Abstract—Mobile agent technology is often described as the future of distributed computing. It promises to offer a unified and scalable framework for such applications in widely distributed heterogeneous open networks as e-commerce, information retrieval, process coordination, mobile computing and network management. This paper describes the Strategies adopted by a mobile agent to complete the task assigned by the user. Whole working of mobile agent is depend upon the Network, because each individual task has some resource requirement and duration. To get an optimal solution for a user define task an agent require a network graph or resource graph (information) which provide information like bandwidth, connection state, network load, suitable services and resource available etc.

Index Terms-Information, Network graph, Resource graph, Task.

I. INTRODUCTION

A mobile agent is a program that represents a user in a computer network and can migrate autonomously from node to node, to perform some computation on behalf of the user [1]. A Mobile agent is a kind of software program that can migrate from one host to another in a heterogeneous network. It can communicate and interact with other agents and the distributed resource system on heterogeneous networks. A mobile agent can also decide when to migrate, and which nodes to access. After migrating to the desired host, the mobile agent resumes the execution of previously broken off or awaiting tasks. On completing the task, the mobile agent returns the result to the client. Therefore, the client need not be constantly connected to the server. This will not only save lots of unnecessary transmission load, but also helps in the application of mobile calculation. The mobile agent can single-handedly execute all tasks assigned by the user. It can meet and interact with other agents when necessary while still executing its task. Therefore, a mobile agent can be viewed as an independent program. A User need only assign tasks to a mobile agent. The agent then migrates to the remote network management servers to assign tasks to the servers. Once the tasks are completed, the mobile agent brings the result back to the user for analyses and processing. During the self-initiated migration, the agent carries its code and some kind of execution state with it. What comprises the execution state depends on the underlying programming language like Java.

Purpose of this paper is to describe the importance of network graph or resource graph (information), information play an extra ingredient when we are dealing with mobile agent. We know that a particular task depend upon the set of resource which are distributed on the network node, so we must acquire the information of the resources, services and the particular agent server which are providing that resource or services, once we get the information of the resources, we need the information of the network characteristics, like bandwidth, connection state, etc which is called network graph. The network graph and resource graph both are input for optimizations of user optimal solution for given task and selecting the suitable migration strategy.

One of the most important issue while gathering the information is mobile agent communication process, we need an efficient communication process by which mobile agent on network node communicate and cooperate with each other to form a network and resource graph. In second section 2 we will put light on mobile agent task, migration strategies, Communication strategies of mobile agent, section 3 describe the importance of network graph and resource, information. Section 4 provides a summary of these strategies.

II. STRATEGIES OF MOBILE AGENT

A. Task

A task can be defined as a set of goal to be attained in given conditions. Each individual task has some resource requirements, all of which have to be satisfied before the task can be started. A resource is an abstraction and can correspond to a processor, device, file, data structure, etc. A task also has a given duration for which the task runs. The mobile agent acts on behalf of a user. This means the user has to hand over a task to one or more mobile agents. A mobile agent has to communicate and cooperate with other agents to use the services available in the network, provided by the agent.

B. Task That Can Be Assigned To Mobile Agents [13]

Many commercial transactions require access to resources in real time. The ability of a mobile agent to personify their creator’s intentions and to act and negotiate on behalf of them makes it well suited for electronic commerce. An agent can act as a personal assistant to the user and perform tasks for user on a remote host regardless of whether or not user is connected to the network. For instance, to schedule a meeting, a user can dispatch a mobile agent onto the network to interact with agents belonging to other users. The agent can negotiate with other agents the convenient time for all of the users and can schedule a meeting.

Mobile agent technology is an attractive solution to brokering, particularly in the context of untrustworthy collaborators. In such a situation, the interested parties can let the agents meet and negotiate at a mutually agreed impartial secure host and form alliances. Mobile agent technology provides efficient information retrieval. When dealing with large amounts of data, rather than moving all
the data to search engine to create search indexes, user can simply dispatch mobile agents to remote sources to create those indexes locally and to ship them back later to its origin.

Mobile agents provide an effective and flexible solution to the management of advanced telecommunication services by providing dynamic network reconfiguration and user customization. Mobile agents because of their features such as autonomy, mobility etc. provide autonomy to the workflow and data state of the mobile agent [2]. The impact of this migration is handled transparently by the underlying system scheme for migration [3].

Weak migration involves only the migration of the code and data state of the mobile agent [2]. The impact of this scheme for the programmer is an added level of complexity in terms of an entry point mechanism as well as additional logic to keep track of the location of the mobile agent .The majority of Mobile Agent Systems are implemented with this scheme for migration [3].

Migration process describe above is define as a programmer point of view migration process. Some author also describe migration process which depend upon agent point of view where agent can choose different migration strategies (pull code, push code, etc.). Push code send the complete code over the network before migration. Pull code Download the code from destination platform on demand After Migration [4].

E. Agent Communication strategies

In order to perform a given task effectively, a mobile agent should interacts and cooperates with others. Mobile Agents therefore need to be able to communicate between themselves. It has been realized that communication between mobile agents is not as simple as the common communication method in a computer network. The problem
is how to deliver a message to a migrating-recipient with unknown location.

To communicate with a remote mobile agent, we must find the location of the agent and route the message to it. A naming scheme is needed to identify agents in a unique fashion. The name should not change whenever the agent migrates to other hosts and it is up to the tracking algorithm to map the name to the agent’s current address. The routing process can be done either in parallel with agents tracking [5] or in a second phase after the address has been got [6]. The usual way to name an agent [5, 6] is to append the address of an agent’s origin host (i.e. agent home) with its title (a free form string used to refer to this agent). Thus it is impossible for agents born at different agent platforms to have the same name. For agents created at the same host, the origin host is responsible to manage the name space to ensure that each agent has a unique title.

There are three basic schemes for locating agents, namely searching, logging and registration [7]. In the searching approach, we either send an agent to visit every host that the target agent might reside in or broadcast locating messages to these hosts [8]. The overhead is unaffordable when the network is large. The logging method locates the mobile agent by following the trail information indicating its next destination; left in every host the agent has ever visited [5]. If the trail information is lost or if one of the hosts is down, the target agent would no longer be found. With the registration scheme, an agent needs to update its location in a predefined directory server (e.g., its home host) that allows agent to be registered, deregistered or located. The directory server can be either a central node, which may become the bottleneck of the system performance and/or a single point of failure, or the agent’s home host, which follows the idea of Mobile IP [9].

There are many mechanisms implemented by researchers to enable mobile agents to communicate. These mechanisms range from basic object oriented message passing to events that are generated to facilitate the communication. The mechanisms discussed here are message passing, streams, events, proxies, blackboard mechanism and CORBA. The mechanism of message passing is based on the concepts of object interaction by means of passing messages. This means of communication depends on the communicating parties having a referencing to each other in order to be able to invoke each others methods. Local communications is achieved by means of this mechanism and is applied in systems such as for example ARA. Streams are a direct socket connection established between mobile agents and it is maintained while data is exchanged via a stream of bytes [10] This is a communication mechanism employed by traditional distributed systems and could pose the danger of breaking the mobile agent paradigm. It may however be necessary, as a result of its computation, for the mobile agent to transfer a large amount of data to its owner and this mechanism can be utilized to achieve that. Streams therefore need to be used with care in a global communication scenario.

The Common Object Request Broker Architecture is a widely used communication mechanism based on the IIOP and is utilized to achieve transparent message passing in a distributed system. The MASIF standard places the responsibility of communication on this standard. In systems such as Aglets a proxy object is created that represents the mobile agent. The proxy object is responsible for receiving the Aglet’s communication and to pass the data to it [11]. The Tracy [4] system makes use of a blackboard mechanism whereby mobile agents leave messages for each other.

III. NETWORK GRAPH AND RESOURCE GRAPH (AGENT ITINERARY)

Agent itinerary can be considered as a network graph and resource graph. An agent’s itinerary describes the tasks of the agent and the locations where those tasks are to be performed [12]. The itinerary model defines the mobile agent travel planning. The travel planning can be determined either statically or dynamically. In static travel planning a mobile agent can carry along a “travel plan”, provided by its programmer/owner, consisting of the places to be visited one after another while attending to its chores it can be calculated either before the agent is dispatched or while the agent is migrating and use statistical network information to plan agents’ itineraries In some papers such as [15, 16], this static agent planning has been recognized as a Traveling Salesman Problem (TSP) [17] or TSP variations such as Vehicle Routing Problem (VRP). [18] Traveling Agent Problem (TAP), which is analogous to the TSP, in deciding the sequence of nodes to visit, in order to minimize the total execution time until the desired information is found. Although static travel schema cannot adapt to the network change, it is able to save both computation and power since the travel planning only needs to be calculated once. Computation-efficiency, power-efficiency, and flexibility are three parameters that cannot be satisfied at the same time. Dynamic travel planning is designated to provide network sensitivity with the ability to change the agent’s itinerary appropriately it is more flexible, and can adapt to environmental changing in real time [19].

Information is input of agent itinerary. Two approach can be used two manage the information one is central server and second is distributed server, with central server approach a single agent server hold global data base, with central server approach some problems are associated, firstly it contain huge amount of data, secondly continuous connection of agents to the central server, thirdly failure of central server cause breakdown of information network graph. In Distributed server approach each mobile agent server on network provides information about available services and the network characteristics. Information gathering in distributed approach can be divided in three ways (a)In
neighbor information each mobile agent server store information of the adjacent mobile agent server only (b) community information form a group of mobile agent server and store the information of local region (c) society information provide information of whole network.

For travel planning a mobile agent can use both static itinerary and dynamic itinerary [19]. To obtain static itinerary system must be composed of domain manager and network sensing module . The network-sensing facility reports to the agent on the current latency of the path (itinerary), which the agent presents. Static itinerary contain full information of local network and partial information of remote network. If assigned task resource falls in local network then agent use the static itinerary otherwise it takes static itinerary as input for calculating dynamic itinerary.

IV. SUMMARY

In this paper we analysis the Strategies which mobile agent adopts to handle a task. After analyzing the whole process (1) The tasks execution planning which describes the mapping of tasks to be performed on different locations. (2) The migration action (weak or strong). (3) The mobility paths which describes the different network nodes. (4) The interactions which describe the communication acts. (5) The itinerary which describes the list of locations to visit or to reside and the set of tasks to be performed in order to complete a specific mission (6) The travel schema which describes the travel planning. We conclude that that information plays a booster in all steps of mobile agent processing. Whether it is to prepare a resource graph for routing or network graph for migration process or communication.

REFERENCES