*Proceedings published by International Journal of Computer Applications® (IJCA)*
*International Conference on Computer Communication and Networks CSI- COMNET-2011*

# SaaS Service Recommendation System Based on SLA Ontology

B.Saravana Balaji
Sri Ramakrishna Engineering College, Coimbatore

N.K.Karthikeyan
Sri Krishna College of Engg & Tech., Coimbatore

S.Mani, P.Bhuvaneswari
Sri Ramakrishna Engineering College, Coimbatore

## ABSTRACT
The emergence of Grid computing and Service Oriented Architectures has led to evolution in terms of how applications are built and managed. Software as a service (SaaS), also known as the on-demand model, is changing the way businesses of all sizes and in all industries use software. These services are provided by different vendors. Some time similar services are provided by the various vendors. Hence to choose the best service is becoming a cumbersome one to the consumer. In this paper we proposed a recommendation system for selecting the best SaaS service according the user requirement by processing the SLA of SaaS services. This system is based on the semantic web technology to populate the services in the service population system and also it has the semantic annotation sub system to generate the annotation for the query generated by user.

## Keywords
Ontology, OWL ,SaaS, SLA,, SW, and XML.

## 1. INTRODUCTION
Cloud Computing solutions including Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), and Infrastructure-as-a-Service (IaaS)[12] are radically changing the way organizations acquire and utilize business applications and other computing resources, and how hardware and software vendors develop, deliver, package, price, sell, and support these solutions. The SaaS model means the end product of software business world. For customers, the benefits are obvious and compelling:

They get sophisticated functionality without up-front expenses or the hassles associated with the installation and maintenance of traditional software. For vendors of such services, the model provides low barriers to entry and unprecedented opportunities, as well as new risks and challenges. Creating and managing a SaaS company demands a new way of running a business—one that extends to all business areas that make up an organization. A service-level agreement (SLA)[5] is nothing more than a type of contract between two parties. In the context of managed IT services (in which SLAs most frequently appear), SLAs dictate the quality and type of service that will be provided to the client in exchange for a fee. SLAs also provide the remedy, such as a reduced fee structure, that will apply in the case of a service outage. The SLA is developed in accordance with a SLA template which contains scope, terms and validity. The scope descries the nature of service and covers the domain of service. The terms consists of several parameters such as Qos, Cost factor, legal part, Failure of service issue resolution, Penalty charge and help desk support details. The validity provides the SLA validation, in which the SLA expiration periods are specified. Here we considered the SLA description which indicates the type of service the service provider offers and we considered the Qos terms, it consists of response time, availability, throughput and latency, and these parameters are measured as the overall performance of the service. The evolution of the cloud computing leads us to the utility computing, in which several applications are being developed and offered to various domains. email application service, multimedia application service, e-business and web application service are some of the popular services provided by the several vendors. Hence the complexity in finding the better service is becomes a tedious job for the end user. Keeping this in mind, we have proposed a semantic recommendation system to select accurate and better service. Our proposal is not a traditional search engine, but a semantic one. Our solution is inspired by the Semantic Web (SW) approach, whose main challenge is to enable better machine information processing. These technologies can help users with the proper support to take advantage of the URL information available on the UDDI. In the Semantic Web, knowledge is represented by means of ontologies[3][11], which are viewed in this work as a formal specification of a domain knowledge conceptualization[2].

The remainder of this paper is organized as follows. In Section 2, the fundamentals of knowledge representation and ontology population are briefly described. The proposed architecture of the service recommendation system for service selection is shown in Section 3. Section 4 describes about the evaluation of the system. Finally, our conclusion and future works are outlined in Section 5 and 6 respectively.

## 2. BACKGROUND
### 2.1 Towards Semantic Search Engines
Semantic search engines differ from traditional ones in two main key respects : (1) they use a logical framework that makes more intelligent retrieval possible; (2) the management of complex semantic relationships makes the meta-data maintenance harder though it favors more sophisticated ranking mechanisms[16]. The traditional search engines, as Google or Yahoo are constantly building indexes so they create ties between words and documents, so that when a user submits a query, the search engines return its related documents. The result is a large set of documents, in most cases ‘‘shorted’’ (i.e. sorted and shortened) by an algorithm such as Page Rank[18][9]. The engine does not understand the meaning of the query, so the results include all the possible alternatives. The quality of the results can be improved by categorizing Knowledge entities, therefore semantic search engines obtain better results because

they understand the query meaning. Consequently their accuracy is higher.

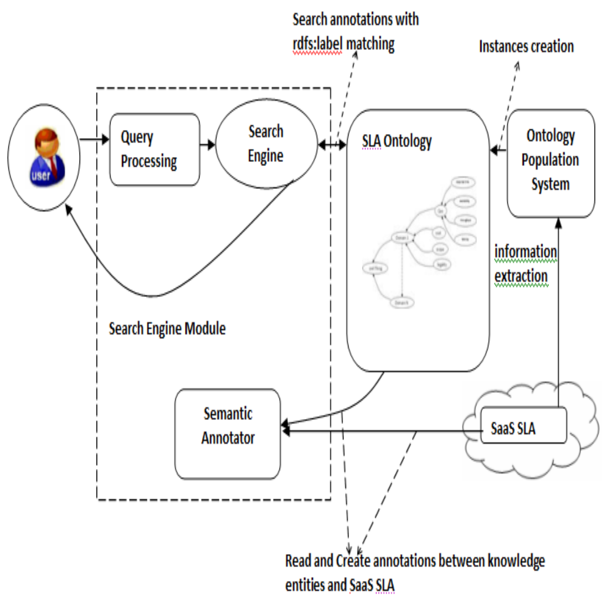


**Fig.1 System Architecture**

A fundamental prerequisite of the Semantic Web is the existence of large amounts of meaningfully interlinked RDF/OWL data on the Web. RDF is a data model for information representation and OWL is the Web Ontology Language used for publishing and sharing explicit and common descriptions of domain knowledge, and providing support for efficient knowledge management. Both representations are W3C recommendations for modeling ontologies in the SW.

There are four categories of semantic search engines according to their user interface

- Form-based, engines which provide sophisticated web forms that allow users to specify queries by selecting ontologies, classes, properties, and values.
- RDF-based query languages, which provide sophisticated querying languages to support semantic search.
- Semantic-based keywords, which increase the performance of traditional keyword search techniques by making use of available semantic data.
- Question answering tools, which exploit available semantic markup to answer questions asked in natural language [6] format.

The system that we present in this paper is a semantic-based service recommendation system, because the system uses SLA ontology as kernel of its processing system.

## 2.2 Uses of Ontologies in Semantic Web Technologies

The formal semantics underlying ontology languages enables the automatic processing of the information and allows the use of semantic reasoners to infer new knowledge[13]. Ontologies provide a formal, structured knowledge representation, with the advantage of being reusable and shareable. Ontologies provide a common vocabulary for a domain and define with different levels of formality, the meaning of the terms and the relations between them[19]. Ontologies also provide the meaning and facilitate the efficient retrieval of contents and information as well as improving crawling. Knowledge in ontologies is mainly formalized using five kinds of components: classes, relations, functions, axioms and instances. Classes in the ontology are usually organized into taxonomies. In this work, the Ontology Web Language (OWL), which is the Semantic Web standard language, has been used to represent the knowledge extracted from SaaS service SLAs.

## 2.3 Ontology Population Method

Ontology population, is a knowledge acquisition activity that relies on (semi-) automatic methods to transform unstructured, semi-structured and structured data sources into instance data. In other words, Ontology population pursues the extraction and classification of instances of the concepts and relationships defined in the ontology[7]. The instantiation of the ontology with new knowledge is a significant step towards the provision of valuable ontology based knowledge services.

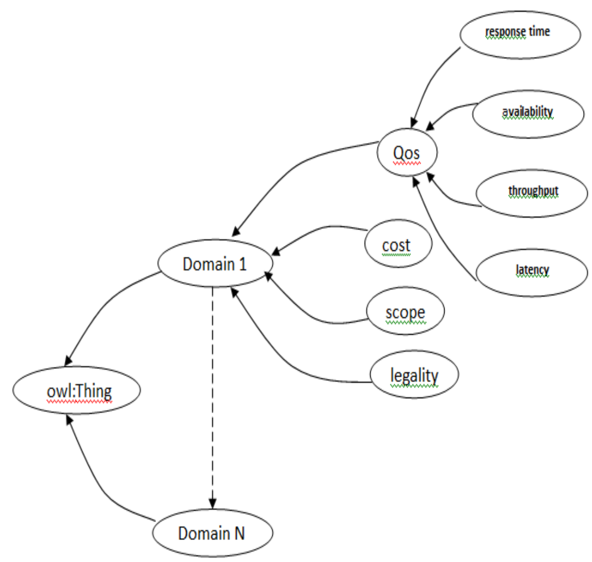


**Fig.2 SLA Ontology**

We can distinguish two types of ontology population: (i) from free text, and (ii) from semi-structured documents such as XML, HTML, RSS, etc. However, in this work we have developed a semi-automatic method for ontology population from semi-structured texts. Most SaaS SLAs are provided as semi-structured or unstructured HTML or XML documents Most of the SaaS SLAs are provided as semi-structured, which are transformed into semantic annotations[15]. There are different approaches are for populating ontologies from semi-structured or unstructured XML documents. For example, in the work presented in [8], an ontology is populated using RDF triples[20] obtained from XML documents. XML documents are obtained from a SaaS Service SLA's parameters and metrics, which are processed using XML wrappers based on predefined patterns.

# 3. SERVICE RECOMMENDATION SYSTEM ARCHITECTURE

The proposed architecture for the Service recommendation system consists of three main modules (see Fig. 1). The SLA ontology, the ontology population module, and the ontology-based service search engine module. We will now look into a detailed explanation of these models.

## 3.1 Service Level Agreement (SLA) Ontology

We have developed a SLA ontology based on the SLA constraints and its parameters. This ontology has 14 classes, 16 subclass axioms, 16 data type properties, 8 object 41 properties and 27 restrictions. The ontology covers four main SLA concepts (see Fig. 2).

### *3.1.1 Quality-of-Service*

QoS factors involves in finding the performance of the service, the Qos terms consists of a few valid parameters such as response time, service availability, throughput and latency etc. With these information the performance degree and level of the service is calculated as one of the knowledge entities.

### *3.1.2 Cost of the Service*

The entities that typically represent the amount charged for the service of specific period of time. The service selection could be carried out based on cost factor.

### *3.1.3 Scope*

This knowledge entity represents the scope of the e service and also describes the business objectives which the service intent to offers. Apart from all the above concepts, the SLA ontology covers various domains under which the service classification and their relationships are categorized.

## 3.2 Ontology population from SLAs

The ontology population system [8][17] gathers knowledge from SLAs which are generally developed by using the XML tag and validated through XML Schema. The ultimate goal of our approach is to populate the SLA ontology with all the relevant information identified. The populated ontology will then serve as the keystone component for an up-to-date, service-based recommendation system. The architecture of the proposed ontology population system is shown in Fig. 3. It is composed of two main components: (i) a set of Selection Parsing and Conversion Systems (SPCS) (ii) the Ontology Population Algorithm (OPA) module. The input of the system is represented at the top of Fig. 3. It consists of a collection of registered-available information resources in the SaaS Service Registry. The output of this module is a number of ontology instances that are stored in the repository called Knowledge Base.

As a whole, the system works as follows. The SLAs available on the registered service provider’s URL are parsed to extract the information that can be gathered from the XML tags. Users are shown the parts of the extracted information identified by the parser. Such that users must choose which of the found elements are relevant and have to be stored in the knowledge base. Users have to set up two parameters: (i) a set of transformation rules, which will be used by the SPCS module to transform the information into the appropriate format, and, (ii) the set of ontological concepts that are related to the information elements to be gathered from the XML documents. It explores the leading URLs concerning the registered service Provider in the SaaS Service Registry to populate the SLA ontology. For this purpose, the aforementioned user-defined transformation rules are applied. During this process, the position of the information in the XML node is taken into account to form domains. Each domain is represented in the form of tuples (attribute, literal). The Result produced by the SPCS and the set of ontology concepts pointed out by the user are the input of the OPA module. With this information, OPA generates the correspondences between the data in the XML documents and the concepts in the ontology. Finally, the new discovered ontology instances[1] are stored in the knowledge base.

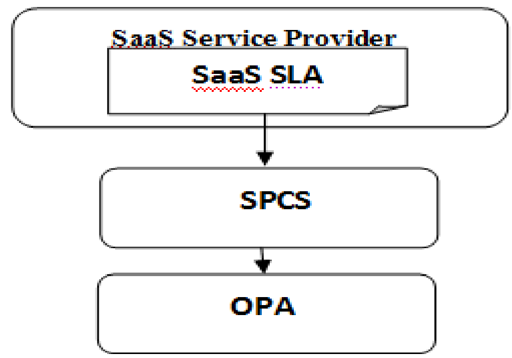


**Fig.3 Ontology Population System**

## 3.3 Ontology-Based Service Search Engine

The Service search engine has been divided into three modules (see Fig. 1): (i) Semantic Annotator, (ii) Query Processing, and (iii) Search Engine. Semantic Annotator: In this module, SLA description parameters and metrics are annotated to semantic concepts by using the domain knowledge conceptualization[2][14]. The process that takes place during the semantic annotation is as follows. First, the most important linguistic expressions are identified using statistical approaches based on term extraction methods. Then, for each linguistic expression, the system tries to determine whether the expression under question is an individual of any of the classes of the domain ontology. Next, the system retrieves all the annotated knowledge that is situated next to the current linguistic expression in the text, and tries to create fully filled annotations with this knowledge.

Query Processing: Users can query the system through natural language queries[6][13]. For this, four main steps are carried out. First, a POS-Tagging process is performed. This allows the system to identify the grammar category of each word in the sentence and removes the non-content words. Then, the system identifies the lemma of each word by means of a lemmatizing process. A chunking and name entity recognition process is performed in order to obtain the focus of the query. Finally, the synonyms related to the SLA domain are listed.

Search Engine: In OWL-based ontologies, the rdfs:label is an instance of rdf:property that may be used to provide a human readable version of a resource name. In this work, all the resources in the ontology have been annotated with the label descriptor. The main objective of this module is to identify the service information that is related to the expanded query obtained from the Query Processing module and to sort these results. The sorting function is based on semantic similarity function[4] and indexing functions in order to obtain the most related as well as first top most services from the user query. The system is constantly crawling service information from SLAs with the Annotator Module, and it generates Semantic annotations for the each SLA. If there is no annotations have been created for any template then the service information is not stored in the knowledge base.

The SLA ontology is continuously populated, so the annotation process is executed periodically. The user queries the system through the Query Processing module and the semantic information of the query is then passed to the Search Engine which is in charge of obtaining the service information that is semantically related to the named entity reorganization[10] obtained from the query.

# 4. EVALUATION OF SYSTEM

In the SaaS service recommendation system, Client interface is developed based on Web technique, and it adopts XHTML, JavaScript, and JSP. Jena API is used to implement semantic analysis and relevant item confirmation processing, because the information is stored as OWL (Web Ontology Language) type ontology files. OWL type ontology files are stored in ontology base.

## 4.1 Establishment of Ontology

We adopt Skeletal Methodology to establish the SLA ontology of SaaS services; it contains three steps; which are as follows.

- Confirm the Service domain of ontology.
- Create SLA ontology & Population.
- Ontology maintaining.

## 4.2 Semantic Analysis of Query Request

Query processing module is responsible for converting users keywords into the conceptions of ontology, and inferring new indexing keywords from the selected domain. It is common that different users input different keywords to describe the same object. So it is important to use ontology based approach to understand the keywords in nature language. According to users query keywords, indexing domain, and the relationship between ontologism, the service search engine can query ontology model, which is a RDF (Resource Description Framework) triple. Here, we adopt RDQL (RDF Data Query Language) as query language. RDQL is a query language for RDF in Jena models. RDF provides a graph with directed edges — the nodes are resources or literals.

## 4.3 Experimental Results

Table 1 shows the simple statistics on the gathered data set that is used for the experimental evaluation of the system. Table 2 summarizes the performance of the proposed SaaS service recommendation system. ‘Random Selection’ in this evaluation is to choose the target service at random.

# 5. CONCLUSION

The approach followed by traditional search engines suffers from practical limitations. First, the ever-changing nature of technology prevents users to select the appropriate and best service from various service providers. This problem can be partially overcome by developing category-specific semantic search engines. A further limitation of traditional search engines is that simple keyword-based queries often return vast amount of irrelevant information which produces the low precision and recall problem. Besides, keyword-based search engines present other serious problems such as language ambiguity (synonymy and polysemy).

All of this leads us to the need for better search strategies. Semantic Search Engines are a relatively recent phenomenon, supported by the Semantic Web trend. Semantic Search is a process, which is logic-based underpinning of the Semantic Web which enables the intelligent retrieval of data and helps in dealing with semantic heterogeneity. In this work, we propose a semantic service recommendation system specially suited for selecting best and appropriate service. We have focused on this area due to the demand in selecting better service, and therefore better utilization and accuracy of service is obtained.

**Table1. Statistics on the data set used in the experiment**

| **Data Set** | **Value** |
|---|---|
| No of SLA Documents(XML) | 341 |
| No of Domains | 18 |
| Average SLA per Domain | 18.9 |

**Table 2. Experimental Results of the System**

| **Method** | **Accuracy** |
|---|---|
| Random Selection | 54% |
| SaaS Service Recommendation System | 79.3% |

# 6. FUTURE WORK

In this paper we considered the SLAs are in the form of XML documents, However the SLAs are defined in various formats such as they are in Word document, EXCEL document and sometimes in text format, there for these problems throughout the course of these developments, has remained unsolved. And another problem which is unsolved is that the automatic ontology construction. Several methodologies have been developed to assist in building ontologies. Yet, the manual construction of ontologies is still considered a major bottleneck. Furthermore, this approach can be upgraded to different cloud domains like PaaS and IaaS.

## 7. REFERENCES


[1] Gruber, T. R. (1993). A translation approach to portable ontology specifications.Knowledge Acquisition, 5(2), 199–220.

[2] Studer, R., Benjamins, V. R., & Fensel, D. (1998). Knowledge engineering: principles and methods. Data & Knowledge Engineering, 25(1–2), 161–197.

[3] Dieter Fensel and Frank van Harmelen, Vrije Universiteit and Amsterdam. 2001 'An Ontology Infrastructure for the Semantic Web', IEEE Transactions on Knowledge and Data Engineering Vol.19 No.2 pp.261–272.

[4] Maedche, A., & Staab, S. (2002). Measuring similarity between ontologies. In EKAW '02: Proceedings of the 13th international conference on knowledge engineering and knowledge management. Ontologies and the Semantic Web (pp. 251–263). London, UK: Springer-Verlag.

[5] Lamanna, D.D., Skene, J., Emmerich, W.: Slang 2003 'A language for defining service level agreements'.

[6] Ekaterina Buyko, Christian Chiarcos, Antonio Pareja Lora. 2004 'Ontology-Based Interface Specifications for an NLP Pipeline Architecture', IEEE Transactions on Knowledge and Data Engineering Vol.15, No.2 pp.261–272.

[7] Delen, D., Pratt, D. B., 2006. An integrated and intelligent DSS for manufacturing systems. Expert Systems with Applications. 30, 325–336.

[8] Alain-Pierre Manine, Erick Alphonse, Philippe Bessi`eres 2006 'Information Extraction as an Ontology Population Task and its Application to Generic Interactions', MIG, INRA UR, Vol. 1, No.3, pp.1077.

[9] Langville, A. N., & Meyer, C. D. (2006). Google's PageRank and beyond: The science of search engine rankings. Princeton University Press.

[10] Park.J and Kim. Text Mining for Biology, chapter Named Entity Recognition. Artech House Books, 2006.

[11] Dejing Dou, Drew McDermott, and Peishen Qi 2007 'Ontology Translation on the Semantic Web', Journal on Data Semantics II, Vol. 7, No. 2, pp 33-60.

[12] Foster, I., Zhao, Y., Raicu, I., Lu, S. 2008, 'Cloud Computing and Grid Computing 360-Degree Compared' In: Grid Computing Environments Workshop, 2008. GCE'08.

[13] Ruiz-Martínez, J. M., Castellanos-Nieves, D., Valencia-García, R., Fernández-Breis, J. T., García-Sánchez, F., Vivancos-Vicente, P. J., Castejón-Garrido, J. S., Bosco-Camón, J., & Martínez-Béjar, R. (2008a). Accessing touristic knowledge bases through a natural language interface. In: Proc. PKAM, Hanoi Vietnam (pp. 147– 160).

[14] Ruiz-Martínez, J. M., Miñarro-Giménez, J. A., Guillén-Cárceles, L., Castellanos- Nieves, D., Valencia-García, R., García-Sánchez, F., Fernández-Bréis, J. T., & Martínez-Béjar, R. (2008b). Populating ontologies in the etourism domain. In: Proc. international conference on web intelligence and intelligent agent technology, IEEE/WIC/ACM, Sydney, Australia (Vol. 3, pp. 316–319).

[15] Tao, C. and Embley, D. W. 2009 'Automatic hidden-web table interpretation, conceptualization, and semantic annotation', Data and Knowledge Engineering Vol. 68 No.7 pp. 683–703.

[16] Riad,Hamdy, A. M. Elminir, K. Mohamed Abu ElSoud and Sahar, Sabbeh, F. 2009 'Architecture for a Personalized Semantic Search Engine', Journal on Data Semantics II, Vol. 2, No.1.9, pp.204.

[17] García-Manotas, I., Lupiani, E., García-Sánchez, F., & Valencia-García, R. (2010). Populating knowledge based decision support systems. International Journal of Decision Support System Technology (IJDSST), 2, 1–20.

[18] Miriam Fernández, A. Ivan Cantador, B. Vanesa Lopez, A. David Vallet, B. Pablo Castells, B and Enrico Motta, A. 2010 'Semantically enhanced Information Retrieval: An ontology-based approach', Web Semantics Science, Services and Agents on the World Wide Web- Elsevier Vol. 30, pp 897-913.

[19] Ronald Denaux, A. Catherine Dolbear, C. Glen Hart, B. Vania Dimitrova, A. and Anthony G. Cohn 2011 'A Supporting domain experts to construct conceptual ontologies', Web Semantics: Science, Services and Agents on the World Wide Web - Elsevier Vol.30.

[20] http://www.w3.org/RDF/.