International Journal of Research
Available at https://journals.pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 06 Issue 10
September 2019

Introduction to Mobile Agent Technology

Mar Lar Htun[1], Win Myat Thuzar[2], Thae Thae Han[2], Thida San[3]
[1]Fauclty of Computer Science, University of Computer Studies (Pyay), Myanmar
[2,3]Information Technology Supporting and Maintenance Department, University of Computer Studies(Meiktila), Myanmar
marlarhtun19@gmail.com

Abstract:

Mobile agent technology offers a dramatically evolving computing paradigm in which a program, in the form of a software agent, can suspend its execution on a host computer, transfers itself to another agent-enabled host on the network, and resumes execution on the new host. It can communicate in an agent communication language; it is also a computer system in a complex environment that realize a set of tasks and goals it was designed for. It can be deployed in many complex applications such as Internet, Mobile Data Computing, Electronic commerce, Networking, Manufacturing and scientific computing. Today's mobile agents can be characterized in a number of ways ranging from simple distributed objects to highly organized intelligent software.

Keywords

mobile agent, Jini, Aglets

1. Introduction

Mobile Agent is an agent that can migrate between various nodes in the network and act on behalf of a user or owner to accomplish tasks. It motivates force in reducing network traffic and overcoming network latency. Mobile agent technology is a new networking technology that deals with both forms of logical and physical mobility. It offers a new computing paradigm in which a program, in the form of an intelligent software agent can suspend its execution on a host computer, transfer itself to another agent-enabled host on the network and resume execution on the new host. Mobile agent is a technology attracting interest from the fields of distributed systems, mobile computing, electronic commerce, and information retrieval on the World Wide Web. Mobile Agent is a type of software system which acts “intelligently” on one’s behalf with the feature of autonomy, learning ability and most importantly mobility. Now mobile agents are gaining interest in the research community.

2. Mobile Agent

Mobile Agent is free to travel among the hosts in the network. Created in one execution environment, it can transport its state and code with it to another execution environment in the network, where it resumes execution. It has the unique ability to transport itself from one system in a network to another in the same network. How can it start the execution is not bound to the system [1]. The term "state" typically means the attribute values of the agent that help it determine what to do when it resumes execution at its destination. The term "code" in an object-oriented context means the class code necessary for an agent to execute. A mobile agent initially resides on the home machine. The agent is then dispatched to execute on a remote computer called a mobile agent host. When a mobile agent is dispatched, the entire code of the mobile agent and the execution state of the mobile agent is transferred to the host. The host provides a suitable execution environment for the mobile agent to execute. The mobile agent uses resources of the host to perform its task. After completing its task on the host, the mobile agent migrates to another computer. Since the state information is also transferred to the host, mobile agents can resume the execution of the code from where they left off in the previous host instead of having to restart execution from the beginning. This continues until the mobile agent returns to its home machine after completing execution on the most recent machine in its program [6].

Mobile agents come in a variety of flavors and perform numerous functions:

- An **information agent** searches for information residing on remote nodes and reports back to the source
- A **computation agent** seeks underutilized network resources to perform CPU-intensive processing functions
- A **communication agent** couriers messages back and forth between clients residing on various network nodes

International Journal of Research
Available at https://journals.pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 06 Issue 10
September 2019

2.1. Mobile Agent Framework

This mobile agent framework consists of two main components. The first component is the mobile agents themselves; that is, entities with some job to do. The second component is the mobile agent host(s), the service that provides the mobile agents' execution platform. In a distributed environment, we can have one-to-many agent hosts as well as one-to-many agents. To be an active agent platform, a given node in the system must have at least one active agent host. Figure 1 depicts the framework components.

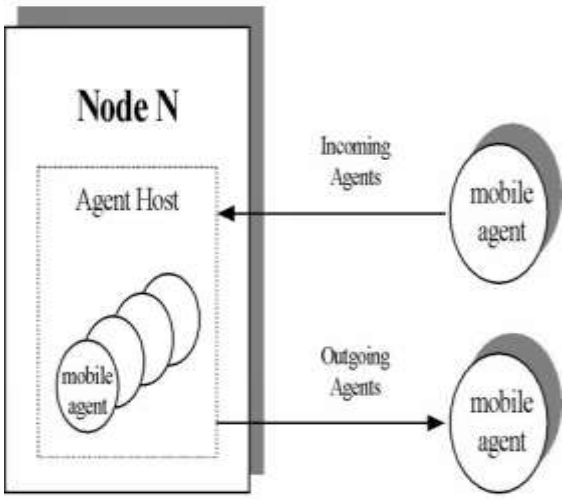

Figure 1. Framework Components.

These two components map quite nicely to the Jini model. Jini, at the highest level, provides the infrastructure that enables clients to discover and use various services. Jini also provides a programming model for developers of Jini clients and services. In the context of this mobile agent framework, the agent host(s) provides Jini services. The mobile agent(s) is the Jini client.

Jini services register with one or more Jini lookup services by providing a service proxy for perspective clients. In turn, clients query the lookup service(s) for particular services that might be of interest.

2.2. Mobile Agent Characteristics

Mobile agents have different kinds of characteristics;

- **Autonomous**: An agent is able to take initiative and exercise a non-trivial degree of control over its own actions.
- **Interactive:** means Mobile Agents should communicate with other agents and their environment. In addition, mobility is the most important property in the Mobile Agent concept, where agent migrated from one node to another within the same environment or in different environment.
- **Coordinative;** means perform data transfer with other agents in a given environment.
- **Proxy:** Mobile agents may act on behalf of someone, so they should have certain degree of autonomy.
- **Ragged**: Mobile Agents should have the ability to deal with the errors whenever occurred.
- **Proactive**: means they should be goal oriented.
- **Cooperative:** means coordinate with other agents to achieve a common goal. Mobile Agents should have the capability of learning the current environment and modify its behavior based on this information.
- **Intelligent**: means Mobile Agent should be too smart in order to act efficiently.

Based on these characteristics, numbers of agents have been proposed by the researchers. It is not required that agents have all these properties .This is determined by the purpose that agents have to achieve.

3. Applications of Mobile Agent Technology

Mobile agent technology has been used in many areas from network management task to information management. Mobile agents have significantly used in the wireless environment because it supports the disconnected mode. As the mobility has been migrated from the PDA (Personal Digital Assistant) to the network, the PDA could be disconnected and when the mobile agent finishes its job then they can reconnect in the network with the desired result from the agents. This gives advantage over conventional communication methods such as client/server model, RPC etc. Mobile agent technology is frequently used in other applications such as data warehouse, software updates, information management tasks such as searching for information, information filtering, information monitoring etc. Mobile agent technology has been also applied in M-commerce for information retrieval regarding the lower price of any particular item. In this, mobile agent have been issued by the PDA and disconnected from the network. This agent will then roams from one node to another on the internet and compare the price. When the mobile agent finds the cheapest result, it will then return to the PDA. This concept of information retrieval is further explained by considering a case study of multiparty event scheduling.

4. Technological Base of Mobile Agents

In order for mobile agents to be prevalent a suitable technology base needs to be installed on a large number of machines on the net. Selecting the technology is not a small task: it needs to fulfill all the characteristics discussed above and it needs to be real, deployable. No system perfectly meets all the requirements, but the Java system from Sun Microsystems is the best available system today. Java is a combination language and operating system that has working implementations and a large degree of market acceptance. Java does not address the issue of security guarantees for agent code running on a Java virtual machine. Secure hardware implementations of Java should be possible but are unlikely to be commonly used. Several obfuscation programs exist to hide the structure of transported agents, but obfuscation does not stop determined attackers. It is not clear that it is possible to protect agents from servers that deliberately misexecute their code in any system.

Several systems exist today to implement mobile agent facilities in Java. One limited application is found in ``servlets,'' Java code that runs on Web servers (analogous to applets running on a browser). JavaSoft's Jeeves is the most active servlet platform under development now. Jeeves is an HTTP server written in Java; it can be extended by loading in new Java classes from either the local disk or the network. One possible application is to allow clients to upload a servlet into a Jeeves server to perform computation on the server. Servlets run in their own restricted environment but are allowed to open a network socket back to their home to perform arbitrary communication. Jeeves is essentially a simple mobile agent system but is restricted in that servlets are intended to run in only limited ways inside the HTTP server. Servlets are not commonly expected to travel multiple times.

Several preliminary efforts are under way to create full mobile agents in Java. Implemented packages include Mole from the University of Stuttgart and Java2Go from Berkeley. The SAIC also has published an API for mobile agents. All of these systems are based on Java 1.0: they use some sort of ad hoc implementation of object serialization and remote method invocation to create mobile agents. These systems are of interest in their different design approaches but probably will not be contenders for serious world-wide deployment.

Finally, a promising mobile agents project called ``aglets'' is under development at IBM. Aglets rely on two basic specifications: an API for aglets (J-AAPI) and the Agent Transport Protocol (ATP). IBM has explicitly stated their intention to make aglets ubiquitous. ATP and J-AAPI have been put forward as standards. The specifications and implementations are available for free (with a restricted prerelease evaluation license).
Aglets implement mobile agents in a straightforward way. Resource providers run servers that accept agents; anyone can upload an aglet into a server. Java and the J-AAPI give aglets portability. Aglets rely on the object serialization and remote method of invocation of Java 1.1 for effective network communication. Server security is handled by special aglet security managers. The security system is configurable and should be flexible enough to allow a variety of options, including policies based on the upcoming code signature specification. Aglets do not address agent security or resource accounting. A usable alpha version of aglets is currently being shipped; however, it requires Java 1.1 which is itself only available in beta for a few operating systems. As the software becomes more mature aglets looks to be the most likely platform for future mobile agent research.

5. Conclusions

Mobile agents are an emerging technology that makes it very much easier to design, implement, and maintain distributed systems. Mobile agents reduce the network traffic, provide an effective means of overcoming network latency, and perhaps most importantly, through their ability to operate asynchronously and autonomously of the process that created them, helping to construct more robust and fault-tolerant systems. There is a strong case for the use of mobile agents in many Internet applications. Moreover, there is a clear evolutionary path that will take user from current technology to widespread use of mobile code and agents within the next few years. Once several technical challenges have been met, and a few pioneering sites install mobile-agent technology, use of mobile agents will expand rapidly..

6. References

[1] Adel Aneiba, S. J. Rees, “Mobile Agents Technology And Mobility”, *Faculty of Computing, Engineering and Technology,Staffordshire University,PO BOX 334, Beaconside, Stafford ST18 ODG, UK*

[2] Lange, D., and Oshima, Mitsuru, Seven Good Reasons for Mobile Agents, *Communications of the ACM, Vol.*
42(3): 88-89.

[3] Mats Persson, "Mobile Agent Architectures", FOA-R--00-01700-503—SE, December 2000, ISSN 1104-9154

[4] Mohit Mittal, "Moblie Agent", *International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 1 Issue 8, October - 2012*

[5] Wayne A Jensen, Countermeasures for Mobile Agent Security, Computer Communications, *Special Issue on Advances in Research and Application of Network Security, Elsevier Science BV, Summer 2000*