A Survey on Problems in Distributed UDDI

R.Rajmohan
Dept. of CSE
Pondicherry University
Pondicherry, India

N.Padmapriya
Dept. of CSE
Pondicherry University
Pondicherry, India

S.K.V.Jayakumar
Asst. Professor, Dept. of CSE
Pondicherry University
Pondicherry, India

ABSTRACT
UDDI (Universal Description, Discovery, and Integration) is a directory to register and locate web service application. It facilitates businesses to easily and dynamically find and transact with one another through their preferred services. The present centralized UDDI structure is less robust and it is difficult to support a large number of Web services. It has become the bottleneck of the whole system and would cause single node failure problems. There is lack of interoperability between UDDI registries because different registries have different usage policies and pose various requirements on acceptable announcements and retrieval demands of Web services. These problems can be solved by distributing the functions and volume set of data of UDDI. Distributed UDDI is built upon the functionalities similar to distributed database. This suggests that the problems in distributed database can also be experienced in distributed UDDI. This paper discusses the issues and challenges in distributed UDDI, similar problems between distributed UDDI and distributed database and the existing solutions for them.

Keywords
Distributed database, Distributed UDDI, Federated, P2P.

1. INTRODUCTION
UDDI acts a very important role in the web service. It can be used to publish and lookup services. Most of the current UDDI models are centralized so that the performance will decrease if there are too many services to be registered or queried. The centralized UDDI structure is less robust and experiences bad interoperability. Current UDDI is a passive service directory, which makes detection of service changes and failure web services to be difficult. UDDI uses static descriptions of services for registering. But their dependability characteristics (e.g. avail-ability, response time, etc.) can vary significantly during service operation. UDDI Registry should publish actual dependability of Web Services registered i.e. they should perform periodical monitoring, gather user's side feedbacks, update WS meta-data, etc. UDDI-specific systems have to guarantee real time responses to service enquiries with the service data that reflects the most updated information of services. But they do not have an automatic mechanism for updating the registry as services (and service providers) change. The current UDDI attempts to reduce the disadvantages of the centralized approach by replicating the entire information and putting them on different registries. It temporarily improves the performance of UDDI but it involves a high cost in deployment and maintenance. Replicating the UDDI data is inconsistent and not a scalable approach. Several decentralized approaches have been proposed moving from a central design to a distributed structure by connecting registries with distributed technologies. It's the distributed architecture that allows the system to easily process data process requirements from service providers and users in a correlated and synchronized way. The system is also designed with a special support of concurrent access to service data.

This paper proceeds in the next section with a survey of problems in distributed database. The next section formalizes an extensive study and material collection on distributed uddi. Next, we map the problems of distributed database in distributed uddi. We next introduce the available approaches for solving the problems in Distributed UDDI. Finally, we conclude the paper.

2. PROBLEMS IN DISTRIBUTED DATABASE
A distributed system usually exhibits more complexity and cost more than a centralized one. This is true because the hardware and software involved need to maintain a reliable and an efficient system. All the replication and data retrieval from all sites should be transparent to the user. The cost of maintaining the system is considerable since technicians and experts are required at every site. Deploying a distributed database produces problems like

- **Distributed Database Design:** Distributing the data, becomes critically important in a decentralized environment. Another key issue is that has to be resolved in evaluating the feasibility of a distributed database is the degree of decentralization.
- **Query Optimization:** In a distributed database the optimization of queries by the DBMS itself is critical to the efficient performance of the overall system. Query optimization must take into account the extra communication costs of moving data from site to site, but can use whatever replicated copies of data are closest, to execute a query. Thus it is a more complex operation than query optimization in centralized databases.
- **Concurrency Control:** The problems of concurrency control in a distributed DBMS are more severe than in a centralized DBMS because of the fact that data may be replicated and partitioned. If a user wants unique access to a piece of data, for example to perform an update or a read, the DBMS must be able to guarantee unique access to that data, which is difficult if there are copies throughout the sites in the distributed database.
- **Replication:** Replicated data is most efficient when multiple reads of the data are expected, but updates are not as frequent. When replicated data must be updated, however, an update to a record at one node should

cause an identical update at all other nodes where that record resides. If anyone replica is unavailable there could be problems.

- **Maintaining data integrity:** Preservation of integrity is much more difficult in a heterogeneous distributed database than in a homogeneous one A clear concept of a transaction is essential in coordinating multiple updates to distributed data. The multiple nodes and multiple copies of data items can mean distributed chaos if transactions are not carefully implemented and monitored.
- **Security issues:** In developing a distributed database, one of the first questions to answer is where to grant system access. Furthermore, the maintenance of replicated clearance tables is computationally expensive and more prone to error. Finally, the replication of passwords, even though they're encrypted, increases the risk of theft.

3. DISTRIBUTED UDDI

UDDI enables businesses to publish their web services and search them based on consumer preferences. The present centralized UDDI structure is less robust and it is difficult to support a large number of Web services. UDDI registries collects service information in a passive manner, which means it waits for service publication, updating or discovery request passively and thus cannot guarantee the real-time validity of the services information. If distributed environment is introduced in UDDI there will be a rapid increase in services and resources. In the following, we will highlight the characteristics of distributed UDDI systems

3.1 Salient Features

Some of the essential features for better performance of Distributed UDDI are discussed below.

- Efficiency: UDDI should support storage and retrieval of without consuming significant resources.
- Scalability: Efficiency of searching in UDDI should not degrade with increase in network size.
- Flexibility: Semantic service descriptions to be provided for retrieval of exact information requested by users.
- Search Completeness: UDDI should offer maximum discovery of requested objects with high precision and recall.
- Fault Resilience: Maintaining data availability and reducing overhead of node failure is an important aspect of UDDI.
- Load Balancing: distribute routing, storage and processing loads according to the capabilities of the participating nodes.

3.2 Architecture

Based on content sharing and service discovery, the architecture for distributed uddi can be broadly classified into three categories.

Decentralized: In decentralized structure, the web service information are distributed across different UDDI registry. UDDI registry is collection of one or more UDDI nodes. UDDI nodes are servers which support the UDDI specification and belong to a UDDI registry. Some of the works on decentralized model are described below. A novel interoperable model [3] and a Federated UDDI model [2] are used for decentralizing the UDDI registries. The METEOR-S Web Service Discovery Infrastructure (MWSDI) [12] is proposed for discovering web service in federated registry environment. An active and distributed UDDI architecture called Ad-UDDI [20] is proposed to organize the private or semi-private UDDI based on industry classifications. A flexible Hybrid registry deployment architecture [23] is proposed for Management of Distributed UDDI Registries. These decentralized architectures overcome the single node failure problems and increases registry performance.

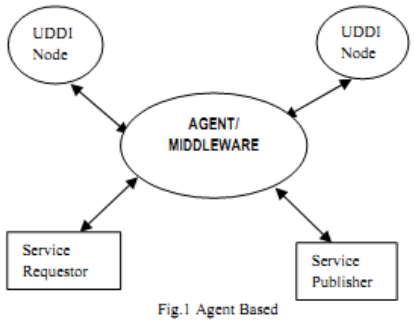

Fig.1 Agent Based

Agent or middleware: Agent is an autonomous trusted authority who acts on behalf of an organization or business entities. These agents will act as proxy to a community of UDDI registries. They receive service requests from the service consumer, find the services that meet the user requirements, and then return the list of services to the consumer. Some of the research works using agent technology are described below. Middleware architecture [22] is proposed for organizing web services in community based on business specific domains. An autonomous proxy agent [24] is proposed for collecting user input and performing service discovery. An optimal agent based architecture [25] is proposed for dynamic discovery of web services based on Qos matching. All these architectures support an infrastructure for Distributed UDDI Registries. Using the agent technology, service descriptions are distributively stored and parallely searched. It reduces the overhead of centralized storage and increases the performance of UDDI registries.

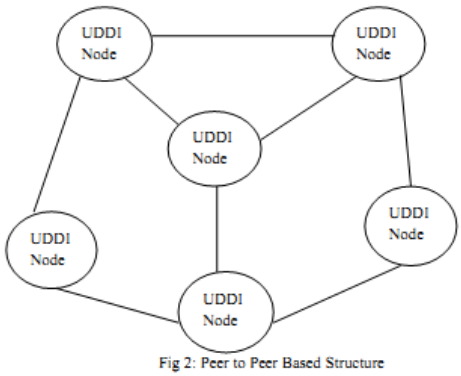

Fig 2: Peer to Peer Based Structure

P2P: In P2P architecture, the functionalities of UDDI are distributed among all the peers in a network. Peers share their resources, such as processing power, disk storage or network bandwidth, with other network participants, without the need for central coordination by servers. These characteristics make P2P more flexible for implementing distributed environment for UDDI. Through P2P we can achieve content and node dynamism, flexibility in query expressiveness and fault-resilience of the search mechanism. Many frameworks like PDUS [1], DWSDM [18], Chord [17, 18, 19], DHT [16], CAN [7, 10], and Semantic Peer [15] are proposed to implement Distributed architecture for UDDI.

3.3 Registration Components

The core component of UDDI is the UDDI business registration. It is an XML file which is used to describe a business entity and its Web services. The information provided in a UDDI business registration consists of three components [26]: "White page" includes the administrative information like address, contact, specifications regarding the license agreement and distribution forms;"Yellow page" includes the classification of the service or business, based on standard taxonomies; "Green page" contains the technical descriptions such as how to access a Web Service i.e. location of a Web service and its interface. Unfortunately the UDDI specification does not provide any conceptual information of a web service. Therefore, "blue pages" [27] are to be introduced in order to provide semantic information of a Web service, which focuses on terms, concepts, and relationships between them.

3.4 Storage Mechanism

In distributed architectures, the directory information is stored at different network locations. Distributed systems can be categorized as replicated, distributed or hybrid.

- Replicated- In the replicated case, the entire directory information is stored at different UDDI nodes. Maintaining consistency in replicated registries is bandwidth consuming problem and it also results in performance bottleneck.
- Distributed- In the distributed case, the directory information is partitioned, and the partitions are stored in different UDDI nodes. When the directory information is distributed in different UDDI nodes, the failure of one of them leads to the unavailability of part of the directory information.
- Hybrid - In the hybrid case, the nodes stores multiple copies of the partitioned directory information in different UDDI nodes. Hybrid architectures offer the best compromise between bandwidth consumption, scalability, and fault-tolerance.

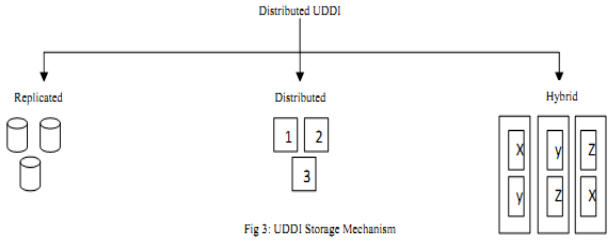

Fig 3: UDDI Storage Mechanism

3.5 Service Description

UDDI is registry which stores description of web services. The most two common methods to publish and discover services are based on syntax and semantics. UDDI standard defines web services based on syntactical structure. It reduces the exactness of the discovered services. Traditional UDDI-Based Web Service systems lack support of Semantic Web Services An ideal method to discover web services is to combine web services and semantic web technologies. There are many modeling languages available for describing the web services semantically. WSMO (Web Service Modeling Ontology) [28] defines a meta-ontology for modeling web services semantically. It is a formal language for semantically describing Web services to facilitate the automation of publishing and discovering services over the Web. OWL-S (Ontology Web Language for Web Services) [29] defines ontologies for describing a web service based on semantic model. OWL-S defines three ontologies: the service *profile* for advertising and discovering services; the *process model*, for describing service operation; and the *grounding*, which provides details on how to interoperate with a service through messages. Web Service Description Language (WSDL-S) [30] is a mechanism to annotate Web Services with semantic descriptions. It provides automatic service discovery, invocation, composition and interoperation. WSDL-S has the ability to reuse existing domain models expressed in modeling languages like UML, OWL etc. These languages are to be incorporated with Distributed UDDI for describing web services based on semantic knowledge.

The UDDI functionalities can be distributed in different ways by using different distributed computing infrastructure. There is no standard architecture for distributed UDDI since now. Many researchers have taken effort to define a standard architecture for distributed UDDI. Some of the research works on distributed UDDI are discussed below

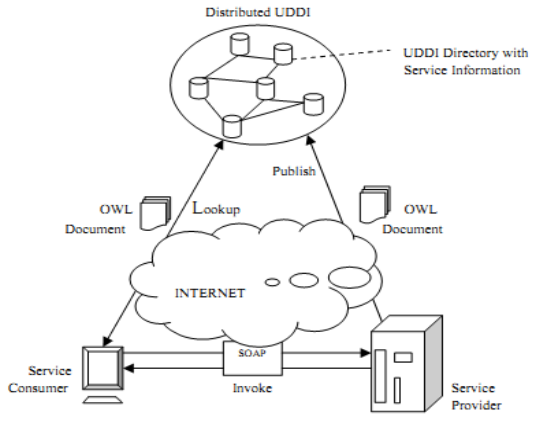

Fig.4 Service Discovery model using Distributed UDDI

Figure 4 depicts a new model for service discovery using Distributed UDDI. In this figure we have proposed a distributed model for UDDI by distributing the functions and volume set of UDDI. UDDI registry functionalities will be in a peer-to-peer fashion. Service providers publish their web service descriptions in the UDDI registry. For describing web services we can use Web Ontology Language (OWL) which provides semantically enriched web service discovery. SOAP protocol is used for exchange of information between the service consumer and provider.

A novel P2P-based Distributed UDDI Web Service Discovery (PDUS) [1] is proposed to overcome poor scalability and less consistency on large registries. It applies P2P technology to distribute, store and query web services over UDDI registries. A federated UDDI system [2] is presented to easily process data process requirements from service providers and users in a correlated and synchronized way. This model is a collection of UDDI nodes connected by a access controller. A novel interoperable model of distributed UDDI [3] is presented to enhance the query efficiency, availability, and offer the way to improve information exchange among servers. The interoperable model maintains three layers namely root UDDI server, super domain UDDI server and normal UDDI server. The Root UDDI server records the information of super domain Servers and provides registration and inquiry of super domain UDDI servers. The super domain UDDI server manages the normal servers and provides registration and inquiry services. A solution based on a inter-node operation enabling the replication APIs to distribute the entries across a Distributed Hash Table (DHT)is discussed [4]. It reduces the number of replicated entries to a sub-set of nodes participating in the UDDI cloud and opens a new possible scenario for the adoption of the UDDI standard.

A novel distributed Web services catalog system is proposed to support decentralized Web service publication and discovery, by utilizing the distributed hash table (DHT) functionality of peer-to-peer (P2P) computing [5]. A distributed web service directory in SOA-oriented urban spatial information sharing platform is proposed using which, we can reasonably organize private or semi-private services registered libraries and easily implement mutual communications among them [6].

4. PROBLEMATIC RELATIONSHIP BETWEEN DBMS AND DISTRIBUTED UDDI

UDDI registries are lack of interoperability and they collect information about the services in a passive manner. Distributed technologies are used to distribute the functions of UDDI, so that they can overcome the problems in centralized architecture. Since UDDI uses the distributed framework, they can experience the problems that occur in distributed database, especially problems like interoperability, scalability, redundancy, and query processing and data management. Some of the problems are discussed below.

- **Concurrent Access:** The concurrent processing on UDDI registries supporting the management of web services may have a significant impact on overall system performance. The fluidity of services and updates in service offerings affect service users via access operations of service data in registry repositories conformed to the protocol.
- **File Redundancy:** Several UDDI registries are spread across different regions. The UDDI registries may belong to different businesses; have different usage policies, and pose various requirements on acceptable announcements and retrieval demands of Web services. The problem of dynamically managing the multiple UDDI has some overlapping with the replication over a set of distributed databases.
- **Heterogeneity:** The potential of a large-scale growth of private and semi-private registries is creating the need for an infrastructure, which can support discovery and publication over a group of autonomous registries. Heterogeneity problem deals with the selection of registry to perform Web service publication or discovery.
- **Intercommunication:** The problem is absence of exchanging and connecting mechanism among the private service registry libraries. This leads to performance bottleneck and maintenance problem of information.
- **Query Optimization:** By federating service registries, advertisement entries in one registry can be discovered by service consumers who query via other remote registries. The problem is defined as how to query in a distributed UDDI environment. Also UDDI does not support searching with semantic information.

5. SOLUTIONS TO THE DISTRIBUTED UDDI PROBLEMS

Distributed UDDI is adopted to overcome the performance bottleneck and single node failure problems of centralized UDDI. Implementing UDDI registries in a distributed environment puts forth many problems like Concurrent access, Query processing, File replication, Reliability, Scalability etc. These problems can be solved using Distributed and P2P technologies. Moreover there is no standard architecture for distributed UDDI. Many research works are described below in Table 1 which has proposed Distributed architectures to solve these problems.

Table 1

Summary of Distributed UDDI Architectures

SYSTEM INFRASTRUCTURE	PROTOCOL/ FUNCTIONS	ADVANTAGE
P2P-based Distributed UDDI Web Service Discovery (PDUS)[1]	CHORD Protocol	Performance Efficiency
Federated UDDI registry[2]	Timestamp revising function	Concurrent Access
A Novel Interoperable Model of Distributed UDDI[3]	Data real-time synchronization	Response Time
Distributed Web Service Directory[6]	Service Directory	Service Availability
CAN-based P2P network[7] [10]	CAN protocol	Scalability Stability
METEOR-S Web Service Discovery Infrastructure (MWSDI) [12]	Peer Initiation Protocol for Client and Operator	Registry federation
CHORD-based P2P Network	CHORD Protocol	Efficiency Accuracy
Peer-to-Peer network based service registry[15]	Edutella JXTA	Semantic WS Discovery
DHT-based P2P Network[16]	SHA-1 Algorithm DHT	Semantic WS Discovery
Chord Overlay Network[17	CHORD Algorithm	Scalability Flexibility
A decentralized WS discovery mechanism (DWSDM)[18]	CHORD Algorithm	Robustness Scalability
pService system for WS Discovery and Matching[19]	CHORD Protocol SKIP Graph	Scalability Query Processing

A federated UDDI system for service-oriented environment allows the system to easily process data process requirements from service providers and users in a correlated and synchronized way [2]. A P2P framework is adopted for deploying and managing federated services that run on federated systems spanning multiple collaborative organizations [7]. A replication mechanism is developed for the dynamic management of the content of several Universal Description, Discovery, and Integration (UDDI) registries [8]. These replication mechanisms are intended to be deployed in an environment of Web. A novel discovery algorithm is introduced to address semantic heterogeneity with respect to multiple ontologies from the same domain. A framework is also provided for web service discovery within a federation [9].

The main problems of UDDI are centralized management, passive service directory, bad interoperability etc. An interoperable model of distributed UDDI is discussed to enhance the query efficiency, availability, and offer the way to improve information exchange among users [3]. Coordinating UDDI public registries is a challenging task. A lightweight internode operation is available for replicating the APIs to distribute the entries across a Distributed Hash Table (DHT) [4]. It solves the communication problem to a certain extent. Discovering web services in a distributed UDDI environment becomes a more critical aspect than in centralized UDDI. P2P is such an infrastructure, which allows the discovery of resources by distributing the functions of the UDDI among all the peers in the P2P network [10]. The federation of service registries allows discovery of advertisement entries in one registry by service consumers through remote registries [11].

The METEOR-S Web Service Discovery Infrastructure (MWSDI) is proposed for providing transparent access to private and public Web service registries [12]. The focus of this paper is the creation of registry federations which allows businesses to share their data while maintaining their privacy. Toshiba and Carnegie Mellon University joined together to develop a Matchmaker mechanism that enhances UDDI's search functionality through the use of semantic information [13]. A two-tier service registration management system is constructed using ontology reasoning and it extends the centralized UDDI by distributing the publishing and discovery of semantic web services in chord ring [14]. A distributed discovery service, based on a peer-to-peer infrastructure is proposed which uses DAML-S service descriptions to provide enhanced semantic search capabilities [15].

To overcome the problems in discovery of Semantic Web Service based on centralized UDDI registry, an efficient algorithm to publish and discover semantic web services in DHT (Distributed Hash Table) based on decentralized systems [16] . A distributed architecture for SWS registry is proposed by adopting P2P technology to organize a large number of SWS Registries into a network so that they can cooperate with each other [17]. A decentralized web services discovery mechanism (DWSDM) is propose which uses distributed hash table (DHT) structure to solve the centralized UDDI problems [18]. A P2P overlay network for web service discovery and matching system is proposed which uses WSDL-S documents for service description [19]. This architecture uses CHORD protocol for routing the queries. An active and distributed UDDI architecture called Ad-UDDI [20] is proposed to organize the private or semi-private UDDI based on industry classifications. An active monitoring mechanism is adopted in Ad-UDDI which promotes automatic updation of service information and convenient search of latest service information.

A scalable distributed web services discovery architecture called DUDE (Distributed UDDI Deployment Engine) [21] is proposed to solve the scalability issue and to find services across multiple registries. Distributing UDDI functionality is achieved by using Distributed Hash Table (DHT) technology which makes UDDI more robust and scalable. Middleware architecture [22] is proposed for organizing Web services in community based on business specific domains. This model supports an infrastructure of Distributed UDDI registries with compatible service

registering and composition. A flexible Hybrid registry deployment architecture [23] is proposed for Management of

Distributed UDDI Registries. This model is based on meta-directory system which uses a communication overlay for interconnection among UDDI nodes.

6. CONCLUSION

We have seen that, a large amount of research works has already explored the issues related to Distributed UDDI. The combination of distributed architecture and UDDI mechanisms seems to be a promising methodology and a number of alternate solutions also have been proposed. Each of these solutions has its own merits and demerits, but none of them is satisfactory with respect to bandwidth efficiency and query expressiveness requirements, simultaneously. There are different distributed computing infrastructures which can be used to distribute the functionalities of UDDI. P2P architecture is one among the technologies which effectively distributed the functions of UDDI. Using a partially decentralized P2P architecture would make the network efficient and flexible. We can use OWL language for describing the web services which facilitates efficient and semantic enabled discovery. For decentralizing UDDI, the multiple copies of the entire directory information are stored at different network locations. This hybrid directory structure provides better bandwidth consumption, scalability, and fault-tolerance. As explained in Table I, all possible combinations of architectures and distributed technologies have not been explored yet. Hence, future research in distributed UDDI should focus on unveiling the unexplored alternatives in order to realize the promise for an efficient and flexible distributed discovery of web services in large scale distributed UDDI.

7. REFERENCES

[1]. Ni Yulin, Si Huayou, Li Weiping, Chen Zhong "PDUS: P2P-based Distributed UDDI Service Discovery Approach" 2010 International Conference on Service Sciences.

[2]. Qianhui LIANG, Jen-Yao CHUNG "A Federated UDDI System for Concurrent Access to Service Data" IEEE International Conference on e-Business Engineering.

[3]. Libing Wu, Yanxiang He, Dan Wu, Jianqun Cui "A Novel Interoperable Model of Distributed UDDI" International Conference on Networking, Architecture, and Storage.

[4]. Roberto Podesta' France roberto.podesta@inria.fr "A Lightweight Inter-node Operation for UDDI Cloud" IEEE conference paper.

[5]. Shoujian Yu, Qin Zhu, Xiaoling Xia, and Jiajin Le "A Novel Web Service Catalog System Supporting Distributed Service Publication and Discovery" Proceedings of the First International Multi-Symposiums on Computer and Computational Sciences (IMSCCS'06).

[6]. Ming-Qiang Guo, Ying Huang, Xian-Gang Luo, Yong Liu "Design and Implementation of a Distributed Web Service Directory in SOA-oriented Urban Spatial Information Sharing Platform" 2009 Second Asia-Pacific Conference on Computational Intelligence and Industrial Applications.

[7]. Fu-zhen Sun, Shen-gang Hao "A Discovery Framework for Semantic Web Services in p2p Environment" 2010 International Conference on Electrical and Control Engineering.

[8]. Zakaria Maamar "Replication mechanism over a set of Distributed UDDI Registries".

[9]. S.Oundhakar, K.Verma, K.Sivashanugam, A.Sheth and J.Miller "Discovery of Web Services in a Multi-Ontology and Federated Registry Environment", International Journal of Web Services Research.

[10]. Nizamuddin Channa, Shanping Li, Wei Shi and Gang Peng, "A CAN-Based P2P Infrastructure for Semantic Web Services Publishing and Discovery", IEEE Conference 2005.

[11]. Pornpong Rompothong and Twittie Senivongse, "A Query Federation of UDDI Registries", Chulalongkorn University-Industry Linkage Research Grant Year 2002.

[12]. Kaarthik Sivashanmugam, Kunal Verma, Amit Sheth, "Discovery of Web Services in a Federated Registry Environment", Large Scale Distributed Information Systems (LSDIS) Lab, University of Georgia.

[13]. Takahiro Kawamura, Tetsuo Hasegawa, Akihiko Ohsuga, Massimo Paolucci, and Katia Sycara, "Web Services Lookup: A Matchmaker Experiment", IEEE Computer Society, 2005.

[14]. Bingfei Nan, Shuzhi Li, Shuxing Yang, "The Study of Distributed Semantic Web Services' Publishing and Discovery Mechanism", 2010 International Conference on Communications and Intelligence Information Security.

[15]. Jing Zhong, Hong Ying, "A Semantic Web Based Peer-to-Peer Service Registry Network", First International Conference on Semantics, Knowledge, and Grid, IEEE Computer Society, 2006.

[16]. Xin Wang, Chen Liu, Zhengqiu Yang, "An Efficient Semantic Web Service Discovery Algorithm in DHT-based P2P Network", 2009 First International Conference on Future Information Networks, IEEE Computer Society.

[17]. Zhao Yun, Si Huayou, Qi Hengnian, Ni Yulin, "An Approach to Discover Semantic Web Services in Distributed Environment based on chord", IEEE Computer Society, 2010.

[18]. Quanhao Lin, Ruonan Rao, Minglu Li, "DWSDM: A Web Services Discovery Mechanism Based on a Distributed Hash Table", Fifth International Conference on Grid and Cooperative Computing Workshops, IEEE 2006.

[19] Weifeng Lv, Jianjun Yu, "pService: Peer-to-Peer based Web Services Discovery and Matching", Second International Conference on Systems and Networks Communications, IEEE 2006.

[20]. Zongxia Du, Jinpeng Huai, Yunhao Liu, "Ad-UDDI: An Active and Distributed Service Registry" Hong Kong Univ. of Science and Technology, Hong Kong

[21]. Sujata Banerjee, Sujoy Basu, Shishir Garg, Sukesh Garg, Sung-Ju Lee, Pramila Mullan, Puneet Sharma, "Scalable Grid Service Discovery Based on UDDI" ACM Transaction, 2005.

[22]. Emil Stanescu, Ileana Stanescu, Victor Popa, "Distributed Infrastructure for Semantic Web Services", MGC'05, ACM.

[23]. Afiya Kassim, Babak Esfandiari, Shikharesh Majumdar, Laura Serghi, "A Flexible Hybrid Architecture for Management of Distributed Web Service Registries", Fifth Annual Conference on Communication Networks and Services Research, IEEE-2007.

[24]. E.Michael Maximilien, Munindar P.Singh, "Agent-based Architecture for Autonomic Web ServiceSelection", IBM Corporation and North Carolina State University.

[25]. T. Rajendran, Dr.P. Balasubramanie, "An Optimal Agent-Based Architecture for Dynamic Web Service Discovery with QoS", 2010 Second International conference on Computing, Communication and Networking Technologies.

[26] UDDI Consortium. UDDI Technical White Paper, 2002. [Online]. Available: http://www.uddi.org/pubs/Iru_UDDI_Technical_White_Paper.pdf.

[27] Sven Overhage, "On Specifying Web Services Using UDDI Improvements", Darmstadt University of Technology, Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.81.3223.

[28] W3C Member Submission, "Web Service Modeling Ontology (WSMO)", Available: http://www.w3.org/Submission/WSMO

[29] W3C Member Submission, "OWL-S: Semantic Markup for Web Services", Available: http://www.w3.org/Submission/OWL-S