WebPeer: A P2P-based System for Publishing and Discovering Web Services

Ruixuan Li, Zhi Zhang, Zhigang Wang, Wei Song, Zhengding Lu
College of Computer Science and Technology,
Huazhong University of Science and Technology,
Wuhan, 430074, P. R. China
rxli@hust.edu.cn

Abstract*

The use of Web Services as an infrastructure of service sharing has made it possible to provide collaboration and interoperability in distributed computing environment. In this environment, service publishing and discovery are required as elementary functionalities for users to be able to locate the shared resources. The mechanism of service publishing and discovery with centralized architecture restricts the reliability and scalability of the distributed computing environment as the services and resources on the web are fast emerging. The peer-to-peer (P2P) systems and applications, on the other hand, employ distributed resources to perform critical functions in a decentralized manner. This paper introduces Web Services Oriented Peer-to-peer (WSOP) architecture with a combination of centralized and decentralized characteristics, and presents a framework of service publishing and discovery model based on WSOP architecture. The prototype system - WebPeer implemented based on this model demonstrates the WSOP architecture can not only help to overcome the known obstacles in common Web Services infrastructure such as single node failure, but also extend the ability of the pure P2P systems, such as more efficiently locating the resources, increasing the interoperability between different P2P systems.

Key words: *Peer-to-Peer, Web Services, Service Publishing, Service Discovery, Interoperability*

1. Introduction

Web Services technologies have received great development in recent years. The Web Services architecture is the classic client/server model [1]. The service provider (server) registers with the UDDI registry and the consumer (client) contacts the registry to discover the server location so that it can interact with it. The use of Web Services as an infrastructure of service sharing has made it possible to provide collaboration and interoperability in distributed computing environment [2]. In this environment, service publishing and discovery are required as elementary functionalities for users to be able to locate the shared resources. However, the mechanism of service publishing and discovery with centralized architecture, such as UDDI, restricts the reliability and scalability of the environment as the services and resources on the web are fast emerging [3].

The peer-to-peer (P2P) systems and applications, on the other hand, employ distributed resources to perform critical functions in a decentralized manner. Peer-to-peer computing is the sharing of computer resources and services through direct communication between systems. Each functional unit in the network, called a peer, behaviors similarly and is logically capable of both providing and consuming resources [4]. Compared with a centralized system, a P2P system provides an easy way to aggregate large amounts of resources residing on the edge of Internet or in ad-hoc networks with a low cost of system maintenance. Using P2P-based technologies to publish and discover Web Services will extend the reliability and scalability of the current distributed computing architecture [5].

This paper introduces Web Services Oriented Peer-to-peer (WSOP) architecture with a combination of centralized and decentralized characteristics, and presents a framework of Web Services publishing and discovery model based on WSOP architecture. In the WSOP-based system, the node providing web services act as a peer and each peer can request web services

* This work was partially supported by National Natural Science Foundation of China under Grant 60403027, National Key Technologies R&D Program of China under Grant 2002BA103A04, and R&D Program of Hubei Provincial Department of Education under grant 2003A011.

from other peers. Typically, a peer is all three of the aforementioned roles: service provider, service consumer and service registry. However, like Web Services, peers in the WSOP-based system should also publish a service to allow it to be found and accessed by other peers and discover a service with efficient precision when they have certain requirements. Service publishing and discovery are paramount for both infrastructures; however, the approach in WSOP-based system is different from the conventional Web Services.

The rest of this paper is organized as follows: section 2 discusses the related work on Web Services and peer-to-peer system. Section 3 introduces Web Services Oriented Peer-to-peer (WSOP) architecture. Section 4 presents our approach to service publishing and discovery model based on WSOP architecture. The implementation work of the WSOP-based WebPeer system is given in Sections 5. In the last section, we present conclusions and directions for future work.

All manuscripts must be in English. These guidelines include complete descriptions of the fonts, spacing, and related information for producing your proceedings manuscripts. Please follow them and if you have any questions, direct them to the production editor in charge of your proceedings at the IEEE Computer Society Press: Phone (714) 821-8380 or Fax (714) 761-1784.

2. Related work

Web Services and peer-to-peer technologies widely emerged during the last several years and these two diagrams tend to be polled together in the recent researches. Recent work in content-based search include content-addressable networks – where the content of queries is used to efficiently route messages to the most relevant peers – such as CAN [6], Pastry [7] and Chord [8] as well as some variations of publish/subscribe networks [9]. These content-based P2P networks place emphasis on locating and distributing the contents rather than on a logical organization of the system architecture and on publishing and discovering Web Services through the peer-to-peer network.

Several other projects concern the combination of web services and peer-to-peer networks where [10] gives a general overview and a classification of P2P based web services. An example is Self-Serv [11], a platform for rapid composition of Web Services in a peer-to-peer environment. It focuses on the Web Services composition through using peer-to-peer technology rather than on publishing and discovering the services efficiently in P2P-based system. Peermetrics [12] is a Java-based peer-to-peer development platform for implementation of distributed services with dynamic protocol binding, resource identification, peer identification.

Another project should be mentioned is JXTA, the P2P framework initiated by Sun Microsystems [13]. JXTA is making adjustments to its core platform to make peers interoperate with web-services using protocols like SOAP and WSDL. It is not based on a publish/subscribe and event notification scheme. Its search peers act as hubs and can register with other hubs as information providers to field queries from other peers based on arbitrary content description registrations. JXTA tends to centralize registration/subscription information and control in hubs. In JXTA, peers do not build up peer groups and use the super peer to locate each other or communicate with one another [14]. In our approach, we employ the JXTA as the basic infrastructure of the peer-to-peer computing environment, and each peer joins a peer group (constantly changing) and stores some minimal information about other peers.

3. Web Services oriented peer-to-peer (WSOP) architecture

We present a Web Services oriented peer-to-peer (WSOP) architecture based on the integration of different peer-to-peer systems and Common Web Services (CWS) with the SOAP (Simple Object Access Protocol) connectivity (see Figure 1).

In our approach, peers residing as the neighbors on the same P2P network are pulled together to form a peer group [15]. Usually, the peers in the same peer group have the same interests or come from the same organizations. There is at least one super peer in each peer group. Super peers, the special control nodes, are responsible for managing and mediating the other nodes and providing security services in their own peer group. They also maintain a Local Service Registry Broker (LSRB), providing the fast service registration and invocation in the peer group environment (e.g., independent organizations or corporations). The CWS, hosted on the SOAP server, consists of service provider, service requestor and Common Service Registry Broker (CSRB). The CSRB provides access to a P2P network interconnecting nodes (i.e. super peers) in different physical networks using different transport protocols and maintains the mappings of service descriptions between CSRB and LSRB.

When a peer wants to publish its services, it should register the services to LSRB on the super peer in its

peer group. Some peers may only want to share their services within the peer group, and it's no need to register the services onto CSRB for these peers. While service mappings among different LSRBs should be built if the peers would like to provide their services to those peers in other peer groups. For this case, the super peer should register the services in CSRB so that other peers can discover and connect to them.

When a peer wants to find a certain service, the requests will first be delivered to the super peer in its own peer group. The super peer will look up the LSRB on its own site. The service description will be returned back to the requestor if the search is exactly matched. Otherwise, the request will be forward to the CSRB and the mapping in the CSRB will indicate the service provider's information. Then the requestor will get the service description from the CSRB. After that, the requestor will connect and bind the peer directly to obtain the services.

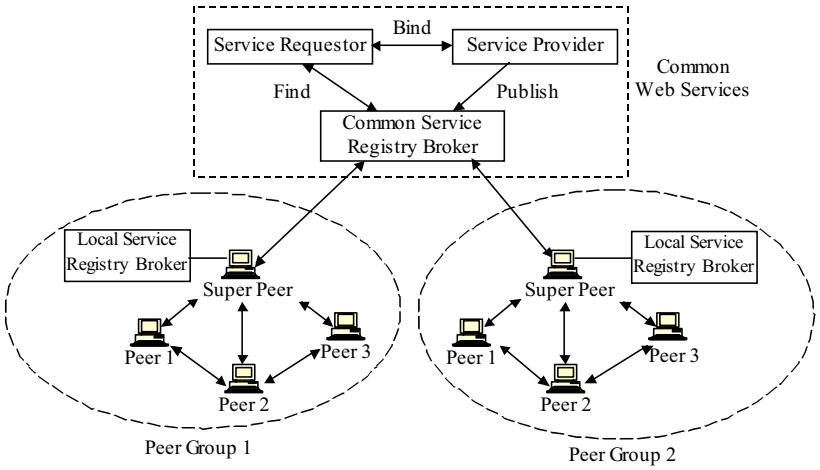

Figure 1. Web Services oriented peer-to-peer (WSOP) architecture

An important advantage of this architecture would be the flexibility of registering new services to the system via employing the super peers and LSRBs. It will definitely reduce the heavy load of the CSRB when the number of nodes and services in the environment are very large. Another advantage is that service request delivering would not be flooded in the whole systems.

4. Service publishing and discovery model based on WSOP architecture

4.1. The system framework

In today's distributed computing environment, service publishing and discovery are the two basic functionalities for users (peers) to share and locate the required resources. The fashions of service publishing and discovery in the WSOP-based architecture are different from the conventional web services infrastructure. In traditional manner, the mechanism of service publishing and discovery are centralized through using common UDDI. The WSOP-based architecture will employ the peer-to-peer technologies and decentralize the UDDI service directory. Each peer in the P2P overlay network plays the roles of service provider and service consumer. Peers with the similar interests are polled together to form a peer group. The super peers in the peer group will take the responsibility for service registry in the group. So, the peer will take most of the work for service publishing and discovery.

Figure 2 illustrates the framework of service publishing and discovery model based on WSOP architecture. It gives the basic infrastructure of the P2P network. Some nodes physically residing in the Internet are logically pulled together to form a P2P overlay network. The peers with the same interests live in the same peer group. The figure mainly shows the architecture of the peer node. It can be divided into two parts: one is the peer management; the other is the Web Services processing. Each peer maintains a security policy base. The security policy module provides the authentication policies for the P2P

communication and the access control policies for the Web Services processing. Each peer has a local SOAP server to deal with the SOAP messages sent from other peers. It can also communicate with the remote SOAP server when it wants to utilize the Web Services located on the other peers.

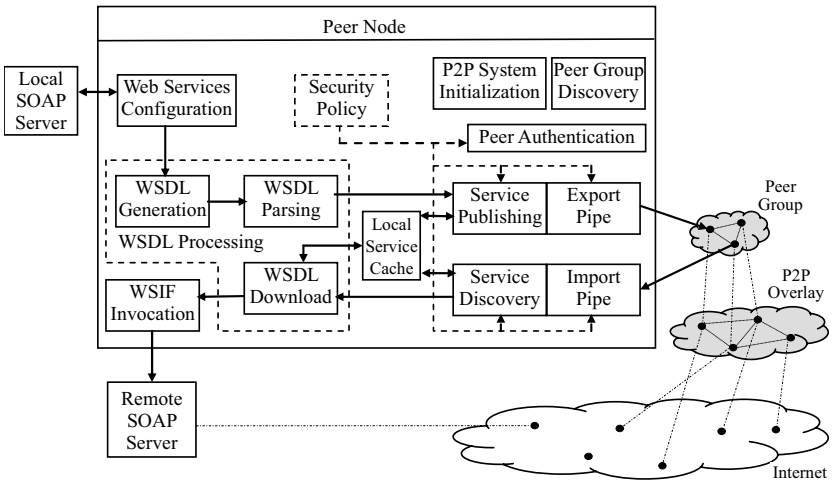

Figure 2. The framework of service publishing and discovery model based on WSOP architecture

4.2. The peer management

The peer management module consists of P2P System Initialization, Peer Group Discovery and Peer Authentication. These three parts provide the basic P2P functions for each peer.

P2P System Initialization. The initialization step will set up the network environment of the fundamental P2P system platform, such as the peer name, login and password, HTTP address and TCP port, rendezvous peers and relay peers. While the P2P system platform is initialized, the peer will join the default peer group according to the predefined configurations.

Peer Group Discovery. Generally, the peer group in the WSOP-based system will be self-organized according to the peer's interests or the independent organizations (e.g. the ticket booking group). The peer can search and join the existing peer groups or create a new peer group in order to share and obtain the certain Web Services (e.g. looking up the airlines, booking the tickets).

Peer Authentication. The peer can freely connect those peer groups with no certificate requirements. However, some peer groups may require the authorized certificate when a peer wants to connect them. This can be done through using Peer Member Protocol (PMP) or other third party certificate authentication (CA) server.

4.3. Web Services processing

The Web Services processing module includes five parts: Web Services Configuration, WSDL Processing, Service Publishing, Service Discovery and WSIF Invocation. The functions of each part are as follows.

Web Services Configuration. To provide services, the peers can install different types of local SOAP servers, such as Apache SOAP, AXIS, JBoss, Glue, etc. Moreover, these SOAP servers can be hosted on different service containers, such as Apache Tomcat, BEA WebLogic Application Server, IBM WebSphere, etc. This will result in the heterogeneity of Web Services Deployment. For example, the service ports are different, with 8080 for AXIS and 8083 for JBoss. The Web Services Configuration module provides a uniform interface to configure the Web Services parameters and prepares for generating the Web Services description documents (i.e. WSDL documents).

WSDL Processing. WSDL Processing includes the function modules of WSDL Generating, WSDL Parsing and WSDL Download. WSDL documents can be generated using Java2WSDL tools and parsed by

WSDL4J tool kits. The core parameters for publishing the Web Services in the WSOP-based system can be obtained from parsing the WSDL documents, such as portType, operation, message and location. The file sharing mechanism of P2P system can be used as the WSDL documents downloading methods.

Service Publishing. When a peer wants to publish a service, a Web Services advertisement containing the service information should be published. An Export Pipe will be created to transfer the local Web Services responses. In addition, the peer also maintains the local service cache, including saving the published Web Services advertisements, and eliminating the lifetime-ended advertisements.

Service Discovery. The main functions of the service discovery module are as follows: Looking up the discovered Web Services advertisements in the local service cache; Discovering new Web Services advertisements in the WSOP-based system through using static and dynamic discovery methods; Creating Import Pipes to receive other peers' requests of service discovery; Processing the Web Services responses received by Import Listener; And caching the discovered valuable Web Services advertisements.

WSIF Invocation. Once the services are discovered, the peer can invoke and utilize the service through WSIF (Web Services Invocation Framework) protocol. Usually, these services are located on the remote SOAP servers in other peer groups (because the service advertisements in the local SOAP server have been cached). The WSIF provides a standard API to access the Web Services described with WSDL language. It uses the same programming model whatever the Web Services are implemented through using WSIF protocol. It can also access the services dynamically generated without stubs.

4.4. Advantages of the architecture

The service publishing and discovery model based on WSOP architecture has following advantages:

- **Reliability and robustness.** There is no single-node-failure problem in the WSOP-based system. Even if there are some peer nodes crashed, the service discovery process will still be put forward through other alive peers. If the service provider (peer) goes down (e.g. getting offline or having some communication problem), it will be disconnected from the peer group automatically and notify other peers in the group. Other peers may cache the Web Services advertisements published by this peer, and the services are still capable for the future's invocation and utilization.
- **Scalability and effectiveness.** The performance of the WSOP-based system will be improved with the increasing of the number of the peers in the system. That is because the Web Services advertisements will be cached in all the peers they are past through. The advertisements intend to move to the final users (peers) and the peers don't need to connect a centralized server or even the service providers to get the advertisements. This will greatly accelerate the service discovery process. Another reason is there will be more peers giving hands to discover the services while more peers join the system. The final users can get more valuable Web Services through these effective ways. The high edge service computing capability of the WSOP-based system can also make full use of the storage spaces on the network and gains high scalability.
- **Security.** In the WSOP-based architecture, the peer groups are formed according to the users' interests and each peer maintains its own Web Services. That is, the Web Services are decentralized through employing the P2P network and it will cut down the security problem of the centralized UDDI servers, such as centralized denial of services (DoS) attack. The risks of the e-business transactions on the Internet will also be reduced through the WSOP-based system while the different Web transactions interoperate between the different peers, not through the centralized servers, and they are much harder to be traced and recorded.

5. The WebPeer System

According to the above WSOP architecture, we have implemented WebPeer - a Web Services oriented peer-to-peer system. The system uses Java as the programming language and employs the JXTA as the basic P2P system platform [16]. The following parts will give the main implementation codes and algorithms of the service publishing and discovery in the WebPeer system.

5.1. The implementation of service publishing

Figure 3 gives an instance of the Web Services advertisement - the Module Specification Advertisement (MSA) for the WebPeer system. Each MSA advertisement has an MSID, a name and WSDL URI as the parameters of the Web Services. It also gives the pipe advertisement that the MSA advertisement will use. The service advertisement contains the basic information of a service and

indicates where the Web Services can be found and how to obtain the service.

```
<?xml version="1.0"?>
<!DOCTYPE jxta:MSA>
<jxta:MSA xmlns:jxta="http://www.jxta.org">
  <MSID>urn:jxta:uuid-D110E7397F24401EA8318F383CFF29
    4035B8C3C1CF6645EABD13B9C76EBB115906</MSID>
  <Name>JXTASPEC:WebServices:urn:helloservice</Name>
  <Crtr> Example.org </Crtr>
  <SURI> http://www.example.org </SURI>
  <Vers> 1.0 </Vers>
  <Desc> A service allow you to say hello </Desc>
  <Parm>
    <wsdlURI>
      http://www.example.org/helloservice/
      helloservice.wsdl
    </wsdlURI>
  </Parm>
  <jxta:PipeAdvertisement xmlns:jxta="http://jxta.org">
    <Id>urn:jxta:uuid-2EC8CDF870744C468B7CB111E337A0
      1EE5E3818 F9BBD405B90D2B7626E1549C504</Id>
    <Type> JxtaUnicast </Type>
    <Name> WebServices:RespPipe:urn:helloservice</Name>
  </jxta:PipeAdvertisement>
</jxta:MSA>
```

Figure 3. An instance of the Web Services advertisement for the WebPeer system

5.2. The service discovery algorithms

```
private void discoverWebServicesAdvertisement() {
  while (the number of the discovery requests isn't 0){
    Looking up the Web Services Advertisements in
      the local service cache;
    if (The search result doesn't match the request){
      Sending the discovery request to the peer group;
      if (The search result matches the request)
        Saving the result in the local service cache;
      else {
        Sending the discovery request to the CSRB;
        if (The search result matches the request)
          Saving the result in the local service cache;
        else
          Return the result with no matches;
      }
    }
    return the discovery result;
  }
}
```

Figure 4. The algorithm of discovering the Web Services Advertisements in the WebPeer system

Since the Web Services are published through the service advertisements, the main task for discovering the Web Services is to find the appropriate advertisements and get the corresponding WSDL documents. Figure 4 shows the algorithm of discovering the Web Services advertisements in the WebPeer system. The peer will look up its local service cache for the certain advertisement. If it doesn't match, the discovery request will be delivered to the peer group and even to the CSRB. While finding the Web Services advertisement, the peer will get the WSDL documents indicated by the advertisement and request the Web Services through using the WSDL document. Figure 5 gives the algorithm of getting the WSDL documents in the WebPeer system.

```
private void getWsdlDocument() {
  while (peers.hasMoreElements()) {
    Getting a Web Services advertisement;
    if (the parameters of the advertisement arenot null)
    {
      Call doc(StructuredTextDocument) to create a
        structured text document;
      Saving the parameters of the advertisement into
        the document;
      elements := doc;
      while (elements.hasMoreElements()) {
        Getting the names of the subitems;
        if (the subitem is wsdlURI) {
          Getting the value of the wsdlURI;
          if (the value of the wsdlURI is not null) {
            return the value of the wsdlURI;
            break the inner loop;
          }
        }
      }
    }
    else
      repeat the outer loop;
  }
}
```

Figure 5. The algorithm of getting the WSDL documents in the WebPeer system

5.3. The system interface

The WebPeer system can run in any platform that supports the Java virtual machine. Figure 6 shows some of the screen shots of the running interfaces of the WebPeer system. It has four parts: Peer Group, Local Service, Remote Service and Peer information. With this system, the user (peer) will be able to create

a new peer group or select an existing group for joining, generate a WSDL document, look up generated WSDL documents, publish a new Web Service, obtain a list of the cached Web Services advertisements and utilize these services, submit a request and discover a certain service, and configure the parameters for the peer itself. We plan for this system to enable the user to manipulate e-business services such as ticket booking, sales and buying, etc.

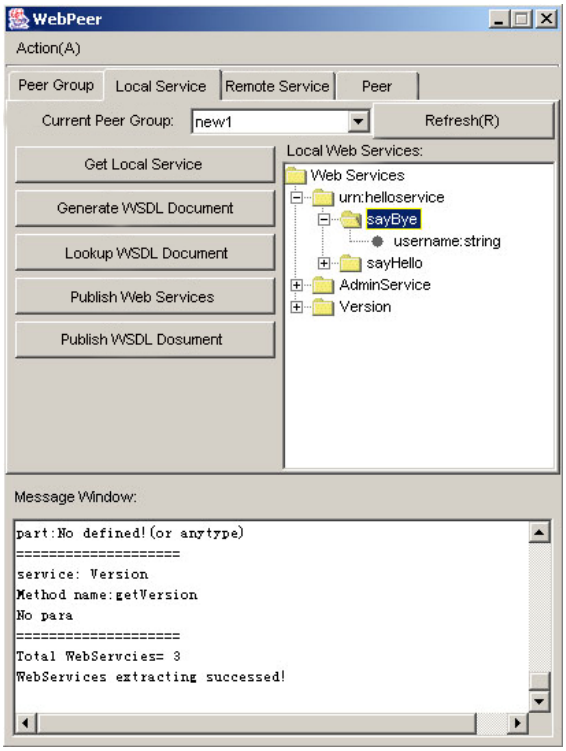

Figure 6. The WebPeer system interfaces

6. Conclusions

In this paper we highlighted key intersect points that enable using Web Services infrastructure and peer-to-peer technologies together and presented an architectural approach and implementation framework towards unifying them.

We introduced Web Services Oriented Peer-to-peer (WSOP) architecture with a combination of centralized and decentralized characteristics and presented a framework of service publishing and discovery model based on WSOP architecture. We also gave the main implementation codes and algorithms of the service publishing and discovery in the WSOP-based prototype - WebPeer system. The implementation work demonstrates the WSOP architecture can not only help to overcome the known obstacles in common Web Services infrastructure, such as single node failure, denial of service (DoS) attacks, but also extend the ability of the pure P2P systems, such as more efficiently locating the resources, increasing the interoperability between different P2P systems.

There are several advantages that the service publishing and discovery model based on WSOP architecture offers. These include simplicity and ease of use, openness, reliability, scalability and security. In the future, we plan to develop a semantic-based model for service publishing and discovery in the WebPeer system.

References

[1] R. Perrey, and M. Lycett, "Service-Oriented Architecture", In: Proceedings of the 2003 International Symposium on Applications and the Internet Workshops (SAINT 2003), Orlando, FL, USA, IEEE Computer Society, 2003, pp. 116-119.

[2] M.P. Papazoglou, B.J. Kramer, and J. Yang, "Leveraging Web-Services and Peer-to-Peer Networks", In: Proceedings of the 15th International Conference on Advanced Information Systems Engineering(CaiSE2003), Lecture Notes in Computer Science, Vol. 2681. Springer-Verlag, Berlin Heidelberg New York, 2003, pp. 485-501.

[3] C.T. Qu, and W. Nejdl, "Interacting the Edutella/JXTA Peer-to-Peer Network with Web Services", In: Proceedings of the 2004 International Symposium on Applications and the Internet, 2004, pp. 67-74.

[4] D.S. Milojicic, V. Kalogeraki, R. Lukose, et al, "Peer-to-Peer Computing", HP Laboratories, Technical Report HPL-2002-57, Palo Alto, California, USA, 2002, pp. 2-6.

[5] B. Benatallah, et al, "Declarative Composition and Peer-to-Peer Provisioning of Dynamic Web Services", In: Proceedings of International Conference on Data Engineering (ICDE 2002), IEEE Press, 2002, pp. 297-308.

[6] S. Ratnasamy, P. Francis, M. Handley, et al, "A Scalable Content Addressable Network", In: Proceedings of the ACM SIGCOMM 2001 Technical Conference, San Diego, CA, USA, 2001, pp. 161-172.

[7] A. Rowstron, and P. Druschel, "Pastry: Scalable, Distributed Object Location and Routing for Large-Scale Peer-to-Peer Systems", In: Proceedings of the 18th IFIP/ACM International Conference on Distributed Systems Platforms (Middleware 2001), Lecture Notes in Computer Science, Vol. 2218.

Springer-Verlag, Berlin Heidelberg New York, 2001, pp. 329-350.

[8] I. Stoica, R. Morris, D. Liben-Nowell, et al, "Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications", IEEE/ACM Transactions on Networking, 2003, 11(1): 17-32.

[9] D. Heimbigner, "Adapting Publish/Subscribe Middleware to Achieve Gnutella-like Functionality", ACM Symposium on Advanced Computing (SAC): Special Track on Coordination Models, Languages and Applications, 2001, pp. 176-181.

[10] C. Schmidt, and M. Parashar, "A Peer-to-Peer Approach to Web Service Discovery", World Wide Web archive, 2004, 7(2): 211-229.

[11] B. Benatallah, Q.Z. Sheng, and M. Dumas, "The Self-Serv Environment for Web Services Composition", IEEE Internet Computing, 2003, 5(3): 40-48.

[12] "Peermetrics: Technical Overview", http://www.peermetrics.com/peer_system-0.8/docs/guide/technical_overview.html, 2003.

[13] S. Botros, and S. Waterhouse, "Search in JXTA and other Distributed Networks", In: Proceedings of the 2001 International Conference on Peer-to-Peer Computing, 2001. pp. 30-35.

[14] S.R. Waterhouse, D.M. Doolin, G. Kan, et al, "JXTA Search: a Distributed Search Framework for Peer-to-Peer Networks", IEEE Internet Computing, 2002, 6: 68-73.

[15] B. Yang, and H. Garcia-Molina, "Designing a Super-peer Network", In: Proceedings of the 19th International Conference on Data Engineering (ICDE), Bangalore, India, IEEE Computer Society, 2003, pp. 49-60.

[16] L. Gong, "JXTA: A Network Programming Environment", IEEE Internet Computing, 2001, 5(3): 88-95.