A Survey on Distributed Service Discovery Mechanisms with the Focus on Topology Awareness

Article · January 2015

DOI: 10.1007/978-3-319-13153-5_31

2 authors:

Mohamed Saleem Haja Nazmudeen
Universiti Teknologi Brunei

Seyed Buhari
King Abdulaziz University

A Survey on Distributed Service Discovery Mechanisms with the Focus on Topology Awareness

Mohd Saleem Nazmudeen[1] and Seyed Mohamed Buhari[2]

[1] Institut Teknologi Brunei, Brunei Darussalam
mohamed.saleem@itb.edu.bn
[2] King Abdul Aziz University, Kingdom of Saudi Arabia
mesbukary@kau.edu.sa

Abstract. Distributed Service (or resource) Discovery (DSD) is becoming an important research area in Service Oriented Computing (SOC) because many software applications are now developed with services from different vendors. The query routing mechanism of the current DSD applications functions purely on the overlay without incorporating the topological and routing knowledge of the underlying physical topology of the network. Consequently, Internet Service Providers (ISPs) are tested to their limits due to underlay-ignorant query forwarding that are employed by the overlay applications such as Peer-to-Peer and DSD. This paper surveys the existing query routing approaches in various domains of Distributed Service Discovery and summarizes their level of awareness with respect to the underlying network topology. We have identified various characteristics required for the query routing algorithms to be intelligent and the comparisons are performed based on those characteristics.

Keywords: Service discovery, topology awareness, intelligent query routing.

1 Introduction

The query routing mechanism of the current DSD applications functions purely on the overlay without utilizing the topological and routing knowledge of the underlying physical network topology. Consequently, Internet service providers (ISPs) are tested to their limits due to underlay-ignorant query forwarding that is employed by the overlay applications such as Peer-to-Peer (P2P) and DSD. This paper surveys the existing query routing mechanism in DSD and their implementation approaches with respect to underlying topological awareness. In order to critically analyze various existing literatures in the area of DSD, a thorough comparison of those works with respect to the parameters that are relevant to this survey need to be performed. The main characteristics that a query routing algorithm which supports underlay based query routing are, Service Discovery (SD) architecture that can perform informed searches [1], intelligent query processing SD system [1], underlay location awareness [2, 3], reduction of inter-ISP traffic [4] and avoiding redundant traffic in the underlay [5, 6].

S. Phon-Amnuaisuk and T.-W. Au (eds.), *Computational Intelligence in Information Systems,*
Advances in Intelligent Systems and Computing 331, DOI: 10.1007/978-3-319-13153-5_31

Taking the above mentioned criteria into consideration; the following parameters were identified and used in the following sections for comparison of existing literatures,

1. System Architecture
2. Query routing approach
3. Underlay and location awareness
4. Range queries
5. Intelligent query processing
6. ISP consideration
7. Layer in which query routing is performed
8. Traffic redundancy in the underlay

The rest of the paper is organized as follows, section 2 discusses the service discovery in the P2P domain, sections 3 and 4 discuss that of the grid and web service domains respectively. Section 5 discusses the existing query routing algorithms that provide location awareness to some extent and section 6 concludes with the existing research opportunities in location aware query routing process.

2 Service Discovery in P2P Systems

Gnuetella [7, 8, 9, 10-13, 3, 4, 14, 15, 16, 17, 18, 19-23] is one of the early P2P file sharing applications which is based on an unstructured approach. Each peer in the Gnutella system maintains a connection to at least one other peer. Neighbours are formed as they come to know about their existence. The query for file discovery is flooded into the network. A peer sends the query to all its neighbours and the neighbours in turn forward it to their neighbours and the process is triggered. Any query hit is propagated back to the query initiating peer. The flooding approach is fine with a small number of nodes and users. However, when the number of users using the system proliferates the performance deteriorates. As a consequence many variations of the protocol have been introduced following the Gnutella protocol. All of those routing protocols function purely on the overlay.

Kalogeraki et al. [24] has provided an implementation of the extended Gnutella protocol with two improvements. The problem area they improved is actually the searching and routing mechanism among the nodes of Gnutella. Those improvements are modified Breadth First Search (BFS) and intelligent search mechanism based on past behaviour. Unlike the original Gnutella protocol, the modified Breadth First Search chooses only a subset of the neighbours. Intelligent searching is performed by monitoring the past queries so that only those neighbours which are likely to reply to the queries are used for forwarding. The authors' objective was to minimize the search cost and to improve the search efficiency of the system. However, it still remains an unstructured overlay without any underlying topological consideration.

Freenet [25] is another unstructured P2P system which employs a better query routing process compared to Gnutella. It makes use of key-based query processing. The neighbour for query forwarding is selected if the key has been answered by the neighbour earlier. Here the links between the neighbours are dynamically altered by making new links and dropping old links based on the responsiveness of the peers. The overlay layer does not possess any knowledge about the underlying topology.

Chord [10] is the first structured system which employs the Dynamic Hash Table (DHT) as a way of implementing a distributed look up service in the Internet. It employs a key mapping mechanism in which a given keyword of a particular resource is mapped into a node with the help of a hash function. If there is no exact node identifier for the given key, then the node next in the sequence will be used for storing the data. This node is called the successor node. The complexity of the resource look up in Chord is O(log N). The main drawback of the Chord system is its tight administrative overhead and difficulty in providing support for range and partial queries.

The Content Addressable Network [22] is another popular structured P2P system which makes use of tree structured nodes and a virtual Cartesian coordinate space with a d-dimension. A node learns the set of IP addresses of the nodes that falls as neighbours in the coordinate space and not in the physical topology. In this way there is no information about the physical location information of the underlying network. Also, the routing complexity in CAN is O(dN1/2) where d is the dimension of the Cartesian coordinate space. Therefore, when d is small, the complexity is worse than Chord which is a DHT based implementation.

An improvement to CAN was proposed by [26], which tried to modify the way the routing was performed in the coordinate space of CAN. It was argued that in CAN, routing was inefficient especially for small values of dimension for the coordinate space because a node in the coordinate space maintained only the set of IP addresses of the neighbours which was constant irrespective of the number of peers. Therefore, some 'Long Distance Pointers (LDP)' were added which were IP addresses of some distant peers. This could eventually reduce the routing complexity. The underlying topological consideration and the traffic redundancy control were not in their scope of work.

FastTrack [27] is a proprietary protocol that uses encryption for message exchanges. Until today the operation of FastTrack still remains a mystery. It has been found through packet sniffing that it operates in a hybrid structure with super nodes. How the topology is constructed and whether there is any underlay awareness in the overlay layer is not known.

HyperCuP (Hypercube P2P) [28] is an attempt to reduce the flooding mechanism usually employed in the unstructured P2P systems. It also introduces a partial structure to the unstructured systems by building a graph for each formation of neighbour. The main objective of the HyperCuP is to form a TCP overlay which is used for the formation of graph structure on the nodes, so that the network diameter for finding a resource in the worst case can be reduced as compared to that of unstructured systems. Here again, it is a pure overlay construction and the underlay awareness is out of their scope.

Kademlia [12] is a structured P2P system which is based on DHT similar to that of Chord [11]. However, Kademlia differs from Chord in the way the configuration messages are learned and in the topology construction. Kademlia uses an XOR metric based tree topology for the routing of query messages. In pure DHT based systems the path which has a higher latency has the same weight as the one with a lower latency so that the routing process cannot differentiate between them. In Kademlia, these configuration messages are automatically spread to the nodes so that nodes can route queries with enough information.

Pastry [29] is a hybrid architecture which can perform either tree based routing like Kademlia or ring based routing like Chord. It resorts to ring based routing if the tree routing fails. For this purpose each node maintains a set of leaf, routing and neighbour tables. The neighbour table is used to perform routing based on proximity. However, redundancy and ISP issues are not taken into account.

The characteristics of the reviewed P2P systems are summarised in Table 1.

Table 1. Service discovery in P2P applications

	Architecture	*Query routing mechanism*	*Underlay location awareness*	*Partial, Range, and Multi attribute queries*	*Intelligent Query processing*
Gnutella [13]	Unstructured	Flooding	No	Yes	No
Kalogeraki et al. [24]	Unstructured	Controlled flooding	No	Yes	Yes(Intelligent search mechanism based on past behaviour)
Freenet [25]	Unstructured	Controlled flooding	No	No	No
Chord [10]	Structured	DHT	No	No	No
CAN [22]	Structured	Selective neighbour selection based on Cartesian coordinates	No	No	No
Sahin et al. [26]	Structured	Selective neighbour selection based on Cartesian coordinates Adds ‘Long Distance Pointers (LDP)’ to the set of neighbours’ list	No	No	No
FastTrack (Proprietary) [27]	Hybrid	N/A	N/A	-	N/A
HyperCuP [28]	Hybrid	Controlled flooding	No	No	No
Kademlia [12]	Structured	DHT	No	No	Partial

3 Service Discovery in Web Services

Fatih et al. [30] contributed in providing a distributed service discovery so that trust and Quality of Service (QoS) are taken into account during service discovery. They incorporated this feature by providing a ranking system for similar services based on trust and QoS. Therefore, if there is an option in choosing more than one service in the discovery process, the one with the highest ranking will be chosen. A sample scenario provided by the authors is that of a user who wishes to purchase a book online with the most secured payment service and the fastest possible delivery. These choices can be automated in the SD process of their system. The system here is based on the Chord based structured P2P system.

Liu et al. [31] provided a two layer P2P model for distributed service discovery based on semantics. Their approach in organizing the peer structure is based on ontology. They argued that ontology based SD is more time consuming because of the logical reasoning, and therefore needs to be distributed. The structure here is similar to the super peer architectures. The super peer is the ‘Ontology Agent’ under which a particular ‘Ontology Community’ is organized. The ‘Ontology Agent’ forwards the query to other ‘Ontology Agent’ if the query requests services from different ‘Ontology Communities’. It also maps the ontology from one form to another form as per the requirement. Here, they employ a hybrid approach similar to cluster formation and functions by forming an overlay.

Lin Zhang [32] provided another two layer approach which has the cluster based super peer structure. However, the main difference from this work and [32] is that the centralized discovery is performed within the cluster and the P2P model of discovery is performed only across the clusters. Here, the clusters are formed based on semantic similarity. Ling Zing’s approach uses the overlay hybrid model and the topology consideration is out of his scope. The distributed structure of Lin Zhang’s implementation is shown in Figure 2.6. The black nodes are the super nodes and the white nodes are the normal nodes grouped according to their semantic similarity. The nodes that need to be searched for service discovery are reduced due to the number of super nodes in the system.

Sioutas et al. [33] provided a remarkable contribution in distributed web service discovery. The main advantage of the DHT based system is that the lookup takes O(log N) complexity where N is the number of nodes. Sioutas et al. have gone a step further and have come up with a structured system which can provide the complexity of O(log log N). Their system is called ‘NIPPERS (Network of InterPolated PeERS)’. This system is in fact very suitable for highly static systems. However, the tight administrative overhead due to the high structured nature makes it not suitable for situations where dynamic partnership among organizations (frequent exit from and entry into the P2P system) is of higher priority.

Scoutas et al. [34] has given an approach where semantic web service discovery can be applied in both the centralized and P2P environments. In order to improve the response time they adopt an approach called progressive query matching where better matches are returned first while the query processing proceeds further. They make use of the structured P2P organization which employs spatial coordinates.

Chord4S [11] is a system which attempts to improve the original Chord [10] so that the services are distributed evenly across nodes rather than concentrating on a particular node. The services have the tendency to get hashed to the same node if they belong to the same category as they are usually mapped to closer hash values. Chord4S has avoided this by adding some provider bits to the service bits for the purpose of hashing so that they end up with different hash values and eventually different nodes. Figure 2.7 shows an example of how the provider bits are added to functional bits. In this example the bits corresponding to service provider IP address '10.0.0.1' is added with the functional bits of the service provided.

Table 2. Service discovery in the domain of web services

	P2P Approach	*Problems Addressed*	*Underlay awareness*	*ISP Considerations*
Fatih et al. [30]	Chord-based, structured	Addresses QoS and trust attributes for service discovery	No	No
Liu et al. [31]	Super peer cluster (Hybrid)	Performance improvement in ontology processing	No	No
Lin Zing [32]	Super peer cluster (Hybrid) Centralized within the cluster and distributed across	Semantic classification	No	No
Sioutas et al. [33]	Tree structure with interpolated peers	Routing efficiency	No	No
Scoutas et al. [34]	Spatial coordinates, structured	Progressive query matching for improved response time	No	No
Chord4S [11]	Chord	Load balancing	No	No
JaxSON [35]	Hybrid, Clusters based on semantic affinity	Semantic based service discovery	No	No
Zhou et al. [36]	Tree-based, structured	Service discovery in cloud	No	No
Schmidt et al. [37]	Structured	Range queries	No	No

JaxSON [35] is a semantic overlay which is hybrid in nature with cluster formation based on 'Semantic Affinity (SA)'. According to this structure the cluster which can answer a particular query can be identified by the system based on SA. The SA function is used to identify the group in which a new node can join. The SA would be compared with the existing groups and the entering peer. The group which has the highest value of SA with the entering peer would add the peer into its group. All the processes that are taking place here are completely on the overlay of the system and have nothing to do with the underlay. This approach is similar to Lin Zang [32], however JaxSON uses SA for identifying the cluster to which a node needs to be joined. In JaxSON, it is possible that a node can join more than one group with slightly different SA. During the query processing SA will determine to which group a node belongs.Zhou et al. [36] provided an interesting P2P model for service discovery in the cloud environment. They provide a Kd-tree approach for implementing range queries with a semantic and locality enabled service discovery process. The multidimensional query is routed based on the Kd-tree of the peers. However, it still functions in the overlay layer. A sample Kd-tree is shown in Figure 2.9 in which the point (200,50), which represents 200GB and 50Mb/s, splits the data space into two subspaces and accordingly other nodes and points are formed. If a node has branches the rectangle is split into two subspaces otherwise the point is just marked without splitting into further subspaces. This approach helps in localizing the region for searching range queries.

Schmidt et al. [37] provides another distributed web service discovery model which uses Chord as the P2P structure and Hilbert's space filling curve to map the multi-dimensional queries. It is a structured overlay system. Table 2 summarises the contributions of the above works.

4 P2P and SD Approaches with Location Awareness

PIPPON [15] attempts to bring the underlay awareness to the overlay by clustering the peers based on the proximity. The proximity of the peers is measured with two parameters namely, Longest prefix IP matching (LPM) and Round Trip Time (RTT).

Apart from cluster formation, it forms a dynamic "Key Tree" for routing the query messages on the overlay. This tree is employed to route the queries based on the IP prefixes as shown in Figure 2.10. However, PIPPON does not take into account the similarity of queries in the service discovery which is a key factor in improving efficiency. Also, it does not solve the problem of redundant traffic in the underlay.

TOPLUS [38] has adopted a slightly different approach to PIPPON in terms of underlay awareness. The cluster formation in TOPLUS involves three tier hierarchies such as groups, super-groups and hyper-groups so that the queries can be easily forwarded with a minimal number of hops. In contrast to the longest prefix match of the IP addresses, TOPLUS makes use of the XOR metric for identifying the proximity. In each tier the routing of the query messages is performed with the XOR metric. TOPLUS claims that the routing performance it provides is very close to IP routing. TOPLUS does not cover the aspects of inter-ISP, redundant traffic and interlayer communication overhead.

In general there are two approaches with respect to finding the locality information of the underlay:

- Dynamic (on the fly calculation with RTT and LPM), and
- Static (Prior knowledge of the Internet such as AS numbers).

Plethora [16] follows the second approach which is static whereas TOPLUS and PIPPON follow the first approach. Plethora makes use of cSpace, which uses a local broker, and gSpace which uses a global broker. Hence, it ends up with two layer architecture. If a request is generated, it checks the cSpace for answers; if cSpace could answer it, then the query is not forwarded to gSpace. It only forwards it to the gSpace if the query is not answered by the cSpace.

P4p [39] addresses the locality awareness problem by two interfaces iTrackers and appTrackers. The purpose of iTrackers is to provide the locality parameters of a particular node to the querying node. iTrackers are designed to reside in each of the ISPs. appTrackers are designed to reside globally so that they have a complete picture of the P2P applications. appTrackers would decide whether a particular peer "a" could establish a neighbour relation with another peer "b" based on the information from iTrackers. A sample scenario of the tracker interfaces are shown in Figure 2.11.

Table 3. SD approaches with location awareness

	Locality awareness approach	*Targeted performance criteria*	*Inter-ISP traffic reduction*	*Security*	*Query routing layer*
PIPPON [15]	Pure overlay based on RTT and LPM	Improved locality awareness	No	No ISP related security issues as it is overlay-based	Overlay
TOPLUS [38]	Pure overlay based construction with IP prefix-based location awareness	Improved locality awareness	No	No ISP related security issues as it is overlay-based	Overlay
Plethora [16]	Overlay-based, uses static ISP information	Improved locality awareness	No	No ISP related security issues as it is overlay-based	Overlay
P4P [39]	ISP Assisted approach	Traffic control with P2P application and ISP cooperation	Yes	Security concerns relating to exposing ISP information to the peers in the overlay are not addressed	Overlay
Seedorf et al. [2]	ISP Assisted approach	Traffic control with P2P application and ISP cooperation	Yes	Security concerns relating to exposing ISP information to the peers in the overlay are not addressed	Overlay
SLUP [3]	RTT based proximity identification without ISP involvement	Traffic control with locality based clustering	No	No ISP related security issues as it is overlay-based	Overlay
Bindal et al. [40]	Purely ISP - based. Specific for BitTorrent applications	Traffic control for BitTorrent	Yes	No specific security issues as the ISP information is not exposed to peers in the overlay.	Overlay
P4P Pastry [41]	ISP Assisted approach	Traffic control with P2P application and ISP cooperation	Yes	Security concerns relating to exposing ISP information to the peers in the overlay are not addressed	Overlay
Haja M. Saleem et al. [42]	AON routing	complete locality awareness	Yes	Security concerns with respect to AON route corruption	Underlay

IETF formed an Application Layer Traffic Optimization (ALTO) group in 2009 and has been tasked with identifying the means of optimizing the traffic in the underlay that are generated by the ever-increasing P2P applications. Seedorf et al. [2] elaborate on the efforts of the ALTO group which intends to provide the ALTO service to the P2P applications. With reference to the ALTO approach, any P2P application which needs to know the topological and proximity information could obtain that information from the ALTO server. Information like location of the target peer, operational cost and policies from the ISPs could be provided to the querying peers. A sample scenario is shown in Figure 2.12. In this approach, letting the peers be aware of these network related parameters could be misused by any malicious user. As an alternative the proposed approach in this research for location awareness could be used for the same goal as that of IETF in optimizing traffic in the underlay.

SLUP [3] is another approach which forms clusters based on semantics and RTT. It forms a three-tier clustering: normal peer, level-2 super peer and level-1 super peer. Redundancy in the underlay and location awareness are not considered in this work.

Bindal et al. [40] provided a strategy for improving the traffic locality in the BitTorrent network. They highlighted the fact that ISPs often end up throttling the P2P traffic as BitTorrent like unstructured P2P systems do not take ISPs cost into consideration. Their work focuses on biased neighbour selection in which, a peer chooses the majority of its neighbour from the same ISP so as to reduce the cross-ISP traffic. The authors point out here that if there are N users within the ISP wanting to download a particular file, then in the current BitTorrent system the file is downloaded N times, which heavily increases inter-ISP traffic. In their work they have demonstrated that biased neighbour selection could improve this situation. Here, their application is specific to the BitTorrent system which uses the rarest first piece algorithm at the client side.

P4P Pastry [41] is an approach to bring the locality features to the PASTRY [29] structured system. Its features are similar to that of P4p [38] except that it is applied in the context of PASTRY [29]. It has the same limitations as that of P4p [38].

Haja M. Saleem et al. [42] has proposed a radical approach in terms of location awareness as the routes to the queries are dynamically learned and stored for further query routing. Here they have proposed to perform the query routing process in the underlying network layer with the help of Application Oriented Networking (AON). Table 3 summarises the contributions of the above works and highlights the need for underlay awareness.

5 Conclusion and Future Work

This paper provided a comprehensive review of SD approaches in the field of P2P systems, web services and current topological awareness techniques. It can be concluded from the literature survey that there is a need for a mechanism which aids the process of DSD with intelligent message routing so that location awareness can be incorporated in the overlay of the distributed systems. There are still research needed to be done in the following areas such as, a study on how ISPs could be encouraged to provide location awareness for query routing, the performance of the DSD systems while clustering the peers purely based on service classes needs to be investigated further and the effect of clustering the registries across ISPs also demands a careful study.

References

1. Meshkova, E., et al.: A survey on resource discovery mechanisms, peer-to-peer and service discovery frameworks. Computer Networks 52, 2097–2128 (2008)
2. Seedorf, J., et al.: Traffic localization for P2P-applications: The ALTO approach. In: IEEE Ninth International Conference on Peer-to-Peer Computing, P2P 2009, pp. 171–177 (2009)
3. Xin, S., et al.: SLUP: A Semantic-Based and Location-Aware Unstructured P2P Network. In: 10th IEEE International Conference on High Performance Computing and Communications, HPCC 2008, pp. 288–295 (2008)
4. Karagiannis, T., et al.: Should internet service providers fear peer-assisted content distribution? In: Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement, p. 6. USENIX Association, Berkeley (2005)
5. Abboud, O., et al.: Underlay awareness in P2P systems: Techniques and challenges. In: IEEE International Symposium on Parallel & Distributed Processing, IPDPS 2009, pp. 1–8 (2009)
6. Jie, D., et al.: The disparity between P2P overlays and ISP underlays: issues, existing solutions, and challenges. IEEE Network 24, 36–41 (2010)
7. Cai, M., et al.: MAAN: A Multi-Attribute Addressable Network for Grid Information Services. Journal of Grid Computing 2, 3–14 (2004)
8. Talia, D., Trunfio, P., Zeng, J.: Peer-to-Peer Models for Resource Discovery in Large-Scale Grids: A Scalable Architecture. In: Daydé, M., Palma, J.M.L.M., Coutinho, Á.L.G.A., Pacitti, E., Lopes, J.C. (eds.) VECPAR 2006. LNCS, vol. 4395, pp. 66–78. Springer, Heidelberg (2007)
9. CISCO, Cisco AON: A Network Embedded Intelligent Message Routing System, http://www.cisco.com/en/US/prod/collateral/modules/ps6438/prod_bulletin0900aecd802c201b.html
10. Ion, S., et al.: Chord: A scalable peer-to-peer lookup service for internet applications. SIGCOMM Comput. Commun. 31, 149–160 (2001)
11. Qiang, H., et al.: Chord4S: A P2P-based Decentralised Service Discovery Approach. In: IEEE International Conference on Services Computing, pp. 221–228 (2008)
12. Maymounkov, P., Mazières, D.: Kademlia: A peer-to-peer information system based on the XOR metric. In: Druschel, P., Kaashoek, M.F., Rowstron, A. (eds.) IPTPS 2002. LNCS, vol. 2429, pp. 53–65. Springer, Heidelberg (2002)
13. Gnutella-Protocol, http://rfc-gnutella.sourceforge.net/src/rfc-0_6-draft.html (accessed March 7, 2012)
14. Hung-Chang, H.: A Near-Optimal Algorithm Attacking the Topology Mismatch Problem in Unstructured Peer-to-Peer Networks. IEEE Transactions on Parallel and Distributed Systems 21, 983–997 (2010)
15. Hoang, D.B., Le, H., Simmonds, A.: PIPPON: A Physical Infrastructure-aware Peer-to-Peer Overlay Network. Presented at TENCON 2005 2005 IEEE Region 10 (2005)
16. Ferreira, R.A., Grama, A., Jia, L.: Plethora: An Efficient Wide-Area Storage System. In: Bougé, L., Prasanna, V.K. (eds.) HiPC 2004. LNCS, vol. 3296, pp. 252–261. Springer, Heidelberg (2004)

17. Wei-peng Chen, Y.G., et al.: J-Sim 1.3, http://sites.google.com/site/jsimofficial/
18. Saroiu, S., Gummadi, K.P., Gribble, S.D.: Measuring and analyzing the characteristics of Napster and Gnutella hosts. Multimedia Systems 9, 170–184 (2003)
19. Guttman, E., et al. (eds.): Service Location Protocol, Version 2. RFC 2165 (1997), http://www.ietf.org/rfc/rfc2608.txt
20. Arnold, K., et al. (eds.): Jini Specification, 1st edn. Addison Wesley Longman Publishing Co., Inc, Boston (1999)
21. Leibowitz, N., et al.: Deconstructing the Kazaa network. In: Proceedings of the Third IEEE Workshop on Internet Applications, WIAPP 2003, pp. 112–120 (2003)
22. Ratnasamy, S., et al.: A scalable content-addressable network. SIGCOMM Comput. Commun. Rev. 31, 161–172 (2001)
23. Ripeanu, M.: Peer-to-peer architecture case study: Gnutella network. In: Proceedings of the First International Conference on Peer-to-Peer Computing, pp. 99–100 (2001)
24. Vana, K., et al.: A local search mechanism for peer-to-peer networks. In: Proceedings of the Eleventh International Conference on Information and Knowledge Management, pp. 300–307. ACM, McLean (2002)
25. Freenet,, Available: freenetproject.org/
26. Sahin, O.D., et al.: Techniques for efficient routing and load balancing in content-addressable networks. In: Fifth IEEE International Conference on Peer-to-Peer Computing, pp. 67–74 (2005)
27. Liang, J., et al.: The FastTrack overlay: A measurement study. Computer Networks 50, 842–858 (2006)
28. Schlosser, M.T., Sintek, M., Decker, S., Nejdl, W.: HyperCuP - Hypercubes, Ontologies, and Efficient Search on Peer-to-Peer Networks. In: Moro, G., Koubarakis, M. (eds.) AP2PC 2002. LNCS (LNAI), vol. 2530, pp. 112–124. Springer, Heidelberg (2003)
29. Rowstron, A., Druschel, P.: Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. In: Guerraoui, R. (ed.) Middleware 2001. LNCS, vol. 2218, pp. 329–350. Springer, Heidelberg (2001)
30. Fatih, E., et al.: A peer-to-peer framework for Web service discovery with ranking. In: Proceedings of the IEEE International Conference on Web Services, pp. 192–199 (2004)
31. Zhizhong, L., et al.: A two-layered P2P model for semantic service discovery. In: 2010 4th International Conference on New Trends in Information Science and Service Science (NISS), pp. 41–46 (2010)
32. Lin, Z.: A Scalable Model for Service Discovery in P2P Environment. In: 2011 International Conference on Intelligence Science and Information Engineering (ISIE), pp. 495–498 (2011)
33. Sioutas, S., et al.: Dynamic Web Service discovery architecture based on a novel peer based overlay network. Journal of Systems and Software 82, 809–824 (2009)
34. Skoutas, D.N., Sacharidis, D., Kantere, V., Sellis, T.K.: Efficient Semantic Web Service Discovery in Centralized and P2P Environments. In: Sheth, A.P., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 583–598. Springer, Heidelberg (2008)
35. Bisignano, M., Di Modica, G., Tomarchio, O.: JaxSON: A Semantic P2P Overlay Network for Web Service Discovery. Presented at 2009 World Conference on Services - I (2009)
36. Zhou, J., Abdullah, N.A., Shi, Z.: A Hybrid P2P Approach to Service Discovery in the Cloud. IJITCS 3 (2011)
37. Schmidt, C., Parashar, M.: A Peer-to-Peer Approach to Web Service Discovery. World Wide Web 7, 211–229 (2004)

38. Garcés-Erice, L., Ross, K.W., Biersack, E.W., Felber, P., Urvoy-Keller, G.: Topology-Centric Look-Up Service. In: Stiller, B., Carle, G., Karsten, M., Reichl, P. (eds.) NGC 2003 and ICQT 2003. LNCS, vol. 2816, pp. 58–69. Springer, Heidelberg (2003)
39. Xie, H., et al.: P4p: provider portal for applications. Presented at the Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication, Seattle, WA, USA (2008)
40. Bindal, R., et al.: Improving Traffic Locality in BitTorrent via Biased Neighbor Selection. In: 26th IEEE International Conference on Distributed Computing Systems, ICDCS 2006, pp. 66–66 (2006)
41. Zhengwei, G., et al.: P4P Pastry: A novel P4P-based Pastry routing algorithm in peer to peer network. In: 2010 The 2nd IEEE International Conference on Information Management and Engineering (ICIME), pp. 209–213 (2010)
42. Saleem, H.M., Hassan, M.F., Asirvadam, V.S.: Proxy-Based Selective Forwarding in Distributed Service Discovery Using Application Oriented Networking. Advances in Information Sciences and Service Sciences 4(8) (2012)