A Survey on Mobile Agent

Pankaj Mehta¹, Vipin Chandra², Divya Bishit³, Bhuwaneshwari Pandey⁴
¹²³⁴MTech (Computer science & Engineering), Graphic Era Hill University, Bhimtal, Uttarakhand, INDIA

ABSTRACT

In the era of high speed internet there is a need of such communication medium which works smartly and intelligently and establishes an effective connection between different computer systems. As in the distributed architecture different systems are located in different places and work together to achieve a common goal. This creates a need of such communicating agent which establishes a persistent and result oriented connection between different machines. Mobile agent works so smartly in such medium as it can move host to host in network when and where it chooses. The intelligent and autonomous behaviour of mobile agent makes it a powerful tool for distributed architecture.

In this survey paper we are giving a brief description of Mobile agent, the existing system which inspires the advent of mobile agent, its life cycle, characteristics, advantages and applications.

Keyword: Agent, Software agent, Mobile agent- life cycle, characteristics, advantages and application

I. INTRODUCTION

I. Agent:

An Agent is one who agrees and is authorized to act on behalf of another. It is a program that performs some information gathering or processing task in the background. Typically, an agent is given a very small and well-defined task [1].

The term agent comes from greek ‘agein’, which means to drive or to lead. Today the term agent denotes something that produces or is capable of producing an effect. It can be a chemically, physically or biologically active principle [2]. In computer science, there is a school of thought that believes that the human mind essentially consists of thousands or millions of agents all working in parallel. To produce real artificial intelligence, this school holds, we should build computer systems that also contain many agents and systems for arbitrating among the agents' competing results [1].

II. Software Agent:

A software agent is a persistent, goal-oriented computer program that reacts to its environment and runs without continuous direct supervision to perform some function for an end user or another program. Some, but not all, software agents have UIs (user interfaces) [3].

Software agents represent an evolutionary step beyond conventional computer programs. Software agents can activate and run themselves, not requiring input from or interaction with a human user. Software agents can also initiate, oversee, and terminate other programs or agents including applications and online intelligent agents [3]. There are a number of different software agents, including [4]:

Buyer Agents or Shopping Bots: These agents revolve around retrieving network information related to good and services.

User or Personal Agents: These agents perform a variety of tasks such as filling out forms, acting as opponents in games, assembling customized reports and checking email.

Monitoring and Surveillance Agents: These agents observe and report on equipment.

Data-Mining Agents: These agents find trends and patterns in many different sources and allow users to sort through the data to find the information they are seeking.

III. Classification of Software Agent:

[5]According to Gilbert et al, software agents can be classified in terms of a space defined by the three dimensions of intelligence, agency and mobility.

The first dimension, intelligence, is rooted in artificial intelligence research and dates back to the fifties, where the term agent was coined by Selfridge for a ‘soft robot’ living and acting within a computer. The goal within this approach was an agent that could apply techniques of symbolic artificial intelligence in order to fulfill a given task or to recover when it was stuck. These intelligent agents can be classified according to their capabilities to express preferences, beliefs and emotions and according to their ability to fulfill a task by reasoning, planning and learning techniques.

The second dimension, agency, is the degree of autonomy and authority vested in the agent and can be measured at least quantitatively by the nature of the interaction between the agent and other entities of the system. At a minimum an agent must run asynchronously.
The degree of agency is enhanced if an agent represents a user in some way. A more advanced agent can interact with data, applications, services or other agents. According to their capabilities, agents are called autonomous, collaborative, cooperative or negotiating agents.

The third dimension of software agent research, mobility, has emerged in the nineties and is motivated by the rise and rapid growth of a networked computing environment and the need for techniques to exploit this huge resource. The goal within this dimension of software agent research is remote action and mobility of data and computation. These mobile agents can be classified according to the scope of their mobility into desktop (or static) agents, intra-net agents, Internet agents or network agents.

IV. Mobile Agent:

A mobile agent is a running program that can move from host to host in network when and where it chooses. Mobile agents are one form of mobile code. In its simplest form, the concept of mobile code involves the dynamic installation of code on a remote host. In Web applications, applets and servlets are a common form of mobile code. The mobile code concept also appears in “remote evaluation” systems, which extend the notion of remote procedure call to transport the procedure to the server along with the call. A mobile agent runs in one location, moves (with its state) to another host, and continues at that host. Mobile code and mobile objects are normally moved by an external entity; mobile agents usually have migration autonomy[9].

Mobile agents have many potential advantages. By moving the computation to another host it is often possible to co-locate the computation with an important database, allowing high-throughput low-latency access to that database. Compared to more traditional client-server approaches, mobile agents can avoid transmitting as much data across the network. The mobile agent can move, with partial results, from one server to another until it has accomplished its task, and then return to the originating host, which may freely be disconnected during the agent’s travels[8].

V. Traditional Approaches:

<table>
<thead>
<tr>
<th>Computing Paradigm</th>
<th>Mobility Mechanism</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client-server/ Message Passing</td>
<td>Transport of data</td>
</tr>
<tr>
<td>Remote evaluation/ Code on demand</td>
<td>Transport of code + data</td>
</tr>
<tr>
<td>Mobile object/ Weak migration</td>
<td>Migration of code + data</td>
</tr>
<tr>
<td>Mobile Agent/ Strong migration</td>
<td>Migration of code + data + state</td>
</tr>
</tbody>
</table>

Table 1: Distributed Computing Paradigms [6]

[10] Before the advent of Mobile agents, the communication between the client and server is achieved by different approaches such as message passing, Remote Procedure Call (RPC) and Remote Evaluation (REV). In RPC method, the procedure resides in the server and client sends a data to the procedure that will be executed there, finally the result is back to the client. In REV approach which is different from the RPC, the procedure itself will be sent and the desired result is returned to the client.

A Client/Server model is that in which server provides services to the client. When a client needs a service, it usually sends a request message to the server as shown in figure 2. In case, the server does not have resources to satisfy the request made by the client, the client sends request to other server having the needed resource to satisfy the client that usually increase the inefficient use of network bandwidth. This also increases the network traffic and causes delays due to the involvement of more servers.

Mobile agent provides solution for this mobile device because they do not depend on the server operation. Once the mobile agent has migrated, the connection between the client and server is disconnected, later when mobile agent finishes its job at the server, then it will reconnect to the client or host with the result shown in figure 3. This clearly saves the network bandwidth.
especially in the wireless environment where disconnection is frequent and bandwidth play a major role.

**Figure 3: Mobile Agent Model [10]**

**VI. Life cycle of Mobile Agent:**

The life cycle of a mobile agent includes following phases [2]:

- **Creation:** Analogous to the constructor of an object. A handler for this event should initialize state and prepare the agent for further instructions.
- **Disposal:** Analogous to the destroyer of an object. A handler for this even should free whatever resources the agent is using.
- **Dispatch:** Signals the agent to prepare for departure to a new location. This event can be generated explicitly by the agent itself upon requesting to migrate, or it can be triggered by another agent that has asked this agent to move.
- **Arrival:** Signals the agent that it has successfully arrived at its new location and that it should commence performing its duties.
- **Communication:** Notifies the agent to handle messages incoming from other agents and is the primary means of inter-agent correspondence.

**VII. Characteristics Of Mobile Agent:**

Mobile agents have different kinds of characteristics [11, 12]. They should be.

**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 behaviour based on this information.

**Intelligent:** means Mobile Agent should be too smart in order to act efficiently.

**VIII. Advantages of Mobile Agent:**

- **They reduce the network load [13]:**
  
  Distributed systems often rely on communication protocols involving multiple interactions to accomplish a given task. The result is a lot of network traffic. Mobile agents allow users to package a conversation and dispatch it to a destination host where interactions take place locally. Mobile agents are also useful when reducing the flow of raw data in the network.

- **They overcome network latency [13]:**
  
  Critical real-time systems, such as robots in manufacturing processes, need to respond in real time to changes in their environments. Controlling such systems through a factory network of substantial size involves significant latencies. For critical real-time systems, such latencies are not acceptable. Mobile agents offer a solution, because they can be dispatched from a central controller to act locally and execute the controller’s directions directly.

- **They execute asynchronously and autonomously [13]:**
  
  Mobile devices often rely on expensive or fragile network connections. Tasks requiring a continuously open connection between a mobile device and a fixed network are probably not economically or technically feasible. To solve this problem, tasks can be embedded into mobile agents, which can then be dispatched into the network. After being dispatched, the agents become independent of the process that created them and can operate asynchronously and autonomously. The mobile device can reconnect at a later time to collect the agent.

- **They adapt dynamically [13]:**
  
  Mobile agents can sense their execution environment and react autonomously to changes. Multiple mobile agents have the unique ability of distributing themselves among the hosts in the network to maintain the optimal configuration for solving a particular problem.

- **They are naturally heterogeneous [13]:**
  
  Network computing is fundamentally heterogeneous, often from both hardware and software perspectives. Because mobile agents are generally computer and transport layer-independent (dependent on only their execution environments), they provide optimal conditions for seamless system integration.

- **They are robust and fault-tolerant [13]:**
  
  Mobile agents’ ability to react dynamically to unfavorable situations and events makes it easier to build robust and fault tolerant distributed systems. If a host is being shut down, all agents executing on that machine are warned and given time to dispatch and continue their operation on another host in the network.
**Efficiency savings [2]:**

CPU consumption is limited, because a mobile agent execute only on one node at a time. Other nodes do not run an agent until needed.

**Space savings [2]:**

Resource consumption is limited, because a mobile agent resides only on one node at a time. In contrast, static multiple servers require duplication of functionality at every location. Mobile agents carry the functionality with them, so it does not have to be duplicated.

**IX. Mobile Agent Application [7]:**

**Remote information retrieval:**

This is one of the most traditional applications of mobile agents. If all information were stored in relational databases, a client could send a message containing SQL commands to database servers. However, given that most of the world’s data is in fact maintained in free text files on different computers, remote searching and filtering require the ability to open, read, and filter files. Since mobile agents can perform most of their tasks locally at the destination, Client can send its agents to database servers so that they locally perform a sequence of query or update tasks on the servers. Communications between the client and server can be minimized. Since agents contain program codes for filtering information that is of interest to their users from databases, they only need to carry wanted information back to the client to reduce communication traffic. Furthermore, agents can migrate among multiple database servers to retrieve and gather the interesting data from the servers.

**Network management:**

Mobile agent technology provides a solution to the flexible management of network systems. Mobile agents can locally observe and control equipment at each node by migrating among nodes. Mobile agent-based network management has several advantages in comparison with traditional approaches, such as the client/server one.

- As code is very often smaller than the data it processes, the transmission of mobile agents to sources of data creates less traffic than transferring the data itself. Deploying a mobile agent close to the network nodes that we want to monitor and control prevents delays caused by network congestion.
- Since a mobile agent is locally executed on the node it is visiting, it can easily access the functions of devices on this node.
- The dynamic deployment and configuration of new or existing functionalities into a network system are extremely important tasks, especially as they potentially allow outdated systems to be updated in an efficient manner.
- Network management systems must often handle networks that may have various malfunctions and disconnections and whose exact topology may not be known. Since mobile agents are autonomous entities, they may be able to detect proper destinations or routings on such networks.

**Cloud computing:**

Load-balancing is a legacy application of process migration and mobile agent technologies. In a distributed system, e.g., a grid or cloud computing system, computers tend to be numerous and their computational loads are different. Computers may also be dynamically added to or removed from the system. Tasks should be dynamically deployed at computers which loads light rather than those lose with heavy loads. Since mobile agents can migrate to other computers, tasks that are implemented as mobile agents can be relocated at suitable computers whose processors can execute the tasks. This is practical in implementing massively multi agent systems that must operate a huge number of agents, which tend to be dynamically created or which terminate, on a distributed system that consists of heterogeneous computers.

**Mobile computing:**

Mobile agents use the capabilities and resources of remote servers to process their tasks. When a user wants to do tasks beyond the capabilities of his or her computers, the agents that perform the tasks can migrate to and be executed at a remote server. Mobile agents can also mask temporal disconnections in networks. Mobile computers are not always connected to networks, because their wired networks are disconnected before they are moved to other locations or wireless networks become unstable or non-available due to deteriorating radio conditions or are not uncovered by the area at all. A stable connection is only requested at the beginning to send the agent, and to take the agent back at the end of the task, but this is not requested during the execution of the whole application execution.

**II. CONCLUSION**

In this paper we have reported about the advent of mobile agent from the existing techniques and there is a brief discussion of the characteristics of mobile agent which make it most popular in present scenario. The main conclusion from the study is that mobile agent technique is the most powerful technique for communication in distributed architecture and it increases the performance of a network. It provides important advantages in context of network management, asynchronous autonomous interaction, interaction with real-time entities, local processing of data, support for heterogeneous environment and having robust and fault-tolerant behaviour. Due to the future aspect of mobile agent technology a lot of work is still required so that the applications developed for mobile agent can perform better and will provide more security.

**REFERENCES**

[2] A report on Mobile Agent By: Nriyesh Kumar, Department of computer engg, N I T
Kurukshetra