Mobile Agent Paradigm: A Tactic for Distributed Environment

Balram Singh
M.Tech Student, JEC Kukas, Jaipur

Shankar Singh
M.Tech Student, JEC Kukas, Jaipur

Narendra Kumar Agrawal
Reader & HOD, JEC Kukas, Jaipur

Abstract- The mobile agent paradigm has revolutionized the distributed computing environment. There are different approaches used in distributed computing. These are - client- server architecture, remote procedure architecture, code on demand architecture and mobile agent architecture. The mobile agent technology is built upon the advancement in computing and communication technology. Further, it suits homogeneous as well as heterogeneous networks either using a wired or wireless media. It is simple, light in weight and performs the desired task efficiently, moreover this paradigm does not load the network and also the memory requirement is low. There are different programming languages used for writing programs that incorporate the mobile agent paradigm. Each of these languages has certain characteristics, which make them more suitable for a certain type applications that make use of mobile agent paradigm. Keeping in view the advantages and limitations of mobile agents it is beneficial for us to use the mobile agent paradigm. PMADE (Platform for Mobile Agent Distribution and Execution) is framework for large-scale multi-agent systems. The work presented makes use of mobile agent paradigm on PMADE platform for the process of data mining. The mobile agent paradigm is implemented on this platform and the process of data mining is performed and the desired output is presented.

Keywords- Distributed Computing, Mobile agent, Data Mining, PMADE platform.

I. INTRODUCTION

A computer network is a set of computers or devices connected to each other with the ability to exchange data. Network users are able to share files, printers, and other resources; send electronic messages; and run programs on other computers. Computer network play an important role in today's life. It would be appropriate to learn the world as "global village", because of computer network. Without networks almost all communication in the world would cease. Modern amenities like- televisions, telephone, the Internet, etc are available today due to computer network. Components of Network layer are:

1.1.1 Application software: - consists of computer programs that interface with network users and permit the sharing of information, such as files, graphics, and video, and resources, such as printers and disks. One type of application software is called Client-Server. The Client computers send requests for information or requests to use resources to other computers, called Servers that control data and applications. Another type of application software is called peer-to-peer, in which the computers send messages and requests directly to one another without a server intermediary.

1.1.2 Network software: - consists of computer programs that establish protocols, or rules, for computers to talk to one another. These protocols are carried out by sending and receiving formatted instructions of data called packets. Protocols make logical connections between network applications, direct the movement of packets through the physical network, and minimize the possibility of collisions between packets sent at the same time.

1.1.3 Network hardware: - Network hardware is made up of the physical components that connect computers. Two important components are the transmission media that carry the computer's signals, typically on wires or fibre-optic cables, and the network adapter, which accesses the physical media that link computers, receives packets from network software, and transmits instructions and requests to other computers.

Transmitted information is in the form of binary digits, or bits (1s and 0s), which the computer's electronic circuitry can process.

II. RELATED WORK

A mobile agent is not bound to the system where it begins execution. It has the unique ability to transport itself from one system in a network to another. The ability to travel allows a mobile agent to move to a system that contains an object with which the agent wants to interact and then to take advantage of being in the same host or network as the object. A key characteristic of the mobile agent paradigm, as shown in Figure 3.1 is that any host in the network is allowed a high degree of flexibility to possess any mixture of know-how, resources, and processors. Its processing capabilities can also be combined with local resources.

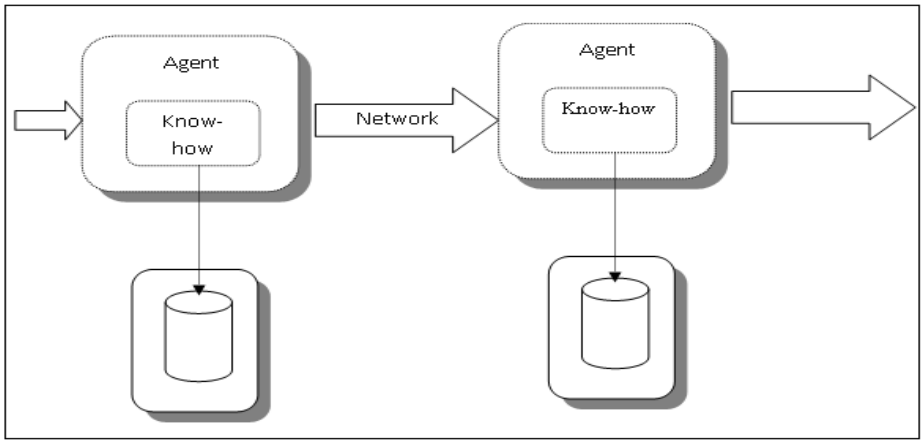

Figure 3 1: Mobile agent paradigm

In a nutshell, a MA is a software agent that can migrate, from machine to machine on a heterogeneous network. On each machine, the agent interacts with stationary service agents and other resources to accomplish its task. The agent performs its job wherever and whenever appropriate and is not restricted to be co-located with its client. Thus, there is an inherent sense of autonomy in the mobility and execution of the agent. We have explained about the MA paradigms and its comparison with other computational paradigms, are listed in the table below.

Paradigms/Attributes	**Mobile Agent**	**Remote Evaluation**	**Client-server**
Implementation	Hard	Easy	Very easy
Security	Very low	Low	Very high
Performance	high	Very high	Low
Elements a) Data b) Code **c)** Stack	 semimobile mobile mobile	 static mobile static	 mobile static static
Itinerary	Static/Dynamic	Both	Static
Mobility	Code to data	Code to data	Data to code
Platform	Dependent	Dependent	Independent
Programming code	Hard	Hard	Easy
Examples	Aglet	Aglet	CORBA,

Table: Comparison between various Distributed Computing Paradigms.

III. PROPOSED PMADE (PLATFORM FOR MOBILE AGENT DISTRIBUTION AND EXECUTION) MODEL

Peer to Peer (P2P) networks can be used to tie together the computing power of hosts in a network. In multi- agent systems, the basic interaction pattern between agents is P2P. For the diverse connectivity, a very popular concept P2P is used. It is save the bandwidth of network participants rather than conventional centralized resources where a relatively low number of servers provide the core value to a service or application. For connecting the nodes via largely *ad hoc* connections P2P networks are used. Such networks that are P2P are very useful for many purposes such as sharing the files containing audio, video, data or anything in digital format is very common, and real time data, such as telephony traffic, is also passed using P2P technology. P2P concept is used in PMADE framework. The migration of MAs is associated with different movement costs, viz., transmission time, round trip time, number of hops, etc. Costs are also generated by executing the agent's algorithms on the agent host (AH) [14]. Autonomous agents can observe their environment and reason and act on the basis of these observations. The integration of the functionality of P2P within the **PMADE** framework for large-scale multi-agent systems (MASs) is to simplify P2P application and service development and deployment by freeing the application developer of all low-level details including communication, security and scheduling. Large-scale MASs [13] needs to satisfy the multiple strict requirements such as reliability, security, interoperability, scalability, treatability, reusability and maintainability.

The important characteristic of PMADE is: -

(i) To provide a platform for large-scale agent systems

(ii) Support multiple code bases and operating systems

(iii) Interoperability with other agent platforms

(iv) Reduce network traffic

PMADE is a new platform. It provides a very simple, yet effective interface for the common user. It is client-Server based paradigm but difference is it uses the code to data approach for information gathering. Therefore it can save the bandwidth and resources as for needed. Each node of the network has a server called Agent Host (AH), which accepts and executes incoming agents and a client called Agent Submitter (AS), which is responsible for submitting the Mobile Agent on behalf of the user to the Agent Host.

Looking into the working of this model, how the Agent Host and Agent Submitter communicate to each other. When a user wants to perform a task or if the applicant requires gathering the information through a Mobile Agent, then the user submits the Mobile Agent designed to perform that task to the Agent Submitter. Agent Submitter checks the connection. If connection is ok with the specified Agent Host, where the user already holds an account; then the Agent Submitter submits the Mobile Agent to it and gets an acknowledgment and can then go offline. If connection is not established with the Agent Host then it pass appropriate message to user.

In our MA based data gathering system, there are four agents shown in Figure 4.4; on the behalf of these agents, we gathering the information from one host to other host. Out of four agents, two are MAs and other two are stationary intelligent agents. Detailed relationship among these agents is illustrated in Figure 4.3. These agents are:

- Pattern Database Generator Agent (PDGA),
- Pattern Database Processor Agent (PDPA),
- Pattern Based Search Agent (PBSA) and
- Pattern Based Processor Agent (PBPA).

1. Pattern Database Generator Agent (PDGA) and Pattern Based Search Agent (PBSA): is the MAs while *Pattern Database Processor Agent (PDPA) and Pattern Based Processor Agents (PBPA)* is stationary agent. These agents maintains static as well as dynamic itinerary. In the static itinerary all parameter defined before the launching the agent i.e. agent know everything about the host while in the dynamic itinerary agent is able to know the information about the host; that is whenever required agent can be updated at any node at any time in the itinerary. These agents maintain two containers (Result Container and State Container) one for transporting result data across the network and other for state variables and their intermediate values. In our system consist from four agents; first is PDGA. PDGA is dynamic and travels from one host to other host during the itinerary. It works on the behalf of the user. User or client gives the pattern to the agent and then agent starts its journey. According to the pattern, agent searches all the files in Public Store (PS) at a node in which that pattern exists in the itinerary. After receiving all the files, three document vectors are generated by the PDGA for each file. There are pattern vector, frequency vector and location vector. P*attern*

vector (PV), which contains all the different patterns or words in a file that are distinguished by the blanks in between the words. The second document vector is called the *frequency vector (FV)*, which contains the corresponding frequency of each pattern or word stored in the pattern vector that is how many times that pattern exists in a file. The third document vector is called the *location vector (LV)* that contains all the corresponding locations of the words stored in the pattern vector, i.e., all the locations at which that pattern or word exists in a file. It then creates Pattern Database (PD) at that node and stores File Name, PV, FV, and LV in it.

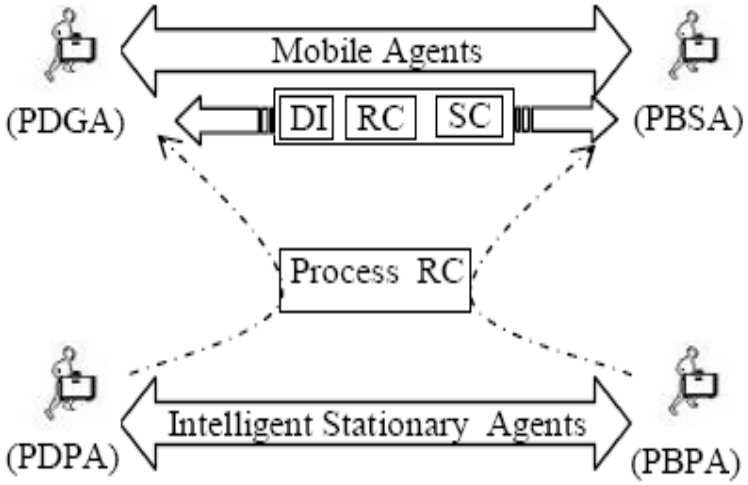

Figure: MA in data gathering

1. *Pattern Based Search Agent (PBSA):* - When a user or client gave an input (i.e. pattern or word) to the agent (PBSA); after getting the input agent starts its journey. We have discuses above three type of document have been generated by the PDGA. PBSA searches all the PV in PD on a node in which that pattern exists in the itinerary. This agent puts results (File Name, PV, FV and LV) on a node into the RC. Now PBSA is ready to migrate to next destination in the itinerary. When agent visits its final destination it submits Result Container to Agent Reply Manager (ARM) of the current visiting host and finally ARM sends result to the client (AS) of the agent.
2. *Pattern Database Processor Agent (PDPA):* -processes the Result Container prepared by the PDGA during the itinerary. It retrieves three vectors from the Result Container (*document vectors*¸ *pattern vector* and *frequency vector*) for every visited node in the itinerary. By processing these vectors it regenerates all the original files and stores at local client node.
3. *Pattern Based Processor Agent (PBPA):* - processes the Result Container prepared by the PBSA during the itinerary. It retrieves three vectors from the Result Container (*document vectors*¸ *pattern vector* and *frequency vector*) for every visited node in the itinerary. By processing these vectors it regenerates all the original files and stores at local client node.

IV. RESULT

In this Paper, we presented a multi-agent system for data gathering using mobile agent. We have identified four agents that are PDGA, PDPA, PBSA and PBPA for four different operations. Two are mobile agent (PDGA and PBSA) and two are stationary agent (PDPA and PBPA). Other system components (Policy, Agent-Agent, Interface, and Agent-Agent Communication layer) are provided in the framework to help the MAs to improve their performance as well as their autonomy and proactively for the navigation through the agent system infrastructure. For the evaluation, our focus was on pattern matching. We use the best algorithm to mobile agent (PMADE) so that it retrieves the data in fastest manner. We find better itinerary to the mobile agent so that it could take minimum time to retrieve the data. The main limitation of mobile agents is the security risk involved in using mobile agents. Firstly, a malicious mobile agent can damage a host and on the other hand a malicious host can tamper with the functioning of the mobile agent.

REFERENCES

[1] R. B. PATEL, K. GARG,"A Comparative Study of Mobile Agent and Client- Server Technologies in Real Application", ADVANCES IN DATA MANAGE- MENT 2005 Jayant Haritsa, T.M. Vijayaraman (Editors) © CSI 2005.
[2] Lange D.B. and Oshima M. "Seven good reasons for mobile agents" Comm.ACM, 42(3):88–89, March 1999.
[3] Robert S. Gray. Agent Tcl:"A transportable agent system" ,In James Mayfield and Tim Finin,editors, Proceedings of the CIKM Workshop on Intelligent Information Agents, Fourth International Conference on Information and Knowledge Manage- ment (CIKM 95), Baltimore, Maryland, December.
[4] Carlos Lucena José Alberto Sardinha Alessandro Garcia, Jaelson Castro Centro de Informática, Alexander Romanovsky, Paulo Alencar Donald Cowan, "Software Engineering for Large-Scale Multi-Agent Systems – SELMAS'2003".
[5] Gurpreet S. Bhamra, R. B.Patel, A.K.Verma,"An Agent Enriched Distributed Data Data Mining On Heterogeneous Networks" in IEEE ICACS 6-7 march 2009.
[6] Rajwinder Singh , Navdeep kaur , and A.K.Sarje " Designing Secure Itineraries Protocols for Mobile Agents" , IJCSNS International Journal of Computer Science and Network Security, VOL.6 No.12, December 2006.
[7] VuAnhPham & Ahmed Karmouch, " Mobile Software Agents: An Overview",IEEE Communication, July 1998, pp. 26-37.
[8] Deitel & Deitel, "Java: How To Program", Prentice Hall, 2000.
[9] R. B. Patel, V. K. Katiayar, Vishal Garg, "Mobile Agents in Wireless LAN and Cellular Data Networks, Journal of Computer Science, Science Publications New York, USA, 2(5): 410-418, 2006.
[10] Deitel & Deitel, "Java: How To Program", Prentice Hall, 2000.