5th International Conference on Control Engineering&Information Technology (CEIT-2017)
Proceeding of Engineering and Technology –PET
Vol.33 pp. 35-39

Semantic web service discovery approach in cloud computing

Souraya Hamida[#1], Okba Kazar[*2]

[#]*institute staps, LINFI Laboratory, University of Batna 2, Mostefa Ben Boulaïd*
University of Batna 2, Mostefa Ben Boulaïd, constantine street, number 53, Fisdis, Batna, Algeria
[1]hamidasouraya@yahoo.fr

[*] *Department of Computer, LINFI Laboratory, University of Biskra*
BP 145 RP, 07000 Biskra, Algeria
[2]kazarokba@yahoo.fr

***Abstract*—In this paper, we propose the integration of three technologies: mobile cloud, web services discovery and mobile agents. This allows wireless users to discover and execute enriched semantic web services. This is without the need for a simultaneous presence of the user. In addition, we will use the context information defining user and Web service. These ones to satisfy the user request.**

***Keywords*—cloud computing, mobile agents, service discovery, SOA, ontology, mobile environment.**

I. INTRODUCTION

During the last decade, the SOA offers the robustness and flexibility. It also allows companies to effectively carry out their activities. In addition, Web services enable simplified implementation of SOA. The web services approach is a key factor in the seamless integration between heterogeneous applications. In addition, it allows the reuse of applications, interoperability, and a loosely coupled architecture.

Nowadays, mobile web services are the application of the web services technology in a mobile environment [1]. In addition, they offer new personalized services to consumers on their mobile devices (such as smart phones, PDAs). Mobile web services classes in three categories. Classification is done through the web service requester and the degree of service provider deployment. The simplest case of deployment of mobile web services is a mobile requester and a fixed applicant provider. Exchanging roles of plaintiff and supplier of fixed and mobile node, the mobile terminal now offer a web service and the fixed node requests it. In the latter case, the requester and the provider are both mobile and communicate with each other in a peer-to-peer [2].

In recent years, the application of cloud computing has been widely recognized as IT infrastructure of the future generation. On the other hand, mobile cloud computing was introduced following the concept of cloud computing. This one associates a combination of mobile Web and cloud computing. In addition, it is the most popular tool for mobile users to access Internet services.

Moreover, mobile devices and wireless environments have different variability factors (frequent disconnection, discharge the battery, bandwidth, resource smallness in terms of memory, processing power, storage capacity and the size of the screen) [3] [4].

The contributions of this article can be summarized as:

- The use of the term "discovery of the Web service as cloud computing based Service (DaaS) ", presented by Algazar et al [5]
- Exploitation semantic web technology to automate the web services discovery stage,
- Exploiting the advantage of mobile agents to overcome scientific obstacles related to performance
- The use of context information (user location and the type of device used) to better meet user needs.

The remainder of this paper is organized as follows. In Section 2, we discuss related work. While in Section 3 we present our proposed architecture. Section 4 is to provide a description of the entities used in our architecture. Finally, Section 7 concludes the paper.

II. RELATED WORK

In this context, there is an infinitesimal use of mobile cloud computing based agent. In [6], the authors propose an architecture that uses mobile agent in cloud computing. Their objectives are the integration of three technologies, namely grid, cloud computing and mobile agents. The specific role of the grid is to provide a common and secure infrastructure to manage the virtual group of the cloud by using mobile agents. The agents introduce the elements which provide to the users a simple way of configuration for the virtual groups. The authors demonstrated that the use of mobile agents is well suited to cloud computing because the application/service based agent adapts better to systems with dynamic evolution.

Priyank et al proposes a trustful model based on security agents [7]. These agents are simple mobile agents. They provide security on the virtual machine, client cloud entry points of the network and the level of service providers. These agents monitor the integrity of the virtual machine and authenticity. Security agents can move dynamically in the network, to reproduce as needed and perform tasks (accounting and control virtual machines).

In [10], the authors present the MABOCCF mechanism (Mobile Agent Based Open Cloud Computing Federation), in this mechanism the data and code are transfer from device to

ISSN 2356-5608

another via mobile agents. Each mobile agent is running in a virtual machine called MAP (Mobile Agent Place); and mobile agents are able to move between MAPs and can communicate and can negotiate with others. The authors have the advantage of combining the mobile agent and cloud computing.

Ramaswamy et al [11] developed a new approach that combines the mobile agent and cloud computing. This ensures reliability in cloud computing. Architecture is articulated around three entities, namely: intermediate cloud, customers' cloud and provider cloud. To ensure trust, the authors use penalties, price points and mobile agents monitoring.

III. PROPOSED APPROACH

Our idea is based on the work of alghazar et al. [5]. Above all, we want to tell in a few words the work alghazar et al. then we present our study.

alghazar et al. introduced a new architecture based cloud. It addresses the basic elements of the discovery of mobile web services (Daas). They provide a complete analysis of needs for the mobile Web service discovery in limited resource environments. They include preferences and user context in service discovery to find services that better correspond to user needs.

Fig. 1 shows an overview of our system that uses mobile agents to discover mobile Web services in mobile cloud computing. We use the mobile agent to solve some scientific obstacles related to the discovery of Web services in the mobile computing cloud.

Resource constraints are a major challenge in the service discovery in wireless environments. These are: battery autonomy, frequent disconnection, narrow bandwidth and limited resources (memory, processing power, storage capacity and size of the screen). In addition, the mobile agent has the ability to resolve these problems, because, it functions in an autonomous and asynchronous way. It dynamically adapts to the runtime environment, reduces network traffic, and move from one site to another to perform the tasks locally. It is noteworthy that several research studies show that the mobile agent is one of the most appropriate technologies that can be used to meet the performance needs for the mobile Web service discovery.

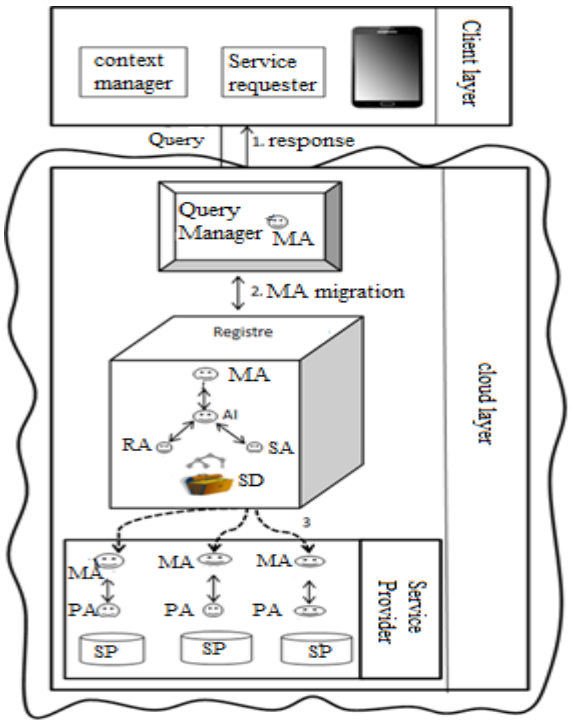

Fig. 1 Proposed architecture

In our system, the Web services discovery is accomplished in two steps [8]. The first step limited to the search of semantic descriptions of Web services to meet the request of the user in terms of functional parameters (inputs, outputs). The second step is to select only the Web service from the research stage, within the parameters of context (user location and user device).

A. Step 1: Semantic search:

At this step, the researcher Agent initializes the reasoning system with domain ontologies. This will be used to calculate the degree of semantic correspondence between the concepts mentioned in the input and output parameters specified in the request and those web services.

The Mobile Agent transmits to the Agent Researcher its request. Once the request is received, the Researcher Agent transmits the input and output parameters to reasoning system. This request is processed by the application of Matchmaking algorithm [9]. The algorithm allows the calculation of the degree of correspondence between the functional parameters (web services) and those claimed by the user. The result is a set of web services, satisfying the request of the user in terms of Inputs and Outputs.

However, the main idea of this algorithm [9] is to match all input and output parameters of the service to all input and output parameters of the respective request. The input and output parameters are mapped to the concepts of the domain

ontology. The degree of similarity between these parameters is obtained on the basis of their relationship and inference in an OWL ontology. In [9], four levels of similarity between the parameters are defined.

- Exact: Two concepts are classified as exact only if these concepts are exactly the same in the ontology or they have a direct relationship with the class and subclass in this ontology.
- Plug-in: Two concepts are defined as plug-in, only if the request input parameter is superclass of service input parameter except the direct relationship that is already treated by exact degree set matched.
- Subsumes: Two concepts are classified as subsumes, only if the request input parameter is subclass of service input parameter.
- Fail: This degree of similarity expresses no semantic relationship between two concepts.

B. Step 2: Select depending on the context:

This step is devoted to filter all Web services whose parameters of the context (location and type of device) are adequate to those determined at the request of the user.

IV. DETAILED DESCRIPTION OF UNITS USED IN ARCHITECTURE

Our architecture includes the following components:

A. Client layer

The user calls a web service from their mobile device. The mobile device gets in contact with any wireless device (PDA, IPhone, …). It consists of:

- Service requester: is the mobile device user. He connects and submits the request. This last is processed to leave the functional parameters (input and output).
- Context manager: is for collecting context information (used device and its location).

B. cloud layer

This layer contains the components that are deployed on the cloud.

1) Query Manager:

Application Manager performs advanced processing operations on the user request. The user request is processed out of the operating parameters (input and output), device type and location of the user. Subsequently, Query Manager creates a Mobile Agent, which migrates to find Web services that meet the needs of users.

2) Mobile Agent:

It Represents the user in the fixed network. He is able to find, run web services and provide results to the user. The mobile agent can also create clones that run web services selected in parallel to minimize the total processing time. The clones can migrate and run web services simultaneously and return the results to the web service requester.

3) Register:

The register of services is composed of three agents, namely the Agent Interface, the researcher Agent and selector agent. Also, it contains descriptions of web services and domain ontologies.

- Agent Interface (AI): This is a stationary agent that acts as an intermediary agent between the mobile agent and the other agents included in the service registry. It acts as a supervisor, controlling the interactions within the Registry Service.
- Researcher Agent (RA): It is a stationary agent, whose role is the discovery descriptions of Web services satisfying the request sent by the user on the semantic level.
- Selector Agent (SA) A stationary agent. Its main role is to filter this set by selecting Web services whose parameters of the context (location and type of device) are adequate with those determined at the request of the user.
- service description (SD): This is a register contains the semantic descriptions of web services

1) Service Provider:

It provides a web service to interested customers.

- Provider Agent: This is a stationary agent that resides in the web service provider. Its purpose is to hide the functionality of the web service provider. The Provider Agent is created and maintained by the service provider. When the mobile agent migrates to the service provider, he gets results with the supplier agent.

V. SEQUENCE DIAGRAM OF GENERAL FUNCTIONING

In this section, we describe the interaction is happening in our system by a sequence diagram (Fig. 2). The description of the various steps marked is the following:

- Client connects to the system and creates the request.
- The request handler creates a mobile agent that migrates to Registry Service to find Web services that meet the needs of users quite taking into account user context (location and device) (step 2).
- In the Web service registry, Mobile Agent (MA) sends the user request to the interface agent (AI) (step 3).
- Interface Agent sent to the Research Agent (RA), the functional parameter of user's query (input and output) (step 4).
- Secondly, the researcher agent initializes the system of reasoning with domain ontologies that will be used to calculate the degree of correspondence between the semantic concepts referred to input and output parameters specified in the query and Web services (step 5).
- In step 7, the interface agent sends to the Selector agent (SA) the context of the user and the result of the Researcher agent. Selector agent is responsible for selecting Web Services which parameters are appropriate context with those determined at the request of the user.
- Mobile Agent, after acquiring the appropriate details (Step 9), migrates to the service provider, invokes the

Web service, collects the results (step 10) and returns to the requesting to get mobile users' results. The Mobile Agent sends his clones to the Web service provider, to invoke services in parallel (and this way we reduce the time of invocation), instead of migrating in series to each of them. User login is not necessary and can get results in the future.

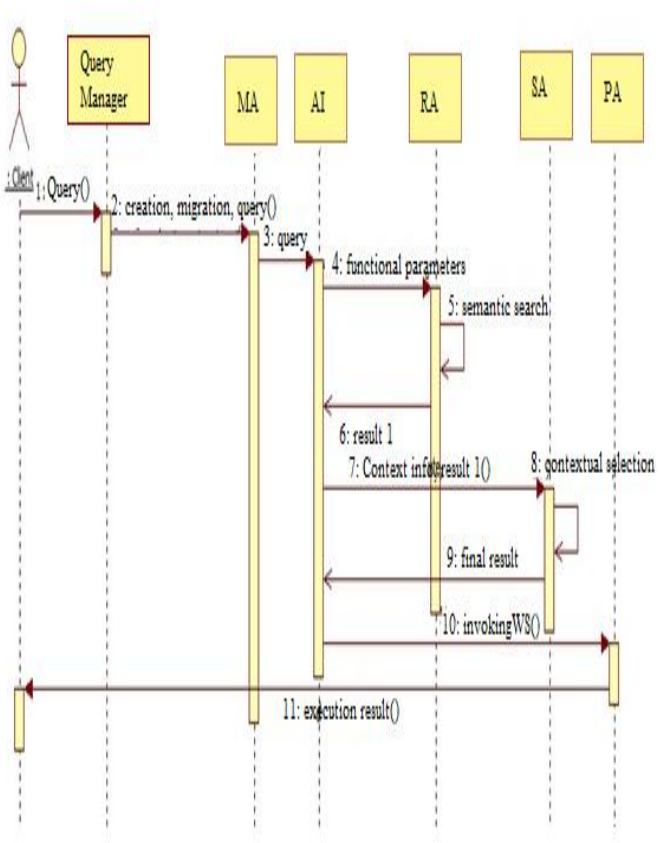

Fig. 2 Sequence diagram

VI. CLASS DIAGRAM

Fig. 3 illustrates the class diagram of our system

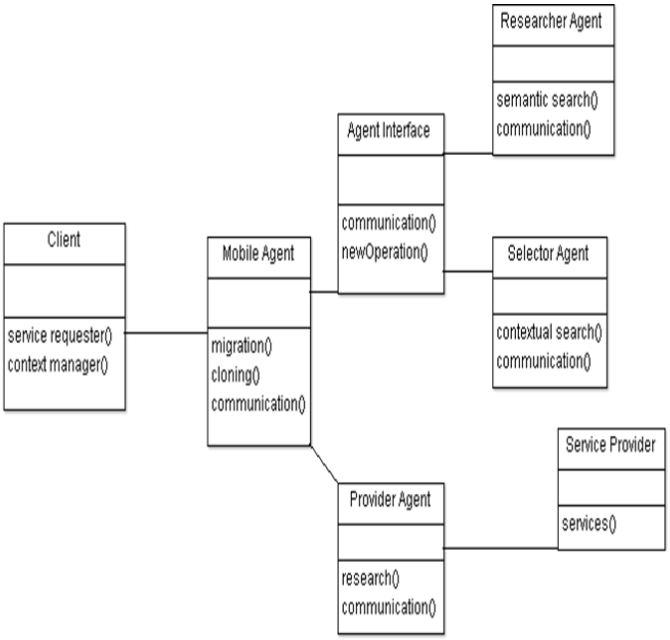

Fig. 3 Class diagram

VII. CONCLUSION

Nowadays, we can conclude that there is a growing interest in information technology in mobile cloud. It follows from the extension of cloud computing. This article outlines an architecture for discovering and invoking Web services in mobile cloud using mobile agent, and context information.

In our system, we use semantic web technology to automate the stage of discovery and invocation of web services. In addition, we use the context information to better meet user needs. In addition, we integrate the mobile agent paradigm to overcome scientific obstacles; that are related to the performance of mobile devices and wireless environments. It represents one of the most appropriate technologies that can be used to meet the performance needs for discovering Web service.

REFERENCES

[1] P. Farley et M. Capp. "Mobile web Services". BT Technology Journal, vol. 23, no. 2, pp. 202-213 (2005).

[2] Guido Gehlen. "Mobile Web Services - Concepts, Prototype, and Traffic Performance Analysis". Thèse de doctorat, Université de RWTH Aachen, Allemagne. 17 october 2007.

[3] Alin COBÂRZAN. "Consuming Web Services on Mobile Platforms" Informatica Economică vol. 14, no. 3, 2010.

[4] Ralf Sesseler, Alexander Keiblinger, Nicolas Varone. " Software Agent Technology in Mobile Service Environments". ISMIS. 2002.

[5] Khalid Elgazzar, Hossam S. Hassanein, Patrick Martin, "DaaS: Cloud-based mobileWeb service discovery", Pervasive and Mobile Computing 13 67–84, 2014.

[6] Rocco Aversa, Beniamino Di Martino, Massimiliano Rak, Salvatore Venticinque ,"Cloud Agency: A Mobile Agent Based Cloud System", International Conference CISIS, 2010.

[7] Priyank Singh Hada Ranjita Singh Mukul Manmohan, "Security Agents: A Mobile Agent based Trust Model for Cloud Computing",

International Journal of Computer Applications 36(12):12-15, December 2011

[8] Souraya Hamida, Okba Kazar et Youssef Amghar. « An Agent Oriented Approach for Modeling Web Services in Mobile Environments », International Journal of Web Applications, Volume 5, Number 1, pp: 13- 26, March 2013.

[9] Paolucci, Massimo., Kawamura, Takahiro., Payne, Terry R. et Sycara, Katia P. (2002). «Semantic matching of web services capabilities». In: ISWC '02: Proceedings of the First International Semantic Web Conference on The Semantic Web, p. 333–347, London, UK.

[10] Z. Zhang, X. Zhang, Realization of open cloud computing federation based on mobile agent, in: IEEE International Conference on Intelligent Computing and Intelligent Systems, ICIS 2009, vol. 3, pp. 642–646.

[11] Aravindh Ramaswamy, Aswath Balasubramanian, Palaniappan Vijaykumar, Varalakshmi, "A Mobile Agent based Approach of ensuring Trustworthiness in the Cloud", IEEE-International Conference on Recent Trends in Information Technology, ICRTIT 2011 MIT, Anna University, Chennai. June 3-5, 2011