Abstract-The enormous growth of Peer-to-Peer networks have evolved new breed of applications. A Web Service is a middleware technology which supports interoperability among the heterogeneous systems. The power of Web Service when coupled with Peer-to-Peer networks can make Web Services accessible by any small group of users to a very larger group of peers constructed over Internet. This approach enables peer-to-peer distributed Web Service discovery within the peer group of common interest. Every peer is allowed to publish and discover Web Service without central coordination. The information on Web services will be available to all the peers within the common interest peer group. When an end-user connects to the network via one peer, he/she will be able to view the list of Web Services and access them across all the peers relevant to his/her request. The objective of our work is to implement a distributed peer-to-peer Web Service discovery system for student community which would enable the students to chat, share, access materials of common interest and group documents in peer student’s system. Moreover, we have proposed Distributed Hash Table (DHT) to search Web Services available in the peer group. This would increase the search efficiency of the system. Our System is implemented in .Net on Window XP service pack 3 platform with Windows Communication Foundation (WCF) which supports easy construction of peer-to-peer applications.

Keywords: Peer-to-Peer Systems, Web service, Windows Communication Foundation (WCF), Distributed Hash Table (DHT)

I. INTRODUCTION

Web Services is a loosely coupled application which provides interface to share resources over the Web. Most of the Web Services were dependent on central registry like UDDI as specified in Service Oriented Architecture (SOA) [14]. This central registry was distributed over time period. However the tremendous growth in number of Web Services, registry size, duplication of Services and maintenance of these Services restricted the competing companies to withdraw[2]. Another, new dimension which arose over time is Peer to Peer Web Service Discovery. It is still, in its nascent stage. This provides room for new developments in this direction.

Gartner [1] has discussed about the growth and the application of Service Oriented Architecture (SOA) in detail. The emerging trend of Software, SaaS (Software-as-a-Service) will come under the roof of SOA paradigm. This facilitates a large number of Web Services to be available on the Internet, which, in turn necessitates the search of Service discovery. Service discovery provides information of Service composition, binding, sharing, and reusing Web Services.

The P2P networks discussed in [11] [12] is a kind of self-organizing network, in which every node is of equal capability. (i.e.) The same node can act as a server and a client depending on the needs, participate in peer activities and also uses the required services available across the peers. The paper also explores on how Distributed Hash Table (DHT) technology can be leveraged to develop scalable Web Service discovery.

A decentralized Web Service discovery mechanism based on DHT established on Peer to Peer network would be a perfect solution for Web Service provision. Combining the advantage of Web Service and P2P technology in a decentralized approach will improve the overall utilization of Web Services.

The rest of the paper is organized as follows: section 2 gives a brief overview of the related work carried out on Web Service, Peer-to-Peer Networks, WCF, JXTA. Section 3 discusses WCF and P2P implementation using WCF. Section 4 elaborates our proposed architecture which will be subsequently followed by Implementation and results in section 5. Section 6 explains an application of the system to student community. Finally, Section 7 concludes the work carried out by us.

II. RELATED WORK

Web Service is applied in many fields like e-Business, e-Government, e-learning and so on. Web Service is an efficient method of data and information integration across the Web. The XML and SOAP like protocols along with Web Services makes distributed computing more appealing. P2P is an emerging paradigm in the field of distributed computing. P2P paradigm combined with Web Service can become a scalable and viable platform for many e-Services. Information pertaining to Service descriptions are published in the discrete registry of peer nodes. [3] Therefore it would be a composition of Web Services offered by all the peers in the group.

P2P network can be formed using simple multicasting or via intended P2P distributed application support technologies like WCF and JXTA. However, Web Service built on P2P architecture using JXTA [4],[5],[8] requires Web Service Broker (WSB). This is due to difference in underlying technologies. In JXTA based P2P network, each peer acts as a hub and registers with other hubs (peers). Each peer may
join or leave the peer group at any time and expected to maintain minimal information about the other hubs (peers).

Web Service is a loosely coupled service; hence the Service consumer cannot get service unavailability intimation (on unavailability or failure) which causes invocation error. [7] This invocation error can be avoided through dynamic invocation of service. In Dynamic invocation of Web Service, the Service consumer can automatically choose other Web Services of same functionality which failed to invoke. This masks the unavailability of a particular failed service and continues to serve the consumer rather than ceasing the operations. Therefore, the Web Service invocation stays more reliable and establishes a favourable environment for wide use and acceptability of Web Services. The foremost task would be to compose Web Services with the same functionality [5] among the peers.

III. WCF AND P2P TECHNOLOGIES

A. WCF Implementing Technology of P2P:

The global acceptance of Web services, which includes standard protocols for application-to-application communication, has changed software development. For example, the functions that Web services now provide include security, distributed transaction coordination, and reliable communication. The benefits of the changes in Web services should be reflected in the tools and technologies that developers use. Windows Communication Foundation (WCF) is designed to offer a manageable approach to distributed computing, broad interoperability, and direct support for service orientation.

WCF simplifies development of connected applications through a new service-oriented programming model. WCF supports many styles of distributed application development by providing a layered architecture. At its base, the WCF channel architecture provides asynchronous, untyped message-passing primitives. Built on top of this base are protocol facilities for secure, reliable, transacted data exchange and broad choice of transport and encoding options.

The typed programming model (called the service model) is designed to ease the development of distributed applications. .NET Framework remoting, and Enterprise Services, are the earlier distributed computing technologies. The WCF service model features a straightforward mapping of Web services concepts to those of the .NET Framework common language runtime (CLR), including flexible and extensible mapping of messages to service implementations in languages such as Visual C# or Visual Basic. It includes serialization facilities that enables loose coupling and versioning, and it also provides integration and interoperability with existing .NET Framework distributed systems technologies such as Message Queuing (MSMQ), COM+, ASP.NET Web services, Web Services Enhancements (WSE), and a number of other functions.

An Excerpt of WCF Specification is illustrated below:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <system.serviceModel>
    <bindings>
      <netPeerTcpBinding>
        <binding name="PeerConfig">
          <security mode="None"/>
          <resolver mode="Pnrp"/>
        </binding>
      </netPeerTcpBinding>
    </bindings>
    <client>
      <endpoint name="TestMesh" address="net.p2p://xxxx" binding="netPeerTcpBinding" bindingConfiguration="PeerConfig" contract="ConsoleApplication_WCF_Sample.ITestMesh"/>
      ...
    </client>
    ...
  </system.serviceModel>
</configuration>
```

The interface comprises the description of service model and a binding is made up of a collection of binding elements. Each element describes some aspect of how the endpoint communicates with clients. In the above excerpt netPeerTcpBinding is used. Every endpoint has an address associated with it, which is used to locate and identify the endpoint. This address consists primarily of a Uniform
Resource Identifier (URI), which specifies the location of the endpoint in the above tested sample of console application the location specified is “net.P2P://xxxx”, The endpoint address is represented in the Windows Communication Foundation (WCF) programming model by the Endpoint Address class, which contains an optional Identity property that enables the authentication of the endpoint by other endpoints that exchange messages with it.

B. Web Service Technology

Web Service adopts SOA (Service Oriented Architecture). It implements service transfer using the entity interaction of service provider, service requestor and service register center etc. An intact Web Service process should include three phases: issuing service description, search for special service and binding service according to service interface description.

The related technologies of Web Service include SOAP (Simple Object Access Protocol) and WSDL (Web Service Description Language).

IV. PROPOSED ARCHITECTURE

Using the below framework user can efficiently access the web services from the peers that are connected across a particular network of common interest. Each peer will maintain a list of web services of its own and the other peers and can consume the Web Services from other peers that are connected across that peer group. The Web Services will persist among the connected peers. A DHT is maintained to efficiently store and retrieve web services using its key identity. The DHT that resides in each peer will contain Hash table and the generation of Hash key to index the table and encrypted value is taken as a Hash value. The user requested key will be taken for a match with the encrypted Hash value. Upon a successful match, the respective Web Service is sent as a response. Now, user can access it. This type of search will improve the efficiency.

V. IMPLEMENTATION AND RESULTS

A. Emulation of P2P Network Platform

We have implemented structured P2P network built using Distributed Hash Table (DHT). Each peer node will have Web Services listing of its own peer and also of other peer that are connected across the network. The DHT helps in efficient retrieval of user requested Web Services. Coding is done in Microsoft Dot Net Framework 3.5 with Peer Name Resolution Protocol (PNRP) support. The main components of this emulator are as follows: (1) Peer Connection Establishment (2) Web Service Discovery (3) Distributed Hashing to Store and Retrieve web service. These components and their interactions are outlined below.

Peer Connection Establishment will setup the wireless network connection across the peers. The identification of available peers in the particular peer network is done by PNRP (Peer Name Resolution Protocol). In this application, the peer identification and communication between the peers without servers is possible only by the PNRP (Peer Name Resolution Protocol) which enables dynamic name publication and resolution, and requires IPv6. Basic PNRP functionality is provided in the System.Net.PeerToPeer namespace classes. It provides the ability to create a peer and register it in a cloud of peers. It also provides the ability to resolve a Peer Name to its network endpoint, TCP port, cloud, or other address identifiers.

Web Service Discovery using DHT enables an end user to access a particular Web Service by providing the keyword along with the category. To carry out this process, search algorithms for locating the Web Services should reside on every peer. Web Services can be identified by using server.mappath method. All the directories and sub-directories are searched till it encounters a Web Service file. The Web Service information is searched in the same way in all other peers that are connected across that network. It finds one if it is available else returns nothing. An alternate text is displayed in this case. The requested Web Service information can thus be accessed by clicking the URL.

Distributed Hashing to Store and Retrieve Web Service

Several DHT algorithms are currently available. Each DHT algorithm has its own procedures for allowing a node to join, to insert a new service, to query for a service, to leave the DHT network. In General, the DHT algorithm describes all mechanisms required for the maintenance of the DHT. However, in order to implement such mechanisms amongst several nodes distributed on an underlying network (usually an IP network or more precisely the Internet) a proper communication protocol is needed, generally RPC protocol is preferred. The use of RPC protocols is motivated by the typical interaction between the peers. All DHT algorithms implements the following basic operations:

- join: this operation allows a peer to join an overlay.
- leave: this operation allows a peer to leave the overlay it is currently enrolled in
- put: this operation allows storing a key/value pair in the DHT;
- get: this operation allows retrieving the information
associated with the given key. All these operations can be performed by peers.

Thus the implemented DHT based Decentralized Web Services Discovery Mechanism is a Peer-to-Peer Web Services discovery mechanism based on a distributed hash table. The end user requests for a service with a keyword. Based upon the category, the Web Services that are available in that category will be fetched as a result. Each Service will be maintained with an encrypted hash key that is generated by means of taking the hash of Web Service category and Web Service name. Thus the category based exact match searching using DHT increases the efficiency of searching.

VI. APPLICATION

Students of same discipline often share subject materials, discuss on different subject matters and hold subject materials collected on their own or from their friends. When students meet in a common place like classrooms, or discussion rooms, they normally exchange the information. A Peer-to-Peer network of the type discussed in this paper would be a perfect platform for students with the Web access to exchange information. Good Universities / colleges provide wireless access to their students. Students can access the information from anywhere (from home, hostel) even if they are away from campus. This would enable students to communicate with their peers (i.e.) their classmates or students from same discipline or other disciplines. The only requirement is that, the peers (Student) should be given identity which could be hashed and scaled up on requirement. Each peer (Student) in turn will act as a peer, put their documents, materials which they wanted to share with their peers. One such exchange would be their study documents. These documents on peers can be classified and retrieved by any peer (Student) on requirement. Documents might be study materials like Unix materials, Data Mining materials, Intelligent Systems materials. Documents can be characterized by the words appearing in them. One way to apply machine learning to classify documents is to treat the presence or absence of each word as a Boolean attribute. Naïve Bayes is a popular technique for this application. Instead of counting on number of occurrences of words, bags can be considered, where the bags can contain repeated words.

VII. CONCLUSION

In this paper, we have implemented a distributed Web Service discovery system built on a DHT based Peer-to-Peer network using .Net framework and WCF. This model avoids the single node failure, and improves the stability of the whole system. Also we have discussed the sharing and exchange of information through Peer-to-Peer Web Services. Classifying or clustering documents dynamically may be extended based on this platform further.

ACKNOWLEDMENT

We extend our heart felt gratitude to Dr. G. Viswanathan, Chancellor, VIT University, Dr. M. Khalid, Director, School of Computing Sciences, Dr. Rao Tatavatri, Research Director, VIT University for their continuous support rendered to us throughout our work.

REFERENCES


