Role-Based Mobile Agent for Group Task Collaboration in Pervasive Environment

Chih-Hao Liu and Jason Jen-Yen Chen

Department of Computer Science and Information Engineering
National Central University
945402024@cc.ncu.edu.tw

Abstract. The agent paradigm truly enhances the power of network software in pervasive environment. Agents with features such as autonomy and social ability form an agent community called agent group. Moreover, mobile agent technology that integrates mobile device with application server enables mobile users to obtain information or services over the Web anytime anyplace. However, one problem with agent community is that there seems no definition or knowledge representation for role-based agent to form an agent group that resembles a human group with roles. In this work, we propose the role-based mobile agent architecture, in which agents will be grouped into a specific group according to their roles. Further, through agent communication, mobile agents of an agent group can collaborate with each other to contribute to group tasks. A hospital example based on the Java agent development environment (JADE) platform is included for illustration.

Keywords: Mobile Agent, Role-based, Group Task Collaboration.

1 Introduction

The current research on web service primarily concentrates on somewhat of "low-level" distributed techniques and standards such as web service description language (WSDL), simple object access protocol (SOAP) and universal definition and discovery integrated (UDDI). The main advantage of this is to provide consistent interface for software entity to access web services. However, these web services are neither reliable nor easy-to-use due to the fact that they are at remote sites where a user has no control at all. Additionally, the web services provide a static description, thereby making it extremely difficult to update in a real-time manner.

On the contrary, the research on agent focuses on problem solving mechanisms in pervasive environment. Multi-agent system (MAS), which is composed of multiple interacting agents to form an agent community, is a powerful paradigm in pervasive environment for collaborative systems. Notably, an agent is a program with such features as autonomy and social and reasoning abilities. Additionally, it autonomously manages its resources and proactively collaborates with other agents with the same purpose to execute complex tasks such as making appointments or scheduling [5, 7, 8]. For example, an agent is able to proactively request other agents for medical recommendation, and then the requested agents respond reactively with the name of

R.-S. Chang, T.-h. Kim, and S.-L. Peng (Eds.): SUComS 2011, CCIS 223, pp. 234–240, 2011.

some recommended medicine through agent communication. Further, the Foundation for Intelligent Physical Agent (FIPA) standard that includes 22 communicative acts and some interaction protocols is widely used by agent developers to coordinate interoperation among agents [6]. However, one problem with agent community is that there seems no definition or knowledge representation for role-based agent to form an agent group that resembles a human group with roles.

In addition, a mobile agent is an agent with mobility, meaning it can suspend its execution, store its state, migrate to another web site, and resume its execution. Thus mobile agent is a promising technology for mobile applications [11], in which a mobile agent migrates to a remote web site to gather information or execute tasks. This technology is therefore suitable for pervasive environment.

Incidentally, the Semantic Web technology is highly-anticipated in developing an infrastructure for agents to perform complex tasks for their users [1, 2, 4]. The web ontology language (OWL) is a popular language to describe domain knowledge called domain ontology, and it allows people to utilize the uniform referential identifier (URI) to give each and every concept a formal term to be used in agent communication. It also defines semantics of the terms, and organizes all kinds of terms by using relations among them. Further, using OWL to annotate web services forms the semantic web service called web ontology language for service (OWL-S) with three OWL documents: 1) service profile, 2) service process, and 3) service grounding. Those documents can be viewed as the resources that an agent can autonomously manage.

Much recent research attention of multi-agent system has been upon agent group. Veiel, et al. propose agent-based approach to facilitate context-based adaptations using context information of a group of users [9]. Jiang, et al. simulate group behavior in E-Government to help making policy decisions [10]. However, these seem to lack knowledge representation about specific role of agent and agent group task. We thus proposed a role-based mobile agent architecture, in which agents will be grouped according to their role. Further, each agent in the group will contribute to the group task according to its role and personal knowledge.

To summarize the above, we propose a role-based mobile agent architecture in which an agent manages or updates its knowledge and resources and will be grouped according to their role and relationship. Through the communication mechanism, the role-based agents can collaborate with each other to perform their group tasks that will generate various actions that the agents will execute. In this work, the actions are implemented as semantic web services that are annotated web services.

2 Role-Based Mobile Agent Architecture

In this section, we will explain the role-based mobile agent architecture as shown in fig. 1. Our architecture contains Location Server, which provide web services to user. The server includes five parts: 1) provider service, 2) user service, 3) role-based agent model, 4) role-based agent ontology, and 5) multi-agent system. Firstly, the provider service is a web service, which is responsible for registering a service provided by service provider. Secondly, the user service is a web service running on the server and it is responsible for receiving user's request. Thirdly, the role-based mobile agent model is responsible for dynamically creating role-based agent according to role information, which is described in role-based agent ontology.

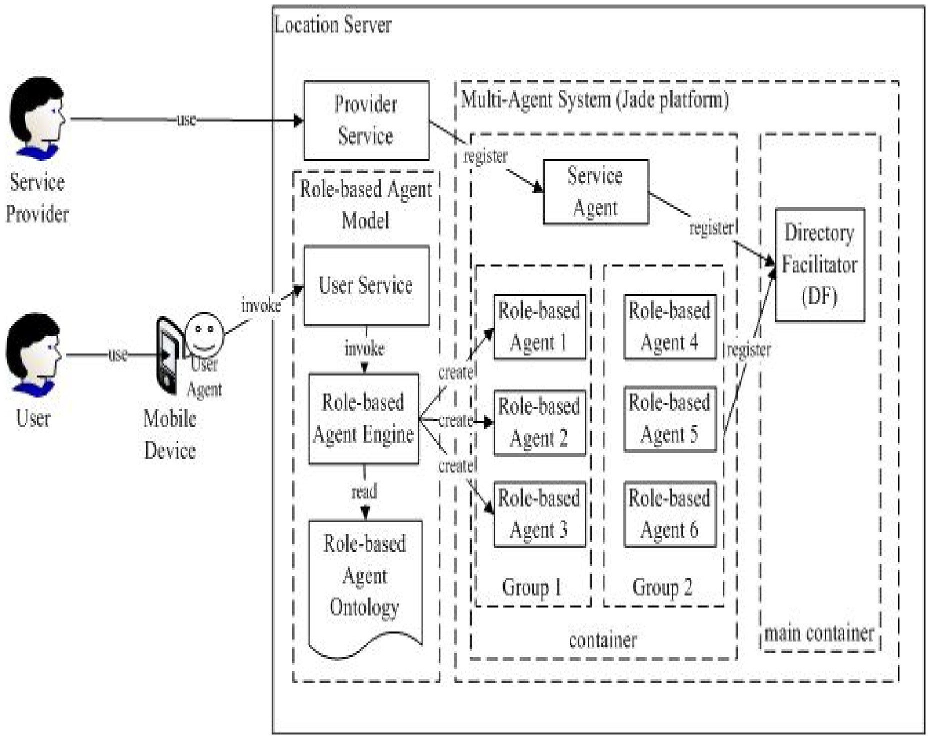

Fig. 1. Role-based Mobile Agent Architecture

The role-based agents could compose a group according to their role and relationships to share information or to achieve tasks through collaboration. In the Fig. 1, the role-based agent 1, the role-based agent 2 and the role-based agent 3 are in the group 1, because the friendship exists among them. Fourthly, the role-based agent ontology will be described shortly. Fifthly, the multi-agent system is based on the Java agent development framework (JADE) platform [3], which is a widely-used agent development tool that supports efficient deployment of mobile agents. And, it contains main container and the container. The main container includes directory facilitator (DF) agent, to which an agent can register. And the container includes service agent and role-based agent, which are created by the service provider and the user, respectively. Next, the role-based agent ontology is shown in Fig. 2.

In Fig. 2, an agent in "Agent" class has roles defined in "Role" class. Agents are of a group if their roles have relationships in a specific domain, which is defined in "Domain" class. For example, a patient named John and his doctor has medical relationship in the medicine domain, thus they belong to the same group. A group is defined in "Group" class that has tasks, which group members will collaborate to achieve. The task is defined in "Task" class and it belongs to the domain. When an agent executes a task, it will trigger an event. And then, the event will generate an action, which an agent will execute, to satisfy the task.

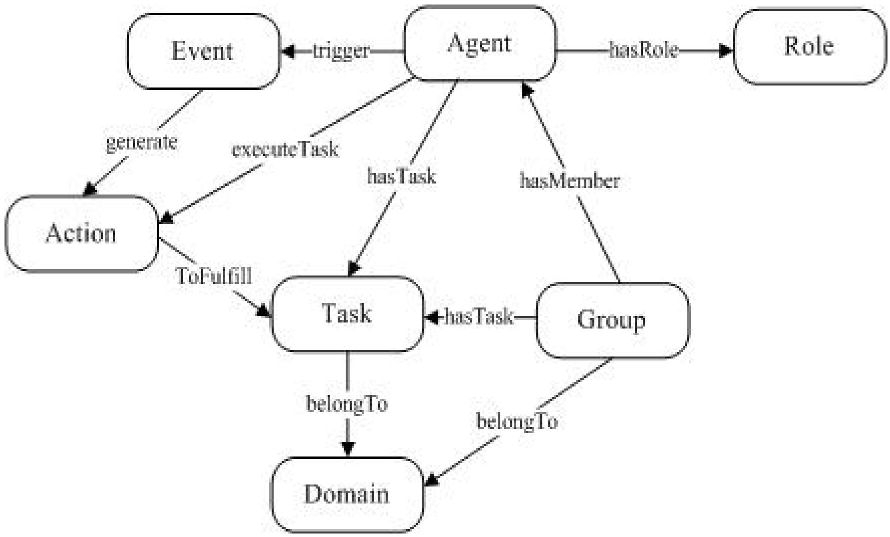

Fig. 2. Role-based Agent Ontology

When composing a group, agents refer to some rules. There is a rule definition about simple medical relationship of the medicine domain as shown below:

$(Role_{Doctor} \wedge Role_{Nurse} \wedge Role_{Patinent} \wedge$ Treat (doctor, patient) $\wedge$ Nursing (nurse, patient) $\mid Domain_{Medicine}) \rightarrow$ trigger (ComposeGroup)

where RoleDoctor stands for a doctor; RoleNurse for a nurse; RolePatinent for a patient. And, there exists a "Treate" relationship between the doctor and the patient in the medicine domain. Note "|" specifies the domain. Similarly, there exists a "Nursing" relationship between the nurse and the patient. Thereby, they will be grouped into the same group.

3 An Example

This section illustrates a hospital example based on our architecture as shown in Fig. 3. The JADE platform runs on a hospital server that includes main container and container. A DF agent runs on main container. Agents in container will be registered into the DF agent.

In the container, the service agent provides all web services of the hospital server. Each and every person will use a mobile device, on which a personal agent runs. For example, a patient, called John, uses a mobile device on which his personal agent, John agent, runs. And, John's doctor and nurse also have their personal agents "Doctor 1" agent and "Nurse 1" agent, respectively. When those personal agents migrate to the hospital server, they will be grouped into "Group 1" according to their role and relationships. Because John is a patient, we know that John has medical relationship with his doctor and his nurse. Similarly, when Mary agent migrates to the hospital server, it will be grouped into "Group 2" along with her doctor and her nurse.

Assume that John measures his temperature, pulse, respiration (TPR) values. If he gets an unusual temperature of 38.5℃. The John agent will trigger a TPR unusual event with an event description, which will be written into a task document as the group task. The John agent will also notify doctor 1 agent and nurse 1 agent about the event. After that, doctor 1 agent gets the unusual temperature value and requests nurse 1 agent for TPR curve. When nurse 1 agent receives the request, it will reply to doctor 1 agent with the TPR curve. Next, doctor 1 agent determines to prescribe an antipyretic named "Ibuprofen Suspension" and to send a call-for-proposal communicative act with unusual temperature to nurse 1 agent.

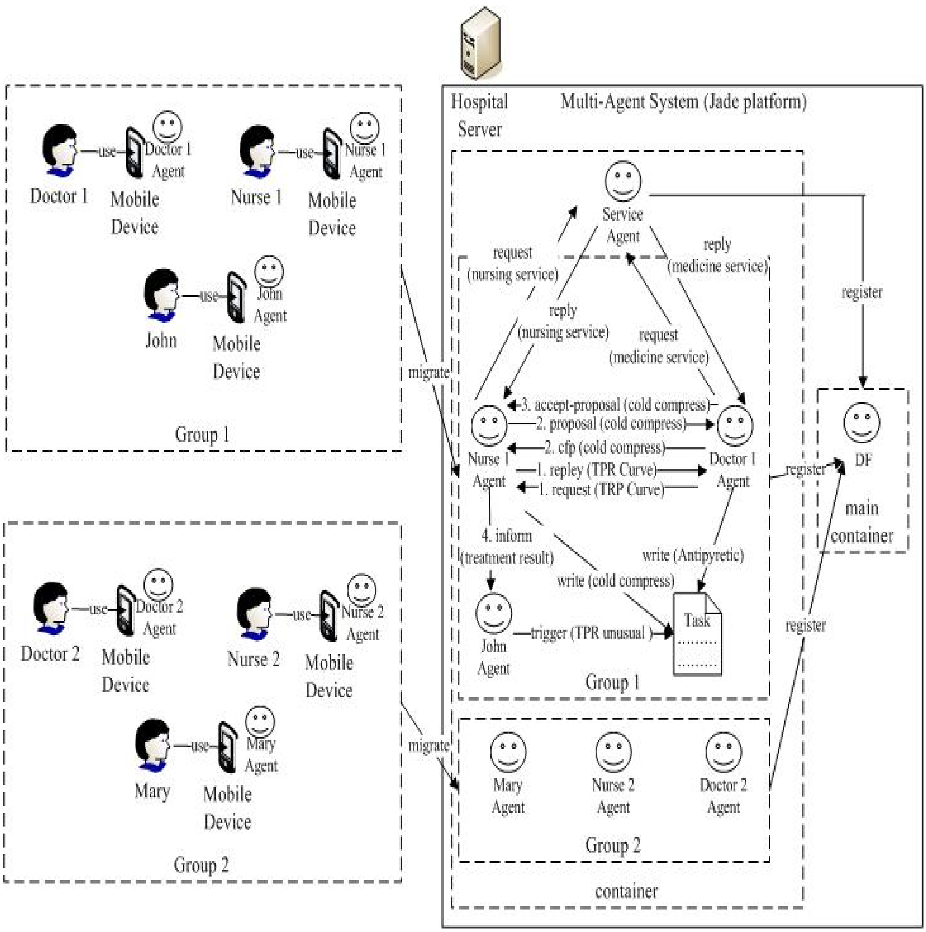

Fig. 3. Hospital Example

For the prescription, it requests service agent for medicine service. Notice that the medical service here is assumed to be a unique service in the hospital service agent, making it unnecessary to annotate the web service to turn it to a semantic web service. Now, the service agent replies to doctor 1 agent with the URI of the medicine service.

After that, doctor 1 agent will write the antipyretic named "Ibuprofen Suspension" into the task document. As for the call for proposal, nurse 1 agent proposes cold compress to doctor 1 agent. Doctor 1 agent decided to accept this proposal. After that, nurse 1 agent requests the service agent for nursing service. Next, the service agent will reply to nurse 1 agent with the URI of the nursing service. Nurse 1 agent will then write the cold compress procedure into the task document. Finally, it will inform John agent about the results of the treatment. A segment of the results is shown in Fig. 4.

```
…
<A0:Task rdf:ID="TPRUnusual">
   <A0:BelongTo>
      <A0:Domain rdf:ID=medicine/>
   </A0:BelongTo>
</A0:Task>
…
<A0:Action rdf:ID="MedicineService">
   <A0:ToFulfill>
      <A0:Task rdf:resource="#TPRUnusual"/>
   </A0:ToFulfull>
   <A0:hasContent> Perscribe antipyretic "Ibuprofen
                   Suspension"
   </A0:hasContent>
</A0:Action>
…
<A0:Action rdf:ID="NursingService">
   <A0:ToFulfill>
      <A0:Task rdf:resource="#TPRUnusual"/>
   </A0:ToFulfull>
   <A0:hasContent>
      Cold Compress Procedure
      …
   </A0:hasContent>
</A0:Action>
…
```

Fig. 4. Group Task Document

4 Conclusions

This work proposes the role-based mobile agent architecture. The expected advantages of it are:

1. The role-based mobile agent will be grouped into a specific group according to their roles and relationships. Through agent communication, all of the mobile agents in the group collaborate with each other to contribute the group task according to their role and knowledge.
2. The definition of role and group is defined in the role-based agent ontology that provides knowledge representation for the role-based mobile agent to facilitate collaboration among agents.

References

1. Tim, B.L., Hendler, J., OLassila, O.: The Semantic Web. In: Scientific American (May 2001)
2. Palmer, S.B.: The Semantic Web An Introduction (2001), `http://infomesh.net/2001/swintro/`
3. JADE Java Agent DEvelopment Framework (May 2003), `http://jade.tilab.com/`
4. Hendler, J.: Agents and the Semantic Web. Intelligent Systems 16(2), 30–37 (2001)
5. Yen, J., Fan, X., Sun, S., Hanratty, T., Dumer, J.: Agents with shared mental models for enhancing team decision makings. Decision Support Systems 41, 634–653 (2006)
6. Foundation for Intelligent Physical Agents, FIPA (2002), `http://www.fipa.org`
7. Liu, C.H., Lin, Y.F., Chen, J.Y.: Using Agent to Coordinate Web Service. The International Journal of Computer Science and Information Security (IJCSIS) 2(1), 18–25 (2009)
8. Francisco, G.S., Rafael, V.G., Rodrigo, M.B., Leonardo, C., Jesualdo, T.F.B.: An ontology intelligent agent-based framework for the provision of semantic web ser-vices. Expert Systems with Applications 36(2), 3167–3187 (2009)
9. Dirk, V., Stephan, L., Martijn, W., Michel, A.O., Jorg, M.H.: Facilitating Group-based Adaptation of Shared Workspaces using a Multi-Agent System. In: The Proceedings of the Context-Adaptive Interaction for Collaborative Work (CAICOLL 2010) Workshop (2010)
10. Wu, J., Hu, B., Zhang, J., Fang, D.: Multi-agent Simulation of Group Behavior in E-Government Policy Decision. Simulation Modeling Practice and Theory 16(10), 1571–1587 (2008)
11. Zhiyong, W., Thomas, T.: A Mobile, Intelligent Agent-based Architecture for E-Business. International Journal of Information Technology and Web Engineering 2(4), 63–80 (2007)