Web Service Composition and Service Selection Based On User Requirements through XML Interface

[1]P Gokula Krishnan , [2]G Senthil Kumar
[1]M.Tech Student, [2]Assistant Professor
[1]Software Engineering, SRM University , Chennai, India
[2]Software Engineering, SRM University , Chennai, India
[1]gokulkrishna14@gmail.com , [2]senthilkumar.g@ktr.srmuniv.ac.in,

***Abstract*— Studying user's varied requirement, composition of Web services becomes more and more important in order to fulfill user's demand. The traditional systems of web services composition don't support variability of user's requirement. Therefore, introducing user's requirement into system of web services composition and extend the modules of composition system. In this paper, we deal with the issues of reconfigurable service modeling and efficient service composition decision making, design web services composition framework based on user's requirement, and the composition module in detail. User requirement are read by defining the semantic information of web service, defining the user requirement provide trust for web services, extend the need to understand the exact attribute of web services in all aspects like, QoS Choreographies and Trust helps in Composition and Automation Framework for reconfiguration.**

***Index Terms*— WSDL, XML, SWSCF, CLSA, QoS, QSCRS. RTP, CTS**

I. INTRODUCTION

In recent years, the service-oriented architecture (SOA) has become a popular paradigm for system development in many application domains. In an SOA environment, services can be selected and composed together to satisfy the desired functional and QoS goals[1]. There have been many research works addressing the issues in service selection, based on end-to-end system QoS requirements [1]. An alternative approach that can complement service selection techniques to satisfy both static and dynamic system QoS requirements is reconfigurable services. In many application domains, basic entities that can be used to assemble the system are designed to be reconfigurable. With the rapid growth of web service on the network, user's requirement becomes changeable and real-time [1]. Therefore, it is an urgent task to compose web service quickly and generate composition service for user's requirement, according to user's service request [1],[3]. User's request consists of functional requirement as well as need for QoS. For implementing user's functional requirement, service discovery not only depend on key words search but also on function search provided by service [2],[5]. Service function is not possible to be described as some key words. Service call and automation composition need inter-operation semantic information which can be understand between services. Although one service's output parameter is same as another service's input parameter and they have same type, we can't connect them. Because we are not sure that their meaning is same. It needs support of semantic information. Because user brings up higher or personal requirement for web service's availability, reliability, price and reputation of provider, we must consider user's QoS need. User's Qos need affects all aspects such as selecting component service and optimizing compose schema [1],[2]. To find composition service which satisfies user's QoS need is the main aim. It takes an important role in development process of web service. In this paper, we deal with the problem of QoS-driven service composition involving reconfigurable services. The composition can be done at design time or runtime, which uses Cross-layer Scheduling Algorithm for reconfiguring the running web service which fails.

II. RELATED WORK

QOS-RECONFIGURABLE SERVICES

A QoS-reconfigurable service is a service with externally visible configurable parameters that can be adjusted by invokers and, when adjusted, can impact the QoS behavior of the service [1][5]. For example, A security service can choose different encryption key sizes (EKS) to achieve computation and confidentiality protection tradeoffs [1].

To facilitate the configuration of QoS-reconfigurable service, it is necessary to specify its QoS properties,[1] i.e., it is necessary to specify 1) the set of QoS attributes of potential interest, 2) the configurable parameters in the service, and 3) the QoS measures in terms of the QoS attributes for different settings of the configurable para-meters[1].

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
 <xs:element name="QoS_behavior">
  <xs:complexType>
   <xs:sequence>
    <xs:element name="QoS_attribute" type="xs:string"
                maxOccurs="unbounded"/>
    <xs:element name="environment_parameter" type="xs:string"
                minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="configurable_parameter"
                maxOccurs="unbounded"/>
     <xs:complexType>
      <xs:sequence>
       <xs:element name="name" type="xs:string"/>
       <xs:element name="type" type="xs:string"/>
       <xs:element name="domain" type="xs:string"/>
      </xs:sequence>
     </xs:complexType>
    </xs:element>
    <xs:element name="measurement">
     <xs:complexType>
      <xs:element name="attribute" type="xs:string"/>
      <xs:element name="unit" type="xs:string"/>
      <xs:element name="variables" type="xs:string"/>
      <xs:choice>
       <xs:element name="data" type="xs:string"/>
       <xs:element name="fuction" type="xs:string"/>
       <xs:element name="array">
        <xs:complexType>
         <xs:element name="array_index" type="xs:string"/>
         <xs:element name="array_data" type="xs:string"/>
        </xs:complexType>
   ...
</xs:schema>
```

Fig. 1. The XML schema for service QoS behavior specification

A. QoS Attributes and QoS Property Space of a Concrete Service

We use mathematical notations for the specification of concrete services, like those used in. This is necessary for the later discussions regarding the decision algorithms and proofs. Consider a concrete service c. $A_c = (a^1_c, a^2_{c,...} a^n_c)$ denote the vector of the QoS attributes of c [1]. Examples of QoS attributes include execution time (ET) and service reliability. A QoS property pi c of c is defined by the measurement of its QoS attribute a_{ci}. Though there are many works on QoS specification, they do not support the specification of reconfigurable services. We leverage existing techniques to annotate WSDL with OWL-S ontology and extend OWL-S by adding a QoS_behavior class to the profile class Then, we specify the QoS properties of a reconfigurable service in the QoS_behavior class and store them at the UDDI The schema for the specification is shown in Fig. 1. The <QoS_attribute> tag defines the set of QoS attributes of interest in the service and the <configurable_parameter> tag defines the set of configurable parameters and their ranges. The <environment_parameter> tag defines the parameters that need to be specified for the measurement. For example, since measuring time on different CPUs may yield different results, it is necessary to specify the CPU used for the measurement. The <measurement> tag specifies the actual QoS measurements (QoS properties). It is necessary to specify which QoS attribute (defined by the <attribute> tag) is measured, what the unit of measurement (defined by the <unit> tag) is, which configurable parameters are being changed (defined by the <variables> tag), and the measurement data. The measurement data can be expressed as a single data item, by a function, or by an array in which the raw data are kept. (Note that toward the end of the schema, some trivial closing tags are omitted.)

B. Web Services Composition Based on user Requirement

For implementing user's functional requirement, service discovery not only depend on key words search but also on function search provided by service. Only in this way can we find service needed. Service function is not possible to be described as some key words. Service call and automation composition need inter-operation semantic information which can be understand between services[2],[4]. Although one service's output parameter is same as another service's input parameter and they have same type, we can't connect them. Because we are not sure that their meaning is same. It needs support of semantic information. we analyze composition methods of on semantic web service. Then we propose a web services composition framework based on user's requirement and describe the composition module.

C. SEMANTIC WEB SERVICE COMPOSITION

Semantic Web is an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation. The foundation to build up semantic web is knowledge conceptualization, formalization and corresponding inference. In the research on semantic web, knowledge representation, ontology and intelligent agent which are inextricable are hot areas[2][6]. Ontology is relations between concepts, which promotes research on semantic web. At the same time, we can lead semantic thought in research on web service. It can improve composition accuracy and degree of automation of integration.

According to automation degree, it is classified into manual, semi-automation and full automation composition of web service. According to technology or theory, it is classified into web service composition based on work flow and based on artificial intelligence. It begins with input object user provided, and ends with output object user expected. It use forward search strategy and improved shortest path algorithm, which can search shortest path from start to finishing in relation diagram of service ontology.

III. ARCHITECTURE DESIGN

A. Proposed Architecture

The Compositional Decision Making Problem is identified the proper concrete services for grounding the abstract services in a given workflow and deciding the proper QoS-configurable parameter settings for the selected reconfigurable concrete services. To make best use of reconfigurable services, it is necessary to investigate the techniques for QoS-driven service composition analysis considering reconfigurable services. When reconfigurable services are considered, it is necessary to not only consider the selection of services that best satisfy the system QoS requirements, but also the proper settings of configurable parameters in reconfigurable services. Often, the decision problem involving reconfigurable services can be very challenging. The RTP packetization service packs the speech data after it are encoded. Multiple frames can be packed into one packet to further reduce the bandwidth requirement. Reconfiguration from a choreography perspective means either replacing certain services at runtime or modifying the composition. In our proposed system, we are trying to integrate the self healing or automatic redirection to the original state after validating the state of the affected web services.

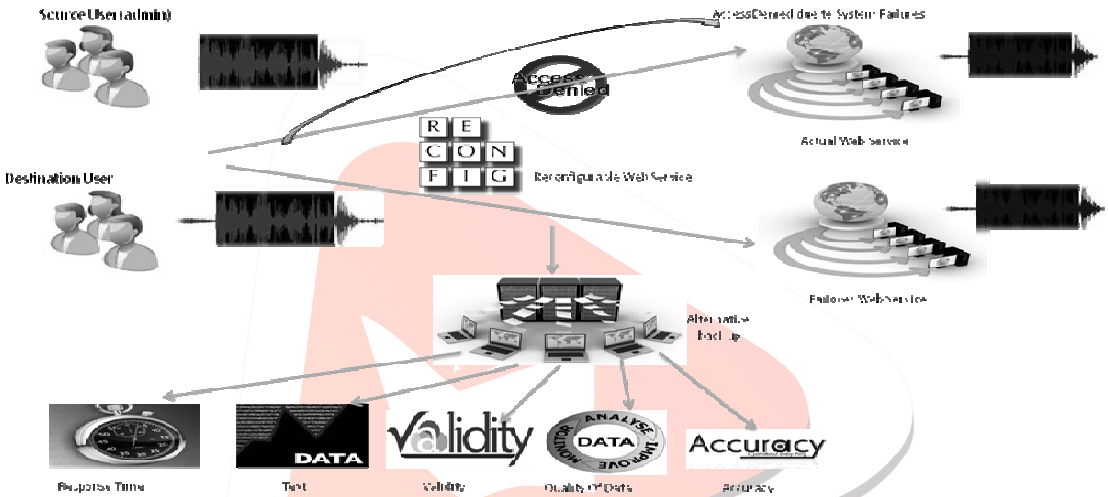

Fig. 2. System Architecture

B. Semantic Web Services Composition Framework

Semantic web services composition framework has two function modules, component service management and composition service management. The former contains functions of service register and semantic service classifying. The latter is divided into Service composition building up[2] , composition verification and so on. To implement the above function of SWSCF, we need corresponding function component. SWSCF only explains specific function which each component provides, and doesn't mention implementation plan of each component.

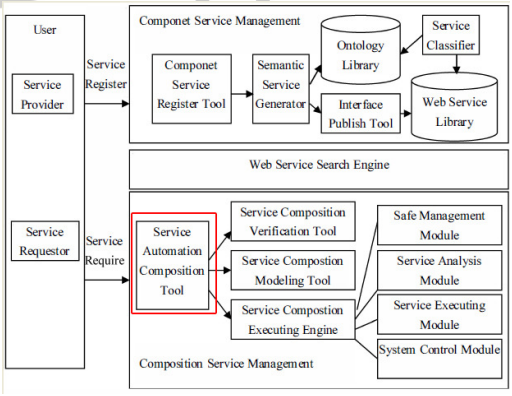

Fig. 3. Semantic Web Services Composition Framework

The composition process of user-driven web service is launch by user. When user put forward unsolved task by service automation composition tool, user need to give precondition provided by solving task, point out his expected result and QoS constraint of request service. After achieve task description, web service automation composition tool searches service collection relevant to task by service search engine. Then, it uses automation composition methods to package service immediately into implement plan.

C. Problem Addressed

Web service composition and discovery The most promising feature of the Web services platform is its ability to form new (composite) services by combining the capabilities of already existing (component) services. The existing services may themselves be composite leading to a hierarchical composition. QoS constraints . Algorithms are needed to select service components with various QoS levels according to some application-dependent performance requirements. Automatic recovery and reconfiguration The web services are created and stored in the service registry. To discover the efficient services based on the user request domain ontology is created. User- friendly trusted service discovery Trust is one of the most critical quality factors for service requestors when they select the services from a large pool of Web services.

However, existing web service trust models either do not focus on satisfying user's preferences for different quality of service (QoS) attributes, or do not pay enough attention to the impact of malicious ratings on trust evaluation. Constructing Methods of Web Service Composition, semantic information of interface parameters. Using which Service select algorithm is selected and they actually use shortest path and graph search to select web service. QoS definition of service, are embedded with the WSDL schema in Open Specification format helps maintaining the Detail of web service available. Service deployment in is always been an issue ever since as it have include the application logic as well as the Database required, so deployment complexity is considered to design new framework. Trust is one of the most critical quality factors for service requestors when they select the services from a large pool of Web services.

IV. EXPERIMENTAL SETUP

Entire proposed system is divided into sub components which together construct the required system. A QoS Verifications component which runs as soon as the web service is started, listen to the service request and also measure the service QoS in form of Semantics. All the available WSDL documents with describes the service are inherited using Microsoft web.

When reconfigurable services are considered, it is necessary to not only consider the selection of services that best satisfy the system QoS requirements, but also the proper settings of configurable parameters in reconfigurable services. The RTP packetization service packs the speech data after it are encoded. Multiple frames can be packed into one packet to further reduce the bandwidth requirement. Reconfiguration from a choreography perspective means either replacing certain services at runtime or modifying the composition.

A. ADMINISTRATION DETAILS:

It is important for handling the data in the database; since data come from different user to the server it is mandatory to have an administration to handle the data in serve. Which is later can be backed up in using the similar kind of web service.

B. WSDL DOCUMENTS:

Microsoft defined WSDL documents which are available in open document format are used to understand the semantic information of web service. WSDL explains the type use and constraints for new composable web service.

C. QoS BEHAVIOR SPECIFICATION:

it is necessary to specify its QoS properties, i.e., it is necessary to specify :

1) The set of QoS attributes of potential interest.
2) The configurable parameters in the service.
3) The QoS measures in terms of the QoS attributes for different settings of the configurable parameters

.

D. SEMANTIC FILE SYSTEM:

Web service semantic information are extracted in order to provide the accuracy for the user requested service. Semantic information defines the behavior of the service in accordance with the working and purpose of the composition. SemanticFileSystem uses the Semantics in an .dll file.

E. SAMPLE DATA:

A set of audio files is taken as the sample data in the database. Sample data consist of minimal constraints in terms of memory, Bit rate, and formats (.wav,.mp3).

V. JUSTIFICATION

A. Service availability:

There may be many services available for the user convenience. So whenever a particular web service becomes unavailable, user can able to choose alternate services to overcome the response time. All the changes can be managed and maintained in the database, so that it will be automatically updated. These changes will not affect any existing selected services. Web service requested is made available 24x7 using Load balancing technique

B. Reliability:

Service defined remains operational throughout the session of the request and service release. Service is reliable till the execution gets over.

C. Performance:

Execution Time and Response time is very important for a web service, so new framework would concentrate on performance. Cross-Layer Scheduling Algorithm ensures that the performance of the service doesn't lag the original service which the user working on.

VI. CONCLUSION

In this paper, we study on the methods of web service composition. Mainly, we illustrate the design of web service composition system base on user's requirement, and introduce each module, especially automation composition module. After composition process, QoS management module chooses component service. Then composition scheme obtained can satisty user's function and non-function requirement. We have proposed the concept of reconfigurable services, developed a model for specifying the QoS properties of reconfigurable services, and considered the problem of QoS-driven service composition with reconfigurable services. Due to the large composition solution space introduced by reconfigurable services, we have proposed the CLS approach for efficient search of Pareto-optimal composition solutions in QSCRS. In future, the proposed system can be extended to heavy weight web applications with full trust and QoS choreographies.

REFERENCES

[1] A User-Oriented Trust Model for Web Services Bixin Li, Rui Song, Li Liao, Cuicui Liu, School of Computer Science and Engineering, Southeast University, Nanjing 211189, Jiangsu, China, 2013

[2] Web Services Composition Based on User Requirement, The 8th International Conference on Computer Science & Education (ICCSE 2013) April 26-28, 2013. Colombo, Sri Lanka

[3] Adaptive MOEA/D for QoS-Based Web Service Composition, M. Middendorf and C. Blum (Eds.): EvoCOP 2013, LNCS 7832, pp. 73–84, 2013.c┐ Springer-Verlag Berlin Heidelberg 2013

[4] QoS Management of Web Services, ZHENG, Zibin, Computer Science and Engineering,The Chinese University of Hong Kong, January, 2011

[5] QoS-Driven Service Composition with Reconfigurable Services Hui Ma, Favyen Bastani, I-Ling Yen, and Hong Mei, Senior Member, IEEE, IEEE TRANSACTIONS ON SERVICES COMPUTING, VOL. 6, NO. 1, JANUARY-MARCH 2013

[6] "3GPP TS26.071: AMR Speech CODEC," http://www.3gpp.org/ ftp/Specs/html-info/26071.htm, 2012.

[7] A. D'Ambrogio, "A Model-Driven WSDL Extension for Describing the QoS of Web Services," Proc. Int'l Conf. Web Services (ICWS), pp. 789-796, Sept. 2006.