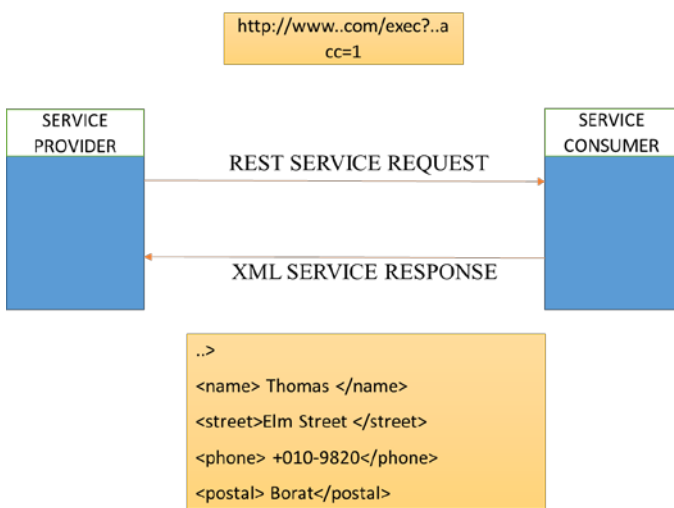

Fig1. RESTFul SERVICES

F. SERVICE DESCRIPTION

Web Services Description Language (WSDL) is a service description language which is maintained by the W3C.One of the key issue is manual composition of Web services. Because of the dynamic behavior and flexibility of the Web manual composition of Web services is complex and susceptible to errors.

G. AUTOMATIC WEB SERVICE COMPOSITION SOLUTIONS

Proposal By Kim and Gil's

Kim and Gil [Kim and Gil 2004] explain that as the composition technique is a complex application because users interaction is much needed. Thus they developed interactive tools for composing Web services which had the clients or the users defining a high-level or partial/incomplete description of the desired composition of the various available service applications. This system was capable of assisting the users by providing analytical and intellectual suggestions based on

usage. Service and domain ontologies supported these suggestions. This gives linear compositions alone (without any cycles and conditions).

SYNTHY

Synthy was proposed by Agarwal et al [Agarwal et al. 2005] that would solve the problems cause by automatic Web Composition. According to the authors of this proposed system, it was based on a two-staged approach that addressed the information modelling areas of the Web Services, providing the necessary support required for contextual information encountered during the composition itself. This system made use of efficient decoupling of the functional and non-functional requirements, and it lead to a better scalability and failure/error handling mechanism. Logic composition formed the first stage here, which generated workflows that were abstract in nature, using matchmaking techniques and planning. Physical composition was the second stage that generated the workflows that were executable which were used as instances of the services. Optimization techniques were used to choose the best instance that were associated to the service types. OWL-S has an extension that supported service types and it was made use of by this approach, thus permitting to collaborate with a large collection of Web Services and also with the two-staged composition [7].

WEB SERVICE COMPOSITION LIFE CYCLE

Wrapping of native services: This step in the life cycle makes sure that a service is or can be invoked by other applications or machines, no matter what the data model that lies beneath it or the message format or even the protocol that is used for interaction.

Service advertisement and/or discovery: This step generated the service descriptions and publishes the same in registries for discovery later on.

Assembling of composite services: Services are identified to realize the given composition method. This specifies their interactions at a high abstraction level and deriving descriptions that are external and service level agreements for the composite services that result.

Executing composite services: sanctioning composite administration determinations concerning execution models fulfilling certain functional requirements, e.g. execution value, effectiveness, accessibility and different QoS measurements.

Monitoring services :regulating composite administration executions (e.g., logging administration summons, state changes, and message trades) keeping in mind the end goal to distinguish contract infringement, measure execution and anticipate exemptions.

Evolving services: adapting composite services to accommodate organizational changes, to take advantage of new technological opportunities or to take into account feedback from monitoring.

WEB SERVICE COMPOSITION

WSDL is an XML format. WSDL describes Web Services and the syntactic front end of the Web Services which is used for interaction, that cannot be used to merge Web Services with the help of automatic methods alone.Web Service Description

Language-Semantic (WSDL-S), Web Service Modelling Ontology(WSMO),Ontology Web Language – Service(OWL-S), and Semantic Annotations for Web Service Description Language(SAWSDL) which are all Semantic protocols were therefore, innovated to merge or compose web services automatically. Also, information about web services is exposed by a virtual registry called UDDI [4]

H. ONTOLOGY

Ontology is a shared conceptualization. A common understanding of a particular domain is provided by the ontologies. It also provides a well-founded constructs set to help building meaningful knowledge of higher levels that helps in specifying the semantics of terminology systems. Ontologies provides a platform for enriched languages that can be used in creating much more complex constraints on resource types and their attributes in a particular domain. Ontologies are usually expressed using logical language, such that understandable and meaningful information can be gathered from amongst relations, attributes and the classes.

I. SEMANTIC WEB ARCHITECTURE

Semantic Web Architecture defines a group of technologies, software and protocols which helps in keeping the aim of the web which is identified along with the definition from the fundamental building blocks of an architecture. Sematic Web Architecture gives the description at processing level for the Web Services which have models of preconditions and post conditions in addition to functional information that is to be seen and understood as being the growth of the domain in a logical manner. The semantic web architecture consists of a certain set of protocols that are put into a defined layout. [8][9] Devices that can be used for interpreting, merging and for making use of data on the Web are all enabled by the semantic web. A machine-understandable description of resources forms the basis of the Semantic Web. It is used for appending semantics to Web Service. It is built on the basis of logic and knowledge that helps the machines in finding the perfect type of data [10] and to introduce the framework for semantic description of web services and the connected features, so that all of the information and the related services can be understood. [11]

Dynamic identification and the discovery of services, merging or composition of services, calling of Web Services and the automatic handling of all the services instead of the clients are all performed by the software components that Semantic web services. Web services and the semantic web are both termed as being synergistic. What it basically means is, although services give the required software for automatic usage of information, the Semantic Web converts the web into a storage for machine readable data. The idea of Semantic Web Service has thus been established. [12] [13]. UDDI, WSDL, SOAP, WSDL-S, OWL – S and other such languages proclaim the standards for service identification or discovery, description and communication. Two groups are specified based on the description of Web Services:

Syntactic Description: Syntactic Description says that the Web Service Description Language is used to define the interfaces

of Web Services and that forms its main goal. Meaningful descriptions of Web Services cannot be created using services at the syntax level as they are not enough[14]. Different constraint specifications, management statements, various class for services, SLAs, other contracts and protocols between Web Services are not supported by WSDL. Semantic Web Service (SWS) was established precisely due to this. [15][16].

J. SEMANTIC DESCRIPTION

A semantic framework which is the amalgamation of ontologies, provides the Service description. Semantic frameworks of the same type are also used in describing software artefacts. An example for a programming language that enables the capability-driven description of services is OWL-S. WSDL-S, WSMO, SAWSDL etc. are certain examples for semantic description languages.

K. QUALITY OF SERVICES (QOS)

Quality of Service (QoS) talks about the non-functional requirements of service such as response times, rate, availability etc. Measurable and non-measurable are the two categories that Quality of Service attributes are separated into. Measurable includes latency, throughput etc. while Quality of Service includes reputation and security etc. During the important step of deciding which services to put into the schema of composition, making use Quality of Service aspects is of importance. [16]

L. WEB SERVICE COMPOSITION METHODS

What exactly do we mean by Web Service Composition? As explained earlier in the paper, Web Service composition involves the combining or merging of numerous services that already exist to create a very complicated and usable service. It is rather interesting area of research and offers complex issues with respect to the composition of services. Complex problems can be addressed and given better solutions using existing, much simpler web service that cooperate with each other. There a number of potential issues with respect the field of web service composition techniques [17].
One of the rather interesting goals of Web Service composition is to recycle services and in merging or rather, "composing" into a process. Services like these allow the user to manually announce the composition of numerous programs to perform a complex task. This is on a much higher level considering that the whole process is much beyond the ability of a human being to deal with the entire process manually. Although attempts are made to make web service simple, it is still a very complicated task and usually, this complication of the process is from [2]

- The sheer amount of the available web services on the web. Web Services for particular problems are increasing dramatically over the years and this has led to a huge repository of web services that makes it all the more difficult for searching for them online.
- Web services can be made on the go. That is, web services can be made and updated whenever required

by the developer and therefore, the composition systems need to identify updates dynamically.

- Different organizations can create different web services that address the same problems, but with a different concept model that helps to describe the web services.

COMPARISONS

M.   SOAP BASED WEB SERVICES VS RESTFUL WEB SERVICES

Two methods which are mainly used for developing Web Services do exist and we have discussed about both types in this paper. They are the traditionally inclined SOAP-based Web Services and RESTful Web Services [6]. Web Services which is based on SOAP (WS/Web Services) rely upon three standard initiatives, i.e. SOAP, Web Services Description Language (WSDL) and the Universal Description, Discovery and Integration (UDDI). SOAP calls are responsible for implementing service registration, service discovery and service invocation which is done with the help of UDDI. SOAP based services are independent of protocols and resources. A lot of computational resources are required by them, especially when handling messages that are based of the SOAP protocols. Complex enterprise applications are all integrated using SOAP based web services. In stark contrast to SOAP based web services, RESTful services make use of the REST model which stands for REpresentational State Transfer. This architectural style was brought into picture to build large-scale distributed hypermedia applications or methods [Roy Thomas Fielding, Architectural styles and the design of network-based software architectures, PhD thesis, 2000]. URIs (Uniform Resource Identifier) are used to identify the various RESTful services. They offer a global space for addressing the resources and services discovery. A uniform interface of the RESTful services ensure that a fixed set of operations with respect to the context of Web and HTTP are made available to the RESTful services for all interactions between one another. Interaction between the services include exchanging requests and response messages which includes enough data describing the methods about how that message can be processed. Information describing how a message must be processed is included in the request and response messages, which are exchanged by the Services as their mode of interaction. RESTful services are used for tactical and ad hoc integration over the internet as they are stateless and are also lightweight. Mashups are popular techniques that are used by the users to enable them to create suitable applications that are based on already existing components. [5].

Table 2.COMPARISONS OF TECHNIQUES

COMPOSITION APPROACH	COMPOSITION TYPE		COMPOSITION MANNER			COMPOSITION STRATEGY		STANDARD LANGUAGES FOR SERVICE DESCRIPTION
	SYNT.	SEMA.	MAN.	AUT.	SA	STAT.	DYNA.	
BASED ON HPN	Yes		Yes			Yes		BPEL
BASED ON PETRINET	Yes		Yes			Yes		WSDL
BASED ON SEMANTIC ANNOTATIONS		Yes	Yes				Yes	OWL-S
BASED ON ARTIFICIAL INTELLIGENCE		Yes	Yes				Yes	OWL-S
BASED ON SUGGESTIONS		Yes			Yes		Yes	SAWSDL
BASED ON MULTI AGENT		Yes	Yes				Yes	OWL-S
BASED ON CODE GENERATION		Yes		Yes	Yes			BPEL4WS

N.   ATOMIC VS. COMPOSITE WEB SERVICES

Atomic services also called elementary services [3] acts as points of access for an application which does not depend on some other Web Service to fulfill the requests of the user. An interface that is programmatic and is based on SOAP methods and Web Service Description Language is provided by the atomic service. Appropriate adaptors can be created or developed for legacy applications like CORBA, so that they can be called upon or looked upon as a Web Service.

An overview that puts together all of the remaining composed services and atomic services, that combine to set in motion, a group of operations can be called a composite service [20]. All of the services which are brought under the same service by any composite service is called its "component services". Examples of composite services include holiday package tour that puts together flight reservations, hotel reservations, locating tourist spots etc. It does not matter if a web service is atomic or composite. It is specified by a set of attributes (characteristics), a set of operations and an Identifier.

O.   ORCHESTRATION VS. CHOREOGRAPHY

XML, SOAP and WSDL are some of the standard sets of service technologies that can describe, locate and invoke a web service which is an independent entity. Web service can bring forth many tasks but every WSDL document provides information about moderately atomic and low level operations. These basic technologies do not provide lush detail on the attributes that define the role played by the service as the part of a larger, highly complicated combination [18]. Service orchestration and service choreography can be used to define the flow of activities that makes up a business process [19]. A single, executable business process which ensures the coordination of the interactions amongst the various services is represented by Service orchestration, which describes the sequence from the angle and is under control of an endpoint.

CONCLUSION

The intention of this paper is to provide an overview of Web Service Composition and the various methods that are used for

the same. Different etymologies that are related to the term Web Service, is dealt with in this paper.

REFERENCES

[1] Brian Suda, SOAP Web Services,,University of Edinburgh,2003.

[2] Web services orchestration and choreography, Athman Bouguettaya, Quan Z. Sheng, Florian Daniel (Eds.), Web Services Foundations, Springer, 2013., C. Peltz, Computer2003.

[3] Quan Z. Sheng, Boualem Benatallah, Marlon Dumas, Eileen Mak, SELF-SERV: a platform for rapid composition of Web services in a peer-to-peer environment, in: Proceedings of the 28th International Conference on Very Large Databases (VLDB'02), Morgan Kaufmann, 2002.

[4] Hassina Nacer Talantikite, Djamil Aissani, and Nacer Boudjlida, "Semantic annotations for web services discovery and composition," Computer Standards & Interfaces 31 2009 vol.31,, November 2009.

[5] Anne H.H. Ngu, Michael P. Carlson, Quan Z. Sheng, Hye-young Paik, Semantic-based mashup of composite applications, IEEE Trans. Serv. Comput. 2010.

[6] C. Pautasso, O. Zimmermann, F. Leymann, Restful Web services vs.''Big'' Web services: making the right architectural decision, in: Proceedings of the 17th International World Wide Web Conference (WWW 2008), Beijing, China, 2008.

[7] Luciano A. Digiampietri1, Jos´e J. P´erez-Alc´azar , Claudia Bauzer Medeiro,AI Planning in Web Services Composition: a review of current approaches and a new solution.(2007).

[8] Alistair Barros Marlon Dumas Phillipa Oaks,A Critical Overview of the Web Services Choreography Description Language (WS-CDL),BPTrends,2005**.**

[9] abdaladhem albreshne, patrik fuhrer, jacques pasquier,Web Services Orchestration and Composition Case Study of Web services Composition,2009.

[10] Narges Hesami Rostami and Esmaeil Kheirkhah and Mehrdad Jalali,Web Services Composition Methods And Techniques: A Review , International Journal of Computer Science, Engineering and Information Technology(IJCSEIT),Vol.3,No.6,December, 2013.

[11] Lars-˚Ake Fredlund, Web Services and Business Processes:An Overview.

[12] Athman Bouguettaya, Quan Z. Sheng, Florian Daniel (Eds.), Advanced Web Services, Springer, 2013.

[13] Athman Bouguettaya, Quan Z. Sheng, Florian Daniel (Eds.), Web Services Foundations, Springer, 2013.

[14] Massimiliano Colombo, Elisabetta Di Nitto, Marco Mauri, SCENE: a service composition execution environment supporting dynamic changes disciplined through rules, Lecture Notes in Computer Science, vol. 4294, Springer-Verlag, Berlin, Heidelberg, 2006.

[15] Service composition issues in pervasive computing,J. Bronsted, K.M. Hansen, M. Ingstrup, , IEEE Pervasive Computing, 2010.

[16] Schahram Dustdar, Wolfgang Schreiner, A survey on Web services composition, International Journal Web Grid Service.2005.

[17] D. Fensel, C. Bussler, The Web service modeling framework WSMF, Electron. Comm. Res. Appl. 2002.

[18] XitongLi,Yushun Fan, QuanZ .Sheng ,Zakaria Maamar, Hongwei Zhu, A petrinet approach to analyzing behavioral compatibility and similarity of Web services, IEEE Trans. Syst. Man Cybernet. Syst. 2011.,

[19] Athman Bouguettaya, Quan Z. Sheng, Florian Daniel (Eds.), Web Services Foundations, Springer, 2013., C. Peltz, Web services orchestration and choreography, Computer 2003.

[20] Maurice Beek, Antonio Bucchiarone, Stefania Gnesi, A survey on service composition approaches: from industrial standards to formal methods, IEEE Computer Society, Washington, DC, USA, 2007,