A Role-Based Agent-Oriented Approach to Medical Device Integration

Francesco De Mola
Università di Modena e Reggio Emilia, Italy

Giacomo Cabri
Università di Modena e Reggio Emilia, Italy

BACKGROUND

Monitoring heterogeneous medical devices is a key issue for efficient medical information management in high-quality health care delivery. Collecting vital signs and integrating data from various sources is very important in order to make precise diagnosis and provide effective emergency response. Current commercial rules and the scarce adoption of standard protocols are the first main obstacle to achieve a real integration.

We propose a work-in-progress, role-based, agent-oriented solution to overcome this obstacle: proper mobile agents are in charge of acquiring data from each kind of medical device (e.g., electrocardiograms, blood pressure, and oxygen saturation gauges), providing a uniform interface toward the business logic; in its turn, each medical device is provided by a vendor-specific role, embodying the capability to interface with the device by means of its own protocol. Thus, each time a new medical device requires the integration in a wider system, a generic mobile agent migrates to the device node; here, it assumes the role of the specific device manufacturer and starts delivering the acquired data hiding the heterogeneity of the device.

We will propose such a solution describing the implementation of a demonstrative prototype exploiting the RoleSystem infrastructure, which dynamically makes role capabilities available to the agents.

INTRODUCTION

Information technology can provide an interesting support to manage applications in health care scenarios. The tasks involved in health care exhibit wide heterogeneity and are often strictly related to one another. This demands the adoption of innovative and powerful paradigms, in particular to achieve device integration that is more dynamic and flexible than those usually adopted for traditional distributed systems.

Agents are software components that exhibit a high degree of autonomy (Jennings, 2001) and provide a useful paradigm to meet health care requirements in general (Nealon & Moreno); in the literature, we also find several approaches for medical applications (Lhotska & Stepankova, 2005; O'Donoghue & Herbert, 2006; Shnayder, Chen, Lorincz, Fulford-Jones & Welsh, 2005). This chapter aims to show a possible agent-oriented solution for the easy integration of medical devices facing heterogeneity by means of the concept of agent role (Cabri, De Mola, Quitadamo & Zambonelli, 2006). A first prototype, UbiMedic-RED (Role-Enabled Devices) has been developed on the basis of the proposed approach. In particular, the implemented system is based on the *UbiMedic framework* (De Mola, Cabri, Muratori, Quitadamo & Zambonelli, 2006), a context-aware middleware built on top of the multi-agent platform JADE (JADE, 2001). It provides several services of high and low (core services) levels to support medical and territorial emergencies in distributed contexts. The concept of role has been introduced in UbiMedic, thanks to the integration with *RoleSystem* (Cabri, Leonardi & Zambonelli, 2003b), an interaction infrastructure implementation allowing agents to dynamically assume roles and interact accordingly. Section Staring Points presents the starting points of our work: the UbiMedic framework and the Rolesystem infrastructure. The rest of the article proposes the UbiMedic-RED architecture based on the previously introduced systems, exploiting an application scenario.

STARTING POINTS

UbiMedic aims at building a distributed context-aware platform with mobile agents implementing helpful services for user applications in medical domain, focused on real-time monitoring and delivering vital signs and medical information. UbiMedic is an agent-based framework, designed to take into consideration the importance of portability even on small devices (e.g., PDAs or other limited user's terminals) (De Mola et al., 2006). The framework exploits the execution environment and the facilities provided by the JADE agent distributed platform. The latter is composed of several *Peripheral Containers,* representing different nodes with their own JVM, and a centralized *Main Container*, providing the basic facilities for the platform management. In particular, the *Directory Facilitator (DF)* provides a yellow page service useful for agent discovery and registration.

About monitoring and integrating physically distributed medical devices, the framework proposes a three-level decomposed approach related to three corresponding agents.

The main idea of each application accessing a medical device (e.g., an electrocardiograph or pulse oximeter sensor) is the definition of the following three kinds of agents (Figure 1): a *User Interface Agent* (UIA) in charge of managing user interactions by means of a more or less complex GUI providing all the suitable tools to end-user monitoring of the detected patient data; a *Physical Resource Interface Agent* (PRIA), which has to collect and make available the patient's vital signs measured by medical devices to requesters all over the platform; and a *Proxy Agent* (PA), an intermediate entity that includes the proper logic to process and filter the data collected by PRIA according to the specific medical device typology and to the particular goals of the application. The PA is also the logic unit where further intelligent and cooperative diagnostic capabilities can be included; for example, exploiting the agent social ability, the PA can compare the received data with medical traces coming from other devices or databases before returning to the UIA the processed diagnostic results.

Rolesystem is an interaction infrastructure (Cabri et al., 2003b) that implements the role interaction model of BRAIN (Cabri, Leonardi & Zambonelli, 2003a). It is completely written in Java and can be associated with the main agent platforms; in the following we consider JADE (2001) as the agent platform.

As shown in Figure 2, the Rolesystem infrastructure is divided into two parts: the upper one, which is independent of the agent platform; and the lower part, which is bound to the chosen agent platform. Our effort was in the direction of reducing the platform-dependent part.

In applications exploiting the Rolesystem infrastructure, agents can be seen by two points of view: they are both *subjects* of the role system and *agents* of the JADE platform. Accordingly, an agent is composed of two layers: the *Subject Layer*, representing the subject of the role system independent of the platform; and the *Wrapper Layer*, which is the JADE agent in charge of supporting the subject layer. For each context/environment, a specific agent called *Server Agent* is in charge of managing the roles and their interactions. It interacts with the wrapper layer of agents by exchanging ACL messages formatted in an appropriate way.

In the following sections, we will show how the UbiMedic framework has been combined with the RoleSystem infrastructure.

Figure 1. The three-level decomposition of UbiMedic for device monitoring

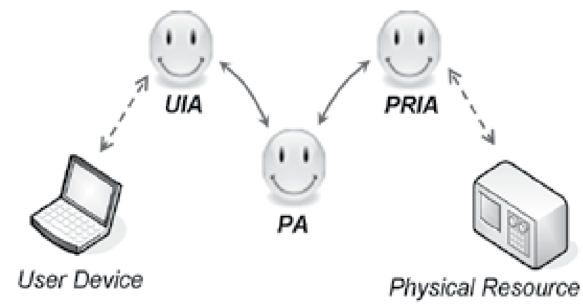
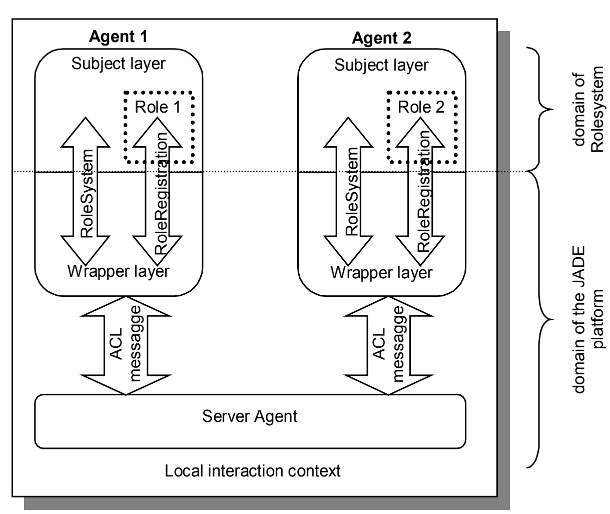


Figure 2. Domain separation in Rolesystem



Related Content