An Organisation-based Multiagent System for Medical Emergency Assistance *

Roberto Centeno[1], Moser Fagundes[1], Holger Billhardt[1], Sascha Ossowski[1]
, Juan Manuel Corchado[2], Vicente Julian[3], and Alberto Fernandez[1]

[1]CETINIA, University Rey Juan Carlos, Madrid
[2]Universidad de Salamanca
[3]Universidad Politecnica de Valencia
Spain

Abstract. In this paper we present a demonstrator application for a real-world m-Health scenario: mobile medical emergency management in an urban area. Medical emergencies have a high priority given the potential life risk to the patients. And the use of advanced applications that support the different actors involved can improve the usage of appropriate resources within an acceptable response time. The demonstrator is implemented using the THOMAS approach to open multiagent system based on an organisational metaphor. This metaphor is very suitable for capturing the nature and the complexity of the mobile health domain and, thus, it provides an appropriate mechanism for developing next-generation m-Health applications.

1 Introduction

In the healthcare domain the management of medical emergencies has a huge social impact given the immediate threat to a patient's life or long term health. Such extreme circumstances demand the usage of the appropriate resources within a limited response time in order to provide an efficient assistance.

The employment of advances on computer science to improve the provision of healthcare services is an area of significant economic and social potential. Recently, the notion of m-health has become prominent, focusing on applications that provide healthcare to people anywhere, anytime using broadband and wireless communications, as well as mobile computing devices [2].

In this work we apply an organisational approach to develop a prototype application for a next-generation m-health scenario: the coordination of the different actors (*patient*, *emergency centre*, *ambulances*, *hospitals*, etc.) involved in medical emergency management based on wireless mobile technology. The application is constructed on top of an abstract architecture specified in the THOMAS project (MeTHods, Techniques and Tools for Open Multi-Agent Systems) [7].

* The present work has been partially funded by the Spanish Ministry of Science and Innovation, grants TIN2006-14630-C03-02 (FPI grants program) and CSD2007-00022 (CONSOLIDER-INGENIO 2010).

The THOMAS platform uses the notion of organisation as a key abstraction to build MAS in open environments. It defines an abstract architecture, as well as a software platform, where agents interact according to norms and rules put forward by the organisational structure. Furthermore, the THOMAS platform allows for the integration of two well established research fields: Service-Oriented computing and Multiagent Systems. Thus, it provides a way to build Service-Oriented Multi-Agent Systems (SOMAS) – an approach proposed recently [5].

Our system is based on real-world data provided by the Medical Emergency Service of the Region of Madrid (SUMMA112). This service handles the medical emergencies, including the transportation of patients to hospitals. It counts with approximately 100 call operators, 36 medical doctors, 36 coordination technicians and 83 vehicles (ambulances and helicopters). Its emergency centre receives more than 1.200.000 emergency calls per year almost one call each 30 seconds, and around 60.000 of them are classified as situations of life risk.

The paper is organised as follows. Section 2 describes how the medical emergencies can be modelled as an organisation. Section 3 presents the application prototype designed to deal with these problems. Section 4 explains a case study chosen to illustrate how the application works. Finally, section 5 concludes and points out some future lines of work.

2 mHealth as an Organisation

A typical medical emergency assistance starts when a patient calls SUMMA112, asking for assistance. The call is received by an Operator, who gathers the initial data from the patient. Then she forwards the call to a Medical Doctor, who assigns the resources to attend the request according to its evaluation, however it does not mobilise the resource. Technicians do this task by assigning mobile units taking into account their availability, distance and time to reach the patient, type of resource (ambulances with different capabilities and available resources from other organisations, such as *Red Cross*, *Civil Protection*). Finally, according to the patient condition, she is transported to a hospital.

Analysing this typical scenario, we have determined basically the following actors which are involved along the whole process: *patients*, *coordinator staff*, *physicians*, *ambulances* and *hospitals*. These participants collaborate with each other in order to obtain their goals - to assist patients as fast as possible. Most of the time they act following some protocols/norms which have been determined by some authority. The quality of the whole system depends strongly on the efficiency and effectiveness of this collaboration, that is, on the mechanisms used to organise and coordinate the interactions among the participants.

In order to address both, the nature of the application domain and its high complexity, we propose to model the medical emergency assistance system as a multiagent organisation. In particular, following the framework proposed in [3], we have modelled a mHealth organisation as a particular type of organised multiagent system which is endowed with an organisational mechanism based on norms and roles. The participants in a medical emergency event will be represented by agents that join the organisation playing a particular role. Regarding intervention protocols, they will be modelled as norms that are shared by all participants – agents – within the organisation. We assume that norm compliance is effectively assured through the organisational mechanism. That is, in the realm of the framework in [3], the organisational mechanism used can be classified as a regulative coercive organisational mechanism.

We have identified the following basic roles within medical emergency assistance processes:

- **Patient:** agents playing role patient represent potential users of medical emergency assistance.
- **Hospital:** this role represents a hospital within the organisation.
- **Ambulance:** agents joining the organisation with the role ambulance are representing an ambulance vehicle together with the human resources assigned to it. In our current prototype, we do not differentiate between ambulance vehicles of different types – all emergency vehicles are assumed to offer the same services.
- **Emergency-Coordinator:** this agent is the main piece in the mHealth organisation. An agent playing this role is able to receive emergency calls from patients, and to find the "best" ambulance for each case, thus performing the high-level management of the emergency assistance procedure.

Whereas the roles *Patient, Hospital*, and *Ambulance* can be joined by more than one agent, the role *Emergency-Coordinator* can only be joined by one agent within the mHealth organisation.

Regarding the functional structure of the organisation, we have specified a set of norms that define the possible action patterns that can take place. We have used a very simple set of norms defining: *i) permissions*, and *ii) obligations*. Some examples for the used norms are the following:

- $\mathcal{N}_1$: OBLIGATION $isPlaying(a_i, Ambulance) \land isPlaying(a_j, Patient)$ EXECUTE $takePatient(a_i, a_j)$ WHEN $assignMission(a_i, a_j)$
- $\mathcal{N}_2$: OBLIGATION $isPlaying(a_i, Hospital) \land isPlaying(a_j, Patient)$ EXECUTE $beAdmited(a_j, a_i)$ WHEN $hasMedicalEmergency(a_j)$
- $\mathcal{N}_3$: PERMISSION $isPlaying(a_i, Patient) \land isPlaying(a_j, Emergency\text{-}Coordinator)$ EXECUTE $emergencyCall(a_i, a_j)$

Finally, the identified entities in the process of medical emergency assistance can be classified into two classes: *i)* active entities whose actions are driven from objectives and that possess a certain degree of freedom in their actions, and *ii)* entities that actually act as providers of certain services (e.g. find an ambulance, an hospital, etc.). The former - active entities - will be matched as agents populating the organisation, playing the defined roles. Non-active entities will be considered as resources which can be accessed by agents within the organisation.

3 System Architecture

The aim of our work is to build a system that provides decision support and value added services to the participants who are involved in medical emergency assistances - physicians, patients, hospitals and coordination staff. Following the design proposed in section 2, we have implemented a prototype of the mHealth organisation. In that prototype, the mHealth organisation has been populated with one type of agent for each possible role in the organisation. Thus, we have implemented a Patient agent, Hospital agent, Ambulance agent and a SUMMA agent. The latter plays the role *emergency-coordinator*. All agents have been implement as norm-compliant, that is, they always obey the existing norms.

Regarding the non-active entities, we have modelled them as web services because they only provide a static functionality without following intentions or objectives. We have designed a new role and a new type of agent – *Services-Provider* – which is in charge of providing the web services that implement the non-active entities. Agents which play this role have the obligation to answer when they receives a request of a web service call.

The web services used to implement the resources are the following:

- *Emergency Centre Finder*: this service finds the responsible emergency coordinator for a given location. The result of the invocation of this service will be the identifier of an agent which is playing the *emergency-coordinator* role – in our system this role will be played by the SUMMA112 agent.
- *Medical Record Store*: it allows agents to store and retrieve patients' medical history information trough an user name and password.
- *Ambulance Finder*: This service is an internal service for the emergency-coordinator centre, thus, it can be accessible only by the SUMMA112 agent. The service is able to find an ambulance for a particular patient, taking into account information such as his position, his symptoms, etc.
- *Hospital Finder*: this service is similar to ambulance finder service. It is only accessible for agents which are playing the ambulance role because of they have the obligation to find a hospital when a patient needs to be admitted. The service is able to find a hospital taking into account the patient's position, symptoms, etc.

The main advantage of implementing the non-active entity as web services, specially the ambulance and hospital finder, is that we can easily compare different strategies to assign ambulances and hospitals to patients, simply providing a new implementation of these web services. In the current version both, ambulance finder and hospital finder, return the closest ambulance and the closest hospital (able to treat the diagnosed disease) to the patient's location.

We have developed our prototype on top of the *THOMAS* platform [1]. This platform has been chosen because, on one hand, it uses the organizational paradigm and allows to define the multiagent system based on roles and norms that regulate the agents' behaviours. On the other hand it allows the integration of open multiagent systems and web services. The web services have been specified using the standards OWL-S and WSDL. This makes it possible to use standard techniques for service discovery, composition, etc.

3.1 THOMAS Platform

The THOMAS platform implements an abstract architecture for open multiagent systems based on the organisational paradigm. The agents have access to the functionality offered by THOMAS through a range of services included in several modules. The main components of THOMAS are the following:

– *Service Facilitator (SF)*: this component provides simple and complex services to the active agents and organisations. Basically, its functionality is like a yellow page service and a service descriptor in charge of providing a green page service.
– *Organisation Manager Service (OMS)*: it is mainly responsible of the management of the organisations and their entities. It allows to create and to manage organisations.
– *Platform Kernel (PK)*: it maintains basic management services for an agent platform; it is in charge of communication between agents, etc.

Besides the possibility to structure an application through the concept of organisation, THOMAS allows for a seamless integration of multiagent systems with web services. Web services can be included and can be registered in the service facilitator by means of the services provider agents.

3.2 Spatial Environment Simulator

In order to evaluate our system we have created a spatial environment simulation tool in which the mHealth organisation can be embedded. The environment simulator is an independent module that captures key features of our problem domain. It has two fundamental functions: *i)* it allows to set up agents (and thus, organisations), and *ii)* it recompiles information about the actions that take place in the organisation.

The agents interact with the simulator through a request protocol, where the agents request the execution of an action. The simulator executes that action if the agent plays the required role. The THOMAS platform provides the information about the agents and their roles. In our case, three roles are represented in the physical environment: *patient*, *hospital* and *ambulance*. Once the simulation starts, the environment is constantly updated. Basically, the simulator controls the location of each agent along the time as well as their displacements.

Figure 1 shows the set of available actions for each role as well as the simulator design. *Patients* have only one available action which consists of informing their position in the map. The *hospitals* are able to accept or refuse patients brought by the ambulances, as well as to release the patients. Finally, the *ambulances* are able to catch patients that ask for help, release patients at the

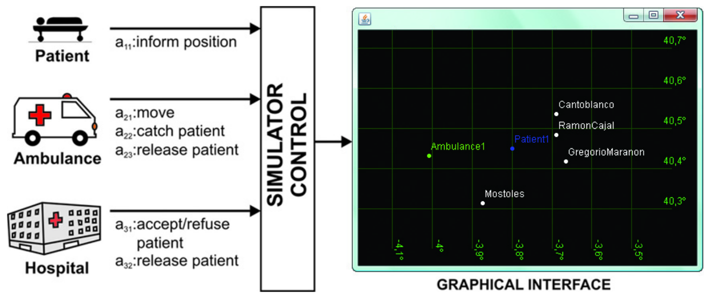

Fig. 1. Simulator design and set of available actions for each role

hospitals, and move around the physical environment by specifying a destination position (complex routes can be composed by specifying a sequence of positions). The simulation tool is composed of a control layer which handles the agent interactions, and a graphical interface that presents the current state of the system. In Figure 1 the graphical interface shows a simulation with one ambulance (*Ambulance1*), one patient (*Patient1*), and four hospitals (*Cantoblanco*, *GregorioMaranon*, *RamonCajal*, *Mostoles*).

4 Case Study

In this section we present an example of a concrete use case that has been simulated using the mHealth organisation.

In this scenario, Bob is a British tourist visiting Madrid. He suddenly feels a strong pain in his chest and as it is his first time in Madrid, he does not know what he should do. He has his personal agent which joins the mHealth organisation as patient. Immediately, the patient looks for a service that provides the functionality of looking for a medical emergency centre. The services provider agent, using its web service, returns the *SUMMA112's* identifier and a form for specifying personal data and symptoms.

Once *Bob* has selected his symptoms the form is sent to the *SUMMA112* agent. It uses its classification method to make a decision about *Bob*'s case. The preliminary diagnosis says that Bob may be suffering a hearth attack. Therefore, he is classified as a patient that needs an ambulance urgently. The *SUMMA112* agent uses its ambulance finder service to select the most adequate ambulance to take *Bob*. In that case, the closest ambulance to *Bob* is *ambulance3* so the *SUMMA112* agent assigns a new mission to take *Bob*. The ambulance agents have the obligation to take a patient when he needs a medical emergency assistance. Thus, *ambulance3* accepts the mission and goes to take *Bob* immediately.

While *ambulance3* is going to take *Bob*, the *ambulance* agent uses the service "medical record store" to retrieve *Bob*'s medical history. This is an important characteristic because it helps the physicians in the ambulances to make a preliminary decision before actually seeing the patient.

When *ambulance3* arrives at *Bob*'s location, the physician performs an "in situ" diagnosis and decides to which hospital *Bob* should be taken. The *ambulance* agent in *ambulance3* supports this decision by using the *hospital finder service*. Taking into account Bob's location, symptoms and diagnosis, as well as the available units in the medical centres, the *Fuenlabrada hospital* is selected because is the nearest hospital which is able to treat heart attacks. Since hospital agents have the obligation to admit patients if they have the capacity, *Bob* is admitted at the *Fuenlabrada hospital*. The hospital agent informs the *SUMMA112* agent about this fact and the *SUMMA112* agent closes the case.

In this scenario we have populated the organisation with 31 agents playing the hospital role, 60 agents playing the role ambulance and one SUMMA112 agent which is playing the emergency-coordinator role. The information of the hospitals (location, bed capacities, etc.) correspond to the hospitals located in the Region of Madrid. Figure 2 shows the sequence of interactions that take place within the organisation in this scenario.

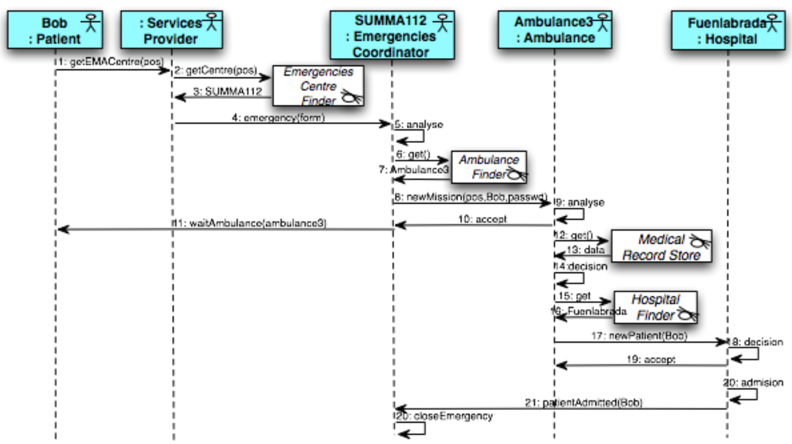

Fig. 2. Case study sequence diagram

5 Conclusions

In this work we have presented an application of a multiagent system based on organisational principles in the healthcare domain: a prototype application for the support the processes involved in medical emergency assistance. The application provides value added services and decision support to the different actors (patient, emergency centre, ambulances, hospitals, etc.) involved in emergency assistance processes. The organisational paradigm provides a straightforward approach to model the functioning of complex application domains by means of roles and norms. We developed our application as an instantiation of the THOMAS abstract architecture, which explicitly supports multiagent organisations. Furthermore, the THOMAS platform facilitates the integration of non-active entities as web services, following a SOMAS approach. In order to show the system dynamics, we have implemented a case study based on real data provided by the Madrid Medical Emergency Service *SUMMA112*.

Related work, such as [6], has been developed to support medical emergencies processes. In this paper, the authors deal with the problem of selecting the best ambulance to be assigned to an emergency call. In contrast to this work, our approach concentrates on the whole assistance process and tends to provide support to all involved actors. Other related work deals with coordination issues in disaster management in general ([8]), where different teams such as polices, fire brigades, ambulances, etc. have to coordinate each other to assist emergencies. In our work we focus on medical emergencies, in a similar way to the work presented in [4], taking into account just hospitals and ambulances.

In the future we plan to work on mechanisms that may optimise the global utility of the system (e.g., response time and cost) by studying different mechanisms to select the ambulances and hospitals. Finally, we also plan to extend the functionality of the application in order to support other processes of the medical emergency service (e.g., inter-hospital transportation of patients, management of bed capacities of hospitals, etc.).

References

1. E. Argente, V. Julian, and V. Botti. Mas modelling based on organizations. In *9th Int. Workshop on Agent Oriented Software Engineering*, pages 1–12, 2008.
2. César Cáceres, Alberto Fernández, and Sascha Ossowski. Cascom - context-aware health-care service coordination in mobile computing environments. *ERCIM News*, (60):77–78, 2005.
3. R. Centeno, H. Billhardt, R. Hermoso, and S. Ossowski. Organising mas: A formal model based on organisational mechanisms. In *24th Annual ACM Symposium on Applied Computing (SAC2009), Hawaii, USA, March 8-12*, page to appear, 2009.
4. Anna Ciampolini, Paola Mello, and Sergio Storari. A multi-agent system for medical services synergy and coordination. In J. Nealon, U. Cortes, J. Fox, and A. Moreno, editors, *International ECAI 2004 workshop on Agents applied in health care*, pages 38–46, 2004.
5. Michael N. Huhns. A research agenda for agent-based service-oriented architectures. In Matthias Klusch, Michael Rovatsos, and Terry R. Payne, editors, *CIA*, volume 4149 of *Lecture Notes in Computer Science*, pages 8–22. Springer, 2006.
6. Beatriz López, Bianca Innocenti, and Dídac Busquets. A multiagent system to support ambulance coordination of urgent medical transportation. *IEEE Intelligent Systems*, 23(5):50 – 57, 2008.
7. S. Ossowski, V. Julián, J. Bajo, H. Billhardt, V. Botti, and J.M. Corchado. Open mas for real world applications: An abstract architecture proposal. *Proc. XII Conference of the Spanish Association for Artificial Intelligence (CAEPIA)*, 2(1):151–160, 2007.
8. Robocup Rescue. http://www.robocuprescue.