Aingeru: an Innovating System for Tele Assistance of Elderly People

Alberto Tablado, Arantza Illarramendi, Miren I. Bagüés*, Jesús Bermúdez, and Alfredo Goñi **

University of the Basque Country, Donostia, Spain

Abstract. In this article we present the main features of AINGERU, a system that provides a new kind of tele assistance service. The purpose of developing AINGERU has been to overcome the main constraints that tele assistance services nowadays present: they are passive (they react only when the user requires it); their coverage is limited; and, they do not monitor automatically vital signs.

By contrast, AINGERU, by using PDAs (Personal Digital Assistants), wireless communications and semantic web and agent technologies, provides a service with the following features: it is active (it reacts automatically in the face of anomalous situations); it can work anywhere and anytime; and, it can monitor vital signs and generate an alarm when necessary. Furthermore AINGERU permits physicians and relatives concerned with AINGERU users to consult data about them through the web.

The main focus of the paper is on presenting the architecture of AINGERU and three aspects that are used in its development: agent technology, semantic web techniques and web services which constitute the pillars to obtain the new tele assistance features provided by AINGERU.

1 Introduction

It is a reality that the number of elderly people is growing and so, the number of these people that live alone. Although many of them can manage themselves, it is also true that they feel a kind of defenceless situation and many of them sign a contract with companies that offer a tele assistance service.

These tele assistance services, although they accomplish an interesting and necessary function, present the following main constraints: they are passive, their coverage is limited and they do not monitor vital signs.

Having as a goal to overcome the previous constraints, we are developing the system AINGERU[1], that is our proposal for a new way of tele assistance

---

* This work is supported by a grant of the Basque Government.

** All authors are members of the Interoperable DataBases Group, online at http://siul02.si.ehu.es. This group is part of the Department of Computer Languages and Systems, at the University of the Basque Country in Spain. This work is mainly supported by the University of the Basque Country and Diputación Foral de Gipuzkoa (cosupported by the European Social Fund).

[1] AINGERU is the word in the Basque language for expressing the notion of guardian angel.

Tablado A., Illarramendi A., I. Bagüés M., Bermúdez J. and Goñi A. (2004).
Aingeru: an Innovating System for Tele Assistance of Elderly People.
In *Proceedings of the 1st International Workshop on Tele-Care and Collaborative Virtual Communities in Elderly Care*, pages 27-36
DOI: 10.5220/0002681100270036

for elderly people. Hence, apart from supporting the functionalities provided by current tele assistance services, Aingeru also offers: an *active assistance* by using stationary agents that behave in the face of anomalous situations without a direct intervention of the user; an *anywhere and anytime* assistance by using wireless communications and PDAs; and the *monitoring of vital signs* by using sensors that capture the values of those signs and feed a decision support system that analyzes them and generates an alarm when necessary.

Concerning related works, we classify tele assistance systems in two groups. In the first group, we include those systems that provide limited coverage, such as existing tele alarms. The main features of those systems are the following: they use wired phone communications to contact tele assistance services providers, their coverage is restricted usually to the user's home and their activation is triggered by the user, generally using a button. Therefore, they do not support an *active* anywhere and anytime assistance. In the second group we include more advanced systems. Some of them focus their efforts on offering an active assistance (e.g. TeleCare [1]) while others use PDAs and take advantage of wireless communications to provide anywhere and anytime assistance (e.g. doc@HOME [2] and TeleMediCare [3]). In the majority of the last kind of systems, PDAs are used only as intermediary elements and their goal is merely reduced to transmit data from sensors to a central computer where data analysis is made. They do not take advantage of the ability of the PDAs to carry out a certain pre-analysis before sending data to the central computer. Notice that wireless communications are slow, expensive and unstable so, analysis made in the PDA can save communication costs and can permit to detect anomalies earlier. Other related works use different "devices" to provide anywhere and anytime assistance (e.g. Sensatex [4] and SILC [5]).

Aingeru goes one step further by providing not only anywhere and anytime assistance using wireless communications and PDAs, but also a high quality assistance. The combined use of three technologies: agents, semantic web and web services constitutes the core that differentiates Aingeru from other related works.

In the rest of this paper we first present a brief introduction to the global architecture of Aingeru. Then, we explain the main features of: the agents that take part in Aingeru, the semantic web tools developed, and the web services. Next, we show some performance features and we finish with the conclusions.

2 Overview of the Global Architecture of Aingeru

The goal of this section is to present briefly the main features of the Aingeru architecture. As can be observed in Fig. 1, there are five different types of components:

– **User PDA**. Each person monitored by Aingeru carries a PDA. Its main goal is twofold, first to monitor the user and then, when special circumstances require, to be the link between the person and the center (Control

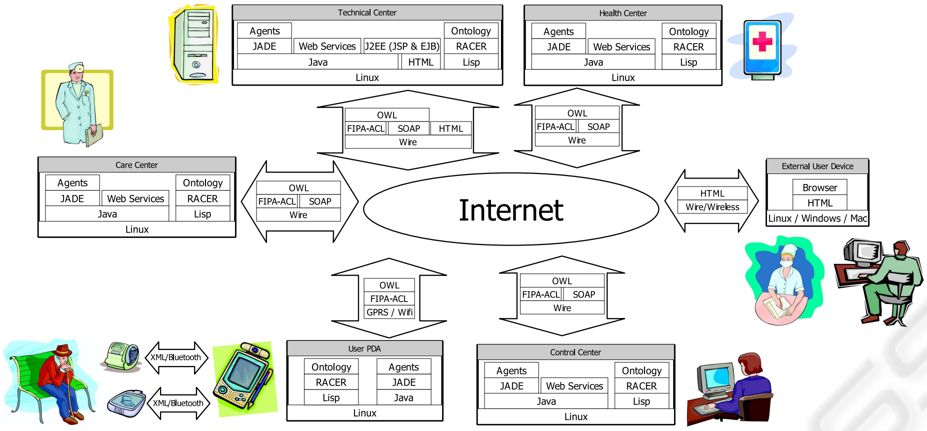


Fig. 1. Architecture of Aingeru .

Center) responsible for monitoring her/him. Three are the basic functions of PDAs: *manual activation of an alarm* (when the user feels bad, or something is happening in his environment and he wants to notify it, he presses the button that appears in the interface and an alarm is activated in the Control Center), *automatic activation of an alarm* (the PDA receives data sent by sensors related to pulse, mobility, etc., and an agent situated at the PDA analyzes these data in order to activate an alarm when an anomalous situation is detected) and *agenda services* (PDAs provide their users with classical agenda services, such as remembering when they have to visit the doctor, etc.). In each PDA two ontologies are incorporated: MedOnt and OperOnt (they are described in Sect. 3) and several agents.

- **Control Centers** are the centers in charge of monitoring people. Their main task is to react in the presence of user alarms and to take the adequate actions. In a computer of each Control Center are incorporated: the OperOnt ontology, several agents, web services and a database that contains data about monitored users. The OperOnt ontology permits the agents to communicate at a semantic level.
- **Care Centers** are the public health centers for primary assistance. Part of the Aingeru application must run in Care Centers in order to provide Aingeru users with new functionalities such as: accessibility of physicians to data stored in the user PDA, direct insertion of medical appointments in the PDA, etc. In a computer of each Care Center are incorporated: the OperOnt ontology, several agents, web services and a database that contains clinical data.
- **Health Centers** correspond to hospitals. In order to interoperate with Aingeru system, in a computer of each Health Center are incorporated: the OperOnt ontology and some deployed web services.
- **Technical Center**. The goal of this center is the development and support of the Aingeru application. For example, there is an agent in this center,

in charge of sending new software releases and activating them, in the corresponding component of the global architecture.

Concerning communication aspects among the Aingeru components, in Fig. 1 we can observe the different levels that are used. At the physical level, the wired or wireless communication appears. At the transport level, FIPA is used for inter agent communications and SOAP for web services. At the application level, agents and web services communicate through terms described in ontologies written in OWL[2].

3 Distinctive technologies of Aingeru

3.1 Agents in Aingeru

As we have mentioned in the introduction, one pillar of Aingeru is the use of agents. There are software agents working in the various elements that take part of its architecture (i.e., in the PDA, Control Centers, etc.). Moreover, all kinds of agents that take part in the Aingeru system are described in the OperOnt ontology (explained in Sect. 3.2), with the goal of making them interoperable with external applications.

For the implementation of those agents, we have selected the JADE Agent System platform [8, 9] for the following reasons among others: *standards compliant*, it adheres to the latest FIPA standards; *lightweight*, as some parts of Aingeru must run in PDAs (JADE has versions that can run on J2ME/MIDP compliant devices, like phones and watches); *flexibility*, it allows us to use different transport drivers to convey messages among agencies; Moreover, we make use of the authentication mechanism supported by JADE and the SSL protocol to transmit sensitive data among agents. If additional security is required, message content is also encrypted.

There are other systems that also promote applying agents in Health Care [1, 10, 11, 12, 13]. However, as far as we know, they do not put special emphasis on combining the agent technology with the use of PDAs.

3.2 Semantic Web Techniques

Two formal logic-based ontologies that are machine understable have been defined for the Aingeru system: MedOnt and OperOnt (see [14] for more details).

Medical Ontology (MedOnt). To offer a high quality assistance is one of the aims of the Aingeru system. This means that Aingeru not only will generate an alarm when the user requires it, but also when the system autonomously detects anomalous situations. Thus, its behaviour is reactive when the user's health is in danger. In our opinion, the latter feature is a step forward in the field of tele

---

[2] OWL [6, 7] is an ontology representation language from the Semantic Web forum.

assistance applications. In order to accomplish the goal of detecting anomalous situations, two components are necessary: first, sensors that capture users' vital signs values such as temperature, pulse, etc. and then, a decision support system that analyses the captured values. Knowledge for the decision support system is represented by a formal domain ontology called MedOnt. Ontology terms describe states of illnesses that elderly people can suffer from as well as states generated by the values captured by the sensors.

So far, to build the Aingeru prototype, we deal with a "toy" MedOnt ontology that we have built with the collaboration of physicians (see Fig. 2). For example, in this ontology we can observe the `Pulse>120` term. This term describes that the pulse of the user is above 120 beats per minute.

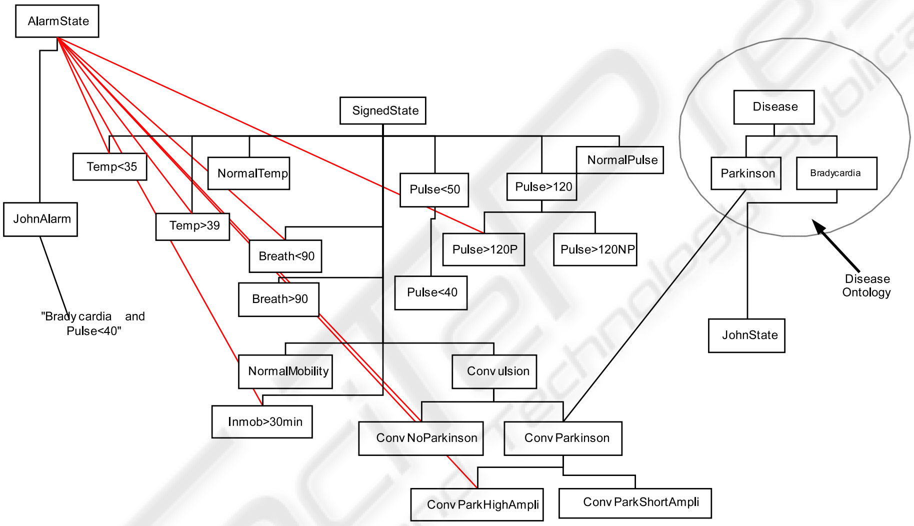


Fig. 2. Graphic representation of a fragment of a personalized MedOnt.

The `JohnAlarm` term illustrates the desirable customization of the MedOnt ontology loaded into each user's PDA. The `JohnAlarm` term is a kind of `AlarmState` that subsumes the potential anomalous situations of the user John. We consider customization a very interesting feature because every single person has peculiarities added to general symptoms.

There are also some terms in the MedOnt ontology that describe situations combining signs detected by sensors with states of diseases. For example, the term `Convulsion` describes the situation when the Mobility Sensor reports shaking over a certain threshold and `ConvParkinson` describes the state of the kind of `Convulsion` of a person who suffers from `Parkinson` disease.

Operational Ontology (OperOnt). Another goal that we planned to reach when developing the Aingeru system was to have a semantic representation of the agent communications framework. Such a formal representation facilitates: (i) the interoperability of Aingeru with other related systems, (ii) the understanding of Aingeru features by external actors, and (iii) the Aingeru evolution.

So far, the OperOnt ontology[3] is basically divided into three interrelated areas of descriptions: (i) actors, which interact using various kinds of messages, (ii) messages with different purposes and dealing with different kinds of contents and (iii) subjects representing the kinds of message contents. Next we outline each area, but we want to stress that there are axioms in the OperOnt ontology that describe their interrelationships.

- Actors. We divide this area into human agents, software agents and Aingeru web services. Human agents are classified into two groups: on one hand Aingeru users and, on the other hand all those people who are concerned with the user assistance, from sanitary people to relatives. Software agents are described taking into account their location and their goals (for instance, whether they work in a PDA or in a computer, if they are attending a sensor or interacting with an ontology, and so on). Web services are described on the basis of their functionality.
- Messages. Considering the standards-compliant principle in Aingeru , we have included descriptions of messages according to their functionality in FIPA [15] protocols. Message descriptions are based on three FIPA protocols: FIPA Request Interaction Protocol, FIPA Subscribe Interaction Protocol and FIPA Query Interaction Protocol. Kinds of messages that appear in OperOnt are: `Agree`, `Cancel`, `Failure`, `Refuse`, `Query`, `Subscribe` and `Inform`.
- Subject of Messages. This area describes terms about the subject or the topic concerning the message. For example, `InputOutput`, `Location`, `Urgency`, `Emergency`, `Sensor`, `Hospital`, `Appointments`, `Medicines` and `Dump` describe different subjects of messages.

Both ontologies — MedOnt and OperOnt — are specified in OWL to take advantage of semantic web technology and its richer primitives for class and property descriptions. Furthermore, OWL supports datatype descriptions using XML Schema [16], so we are able to deal with properties with integer values or with structured values such as HL7[4] [17] rows.

Moreover, for the reasoning process we selected the RACER [18] system, which covers all the OWL constructors we need; in particular, it deals with reasoning about individuals of classes, which is crucial in our framework.

---

[3] See `http://siul02.si.ehu.es/Aingeru/OperOnt.owl` for an OWL specification of the OperOnt ontology.

[4] HL7 is the first health care data interchange standard.

3.3 Web Services

One interesting feature that can be incorporated to the tele assistance services is the possibility of allowing external persons (relatives, physicians) to consult data about monitored persons (only authorized users will be allowed to consult) through Internet. The goal of this feature is accomplished in Aingeru by web services that have been developed using JAX-RPC and specified by WSDL. As examples of those web services we can mention:

- Vital Signs: A web service exported by the Care Center. It provides information about user vital signs in real-time. Authorized actors (physicians, for example) can obtain data about the current values of the sensors that monitor the user.
- Location: A web service exported by the Technical Center. It provides information about the current location of the user. Location information is managed by the Location Agent residing in the User Agency.

Moreover, the architecture of Aingeru allows the incorporation of public web services such as those that provide information about sports, weather forecasts, etc. Web Services are described in the OperOnt ontology in order to be discovered and accessed by different actors.

4 Feasibility Features of Aingeru at Work

As Aingeru is being implemented, we cannot test it yet in a real environment. For this reason, in this section we show some tests that we did in order to check its feasibility in particular: (i) influence of the bandwidth in the time needed to respond to alarms and (ii) capacity of the PDAs to support all the requirements defined in Aingeru . Moreover, we compared, in one concrete scenario, local data analysis versus remote data analysis.

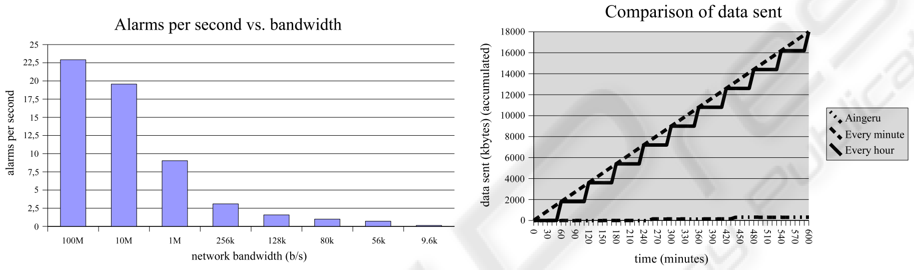
Influence of the Bandwidth in the Time Needed to Respond the Alarms. In this test, we show how network communications severely reduce the number of alarms that can be managed simultaneously (see Fig. 3.a). On one hand, nowadays GPRS wireless communications provide around 80 kb/s bandwidth, so each PDA can send only one alarm per second[5]. On the other hand, a Control Center connected to internet using a 10 Mb/s link can handle 20 alarms per second. Therefore, the Control Center will be able to cope with 20 users simultaneously, a number greater than that supported by current tele assistance services.

---

[5] This frequency of alarms widely exceeds the actual behavior of the users of tele assistance services as experts told us.

Capacity of the PDAs. In the PDA we need to run in a Java virtual machine, Linux, RACER reasoner and JADE agent platform. Except for RACER (there is not a version for PDAs yet[6]), all the others run in our test. More than this, we also tested JADE, and we verified that it could manage up to 60 agents in the PDA. Taking into account that so far we have developed only eleven agents for the PDA, we concluded that Aingeru will be able to operate in the PDA.

As RACER cannot be tested on a PDA, we run it in a normal computer and we registered how much memory it consumed. In our tests, RACER never went beyond 10 MB of RAM and, even when all other software was running in the PDA, about 40 MB of RAM remained free. This means that RACER could be run in the PDA.



(a) (b)

Fig. 3. Data sent by different models and Aingeru .

Local Analysis versus Remote Analysis. Finally, we compared our approach of making local data analysis at the PDA with the other approaches that advocate external data analysis. The comparison is based on communication costs (how many bytes must be sent) in the following scenario: a user has to be monitored via a 2-lead ECG sensor. It sends 250 samples of a byte every second. In short, this configuration sends 30 KB per minute. And, to perform this task, we defined three configurations. The first one was Aingeru , equipped with a wireless ECG sensor and a wireless thermometer. The second and third ones had the same sensors, but did not perform local analysis of the data received. These two differ in that they sent data at different intervals: one (A) every minute and the other (B) every hour.

Figure 3.b shows the data sent as time goes on. We simulated an alarm at 260 minutes (four hours and twenty minutes) and 440 minutes (after seven hours). The first and most relevant fact is that the line that lies over X axis, which represents Aingeru , sends hardly any data, even though Aingeru is configured to send five minutes of ECG data within the alarm notification.

---

[6] We are working in collaboration with the authors of RACER to port it to the PDAs.

Related to communications costs, Aingeru needs to send data only twice, in contrast to other configurations. The A configuration sends 30 KB of data every minute (600 times) and the B configuration, 1,8 MB every hour (10 times) for a total of 18 MB. Aingeru only sends 300 KB in the same interval and offers better assistance.

Furthermore, we analyzed the latency, that is, how long it takes since the risk situation arises until the system knows about it. In A or B, the average time is half of the time between data arrival, i.e., half a minute for A or half an hour for B. Aingeru can report the alarm in less than a second (only the time needed to send the alarm over the network, because software components have response times much slower). To finish, we want to mention that savings communication costs implies also savings in battery consumption, which is relevant when working with PDAs.

5 Conclusions

In this article we have presented the Aingeru system, which main contributions can be summarized under two different perspectives: one related to the application domain; and, the other related to the technical issues.

Concerning the application domain, Aingeru goes one step further in the tele assistance service by providing: (i) anywhere and anytime assistance using PDAs and wireless communications; and, (ii) a *high quality assistance* reacting in the face of anomalous situations, mainly due to the use of agents and semantic web technology.

With respect to technical issues, for the development of Aingeru we have built: (i) agents that are specialist in different tasks, that can be executed in the PDA and in other servers and that can communicate among them at the semantic level using OperOnt ontology; (ii) two ontologies that allow sharing knowledge, medical and operational, with other systems; and (iii) web services that bring Aingeru closer to physicians and relatives concerned with the users of Aingeru .

So far we have got only a limited prototype and as future work we plan to work in the following directions: (i) defining a mechanism that allows us to do a semiautomatic customization of the domain ontology MedOnt, (ii) analyzing the interest of changing the agent platform to another one that supports agent mobility and (iii) incorporating the WSAI technology to improve the integration of agents and WebServices.

6 Acknowledgements

The authors would like to thank Volker Haarslev and Ralf Möller for their help in carrying RACER to PDAs.

References

[1] (2004, Feb.) Telecare, a multi-agent tele-supervision system for elderly care. [Online]. Available: http://www.uninova.pt/~telecare

[2] (2004, Feb.) doc@HOME. [Online]. Available: http://www.docobo.com

[3] (2004, Feb.) TelemediCare. [Online]. Available: http://www.telemedicare.net

[4] (2004, Feb.) Sensatex. [Online]. Available: http://www.sensatex.com

[5] (2004, Feb.) Supporting Independently Living Citizens. [Online]. Available: http://www.fortec.tuwien.ac.at/silcweb/SILC.htm

[6] *OWL Web Ontology Language Reference*, World Wide Web Consortium W3C Candidate Recommendation, Aug. 2003. [Online]. Available: http://www.w3.org/TR/owl-ref

[7] (2004, Feb.) Web-ontology (webont) working group. [Online]. Available: http://www.w3.org/2001/sw/WebOnt/

[8] (2004, Feb.) JADE. [Online]. Available: http://jade.cselt.it

[9] F. Bellifemine, A. Poggi, and G. Rimassa, "Developing multi-agent systems with JADE," in *7th International Workshop on Agent Theories, Architectures, and Languages*, 2000, pp. 85–99, Boston, MA.

[10] H. Mouratidis, G. Manson, P. Giorgini, and I. Philp, "Modelling an agent-based integrated health and social care information system for older people," in *15th European Conference on Artificial Intelligence*, 21 July 2002.

[11] S. L. Mabry, S. J. Kollmansberger, T. Etters, and K. L. Jones, "IM-Agents for patient monitoring and diagnostics," in *15th European Conference on Artificial Intelligence*, 22 July 2002.

[12] A. Moreno and D. Isern, "Offering agent-based medical services within the AgentCities project," in *15th European Conference on Artificial Intelligence*, 22 July 2002.

[13] L. M. Camarinha-Matos and H. Afsarmanesh, *Advances in Automation, Multimedia and Video Systems, and Modern Computer Science.* WSEAS Press, 2001, ch. Virtual Communities and Elderly Support, pp. 279–284, ISBN: 960 8052 440.

[14] M. I. Bagüés, J. Bermúdez, A. Illarramendi, A. Tablado, and A. Goñi, "Using ontologies in the development of an innovating system for elderly people tele-assistance," in *Proceedings of the International Conference on Ontologies, Databases and Applications of SEmantics*, ser. Lecture Notes on Computer Science, vol. 2888. Springer-Verlag, Nov. 2003, pp. 889–905, Sicily, Italy.

[15] (2004, Feb.) Foundation for Intelligent Physical Agents. [Online]. Available: http://www.fipa.org

[16] XML Schema Working Group, *XML Schema Part 0: Primer*, World Wide Web Consortium Std., 2 May 2001.

[17] (2004, Feb.) Health Level Seven. [Online]. Available: http://www.hl7.org

[18] R. Möller and V. Haarslev. (2004, Feb.) RACER: Renamed ABox and Concept Expression Reasoner. [Online]. Available: http://www.fh-wedel.de/~mo/racer/