ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

Locality Aware Peer to Peer Resource Discovery

Swathi.S[1], Thushara.A[2]

M.Tech student, Computer Science and Engineering, TKM College of Engineering, Kollam, India[1]

Associate Professor, Computer Science and Engineering, TKM College of Engineering, Kollam,India[2]

Abstract: Peer to Peer System enables the sharing of globally distributed resources by pooling them effectively. To achieve this goal, numerous resource discovery approaches have been proposed. The current approaches consume a large amount of bandwidth which in turn increases communication link cost. This paper presents an architecture for resource discovery to find the nearest neighbouring resource node. DHT based hashing algorithm which is mapped to structured p2p is exploited to find the resource node which is having close proximity. The performance analyses of the proposed system on various vector parameters are studied using the simulation software. The simulation studies show a significant reduction in the communication cost as the number of nodes increases.

Keywords: Peer to Peer, DHT, Kademlia, Oversim.

I. INTRODUCTION

Peer to Peer systems are massively computing distributed systems in which each peer can act as a client or server as in the client server architecture. A p2p system can be defined as the system which satisfies the criteria such as the self organisation without a central directory, symmetric communication where each node can act as a client or server and decentralised control. There are many ways to classify a p2p system [2]. Based on the way data is indexed, a p2p system can be centralized, local and distributed. In a centralized p2p system, a central server keeps reference to all its peers in the network whereas in a local p2p system, each peer keeps reference to its own data and in a distributed system, every node keeps reference to its own data along with the references to several other nodes in the system. Another classification is hybrid p2p systems and pure p2p systems which is based on the level of decentralization applied on the, as hybrid p2p systems and pure p2p systems. Hybrid P2P systems are those which needs a central entity to provide their service and these systems can be characterized as P2P with centralized index. Pure P2P systems can continue to provide service even if any single peer is removed from the system. P2P systems with local and distributed indices belong to this category. Pure P2P systems can be classified on the basis of level of structure as unstructured systems and structured systems. In unstructured systems, peers join the system by connecting themselves to any other existing peers. Properties of these systems are that they uses local index and they had no structure. In structured systems, peers join the system by connecting themselves to well defined peers based on their logical identifiers. Properties of structured p2p systems are that they uses distributed index for organizing nodes in the network. The third classification is based on generations of P2P systems as first generation P2P systems those that are using centralized index. Second generation P2P systems uses local indexes and third generation P2P systems are using Distributed Hash Tables (DHTs).

The Proposed system is a p2p system that comes under the category of Pure P2P system under which it is a structured system that uses a distributed indexing mechanism. So it belongs to the third generation P2P system.

This paper mainly concentrates on the resource discovery in file sharing applications of a p2p system. The file sharing application of a p2p system, with its introduction onwards is highly used by the internet community and its popularity increases each day. There are ethical and technical issues related with a p2p system that is used for file sharing. The main issue considered for study in this system is efficient resource discovery. The file sharing between closer peers in the p2p system must be encouraged. For that, whenever a resource is required for a peer, then it should be made available from the peer that is at a distance minimal to the requested peer. Many resource discovery mechanisms were introduced for peer to peer systems which are discussed in the next section.

The main data structure used for developing the proposed system is the distributed hash table (DHT)[3]. A DHT is an effective data structure to store, index and access node information as well as resource or content information in a

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

distributed system. A layered approach is followed to implement the system in oversim simulator[4][5][6]. The main concepts for node placement, indexing and routing is by considering distance as a constraint and it is introduced in the overlay layer of the system. Overlay layer thus deals with the key based routing associated with the DHT. The main concepts of resource management and discovery are handled in the two application layer tiers which use DHT as the underlying data structure. Thus the concept of DHT is introduced in the peer management as well as in the resource management in the proposed system. A DHT which effectively considers the distance between nodes in p2p system is used as the overlay layer for developing the proposed system. The Key Based Routing is thus implemented using the concept of kademlia[7] in the overlay layer. Application layer in the layered architecture effectively handles resource discovery and management.

Section II presents various resource discovery approaches briefly. The advantages and disadvantages of each of this approach are discussed briefly in this section. Section III gives an insight into the design and implementation of proposed system. Section IV gives the result and discussion of the proposed system followed by the future scope and conclusion of the work in section V.

II. RELATED WORK

Many resource discovery approaches were introduced in peer to peer systems. Resource management in a distributed system suffers many challenges such as where to store and how to find certain data item in a distributed system without any centralized control or coordination. Many approaches were proposed inorder to support resource management in distributed system.

The three basic strategies for data retrieval in a distributed system are search in centralized servers, flooding search and distributed indexing. A p2p system that followed a centralized server approach is Napster in which the current locations of data items are stored in a central server. The advantage of this approach is that the location of a desired information can be retrieved with a search complexity of O(1), as the requester need to know the central server only. Here fuzzy and complex queries are possible since the server has a global overview of all available contents. The problems with this approach are that a central point of failure and attack to the central server are possible. Also the problems concerning scalability and availability are also there. Complexity in terms of memory consumption is O(N) as all location information are stored in a single machine. So this approach is suitable for a system with a limited number of nodes and it is not an efficient resource management strategy in large distributed systems.

Another approach is flooding search where nodes only keep information about their own data items and doesn't keep any explicit information about location of data items in other nodes. There is no additional data regarding where a specific item can be found. Thus a resource requester needs to broadcast a request among nodes in the distributed system. Each of the nodes that receive such a request message floods this message to other nodes until a certain hop count is exceeded. One of the well known p2p application that adopted this strategy is Gnutella. The advantages of this approach are that storage cost is reduced, complex and unsharp queries can be placed as nodes implicitly use proximity due to expanding search mechanism and also network maintenance efforts are comparatively low. The disadvantage is that a broadcast mechanism doesn't scale well. Also the number of messages and bandwidth consumed is extremely high and it increases linearly with the increase in number of participants.

The third approach is the distributed indexing with the concept of distributed hash tables. The concept of DHT is the main concept of the proposed system. Nodes in a distributed system organize themselves in a structured overlay and establish small amount of routing information for quick and efficient routing to other overlay nodes. Many DHT protocols have been proposed so far. The one used in the proposed system is kademlia. Chord [8] was another DHT which is used popularly. These protocols specify how keys are assigned to nodes and how a node can discover the value for a given key by first locating the node responsible for that key. IDs and keys are assigned an m-bit id using consistent hashing where the base hashing function in most cases is SHA-1 algorithm. Based on consistent hashing, both keys and ids are uniformly distributed and in the same id space. Consistent hashing is needed to let nodes join and leave the network without disruption.

In order to support locality awareness many solutions were proposed which are briefly discussed in this section. First approach among this is to discriminate p2p traffic from ongoing traffic and deliberately slow down or block them. Some ISPs adopted this technique in order to save the bandwidth consumed by these kinds of systems which involve many inter ISP traffic. Main advantage of this solution is that it reduced internet transit expenses and infrastructure upgrade costs. But this solution has drawbacks as they blocked a particular category of traffic it collided with network neutrality principle. This approach also resulted in loss of customers and government investigations. All other solutions rely on downloading resources when possible within the boundaries of the ISP.

Another solution for the problem is caching mechanism adopted for p2p traffic. This ensures a high probability for users to download contents without crossing the ISP's boundary. Here the main feature is that ISPs control the downloaded content. But there are copyright issues as ISPs are allowed to cache the resources. This approach also introduced complexities due to selection of resources to store and replace in caches.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

Modification of one or more system components is a promising approach to provide traffic locality in p2p file sharing systems. Here the traffic localization is ensured by directing the requests to closest resource providers through modification of some components of p2p systems. The proposed system belongs to this category. There are many possibilities in this kind of solution: Modification of indexing system, Modification of p2p application and Modification of both indexing system and p2p application.

Modifying behavior of p2p application is one such solution where the p2p application autonomously acquire their localization information and provide its localization information to peers which are interested in the resources that they share. Here the main steps in resource acquisition are:

1. Node acquires a list of resource providers from the indexing system.
2. Contacts the Resource providers present in the list asking them for their localization information.
3. Compares the obtained result with its own localization information.
4. Select the closest Resource Provider from the list.

Example Applications that belongs to this category are ONO [9], which is a software extension of Azureus BitTorrent client and KONTIKI [10]. In ONO Approach each ONO instance determines its location by querying a Content Delivery Network (CDN) for a fake resource. Then they will collect the mirror sites the CDN chooses for them according to the principle that users are redirected to a set of mirrors that are probably close to them. KONTIKI implements a simpler localization approach. KONTIKI nodes obtain their ASN (assigned to ISPs by IANA) starting from their IP addresses from public databases.

Modifying either p2p application or indexing system is the second approach that creates and exploits a strong collaboration between users and ISP's to be used in conjunction with some modifications to either the p2p application or the indexing system. Each ISP deploys a special equipment providing localization information to either the p2p application or to the indexing system depending on the specific solution. Many proposals were there which utilizes this concept of traffic localization. Aggarwal et.al [11] proposed to deploy a centralized equipment called ORACLE that users can query once they have acquired the list of Resource Providers from indexing system. P4P [12] solution proposes to deploy an iTracker which is equivalent to the oracle facility, but here the indexing system contacts the iTracker for localization information before sending the list of Resource Providers to a querying user. Advantage of this approach is it provides more precise localization information than that of ONO and KONTIKI. But here there are disadvantages also. It involves additional effort for ISP which has to deploy and maintain the equipment for ordering the list of possible service providers. Another disadvantage is that the presence of this equipment may allow malicious users to reconstruct the ISP topology.

III. LOCALITY AWARE P2P RESOURCE DISCOVERY

The main concept behind the proposed system is locality awareness principle. Locality awareness techniques are those methods that aim at lessening the usage of expensive international links by exploiting traffic locality. A resource is downloaded from inside the ISP whenever possible. Here the main concept used is to locate the resource provider which is closest to the user. This is ensured by utilizing the concept of Kademlia as the overlay key based routing topology. Also the concept of Distributed Hash Table (DHT) is introduced to lookup the resources from the indexing system.

System is implemented using the OverSim simulator. OverSim is an open source, powerful, and fast Peer-to-Peer (P2P) simulation platform based on the OMNet++ simulation environment. Simulation in OverSim follows a layered strategy.

The Simulation in OverSim is by defining layers needed for the system. A modular approach is followed in implementation. Each layer is implemented as several modules based on the requirement. Each component for simulation is thus defined separately in separate files and folders.Each layer in an oversim simulation consists of implementing modules. After defining all modules for every layer in a system to be developed, the parameters must be specified by defining the configuration for the system under development in a simulation file (.ini file) which must be in the simulations folder under oversim directory. Then the makefiles can be generated by building the project. After generating the makefiles simulation can be launched by running it by specifying the defined configuration as input. The simulation run when completed results in two files where one file holds the scalar values for the output parameters based on the global statistics and another is a vector file which is also based on the global statistics generated as a result of running the simulation. These two analysis files can be used for plotting graphs to analyse various network properties. These steps are explained in detail in the following section.

Basic layered architecture used in simulation of proposed system is shown in figure 1. The underlying network topology describes the basic communication criteria, protocols used and topology defined by the physical network of systems. Overlay network is the virtual network over the underlying physical network which defines the communication between peers of the p2p system. Application which is the basic file sharing is implemented on top of the overlay network by defining two tiers over the overlay network. This classification is mainly done in order to implement the proposed system in simulator.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

APPLICATION LAYER	TIER 2 APPLICATION LAYER- DHTApplication
	TIER 1 APPLICATION LAYER- DHT
OVERLAY TOPOLOGY- Kademlia	
UNDERLYING NETWORK TOPOLOGY- Simple underlay	

Fig. 1. Basic Layered architecture

The overall layered implementation strategy defines the layered architecture of the system in detail. The overall layered architecture is shown in figure 2. Each of the layers contains components that define the modules in each layer. It is mainly the detailed view of basic layered architecture shown in figure 1 with the modules and components. Each of the layers defines the basic topology and behaviour of nodes for that layer through the modules defined for that layer. The details of implementation of modules in each layer are discussed in this section.

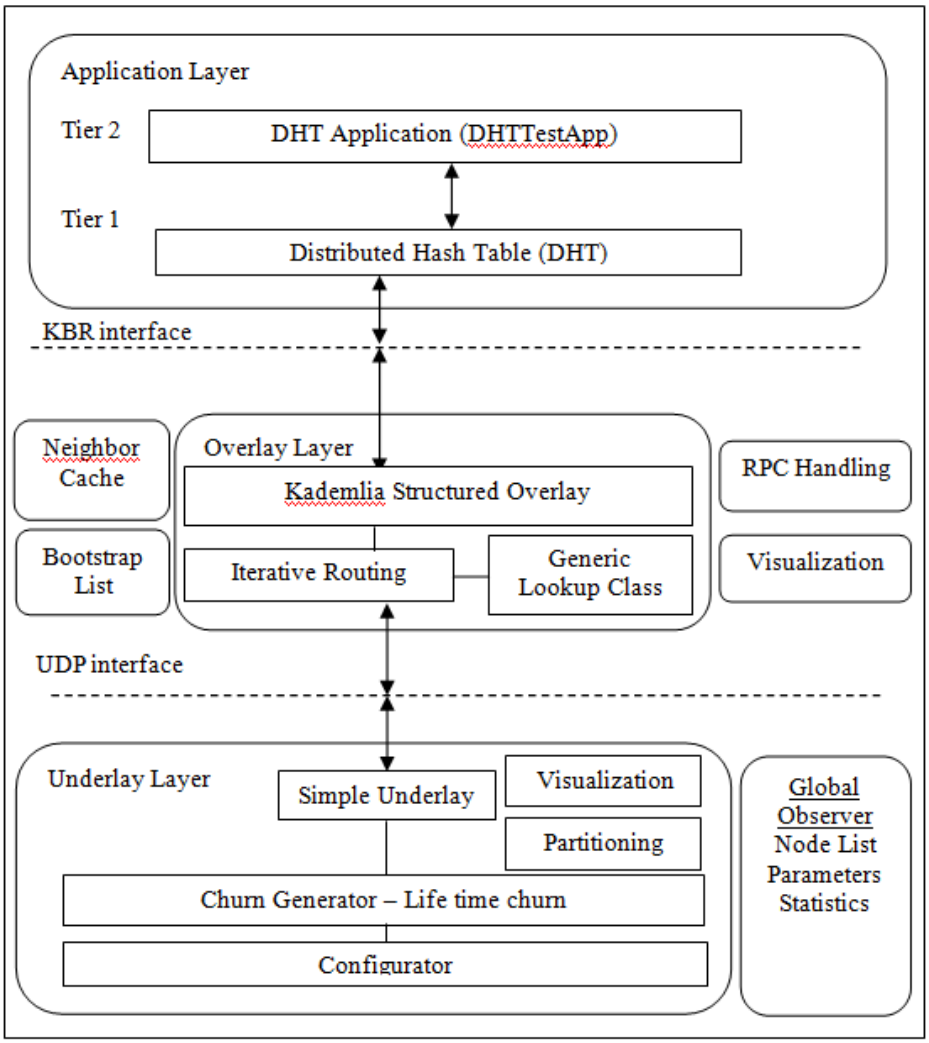


Fig. 2. Overall Layered Architecture

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

The underlying network used to implement the proposed system is simple underlay. In this model data packets are sent directly from one overlay node to another by using a global routing table. Packets between overlay nodes get delayed either by a constant period of time or by a delay calculated from the nodes' distance for more realistic scenarios. Due to the low simulation overhead of these techniques, the Simple model leads to a high level of accuracy and the ability to simulate networks with a large number of nodes.

OverSim uses churn generators for simulating dynamic networks. The churn generator used for simulating the proposed system is lifetime churn where the parameters are specified in the configuration in the simulation file. A generic module called Global Observer has a global view on the overlay network in every simulation. It can fulfill several user-defined functions. The global observer component in the underlay layer keeps track of the global parameters, global statistics and global node list. Global parameters and node list must be kept in order to trace the simulation and to keep track of the simulation. Global statistics are measured and are given as output scalar and vector files after the simulation to perform analysis.

The overlay layer defines the modules that perform Key Based Routing (KBR). The oversim simulator defines the KBR overlay topology by defining the overlay module. All implemented KBR protocols that make proper use of BaseOverlay support some routing modes, where the proposed system uses an iterative routing mode. The proposed system uses kademlia protocol as the overlay KBR topology.

Kademlia is a distributed hash table which is used in structured peer to peer networks. The structure of the network and the exchange of information is specified in kademlia through node lookups. Nodes in kademlia communicate among themselves through UDP. The participant nodes form a virtual or overlay network.

The principle class to write applications in Oversim is the BaseApp class. All applications in Oversim have to extend the BaseApp class. All application modules should derive from the BaseApp class which is the template application module class. The proposed system develops the application layer in two tiers. First tier implements a Distributed Hash Table (DHT) which is built on top of structured overlays. It is an efficient data structure in a distributed system which is used to index resources and resolve queries with some guarantees on performance. Second tier implements a DHTTestAPP which is built on top of DHT. DHT and DHTTestAPP work together where DHTTestAPP utilizes the DHT to store & retrieve (key, value) pairs. DHT utilize the underlying overlay to route queries.

Distributed Hash Tables (DHTs) are built on top of structured overlay networks. DHT are used to index resources/contents and resolve queries with some guarantees on performance. For example, maximum number of overlay hops to route a put (key, value) or a get (key) message. DHT and DHTTestAPP work together where DHTTestAPP utilizes the DHT to store & retrieve (key, value) pairs. DHT utilize the underlying overlay to route queries.

DHT implementation consists of two modules:

1. DHTDataStorage : A data structure and functions to add, update, and remove resources/contents into/from the DHT is provided by this module.
2. DHT : It is capable of storing and retrieving multiple (key, value) pairs given that they have distinct kinds or IDs. Each entry has an expiration time. So they are removed from the DHT when it expires.

There are some messages used to communicate between the two tiers of application layer. The message sendInternalRPCCall() is used to communicate between DHTTestApp and DHT using DHTputCAPICall and DHTgetCAPICall messages.

DHTTestApp is a Tier2 application. Both the tiers in application layer work together where DHTTestAPP utilizes the DHT to store & retrieve (key, value) pairs. The functions such as adding, updating, and querying resources/contents using a DHT can be demonstrated using this tier. This application can be modified to perform complex DHT functions. DHTTestAPP is implemented with the following two modules:

1. GlobalDhtTestMap : A data structure and functions to maintain a global list of (key, value) pairs in the P2P system is provided by this module. Random keys are taken from this list to issue DHT get and modify queries. GlobalDHTTestMap can be modified to replay traces of (key, value) pairs or queries.
2. DHTTestApp : DHT put, get, and modify queries are issued by this module after some delay derived from a truncnormal distribution.

The proposed system implements the layered approach defined above and each layer is implemented as separate modules.

The parameters for simulation must be given by specifying the configuration in any of the simulation (.ini) file. The configuration file should be located in the OverSim/simulation folder. All the default parameters for simulating the proposed system are specified in default.ini file. And all other parameters are specified in omnetpp.ini file.

IV. RESULTS AND DISCUSSION

The system is studied and analysed in oversim simulator. The configuration parameters for the simulation defines a

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

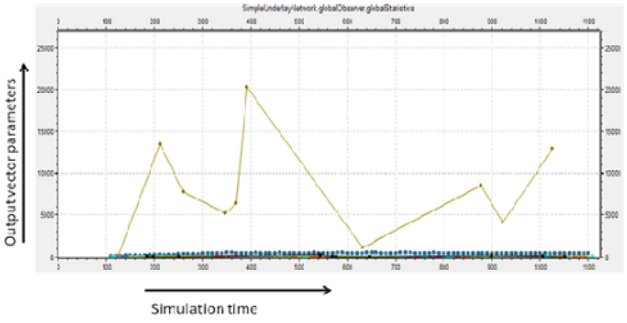
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

lifetime churn as the underlying churn generator and overlay routing topology as kademlia modules. The lifetimeMean parameter for simulation is taken as 10000s and measurementTime as 1000s and a transitionTime as 100s. Two tiers are defined for the application layer as specified in the previous section. Both these tiers are also described in the configuration file as simulation parameters. The configuration parameters required for simulation of kademlia nodes is also described in the configuration file for simulation. Thus the configuration file specifies all the parameters required to simulate the proposed system. Overall performance is plotted and analysed using the plove tool of oversim. The results are obtained as scalar and vector files after running the simulation. Then graphs for various vector parameters as well as for the entire system are obtained and the plots are shown in Figure(3).

Figure 3 represents the plotted output graph which shows the overall system performance against simulation time. The output vector parameters such as time between node creation, time between node deletion, churn session time, packets sent and number of entries stored in DHT etc are plotted against the simulation time in the graph.

In the figure the second graph is obtained by plotting the mean values for the output vectors against simulation time. Thus the difference in values of output vector can be identified from this graph. When simulating the system with more number of terminals the mean session time is higher than that when simulating with lower number of terminals.

Figure 4 shows the graph plotted by taking the hop count against the simulation time in both the simulation scenario. Thus the hop count or the distance taken to traverse between nodes against the simulation time is plotted as graphs. When the number of nodes in the system increases, then the possibility of closer resource providers to a node in the system also increases.



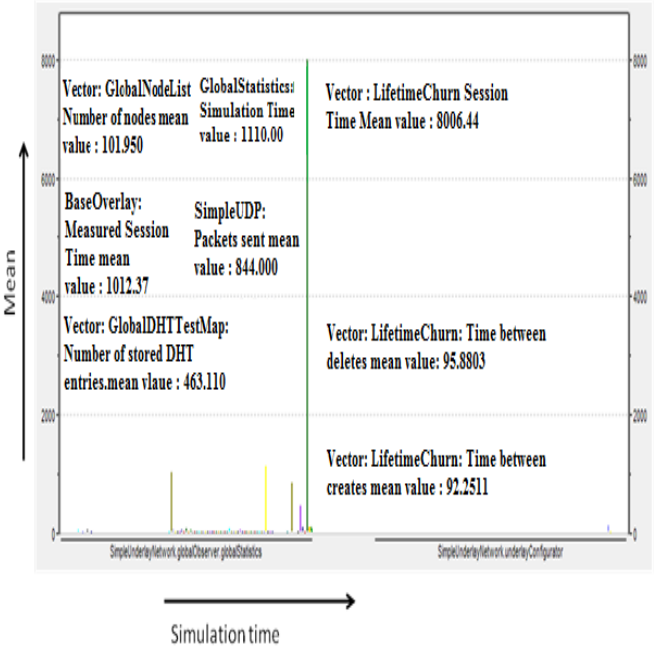


Fig. 3. Overall system performance

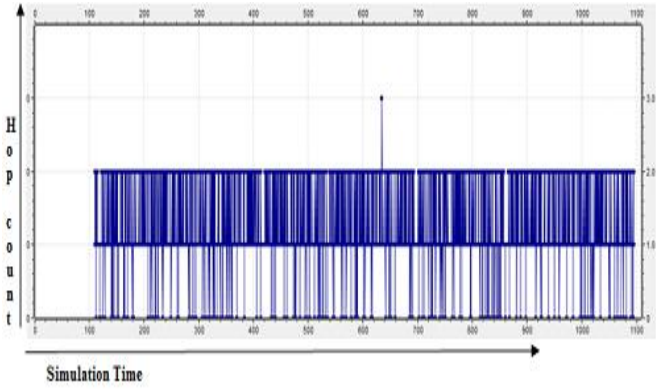


Fig. 4. Hop Count vs Simulation time

V. CONCLUSION

The main problem with an Internet Service Provider which supports a peer to peer technology is the large bandwidth usage on international links, mainly due to peer-to-peer applications adopted for file-sharing. The Locality-aware Peer to Peer application solves this issue by properly modifying the behavior of peer-to-peer application. The "locality-aware" P2P applications are expected to achieve efficient content distribution, with minimum impact on the physical network. To achieve this goal, P2P applications must have access to network locality information. Here the concept is to use the kademlia protocol as the Key Based Routing overlay protocol along with introducing the concept of DHTs for efficient lookup mechanism. The system with

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

kademlia as overlay Key Based Routing(KBR) topology is implemented . The indexing and searching the DHT in both overlay layer and application layer are simulated. Locality aware approach with DHT as the underlying mechanism is implemented. The analysis is performed using oversim and omnet++ tools and studied. The main advantage of this method is that the message is routed between the closest peers. Thus Churn handling is efficient in this way which handles the dynamic behavior of network.

Deployment with real data set is the main concept which can be implemented. The main limitation of the implemented system is that localization information needs to be given in order to effectively index the resource providers. The system can be deployed with the real data set which takes the localization information of the peers are also possible. Another method to enhance the system is to deploy this in planetlab.

REFERENCES

[1] Marco Papa Manzillo, Luigi Ciminiera, Guido Marchetto, Member, and Fulvio Risso "CLOSER: A Collaborative Locality-Aware Overlay SERvice", IEEE Transactions On Parallel And Distributed Systems, Vol. 23, No. 6, June 2012.

[2] G.Camarillo (2009) "Peer to Peer P2P Architecture: Definition, Taxonomies, Examples, and Applicability", RFC 5694.

[3] Klaus Wehrle, Stefan G¨otz, Simon Rieche, "Distributed Hash Tables".

[4] András Varga, and Rudolf Hornig, "An Overview of the Omnet++ Simulation Environment",2008.

[5] Ingmar Baumgart, Bernhard Heep, and Stephan Krause,"OverSim: A Flexible Overlay Network Simulation Framework", in Proceedings of 10th IEEE Global Internet Symposium (GI '07) in conjunction with IEEE INFOCOM 2007, Anchorage, AK, USA, May 2007, pp. 79–84.

[6] Ingmar Baumgart, Bernhard Heep, and Stephan Krause, "OverSim: A scalable and flexible overlay framework for simulation and real network applications", 2009.

[7] Petar Maymounkov and David Mazi`eres "Kademlia: A Peer-to-Peer Information System Based on the XOR Metric".

[8] Stoica, I., Morris, R., Karger, D., Kaashoek, M. F., & Balakrishnan, H., "Chord: A scalable peer-to-peer lookup service for internet applications", Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications, (pp. 149-160). San Diego, California, United States: ACM. doi: 10.1145/383059.383071, 2001.

[9] D. R. Choffnes and F. E. Bustamante, "Taming the torrent: a practical approach to reducing cross-isp traffic in peer-to-peer systems," in Proceedings of the ACM SIGCOMM 2008 conference on Data communication. Seattle, WA, USA: ACM, 2008, pp. 363–374.

[10] "Kontiki." [Online]. Available: http://www.kontiki.com

[11] V. Aggarwal, A. Feldmann, and C. Scheideler, "Can ISPS and P2P users cooperate for improved performance?"SIGCOMM Comput.Commun.Rev., vol. 37, no. 3, pp. 29–40, 2007.

[12] H. Xie, Y. R. Yang, A. Krishnamurthy, Y. G. Liu, and A. Silberschatz, "P4P: provider portal for applications," in Proceedings of the ACM SIGCOMM 2008 conference on Data communication. Seattle, WA, USA: ACM, 2008, pp. 351–362.