Journal of Mobile Communication 1 (1): 13-16, 2007

Using JADE Mobile Agents for Network Resource Discovery in Peer-to-Peer Networks

[1]K.L. Shunmuganathan, [2]K.V. Deeba, [2]K. Deepika and [3]V. Vasudevan
[1]Anna University, India
[2]Noorul Islam College of Engineering, Kumaracoil, Thuckalay, 629 180, Kanyakumari, Tamilnadu, India
[3]Arulmigu Kalasalingam College of Engineering, India

Abstract: Peer-to-peer networks continue to grow in popularity. However network resource discovery still remains a substantial problem within them. In this paper we will cover some of the more popular current solutions to this problem. We will then propose a JADE Mobile Agent based solution to allow for dynamic network resource discovery.

Key words: JADE Mobile agents, resource discovery, peer-to-peer

INTRODUCTION

Peer-to-peer, commonly known as P2P, is the movement away from the more traditional clientserver model to a network where each participating device is acting as both client and server[1]. Some of the main advantages of P2P are greater storage, more computer cycles and greater bandwidth. P2P technology is also gaining popularity due to the current economic slowdown, because users are looking for ways to get more use out of their existing hardware. Most Internet users have encountered some of the more popular P2P systems to date, for example Napster, SETI@Home, or ICQ. P2P is now starting to be used in many more application areas, including e-commerce. eMikola[2] are developing a product called Peer Switches, that is used for more efficient information distribution and is based on P2P technology. Another P2P based example is Groove, a product that allows users to create shared spaces over the Internet that can be used for collaborating projects, virtual meetings, or trading.For P2P applications to succeed, a number of the underlying technologies must be developed. In the remainder of this study, we will be focusing on the issue of efficiently and successfully locating network resources on a P2P network.

Software agents: Although there is no universally accepted definition of a software agent, most agree that it is a computing entity that performs some task or tasks on behalf of somebody or something[3]. There are many additional features that software agents may employ, such as possessing intelligence, or a user interface. However such features are not essential. When the software agent always operates within the system where it was started it is often known as a stationary agent.

JADE mobile agents: A JADE Mobile Agent is a software agent that has the additional property that it is not bound to operate only in the system in which it started. A JADE Mobile Agent has the unique property that during its lifetime it can be halted, its state and code moved to another computer on the same network and then continue executing from where it stopped executing on the previous computer. A mobile agent is autonomous because it may decide itself where it will go, what it will do there and how long it will exist for. However, its environment or other JADE Mobile Agents may also influence it. Although JADE Mobile Agents do not provide a solution to any previously unsolvable problems, they do have advantages over other technologies[4]. They can be used to benefit or to simplify different types of application areas. Some examples of these application areas include ecommerce, distributed information retrieval, telecommunication networks services and monitoring and notification[5].

JADE: There are various different implementations of JADE Mobile Agents. In the remainder of this paper we will be focusing on the JADE implementation of JADE Mobile Agents.The JADE project is a Java[6] based implementation that was originally developed by FIPA. However late in 2000 they made the project open source and it is now developed and maintained by the JADE community. Some of the reasons for Java's suitability for developing JADE are[7]:

- The popularity of Java makes JADE development quicker and cheaper.
- The portability of Java compiled code due to its write once, run anywhere architecture.

Corresponding Author: K.L. Shunmuganathan, Anna University, India

- The Java Virtual Machine has a built-in, fine-grained and very configurable security control mechanism.
- Java has built-in support for network programming.

THE PROBLEM

Currently all searching on the Internet and large networks is carried out by dedicated centralized search engines, such as Yahoo or Google. Developing such systems tends to be extremely expensive in terms of hardware, bandwidth requirements and also the specialized algorithms and software that are necessary. Most Internet search engines maintain a very large centralised database, which is updated by crawling the Internet and indexing websites. When a query is received the database replies with a list of websites that are deemed to be in some way related to the original query. So while current search systems may be suitable for general web searching they do have disadvantages. The current size of the Internet and the rate at which is it growing, make it impossible to visit every website and index it. The most web pages that any search engine has indexed is 1.3×109, yet it is estimated that this is less than 20% of web pages in existence under the .com domain. Another significant disadvantage is that indexing only works well on websites that contain static information, as the search engine may not visit the website regularly enough to be able to index new content.

These disadvantages, along with the general architecture of search engines, make them unsuitable for searching in large dynamic P2P networks. Also, a P2P network will only be as popular as the usefulness of the resources that can be located within it. Therefore being able to efficiently search for and successfully locate these resources is extremely important.

The proposed JADE mobile agent based solution: why use JADE mobile agents: JADE Mobile Agents reduce the need for bandwidth. Very often peers using a distributed protocol establish a communication channel between themselves and then perform multiple interactions over this channel. Each of these interactions generates network traffic. JADE Mobile Agents allow these interactions to be packaged together and sent as a discrete piece of network traffic. This then allows all the interactions to take place locally. JADE Mobile Agents also encapsulate all the required data within themselves. Therefore when a JADE Mobile Agent arrives on a computer it has all its data with it and does not need to communicate with any other computers. In a conventional search protocol all the raw data travels over the network to be processed, even though only a subset of this data may be needed. In this scenario, JADE Mobile Agents reduce the network traffic by moving the processing to the raw data, instead of moving the raw data to the processing. Finally, mobile agents can be very small in size, but can grow dynamically as they need to accommodate more data. JADE Mobile Agents are asynchronous. Therefore when a JADE Mobile Agent is dispatched there is no need to wait for it to return. Indeed the original peer does not even need to remain connected to the network while the JADE Mobile Agents are out. The JADE Mobile Agents can wait until the original peer is back on the network before attempting to return to it. JADE Mobile Agents are autonomous. This particularly suits network discovery, because the JADE Mobile Agent is learning about the network as it progresses through it. The JADE Mobile Agent will visit peers that were unknown when it was originally dispatched. At each peer it can make decisions based on its history of visited peers and the current peer. Information is being disseminated at every peer that the JADE Mobile Agent visits. Every peer benefits from accepting a visiting JADE Mobile Agent, because the JADE Mobile Agent will have either new or more recent information about resources. Also, every JADE Mobile Agent benefits from visiting a peer because it will learn of either new or updated resources. If the JADE Mobile Agents do not contain any new information they may be destroyed. Accepting and hosting JADE Mobile Agents requires the use of physical resources, such as memory and computer cycles. Should these become critically limited, it is easy for the peer to refuse further requests to accept JADE Mobile Agents until more physical resources become available. JADE Mobile Agents may easily be cloned and dispatched in different directions. This allows them to function in parallel. Although this causes more JADE Mobile Agents to be active on the network, it does ensure that the network resource discovery is completed sooner and therefore the mobile agents spend less time on the network. A JADE Mobile Agent based solution is very fault tolerant. Even if some of the JADE Mobile Agents are destroyed, all surviving ones will have a positive impact. Indeed, the dstroyed JADE Mobile Agents will have benefited every peer up to the point where they were destroyed. Finally, a JADE Mobile Agent based solution can be combined with successful features from other P2P based systems to provide an improved final solution.

What is a resource: A resource is something that may have a perceived value to somebody or something. Each peer participating in the network may make some of its resources available to other peers. We have tried not to make any assumptions on what a resource may be. There

are certain mandatory resources that a peer must posses to allow for the most basic network resource discovery. It must know what type of peer it is. The peer must be able to maintain a directory with details of each known peer, its type and its address. This directory must be updateable by visiting JADE Mobile Agents. There are many other additional resources that a peer may have and these may or may not be updateable by visiting JADE Mobile Agents. The resource may be a directory of files and their Networks locations known to the peer. These files might be a particular category, such as music, or movie clips. A complex software object that the visiting JADE Mobile Agent can interact with may be the resource. The resource may be available computer cycles, along with details of when they are available. The resource may be some dedicated hardware, such as a network router or switch.

Peer discovery algorithm: The main algorithm needed is used to decide how the JADE Mobile Agents will behave in their search for other peers. The algorithm is not trying to perform total network discovery because this is not at all scalable to very large networks. So the algorithm is making a trade-off between a more detailed network discovery and a more efficient and practical network discovery. After the JADE Mobile Agents have completed their work, the original peer will have built a view of the part of the network that it now knows about. This view is essentially an undirected graph. The JADE Mobile Agents have performed a basic Breadth-First search in parallel with each other. The algorithm adopted by us is not necessarily the best possible algorithm, as the algorithm was not the focus of our work. We recognize that graph theory is an area of research in its own right. Indeed, work has already been performed on viewing the World Wide Web as a graph and designing suitable searching algorithms based on this view. The algorithm we used is as follows[8]:

- A JADE Mobile Agent is created on a peer wishing to participate in the network. This peer is known as the creator peer for that particular JADE Mobile Agent and all future clones of it. The mobile agent updates itself with information about its creator peer and in particular notes the address of this peer as its home address. A journey-time for the JADE Mobile Agent is set in terms of the maximum number of peers that may be visited before it must return to its creator peer. Also a branching factor is set, that is used to determine how many times the JADE Mobile Agent may be cloned at any one peer. These two parameters control the depth and breadth of the search and therefore the maximum number of peers that may be visited.
- Initially the JADE Mobile Agent is given the addresses of some other peers participating in the network. Typically the number of addresses provided should be small and should be less than the branching factor. For best results these addresses should come from different sources. This increases the chances that these peers are operating in separate sub networks that are as yet unaware of each other.
- The JADE Mobile Agent clones itself enough times to allow a JADE Mobile Agent to be sent to each of these peers.
- After arriving at each peer the JADE Mobile Agent decrements its journey-time and updates the peer with information about its creator peer and other peers encountered along the route so far. The JADE Mobile Agent also updates itself with information on the current peer. If two mobile agents from the same creator peer arrive at the current peer within a preset time period, then the second JADE Mobile Agent destroys itself. This may lead to less information being acquired about the network. However, it keeps the information up to date and yet prevents peers that form cycles from having to deal repeatedly with JADE Mobile Agents from the same creator peer.
- If the JADE Mobile Agent's journey-time has expired, then it returns to its creator peer and updates its creator peer with all the information it has collected on its journeys. Otherwise the mobile agent clones itself enough times to allow a clone of itself to be sent to each peer that was known to the current peer before the JADE Mobile Agent arrived. By excluding previously known peers, the incidence of revisits is reduced.
- The process is continued from step 4.

Implementation and architecture: We implemented our JADE Mobile Agent based solution using the JADE Software Development Kit proposed by FIPA. The standard model of an agent platform, as defined by FIPA, is represented in the following Fig. 1.

Security: Security is generally a problem with JADE Mobile Agents. In our study we did not implement any specific security precautions per se, because we felt that

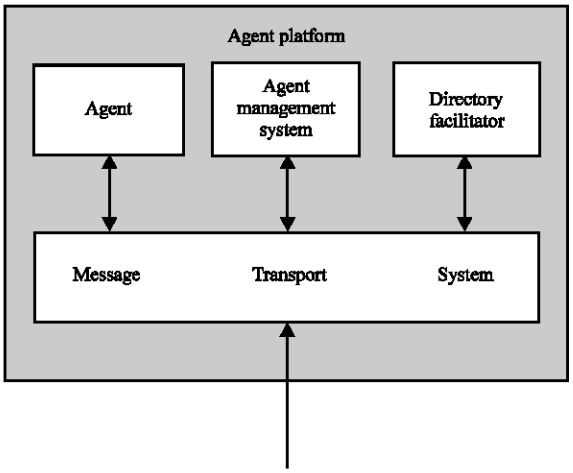

Fig. 1: Reference architecture of a FIPA agent platform

Fig. 2 : Security homepage

we did not need security beyond what is provided by the Java Virtual Machine. We did however use many of their features in different ways. JADE platform provides support for security in multi agent systems[9]. Such a created homepage of JADE for ensuring network security is shown in Fig. 2.

CONCLUSION

In this study we looked at the general problem of resource discovery on P2P networks. We looked at some of the current solutions and examined some of their strengths and weaknesses. We then proposed a JADE Mobile Agent based solution. We discussed the advantages of this solution. We described its architecture and its implementation using JADE.While there are several areas of the work presented here that require further investigation, there are two that particularly interest us. Firstly, we would like to assess the performance of our proposed solution in a large uncontrolled network, because so far all our testing has been in a controlled laboratory. Secondly, we would like to develop more JADE Mobile Agents that are more application specific and which deal with increasingly complex resources that are defined using XML.Finally from our work to date we believe that JADE Mobile Agents do provide a viable means of performing network resource discovery.

REFERENCES

1. Shirky, C, What Is P2P... And What Isn't, O'Reilly Network, 24/11/2000
2. General Magic, Inc., http://www.genmagic.com/
3. Caglayan, A. and C. Harrison, Agent Sourcebook. Wiley Computer Publishing.
4. Scholl, Napster protocol specification byanalysis, http://opennap.sourceforge.net/napster.txt
5. Findlaw, http://www.findlaw.com/napster/index.html
6. Java, Sun Microsystems, Inc., http://java.sun.com/
7. Green, S. *et al*., 1997. Software Agents: A review, Department of Computer Science, Trinity College Dublin.
8. Sedgewick and Robert, 1992. Algorithms in C++, Addison-Wesley.
9. Dan Connolly, Mobile Code Systems, http:// www.w3.org/MobileCode/