

Multiagent Approach for Cancer Automated Registration

Teodora Sanislav*, Dorina Căpățină*, Adina Stoian**, Ioan Stoian*

*IPA – R&D Institute for Automation, Cluj-Napoca, Romania, (e-mail: {teodora, dorina, stoian}@automation.ro)

**Hospitals Leman France, (e-mail: adastoian@yahoo.com)

Abstract: Recording and monitoring patients with cancer in distributed databases, generally called registries, present a priority in cases in which these diseases have become a major public health problem. The cancer registries contain a comprehensive collection of information about patients that enter under the incidence of the certain disease, extended to population level and they are essential in order to improve the quality of care and survival for patients. The aim of our research is to develop a multiagent system with the goal of automating the process of registration using one of the agent-oriented software engineering methodologies. The INGENIAS methodology is selected based on more evaluation criteria: concepts and properties, modeling language, model-related, and organizational. The analysis-to-implementation procedure of the multiagent system, following the INGENIAS approach, consists in the presentation of five basic concepts: organization, agent, tasks/goals, interaction, and environment. Following the methodology a prototype is implemented. The prototype is tested in a medical centre responsible with the cancer regional registry management and the experimental results show the potential of the multiagent approach.

Keywords: agents, multiagent system methodology, automated registration, agent-oriented software engineering.

1. INTRODUCTION

Cancer registry is an information system for the collection, management and analysis of people with a diagnosis of cancer. Cancer registration is a continuous and systematic process of collecting data on disease' occurrence and reported characteristics in order to control the impact of the disease at the population level. The main function of the registry is to maintain a database of all new cases appeared in a specific population, to collect and document, continuously and systematically, from different sources of information, data concerning patient characteristics and clinical and pathological features of disease.

1.1 Registration methods

Methods of new cases of cancer registration in registries were classified into manual (active and passive) and automated methods. Active registration (data collected directly from source) involves registry staff that goes through primary source records, or obtains copies of necessary documents and abstracts the information in the required form. Passive registration is based on medical staff that offers facilities care for patients with cancer, which must complete reporting files, or send out copies of tickets. In practice a combination of these two methods is very useful.

Cancer automated registration is the process through the information is received, validated, collated, selected and used to update the database at registries level. Automated registration resolves multiple pieces of information about a person into concise, distinct patient records by eliminating overlapping and duplicate information and by selecting the

“best information” for each data item so that accurate analyses can be performed (MN-PATRL). Automated registration does NOT mean that there aren't any “manual” steps. However, for a process to be considered “automated”, each stage of the process of registration should be able to work without any “manual” intervention. This capability is solved by a predefined set of algorithms or rules, requiring the availability of multiple electronic data sources (source records).

1.2 Automated versus manual registration

The benefits of an automated registration are significant. Foremost is the ability to consistently apply collection rules to all cases. When collecting cases manually, the results can be different based on the reviewer's: human experience; method of cases registration; knowledge of special ascertainment or coding rules within the registry.

Unusual situations are especially prone to different outcomes when collecting records manually.

The first benefit of automated registration instead the manual process consists in the assurance that all medical cases will be processed consistently. A second benefit of an automated registration is the economy of human resources and time financial and timesaving. While implementation and maintenance costs are involved in automating the case registration process, it is much less than the ongoing cost and time involved in the manual one. A third benefit is the ability to systematically monitor the effects of changes in collection rules (MN-PATRL).

The automated registration also has some disadvantages. There isn't any commercially available software capable to automatically collect medical records. That means that registries have to be designed and implemented on request (expensive and time consuming operation). Additionally, the automated registration of cancer cases represents an area where constant verifications and improvements need to be made. Also, certain decisions cannot be automated without potentially compromising the quality of medical data.

1.3 Automated registration and multiagent systems

The automated registration of cancer cases implies some requirements, such as: verification of the source record data accuracy and consistency, grouping all source records for an individual patient together (record linkage), identification of records that require consolidation and long term availability of source records.

These requirements can be satisfied by a multiagent system (MAS) approach. MAS are computational systems in which several agents cooperate to achieve some task that would otherwise be difficult or impossible for a single agent to achieve. Agents within MAS communicate, cooperate, and negotiate with each other to find a solution to a particular problem. An agent (Russell, et. al., 2003) is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actions. An agent is characterized by: the architecture part, or the agent's behaviour - the action performed after any given sequence of perceptions and the program part, or the agent's built-in part - the internal functionality of the agent. An artificial intelligent agent should be endowed with an initial (built-in) knowledge and with the capability of learning. The learning capability ensures the agent's autonomy, i.e., the capability of deducing his behaviour from its own experience.

2. MEDICAL ELECTRONIC DATA SOURCES

Medical documents taken into account for automated registration of cancer cases were analyzed in order to implement the ontology of the information system. Most of these documents exist in electronic form, but their data structures are more complex and more related than required registration data.

For this reason, the export of only required fields for registration of the patient from medical data structures in "xls" tables represents the best solution. Medical documents taken into account are: forms for reporting cases of cancer, Diagnosis-Related Group (DRG) records, statistical bulletins of decease, and forms from pharmacies, laboratory analysis results, smear results, pathological results, and institutional registries. A special attention was given of the main notification components which are the form for reporting a case of cancer, because the notifications contain all the data needed for the registration process. For these medical electronic components, their registration has been made in parallel with different coding on the nature of the disease in various formats, depending on the medical entity that provide

the forms. A web application to support direct data entry in an "xls" file is developed. This file is scanned by the agents in order to automatic collect the registries data. This web-based application called "Medical data retrieval" is working on a multiuser manner and it was developed using PHP, JavaScript, HTML, MySQL and AJAX and it is accessible through Internet browsers. The web application performs the coding of the patients' passport data, takes and validates the information on each disease-specific health data, patients' vital status and medical entity emitting the registration, respecting the rules of default correlation.

Another category of medical information that gives information for the cancer registries are ultrasound, X-ray, CT, MRI, cytological medical images that are not effectively stored in the registry, because they don't give a definitive information about the presence of the disease.

Yet for the prospective studies it would be of interest that it would be a way for patients from the cancer registries to go back to the support imagistic investigations that were made.

The most modern techniques for the management of medical images correlated with the rest of available „anywhere and anytime" patient data are based on the Picture archiving and communication system (PACS).

PACS completely reengineer medical images practices by enabling images to be electronically viewed, virtually anywhere on a clinical computer. PACS enables ubiquitous availability of images resulting in improved clinical care and productivity throughout the healthcare enterprise (Dreyer, et. al., 2006).

The following sections of the article describe the development of the MAS for automated registration of the cancer cases using an available agent-oriented software engineering (AOSE) methodology.

3. DEVELOPMENT DETAILS

The implementation of the MAS, that generates and manages the cancer registries, involves the selection and use of an AOSE methodology in the preliminary stages of the development (analysis and design) in order to improve the quality of the software.

3.1 AOSE methodologies

The AOSE methodologies are used to design a system composed by several entities with heterogeneous beliefs, desires and intentions. They contain sufficient abstractions to model and support agents and MASs, which are organized in a society of agents playing roles and exchanging information following predefined protocols (Henderson-Sellers, et. al., 2005) and some of them accomplish all the phases of the life cycle of any MAS application. The life cycle starts with a planning of the requirements, the analysis and design detail of the performance of the system, the implementation translates the design details into the functions and procedures that are tested to guarantee the correctness.

Nowadays, there are several AOSE methodologies, classified in two main families named extensions of object-oriented methodologies and knowledge engineering-based methodologies. The selection of an AOSE methodology depends on the MAS requirements. Before selecting one of them for the cancer registries management a comparison should be performed based on the following evaluation criteria, grouped into four main groups: concepts and properties, modeling language criteria, model-related criteria, and organizational criteria (Gómez, et. al., 2007). Table 1 presents the comparison results of the AOSE approaches that could be used in a real implementation (GAIA (Zambonelli, et. al., 2003), PASSI (Cossentino, 2005), INGENIAS (Pavón, et. al., 2005), MaSE (DeLoach, 2004)).

Table 1. AOSE Methodologies Comparison

	GAIA	PASSI	INGENIAS	MaSE
<i>Concepts and properties</i>				
Autonomy	+	++	++	+
Communication	++	++	++	+
Cooperation	+	+	++	+
Adaptability	-	-	+	-
Pro-activity	++	-	++	--
<i>Modelling language</i>				
Formalisation	++	-	++	-
Abstraction	+	+	+	-
<i>Model-related</i>				
Coverage	A/D	A/D/I	A/D/I	A/D/I
Complexity	-	n.a.	++	-
<i>Organizational</i>				
Open systems	--	+	++	+
Topology	+	-	++	-
Social norms	-	-	+	-
Notation: ++: Very high; +: High; ≈: Medium; -: Low; ---: Very low; <i>n.a.</i> : Not available; <i>A/D/I</i> : Analysis/Design/Implementation.				

The INGENIAS methodology skims the evaluation criteria and that's why it was selected to facilitate the live cycle of cancer registration system.

3.2 *INGENIAS methodology*

The INGENIAS methodology guides the developer in the process of analyzing, designing and implementing MASs. The most prominent features of this methodology are: it allows an agent developer to analyze and design MASs; it permits to describe agents in various level of abstraction; it is possible to detail the environment where the agents act; it allows the definition of conversations and messages to be exchanged; it provides a tool that facilitates the implementation of projects as well as its learning (IDK – Ingenias Development Kit) (Isern).

INGENIAS proposes five basic concepts: *organization* (defined as the framework where agents, resources and tasks get together to achieve some goals), *agent* (defined as a program that follows a rational behaviour and process knowledge), *tasks/goals* (description of the relationship among goals and tasks), *interaction* (defined as the exchange of information or requests between agents), and *environment* (identified by all systems with which the MAS interacts) (Isern).

3.3 Using INGENIAS methodology to develop cancer registries

The main requirements needed for the MAS, that generates and manages the cancer registries, were described in the first chapter of this article.

Analysis phase -The analysis phase supposes the study of the main requirements and the extraction of the main entities and their relationships with the environment and between them. Two models should be created in this stage: organization and environment models.

The *organization model* is the most complex and interesting model of this stage. The tree of the organization model has the base element called cancer registries at the root level, which includes three groups defined according to the chronic diseases for which the registration process must to be achieved: Cancer, Diabetes, and Peripheral arterial disease. Five agent types (Collector Agent, Personal Agent, Decision Agent, and Knowledge Management Agent) belong to each group (*OHasMember*) and their tasks satisfy the creation and the management of cancer registries. The Collector Agent interacts (*EResourceBelongsTo*) with various medical applications for monitoring the patients with cancer, from where extracts the data required for the generation of a historical database representing the background for the cancer registries. Figure 1 presents the organizational model.

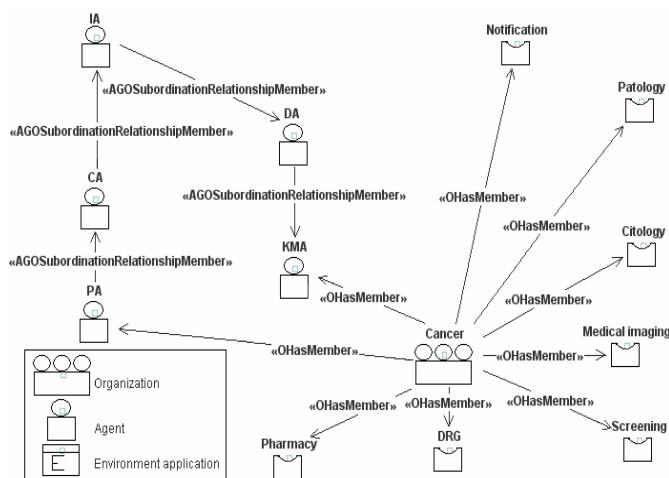


Fig. 1. The organization model

The medical applications, which interact with the MAS system for the management of cancer registries, compose the *environment model*. These applications are residents in various medical centers and were developed in various programming environments. Their main goal is to store the cancer patients' investigations in databases which can be exported to "xls" files. For example, the Collector Agent must go through "xls" records collected from the notifications of a new cancer case, the pharmaceutical recipes, the pathological laboratories reports, the death certificates, and the screening reports in order to gather the information necessary for cancer registry.

Design phase - This phase refines the models identified during the analysis stage and its results are agents' models, models of tasks and goals, decomposition of workflows, and mental state of the agents.

The automated registration of cancer cases into registries needs the following classes of agents:

- **Collector Agent (CA).** It is a mobile software agent responsible with the gathering of relevant medical data from various source records in a temporary warehouse.
- **Personal Agent (PA).** It is an interface software agent responsible with the management of the human user and the system interface interaction. It provides to collector agent favorable migration conditions in order to satisfy data storage to temporary warehouse.
- **Interpretative Agent (IA).** It is a software agent designed for formatting and encoding the source records, obtained from collector agent. It guarantees the data transmission for the historical database (DB) insertion.
- **Decision Agent (DA).** It is an intelligent software agent having the following responsibilities: it saves the data received from the interpretative agent into DB that provides long term availability of source records. Using this database, and its knowledge base (KB), the agent identifies the records that require consolidation and executes the record linkage. Based on these information it decides whether or not to update the registries.
- **Knowledge Management Agent (KMA).** It is a software agent that further analyzes the data from the cancer registries in order to provide various types of reports.

In order to satisfy their goals, the classes of agents are responsible with the tasks presented in Table 2.

Table 2. Agents' Tasks

Multiagent architecture	
Agent type	Tasks (Behaviour)
CA	Extracting data from heterogeneous medical sources Verification of medical data source records vital elements accuracy and consistency Creating files with erroneous data Transferring correct data in a temporary warehouse and their origin labeling
PA	Checking the current configuration of the system Creating a container for CA CA migration and CA behavioral parameters transmission
IA	Data retrieval from temporary warehouse Checking the temporary stored data formatting / coding DA activation indicating its type of action
DA	Data retrieval from IA into DB Medical data codification using KB Record linkage using KB Insert / update the cancer registries
KMA	Records retrieval from cancer registries Reports execution depending on users' requests

The *agent model* describes each agent, its tasks, goals, and played roles. In figures 2 and 3 there are presented tasks/goals model for Interpretative and Decision Agents. Figure 4 shows the interaction between the mentioned agents. The interaction diagram between IA and DA shows for each messaging who is the initiator of the conversation (*Initiate*, *UInitiate*), who engage in conversation (*IColaborates*,

UIColaborates) and previous relationships between conversational actions (*UIPrecedes*). The conversation acts between IA and DA are the following types: *propose*, *inform*, *acknowledge*, *accept-proposal*.

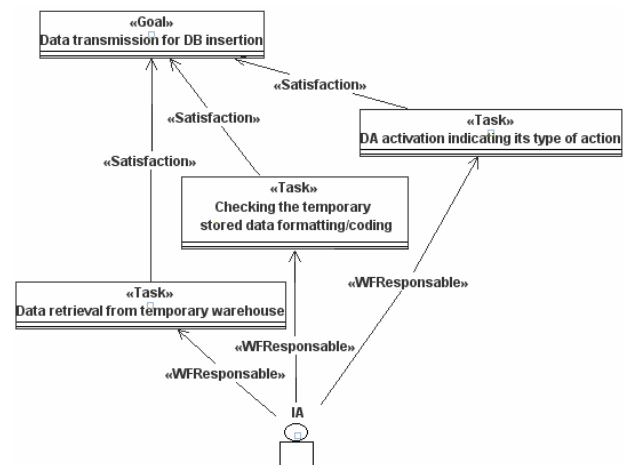


Fig. 2. The tasks/goals model of IA

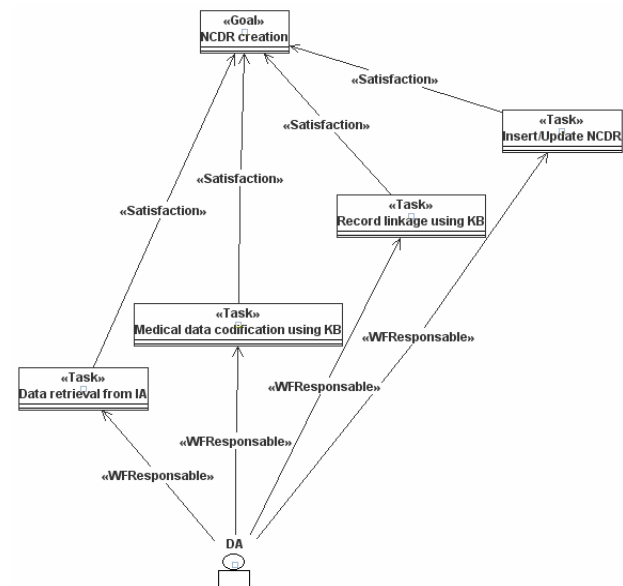


Fig. 3. The tasks/goals model of DA

One of the most important tasks of the DA consists in medical data codification, such as obtaining morphological and topographic codes (according to the International Classification of Diseases for Oncology - ICD-O) from natural language descriptions provided by medical data sources. From a practical point of view, the automatic codification is a very complex problem, due to several reasons: the existence of several different information for the same patient and the same tumor; the records contain not only useful data, but also many other "noisy" words; the records do not contain the expressions given in the dictionary from ICD-O; the order in which the words appear in a record does not coincide with that from they appear in the dictionary. Our approach indicates a possibility of automatic construction of the morphologic code using Logical Analysis of Data (LAD). The essence of LAD consists of: the

detection of patterns and the selection of groups with such patterns which collectively are capable of correctly classifying all known observations (Lupsa, et. al., 2010).

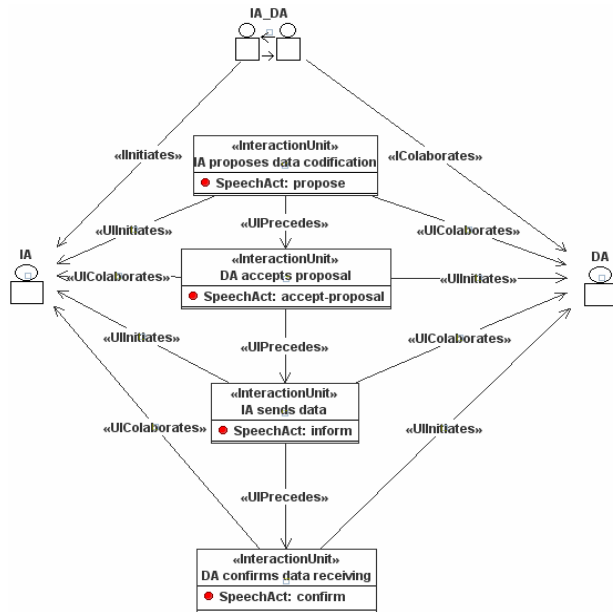


Fig. 4. The interaction between IA and DA

LAD, the input information is an archive of past observations S . To define the mathematical model, each element of S is identified by an n -dimensional vector having as components the values of n attributes. To each element in S we attach a so-called "classification" (positive-0, or negative-1). This classification generates two disjoint sets S^+ and S^- , included in $\{0,1\}^n$, such that each element in S has a unique representation in exactly one of the sets S^+ and S^- . Therefore, set S may be seen as a pair (S^+, S^-) . An archive (S^+, S^-) can be represented by a Boolean function φ , i. e. a function $\varphi: (S^+, S^-) \rightarrow \{0,1\}$ such that $\varphi(s)=0$, if $s \in S^+$ and $\varphi(s)=1$, if $s \in S^-$. One of the ultimate goals of LAD is the determination of an extension of the function φ , f , with the domain $\{0,1\}^n$ set, such that $f(s)=\varphi(s)$, for all $s \in S$, and $f(s)=0$, if and only if s is positive. In our case, the set S is divided in subsets, with different importance and, for every subset, a φ function is defined, obtaining a modified LAD, called "successive LAD" method (SLAD). The LAD method is applied successive on the same record, but with patterns in different sets. The way of constructing these sets, the order in which they are used in SLAD and some possible modifications are described in the (Lupsa, et. al., 2010) chapters "the SLAD Method", "Implementation of SLAD Method". The practical results of the SLAD method implementation according to automated codification of the morphological and topographic codes are presented in the Results chapter – at formatting / coding level.

Implementation phase - This phase converts into source code the models obtained in the design stage. The source code generation is one of the most interesting issues of

INGENIAS. The IDK has two modules to generate a complete documentation in HTML as well as a prototype using the agent-oriented programming language JADE (Bellifemine, et. al., 2007). Even though the internal codifications used during the analysis and design phases are independent of the final language, JADE is a good alternative to translate all specifications into a programming language known by all the developers (Isern).

In order to implement the agents distributed hardware architecture has been defined. System hardware architecture consists of a computers network with two distribution levels bounded by geographical area: regional and national level.

At regional level there is a JADE platform that contains a main container, named *Main Container 1* (the kernel of the platform). This container records instances of the runtime JADE environment represented by *Container 1*, *Container 2*, ..., *Container m* that must be registered through the *Main Container*. These instances are located at the medical entities level and they provide the data sources needed to introduce the records in the registries. *Container 1*, *Container 2*, ..., *Container m* host a number of active agents such as PA, IA, CA, whose role is to collect, format and encode the data from the available data sources, and then to send them to the *Main Container* where the DA is hosted. The AMS (Agent Management System) and DF (Directory Facilitator) are JADE platform specific agents and represent the authority within the platform. These agents provide the services needed by multi-agent platform agents to reach their goals. The multi-agent platform is complemented at the geographical level with a special container, *Container m+1*, which hosts two kind of agents: KMA and PA. These agents provide the statistical results obtained by postprocessing data methods. The links with the historical and knowledge regional level databases is done through facilitator agents (DF) from *Main Container 1*, *Main Container 2*, ..., *Main Container n*.

At national level there is a JADE platform with a *Main Container n+1* and a subordinate *Container* through which the data are collected and processed. These containers host the agents that manage the national platform necessary services and decide which data will be entered in registries: AMS, DF, DA, respectively agents that perform statistics required by users: KMA and PA. The link with the historical and knowledge national level databases is done through the facilitator agent (DF) hosted by the *Main Container n+1* (Caire, 2007; Bellifemine, et. al.).

4. RESULTS

The MAS for recording and management of the cancer, whose design and implementation was described in Section 3, is being tested in a medical center responsible with the cancer regional registry management.

Figure 5 presents the registry data quality assessment from two points of view: automated (left column) and manual (right column) registration. In each column the operations performed are detailed, the rate of incorrectness and also the incomplete data are specified. We mention that the automated

registration operations are performed automatically by the classes of agents described above.

A sample of 280 random cases has been coded manually and automatically, independently, 55% of cases have been correctly coded automatically. Causes of disagreement are mainly due to a lack of description accuracy in sources of information and also problems with the typewriting in the pathological reports. Some errors are due to the algorithms implemented in the software that are being reviewed.

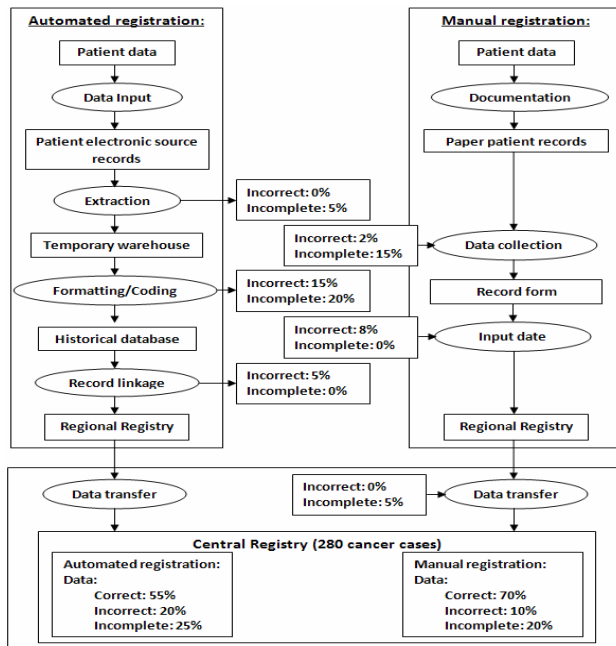


Fig. 5. The type and percentage of errors occurring in different stages of data registration

The figure presents the advantage of the automated registration approach which although decreases slightly the rate of the correct records, it significantly decreases the human effort involved in the registration management process.

The data rejection rates will decrease due to the improvement of the formatting/coding algorithms and the self learning capacity of the system.

5. CONCLUSIONS

This article tried to present the development stages of a multiagent system for automated data registration in cancer registries, using an AOSE methodology that provides developer support in the analysis, design and implementation phases. The obtained framework is very useful to organize a complex MAS and changes in the organization or in the agents' roles can be implemented more easily than in an *ad hoc* implementation. The experimental results have shown the potential of the MAS approach in the cancer automated registration process.

As future work we propose:

- to define an ontology having a role in the communications between agents and security issues

- to improve the data formatting/coding process and also the record linkage
- to document the functionalities of the final software in terms of agents, communication protocols, tasks and goals.

ACKNOWLEDGMENT

This paper is supported in the framework of CRONIS project (11003/2007), financed by Romanian Government, through PN II Programme.

REFERENCES

- Bellifemine, F., Caire, G., and Greenwood, D. (2007). *Developing Multi-agent Systems with JADE*, John Wiley & Sons. Ltd.
- Bellifemine, F., Caire, G., Trucco, T., Rimassa, G., Mungenast, R. *JADE Administrator's Guide*, <http://jade.tilab.com/tutorials>.
- Caire, G. (2007). *JADE Programming for Beginners*, <http://jade.tilab.com/tutorials>.
- Cossentino, M. (2005). *Agent-Oriented Methodologies*, Idea Group, ch. *From Requirements to Code with the PASSI Methodology*, pp. 79–10.
- DeLoach, S.A. (2004). *Methodologies and Software Engineering for Agent Systems, Multiagent Systems, Artificial Societies, and Simulated Organizations*, Springer US, ch. *The MaSE Methodology*, pp. 107–125.
- Dreyer, K.J., Hirschorn, D.S., Thrall, J.H., Mehta, A., Editors. (2006). *PACS – A Guide to the Digital Revolution*, ISBN: 0-387-26010-2, Springer Science and Business Media Inc., Second Edition.
- Gómez, C., Isern, D., and Moreno, A. (2007). *Software Engineering Methodologies to Develop Multi-Agent Systems: State-of-the-art*.
- Henderson-Sellers, B., and Giorgini, P.E., Editors. (2005). *Agent-Oriented Methodologies*, Idea Group.
- Isern, D., Gómez-Alonso, C., and Moreno, A. *Methodological Development of a Multi-Agent System in the Healthcare Domain*, <http://deim.urv.cat/~itaka/CMS/images/pdf/disern/isern08c.pdf>.
- Lupsa, L., Chiorean, I., and Neamtiu, L. (2010). *Use of LAD in Establishing Morphologic Code*, Proceedings of 2010 IEEE AQTR, ISBN: 978-1-4244-6722-8, Tome III, pp. 219–224.
- MN-PATRL: *Technical Specifications for Automated Record Consolidation*, <http://www.health.state.mn.us/divs/hpcd/cdee/mcss/>.
- Pavón, J., Gómez-Sanz, J.J., and Fuentes, R. (2005). *Agent-Oriented Methodologies*, Idea Group, ch. *The INGENIAS Methodology and Tools*, pp. 236–276.
- Russell, S., and Norvig, P. (2003). *Artificial Intelligence: A modern approach*, Prentice Hall, Second Edition.
- Zambonelli, F., Jennings, N., and Wooldridge, M. (2003). *Developing Multiagent Systems: The Gaia Methodology*. ACM Transactions on Software Engineering and Methodology, Vol. 12, No. 3, pp. 317–370.