Implementation of Cloud Computing Approach Based on Mobile Agents

Alwesabi Ali , Almutewekel Abdullah
Computer science department, university of Batna, Algeria
Batna, Algeria
Email: elwessabi {at} gmail.com

Okba Kazar
Computer science department, university of Biskra, Algeria
Biskra, Algeria

Abstract *— **In this paper, we present and evaluate an architecture based on mobile agent for cloud computing, The cloud computing are mainly used for the treatment of loads computer work very intensive and provide very large storage of data, these two objectives are combined with the third goal of potentially reducing management costs and use, cloud computing is attractive to business owners as it eliminates the requirement for users to plan ahead for provisioning, and allows enterprises to start from the small and increase resources only when there is a rise in service demand.***
Today, most applications of cloud computing use the model "client / server" where the exchange is done by sending messages across the network is the most widely used model, This model has the disadvantage of increasing the traffic on the network and requires a permanent connection.
In this paper, we propose a new approach that uses mobile agents in cloud computing, Our architecture is based on mobile agents that have kept the goal of communication in security in cloud computing.
This work is essentially the proposal of an architecture that can respond to user needs through access to a cloud computing secure with mobile agents, we are relying on the ability of mobility and security agents.

Keywords- Multi-agent System, Mobile agent, cloud computing,, java, Aglets.

I. INTRODUCTION

The Cloud computing is a concept that combines several technologies for deliver different services.

It can be schematized by an unknown set of computer resources interconnected in a network (Figure 1).

The Cloud Computing is a passage from the computer to the Internet. Users are no longer owners of their computer servers but may gain many services online scalable without having to manage the underlying infrastructure, often complex. Companies in this context would no longer need for clean rooms or servers or computer scientists. All applications are leased and run through the browser or application servers.

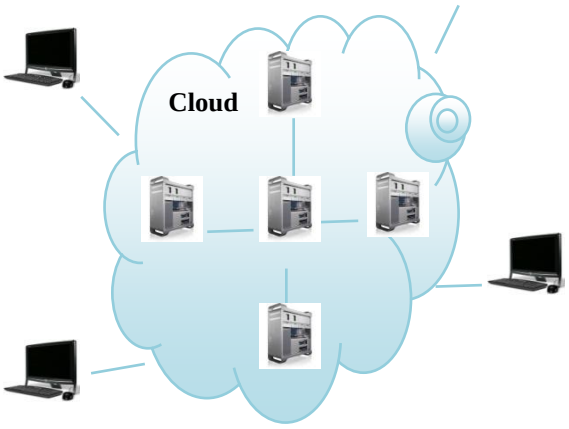


Figure 1: Cloud Computing.

We distinguish three types of cloud computing: the public cloud, private cloud and hybrid cloud is actually a combination the first two (Figure 2).

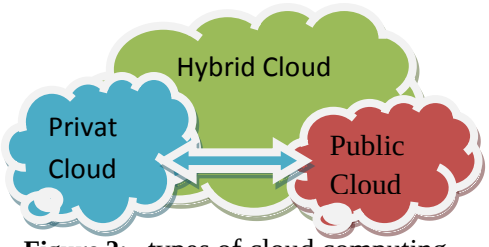


Figure 2: types of cloud computing.

- The Public Cloud: the idea is to host applications, Web applications in general, on shared environment with an unlimited number of users. The implementation of this type of cloud is managed by third parties (such as Amazon, Google, etc...) and is available on a pay-as-you-go model [1]; Providers of the most popular public cloud are Google and Amazon.

- The Private Cloud: this is a deployed environment within an enterprise Thus; it must manage its infrastructure alone. In this case, implement a private cloud signify transform the internal infrastructure using technologies such as virtualization to deliver services to request, more simply and faster. The advantage of this type of cloud from the public cloud lies in the aspect of security and data protection [1].
  Eucalyptus, OpenNebula and OpenStack are examples of solution of the implementation of private cloud.
- The Hybrid Cloud: in general, the term hybrid cloud cohabitation and communication between a private cloud and a public cloud in an organization sharing data and applications

The providers of cloud distinguish three services of cloud computing

- Software as a Service (SaaS)
  SaaS software is used directly on the network, without being downloaded first in the local computer user environments. The software applications are available on the Internet via a SaaS provider, and are executed in the computing environment predefined from this supplier [2].
  Amazon S3 (Amazon Simple Storage Service) is an example of SaaS is a storage platform online. It uses a web interface to store and retrieve data.
- Infrastructure as a Service (IaaS)
  IaaS is a complete computing infrastructure used as a service. To create and use their computing infrastructures freely, according to their needs and only when they need it, users or tenants, access to specific parts of a consolidated pool of federated resources [3].
  Amazon EC2 (Amazon Elastic Compute Cloud) is an example of IaaS allows rent virtual machines predetermined sizes to run the applications.
- Platform as a Service (PaaS)
  PaaS is a computing environment available and accessible, as needed, from an service provider. Used to develop and run software [4].
  Hadoop is an example of PaaS for distributed applications and intensive management of huge amounts of data.

The Cloud computing enlarged the area of distributed computing systems by providing advanced Internet services complement and complete functionality of distributed computing provided by the Web, grid computing and Peer-to-Peer. [5]
In fact, the cloud computing systems provide an infrastructure for large-scale IT has high performance that dynamically adapts to the user and application needs

Today the clouds are mainly used for handling loads very computationally intensive and provide very large storage facilities data, these two goals are combined with the third goal of potentially reducing the cost of management and use.
Another distributed computing paradigm based on multiple agents that are capable of interacting with intelligent behavior.
Multi-agent systems are often used to solve problems using a decentralized approach where multiple agents contribute to the solution by cooperating with each other
Although several differences between cloud computing and multi-agent systems, they are two of distributed computing models, so many common problems can be identified and several advantages can be obtained by integrating the use of cloud computing systems and multi-agents.
In this paper we propose a new approach that uses mobile agents. These agents are entities that move from one machine to another over the network, without losing their codes or their states. Thus, by sending agents where the tasks are, the messages exchanged become local and release all network loads.

This article is structured as follows: in the next section we will briefly present a number of studies linking mobile agents and cloud computing. Thereafter, our new approach to cloud computing which based on mobile agents will be detailed in the third section. In the section 4 we present our validation. Finally, a conclusion is a summary of the research and presents the perspectives considered.

II. RELATED WORKS

In the literature there are few works that use mobile agents in the cloud computing: first we have the work of Priyank et al [6], they propose a trust model based on security agents, which are simple mobile agents that provide security at the virtual machine and the entry point of the network cloud to cloud customers and service providers to manage their resources and data safely and efficiently. These mobile agents not only provide security measures, but also ensured the accounting and monitoring activities in the virtual machine if its malicious or normal state, so that the client is kept informed of the data. If alarming conditions, the client is informed and can take the necessary measures required.

The second architecture is OCCF [7]

Open Cloud Computing Federation (OCCF) is a concept proposed by several researchers, which consists to incorporate and to use several CCSPs (Cloud Computing Service Providers), to provide a uniform resource interface for the clients; the OCCF is based on some notions that can be a good base, to solve the problems of portability and interoperability between the CCSPs. The OCCF is advantageous compared to the other systems, in the

following points: Unlimited Scalability, Availability of resources, and Democratization of the Cloud Computing market, Deploying application on multiple CCSPs, and Reduced the cost to the clients.

Mobile Agents based on Open Cloud Computing Federation (MABOCCF) [8], is a new mechanism that allows the realizing of portability and interoperability, allowing an easy and inexpensive implementation of the OCCF.

The experiment results manifests that, the using of MABOCCF optimize the access to the resources over Internet by 50.35% compared with normal systems that don't support the portability between different CCSPs.

III. Our Approach

In this section are first presented the objectives of the proposed system, its overall architecture, highlighting its four main layers and overall functioning.

A. *The objectives of the systeme*

Most of cloud computing applications use the traditional client/server model in which an operation requires generally a permanent and stable communication between the user and the server, thus the traditional client/server approach constitute an obstacle to the development of cloud computing applications. The concept of mobile agent appears in this context as a solution to facilitate the implementation of dynamically adaptable applications, and provides a generic framework for the development of cloud computing applications. In this model "mobile agent" means an agent is a process with an execution context, including code and data can move from machine to machine (called servers) to perform the task assigned to it [9] . A priori, the advantages of mobile agents are numerous:

I. -The execution of specialized agents offer advantage of flexibility more that running a standard procedure on the server site, and allows transactions more robust than remote transactions.
II. - Agents are able to search for information in a smarter way, for example searching by concepts. Agents are also able to correct queries the user, based on the model attached to them.
III. - Agents can create their own knowledge bases that are updated after each search. If the information exchange site, agents are able to find it and subsequently adapt to this change. In addition, agents are able to communicate and cooperate with each other (and this is their real strength), which accelerates and facilitates research.

As we have already mentioned, the architecture proposed in this work is a mobile agent based approach designed for the execution of a service in cloud computing (SaaS). It defines a set of components (agents) and functional modules described in terms of their behavior and interfaces, and how these components interact in order to accomplish all the tasks correctly in the system.

B. *Our proposed framework architecture*

The general architecture of our system, shown in Figure 1, is divided into four main layers interact.

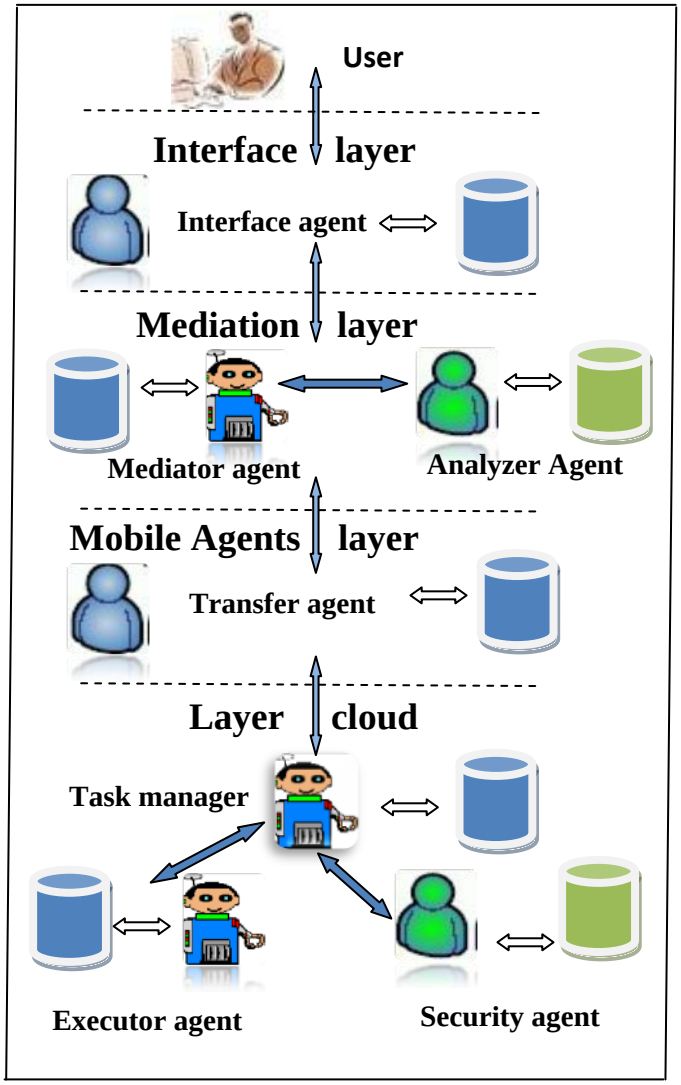


Figure 3: Representation of the layers of our system.

- **Interface layer**

This layer contains the application that allows the client to query the system.

Its primary role is to capture the purpose of the user to meet his needs best. It includes an interface that interacts with the user to help achieve specific task. This interaction results in

a transformation of the demands of the user, then sent a mediator agent.

- **Mediation layer**

This layer contains all the components necessary for the execution process of a written request by the agent interface, it also provides services such as the creation of mobile agents based on user requests. It also has a directory database that stores the addresses of all task managers in cloud computing accessible and can receive information on services available or send other mobile agents providing the services necessary to achieve its goal.

At the end of the process the mediator agent and agent analyzer analyze and collect the information found as a service that is requested by the user.

- **Mobile Agents layer**

This layer contains all transfer agents generated by the mediator agent. At this level, and for each service request, the mediator agent will activate a set of transfer agents. Transfer agents are mobile agents sending to task manager of cloud computing to find suitable services for each service request on various cloud computing.

- **Layer cloud**

This is a class capable of receiving service requests, process and returns the results. This layer includes all task managers, security agents and executor agents which are distributed over the cloud to manage all service associated with it.

*C. **Specifying Agents***

- **Interface agent**

This agent can be seen as simplifying allowing users to interact with the system, it's a stationary agent that runs on a user's device and provides a graphical interface to interact directly with our system it's primarily responsible to acquire all user queries, send those requests to appropriate agents and present the results to users.

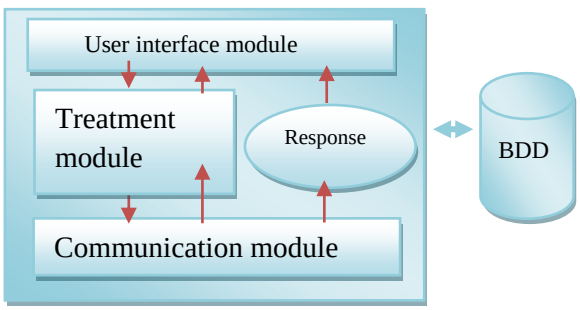


Figure 4: The interface agent.

- **Mediator agent**

The mediator agent is an intelligent agent that treats service request; he plays the role of interface between the user poses a query or request for service or resource and all sources of cloud computing that provides services, for each request the mediator agent will generate a mobile agent to move to the task manager of the cloud to find adequate demand services acquired.
When the execution of request is completed, the mediator gathers the results found by the mobile agent as a service that is requested by the user. The mediator agent has the ability to control the motivations of transfer agents who come to him thereby increase the level of system security.

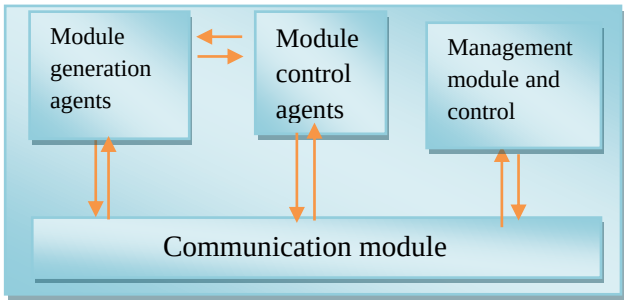


Figure 5: The mediator agent.

- **Analyzer agent**

This agent communicates with the mediator agent to analyze the authentication information and to analyze the request and the response of the system at the request of users. So its role is based on a request of the user, the analyzer agent selects a list of cloud provider that has the requested information with service using a database and when performing the application is complete, the analyzer shows the results found by the transfer agent and the analyzer agent then returns the result to the mediator agent.

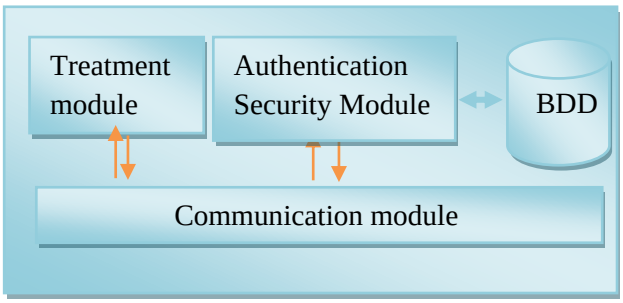


Figure 6: The Analyzer agent.

- **Transfer agent**

The transfer agent in our system is the mobile agent and it's created by the mediator agent who tells him the task. These transfer agents migrate to a cloud specified to satisfy

the request by the interaction with the task manager located in a cloud specified, and then he returned to the mediator agent to get the answer to the analyzer agent. Finally, he destroys himself.

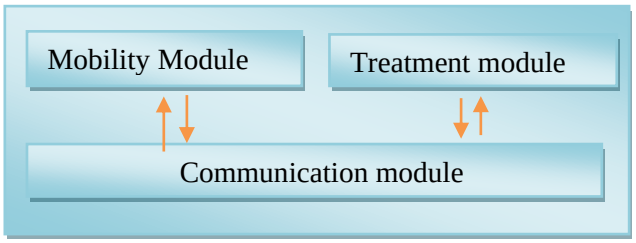


Figure 7: The Transfer agent.

- **Task manager**

The role of this agent is to maintain order in the cloud; it provides all the necessary information for each agent. For example notice security agents of arrival of a transfer agent. It controls and manages all the other agents of cloud as executor agent and security agent. The agent task manager is a temporary agent responsible for carrying out the service requested by the customer service.

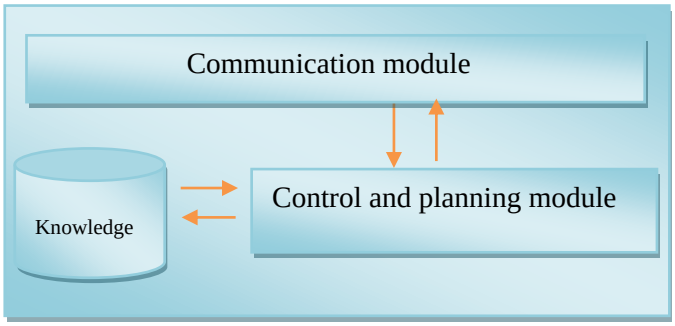


Figure 8: The task manager agent.

- **Security agent**

The role of this agent is to maintain the security, data integrity and authentication of partners cloud

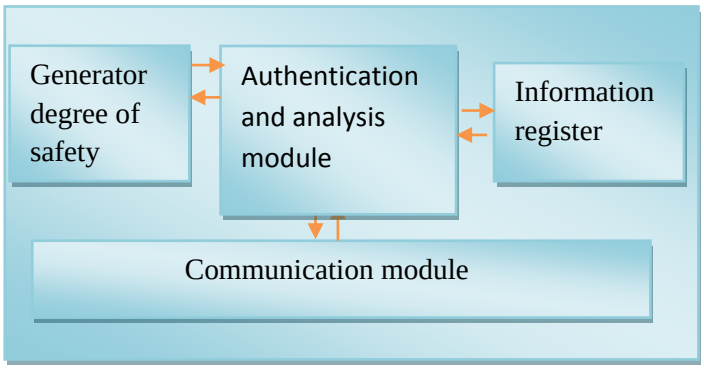
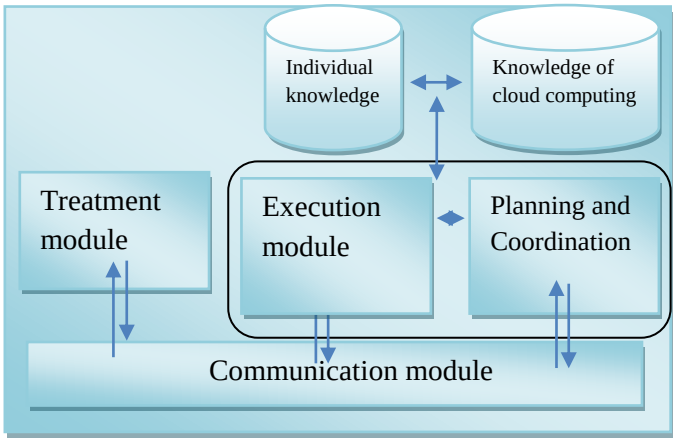


Figure 9: The security agent.

- **Executor agent**

It is a local agent at the cloud, it is responsible for responding to requests from mobile agents arriving at its cloud computing. Executor agent is represented by a structure comprising a communication module, a module for planning and coordination, execution module and treatment module. The executor agent has two other knowledge modules, which are: individual knowledge and knowledge of cloud computing



IV. VALIDATIONS

To show the validity, reliability and scalability of our architecture, we have interest in a case study.

Hence, we will apply our approach on an organization such as travel, this example considered a simple example to simulate working on a cloud. Assume the following scenario: Walid lives in Batna; he wants Skip the vacancy to another city. It prepares travel arrangements.

It can use an airport near Batna and an airport near the city you want. He should consult an airline flight, it must determine the price and must determine the desired location and desired time and then he started to request from their device. From the previous scenario, we can say that the process of organizing the travel is the consultation site of an airline to make a reservation for a flight; this example represents an application come a service whose purpose is to seek reservation a flight, good price in the desired time and to the desired 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. Our system consists of several agents that cooperate to meet the service demands of customers, we distinguish: The interface agent, the mediator agent, analyzer agent, transfer agent and security agent task manager agent and executor agent The agents in our system are implemented using the Java [10] language and development platform Aglets [11, 12, 13],

Aglets supports the development of agents with the ability to transport them from one system to another.

The interface agent, the mediator agent, analyzer agent task manager agent, security agent and the executor agent are stationary agents, while transfer agents are mobile agents traversing into the network to collect information for their clients.

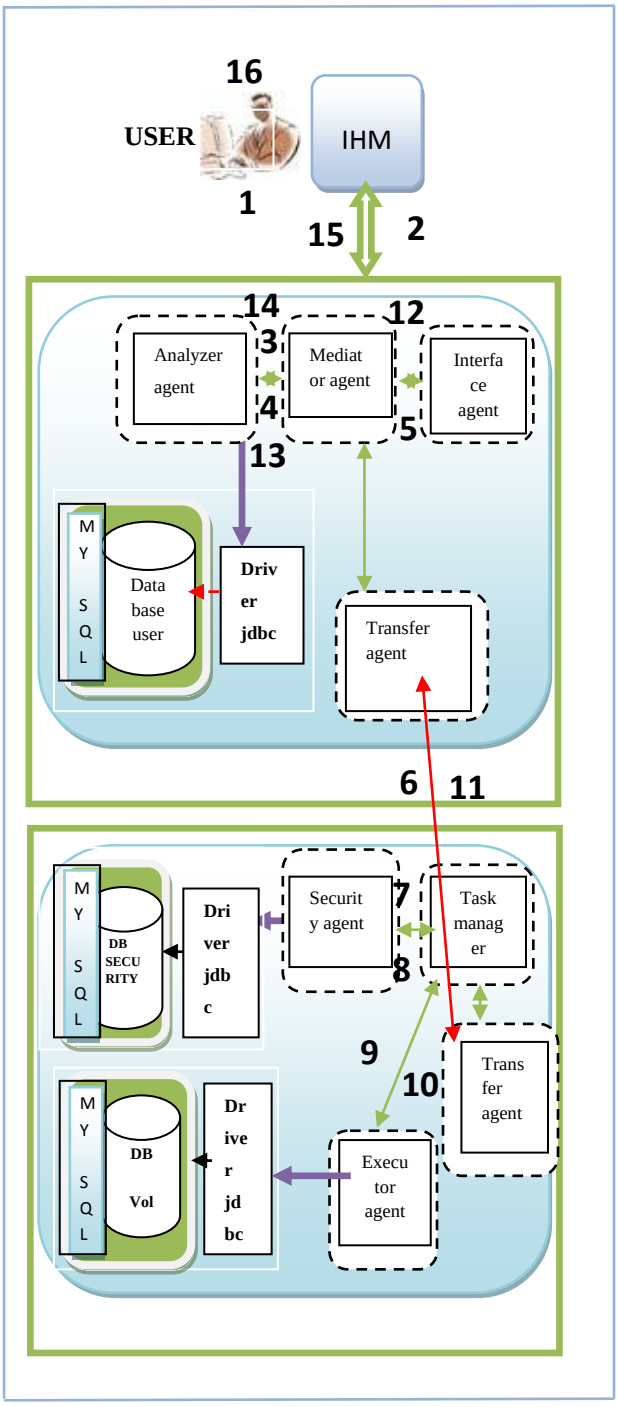


Figure 11: The software architecture.

A. PROPOSED METHOD

STEP 1: Send the user request to an agent interface. STEP 2: Start the agent interface and sends the request to the mediator agent.

STEP 3: The mediator agent sends the request and the information access to agent analyzer.

STEP 4: The agent returns the analyzer response to the mediator agent, if the information access is correct the request will be passed otherwise be abandoned.

STEP 5: The mediator agent creates transfer agents.

STEP 6: The transfer agent migrates to the cloud to transfer the information or service requested by the user.

STEP 7: task manager sends the request to the security agent to determine where the request arrival and verified the information access and the degree of security of our system where the arrival request.

STEP 8: Security Agent sends the response back to the task manager, if the information access is correct and our system where the arrival request are confident the request will be passed otherwise be abandoned.

STEP 9: Start the agent executor

STEP 10: The executor agent performs the requested service

STEP 11: The task manager sending the result to the transfer agent and transfer agent migrates to our system.

STEP 12: The transfer agent sending results to the mediator agent.

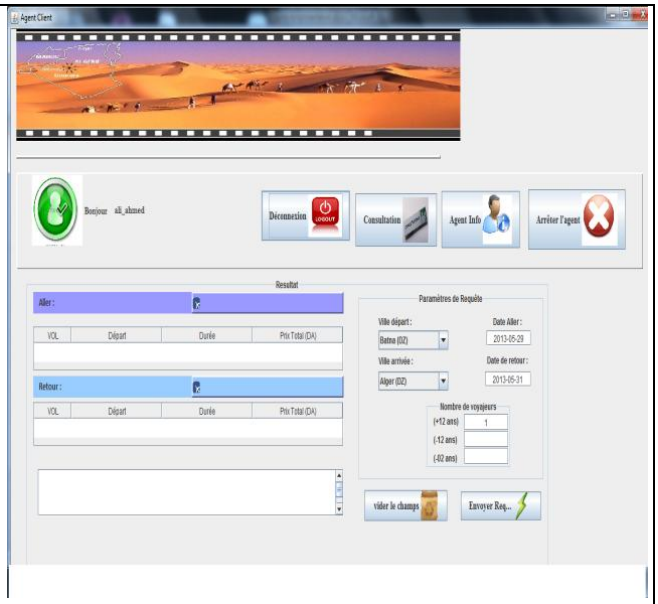
STEP 13: The mediator agent sends the results to the analyzer agent.

STEP 14: analyzer agent sorted the results sent by the mediator agent and returns the answer to this agent.

STEP 15: The mediator agent sending the result to the interface agent.

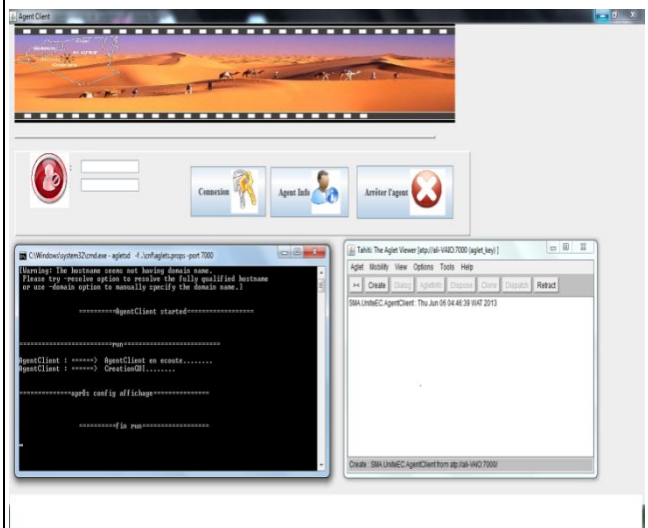
STEP 16: Displaying the result to the client.

B. IMPLEMENTATION RESULTS



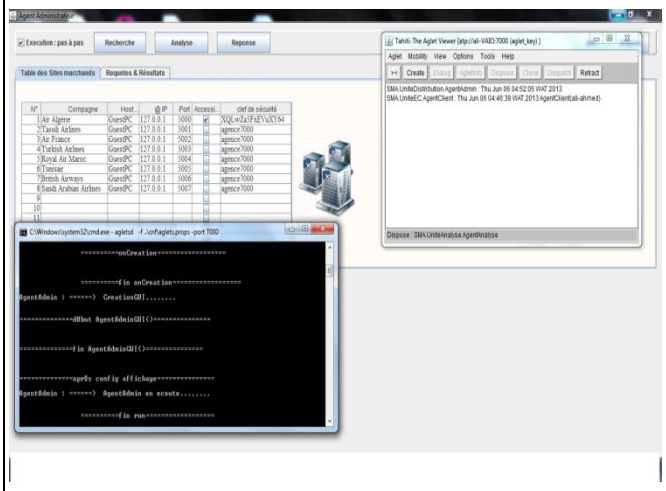
1) Shows the interface to launch the application

Figure 12: Graphics display window.

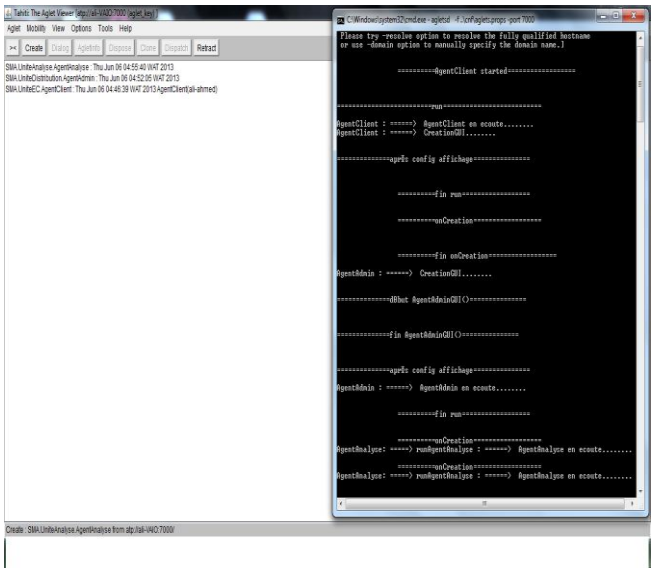


2) launch the interface agent

Figure 13: Graphics display window.

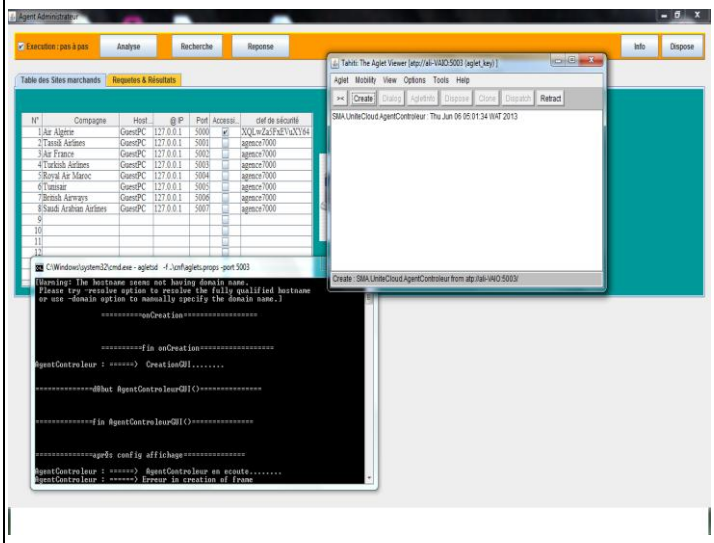


3) launch the mediator agent

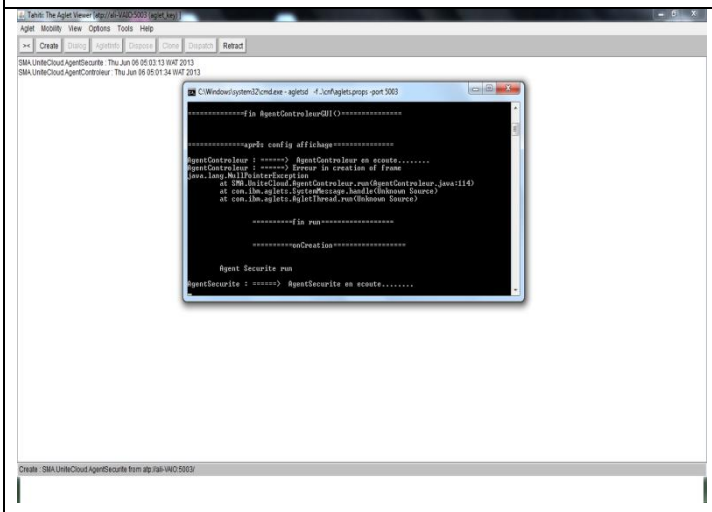


4) launch analyzer agent

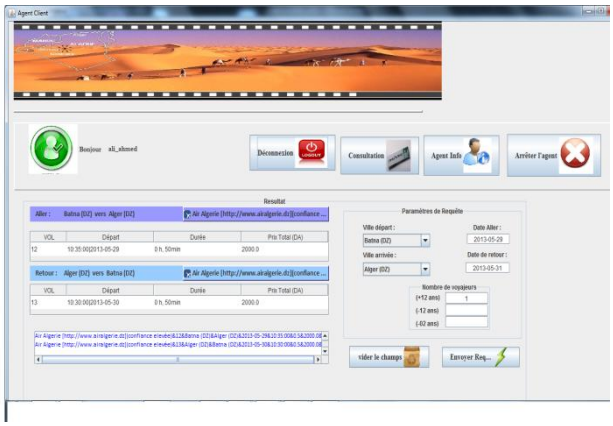
Figure 14: Graphics display window.



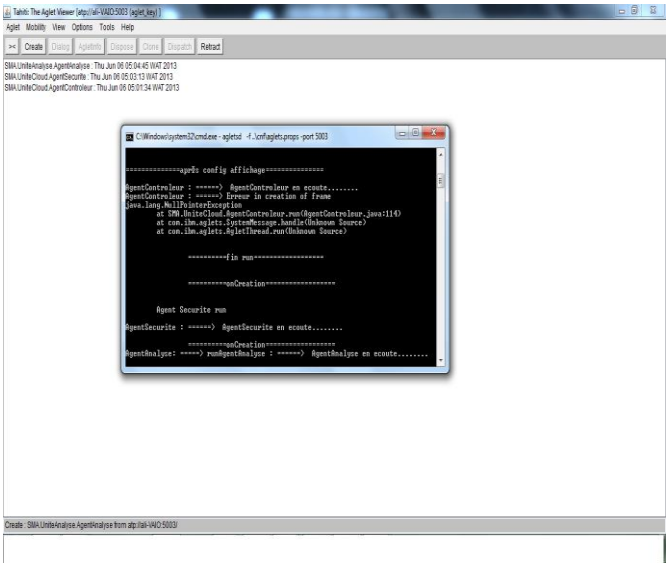
5) Launch the task manager agent.



6) Launch the security agent.



7) launch executor agent



8) Get the result to the user.

Figure 15: Graphics display window.

V. CONCLUSIONS AND PERSPECTIVES

In this paper, we are interested in the technology of mobile agents and their use for cloud computing environments. The purpose of the movement of these agents is generally access data locally or remote resources, making the local treatment and move only useful data. That is why we propose in this context, a new approach based on mobile agents to execute a service in cloud computing (SaaS).

The advantage of this architecture is that it uses mobile agents as a communication entity. This is to reduce traffic on the network and reduce the amount of information exchanged, in which case the agent moves to the source information and performs local exchanges.

To demonstrate the feasibility of the proposed architecture on the one hand, and to better understand how it works the other hand, we have chosen as a case study the organization of travel because it is considered as a typical example to execute a service on the web. In this context, one mobile agent is created. It is responsible for moving to perform the task requested by the client.

We believe that the use of multiple mobile agents to search for information improves the quality of the proposed solution and to reduce the waiting time of the user.

The different results obtained in the experience presented show that mobile agents can positively contribute to the development of a cloud computing system. Further research can be undertaken to improve the work presented.

That we propose the following:

- Integration techniques of natural language processing including user interface level, in order to make the system more efficient.

- Take into account the adaptability of agents' appearance.

- Use of cooperation mechanisms between mobile agents in order to effectively perform the tasks required.

REFERENCES

[1] http : //mysaas.fr/2010/10/04/private-cloud-publique-cloud-et-hybrid-cloud/.

[2] http://www.emc.com/corporate/glossary/software-as-a-service.htm

[3] http://www.emc.com/corporate/glossary/infrastructure-as-a-service.htm

[4] http://www.emc.com/corporate/glossary/platform-as-a-service.htm

[5] W. Kim, "Cloud Computing: Today and Tomorrow," Journal of Object Technology, vol. 8, no. 1, January-February 2009, pp. 65-72.

[6] Priyank Singh Hada,Ranjita Singh,Mukul Manmohan Meghwal " Security Agents: A Mobile Agent based Trust Model for Cloud Computing" *International Journal of Computer Applications (0975 – 8887) Volume 36– No.12, December 2011*

[7] B. Rochwerger, D. Breitgand, E. Levy, A. Galis, K. Nagin, I. Llorente , et al. "The Reservoir model and architecture for open federated cloud computing," IBM Journal of Research and Development, Volume 53, April 2009, in press

[8] Zhang. Z and Zhang. X, "Realization of Open Cloud Computing Federation Based on Mobile Agent", *IEEE*, 2009.

[9] S. E. Falou : Programmation répartie, optimisation par agent mobile. Thèse du Doctorat de l'université de CAEN, soutenue le 29 Novembre (2006).

[10] J. P. Arcangeli, V. Hennebert, S. Leriche, F. Migeon, M. Pantel : Tutoriel JavAct (version 0.5.0), IRIT (2003).

[11] J. P. Arcangeli, V. Hennebert, S. Leriche, F. Migeon, M. Pantel : JavAct 0.5.0 : principes, installation, utilisation et développement d'applications. Rapport de recherche, IRIT/2004-5-R, Février (2004).

[12] Danny B. Lance/Mitsuru Oshima: Programming And Deploying Java Mobile Agents with Aglets: Addison -Wesley, 1998, pp 378-- 491.

[13] http://aglets.sourceforge.net/.