Semantic Web: An Integrated Approach for Web Service Discovery, Selection and Composition

Michael Raj.T.F*
Research and Development Centre
Bharathiar University,Coimbatore, TamilNadu, India
Email:francismike009@yahoo.com

Sivapragasam.P
Associate Prof., Dept. of Computer Science
Sri Vasavi College, Erode
TamilNadu, India

Abstract-Web services are remote methods can be invoked through open standards such as Simple Object Access Protocols. The increasing web services in the repositories makes the selection process very complex. The same can be extended in forming the composition of web services. This research focuses on the semantic web service selection and composition through design and implementation of a framework. The proposed framework is an ontology based service selection approach and the selected services are participating in the composition process. This approach deals with semantic search, which uses Quality of services for service selection and composition. The entire framework is implemented with semantic web technology and the performance of the system is observed with domain specific ontologies.

Keywords:*Semantic web, Service selection, Framework for web service composition, Ontologies, Semantic tool.*

*****

I. INTRODUCTION

Business processes are made easy by means ofService OrientedArchitecture (SOA) and Service Oriented Computing (SOC). SOA is the basic concept for distributed computing, Internet Marketing (IM) and social networks and isdepicted in the Figure 1. SOA provides a platform for sharing functional type resources called web services. Web services are remote methods or APIs or software components can be treated as machine-processable interfaces. These web services can be used for accessing and processing the data in the distributed environments. Web services are enabled with great features, it supports interoperability and provides greater and easier integration. To demonstrate all these features it is necessary to implement two important tasks namely service discovery-selection and service composition. These tasks can be achieved by means of greater efforts, because there exists vast number of services are available in the repositories. The service discovery-selection process starts with processing the UDDI, which consists of syntactic description for the web services and are categorized using the prefixes [1].

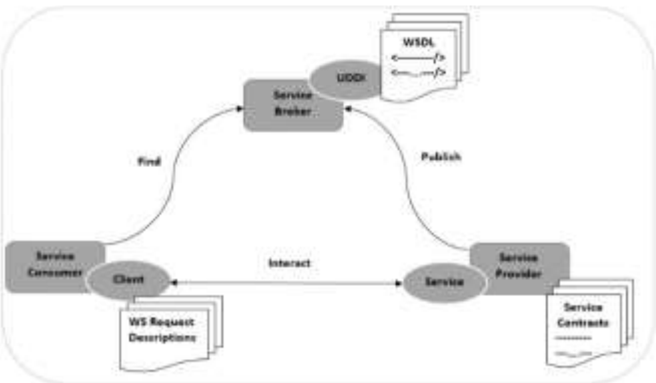

Fig 1. Service Oriented Architecture (SOA)

Other standards such as WSDL and SOAPwere Web Service Description Language (WSDL) describes the web services and make it as a machine-processable entity and Simple Object Access Protocol (SOAP) handles the messages between the requests and responses, commonly conveyed through the use of HTTP and XML [2-4]. Service descriptions are plays a vital role in the web service discovery and composition. It took participate in the automated solutions [5-7]. These descriptions can be classified as syntactic and semantic. Syntactic based service descriptions are distributed across the domain.It can be processed by either syntactic or semantic approach, in which input and output parameters are fetched and matched in all the phases of the designated framework as a central activity [8-15].

The communication between the heterogeneous machines through the message are depends upon the richness of the service descriptions and this enables the automation of the various processes of the framework such as service discovery, service selection and invocation. The richness of the descriptions should be maintained throughout the life cycle of the framework. There is a strong dependency between service fetching and matching but the research in development and implementation of these tasks are being carried out independently.The performance of the discovery or the web service search engines depends upon the following parameters such as well-defined interfaces and categorization of the web services. When the number of services are increased in the repositories will increase the discovery and composition time. This time can be minimized and the efficiency of the composition can be improved by adopting appropriate algorithms in the discovery engine. To implement fuller automatic system one should understand and use the complete semantics of the web service. These semantics improves the automatic features such as verification, simulation, configuration, Supply Chain Management (SCM), contracting and negotiation of services.

The conventional web contents are static and syntactic that focuses on issuing dynamic contents and the interaction among them. World Wide Web(WWW) and the WWW Consortium (W3C) suggested that a software system which supports interactions between the machines in a distributed environment. The inherent of the current web is a Semantic Web (SW), which makes the information retrieval process easier in-terms of searching, finding, extracting the contents, if they are semantically described.Semantic web is quite interesting then the syntactic webs.It can have different definitions, according to Tim Berners-Lee:

The 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. . . . a web of data that can be processed directly and indirectly by machines [16]**.**

In syntactic web if I am searching for "What is semantic web?" in google its giving 62, 60,000 results in 71 milliseconds, but how about the relevancy of the results is a big debate. Some pages containing relevant information and some of them are not. Sematic web provides the solutions to this issue using the key conceptsbe organized as layers, which is described in the following Figure 2. Resource Description Framework (RDF) is the backbone of semantic web, it describes the data on the web by which it gives an abstract model to define the knowledgeandit conveys the meanings among them. The key components of RDF are statement,subject, object and predicate enables the web resources integration through which they become machine understandable.

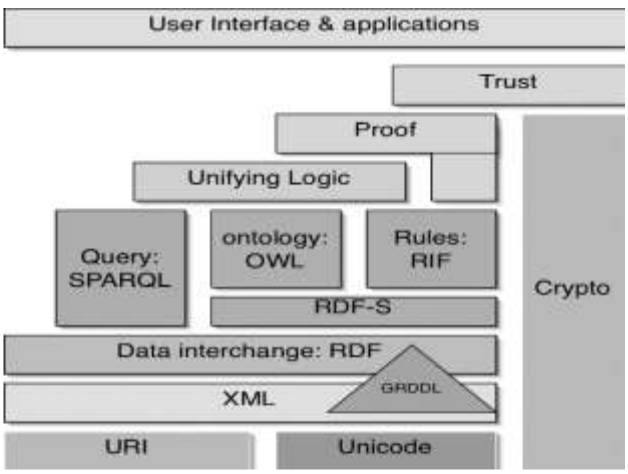

Fig 2 Semantic web layers

Web services in the semantic web are required to be discovered, selected and composited. Web services have four types of operations request response, notification, solicit response and an event. The earlier two operations are synchronous and the latter two are asynchronous operations. The above said operations are influenced by various parameters. The objective of this research is to provide an integrated framework for the web services composition in the semantic web by considering the Quality of Services (QoS) and finding the optimal solution.

The entire article is organized as follows, Section-I discuss the introductory concepts of the web, semantic web and protocols used. Section-II describes the semantic web services, the need of service selection, composition and the challenges. Related work is explained with the background of the research in the Section-III.Section-IV evaluates the proposed approach with the obtained results. The conclusion and the future work is discussed in the Section-V.

II. OVERVIEW OF SEMANTIC WEB SERVICES

Syntactical approaches do not describe the QoS parameters as part of the Web Service (WS) description, whereas semantic standards OWL-S can describe the QoS concepts. UDDI can be extended to specify the QoS description proposed by Ran [17]. Well defined semantics are used to describe the Semantic Web Service (SWS), which enables them as a machine understandable entity [18]. OWL-S is a web ontology language for web services, recommended by W3C, which can be used for developing and annotating semantic web services. OWL integrates the concepts of RDF and new definitions to express the concepts. The existing web service descriptions can be extended to enable the semantic features over WS with the help of OWL-S.Consideration of UDDI entities plays a vital role in the conversion from syntactic into semantic descriptions. These entities are described in the Figure 3.

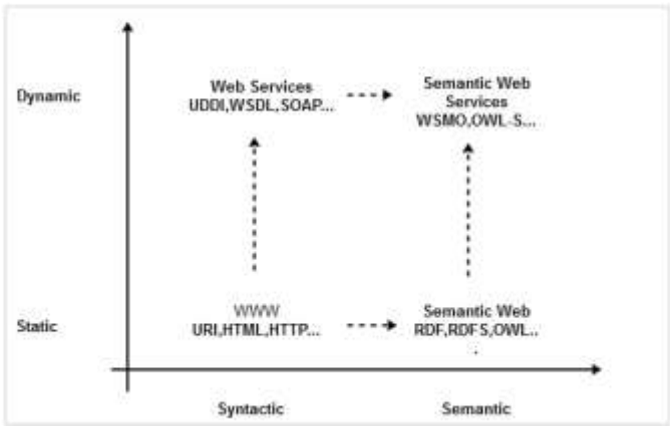

Figure 3. Syntactic vs Semantic web services

OWL-S is a language used to annotate semantic WS through which an unambiguous model can be developed. There are number of languages such as SemTag, SHOE, AeroDAML, SEE available to annotate WSs.Bouchiha et al explained various annotations languages [19] and the architecture is given in [20]. There are four types of semantics in WS Data/Information, Functional/Operational, Non-Functional/ QoS and Execution semantics [21].OWL-S is also providing an upper ontology services for semantic web. Upper ontology used to describe the classes and properties of a WS. Sub ontologies of OWL-S upper ontologies are shown in the following Figure 4.Reasoning and selections to the Information Retrieval (IR) are the two important concepts required by the semantic WSs for the OWL-S, WSMO and SWSF frameworks are used

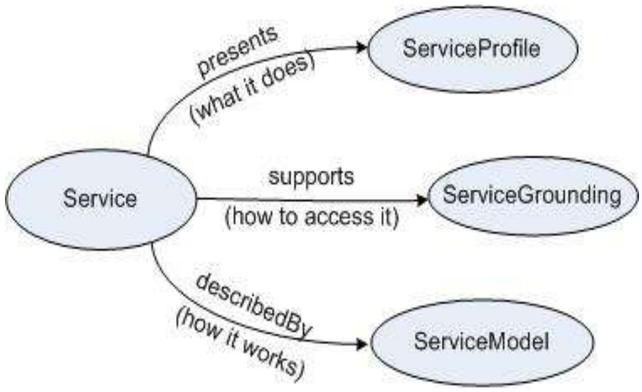

Fig 4. OWL-S Upper ontology

2.1 SEMANTIC WEB SERVICE COMPOSITION (SWSC)

The service composition process starts with service matching and the service discovery.

2.1.1 SERVICE MATCHING

The semantic service-matching algorithm compares the semantics of a desired service with the available services and selects the appropriate service using various matching approaches such as logic based semantic selection, hybrid semantic selection and adoptive hybrid service selection. This matching process takes place either inside or outside the semantic framework. The following table describes the tools that implements various matching approaches is described on the Table 1.

Table 1: Matching Approaches-Overview

S.No	Category	Tools
1.	Logic based	OWL-S, iMatcher, SPARQLENT
2.	Hybrid	SCMAL,OWL-S, SLRlite, XSSD, OPPOSUM, ALIVE,EMMO
3.	Adoptive hybrid	OWLS-MX3, ISem, OWLS-iMatcher2

2.1.2 SEMANTIC SERVICE SELECTION (S3)

Plethora of test patterns were developed by computing the measures like processing time (runtime), precision recall and so on. These patterns are described based on the formalisms OWL-S and SAWSDL and test platforms. The results of the evaluation process for the S3 is fully depends upon the Test Collections (TC) such as OWLS-TC and SAWSDL-TC. Semantic Evaluation At Large Scales (SEALS) project provides numerous challenges to S3*.

2.1.3 CLASSIFICATION OF SWSC

Service composition is the process of combing the web services from the available service to create value added services, and then the composite services are called composite services [28]. The composition process can be implemented in two ways namely syntactic and semantic. The following Table 2 shows the classification of web service composition approaches and its features, Table 3 describes the languages available for composition and Table 4 lists the limitations of the QoS composition approaches. It will be helpful for the proposal of new approaches. Most of the approaches are based on workflow model and AI planning.

Workflow approaches may follow by either static or dynamic workflow generation. Static workflow model process its tasks based on the abstract model, which specifies the set of tasks and their data relations. In a different, the dynamic workflow models select the composite services automatically based on several constraints such as data dependency preferences and so on.EFLOW is an example for static workflow process model [29] and Polymorphic Process Model (PPM) is hybrid process model that it combines the features of both static and dynamic process models [30]. SHOP2 is an AI planning based automatic composition tool [31].Most of the WS compositions are achieved through the AI planning methods in which a planning can be described as a five tuples $<S, S_0, G, A, r>$

Where,

S-is set of states

S_0& G are initial and goal states given by the Service Consumer's (SC) request contained in the query (Q)

A-is set of services in the repository given by the Service Providers (SP)

r- is a function changes the state of a service that is changes from initial state to goal state using input parameters and produces output parameters [32].

ISSN: 2454-4248
314 – 322

Table 2. Composition approaches and features

Composition approaches	Features										
	Automatic Discovery	Complexity	Dynamic composition	Extended goals	Runtime failure	Failure recovery	Monitoring And controlling	QoS support	Scalability and Performance	Cost	Service Markup
ASTRO	±	✓	X	✓	-	X	✓	X			OWL-S
ConGolog	±	✓	X	✓	X	X	±	X			OWL-S
Haley	±	✓	✓	X	NA	✓	✓	✓		✓	SAWSDL
SHOP	X	✓	X	±	-	X	±	X			OWL-S
SWORD	±	X	X	X	-	X	X	X			XML
TLPLAN	✓	±	✓	±	+, -	✓	±	✓	✓		OWL-S

✓ Supported, X- Not supported, ± Partially supports, + Proactive, - Reactive

Table 3 Composition languages

Languages / Features	Modelling the collaboration	Modelling the execution control	Representation of the role	Transaction and compensation	Exception handling	Semantic support	Business agreement support	Software vendor support
BPEL4WS	SS	SS	WS	IS	SS	NS	NS	SS
BPML	IS	SS	NS	SS	SS	NS	NS	PS
WS-CDL	SS	NS	SS	IS	S	NS	NS	NS
WSCI	SS	NS	SS	SS	SS	NS	NS	PS
DAML-S	SS	SS	NS	IS	SS	SS	NS	PS

SS-Strong support
S-Support
WS-Weak support

PS-Partial support
IS-Indirect support
NS-No support

Table 4. Limitations of QoS composition approaches

S.No	Limitations of QoS Compositions
1.	Syntactic QoS based description [22]
2.	QoS constraints & metrics is not sufficient [23]
3.	WSOL- no QoS specification of demand, QoS metrics are developed [24]
4.	WSLAs are described syntactically [25]
5.	Ontology based QoS-WS description are can be inaccurate & incomplete [26]
6.	Unary QoS metric constraints can be expressed and QoS matching is manual and OWL-S extended by means of QoS ontology [27]

2.1.4. SEMANTIC MATCHING OF QOS PARAMETERS

We propose a brand new approach to achieve the QoS matching degree between a service requester and a service provider. We calculate the QoS parameters from both the semantic and numerical components. Step 1 is to calculate the semantic matching degree of QoS parameters;it will be introduced in the following segment 3. Step 2 is to calculate the numerical matching degree of requester and a service provider [33]. QoS ontology is used that can assure the QoS semantic consistency, in which the I/O parameters standards might be substituted through Qos concepts, the I/O domain ontology taxonomy tree may be substituted through QoS ontology taxonomy tree. If there are many QoS parameters, we are able to use the QoSWeight attribute of our QoS ontology to calculate for each parameters fee.

III. RELATED WORK

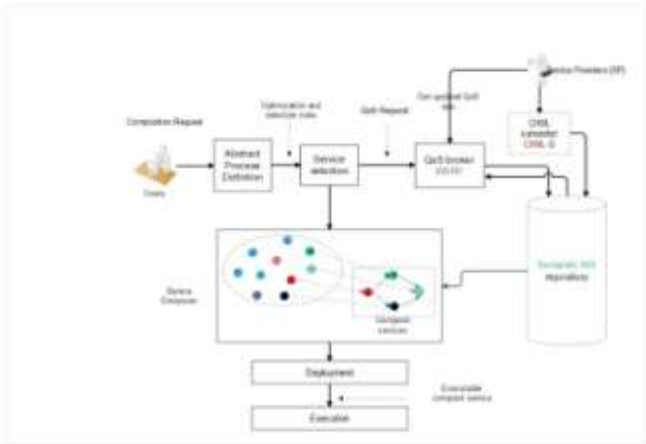

Fig 5. SWSC Framework

Web service composition is a set of atomic web services operates together to achieve a task. We have proposed a framework called Semantic Web Service Composition (SWSC) to compose web services based on the QOS semantics given in the Figure 5, which uses OWL-S based and hierarchically integrated resources contains QOS attributes that is non-functional attributes. The objective of this framework is to provide a composition file consist of the service calls in which each abstract service can have the same structure as that of OWL-S service. The following figure shows the architecture of the proposed framework performs various operations such as OWL-S convertor, service composition planner, QoS service selection and service composition. The entire operations are summarized as follows:

1. Initially web Service Provider (SP) loads the services into the service registry
   1.1. OWL-S convertor converts WSDL files into semantically described contents tomake machine processable.
2. QoS broker retrieves the QoS parameters from semantic repository and calculates the WSRF values and then update the same into the semantic repository
3. Service Consumer (SC) sending the service request details which contains the QoS details and abstract composition plan details
4. Service selection module uses QoS attributes and applies the Matchmaking operations on the semantic repository
5. Selected services are the inputs for the service composition module, where the relationships between the selected services are determined and proposed composition is obtained.

3.1. MATCHMAKING

Matchmaking process can be achieved in-terms of three compare operations input-output matching, functional matching and QoS matching called non-functional matching. Matching can be formally defined as follows:

Let S be the set of web services in the service repository and R be the requesting web service then the matching operation function ((M) of the repository server will returns the set of all the web services (Ws) which are compatible with the matching function M(R) :M(R)={Ws ∊ S|compatible (Ws, R) }.It implies that the descriptions of the web services (WD_1) and (WD_2) are also compatible: Compatible (WD_1, WD_2) $\Leftrightarrow \neg (WD_1 \sqcap WD_2 \sqsubseteq \perp)$.Matching results may belong to one of the five categories are as following [44]

Exact-if web services (Ws) and request ® are equivalent concepts then the match can be defined as Ws ≡R

Plugin-If Web service (Ws) includes the Request (R) R⊆Ws

Subsume- If a request R is a superset of Ws :Ws⊆R

Intersection- If the intersection of web service Ws and request R is capable $\neg (Ws \sqcap R \sqsubseteq \perp)$

Disjoint- if the Request R is not existing in the Ws $Ws \sqcap R \sqsubseteq \perp$

3.2. QOS ONTOLOGY

Different QoS ontology have been proposed to improve the web service selection process based on the nonfunctional attributes. For example DAML-QOS [34], QOSOnt [35], OWL-QL [36], OnQOS [37], QOSMO[38], QOHOnt [39], OSQS[40]. OWL-S profile ontology describes four types of information such as service details, functional descriptions-Input, Output, Preconditions and Effects (IOPE)of a service classification information details about a service and nonfunctional details which includes QoS . The following list describes the OWL-S profile ontology

<!-- IOPE --!>
<profile:hasInput rdf:resource="&getMegaPixelProcess; #model"/>
<profile:hasOutput rdf:resource="&getMegaPixelProcess;#pixel"/>
<!-- end of IOPE -->

Algorithm: ConceptDOM(C_1, C_2)

Input : C_1 and C_2 are the two concepts defined in QoS ontologies and x,y ∊ (0,1) and x>y
Output :Matching results between two concepts C_1 and C_2
begin
If($C_1 \equiv C_2$) then
return (1)
else if (C_2 subsumes C_1) then
return (x)
else if (C_1 subsumes C_2) then
return (y)
else return (0)
end

Algorithm: inputDOM(I_P, I_R)

Input : I_P and I_R are input parameters
Output :Input parameters degree of match
begin
inputDOM=1;
for each i in I_P
find r in I_R
m=Max(ConceptDOM(r,i))
if (m>0) then
*inputDOM= inputDOM*m;*
else
return (0);
return (inputDOM)
end

Algorithm:outputDOM(O_P, O_R)

Input : O_P and O_R are input parameters
Output :Output parameters degree of match
begin
outputDOM=1;
for each o in O_P
find r in O_R
m=Max(ConceptDOM(o,r))
if (m>0) then
*outputDOM= outputDOM*m;*
else
return (0);
return (outputDOM)
end

QoS parameters are defined by the service consumers (SC). Let Q ∊ $\{q_1, q_2, \ldots\ldots\ldots\ldots q_n\}$
Where,
Q is a set consists of nonfunctional parameters, may include quantitative and qualitative values. QoS attributes can be normalized by using the equation Eq(2).

$D_i = [low_i, high_i]$, $D_i = < q_1, q_2, \ldots\ldots\ldots\ldots q_n >$

$$Utility_i(q_i) = \begin{cases} \dfrac{q_i - (q_i)_{\min}}{(q_i)_{\max} - (q_i)_{\min}} & if Q_i \in QoS^{Increment} \\ \dfrac{(q_i)_{\max} - q_i}{(q_i)_{\max} - (q_i)_{\min}} & if Q_i \in QoS^{Decrement} \\ 1 & if\ (q_i)_{\max} - (q_i)_{\min} \end{cases}$$

……………………….Eq(2)

Algorithm: qosSEARCHDOM (Q_R, Q_P)

Input : Q_P and Q_R are input parameters
Output :QoS parameters degree of match
begin
qosSEARCHDOM=0
for each q in Q_R with value V then
qosSEARCHDOM+=W_iS_i (V)
return (qosSEARCHDOM)
end

Where,
S_i is a aggregate functionfor QoS S_i:$D_i \rightarrow [0,1]$
Normalized weight function $\sum_{1 \leq I \geq n} W_i = 1$, W_i is relative weight

Aggregate matching degree:Degree of Match= $\mathbf{W_i}$ **x** inputDOM + W_o x outputDOM + W_q qosSEARCHDOM

ISSN: 2454-4248
314 – 322

3.3 SERVICE COMPOSITION

Composition plan can be described using either as data flow composition or control flow composition. In data flow composition approach, the relation between the input and output messages are considered to find the dependency between the services and then composition plan is generated at run time. Whereas in the control flow composition approach describes the order of execution of the web service to find the composition plan at run time [41][42].

The service composition module gives a composite service using the following steps

1.Gets the SWSs details from the OWL-S service profiles
2. Find the relation between the services based on the Input-Output parameters
3. Sort the services based on the matching criteria
4. Find the composition services
5. Compare with the abstract composite plan
6. Provide the normalized results to the service requesters.

3.4 WEB SERVICE DEPENDENCY RELATION MATRIX (WSDRM)

The relation between the web services can be expressed by matching the I/O parameters of the web services. The Dependency Relation Factor (DRF) can be determined by the relations between the two random web services.

Definition: *Let N e the number of web services in the repository and N x N be the dependency relation matrix. The DRF obtained between w_i and w_j can be calculated by the equation:*

$$DRF\ (w_i, w_j) = \frac{|\ wi.O \cap wj.I|}{wj.I} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad Eq(1)$$

3.4.1 COMPOSITION GRAPH GENERATION

Composition graph can be generated based on the Service Request. An SR consists of input and output definitions that is parameters of a candidate service. The origin and destination of the service composition plan can be identified by the DRF and the weight of each edges can be calculated by means of QoS attributes. Finally, the desired directed composition graph is generated, shown in the Figure 6.

IV. RESULTS AND ANALYSIS

The framework performance is tested with tours and travel web services and results are obtained from various phases of the composition framework to get the feasibility of the proposed approach. Initially the composition request is given in the form I/O parameters and QOS attributes. An abstract composition plan (ACP) is formed based on given details. The ACP gives a template for the composition engine. Comparing Semantic net services is obviously a hard mission a first step in this path changed into made by way of Petrie et al. [43].The definitions of initial and goal ontologies are making the composition process in an easier way. The similarity measures are used to select the web services.

Ontologies are created to define a domain, based on the ontology specification multiple services are selected using the QoS attributes. Framework gives the composition plan as a connected graph. One or more services belongs to the same category but with different QoS values. The composition plan can be modified to obtain a normalized plan. The number of services in the composition plan may vary according to the QoS attributes that we are considering optimization. Table 5 shows the Normalized QoS values for selected services. Figure 7 describes the number of services selected for a specific request.

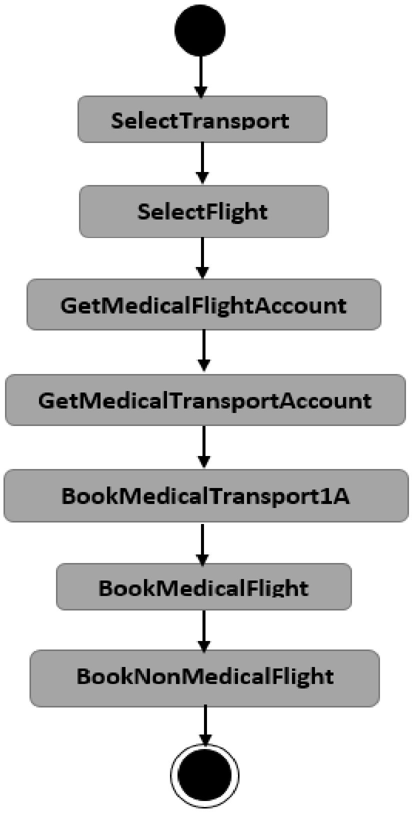

Fig 6. Composition plan

Table 5. Normalized QoS values for selected services

Service Name	Execution Duration	Price	Availability	Reliability
SelectTransport	15	8	0.69	0.55
SelectFlight	17	8	0.79	0.98
GetMedicalFlightAccount	17	8	0.49	0.51
GetMedicalTransport Account	15	9	0.65	0.56

BookMedicalTransport1A	16	10	0.5	0.55
BookMedicalFlight	15	9	0.65	0.56
BookNonMedicalFlight	15	9	0.65	0.56

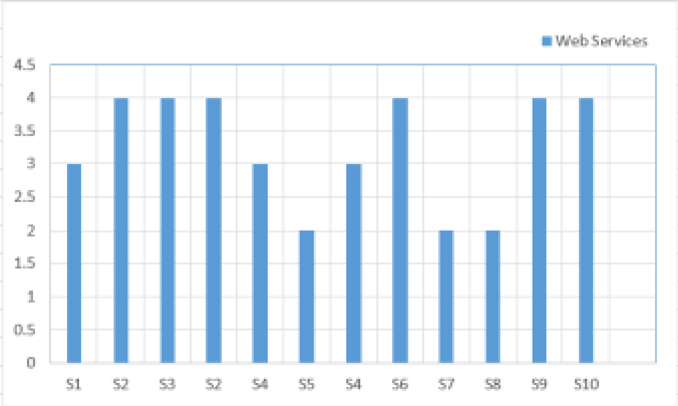

Fig 7. Matching services for a request

V. CONCLUSION

In this work, a semantic web service composition framework is presented in order to generate a composition plan by considering the QoS attributes. The proposed approach is realized in the semantic web service composition system called Semantic Web Service Composition (SWSC). The objective of the work is obtained due to the number of web services increasing every day and are needed to be searched, selected and composed from the repository, which consists of pool of services. The SWSC accepts initial ontology, goal ontology and the QoS parameters of the services. The designated frameworkproduces list of services and obtains the relation between them. The selected services are composed based on the DRF and then composition plan is generated. The feasibility of the SWSC is verified through a set of experiments and the results shows that it is an efficient approach to generate a QoS based composition plans. The future of the work is to improve the performance of the framework by minimizing the execution time and optimizing the composition plan through the selection of appropriate services.

REFERENCES

[1]. G. Alonso, F. Casati, H. Kuno, and V. Machiraju, Web Services, ser. Data-Centric Systems and Applications. Springer, 2004.c
[2]. http://www.w3.org/TR/wsdl
[3]. http://www.uddi.org/pubs/uddi v3.htm
[4]. http://www.w3.org/TR/soap
[5]. B. Srivastava and J. Koehler, “Web Service Composition – Current Solutions and Open Problems,” in ICAPS 2003 workshop on Planning for Web Services, 2003, pp. 28–35.
[6]. J. Rao and X. Su, “A Survey of Automated Web Service Composition Methods,” in Semantic Web Services and Web Process Composition, vol. 3387, 2004, pp. 43–54.
[7]. S. Dustdar and W. Schreiner, “A survey on web services composition,” Int. J. of Web and Grid Services, vol. 1, no. 1, pp. 1–30, 2005.
[8]. S. Kona, A. Bansal, M. B. Blake, and G. Gupta, “Generalized Semantics-Based Service Composition,” IEEE Int. Conf. on Web Services, pp. 219–227, 2008.
[9]. Y. Yan, B. Xu, and Z. Gu, “Automatic Service Composition Using AND/OR Graph,” in 10th IEEE Conf. on E-Commerce Technology and the Fifth IEEE Conf. on Enterprise Computing, E-Commerce and E-Services, 2008, pp. 335–338.
[10]. M. Aiello, N. Benthem, and E. Khoury, “Visualizing Compositions of Services from Large Repositories,” in 10th IEEE Conf. on E-Commerce Technology and the Fifth IEEE Conf. on Enterprise Computing, E-Commerce and E-Services. IEEE, 2008, pp. 359–362.
[11]. W. Nam, H. Kil, and D. Lee, “Type-Aware Web Service Composition Using Boolean Satisfiability Solver,” 2008 10th IEEE Conf. on E-Commerce Technology and the Fifth IEEE Conf. on Enterprise Computing, E-Commerce and E-Services, vol. 1, pp. 331–334, 2008.
[12]. K. Raman, Y. Z. Y. Zhang, M. Panahi, and K.-J. L. K.-J. Lin, “Customizable Business Process Composition with Query Optimization,” 2008 10th IEEE Conf. on E-Commerce Technology and the Fifth IEEE Conf. on Enterprise Computing, E-Commerce and EServices, 2008.
[13]. M. Shiaa, J. Fladmark, and B. Thiell, “An Incremental Graph based Approach to Automatic Service Composition,” in 2008 IEEE Int. Conf. on Services Computing. IEEE, 2008, pp. 397–404.
[14]. P. Hennig and W.-T. Balke, “Highly Scalable Web Service Composition Using Binary Tree-Based Parallelization,” 2010 IEEE Int. Conf. on Web Services, pp. 123–130, 2010.
[15]. S. Oh, D. Lee, and S. Kumara, “Web service planner (WSPR): an effective and scalable web service composition algorithm,” Int. Journal of Web Services Research, vol. 4, no. 1, pp. 1–22, 2007.
[16]. Berners-Lee, Tim; James Hendler; Ora Lassila (May 17, 2001). "The Semantic Web". Scientific American Magazine. Retrieved March 26, 2008.
[17]. S. Ran. A model for web services discovery with qos. SIGecom Exch., 4(1):1-10, 2003.
[18]. Katia P. Sycara, Massimo Paolucci, Anupriya Ankolekar, and Naveen Srinivasan. Automated discovery, interaction and composition of semantic web services. Web Semantics: Science, Services and Agents on the World Wide Web, 1(1):27–46, 2003. 4, 6
[19]. Bouchiha, Djelloul, and Mimoun Malki. "Semantic Annotation of Web Services." ICWIT. 2012.
[20]. Nagarajan, Meenakshi. "Semantic annotations in web services." Semantic Web Services, Processes and Applications (2006): 35-61.
[21]. Verma, Kunal, et al. "Optimal adaptation in autonomic web processes with inter-service dependencies." (2005).
[22]. S. Ran. A model for web services discovery with qos. SIGecom Exch., 4(1):1-10, 2003.
[23]. O. Martin-Diaz, A. R. Cortes, D. Benavides, A. Durian, and M. Toro. A quality-aware approach to web services procurement. In B. Benatallah and M.- C. Shan, editors, Technologies for E-Services (TES), volume 2819 of Lecture Notes in Computer Science, pages 42-53. Springer, 2003.
[24]. E. M. Maximilien and M. P. Singh. Conceptual model of web service reputation. SIGMOD Rec., 31(4):36-41,2002.
[25]. A. Keller and H. Ludwig. The wsla framework: Specifying and monitoring service level agreements for webservices. Technical Report RC22456 (W0205-171), IBM, 2002.
[26]. M. Tian, A. Gramm, M. Nabulsi, H. Ritter, J. Schiller, and T. Voigt. Qos integration in web services. Gesellschaft fur

Informatik DWS 2003, Doktoranden- workshop Technologien und Anwendungen von XML, October 2003.

[27]. C. Zhou, L.-T. Chia, and B.-S. Lee. Daml-qos ontology for web services. In ICWS '04: Proceedings of the IEEE International Conference on Web Services (ICWS'04), page 472, Washington, DC, USA, 2004. IEEE Computer Society.

[28]. Lemos, Angel Lagares, Florian Daniel, and Boualem Benatallah. "Web service composition: a survey of techniques and tools." ACM Computing Surveys (CSUR) 48.3 (2016): 33.

[29]. F. Casati, S. Ilnicki, and L. Jin. Adaptive and dynamic service composition in EFlow. In Proceedings of 12th International Conference on Advanced Information Systems Engineering(CAiSE), Stockholm, Sweden, June 2000. Springer Verlag.

[30]. H. Schuster, D. Georgakopoulos, A. Cichocki, and D. Baker. Modeling and composing service-based and reference process-based multi-enterprise processes. In Proceeding of 12th International Conference on Advanced Information Systems Engineering (CAiSE), Stockholm, Sweden, June 2000. Springer Verlag.

[31]. D. Wu, E. Sirin, J. Hendler, D. Nau, and B. Parsia. Automatic Web services composition using SHOP2. In Workshop on Planning for Web Services, Trento, Italy, June 2003.

[32]. Rao, Jinghai, and Xiaomeng Su. "A survey of automated web service composition methods." In SWSWPC, vol. 3387, pp. 43-54. 2004.

[33]. XML and Semantic Web W3C Standards Timeline" (PDF). 2012-02-04.

[34]. Zhou,C., Chia, L.T., Lee, B.S.:DAML-QoS Ontology for WebS ervices .In: Proceedings of the IEEE International Conference on Web Services, ICWS 2004(2004)

[35]. Dobson,G., Lock,R., Sommerville, I.:QoSOnt: a QoS Ontology for Service-Centric Systems . In: Proceedings of the 2005 31st EUROMICRO Conference on Software Engineering and Advanced Applications (EUROMICRO-SEAA2005) (2005)

[36]. Kritikos,K., Plexousakis, D., Semantic QoS Metric Matching. In:Proc.of European Conf. on Web Services, pp.265–274.IEEE Computer (2006)

[37]. Giallonardo, E., Zimeo, E., More Semantics in QoS Matching. In:Proc.of IEEE Intl.Conf.on Service Oriented Computing and Applications, pp.163–171. IEEE Computer (2007)

[38]. Tondello,G.F., Siqueira,F., The QoS-MO Ontology for Semantic QoS Modeling. In:SAC2008, March16-20 (2008)

[39]. Yu,C., Junyi,S., Yang,Y., Zhizhong,L., A QoS Ontology for Semantic Service Discovery. In:2009 International Conference on Networking and Digital Society (2009)

[40]. Jean,S., Losaviot, F., Matteot, A., Levy, N., An Extension of OWL-S with Quality Standards.978-1-4244-4840 2010 IEEE

[41]. Shin, D., Lee, K.: An Automated Composition of Information Web Services based on Functional Semantics. In: Proceedings of the 2007 IEEE Congress on Services (SERVICES 2007). IEEE, Los Alamitos (2007)

[42]. Kona, S., Bansal, A., Blake, M.B., Gupta, G.: Towards a General Framework for Web Service Composition. In: Proceedings of the 2008 IEEE International Conference on Services Computing. IEEE, Los Alamitos (2008)

[43]. C. Petrie, T. Margaria, H. Lausen, M. Zaremba (eds.), *Semantic Web Service Challenge* (Springer, New York, 2009)

[44]. M.Paolucci,T. Kawamura,T. Payne,and K.Sycara. Semantic matching of web services capabilities. In Proc. Of the 1st International Semantic Web Conference (ISWC) , 2002