International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 12 – No. 20, March 2019 – www.ijais.org

Mobile Agents in Distributed Environment for Information Retrieval

Jackson Mwiti Isaiah
School of Computing and Informatics Meru University of Science and Technology, Kenya

Andrew M. Kahonge, PhD
School of Computing and Informatics,
University of Nairobi, Kenya

E. Abande, PhD
School of Computing and Informatics,
University of Nairobi, Kenya

ABSTRACT

With increase in amount of information being stored in remote locations, there is a need to have an efficient method that facilitates information retrieval. Most information retrieval operations use client server architecture which ends up taking much of bandwidth. Client server architecture moves data during operation and this takes too much bandwidth. This paper discusses mobile agents which present an ideal method that can be used in information retrieval due their ability to move the code instead of data and this property makes them suitable in saving bandwidth. Mobile agent is a technology generating a lot of interest due to its benefits in areas such as ecommerce, information retrieval. Mobile agent is a program that acts on behalf of the user and it is capable of moving from one location to another and performs action. The experiment was carried out to compare the performance of mobile agents and client server method in which it was found that mobile agents formed better that client server.

General Terms

Mobile Agents, distributed systems, information searching.

Keywords

Information retrieval, Agents communications, client server

1. INTRODUCTION

Distributed systems have continued to gain popularity due to the amount of data that need to be accessed. Use of client server architecture has presented bandwidth problems hence there is a need to address the problem. In today's information society, users are getting overwhelmed by the amount of information they are confronted on a daily basis. Most of the operating systems do not allow file manipulations without compromising on bandwidth and computer performance. Instead of moving large amounts of data to the search engine so that it can create search indexes, agents are usually dispatched in remote information sources, where they locally create search indexes that can later be shipped back to the origin [1]

Client server model has scalability and network bandwidth problems. To overcome that, mobile agent based model is the suitable technology for the applications such as information retrieval system [2]. Mobile agents (MAs) run at one location, move with their state to another host, and continue execution at that host. Mobile code and mobile objects are normally moved by an external entity while MAs are usually migrated autonomously.

Agent is an autonomous, problem-solving computational entity capable of effective operation in dynamic and open environments [3]. Agents are often deployed in environments in which they interact, and maybe cooperate, with other agents (including both people and software) that have possibly conflicting aims. Such environments are known as multi-agent systems. These mobile Agents can be distinguished from objects (in the sense of object oriented software) in that they are autonomous entities capable of exercising choice over their actions and interactions [11]. Agents are not usually invoked directly invoked like objects. However, they may be constructed using object technology. This paper discusses how mobile agents can be used in a distributed environment to retrieve information. The paper starts by introducing the concept of agent communication and its importance in mobile agent environment. Section three of this paper discusses code mobility which is the art mobile agent while section four tackles implications of mobile agents in a networked environment. Section five presents experimental results and discussion on the findings of this paper.

2. AGENT COMMUNICATION

When communicating with other agents, an agent uses a specific Agent Communication Language (ACL)[12]. An ACL provides agents with a means of exchanging information and knowledge. [4] Has gone as far to equate agency with the ability of a system to exchange knowledge using an ACL. Mobile agent communications are not only the communications between mobile agents themselves, but most importantly they also communicate with agents, agent servers and agent owners [5].

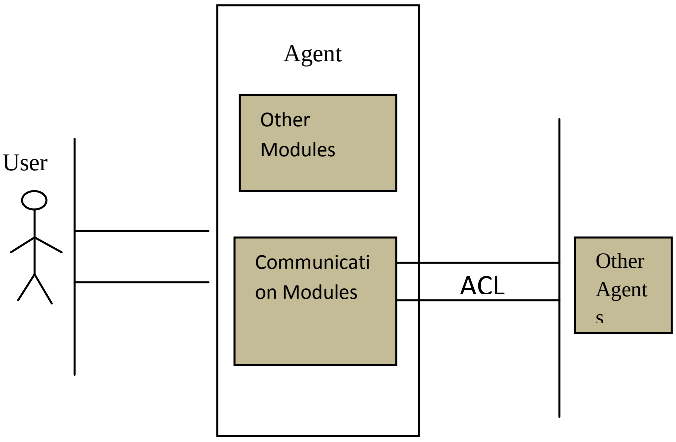

Fig 1: Agent Architecture

In various situations, mobile agents need to communicate with each other hence inter-agent communication is thus a fundamental facility in mobile agent systems. Although process communication has been so much a thing in distributed systems environments, agent mobility raises a number of new challenges in designing message delivery mechanisms for effective and efficient communications between mobile agents [12].

3. CODE MOBILITY

Each operation that involves code mobility can be divided into three steps: determining the code operation targets, transferring the code, and Integrating the code into the target system [6]. In static system architecture, the target determination step can be done at compilation time. If the system architecture is dynamic, such that it is determined at run time, then the operation targets should be computed immediately prior to transferring the code. Mobility is the primary characteristic of mobile agent. Mobility makes use of process migration, which consists of transferring a process from one computer to another [7].

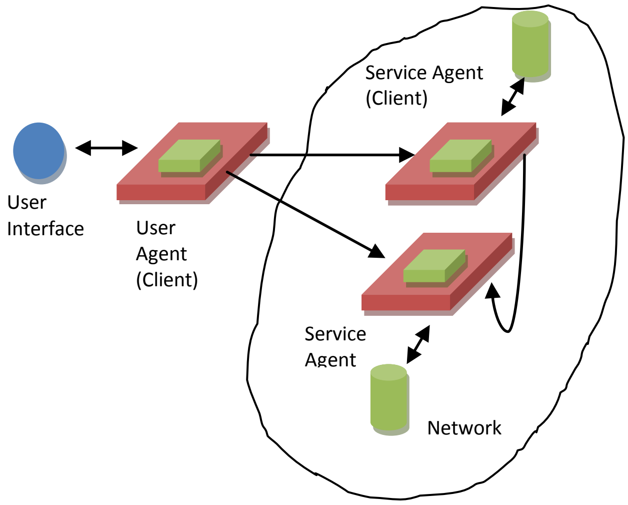

Fig.2: Mobile Code Systems

Following the target determination, the code is transferred by applying one of the design paradigms for code mobility, which extend the traditional client-server in which the code can be integrated with the local target system by activating an instance of it, connecting it to existing data or code, or continuing its transfer over the network [6].

Mobile code is an important programming paradigm and opens up new possibilities for structuring distributed software systems in an open and dynamically changing environment [8]. It can improve speed, flexibility, structure, or ability to handle disconnections and it is particularly well suited if adaptability and flexibility are among the main application requirements. It has applications in many areas, such as mobile computing, active networks, network management, resource discovery, software dissemination and configuration, electronic commerce, and information harvesting here are usually two kinds of code mobility; strong mobility and weak mobility [9]. Strong mobility is the ability of an MCL (called strong MCL) to allow EUs to move their code and execution state to a different site. Executing units are suspended, transmitted to the destination site, and resumed there. For instance, Telescript is a strong MCL. Weak mobility is the ability of an MCL (called weak MCL) to allow an EU in a site to be bound dynamically to code coming from a different site [9]. This encompasses two cases: either the EU dynamically links code downloaded from the network (as can be done in Java), or the EU receives its code from another EU. In the latter case, two more options are possible. Either the EU in the destination site is created from scratch to run the incoming code, or a pre-existing EU links the incoming code dynamically and executes

4. IMPLICATIONS OF MOBILE AGENTS

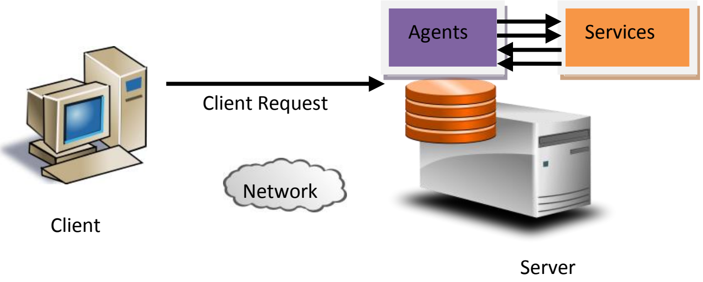

Fig: 3: Client server architecture

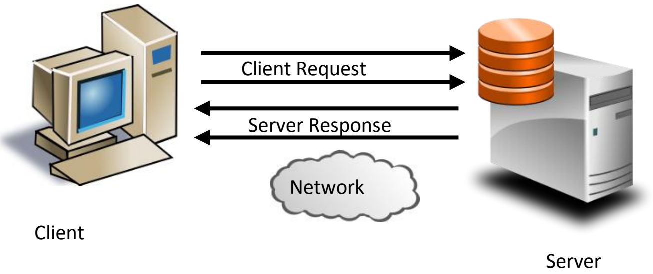

Fig: 4: Agent Based Architecture

Distributed systems often rely on communications protocols that involve multiple interactions to accomplish a given task. This is especially true when security measures are enabled and this result is a lot of network traffic.

Mobile agents can increase the performance in these areas by achieving the following tasks: Reduce the network load, MAs allow users to package and migrate their operations to be carried out locally. This is useful where huge volumes of data are required to be processed; the data will be manipulated locally rather than transferred over the network. In other words, move the computation to the data rather than the data to the computation [10].

5. EXPERIMENTS AND RESULTS

The implementation was carried out using mobile agent platform based on Java programming language due to its platform independence which allows for serialization and persistency and comes with security built into the virtual machine.

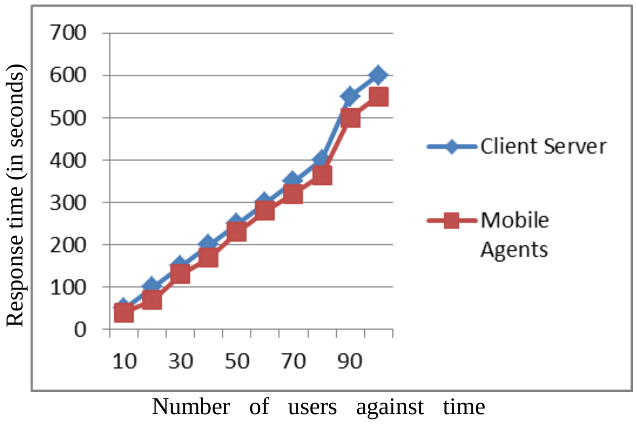

Fig 5: Comparison between client server and agent based application

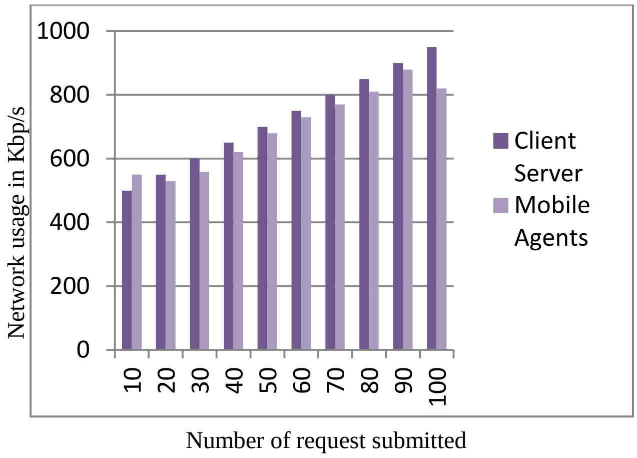

Fig 6 Network Performance

From the above results, mobile agent seems to perform better than client server method which seems to experience delays. For this case mobile agent are far better that client server method which take more bandwidth. Mobile Agents transparently use the network to accomplish their tasks, while taking full advantage of resources local to the many machines in the network. They process data at the data source, rather than fetching it remotely, allowing higher performance operation. They use the full spectrum of services available at each point in the network, such as GUIs at the user and database interface on servers.

The client server seems to be taking too much time to respond as compared to agent based method. This can be attested by the fact that Client makes a request by sending a message to the server and the Server unwraps the message, decodes it and processes the request and sends the reply in the form of a similar message back to the Client. This operation overwhelms the network whereby bandwidth is greatly consumed. Whereas the mobile transport the code to do the processing locally, client server model continue to transmit data over the network for processing.

6. ACKNOWLEDGMENTS

Our thanks to Meru University of Science and Technology as well the University of Nairobi whose resources were vital to achievement of this research.

7. REFERENCES

[1] Mobile Objects and Mobile Agents: The Future of Distributed Computing? Danny B. Lange-1998 Satoh, I., 2006. Building and selecting mobile agents for network management. International Journal of Network and Systems Management 14 (1), 147–169. March.

[2] Kannammal, A., V. Ramachandran and N.C.S.N. Iyengar, 2006. Reliable multi agent system for business applications. Acad. Open Internet J., 18. http://www.acadjournal.com/2006/v18/part6/p2/

[3] Luck, Michael, Peter McBurney, and Chris Preist. Agent technology: enabling next generation computing (a roadmap for agent based computing). AgentLink/University of Southampton, 2003.

[4] M.R. Genesereth and S.P. Katchpel, "Software Agents," Comm. ACM, Vol. 37, No. 7, 1994, pp. 48–53, 147.

[5] A mobile agent platform for distributed network and systems management Damianos Gavalas*, George E. Tsekouras, Christos Anagnostopoulos Freeman Yufei Huang -Communication Infrastructures and Protocols for Mobile Agent December 2000

[6] Reinhartz-Berger, I., Dori, D., & Katz, S. (2002). OPM/Web–object-process methodology for developing web applications. Annals of Software Engineering, 13(1-4), 141-161.

[7] Loke, S.W., Rakotonirainy, A., and Zaslavsky, A., (2000). Enabling Awareness in Dynamic Mobile Agent Environments. (Short paper). Proceedings of the 15th Symposium on Applied Computing (SAC 2000), Como, Italy, ACM Press.

[8] Li Jingyue (2002) Code Mobility Overview (Essay for DIF 8914)

[9] Cugola, G., Ghezzi, C., Picco, G. P. and Vigna, G. (1997), "Analyzing Mobile Code Languages", Mobile Object Systems, Lecture Notes in Computer Science, No. 1222, pp. 94-109, Springer-Verlag (D), February.

[10] Bigus, P., Bigus, J., (1998), Constructing Intelligent Agents with Java, John Wiley &Sons Ltd, England, ISBN: 0 471-19135-3.

[11] Lin, Hong, ed. Architectural design of multi-agent systems: technologies and techniques: technologies and techniques. IGI Global, 2007.

[12] Cao, J., & Das, S. K. (2012). Mobile Agents in Networking and Distributed Computing (Vol. 3). John Wiley & Sons.