

Towards the Distributed Processing of Mobile Software Agents

David K. Wang, Sc.D.
Thomas University
1501 Millpond Road
Thomasville, GA 31792
dwang@thomasu.edu

James K. Wang, Sc.D.
Florida State University
Admin Info. Sys. Coordinator
Tallahassee, FL 32306
jwang@seas.gwu.edu

ABSTRACT

The mobile software-agent paradigm allows the software agent's data or state to relocate from one machine to another. The actual executable code of the mobile software agent could migrate and not be constraint to a single machine or a specific operating environment. Currently, deficiency of open and close standards for mobile software-agents has contributed to the deterrent of its favorable public acceptance. Regardless, numerous efforts are emerging to standardize protocols and application programming interfaces, promising interoperability of mobile software-agents from various vendors. Contemporary development of organizations that focus on various commercial aspects of applied mobile software-agents could increase the public awareness and desire toward this novel technological investment.

1. INTRODUCTION

In essence, a software agent is a piece of software executable code capable of performing certain preprogrammed operations for a user. Quintessential software agents, such as a network management agent and an intrusion detection system agent, are designed to be stationary for a single machine which could perform a specific set of tasks based on external factors and particular user inputs.

A mobile software agent, on the other hand, is a software agent designed to migrate from a single machine to another and perform certain preprogrammed operations at different machines for a user [1]. While in the traditional static paradigm, the data rallies around the executable code; in the unique mobile paradigm, the executable code rallies around the data. The mobile software-agent paradigm allows the software agent's data or state to relocate from one machine to another. The actual executable code of the mobile software agent could migrate and not be constraint to a single machine or a specific operating environment.

2. EXPLICIT ADVANTAGES OF MOBILE SOFTWARE AGENTS

Inherent explicit advantages [3] of the mobile software-agent paradigm are abundant. For instance, take the case of the

ubiquitous internet search, where massive amounts of data need to be intelligently quarried and processed. The process of rallying the executable code around the data could provide notable speed up of the overall throughput while reduces the heavy processing workload at the network server.

During the same time that the mobile software agent is performing its task, the user could detach the originating client machine from the network completely without terminating the already on going task process. After the mobile software agent finishes its assigned tasks, it pauses at a specific network server station to allow the user to attach the originating machine back onto the network before it actually returning back with the specific task result. The mobile software-agent paradigm of able to continue performing certain task beyond the time of detachment from the network is very useful in a dialup or a mobile communication scenario.

Mobile software-agent framework is a distributed processing paradigm that shifts away from the traditional central processing paradigm. In this distributed processing paradigm, it is beneficial for central servers to delegate central processing to the client processors where it is needed, rather than performing all the intelligent data quarrying and processing at one central clustered location. In order to achieve a proactive intranet-style network on the wide-area networked internet, a fine tuned network intrusion detection framework would prove to be beneficial.

3. PRODUCTIVE APPLICATIONS OF MOBILE SOFTWARE AGENTS

Mobile software-agent paradigm could serve to reduce the intricacies of the existing network computing model. Some of the possible applications of mobile software agents could include:

Utilization of mobile software agents to assist bidders and sellers in distributed electronic auctions and bargainings. For a more convoluted, numerous staged auctions, unleashing several mobile software agents may be required to assist a single bidder for achieving cooperation generated from each bidder's requirements derived from previous related auctions and sales. Afterward, mobile software agents could automatically alert bidders of auction

outcomes through wearable computer, smart phone, e-fax, e-mail, or pen-based computer.

Mobile software agents are appropriately suited to the unscheduled, distributed [4], and event-driven processing of software applications, e.g. work flow automation. For instance, electronic commerce may involve processing of an order or a signature authorization that could harness mobile software agents' ability to deliver vital information and methods. It could also analyze, pause, and make mobile decisions based on user input events.

Mobile software agents provide a tremendous opportunity to alleviate heavy centralized server cluster-centric processing in network system administration. It manages to shift heavy processing from network servers to the network client devices via unleashing mobile software agents to execute the preprogrammed task at the network node or device instead of at the cluster-centric network servers.

Management of smart help desk, case-based reasoning facility, or enterprise call center, may sometimes utilize information retrieval techniques based on metasearch, which could involve massive distributed database searches over the internet. Mobile software agents could be applied to process numerous instantaneous queries at different database locations across the internet, via bringing together numerous relevant informational contents into one package format which then could be present to the user.

Mobile software agents are used to resolve problems that arise from a detached network user in the group scheduling process that is associated with the groupware management. Group users may utilize mobile software agents to maintain schedule or prioritize information, and these mobile software agents could combine their forces to resolve scheduling conflicts in a growing, expanding, and distributed fashion.

4. PREREQUISITES FOR A USEFUL MOBILE SOFTWARE AGENT FRAMEWORK

Numerous ideas started out in the national university laboratories with a handful of research results, products, and concepts have been successfully introduced to the commercial world, i.e., Ethernet and Netscape. Before the framework of mobile software agent becomes a popular manifestation in the network communications world, several issues of vital interest would warrant a visit:

Open Network Issue: Internet is a massive interconnection of numerous types of open hardware architectures created by various vendors. A mobile software agent should be able to migrate across a variety of operating system platforms, be able to interpret open

programming source, and be able to work on every platform with compatibility and flexibility.

Stability Issue: Framework of the mobile software agent should be stable in the occurrence of recurring network detachment, system crashes, and network traffic jam. Automatic crash recovery, re-connectivity, and emergency software-agent self-repair are indispensable.

Itinerant Issue: The framework should automatically resolve issues that arise involving migratory code of the mobile software agent from a network server to another. At the same time, it should be able to furnish a secure surrounding for the mobile software agent on a network server.

Authorization Issue: In order to reach the true potential as a viable internet computing framework, mobile software agent applications are steered toward accessing information that situates within corporate enterprises. That is, mobile software agents should be able to bypass firewalls. However, prohibited by the individual corporate policy and a variety of network-related vulnerability issues, a rational company would never grant access for a mobile software agent to roam around their private exclusive-enterprise intranet. Some network-related vulnerability issues [5, 6] surrounding mobile software agents that warrant a visit are as follows:

Shielding the network server from other network servers to be sure that any software executable code presents itself to the mobile software-agent server is a documented network server. This would involve a classification and documentation scheme for network servers in question.

Shielding the mobile software-agent from unauthorized servers to be sure prior to sending the mobile software-agent toward a particular network-server destination, the particular network-server is capable of performing tasks assigned to the mobile software-agent without deviating from its original mission.

Shielding the network server from unauthorized mobile software-agents to be sure that they can not acquire access to private enterprise data on the network server. This requires a smart classification and authorization scheme combines with compatible and flexible access control rules for resource protection based infrastructure.

Shielding the mobile software-agent from other software agents by preventing other mobile software-agent or snooper on the network from acquiring a mobile software agent's executable code and data. This would involve encryption when the mobile software-agent is roaming in internet.

Remote Manageability Issue: The usefulness of able to remotely install, upgrade, manage, and configure network servers in the mobile software-agent infrastructure is beneficial for the mobile wide-area network implementation and manifestation.

Network Communication and Coordination Issue: One of the interesting mobile software-agent traits is its ability to cooperate with other mobile software agents. It is possible to achieve network communication via successful coordination between mobile software agents and message handling. Another noticeable benefit regarding mobile software agents is its ability to roam the internet and bring back results to the user; in some scenarios, it may even be favorable for a user to interact with the remote mobile software-agent.

Miniaturized Footprint Interface Issue: Some of the most exhilarating applications of mobile software-agents involve personal digital assistants and other miniaturized communications devices, along the line of smartcards and intelligent telephones. Although currently it is not be feasible to fit an entire framework of mobile software-agent server onto miniaturized digital devices; to reduce footprint, it is viable to design a miniaturized interface as a part of the mobile software-agent infrastructure.

Legacy Systems Access Issue: In order to fully utilize the internet, mobile software agent framework would be able to provide a compatible, flexible, protected, controllable access to legacy systems' data and resources.

Search and Resource Management Issue: Since resources on the internet can be transitory, mobile software agents are able to quickly locate resources that have been updated, re-linked, or re-directed. Incorporating the network server's resource directory service could be highly beneficial.

Tracking and Controlling Issues: Mobile software-agent utilities are to be use for mobile software agent tracking purposes throughout the deployed wide-area network geography. These utilities would allow mobile software agent managers to debug, control, and reprogram them. In case of derailed mobile software-agents, utilities could then be call in to disable the offbeat mobile software-agents.

5. TECHNOLOGICAL OUTLOOK OF MOBILE SOFTWARE AGENT

In spite the fact that few commercially available mobile software-agent software applications currently available in the commercial arena, numerous mobile software-agent development tools are already available to developers. Hence, several observed tendencies outlined below indicate an otherwise favorable future for the mobile software-agent technology.

6. POPULAR PLATFORM OF THE WIDE-AREA NETWORK AND INTERNET

As ever increasing enterprises embrace mission-critical operations via internet that traverse across the far reaches of geographical locations. No doubt, mobile software-agents could become a ubiquitous computing paradigm of choice that could fully harness the indisputable wealth of information offered by the internet. The Internet's popularity makes it a realistic platform for mobile software-agents. The race toward wide-area connectivity and wide spread acceptance by typical corporate users would determine the ultimate proliferation of this technology.

7. TAXONOMY OF MOBILE SOFTWARE-AGENT APPLICATIONS

For the present and coming decades, electronic commerce is a transaction methodology that has become a segment possessing the highest potential-growth within the network arena. Mobile software-agent technology could very well become an integral part in the next generation of electronic commerce.

In recent publications, there has been discussion regarding how best to apply the mobile software-agents to peak users' need in keeping up with the latest advancement in software technology. Perhaps, a demonstration of an incontestable application could substantiate its usefulness. At this time, it may be true that mobile software-agents have no measurable impact on traditional software applications. However, it is fast becoming an entirely new rationale of software applications gears toward resolving the technical issues of electronic commerce.

Software applications, which involves internet-based electronic auctions, transaction processing, and automated distributive resource allocation, demand a network computing paradigm which delivers immense flexibility in mobile components coordination which aspires a promise of secure transactions for the sporadically linked network.

8. TOWARDS THE STANDARDIZATION OF MOBILE SOFTWARE-AGENTS

The deficiency of open and close standards [6] for mobile software-agents has contributed to the deterrent of its favorable public acceptance. The current condition is verifiable with the utilization of an internet metasearch engine. Regardless, numerous efforts are emerging to standardize protocols and application programming interfaces, promising interoperability of mobile software-agents from various vendors. Development of organizations that focus on various commercial aspects of applied

mobile software-agents could increase the public awareness and desire toward this novel technological investment.

While scholars agree that diversity could sustain novel ideas, in order to reach its destined commercial potential, developer has to refine the mobile software-agent technology's existing empowering features that is capable of providing unctuous mobility throughout the internet. Due to various non standardized legacy proprietary servers and devices still exist within the vast network of different corporate enterprises, what is delineated on the paper may reach its typical operating potential and bypass the optimum operating potential in the real world. Several prevalent provisions facing commercial mobile-software-agent vendors before eventual public acceptance may specifically involve impervious security system, proactive fault tolerance, effective administration of complicated network mobile software-agent or server, and proper cataloguing of mobile software-agent's actual effectiveness.

9. REFERENCES

- [1] Fritz Hohl (Eds.) Kurt Rothermel. *Mobile Agents, Proceedings of the Second International Workshop, MA'98*. Springer Verlag, Germany, 1998.
- [2] M. Breugst. Mobile agents – enabling technology for active intelligent network implementation. *IEEE Network*, pages 53-60, May 1998.
- [3] D. B. Lange and M. Oshima. Seven Good Reasons for Mobile Agents. *Communications of the ACM*, 42(3): 88-91, March 1999.
- [4] M. Feridun, W. Kasteleijn, and J. Krause. Distributed Management with Mobile Components. In *Proceedings of*

the IFIP/IEEE International Symposium on Integrated Network Management, pages 857-870, May 1999. Boston, Massachusetts.

- [5] G. Karjoth, D.B. Lange, and M. Oshima. A security model for aglets. *Lecture Notes in Computer Science*, 1419: 188-201, 1998.
- [6] C. Bryce and J Vitek. The JavaSeal Mobile Agent Kernel. In *Proceedings of the IEEE First International Symposium on Agent Systems and Applications and Third International Symposium on Mobile Agents*, pages 103-116, October 1999.

10. INFORMATION ON AUTHORS

David K. Wang holds a doctorate from the School of Engineering and Applied Science at The George Washington University since 1995. Prior to his association with Thomas University he has been the executive officer for two high-tech corporations. Escalating through the ranks, his previous title is senior systems engineer and later on senior program manager/ chief engineer at the greater Washington area.

James K. Wang holds a doctorate from the School of Engineering and Applied Science at The George Washington University in 2001. He is currently an Administrative Information Systems Coordinator for the FSU provosts' office and serving as an executive officer for a high-tech corporation. Previously, he works at the U.S. Department of the States and later on at the National Aeronautics & Space Administration located in the greater Washington vicinity.