The architecture and implementation of M-business based on mobile agents using Aglets

Ahmed Aloui, Okba Kazar, Khamsa Chouchane, Zerdoumi oussama

1,2,4 Computer science department, university of Biskra, Algeria
3 Computer science department, university of Batna, Algeria

1 ahmed0725@gmail.com
2 kazarokba@yahoo.fr
3 khamsa.info@yahoo.fr
4 oussamazerdom@gmail.com

Abstract: The rapid evolution of mobile technology had a great impact on people’s lifestyle in many domains such as business, thus introducing a new paradigm called Mobile business (m-business) which appeared as the promising approach to drive the vague following one of electronic business (e-business). Most of the e-business applications use the traditional model client/server in which a commercial operation requires generally a link of stable communication being established between the customer and the server, and the traditional approach client/server constitutes an obstacle to the development of application of m-business. In this paper we propose an architecture for mobile business using mobile agents. The proposed architecture introduces several advantages: first, it allows the consumers to manage their commercial business driven by types of mobile devices (phones, PDAs, etc.) at anytime and anywhere. Second, it minimizes the waiting time of the customer, and the quantity of transferring information. Third, this architecture addresses the problem of time limited and expensive connection for mobile users.

Keywords: Multi-agent System, Mobile agent, Aglets, M-business, J2ME.

1. Introduction
Advances in mobile technology have created a wide range of possibilities for the business users, whose want to have the opportunity to do their business anytime and anywhere via their mobile devices. The e-business applications [9] require that the users must connect to Web sites planned by their personal computers or the public terminals. Besides, the users often need to visit numerous sites and are always involved in a boring process. Most of the applications of e-business uses the traditional client/server model in which a commercial operation requires generally a link of stable communication being established between the customer and the server, and the traditional client/server approach suffers numerous problems such as: the commercial business (affairs) requires a permanent connection being established between the customer and the server, the high network traffic, the consumer must deal with the limitation of time and space and synchronous connection. In order to solve these problems we used mobile agents [4] which represent a convenient and effective approach for mobile business applications. The main objective of our study is to propose mobile agents architecture for M-business environment, in order to take the advantage of the mobility of agents.

In this architecture, a consumer can connect his mobile device (such as PDA, mobile phone, etc) to the application server via a wireless connection and sent then a request of creation of a mobile agent which begins a specific business task on his behalf. Application server provides services such as the creation of mobile agents based on the demands of consumers. After having been created, the mobile agents autonomously travel to several servers-based agents on the Internet where a consumer wants to compare several international markets. The consumer needs only to reconnect later to call the agent for the results, so minimizing the use of resources.

The rest of this article is organized as follows: the section 2 presents an overview of related works. The section 3 discusses our proposed approach. Section 4 presents our experimental evaluation. The section 5 discusses communication by message between inter-aglets. Finally, the future works and conclusion are given in Section 5.
2. Related Works

In the literature there are few works that use mobile agents in the business: first we have the work of Zhiyong et al [12] who propose a structure based on the intelligent mobile agent that allows buyers and sellers to execute the business (affairs) by means of mobile devices. This architecture is based on multi-agent systems because of their characteristics, facilitating the consideration of the very high dynamics of the environment in which to run the business using mobile devices. In this approach, the agent begins its migration of it’s the server mediator with an itinerary list acquired from the mediator. If we have N servers, the agent will be visiting these N servers in sequence at a time limited by the mediator server. If the agent reaches its life, he returns to its host where it was created and then finish the migration process, in this case, the agent may be not visited at all servers. This can lead to a loss of opportunity to negotiate with other servers. The disadvantage of this architecture is: increasing the migration time and negotiation, and if the migration agent fails, or the node on which it is running fails, the migration process and negotiation will be destroyed. Second, we present the architecture of J-Phone with mobile agent for auctions [1], in this architecture the server of mediator (AgentGatway) does not contain the itinerary list for auction sites, in this approach the server mediator created the mobile agents according to the number of the auction sites and sends to each auction site a mobile agent. The main advantage of this approach (J-Phone), it is that the mediator server sends a mobile agent for each auction site to minimize the time of migration and avoid the failure of the first approach. The last architecture is IMAGO [10], this architecture represents a distributed environment which allows the consumers to send mobile agents of their portable devices to visit the on-line stores for the search, by comparing, evaluating, the purchase and the payment of the goods. The disadvantage of this approach is that the mobile agent to cloning in remote servers, and this method is not recommended at all because it violates the integrity of servers, is that consumers send mobile agent from their handheld devices, potentially causing long-term connection with the Internet. We believe that it is important to take account of the limitations of mobile devices, such as bandwidth, battery and calculating capacity limited and expensive connection time.

3. Our approach

Most of e-business applications use the traditional client/server model in which a commercial operation requires generally a permanent and stable communication between the customer and the server, thus the traditional client/server approach constitute an obstacle to the development of m-business applications. We used mobile agents to take advantage of this paradigm such as mobility; autonomy etc, in order to address problems mentioned above. The main objective of our study is to propose of architecture based on mobile agents for M-business environment.

In this work, we propose a mobile agents approach designed for M-business. The Figure 1 shows mobile business environment, a consumer can connect his mobile device, such as PDA or a mobile phone, to the application server via a wireless connection and sent then a demand of creation of a mobile agent which begins a specific business task on his behalf.

An application server provides services such as the creation of mobile agents according to the demands of consumers. After being created, the mobile agents travel autonomously to several servers-based agents on the internet when the consumer wishes to proceed to a comparison on several world markets.

![Figure 1. A mobile business environment.](image)

The consumers only have to use twice the connection to low bandwidth, once to introduce a mobile agent and at the next time to collect the results when the task is complete.

3.1. The architecture of the system

The general architecture of our system, illustrated in the Figure 2, is structured around three main sides in an interaction:
Application Server is the heart of the system; it acts as a mediator between the mobile device and supplier sites. In the Application Server, a Server Home responds to any requests (queries) from the client agent.

3.1.1 the user side

It provides a user interface which allows the direct interaction between the system and the user, and act as a mediator in which users could benefit with services provided by the system.

This interface allows the user to send demands (requests), and includes:

- Client Agent is a stationary agent that runs on a mobile device of the user and provides a graphical interface (client interface) to interact directly with consumers and examine its personal preferences (from the mobile device). The client agent represents the interests of consumers and allows consumers to have a choice to produce and distribute a mobile buying or sale agent.

Attributes can be configured by a user via their mobile device, according to the following characteristics: the type of agent that specifies a buying agent or selling agent, an agent server (application server address), user identification which can be a mobile phone number or e-mail address, information on predetermined product, quantity, and the price that the agent can provide.

3.1.2 Application server

Supplies services perform the creation of mobile agents according to the demands of the consumers. After having been created, the mobile agents autonomously travel to several servers-based agents on the Internet where a consumer wants to compare several international markets. It includes:

- The home server is used to process the application of the user side that includes the management of Master Aglets. In the home server, a servlet responds to any requests (queries) of the client agent.
- The mobile agent Server understands Master Aglets to create or send mobile agents (slaves) to suppliers' sites. Master Aglets cooperates with the
repertory server to receive suppliers' list which possesses information on the product asked by customer.

- The **repertory server** has a directory of database which stores the addresses of all the suppliers, and can receive information on available products, or find the other agents supplying the necessary services to realize its purpose. On this server we have two agents:
  - **Agent of the repertory** is a stationary agent which provides a directory of database that stores the addresses of all the suppliers. According to the demands (requests), the agent directory selects a suppliers' list which possesses information on the wanted product, by means of an agent database.
  - **Agent database of repertory** is responsible for access and retrieval of data from the database.

3.1.3 Suppliers' sites

receive and interacts with the mobile agents of searches and to represent markets, it includes:

- **Agent of the supplier**: is a stationary agent, it receives the mobile agents (slaves) who are sent by Master Aglets, create a database agent and bring consumer requests to it.
- **Agent of the database of supplier**: will generate a query, the results corresponding to the request of the consumer are extracted from the database and returned as a result to the **supplier Agent**.

3.2 Types of agents

Our system consists of multiple agents cooperating to meet requests for purchases or sale of client proposals:

- A **client agent** is a stationary agent that runs on a mobile device of the user and provides a graphical interface to allow the user to configure a mobile agent (from the mobile device).
- The **mobile agent** will be sent to suppliers' sites where buying agents and selling agents interact and negotiated between them to reach an agreement, provided that they can communicate in a common language.
- **An aglet** is an autonomous software agent based on Java, used on this system and it works on **application server** and the sites of suppliers.
- **Agent of the repertory** is a stationary agent which provides a directory of database which stores the addresses of all the suppliers.
- **Agent of the supplier** is a stationary agent; it receives the mobile agents of research that are sent by Master Aglets [2].

3.3 Approach operation

We use mobile agents because they are able to search information in a more intelligent way. In addition, they are able to communicate and cooperate with each other, accelerating and facilitating research. We propose the following functional processes:

1. As a first step, the user introduces and manages the mobile agent and configures his preferences via the client agent. The client agent is sent then the request of the user to the application server, the user can choose interface services and sending a request to the application server via an HTTP connection.

2. In the second step Servlet accepts the request and communicates with the Master Aglets. According to the request of servlet and using the repertory agent, **Master Aglets** creates then the mobile agents of researches (aglets slaves) and sends mobile agents at the sites of suppliers who have information about the product, to undertake the task of the user.

3. When the mobile agents arrive at site suppliers (each agent for each supplier site), every mobile agent sends a call for proposals (CFP) to the agent supplier, then the agent suppliers to offer the result to the mobile agent using supplier database agent.

4. When the mobile agent collects all the results, it returns the results to the **mobile agent server (Master Aglets). Mobile agent server** and then pass the result to Servlet or send SMS messages to the user.

This process is asynchronous after which the user can disconnect from the network at will. However, it will always receive an announcement with SMS or with the e-mail of the application server via the mobile agent server or home server. The application server supplies support required for the creation of mobile agents, messaging among agents, agent's migration,
collaboration, protection, destruction and control of mobile agents.

In a mobile business application, agents do not only work and they must communicate with them to cooperate and to generate an aggregation of the aggregate data for further analysis. The existing mobile agent systems adopt several models/protocols of traditional distributed systems.

However, the M-business system adopts a different strategy to deal with this issue. The idea is to deploy intelligent mobile messengers for inter-agent communication. The messengers are thin agents devoted to deliver messages.

In our study we used the Aglets [3] in the server parts and suppliers sites; to benefits with the mobility of the agents provided by this platform, where the mobile agents offer many advantages to improving the performance of several distributed applications.

3.4 The migration of agents

The migration of agents is also reactive because the destination of our researches agents is not determined by the agent itself, but it is dictated by the Master Aglets. A mobile agent on remote sites and at any time can send messages to provide data to designated receivers. For example, suppose that a mobile agent has finished its work to a remote server, it can migrate to the server of origin (home), or send a message to deliver the result to a Servlet.

3.5 Negotiation

We use AUML (Agent-based Unified Modeling Language) [11] which is a variant of UML to model interactions between agents. AUML is a language for modeling multi-agent systems. On the one hand, the agents are active; they are capable of taking initiatives and can control the communication between them. On the other hand, the agents cooperate and coordinate their work to affect a common purpose. With regard to objects, the agents have autonomous activities and purposes. It is this difference which causes failure of UML, to model agents and systems multi-agents and it is for this reason that we used AUML to model our system. A contract Net protocol has been one of the first approaches used in multi-agent systems to solve the problem of allocation of tasks; the agents coordinate their activities through the establishment of contracts in order to achieve specific goals. This protocol allows a Buying Agent (the initiator) to send a call for proposals (CFP) to a set of agents of sales (responders) and then the agent of sale to estimate the propositions of the buying agent and then to refuse or to accept the most favorite.

3.6 Interaction INTER-AGLETS

We present below the protocols of interaction between agents, we model different interactions by sequence AUML diagrams. In the Figure 3: AR: reportory agent, AS: supplier agent, AMR: Mobile agent of search.

Figure 3. Sequence diagram "Launch Agent"

4. Validation

To show the validity, reliability, and extensibility of our architecture, it has interest to do a case study. Where we apply our approach on a typical example for the commercial business conducted using mobile devices at anytime and anywhere. We assume the following scenario: Hatem lives in city Bouzourane (Batna,
Algeria); he wants to buy a mobile phone type: **Nokia N82** from their PDAs (at anytime and anywhere). He prepares the arrangements of purchase, he has to determine the value interval (maximum and minimum), the minimum price is the price below which the customer thinks that the product cannot be good quality, and the maximum price is the price above of which, we are not any more inclined to pay the product, and he must determine the desired location and the desired duration. And then he launches a request from their PDA. Our example represents an m-business (**B2C**) application whose goal is to find a mobile phone of good price in the desired length, and vendor of products desired close to a known location and in a given city. We note that this process is complex and requires more time and attention.

Our goal is to automate this process by our system using the technology of mobile agents. Screenshots of a Client agent are shown in Figure 4. We used an Application Server and four servers represented suppliers of products (two suppliers for laptops and two suppliers for mobile phones) in the local network. The application server uses the Tomcat server to compile and execute Servlets [6] and JSP, and uses the Aglets platform to create and initialize agents. Other servers play the role of a market with agents on the internet. We installed J2ME MIDlet [7] in a simulator of mobile phone, provided GUI for the client on the web, a mobile phone emulator is a tool provided by Sun J2ME wireless toolkit 2.5.2 [5].

Our system is constituted by several agents who cooperate to satisfy purchase orders or propositions of sale of the customers, we distinguish from it: the client agent, repertory agents, Master aglets, the mobile agent of research and the agents of suppliers.

The agents of our system are implemented using the language JAVA and the platform for development of the agents Aglets [4], Aglets supports the development of the agents with the possibility of transporting them from a system to the other one. The client agent, the repertory agents, the agents of suppliers and Master Aglets are all stationary agents whereas the agents of research are mobile agents crossing the network to collect information for their customers.

Figure 4. agent Client (Screenshots mobile phone).
When the servlet receives the results, it sends an email or an SMS to the user to view the results. As we saw, the mobile consumers only need to use a connection to low bandwidth twice, once to initiate a mobile agent and once in the collection of the results when the task is complete.

In addition, no additional calculations are required for the client agent, because the mobile agent discharge calculations available server resources, and this proposed architecture could be extended to business B2B model. An individual user can grow up to become a small business, once it operates its case more professional and more widely.

5. Communication by message

In our work, communication is possible according to several dimensions such as communication between master Aglets and agent directory, communication between master Aglets and mobile agent, communication between the mobile agent and the provider agent. Communication is one of the most important opportunities offered by the Aglets platform; adopted paradigm is the passage of the asynchronous messages (Future-type messaging). Master Aglets send a message to the agent mobile to launch research to satisfy the demand of the user for example (Figure 5).

![Figure 5. Communication by message](image)

6. Conclusions and perspectives

In this paper we present an architecture based on mobile agent that allows the users to do the business anytime and anywhere via their mobile devices, the results obtained from our validation confirm us that the use of several mobile agents allows to improve the quality of the proposed solution and to reduce the traffic on network by passing on only the useful data and allows the user to reach a wide choice of products and services of a way anywhere and anytime. As we have seen, mobile consumers only need to use a low bandwidth connection twice, once to initiate a mobile agent and once in the collection of the results when the task is complete.

In this architecture, we have not taken into account the security of transactions such as payment online; this would also increase as a new feature of the system in our future work. Subsequently, a process of clearing of transactions with mobile devices would be an interesting research in the future.

7. Reference