Enhancements for UDDI using User Preferential Web Service Selection Model based on SLA

N. Balaji*, S.R. Murugaiyan[$], M.S. SaleemBasha*, R. Baskaran[#] and P. Dhavachelvan*

*Department of Computer Science, Pondicherry University, Pondicherry, India
[$]Research Scholar, Department of Computer Science & Engineering, Manonmaiam Sundaranar University, Tamil Nadu, India.
[#]Department of Computer Science and Engineering, Anna University, Chennai, India.
{nbalajime1983, murugaiyansr, m.s.saleembasha}@gmail.com, baaski@annauniv.edu, dhavachelvan@gmail.com

***Abstract*—As we know daily thousands of new service providers register their services in the UDDI. The number of service providers providing the same functionality increases day by day. So it is very difficult for the service requester to choose the best service provider for their requirements from the UDDI. Hence web service selection plays a huge role in providing the best web services to the service requester's. Most contemporary web service selection research is based on the functional and non-functional attributes of web services. In this paper, we have successfully demonstrated a more efficient web service selection model based on user preferences. In this model we have implemented a new web service selection approach based on the user preferred QoS. This proposed model demonstrates in much more effective web service selection compare to the present approaches. We have also successfully shown the difference between the two models.**

Key words-UDDI, Web Services, Service Discovery, Service Selection, QoS, User Preferences

I. INTRODUCTION

Web services are depending on the idea of service-oriented architecture (SOA). The modules of web services are XML, SOAP, UDDI and WSDL [1]. XML stands for (eXtensible Markup Language) it was designed to describe data and tags are not predefined you must define your own tag. SOAP (Simple Object Access Protocol) can be used over any transport protocol such as HEML, SMPT or even TCP. UDDI (Universal Description Discovery and Integration) is a directory for storing information about web services and directory of web service interfaces described by WSDL. WSDL (Web Service Description Language) is an XML vocabulary for describing Web services. It allows developers to describe Web Services and their capabilities, in a standard manner.

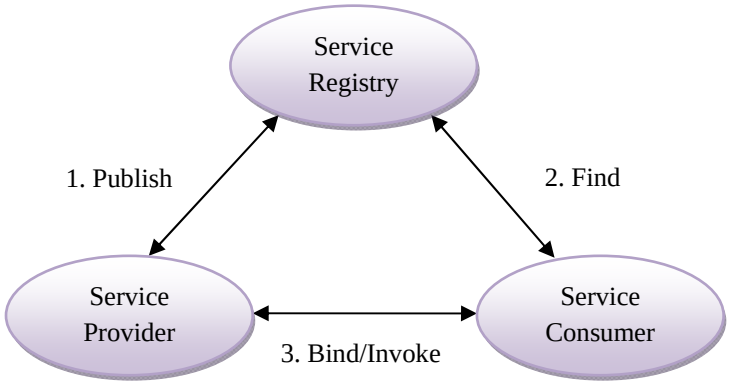


Fig. 1. Operations in Web Service Architecture

IBM has published its web services architecture [2] which captures the infrastructure required to support web services. The Web Services architecture is based upon the interactions between three roles: they are service provider, service consumer and service registry as shown in fig1.The interactions involved thee operations: they are publish operations, find operations and bind operations. A service provider to register the services in the service registry. A service consumer to discover the WSDL via the service registry. A service registry provides a central place where developers can publish new services or find existing is used to bind. UDDI (Universal Description, Discovery, and Integration) defines a Standardized model for service registries (OASIS, 2004). UDDI is an industry initiative that is working to enable businesses to quickly, easily, and dynamically find and

transact with one another. UDDI enables a business to describe its business and its services, discover other businesses that offer desired services, and integrate with these other businesses [4]. The goal of both is cross-platform searching and capabilities. UDDI is an OASIS specification. QoS can be seen as an aggregated evaluate of common requirements such as cost, response time, reliability, scalability, availability, reputation, security, performance, and throughput. Currently various authors proposed QoS parameters, verification mechanisms and measurement metrics based on QoS models [3, 5, 6].

II. Related Works

The authors [7], provides UDDI Version 2 API Specification, in the discovery interface it provides no QoS related inquiry. Service requesters cannot sort out the unqualified service nor can they get and compare between different services without testing them first. Multi-agent framework [8, 10, 21]. The authors [9] proposed an extended UDDI registry for web services. The UDDI offer the QoS support discovery and lifetime organize for UDDI. The author to resolve this difficulty, some effort has been complete through to enhance the UDDI registry's inquiry/publish interface to insert the QoS information in the communication. For example, UDDI task is under attack for the most part towards the QoS-supported interface enrichment for UDDI. UDDI extends the UDDI registry to support searching on quality of a service and extend the discover process to allow queries for UDDI with numerical and logical ranges. The extended UDDI API is provided through the definition of QoS attributes based on the QoS management. Keyword spotting techniques [16]. The service provider to register the services with QoS information's, the service provider can provide random QoS attributes with selected lease for a web service. Distributed Framework [11, 13, 19]. E-Learning Management System in e-services for the web services environment [14, 15]. The authors [12] proposed the semantic web in UDDI. The centralized UDDI registry to focus on discovery of web services. The centralized UDDI can offer efficient technique to discover the Web services, but this process suffer from a few troubles related with having centralized UDDI systems such as a constrained right to use and particular position of failure. P2P networks [17, 23]. The authors [18] proposed the S-QoS4WS. In this approach which formulate use of existing methods within UDDI version 3. By using service provider published QoS declarations,' the authors newly establish third party customer validations and service certifications. Service requesters of web services can trust that the services discovered within a UDDI will meet their all requirements. In this method to resolve current issues involving trust and non-repudiation. The authors [20] proposed integrate web service registries with Web Service Quality Management Systems. These model complete three profits. First, service providers registered the web services with QoS information in the service repository, so service providers to reduce their effort to test and measure web service performance and monitor the performance continuously. The service requesters can get WSDL and WSQDL data with single access to a registry or a WSQMS and determine whether they can use the web service. Secondly, service requester to get more trustworthy web service from the service repository, because when a web service is published in a registry, test and evaluation for it is performed at the same time. Web service provider will try to improve the quality of his web service. Lastly, the incorporation add to activate Web service and its distribution. Population seeding technique [22, 30]. Cloud Service [24, 26, 29]. The authors [25] proposed customize private UDDI registry. In this method to implement query results based on business requirements such as ranking of service list. All the models proposed are designed to suite different practical needs of private registry systems. QoS management [27, 31]. The authors [28] proposed (Q-WSEM) approach; user preferred QoS based Web Service Evaluation Model. They implement a prototype to enhance function of UDDI register. This model is not only meet functional properties it also meet user preferred QoS properties. The QoS-Web Service Evaluation Model approach evaluating results are more efficiently.

III. UDDI Registry

UDDI Registry is a web service repository for storing information about web services and directory of web services interfaces described by web service description language. The service provider will publish the web services based on the QoS is stored in web service repository and the service requester request the services based on QoS from the web service repository [32].

In this paper, we have proposed a web service selection model based on the user preferred QoS. This model is implemented using a 2 Tier User Preferential technique. In the 2 Tier User Preferential Model: Tier I to select the QoS parameters and Tier II to set the range. To demonstrate our model we have designed our own UDDI in java where about 1000 web service providers have registered their services. Each web service stored in the UDDI are allocated different service id and sorted based on their functionality provided. Two different search techniques are used in our approach. First one is (Web Service Search) keyword based web search, second one is (New Web Service Search) semantic based web service search. The data retrieved from these semantic based web service searches are further refined using our 2 Tier user preferential technique. Using the 2 tier user preferential model service requesters are able to achieve better results by filtering out services that provide services in terms to their QoS requirements.

IV. PROPOSED APPROACH

The Fig 2.User Preferential Web Service Selection Model will provide the best web service selection to the service requester. The procedure for using web service selection model. The procedure for service provider based on QoS states the step by step execution of storing a web service in QoS database using web service repository. The service requester based on requirements procedure states the selection of web service using service selection model from discovery. This proposed model demonstrates in much more effective web service selection compare to the present approaches. We have also successfully shown the difference between the two models.

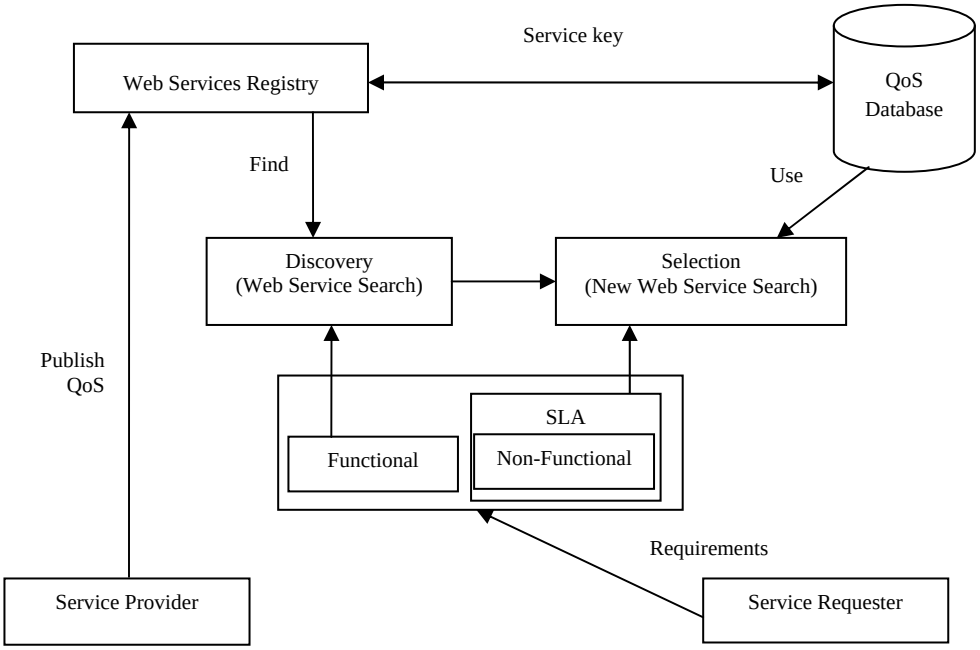


Fig. 2. User Preferential Web Service Selection Model

Web Service Repository: The service repository is a database to store the service providers published services with QoS information. Web Service Provider: The Service provider will publish the web services in the web services repository with QoS. Web Service Requester: The service requester will request the functional, non-functional and SLA based web services. QoS Database: It is a database of QoS information correlative with web services repository by the Service Key. QoS: The Quality of Service is based on non-functional aspect of web service which will distinguish to get a better service. SLA: Service Level Agreement (SLA) is a service contract on the quality of a service (Qos) between a service provider and a service requester. Discovery: The Discovery is a process where the service requesters request a service will be handled by the repository to find appropriate service to the requester. Selection: The initially discovered web service sets are stored in Qos database as Qos information. From the QoS database a web service set is selected based on the service requester's QoS and SLA constraints.

V. EXPERIMENTAL SETUP AND RESULT ANALYSIS

An UDDI is implemented and 1000 web services were registered in it. The service provider will register the web services with Quality of Services Parameters and SLA based QoS in UDDI registry. The service requester request the services based on QoS and SLA Parameters. Two different search techniques are used in our approach. First one is (Web Service Search) keyword based web search, second one is (New Web Service Search) semantic based web service search. The data retrieved from these semantic based web service searches are further refined using our 2 Tier user preferential technique. Using the 2 tier user preferential model service requesters are able to achieve better results by filtering out services that provide services in terms to their QoS requirements.

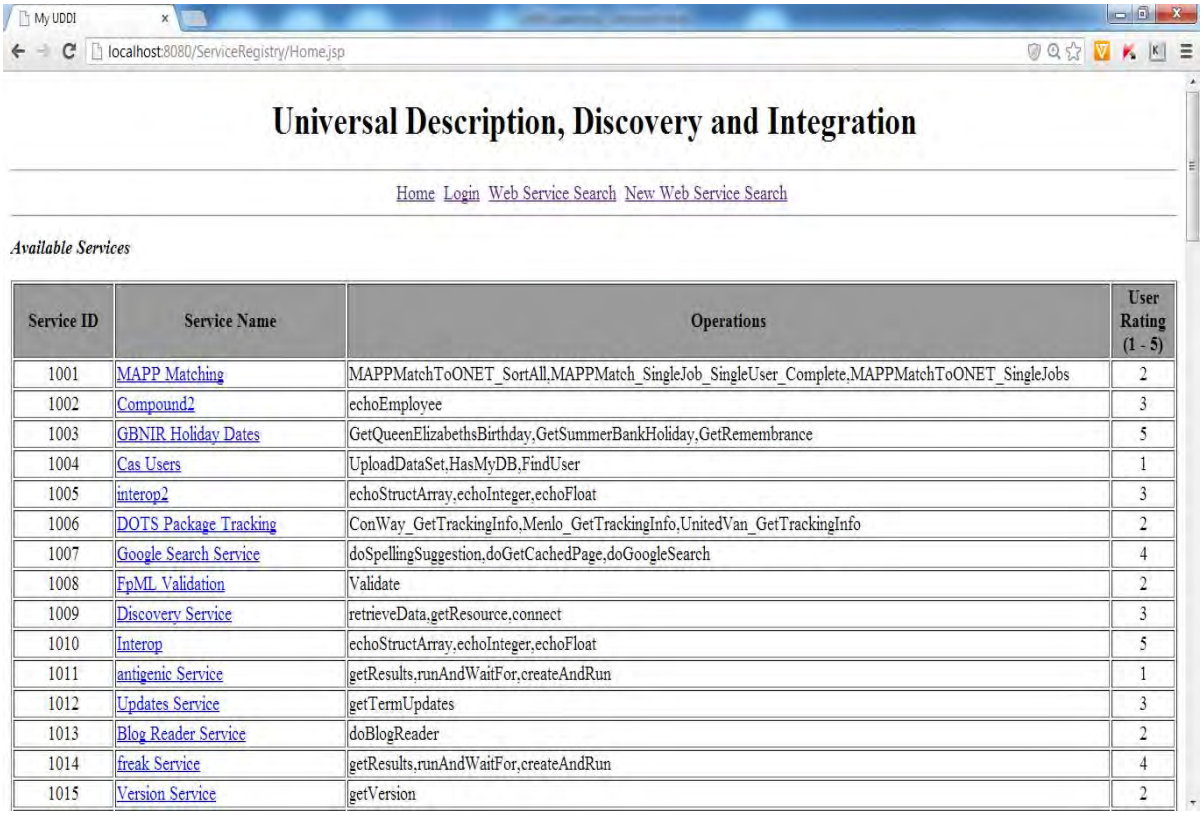


Fig. 3. Available Services Home page

Fig 3.shows the various web services providers registered 1000 web services in our UDDI with their Service ID, Service Name, Operation type and User Rating.

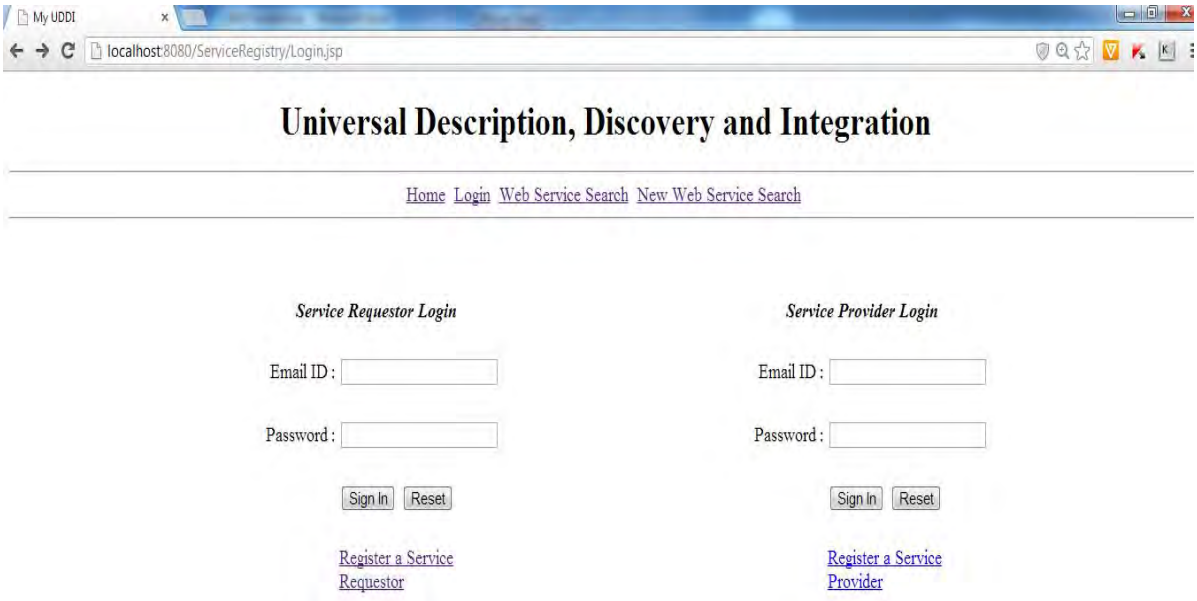
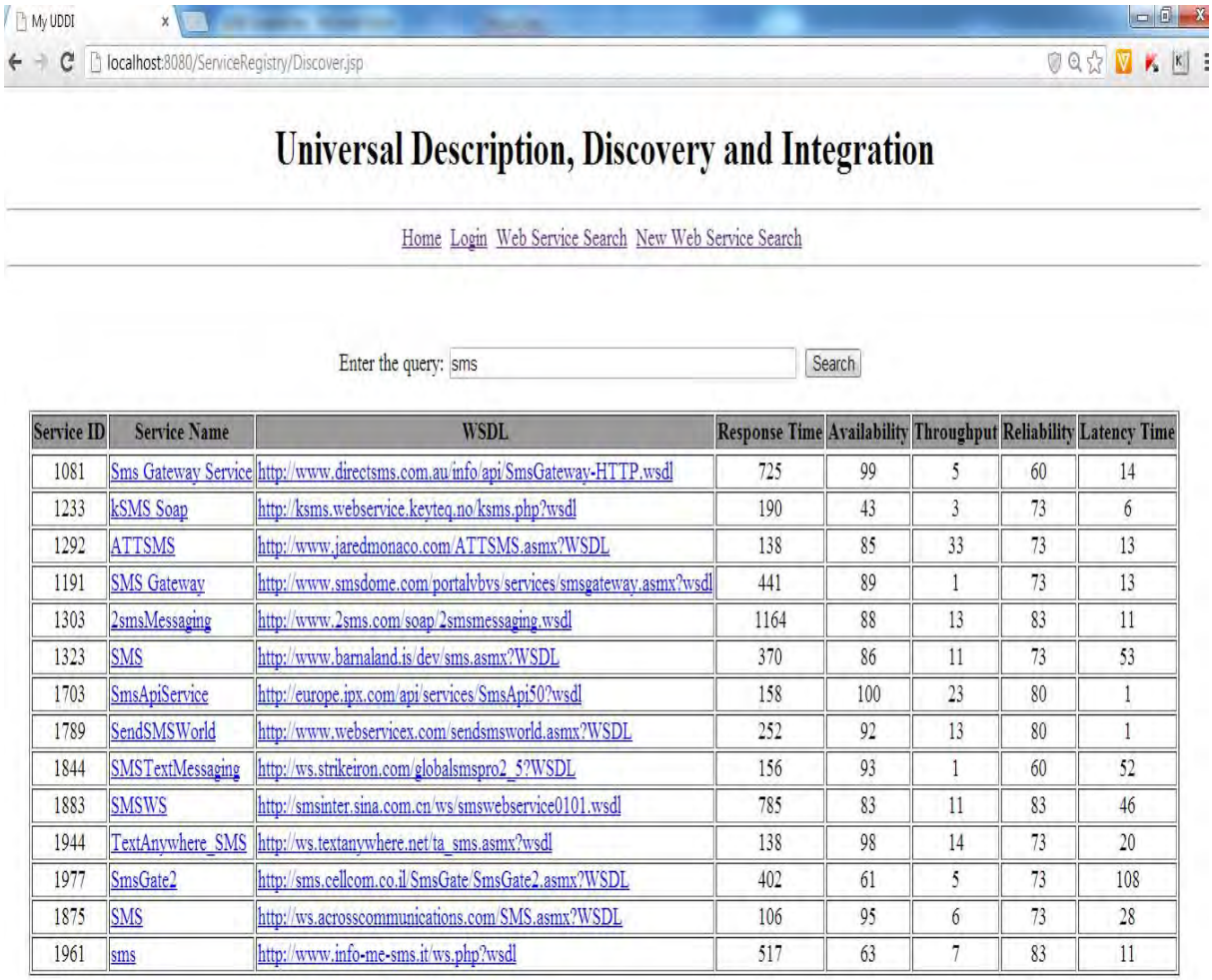


Fig. 4. Login Page

Fig. 4. Shows the login page for service requester and service provider.

Fig. 5. Service Requester Search page

The service requester click on Web Service Search the Fig 5. shows the welcome screen for the Service Requesters. Using the search tab service requesters can search various services from our UDDI. It is Keyword based web service search.



Service ID	Service Name	WSDL	Response Time	Availability	Throughput	Reliability	Latency Time
1081	Sms Gateway Service	http://www.directsms.com.au/info/api/SmsGateway-HTTP.wsdl	725	99	5	60	14
1233	kSMS Soap	http://ksms.webservice.keyteq.no/ksms.php?wsdl	190	43	3	73	6
1292	ATTSMS	http://www.jaredmonaco.com/ATTSMS.asmx?WSDL	138	85	33	73	13
1191	SMS Gateway	http://www.smsdome.com/portalvbvs/services/smsgateway.asmx?wsdl	441	89	1	73	13
1303	2smsMessaging	http://www.2sms.com/soap/2smsmessaging.wsdl	1164	88	13	83	11
1323	SMS	http://www.barnaland.is/dev/sms.asmx?WSDL	370	86	11	73	53
1703	SmsApiService	http://europe.ipx.com/api/services/SmsApi50?wsdl	158	100	23	80	1
1789	SendSMSWorld	http://www.webservicex.com/sendsmsworld.asmx?WSDL	252	92	13	80	1
1844	SMSTextMessaging	http://ws.strikeiron.com/globalsmspro2_5?WSDL	156	93	1	60	52
1883	SMSWS	http://smsinter.sina.com.cn/ws/smswebservice0101.wsdl	785	83	11	83	46
1944	TextAnywhere_SMS	http://ws.textanywhere.net/ta_sms.asmx?wsdl	138	98	14	73	20
1977	SmsGate2	http://sms.cellcom.co.il/SmsGate/SmsGate2.asmx?WSDL	402	61	5	73	108
1875	SMS	http://ws.acrosscommunications.com/SMS.asmx?WSDL	106	95	6	73	28
1961	sms	http://www.info-me-sms.it/ws.php?wsdl	517	63	7	83	11

Fig. 6. Search Results based on Keyword Algorithm

TABLE I
Results obtained Keyword based on the User Query

Service Name	Response Time	Availability	Throughput	Reliability	Latency Time
Sms Gateway Service	725	99	5	60	14
kSMS Soap	190	43	3	73	6
ATTSMS	138	85	33	73	13
SMS Gateway	441	89	1	73	13
2smsMessaging	1164	88	13	83	11
SMS	370	86	11	73	53
SmsApiServie	158	100	23	80	1
sendSMSworld	252	92	13	80	1
SMSTextMessaging	156	93	1	60	52
SMSWS	785	83	11	83	46
TextAnywher_SMS	138	98	14	73	20
SmsGate2	402	61	5	73	108
SMS	106	95	6	73	28
sms	517	63	7	83	11

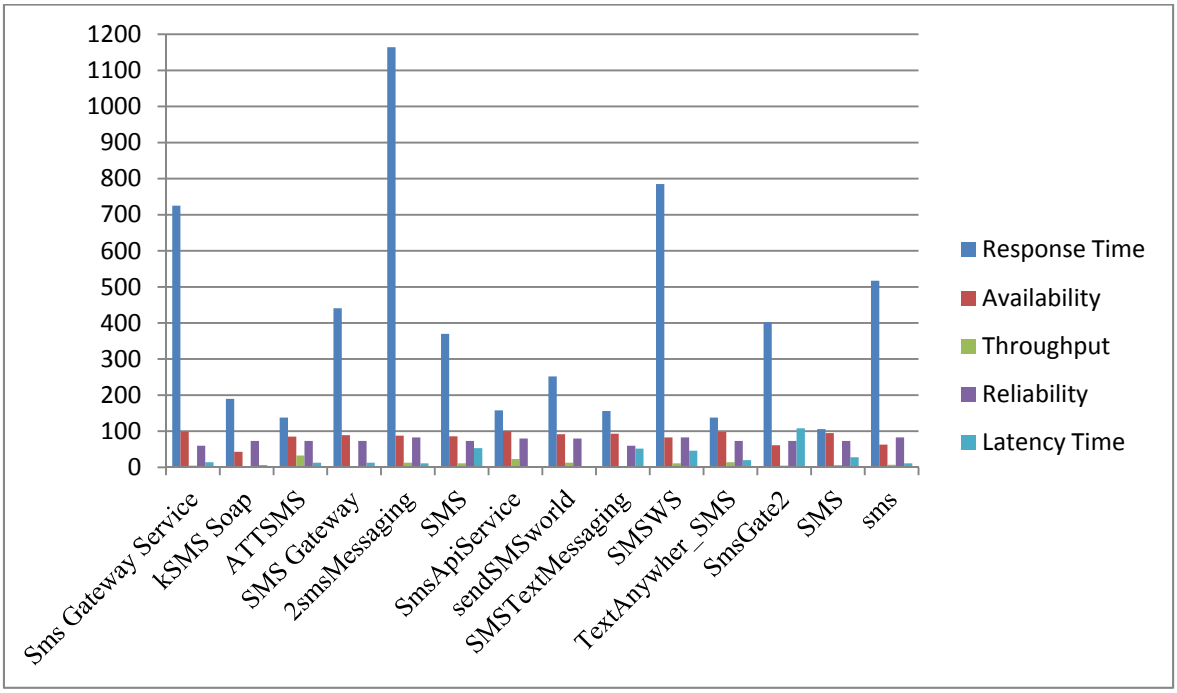


Fig. 7. Keyword based Service Discovered (S_D) with QoS_P Web Services

To explain the model let's assume the service requester searches for "SMS" service in the search tab. We first retrieve the services related to the searched query using Keyword Based web service selection algorithm. Table I and Fig 7. shows the search results of keyword based algorithm. The result shows 14 services with "SMS" keyword from our UDDI.

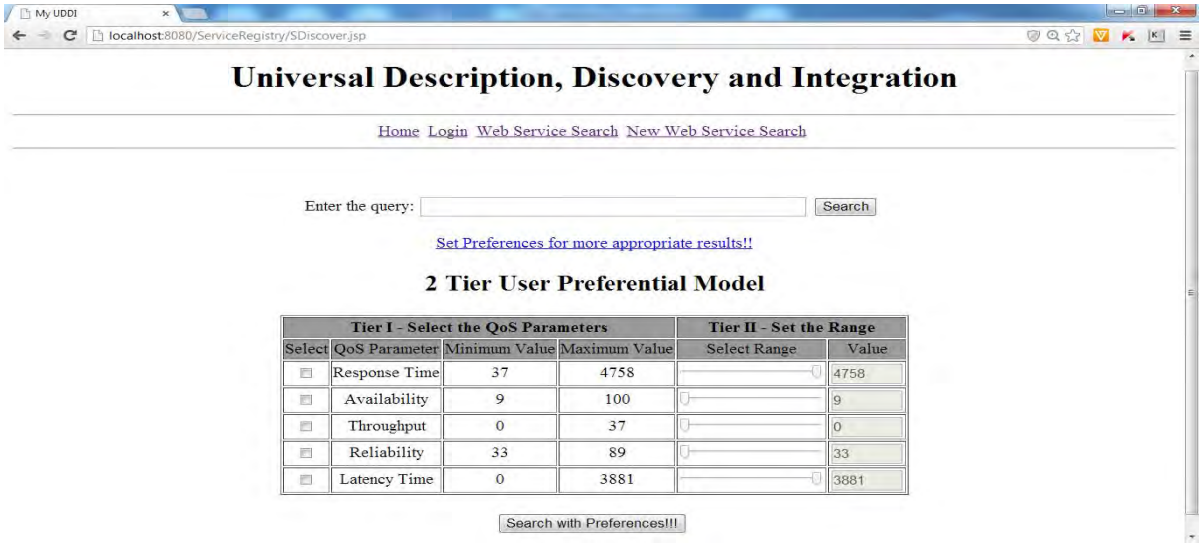


Fig. 8. 2 Tier User preferential Model

The service requesters click on New Web Service Search the Fig 8.shows the page to set user preferred QoS values. Using the search tab service requesters can search various services from our UDDI. It is Semantic based web service search. As per our proposed architecture we refine this search results by implementing a 2 tier user preferential Model. In this model user can mention their own preferred QoS values for the given set of QoS parameters. Users can choose between the minimum and maximum value range for each QoS parameter for the services retrieved from the UDDI.

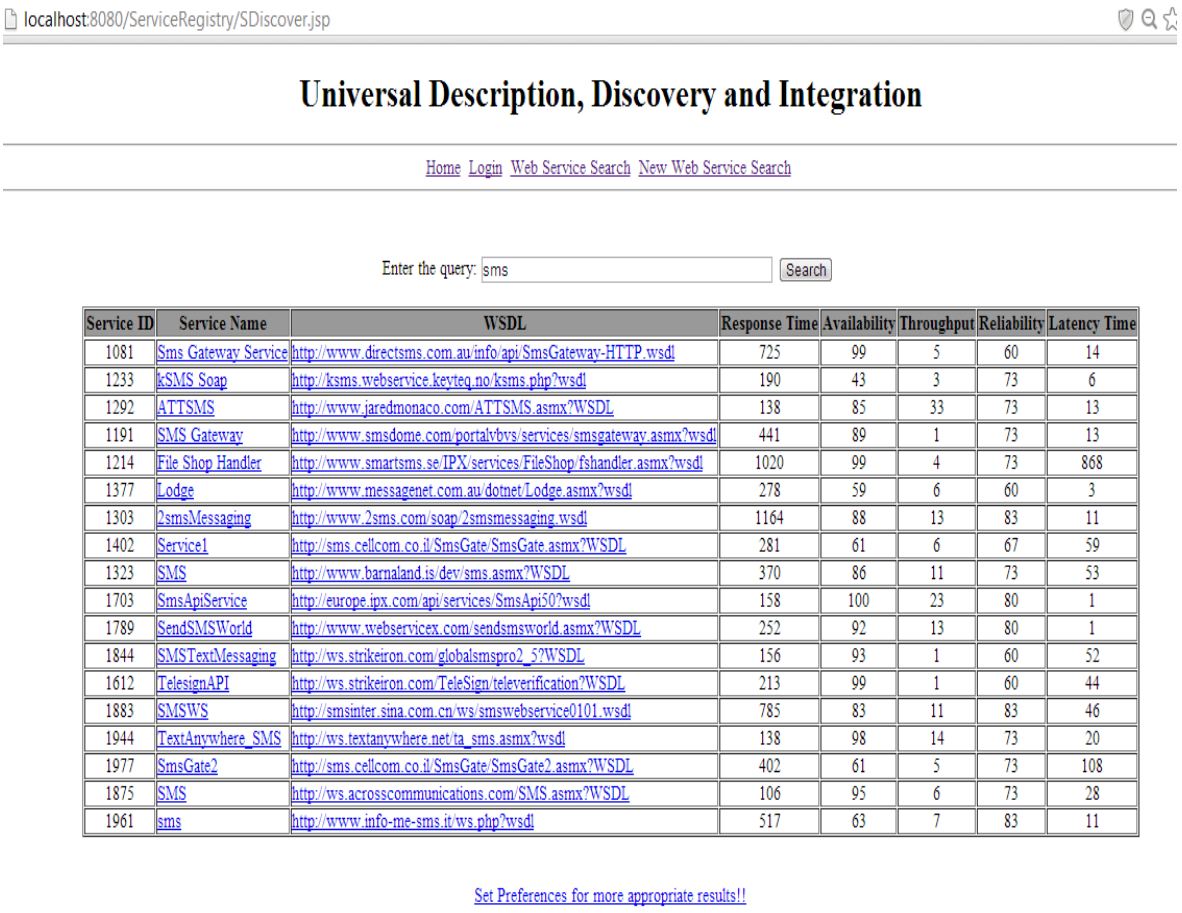


Fig. 9. Search Results based on Semantic Algorithm

TABLE II
Results obtained Semantic based on the User Query

Service Name	Response Time	Availability	Throughput	Reliability	Latency Time
Sms Gateway Service	725	99	5	60	14
kSMS Soap	190	43	3	73	6
ATTSMS	138	85	33	73	13
SMS Gateway	441	89	1	73	13
File Shop Handler	1020	99	4	73	868
Lodge	278	59	6	60	3
2smsMessaging	1164	88	13	83	11
Service1	281	61	6	67	59
SMS	370	86	11	73	53
SmsApiService	158	100	23	80	1
sendSMSworld	252	92	13	80	1
SMSTextMessaging	156	93	1	60	52
TelesignAPI	213	99	1	60	44
SMSWS	785	83	11	83	46
TextAnywher_SMS	138	98	14	73	20
SmsGate2	402	61	5	73	108
SMS	106	95	6	73	28
sms	517	63	7	83	11

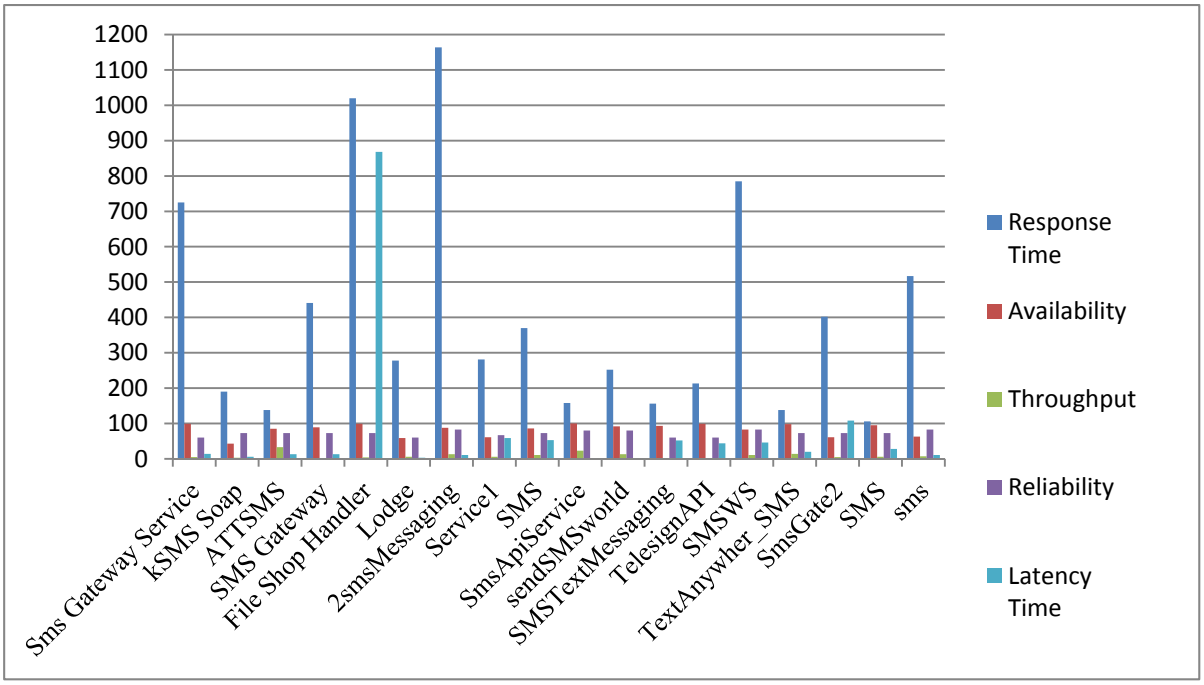


Fig. 10. Semantic based Service Discovered (SD) with QoS_P Web Services

Now in the New Web Service Search the name “SMS” query is searched using semantic based web service selection model. Table II and Fig 10.shows the search results of the semantic based web service selection model for the “SMS” query requested by the service requestor. The result shows 18 services providers with “SMS” functionality located in our UDDI.

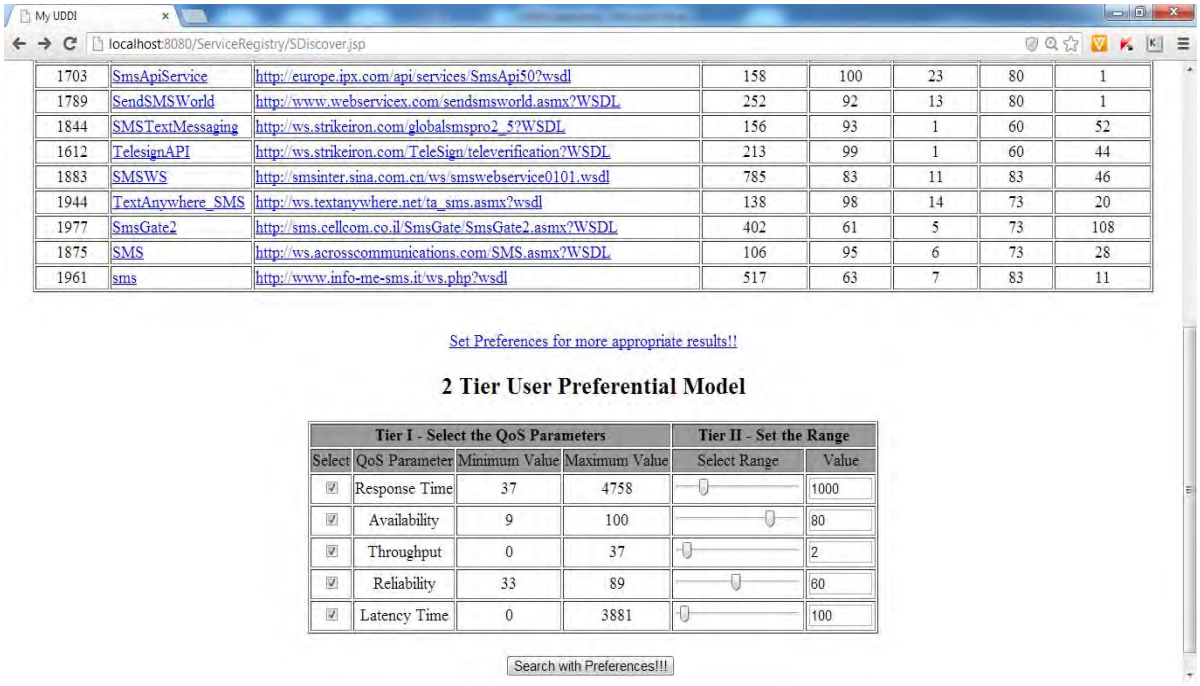


Fig. 11. Service requester setting QoS Parameters

Fig 11.shows the service requester in Tier I selected the QoS parameters and setting QoS range in Tier II for the different Qos parameters in the Tier I. The service requestor has set QoS ranges has Response time ≤ 1000, Availability ≥ 80, Throughput ≥ 2, Reliability ≥ 60 and Latency Time ≤ 100.

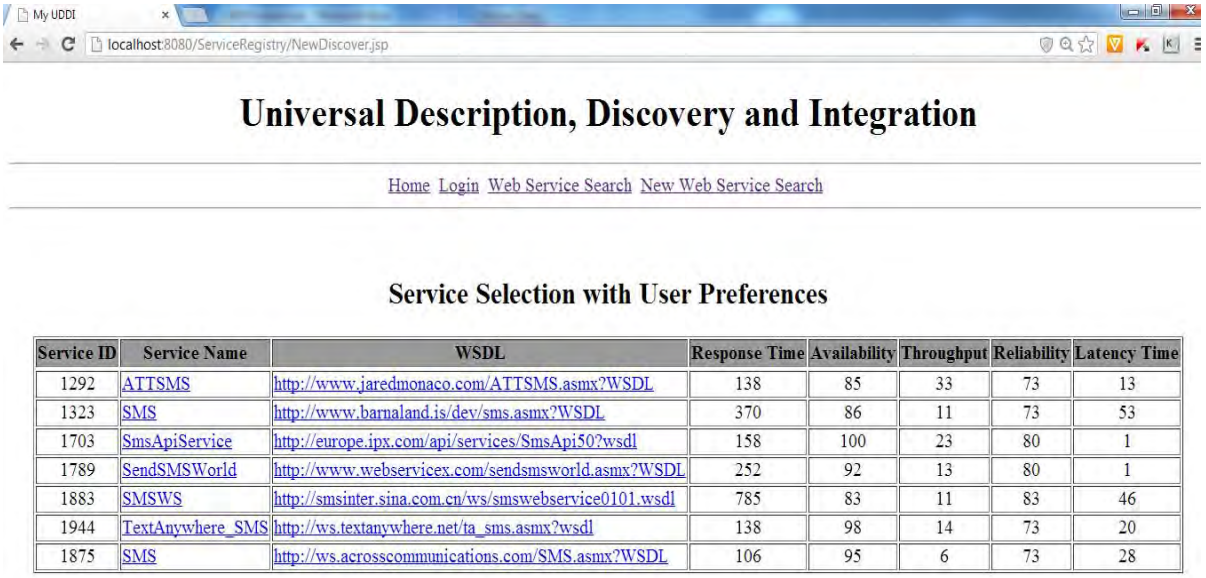


Fig. 12. Search Results based on 2 Tier User Preferential Model

Fig 12.shows the final output of the search query within the user defined QoS parameters. The search results shows seven service providers can provide on terms with the user preferred QoS parameters. Thus by using this 2 tier user preferential search model service requesters are able to much better search results.

TABLE III
Results obtained with user preferences web services

Service Name	Response Time	Availability	Throughput	Reliability	Latency Time
ATTSMS	138	85	33	73	13
SMS	370	86	11	73	53
SmsApiServie	158	100	23	80	1
sendSMSworld	252	92	13	80	1
SMSTextMessaging	156	93	1	60	52
SMSWS	785	83	11	83	46
TextAnywher_SMS	138	98	14	73	20
SMS	106	95	6	73	28

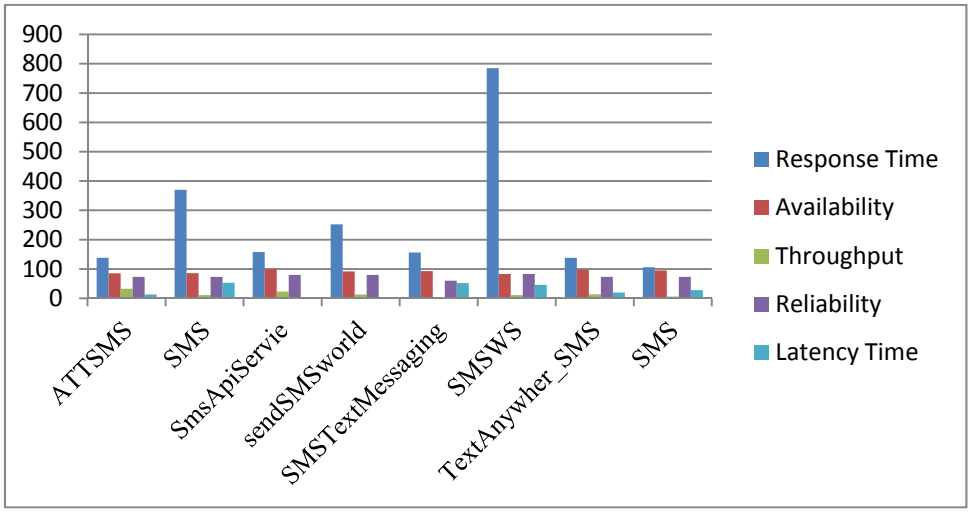


Fig. 13. User Preferences based Service Selection (S_S) with QoS_R Web Services

During the process of service selection, A service requester whose QoS requirement is denoted as: *QR = {(c1 (Response Time,≤,1000), c2 (Availability,≥,80), c3 (Throughput,≥,2), c4 (Reliability,≥,60), c5 (Latency Time,≤,100))}.*

In the QoS requirement *QR*, *QGT = {c1, c2, c3, c4, c5}*. After the execution of 2 Tier user preferential model, we filter selected Web service set S_S = *{ATTSMS, SMS, SmsApiService, SendSMSWorld, SMSWS, TextAnywher_SMS and SMS}* in Table III and Fig 13. Especially Remaining 11 services are not selected because they fail to pass the requester's QoS requirement conditions.

VI. Conclusion

In this paper, we have proposed a web service selection model based on the user preferred QoS. In our experiment we have successfully demonstrated search results of different web service selection methods and illustrated the effectiveness of our search technique proposal by using 2 tier user preferential search model. We have used keyword based and semantic based web search techniques to select the appropriate services from the UDDI for the service requester. We have successfully demonstrated the effectiveness of our 2 tier user preferential search model by implementing our own UDDI in a JAVA platform. The search results will be further enhanced using different search algorithms.

Acknowledgment

This work is a part of the Research Projects sponsored under the Major Project Scheme, UGC, India, Reference No.: F. No. 40-258/2011(SR), dated 29 June 2011, F. No. 41-619/2012(SR), dated 01 July 2012 and F. No.41-639/2012(SR)/Dt.16-07-2012. The authors would like to express their thanks for the financial support offered by the Sponsored Agency.

References

[1] W3C Web Services Workshop, http://www.w3.org/2001/03/WSWS-popa/paper12.
[2] Kreger, H. (2001) "Web Services Conceptual Architecture (WSCA 1.0)." www.ibm.com/software/solutions/webservices/pdf/wsca.pdf (April 13, 2007).
[3] Yeom, G., Yun, T. and Min, D. (2006). "A QoS Model and Testing Mechanism for Quality-driven Web Services Selection." Proceedings of the Second International Workshop on Collaborative Computing, Integration, and Assurance (SES-WCCIA'06), IEEE 2006.

[4] http://www.uddi.org/.
[5] N.Balaji, G.Sambasivam, S.R.Murugaiyan, M.S.SaleemBasha, T.Vengattaraman and P.Dhavachelvan, “Appraisal and Analysis on Diversified Web Service Selection Techniques based on QoS Factors”, International Journal of Engineering and Technology. ISSN: 0975-4024, Vol 5 No 3 Jun-Jul 2013, pp.3012 – 3019.
[6] N.Balaji, RenukaSruthiThorati, M.S.SaleemBasha and P.Dhavachelvan, “An Efficient Web Service Selection Technique using QoS based Service Level Agreement”, International Journal of Applied Engineering Research. ISSN: 0973-4562, Vol 8 No 9 (2013), pp. 983 - 1001.
[7] Bellwood, T. et. al. (2002). UDDI Version 2 API Specification, Available: http://uddi.org/pubs/ProgrammersAPI-V2.04-Published-20020719.htm
[8] P. Dhavachelvan and G.V.Uma (2003), “Multi-agent Framework Construction for Intra-Class Testing of Object-Oriented Software”, 18th ISCIS 2003, Springer-Verlag - Lecture Notes in Computer Science (LNCS), Vol. 2869, pp. 992-999. ISSN: 0302-9743.
[9] Shaikhali, A., Rana, O. F., Al-Ali, R. J., & Walker, D. W. (2003). “UDDI: An extended registry for web services," in Symposium on Applications and the Internet Workshops (SAINT'03 Workshops).
[10] P. Dhavachelvan and G.V.Uma (2005), “Complexity Measures For Software Systems: Towards Multi-Agent Based Software Testing”, Proceedings - 2005 International Conference on Intelligent Sensing and Information Processing, ICISIP'05 2005, Art. no. 1529476 , pp. 359-364.
[11] P. Dhavachelvan, G.V. Uma and V.S.K.Venkatachalapathy (2006), “A New Approach in Development of Distributed Framework for Automated Software Testing Using Agents”, International Journal on Knowledge –Based Systems, Elsevier, Vol. 19, No. 4, pp. 235-247, August-2006.
[12] M. Paolucci, T. Kawamura, T. Payne, and K. Sycara [2002] : Importing the semantic web in UDDI. International Workshop on Web Services, E-Business, and the Semantic Web, pp. 225-236.
[13] Paul P.V., T. Vengattaraman, P. Dhavachelvan (2010), “Improving efficiency of Peer Network Applications by formulating Distributed Spanning Tree”, Proceedings - 3rd International Conference on Emerging Trends in Engineering and Technology, ICETET 2010, Art. no. 5698439, pp. 813-818 .
[14] M. S. Saleem Basha and P. Dhavachelvan (2010), “Web Service Based Secure E-Learning Management System - EWeMS”, International Journal of Convergence Information Technology, Vol. 5, No. 7, pp. 57-69. ISSN: 1975-9320.
[15] S. Venkatesan, P. Dhavachelvan and C. Chellapan (2005), “Performance analysis of mobile agent failure recovery in e-service applications”, International Journal of Computer Standards and Interfaces, Elsevier, Vol-32, No.1-2, pp. 38-43. ISSN:0920-5489.
[16] Murugappan. A, Ramachandran.B and Dhavachelvan. P (2011), “A survey of Keyword spotting techniques for Printed Document Images", Artificial Intelligence Review, 35 (2), pp. 119-136.
[17] P. Victer Paul, D. Rajaguru, N. Saravanan, R. Baskaran and P. Dhavachelvan, "Efficient service cache management in mobile P2P networks", Future Generation Computer Systems, Elsevier, Volume 29, Issue 6, August 2013, Pages 1505–1521.
[18] Phil Bonderud, Sam Chung , Barbara Endicott-Popovsky [2008] : Toward Trustworthy Service Consumers and Producers, IEEE Third International Conference on Internet and Web Applications and Services, 978-0-7695-3163-2/08, pp. 451-456.
[19] D.Chandramohan, T.Vengattaraman, P.Dhavachelvan, R.Baskaran, V.S.K.Venkatachalapathy, "Fewss - framework to evaluate the service suitability and privacy in a distributed web service environment, International Journal of Modeling, Simulation, and Scientific Computing, DOI: 10.1142/S1793962313500165.pp. 1-37 2013.
[20] Youngkon Lee [2008] : Web Services Registry implementation for Processing Quality of Service, IEEE International Conference on Advanced Language Processing and Web Information Technology, 978-0-7695-3273-8/08, pp. 538-543.
[21] T. Vengattaraman, S. Abiramy, P. Dhavachelvan and R. Baskaran (2011), “An Application Perspective Evaluation of Multi-Agent System in Versatile Environments”, International Journal on Expert Systems with Applications, Elsevier, Vol. 38, No. 3, pp. 1405-1416.
[22] P. Victer Paul, A. Ramalingam, R. Baskaran, P. Dhavachelvan, K. Vivekanandan and R. Subramanian, “A new population seeding technique for permutation-coded Genetic Algorithm: Service transfer approach”, Journal of Computational Science, Elsevier, 2013, http://dx.doi.org/10.1016/j.jocs.2013.05.009.
[23] R. Baskaran, P. Victer Paul and P. Dhavachelvan (2012), “Algorithm and direction for analysis of global replica management in P2P network”, International Conference on Recent Trends in Information Technology, ICRTIT 2012, Art. no. 6206826 , pp. 211-216.
[24] Chandramohan.D, Vengattaraman.T, Rajaguru.D, Baskaran.R, Dhavachelvan.P, "A privacy breach preventing and mitigation methodology for cloud service data storage", Proceedings of the 2013 3rd IEEE International Advance Computing Conference, IACC 2013, art. no. 6514199, pp. 83-88.
[25] Huimin HE, Haiyan DU, Dongxia HAN, Yuemei HE [2008] : Research on the Models to Customize Private UDDI Registry Query Results, The 3rd International Conference on Innovative omputing Information and Control (ICICIC'08), 978-0-7695- 3161-8/08.
[26] D.Chandramohan, D.Veeraiah, M.Shanmugam, N.Balaji, G.Sambasivam, S.Khapre, "SVIP-enhanced security mechanism for SIP based VoIP systems and its issues", Advances in Intelligent Systems and Computing, 176 AISC (VOL. 1), pp. 81-86, 2012.
[27] P. Victer Paul, N. Saravanan, S.K.V. Jayakumar, P. Dhavachelvan and R. Baskaran, “QoS enhancements for global replication management in peer to peer networks”, Future Generation Computer Systems, Elsevier, Volume 28, Issue 3, March 2012, Pages 573–582.
[28] Jiuxin Cao, Jingyu Huang, Guojin Wang, Jun Gu [2009] :QoS and Preference based Web Service Evaluation Approach, IEEE Eighth International Conference on Grid and Cooperative Computing, 978-0-7695-3766-5/09, pp. 420-426.
[29] Chandramohan.D, Vengattaraman.T, Rajaguru.D, Baskaran.R, Dhavachelvan.P, "Hybrid authentication technique to preserve user privacy and protection as an end point lock for the cloud service digital information", International Conference on Green High Performance Computing, ICGHPC 2013, art. no. 6533904, pp.1-4.
[30] P. Victer Paul, A. Ramalingam, R. Baskaran, P. Dhavachelvan, K.Vivekanandan, R. Subramanian and V.S.K.Venkatachalapathy, "Performance Analyses on Population Seeding Techniques for Genetic Algorithms", International Journal of Engineering and Technology (IJET), Vol 5, No 3, Jun-Jul 2013, pp. 2993-3000. ISSN : 0975-4024.
[31] Rajeswari, M. et al., Appraisal and analysis on various web service composition approaches based on QoS factors. Journal of King Saud University – Computer and Information Sciences (2013), http://dx.doi.org/10.1016/j.jksuci.2013.08.003.
[32] UDDI.org, “UDDI Technical White Paper”, Retrieved April 30, 2006 from http://uddi.org/pubs/uddi-tech-wp.pdf.