Analysis of searching mechanisms in hierarchical p2p based overlay networks

Isaias Martinez-Yelmo, Ruben Cuevas, Carmen Guerrero
Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid
Av. Universidad 30. 28911 Leganés, Madrid (Spain)
Email: {imyelmo,rcuevas,guerrero}@it.uc3m.es

Andreas Mauthe
Computing Department at InfoLab 21
Lancaster University
Lancaster LA1 4WA (UK)
Email: andreas@comp.lancs.ac.uk

Abstract— **This work presents a study of searching mechanisms in Peer-to-Peer (p2p) networks. The aim of this research line is to analyse cross-searching mechanisms that will allow the hierarchical interconnection of p2p networks. A set of relevant metrics for interconnection scenarios are defined to evaluate scalability, robustness and routing latency.**

I. INTRODUCTION

Overlay p2p networks have had a large impact due to they are a good mechanism to develop distributed applications using an underlaying network infrastructure. Here, it is introduced the problem of interconnecting different overlay networks to increase the availability of sources using a *Hierarchical Interconnection Overlay*.

II. P2P OVERLAY NETWORKS

Usually have been study the peer to peer algorithms from the point of view of efficiency in order to get the best overlay network in terms of robustness, availability of resources, low latency or low cost of maintenance. In a first approach *Unstructured Overlays* were developed. These overlays are based on a flooded request model with N hops TTL, for instance Gnutella [1]. Nevertheless, the consumed bandwidth increases with the number of peers, so the scalability of the network is worse than expected. There are several proposals to solve this problem as building routing tables between peers or making use of the super-peer concept.

Considering the limitations of peer-to-peer algorithms based on flooding, a next generation of *Structured Overlays* has been created as Chord [2] or Kademlia [3]. The idea of these algorithms is to provide semantic-free, data-centric references in a *n-bits* flat name space, all of them based on Distributed Hash Tables (DHTs). The base of these algorithms is due to each peer has a *Peer ID* and each object has a *Key ID* . The Peer ID must belong to the defined *n-bits* space an this ID is usually obtained by hashing the IP address of the peer. Once a peer has its Peer ID, it stores the information related with the objects on the network that have a Key ID next to it. Usually, the Key ID of an object is obtained by hashing the desired object. The routing algorithm on the overlay will lookup and get the information depending on the likehood function for the IDs on the flat name space. This function defines the overlay topology. Finally, it must be considered that if a peer wants to
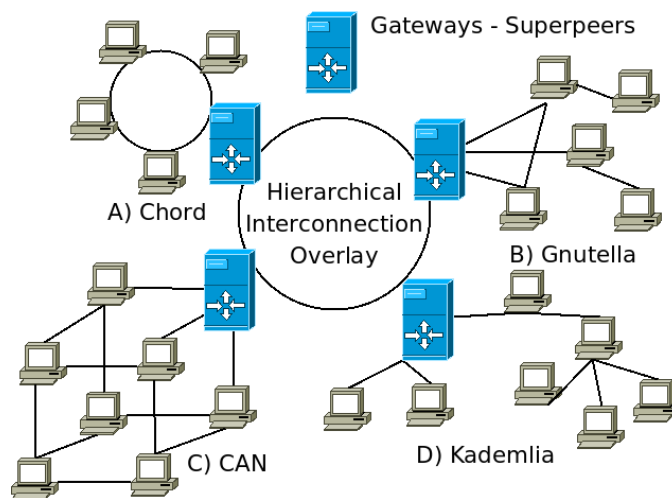

Fig. 1. Hierarchical Interconnected Overlays

find out a resource, it generates the Data key-ID to perform a query. This query is routed to the peer with the most similar Peer-ID to this key-ID. Once the peer storing the desired data is reached, it sends the desired data to the requester.

Many applications have been developed Gnutella [1], Overnet [5], Emule [6], FastTrack [4], but the most widely used overlays used on them have been Gnutella and Kademlia. None of them implements the same mechanism to look the information on the Overlay because any commom mechanism has been defined. Nevertheless, it would be desirable that a Query on an Overlay could be exported to another overlay to increase the availability of sources.

A *Hierarchical Interconnection Overlay* will be the most desirable method because overlays will be automatically interconnected by an overlay where only a subset of most computationally powerful peers will be subscribed (Fig.1).

III. P2P SEARCH MECHANISMS

Different search mechanisms could have used to search in P2P networks: boolean expressions, regular expressions, etc. However, searches based on keywords have been used in almost of the p2p networks. Table I summarises some of these search mechanisms:

Overlay	Algorithm	Keywords	Special features	Matching Processing
Gnutella	Gnutella	Plain Text	GGEP extensions	Neighbours
FastTrack	Gnutella	Plain Text	Substring matching	Neighbours
Overnet	Kademlia	1 Hashed Keyword	-	Querying Node
eMule	Kademlia	Hashed Keyword + Plain Text	Meta-information	Neighbours

TABLE I

Searching mechanisms comparison

i) **Gnutella:** Gnutella uses keywords in plain text. Neighbours have to match all the keywords with a file in order to send a reply to the Querier. Gnutella Generic Extension Protocol (GGEP) can be used to support additional functionalities like Metadata.

ii) **FastTrack:** As Gnutella uses keywords in plain text, but it also allows the search of substrings and to specify the number of matched keywords for a valid object. It is not necessary to match all the keywords on the query.

iii) **Overnet:** Kademlia protocol is used as overlay network. A distributed inverse index is created with the keywords of the objects stored on it which is also stored between all the peers. An Overnet peer sends n queries one per search keyword and all of them are hashed to fit in the flat namespace. Once all request have been received from the neighbours, only the nodes presented in all the requests are candidates to get the object.

iv) **eMule-Kademlia:** It is very similar to Overnet and also uses Kademlia as Overlay network with a reverse keyword index. The reverse keyword index not only stores the peers with the desired object, all the keywords with the associated object are also stored with it. Thus, a query is built hashing the first keyword of the search and is sent with all the keywords in plain text. When a peer receives the query, it looks for the indexes associated to the hash and only answers if all the keywords are matched.

IV. INTERCONNECTION MECHANISM AND ITS METRICS

Considering the mechanisms adopted by p2p applications to search information based only on keywords, interconnection to find information seems feasible. According to Fig.1, interconnection gateways must exchange the keywords to launch the query in the desired overlay. Each gateway should index their searches in a cache in order to not generate extra traffic.

Thus, considering the examples explained in the previous section, all p2p networks are candidates for interconnection except Overnet, because only the hash of the keyword is sent to perform the search, so it is impossible to know the keyword associated to the search. The actions that should perform the gateways on a Hierarchical Interconnection Overlay to locate a peer in other overlays will be the following if all gateways are subscribed to the interconnection overlay:

1. Peers send a Query to the Local Gateway.
2. Local Gateway extract keywords from the Query message.
3. Local Gateway send the keywords to the desired Foreign Gateways on the interconnection overlay.
4. Gateways on the foreign overlay constructs a Query message according to its local overlay.
5. Foreign peers send the results to the Foreign Gateway
6. Structured Overlays have to obtain the IP address of the peers that are storing the objects. A key-ID has not meaning to an unstructured overlay and a hash function perhaps will be different depending on the overlay. Thus, the better option is to get the IPs of the desired peers.
7. The results are sent to the Local Gateway that will send the response with the format used on the Local Overlay.

The next list of metrics will be interesting to be measured on the proposed scenario with the above mentioned steps:

a) $Accuracy\ Improve = \frac{Local\ Overlay\ Success\ Queries}{Interconnection\ Overlay\ Success\ Queries}$
b) $Delay\ on\ Foreign\ Queries$
c) $Memory\ Consumption\ on\ Gateways$
d) $CPU\ Consumption\ on\ Gateways$

Metrics a) and b) define if the adopted solution is good or not. They show the improve of additional available sources registered on other overlays and the needed delay to obtain them. On the other hand, metrics c) and d) are necessary to select the most suitable peers to be interconnection gateways and to study the scalability of the solution on these peers.

V. CONCLUSIONS

Hierarchical interconnection of different overlays is feasible. On the lower level any overlay network can be used if keywords are sent as plain text. The upper overlay allows to maintain interconnected the different lower overlays to allow cross-searches. With this approach, complexity on most of the peers have not to be increased developing several overlay clients, one per each overlay. Next step, it will be to study not only the search mechanism to locate the desired objects but also how to retrieve that objects from different overlays.

ACKNOWLEDGMENT

REFERENCES

[1] T. Kinblerg and R. Manfredi, "Gnutella 0.6," 2002. [Online]. Available: http://rfc-gnutella.sourceforge.net/src/rfc-0_6-draft.html
[2] I. Stoica et al, "Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications,"IEEE/ACM TRANSACTIONS ON NETWORKING, vol. 11, no. 1, p. 17, 2003.
[3] P. Maymounkov and D. Mazieres, *Peer-to-Peer Systems: First InternationalWorkshop, IPTPS 2002, March 2002. Revised Papers*, ser. Lecture Notes in Computer Science. Springer, 2002, vol. 2429/2002, ch. Kademlia: A peer-to-peer information system based on the XOR metric.
[4] Specification, "The fasttrack protocol," 2004. [Online]. Available: http://cvs.berlios.de/cgi-bin/viewcvs.cgi/gift-fasttrack/giFT-FastTrack/PROTOCOL?rev=HEAD&content-type=text/vnd.viewcvs-markup
[5] Specification, "The overnet protocol," 2003. [Online]. Available: https://opensvn.csie.org/mlnet/trunk/docs/overnet.txt
[6] R. Brunner, "A performance evaluation of the kad-protocol," Master's thesis, Mannheim University, Germany, November 2006.
[7] Web page: http://www.ist-content.
[8] Web page: http://www.biogridnet.org/