A Survey on Service Based Discovery Methods and Techniques

[1.] Ms. Rohini Jadhav, [2.]Prof. Dr. S. D. Joshi,
[1.]M.Tech Student, [2.]Professor
Dept of Computer Engineering,B. V. U. College of Engineering,
Pune, Maharashtra 411043, India

***Abstract*— A Web service is a set of related dynamic application methods that can be accessed programmatically over the Internet. Continuously various amount of data is updated and downloaded over an internet. However, the Centralized and distributed nature of the partner services in service discovery present unique challenges to the reliability the composite services on real time application. Businesses can dynamically web services to perform complex functionality with minimal programming effort. Web services allow buyers and sellers all over the world to find each other, connect dynamically, and execute transactions in real time with minimal human interaction with system. Web services are self-contained, self-describing modular Client – Server applications that can be published, located, and invoked across Web. In this condition to generate the method, which can give, secure and optimized result with QoS over dynamic nature of an application. In this paper, we review the various proposed technique to implement cost effective reliable system.**

Keywords*— *Web-services discovery, web services, QoS, Application Program, centralized and Dynamic Service Discovery.

I. INTRODUCTION

Now a day, Service discovery [1,2] requires a need of a language palindrome to allow software agents to make use of one another's new services without the need for dynamically/sequential user intervention. Service discovery has been valuable as an important activity for service-based activity systems. The service discovery mechanism should provide a number of functionality, recognizable at both development phase and execution phase. The identification of services to replace existing services in service-based applications during execution time of these applications has been recognized as an important activity for service-based applications. In such situations, it is necessary to identify services that can fulfill structural, functional, quality, and contextual properties and that can be deployed as part of service- phase. The identification of services to replace existing services in service-based applications during execution time of these applications has been recognized as an important activity for service-based applications. In such situations, it is necessary to identify services that can fulfill structural, functional, quality, and contextual properties and that can be deployed as part of service-based applications. There are several circumstances that may arise the need to identify services during execution phase of service-based applications programs for various cases such as (a) unavailability or malfunctioning of services participating at the run time of an application, (b) changing in functioning , quality of services [3] or types of services participating in the run time application programs (c) changes in context of an application that uses a running service for discovery, or (d) merging of new services that can fully supported for the role of an existing service in an run time application in a superior way, i.e., in a better way than the current service in an run time application[1].

Service discovery addresses the general problem in which some elements in a service-based activity systems, the clients, need to know the services offered by other elements in the network, the servers. The services are change randomly it can directly affect on application and their service at run time. There are some technique are consider, i.e. Static Service Discovery and Dynamic Service Discovery [2].

Static Service Discovery

The service implementation details bound at design time and service retrieval performed on a service registry[10,11]. A human designer examines the results of the retrieval operation usually and the service description returned by the retrieval operation incorporated into the application logic.

Dynamic Service Discovery

The service implementation details left unbound at design time so that they can be determined at run time. The service is requested user have to specify specific preferences to able the application to reason which service(s) the requester is most likely to want to invoke at run time. Depend on type of different logic used in application service quality may be considerations such as cost factor, performance, security certificates, and so on the the application chooses the most suitable service, binds to it, and invokes it[9].

The above circumstances give rise to a basic research challenge i.e. how to support service-based applications when the services that they use disappear or stop functioning as expected as well as in the presence of continuously changing contexts of both the applications and their services at runtime pull and push modes of query execution are performed in order to identify candidate services to rep lace services participating in a service-based application.

The pull mode [1, 2, 5] is not more effective query execution process, discovery process triggered only after the

need for a new service arises and it may take considerable time to complete, affecting the performance of the run application its ability to produce acceptable "real-time application" response to the user. Pull mode is performed by searching service registries when a service-based application is generated, in order to assist with the identification of services that are initially bound to the application, as well as during execution time of an application in order to support cases (a) to (d) above and in this query execution would need to be enhanced with mechanisms for polling regularly service and assist with the identification of candidate replacement services for the bound services.

The push mode [1, 2, 5] of query execution is performed in a proactive way, in which services are identified in parallel to the execution of an application, to support cases (a) to (d) above, based on subscribed services and queries and up-to-date sets of candidate replacement services. These queries are associated services in an application at the run time and propose to maintain up-to-date sets of client replacement services for these services. There are two types of approach we have to consider i.e. distributed approach or following a centralized approach. The distributed approach no additional device needed and the discovery of services can do in two ways:

In push and pull modes, query execution is based on matching technique and the computation of distances between query as well as service specifications. (i) Push mode, in which servers send unsolicited advertisements, and clients listen to these advertisements selecting the web services [13] they are interested in an application; (ii) pull mode, in which clients request for a service when they need it, and servers that offer the service answer the request. In the centralized approach, a new element, the directory, is introduced (be it a single physical host or a hierarchy of hosts). Servers register their services in the directory and clients send their requests to the directory. Now a day, it is necessary to develop new approach to enhance to service discovery and improve the performance at run time. Furthermore, in this survey paper we have to finds out various technique are propose and new frame work to enhance to the service discovery.

The remainder of this paper structured as follows, Section 2. Describe the literature survey of existing system Section 3 Limitation of existing system. Section 4 presents the service discovery process supported by the framework. Finally, Section 5 presents conclusions and plans for future work.

II. LITERATURE SURVEY ON DYNAMIC SERVICES DISCOVERY

In this section, we present literature for runtime dynamic service discovery as follows, Bernstein, Abraham, and Mark Klein, described [15] scheme for new service retrieval approach based on the sophisticated use of process. This approach offers qualitatively higher retrieval precision than existing (keyword and table based) approaches without sacrificing recall and computational ractability / scalability. High retrieval services precision is widely recognized for important uses that range from finding useful software components or applications programs. Process-based queries will produce retrieval precision qualitatively more than existing service retrieval approaches scheme, while retaining complexity for query enhancement and large space is require to store the data is generating at the time of service discovery.

Toma, Brahmananda Sapkota, James Scicluna, Juan Miguel Gomez, Dumitru Roman, and Dieter Fensel addressed [16] the discovery problem for a specific Semantic Web service execution environment, namely Web Service Modeling Execution Environment (WSMX). More precisely, they proposed a P2P Discovery mechanism for Semantic Web services [8] descriptions that registered with Web Service Modeling Execution Environment. The use of shared space makes this architecture reliable, fault tolerant, scalable as each node in the Internet [7] can be imagined hosting a shared space. Propose technique the space is centralized to hold the search query it is difficult when the search traffic is more to enhance the result it is necessary to provide the virtual space. The service discovery content is easily encryption is not provided by the propose model.

Eyhab Al-Masri and Qusay H. Mahmoud proposed[17] a solution by introducing the Web Service Relevancy Function (WsRF) used for measuring the relevancy ranking of a particular Web service based on Qos metrics and client preferences for the purpose of finding the best available Web service discovery process based on a set of given client QoS preferences service during Web services discovery process based on a set of given client QoS preferences or QoS search criteria. QoS parameters from accessible Web services were measured using WS-QoSMan and were used as search constraints in order to retrieve Web services with an accurate relevancy ranking. Propose technique is to provide client's ability to control the discovery process across accessible service registries for finding services of interest as per metadata on web services but author not support to propose the security as well as encryption technique to secure the search result.

Farnoush Banaei-Kashani, Ching-Chien Chen, and Cyrus Shahabi [18]proposed peer-to-peer architecture of the semantic-enabled WSPDS proposed technique improves scalability of Gnutella's flooding-based dissemination mechanism by up to 99%, effectively eliminating the major drawback of this Gnutella-like peer-to-peer discovery system. A fully decentralized and interoperable discovery service with semantic-level matching capability, in this paper users' requests with globally shared concepts explained it can transport to all node for search discovery and generate multiple result at the time of discovery and no security is consider as well as the virtual space concept.

I. Toma, B. Sapkota, J. Secuila, J. M. Gomez, D. Roman, and C. Bussler[19] attempted to bring web service discovery mechanism on top of Peer -to-Peer network thereby reducing human intervention which is concerned with resource linking but nothing has been mentioned about the applications that process these resources and proposed keyword based discovery mechanism is used for the local discovery.

Fatih Emekci, Ozgur D. Sahin, Divyakant Agrawal, Amr El Abbadi [23]proposed a structured peer-to-peer framework for web service discovery in which web services are located based on both service functionality and process behavior. In addition, they integrate a scalable reputation model in this distributed peer-to-peer framework to rank web services based on both trust and service quality. Web services can join and leave the system dynamically. Returned services are ranked based on the trust and quality ratings of the services using reputation model.

Jacek Kopecky intended[20] to research an approach to Semantic Web Service discovery that divide the desire task of finding the most specific offer from available web services into static discovery services and then dynamic offer discovery. Initially it will find all available web services fulfilling the user's goal, by interacting with discovered services, find offers relevant for the goal, filter out the offers, which do not meet the user constraints, rank the offers according to user's preferences and choose one to b e invoked and then use the selected service.

Jorge Cardoso and Amit Sheth presented[21]a set of algorithms for web service discovery based on three dimensions: syntax, operational metrics, and semantics. In this approach, Web service discovery is not only based on functional requirements, but also on operational metrics. Operational metrics of web services described by QoS model representing quantitative and qualitative characteristics of an e – workflow application necessary to achieve initial requirements. An approach based on the use of ontologies to describe workflow tasks and Web service interfaces to facilitate the discovery and posteriori integration of Web service into workflows

K. Verma, K. Sivashanmugam, A. Sheth, A. Patil, S. Oundhakar and J. Miller [22] presented ETEOR - S Web Services Discovery Infrastructure (MWSDI), a scalable infrastructure for semantic publication and discovery of Web services as well as creating an infrastructure of registries for semantic publication and discovery of Web services.

Paulo Sousa Alysson Neves Bessani Miguel Correia[2], describe approach that combines proactive recovery scheme with services that allow correct replica to react , recover suspect to be compromised and despite recoveries, it guarantees the availability of the minimum amount of system replicas necessary to sustain system's correct operation at run time.

III. SCOPE OF RESEARCH

From above discussion of literature, there is scope of research to implement best model for service discovery in dynamic nature with virtual space and design an optimized XML query[12] to make model perfect. Dynamic and self-organizing systems like those found in ubiquitous computing or semantic web based scenarios raise numerous challenges regarding trust and privacy and design a service discovery system to enhance the performance regarding (a) dynamic service discovery based on both pull and push mode of query execution; (b) the use of a pro-active push mode of discovery process in which services are identified in parallel to the execution of an application; and (c) the use of an XML-based query language for expressing complex service discovery queries representing structural, behavioral, quality, and contextual conditions, (d) a new behavioral matching process that considers behavioral constraints described as temporal logic expressions.

IV. SECURITY REQUIREMENTS OF SERVICE DISCOVERY

The service requester has play important role at the time of service provider, at this time some issue is import to discuss.

- Service authentication and Client: the objective of service discovery is to communicate with previously unknown client that provide specific functionalities as per requirements. Discovery services therefore require that the first message (ex: hello message) sent be in clear, also meaning that the content of the message can be accessed. Without authenticate clients and servers nodes; service discovery makes the implementation of attacks possible, a malicious entity being able to wrongly answer of current discovery message. The new registry based discovery schemes obviously make it much simpler than infrastructure-less ones to perform secure discoveries, since the registry is the only element which the client needs to identify and which the client should be identified to, yet at the cost of additional infrastructure deployment requirements.

- Privacy: Service Discovery initialized takes important risk than the other party does since it does not control which entities will receive on request neither the discovery message, nor the potential usage of the information embedded in his request message. The information disclosed by client side requests is likely to reveal a subset of the intentions of the service requester of a same architecture. A different attacker to gather profiles of current or all users of the service discovery mechanism based on the information carried by discovery messages and providing some more information (host name, Certificate, Credentials …). The association of such data with discovery related information is particularly worrying from a privacy protection perspective.

- Access control: Authentication of client/service is problematic in first discovery stage, traditional service oriented architectures do not support access control during discovery. Service provider advertises services exclusively to authorized client. Still, description of a service to any request user potentially increases the risk that a malicious client.

- Availability: Generally, an attack to the availability of resources preventing the authorized access to a system resource or delaying system operations i.e. Denial of Service (DoS). Service discovery descriptions during discovery enables attackers to exploit vulnerabilities by creating speciallynew generated messages for the server.

V. CONCLUSION

This paper has aimed to give an overview of recent progress in automatic service discovery, at first we explain service discovery and find out various circumstance. We have introduced the static, dynamic and discussed how they are different and depending on the application, which one may perform better than the other as well as find out various literature proposed on same technique and last on the limitation of already proposed technique related to privacy and trust explain security requirement to improve the performance of service discovery at centralized environment.

ACKNOWLEDGMENT

My sincere gratitude towards my project guides Prof. Dr. Guide Name for his/her guidance and encouragement.

REFERENCES

1] Andrea Zisman, George Spanoudakis, James Dooley, Igor Siveroni, "Proactive and Reactive Runtime Service Discovery: A Framework and Its Evaluation", IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 39, NO. 7, JULY 2013, page 954-974.

[2] Paulo Sousa Alysson Neves Bessani Miguel Correia, "Resilient Intrusion Tolerance through Proactive and Reactive Recovery " 13th IEEE International Symposium on Pacific Rim Dependable Computing , DOI 10.1109/PRDC.2007.52 page no: 373-380.

[3] A. Coronato, V. De Florio, M. Bakhouya, G. Di Marzo Serugendo, "Formal Modeling of Socio-technical Collective Adaptive System" , 2012 IEEE Sixth International Conference on Self-Adaptive and Self-Organizing Systems Workshops, DOI 10.1109/SASOW.2012.40, page no: 187 – 192

[4] Khaled Mahbub, George Spanoudakis, "Proactive SLA Negotiation for Service Based Systems:Initial Implementation and Evaluation Experience", 2011 IEEE International Conference on Services Computing, DOI 10.1109/SCC.2011.34, page no- 16 -23

[5] Paulo Sousa, Miguel Correia, "Highly Available Intrusion-Tolerant Serviceswith Proactive-Reactive Recovery", IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 21, NO. 4, APRIL 2010,DOI 10.1109/TPDS.2009.83, page no – 452- 465

[6] Christoph Becker, Hannes Kulovits, Michael Kraxner, Riccardo Gottardi, and Andreas Rauber , "Measuring and Evaluating Tool Performance in a Service-oriented Architecture", ICWE, volume 5648 of Lecture Notes in Computer Science, page 221-235. Springer, (2009)

[7] Jinghai Rao and Xiaomeng Su , "A Survey of Automated Web Service Composition Methods" In Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition, SWSWPC 2004. Page no :43-54.

[8] Onyeka Ezenwoye and S. Masoud Sadjadi, "A Proxy-Based Approach to Enhancing the Autonomic Behavior in Composite Services", JOURNAL OF NETWORKS, VOL. 3, NO. 5, MAY 2008, ACADEMY PUBLISHER, page no: 42- 53

[9] Ingrid Buckley, Eduardo B. Fernández, Marco Anisetti, Claudio Agostino Ardagna, Seyed Masoud Sadjadi, and Ernesto Damiani. OTM Conferences 2, volume 7045 of Lecture Notes in Computer Science, page 560-576. Springer, (2011)

[10] Antonia Bertolino, Guglielmo De Angelis, Lars Frantzen, and Andrea Polini. ISSSE, volume 5413 of Lecture Notes in Computer Science, page 106-139. Springer, (2008)

[11] Rebecca L. Collins, and James S. Plank, "Downloading Replicated, Wide-Area Files – a Framework and Empirical Evaluation" NCA, page 89-96. IEEE Computer Society, (2004)

[12] Daniela Grigori, Juan Carlos Corrales, and Mokrane Bouzeghoub. ICWS, page 145-152. IEEE Computer Society, (2006)

[13] Lei Li, and Ian Horrocks. WWW '03: Proceedings of the 12th international conference on World Wide Web, page 331--339. New York, NY, USA, ACM Press, (2003)

[14] Bernstein, Abraham, and Mark Klein [2002]: Discovering Services: Towards High Precision Service Retrieval. Proceedings of the CaiSE workshop on Web Services, e-Business, and the Semantic Web: Foundations, Models, Architecture, Engineering and Applications. Toronto, Canada

[15] Brahmananda Sapkota, Dumitru Roman, Dieter Fensel [2005]: Distributed Web Service Discovery Architecture .

[16] I. Toma, B. Sapkota, J. Secuila, J. M. Gomez, D. Roman, and C. Bussler [2005]: P2p discovery mechanisms for web service execution environment. Second WSMO Implementation Workshop.

[17] Eyhab Al Masri and Qusay H. Mahmoud [2007]: Discovering the Best Web Service. WWW 2007.

[18] F. Banaei Kashani, C.C. Chen, and C. Shahabi. WSDP [2004]: Web services peer-to-peer discovery service. International Conference on Internet Computing.

[19] I. Toma, B. Sapkota, J. Secuila, J. M. Gomez, D. Roman, and C. Bussler [2005]: P2p discovery mechanisms for web service execution environment. Second WSMO Implementation Workshop.

[20] Jacek Kopecký [2007]: Semantic Web Service Offer Discovery . OTM 2007 Ws, Part I, LNCS 4805.

[21] Jorge Cardoso and Amit Sheth [2002] : Semantic e Workflow Composition. Technical Report# 02 004, LSDIS Lab, Computer Science Department, University of Georgia, Athens GA, July 2002

[22] K. Verma, K. Sivashanmugam, Amit Sheth, Abhijit Patil, Swapna Oundhakar, John Miller : METE OR – S WSDI [2003]: A Scalable P2P Infrastructure of Registries for Semantic Publication and Discovery of Web Services

[23] Fatih Emekci, Ozgur D. Sahin, Divyakant Agrawal, Amr-El Abbadi [2004] : A Peer-to-Peer Framework for Web Service Discovery with Ranking. IEEE International Conference on Web Services.