Hybrid Multi-Agent Architecture (HoCa) Applied to the Control and Supervision of Patients in Their Homes

Juan A. Fraile[1], Dante I. Tapia[2], and Miguel A. Sánchez[3]

[1] Centro de Proceso de Datos,
[2] R&D Department, Tulecom Group S.L.
Hoces del Duratón 57, 37008, Salamanca, Spain
[3] Escuela Universitaria de Informática,
Pontifical University of Salamanca,
Compañía 5, 37002 Salamanca, Spain
jafraileni@upsa.es, dante.tapia@tulecom.com, masanchezvi@upsa.es

Abstract. This paper presents a Hybrid Multi-Agent Architecture based on an Ambient Intelligence model, for the control and supervision of dependent environments. HoCa architecture incorporates a management system of alerts based on SMS and MMS technologies, and an automated identification, localization, and movement control system based on Java Card and RFID technologies. HoCa is independent from the programming language and operating system in that it is executable. The core of the architecture is formed by both deliberative agents and reactive agents that interact to offer efficient services. The architecture has been tested in a real environment and the results obtained are presented in this paper.

Keywords: Dependent environments, Ambient Intelligence, Multiagent Systems, Home Care.

1 Introduction

There is currently considerable growth in the development of automation technologies, such as home automation and Ambient Intelligence (AmI) [1] [14]. One of their principal objectives is to look after the user's well-being and obtain a more friendly, rational, productive, sustainable and secure relationship for users within their environments. Several architectures based on agent utilization have emerged thanks to the appearance of intelligent spaces and the integration of devices that are programmable via computer networks [15]. These have stimulated the development of ubiquitous computation [9], which is the most promising technological approximation for resolving the challenge of developing strategies that allow the early detection and prevention of problems in an automated environment.

The main objective of this paper is to define a hybrid Multi-Agent Architecture for the control and the supervision of open environments. It involves developing an architecture that allows automated identification, localization, alarms management and control of movement. The users who utilize the system in which this architecture is applied will be able to gain wireless access to all the information that they need to

E. Corchado, A. Abraham, and W. Pedrycz (Eds.): HAIS 2008, LNAI 5271, pp. 54–61, 2008.

perform their work. The novel innovation at the core of the architecture is a real time communication protocol that allows secure and rapid communication between the reactive agents and the system sensors. These reactive agents, whose response time is critical, are influenced by deliberative BDI agents, which are located inside the platform given that a very fluid communication already exists between them. Additionally, the architecture manages an alert or alarm system across the agents' platform specially designed to work with mobile devices. The alert system contains different levels of urgency. The alert level is determined by the deliberative agent who, depending on the alert level, then emits the alert to either a reactive agent or a deliberative agent.

The paper is organized as follows: The second section presents the problem that prompted this work. The third section presents the proposed architecture, and the fourth section gives the results and conclusions obtained after applying the proposed architecture to a real case in an environment of dependence.

2 General Description of the Problem

The use of intelligent agents is an essential component for analyzing information on distributed sensors [16] [18]. These agents must be capable of both independent reasoning and joint analysis of complex situations in order to be able to achieve a high level of interaction with humans [3] [4]. Although multi-agent systems already exist and are capable of gathering information within a given environment in order to provide medical care [12], there is still much work to be done. It is necessary to continue developing systems and technology that focus on the improvement of services in general. After the development of the internet there has been continual progress in new wireless communication networks and mobile devices such as mobile telephones and PDAs. This technology can help to construct more efficient distributed systems capable of addressing new problems [7].

Hybrid architectures try to combine deliberative and reactive aspects, by combining reactive and deliberative modules. The reactive modules are in charge of processing stimuli that do not need deliberation, whereas the deliberative modules determine which actions to take in order to satisfy the local and cooperative aims of the agents. The aim of modern architectures like Service Oriented Architecture (SOA) is to be able to interact among different systems by distributing resources or services without needing to consider which system they are designed for. An alternative to these architectures are the multi-agent systems, which can help to distribute resources and to reduce the centralization of tasks. Unfortunately the complexity of designing multi-agent architecture is great since there are not tools to either help programme needs or develop agents.

Multi-agent systems combine aspects of both classic and modern architectures. The integration of multi-agent systems with SOA and web services has been recently investigated [2]. Some investigators focus on the communication among these models, whereas others focus on the integration of distributed services, especially web services, in the agents' structure [5] [13] [17].

These works provide a good base for the development of multi-agent systems. Because the majority of them are in the development stage, their full potential in a real environment is not known. HoCa has been implemented in a real environment and not only does it provide communication and integration among distributed agents, services and applications, but it also provides a new method for facilitating the development of multi-agent systems, thus allowing the agents and systems to function as services.

HoCa implements an alert and alarm system across the agent's platform, specially designed to be used by mobile devices. The platform agents manage this service and determine the level of alert at every moment so that they can decide who will receive the alert and when. In order to identify each user, HoCa implements a system based on Java Card and RFID (Radio Frequency IDentification) microchip technology in which there will be a series of distributed sensors that provide the necessary services to the user.

3 Proposed Architecture

The HoCa model architecture uses a series of components to offer a solution that includes all levels of service for various systems. It accomplishes this by incorporating intelligent agents, identification and localization technology, wireless networks and mobile devices. Additionally, it provides access mechanisms to multi-agent system services, through mobile devices, such as mobiles phones or PDAs. Access is provided via wi-fi wireless networks, a notification and alarm management module based on SMS and MMS technologies, and user identification and localization system based on Java Card and RFID technologies. This system is dynamic, flexible, robust and very adaptable to changes of context.

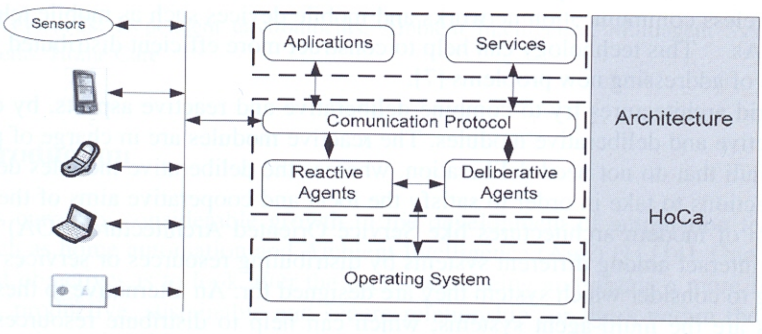

Fig. 1. HoCa Framework

HoCa architecture describes four basic blocks that can be seen in Figure 1: Applications, Services, Agents Platform and Communication Protocol. These blocks constitute the whole functionality of the architecture.

3.1 Agents Platform in HoCa

The agents platform is the core of the architecture and integrates two types of agents, each of which behaves differently for specific tasks.

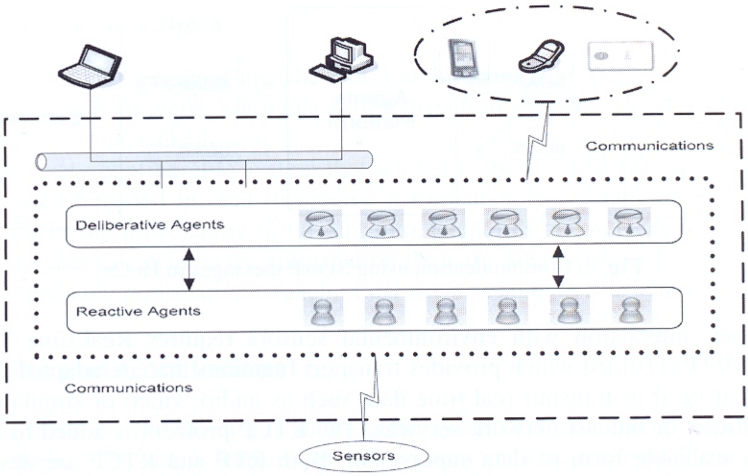

Fig. 2. Agents platform structure in the HoCa architecture

The first group of agents is made up of deliberative BDI agents, which are in charge of the management and coordination of all system applications and services. These agents are able to modify their behaviour according to the preferences and knowledge acquired in previous experiences, thus making them capable of choosing the best solution. Deliberative agents constantly deal with information and knowledge. Because they can be executed on mobile devices, they are always available for the users regardless of physical location.

The second group is made up of reactive agents. Most of the research conducted within the field of multi-agent systems focuses on designing architectures that incorporate complicated negotiation schemes as well as high level task resolution, but don't focus on temporal restrictions. In general, the multi-agent architectures assume a reliable channel of communication and, while some establish deadlines for the interaction processes, they don't provide solutions for limiting the time the system may take to react to events.

It is possible to define a real-time agent as an agent with temporal restrictions for some of its responsibilities or tasks [11]. From this definition, we can define a real-time multi-agent system (Real Time Multi-Agent System, RT-MAS) as a multi-agent system in which at least one of the agents is a real-time agent [6]. The use of RT-MAS makes sense within an environment of critical temporal restrictions, where the system can be controlled by autonomous agents that need to communicate among themselves in order to improve the degree of system task completion. In this kind of environments every agent requires autonomy as well as certain cooperation skills to achieve a common goal.

3.2 HoCa Communication Protocol

Communication protocol allows applications, services and sensors to be connected directly to the platform agents. The protocol presented in this work is open and independent of programming languages. It is based on the SOAP standard and allows messages to be exchanged between applications and services as shown in Figure 3.

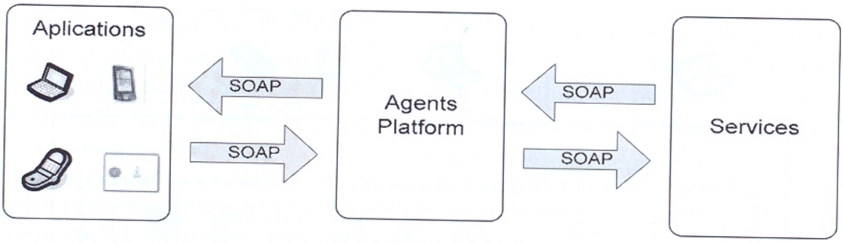

Fig. 3. Communication using SOAP messages in HoCa

However, interaction with environmental sensors requires Real-time Transport Protocol (RTP) [10] [6] which provides transport functions that are adapted for applications that need to transmit real-time data such as audio, video or simulation data, over multicast or unicast network services. The RTCP protocol is added to RTP, allowing a scaleable form of data supervision. Both RTP and RTCP are designed to work independently from the transport and lower network services. They are in charge of transporting data with real-time characteristics, and of supervising the quality of service, managing the information for all the entities taking part in the current session.

The communications between agents within the platforms follows the FIPA ACL (Agent Communication Language) standard. This way, the applications can use the platform to communicate directly with the agents.

3.3 Location and Identification System in HoCa

This system incorporates Java Card [19] and RFID [8] technologies. The primary purpose of the system is to convey the identity of an object or person, as with a unique serial number, using radio waves. Java Card is a technology that permits small Java applications (applets) to be run safely in microchip smart cards and similar embedded devices. Java Card gives the user the ability to program applications that can be run off a card so that it has a practical function in a specific application domain. The main features of Java Card are portability and security. The data are stored in the application and the Java Card applications are executed in an isolated environment, separate from the operating system and from computer that reads the card. The most commonly used algorithms, such as DES, 3DES, AES, and RSA, are cryptographically implemented in Java Card. Other services such as electronic signature or key generation are also supported.

RFID technology is grouped into the so-called automatic identification technologies. But RFID provides more information than other auto-identification technologies, speeds up processes without losing reliability, and requires no human intervention.

The combination of these two technologies allows us to both identify the user or identifiable element, and to locate it, by means of sensors and actuators, within the environment, at which time we can act on it and provide services. The microchip, which contains the identification data of the object to which it is adhered, generates a radio frequency signal with this data. The signal can be picked up by an RFID reader, which is responsible for reading the information and sending it, in digital format, to the specific application.

Table 1 presents the results obtained after comparing the HoCa architecture to the previously developed ALZ-MAS architecture [7] in a case study on medical care for patients at home. The ALZ-MAS architecture allows the monitoring of patients in geriatric residences, but home care is carried out through traditional methods. The case study presented in this work consisted of analysing the functioning of both architectures in a test environment. The HoCa architecture was implemented in the home of 5 patients and was tested for 30 days. The results were very promising. The data shown in Table 1are the results obtained from the test cases. They show that the alert system improved the communication between the user and the dependent care services providers, whose work performance improved, allowing them to avoid unnecessary movement such as travels and visits simply oriented to control or supervise the patient. The user identification and location system in conjunction with the alert system has helped to notably reduce the percentage of incidents in the environment under study. Moreover, in addition to a reduction in the number of incidents, the time elapsed between the generation of a warning and solution decreased significantly. Finally, due to the many improvements, the level of user satisfaction increased with the introduction of HoCa architecture since patients can live in their own homes with the same level of care as those offered at the residence.

Acknowledgements. This development has been partially supported by the Spanish Ministry of Science project TIN2006-14630-C03-03.

References

1. Anastasopoulos, M., Niebuhr, D., Bartelt, C., Koch, J., Rausch, A.: Towards a Reference Middleware Architecture for Ambient Intelligence Systems. In: ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications (2005)
2. Ardissono, L., Petrone, G., Segnan, M.: A conversational approach to the interaction with Web Services. In: Computational Intelligence, vol. 20, pp. 693–709. Blackwell Publishing, Malden (2004)
3. Bahadori, S., Cesta, A., Grisetti, G., Iocchi, L., Leone, R., Nardi, D., Oddi, A., Pecora, F., Rasconi, R.: RoboCare: an Integrated robotic system for the domestic care of the elderly. In: Proceedings of workshop on Ambient Intelligence, Pisa, Italia (2003)
4. Bahadori, S., Cesta, A., Grisetti, G., Iocchi, L., Leone1, R., Nardi, D., Oddi, A., Pecora, F., Rasconi, R.: RoboCare: Pervasive Intelligence for the Domestic Care of the Elderly. In: AI*IA Magazine Special Issue (January 2003)
5. Bonino da Silva, L.O., Ramparany, F., Dockhorn, P., Vink, P., Etter, R., Broens, T.: A Service Architecture for Context Awareness and Reaction Provisioning. In: IEEE Congress on Services (Services 2007), pp. 25–32 (2007)
6. Carrascosa, C., Bajo, J., Julian, V., Corchado, J.M., Botti, V.: Hybrid multi-agent architecture as a real-time problem-solving model. Expert Systems With Applications 34(1), 2–17 (2008)
7. Corchado, J.M., Bajo, J., Abraham, A.: GERAmI: Improving the delivery of health care. IEEE Intelligent Systems. Special Issue on Ambient Intelligence 23(2), 19–25 (2008)
8. Garfinkel, S., Rosenberg, B.: RFID: Applications, security, and privacy, pp. 15–36. Addison-Wesley Professional, Reading (2005)

9. Kleindienst, J., Macek, T., Seredi, L., Sedivy, J.: Vision-enhanced multi-modal interactions in domotic environments. IBM Tecnologías de Voz y Sistemas. República Checa (2004)
10. Jacobsen, V., Fredrick, R., Casner, S., Schulzrinne, H.: RTP: A transport Protocol for Real-Time Applications. RFC 1889. Lawrence Berkeley National Laboratory, Xerox PARC, Precept Software Inc., GMD Fokus (January 1996) (Accessed October 16, 1996), `http://www.connect.org.uk/teckwatch/cgi-bin/rfcshow?1889`
11. Julian, V., Botti, V.: Developing real-time multi-agent systems. In: Proceedings of the Fourth Iberoamerican Workshop on Multi-Agent Systems (Iberagents 2002), Málaga (2004)
12. Mengual, L., Bobadilla, J., Triviño, G.: A fuzzy multi-agent system for secure remote control of a mobile guard robot. In: Favela, J., Menasalvas, E., Chávez, E. (eds.) AWIC 2004. LNCS (LNAI), vol. 3034, pp. 44–53. Springer, Heidelberg (2004)
13. Ricci, A., Buda, C., Zaghini, N.: An agent-oriented programming model for SOA & web services. In: 5th IEEE International Conference on Industrial Informatics (INDIN 2007), Vienna, Austria, pp. 1059–1064 (2007)
14. Richter, K., Hellenschmidt, M.: Interacting with the Ambience: Multimodal Interaction and Ambient Intelligence. In: Position Paper to the W3C Workshop on Multimodal Interaction, July 19-20 (2004)
15. Rigole, P., Holvoet, T., Berbers, Y.: Using Jini to integrate home automation in a distributed software-system. Departamento de Ciencias Computacionales. Leuven, Bélgica (2002)
16. Tapia, D.I., Bajo, J., De Paz, F., Corchado, J.M.: Hybrid Multiagent System for Alzheimer Health Care. In: Rezende, S.O., da Silva Filho, A.C.R. (eds.) Proceedings of HAIS 2006. Ribeirao Preto, Brasil (2006)
17. Walton, C.: Agency and the Semantic Web. Oxford University Press, Oxford (2006)
18. Wooldridge, M.: An Introduction to MultiAgent Systems. John Wiley & Sons, Chichester (2002)
19. ZhiqunChen (Sun Microsystems). Java Card Technology for Smart Cards. Addison Wesley Longman. ISBN 0201703297