Open MAS for Real World Applications: An Abstract Architecture Proposal

S. Ossowski[3] , V. Julián[1], J. Bajo[2], H. Billhardt[3], V. Botti[1], J. M. Corchado[2]

[1] Departamento de Sistemas Informáticos y Computación,
Universidad Politécnica de Valencia, 46022, Valencia, Spain
[2]Departamento de Informática y Automática,
Universidad de Salamanca, Plaza, Salamanca, Spain
[3]Universidad Rey Juan Carlos Campus de Mostoles, E-28933 Madrid, Spain
{sascha.ossowski,holger.billhardt}@urjc.es,{jbajope,corchado}@usal.es,
{vinglada,vbotti}@dsic.upv.es

Abstract. One of the major challenges in the multi-agent systems field is to build systems capable of taking decisions in an autonomous and flexible way, and to cooperate with other systems inside a virtual organization. If this virtual organization is to be applied successfully to real-world problems, it needs to be flexible enough to cope with the openness of many domains. In particular, a variety of problems need to be addressed such as: large-scale a priori distribution, constant evolution of the environment, admission and departure of members, management and adaptation of the organizational structure, requirements and limitations imposed by the (mobile) devices that support (part of) the execution of the virtual organization and its members, etc. This paper identifies and analyses open research issues that need to be addressed to develop truly open multi-agent systems based on virtual organizations, through the analysis of a real-world case study. Moreover, it puts forward an abstract architecture proposal aimed at integrating methods and techniques in order to address these challenges.

Keywords: Multi-agent systems.

1 Introduction

The recent years technological evolution in the areas of Computer Technology and Communications (Internet, WWW, e-commerce, wireless connectivity, etc.) has given rise to a new computation paradigm: ‘computation as interaction'. In this new paradigm, computation is something that occurs by means of and through communication among computational entities. Given this approximation, computation becomes an inherently social activity, instead of solitary, leading to new forms of conceiving, designing, developing and handling computational systems. An example of the influence of this point of view is the emerging model of software as a service, as in service-oriented architectures. In this model, the applications are no longer monolithic single-user applications, or distributed applications managed by only one organization, but rather societies of computational entities (components) that can be conceived as service providers. These components may not have been designed in a joint way or even by the same development team; they may enter or abandon different societies in different moments and for different reasons; and they may form coalitions or virtual organizations between themselves to attain their current goals. The technology of multi-agent systems (MAS), especially of Open MAS, has some characteristics that show its potential to support this new paradigm of computation as interaction. To satisfy the requirements of this new paradigm, we should provide the technology with the methods, agent architectures, techniques, tools and infrastructures that support these new computational needs in a strong and efficient way.

Dynamic organizations of agents that self-adjust to obtain advantages from their present context are becoming increasingly important. These organizations might appear in dynamic or emerging societies of agents, such as the ones suggested by the Grid domains, Ambient Intelligence domains, or any other domain in which agents coordinate dynamically to offer compound services. Social factors in the organizations of multiagent systems are also more and more important to structure the interactions in dynamic and open worlds. Any infrastructure that supports the execution of multiagent applications in these contexts should be strong and efficient. New approaches are necessary to support the infrastructures evolution, and to facilitate their growth and updating in execution time due to the characteristics of these open environments. Essentially, we come up with two major lines of research: (i) dynamism/regulation: flexibility to permit the entrance and exit of agents, evolution of the organisational structure, coercive mechanisms, etc.; (ii) heterogeneity: different types of agents with diverse capabilities, different run-time coordination mechanisms (requires semantic descriptions of capabilities or services), different devices (physical resources of the devices), different communication channels (wireline or wireless networks). In this context, we need to define development methods, standards, and platforms for the interoperability of agents that consider these requirements. This paper presents an analysis of the main open issues related with open multi-agent systems which must be taken into account in the researche in this area. In order to clarify the complexity and to determine specific aspects or lacks, the analysis is supported by the study of a real-world case: a shopping mall. Based on this analysis an abstract architecture for open multi-agent systems development is presented.

The rest of the paper is structured as follows: section 2 defines open multi-agent systems; section 3 presents the analysis of a shopping mall example; section 4 shows the open issues identified after the analysis of the example; section 5 proposes a new abstract architecture and, finally, some conclusions are given in section 6.

2 Open Multi-agent Systems

Multi-agent systems are a general software technology that focuses on fundamental researching questions about autonomy, cooperation, group formation, etc. Currently, important contributions to this technology have been done and it has been applied in a wide variety of domains. MAS can be classified into open or closed. Their main difference is that in an open system, agents can dynamically enter and leave the system and they have also not necessarily been designed to share common goals. A system is open when it is populated by heterogeneous agents, i.e. agents developed by diverse researching groups and that might employ different languages and architectures. Open systems [30] exist in dynamic operating environments, in which continuously new components may integrate into the system, or already existing components may leave it, and also the operation conditions might unpredictability change. Therefore, open systems are characterized by the heterogeneity of their participants, their limited confidence, possible individual goals in conflict and a great probability of non accordance with specifications [10]. Open MAS allow approaching a wide range of problems. Usually, these are systems where several existing entities (applicants) can ask one or several elements or objectives from other different entities (suppliers). As an example, in the leisure and entertainment activities sector, the applicants would be the clients/users/citizens and the suppliers would be the companies or groups of companies offering any kind of leisure activity such as cinemas, theatres, museums, restaurants. Other examples of open systems are e-commerce applications and information agent systems. The development of open MAS is still a recent field of the multi-agent system paradigm and its development will allow applying the agent technology in new and more complex application domains. So, there exist currently some important problems not solved and that must be taken into account in order to develop real open MAS. In order to support the detection and identification of some open issues, the next section presents the analysis of a particular case study: a shopping mall.

3 A Case Study

The real case study proposed for the analysis is a shopping mall. A mall has become one of the most prevalent alternatives to traditional shopping [26]. A shopping mall is a cluster of independent shops, planned and developed by one or several entities, with a common objective. Every shopping mall has a permanent image and a certain common management. A shopping mall needs to be managed and, the management includes solving incidents or problems in a dynamic environment. As such, a shopping mall can be seen as a large dynamic problem, in which the management required depends on the variability of the products, clients, opinions, etc. Within this framework, *open multi-agent system technology* will make it possible to provide better services to the shopping mall clients. The main goal is to develop an open system, capable of incorporating as many agents as necessary, that can provide useful services to the clients not only in a shopping centre, but also in any other environment such as the labor market, educational system, medical care, etc. Such an open system has to be modeled. Thus, an appropriate analysis and design methodology has to be chosen or has to be developed. Moreover, an optimized and secure multi-device development is required.

A use case diagram of the whole system is proposed in Figure 1. It can be considered that the actors who interact within the system are basically the following: shopping mall clients interested in a particular product or promotion, in obtaining suggestions or recommendations to spend their time in the mall, in receiving news or alerts and in meeting other clients with similar preferences; shop managers that need to advertise their promotions to the potential clients and need to manage certain aspects of their business; mall managers that need updated information to provide primary services in the mall such as security or resolving incidents and to manage the business operation; shopping mall security in charge of security issues; analysts that study the mall operation; restockers, cleaners and cashiers, who solve different kind of incidents; providers service, who supply products; publicists that look for the best promotions; and finally, context-aware devices (radio frequency identification - RFID) that help to identify or locate clients as well as personnel of the centre.

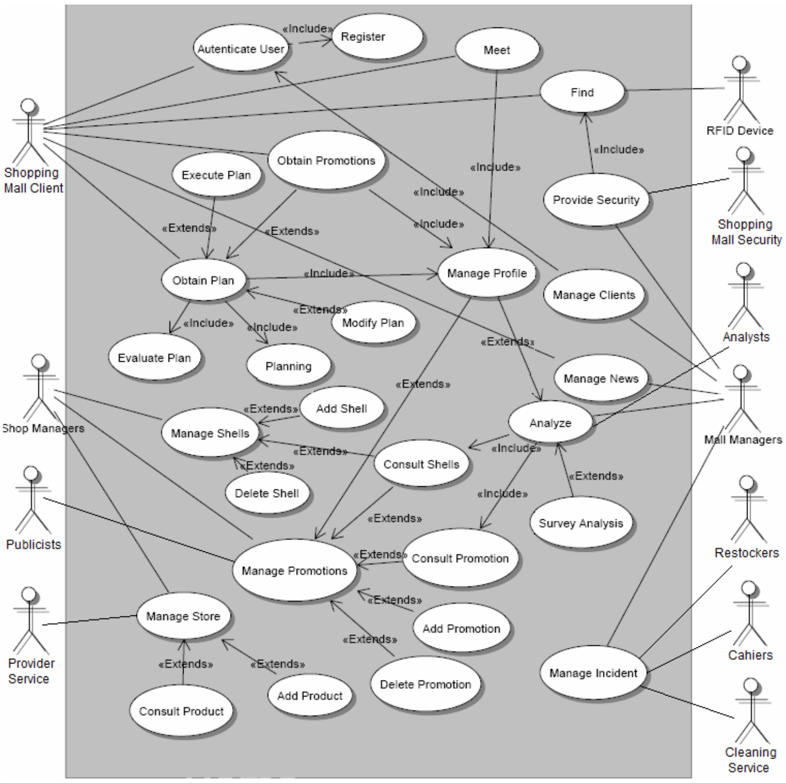

Figure 1. Use Case Diagram for the Shopping Mall Example

Personalization is a value-added service to the traditional commerce techniques. Nowadays users demand personalized services. To provide them, the system needs to identify clients in a transparent and ubiquitous manner (the growing use of handheld devices is a great opportunity to interact with the clients in a simple way) and to manage the clients profiles (preferences, habits, etc.). When a client visits the mall, usually is looking for a product or service. A personal agent installed in his/her personal device can gain access to the services provided in the mall. The client has to be authenticated and his/her personal profile (and not his/her personal data) is then obtained. Security is an important issue in open systems. Once the client has been authenticated, automatically receives personalized offers in the handheld device.

Moreover the client can use different services: consulting service, guiding service (light reasoning mechanisms are required to obtain plans or routes in response to a user's request, looking for the best shopping or leisure time alternatives), meeting service and location service. Personalized services not only help clients to obtain a high quality personal attention, but also help shoppers and mall managers to observe the clients habits (bad behaviours can be penalized by restricting access to certain services and good behaviours can be promoted by means of discounts, etc.). In this sense, an intelligent coordination of the services offered in the mall is necessary. The system also incorporates services for each shop in the shopping mall that facilitate calculating the optimal promotions (those of greater sales success) and services at a given moment by considering the retails data and the user profiles. Besides, the shop managers can manage the store. The shop managers have to fulfil minimum criteria.

The mall managers control important transparent services, such as the coordination of the clients connected to the system and security issues. The security in the mall can be improved if (RFID) devices and an alert message system are installed. These devices allow the mall to provide voluntary location services to the clients (children, elderly or persons in general lost) and to improve the physical security (control the number of persons in a shop, control access to restricted areas, inform about problems, etc.). The mall managers also offer a news service to inform the users about general news of their interest. Furthermore the mall manager carries out periodic evaluations on retails, promotions and surveys data trying to obtain a base for decision making in order to evaluate the clients' satisfaction degree and provide a good quality of service. The last duty of the mall managers is to manage incidents in the mall, solving a wide range of problems (accidents, damages, security, alerts, lost children, cleaning etc.). The great amount of services and information generated in a mall as well as the different roles played in its organization, requires adaptive organizational structures that allow optimizing the coordination in the mall. Moreover, the mall evaluates the client satisfaction degree and the shop agents have to fulfil minimum criteria. These services require the existence of regulatory mechanisms for a global coordination of the system. Furthermore, personalization requires the application of knowledge discovering techniques and of reasoning and planning mechanisms to resolve problems in an efficient way. Finally, the data privacy has to be granted, providing security in data transmissions.

4 Challenges for Open MAS

As has been shown, the complexity of the system proposed in section 3 is very high and there exist some lacks in current technology in order to cover all the described functionalities. Considering the example and the experience that we arrange on the multi-agent area, some new requirements still need to be solved. These requirements are imposed mainly by: computation as an inherently social activity; emergent software model as a service; a non-monolithic application; computational components that form coalitions or virtual organizations with an autonomous and coordinated behaviour; distributed execution environments with wireless connection technology; multi-device execution platforms with limited resources; security and privacy policies for information processing; etc. In order to satisfy all those requirements, current technology must provide interaction features between independent (and intelligent, often enough) entities, which can adapt, coordinate and organize themselves. Agent computation is a promising paradigm since it offers a clear support for those requirements. However, their satisfaction demands agent technology to count on several methods, techniques, tools and frameworks that give support to those new computational needs in an efficient and robust way. The main identified open issues in order to obtain real open MAS are the following:

- A detailed research and analysis of the existing methodologies for the analysis and design of open MAS becomes necessary. Recently, researchers have carried out several studies to offer new procedures and methodologies which enable designing open MAS. Some examples of those approaches are Roadmap [19], Tropos [14], SODA [22], RICA [18] and the new version of Gaia [30]. Many of the recent works do not only focus on the employment of organizational structures during the design process, but also on the regulation of open multi-agent systems (Electronic Institutions [23], Moise+ [17], OMNI [9]), which are normally classified as "virtual organizations". Nevertheless, the abstractions and tools currently available are still not enough for many kinds of open multi-agent systems that deal with real world problems.
- Definition and development of multidevice support platforms that allow the secure and optimized management of multi-agent open systems. In the last years, many research or commercial works have appeared in order to offer this framework, standing out JADE, FIPA-OS, Grasshopper, Jack, ZEUS, AgentScape, MadKit, EIDE, RICA-J [18], SIMBA [28] and SPADE [21]. These platforms cannot be directly applied to the development of open multi-agent systems where organizational structures can emerge and change dynamically at runtime. Therefore it seems necessary to have new platforms that offer a suitable execution framework for the development of virtual organizations.
- Improving the coordination methods in decentralized open systems, based on locating techniques and composition of services, both syntactic and semantic. Most of the current works are strongly related with Web services, so they do not take into account regulation aspects and they are mainly focused on heterogeneity problems in MAS systems ("intelligent" location). It is necessary to employ expressive-marked languages such as OWL [1] for the creation of domain ontologies and also OWL-S [2] or WSMO [25] for the description of what services can do.
- Necessity of adaptive mechanisms for creating organisational structures that allow optimizing the coordination in open systems taking into account the heterogeneity of agents and services. It is necessary to have the possibility to employ basic mechanisms that - based on certain desirable criteria – are able to locate adequate services in open decentralized systems and, if necessary, to construct new services through service composition.
- Existence of regulatory mechanisms that guarantee a globally efficient coordination in open systems taking into account the impossibility of controlling (the majority of) the agents and services directly.
- Adaptation of the most appropriate wireless technology for the implementation of this kind of systems, allowing the communication between agents on intermittent channels to be decentralized. Nowadays, there are still many problems in the

wireless transmission of data and security that need to be resolved. Some of those problems are low bandwidth, security, mobile routing or heterogeneity of devices.

- Development of light reasoning mechanisms based on planning and replanning systems in execution time appropriate for agents in any device. Some of the current works, such as [20], employ planning techniques that cannot tackle with environmental changes nor produce real-time modifications. In [6] an architecture is proposed, that tries to be more flexible in the replanning process but it cannot be applied to problems in which the system dynamics forces to make decisions in real time. In this last field, several based-case reasoning planners have been employed, providing different degree of success, such as PARIS [3], PRODIGY [29], VCBP [7].
- Development of a security policy at agent-level, which is still a problem to solve in agent based systems and it is an even bigger problem in open systems. Suggested solutions are some kind of library or service that offers security mechanisms for agents based on Public Key Infrastructure (PKI), for example [27]. Others have adopted the idea of "intelligent objects" that propose to employ secure agents that communicate between each other in a protected way [11][24].

5 Abstract Architecture for Open MAS

According to the previous analysis, we propose a new architecture for open MAS and a set of related models. This new architecture is called THOMAS (MeTHods, Techniques and Tools for Open Multi-Agent Systems) and will try to tackle all the above mentioned open issues. A general view of the THOMAS architecture is shown in Figure 2.

Agents access the THOMAS infrastructure through a variety of services offered by the Organization *Component*. These include domain-related services, agent-related services, as well as organization-related services. Their provision is supported through the following components:

1. **Environment services component.** This component provides the interface to the services available in the environment and that are accessible through the organization. For this purpose, besides serving as a gateway to those services, a series of meta-services is also provided. First of all, it contains a set of (possibly federated) *directory services* that allow for locating domain services based on a given service description. Furthernore, it may also contain *matchmaking* services in order to account for a more fine-grained semantic search for services that comply with a given query. Finally, the component relies on a composition meta-service endowed with an on-line planning functionality, capable of on-the-fly orchestration of services with regard to a given query. We plan to use OWL-S for service description, but will also have a close look at other initiatives in the field.
2. **Agent platform.** This component of the THOMAS environment provides services usually required by multiagent systems. Firstly, it allows for agent *life cycle management* through a registry. Secondly, it provides a yellow page service that allows agents to locate adequate acquaintance (both facilities can be based on the service directories of the environment services component, together with their

discovery and matchmaking functionalities). Thirdly, it enables agent interaction through an *agent communication language*. Messages are transferred through the different message transport mechanisms offered by the network services layer.

3. **Organizational context manager.** The organizational context manager is in charge of supporting meta-services related to the proper functioning of the organization. These services will be essentially of two types. Firstly, the component will offer *information services*, related to different organizational elements (number of agents playing a certain role, reputation of agents involved in certain interactions, etc.). For this, it will rely both on data gathered proactively, as well as information provided by the agents registered in the organization. Secondly, it will implement policing services aimed at ensuring that agents registered within the organization comply with their obligations (in the simplest case, that agents playing a certain role comply with the (inter-)action constraints pertaining to this role).
4. **Network services layer.** This component's services offer secure connectivity and mobility support across network barriers.

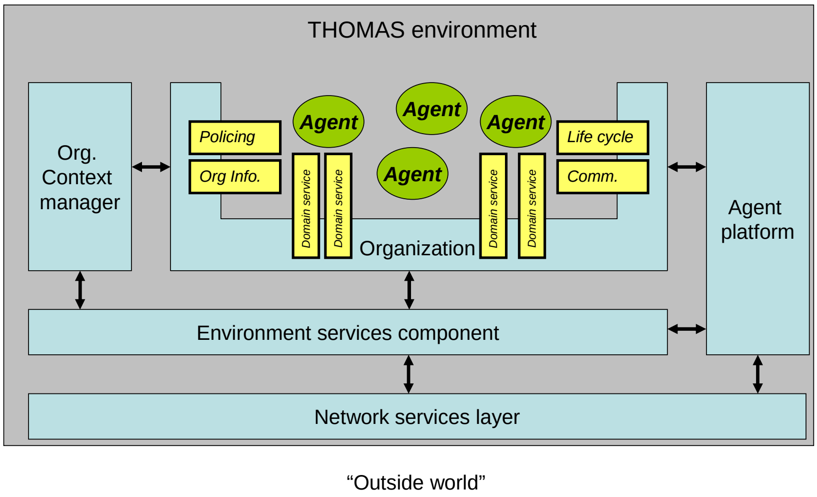

Figure 2. THOMAS Abstract Architecture Proposal

For the implementation of the THOMAS system will set out from several preexisting components such as the TOAST reputation framework [4], [15], the SPADE and Magentix agent platforms [21] and the ALZMAS [11] and TOURIST GUIDE-USAL [12] service composition software, as well as organization-centric support mechanisms for service coordination [13], [12].

6 Conclusions

In this paper we have argued that the new paradigm of computing as interaction is a particularly promising tool for tackling the challenges posed by open software systems supporting mobile users in complex environments. We have presented a shopping mall scenario in order to illustrate such a domain, pointing particularly to functionalities that are not yet supported satisfactorily by existing systems. On this basis, we have identified a set of challenges for the construction of complex software systems that address this kind of problems and have come up with a set of requirements for a multiagent architecture aimed at notoriously open environments. We have put forward such an architecture, providing hints on how it will integrate our previous work in the field of agent platforms, trust, service coordination, and mobility. Although there is an increasing interest in the field of service-oriented multiagent systems [16], current approaches do not exploit sufficiently the organisational aspects of MAS as we suggest in this paper. On the other hand, until very recently, work in the field of Virtual organisations usually did not consider explicit semantics as a key component for the construction of multiagent organisations.

The next steps in our research agenda include a further refinement and a more rigorous specification of our abstract architecture. At the same time, we will advance on the integration of our existing software components, and the development of new functionalities required by the THOMAS architecture. Finally, towards the end of the THOMAS project, we intend to validate our approach through case studies in scenarios similar to the one described in this paper in the Tormes shopping centre at Salamanca.

Acknowledgments. This work has been partially supported by the Spanish Ministry of Education and Science (grant TIN2006-14630-C03), by the Junta de Castilla y León (grant SA104A05), and by the Autonomous Region of Madrid (grant URJC-CM-2006-CET-0300).

References

1. [OWL] The OWL-S Coalition . http://www.w3.org/2004/OWL/
2. [OWL-S] The OWL-S Coalition. http://www.daml.org/services/owl-s/
3. Bergmann R. and Wilke W. (1998). Towards a New Formal Model of Transformational Adaptation in Case-Based Reasoning. European Conf. Artificial Intelligence (ECAI'98).
4. Billhardt, H., Hermoso, R., Ossowski, S., Centeno, R.: Trust-based Service Provider Selection in Open Environments. Proc. of the 2007 ACM Symposium on Applied Computing (SAC), Seoul, Korea (2007), pp. 11-15
5. C. Meadows. Detecting attacks on mobile agents. Foundations for Secure Mobile Code Workshop, pages 64-65, March 1997.
6. Camacho D., Borrajo D. And Molina J. M. (2001) Intelligence Travel Planning: a multiagent planning system to solve web problems in the e-tourism domain. International Journal on Autonomous Agents and Multiagent systems. 4(4) pp 385-390.
7. Corchado, J.M., Laza, R.: Constructing Deliberative Agents with Case-based Reasoning Technology. International Journal of Intelligent Systems. Vol. 18 No.12 (2003) 1227-1241
8. Corchado J.M., Bajo J., de Paz Y. and Tapia D. I.: Intelligent Environment for Monitoring Alzheimer Patients, Agent Technology for Health Care. Decision Support Systems. Eslevier Science. DOI 10.1016/j.dss.2007.04.008. In Press (2007).

9. Dignum, V., Vazquez-Salceda, J., Dignum, F.: OMNI: Introducing Social Structure, Norms and Ontologies into Agent Organizations. PROMAS’05. LNAI 3346 (2005)
10. E. Argente V. Julian and V. Botti. Multi-Agent System Development based on Organizations. Electronic Notes in Theoretical Computer Science pp. To Appear. (2005)
11. F. Hohl and K. Rothermel. A protocol preventing blackbox tests of mobile agents. In ITG/VDE Fachtagung Kommunikation in Verteilten Systemen (KiVS'99). Springer-Verlag, Berlin Germany, 1999.
12. Fernandez, A., Ossowski, S.: Semantic Service Composition in Service-oriented Multiagent Systems: A Filtering Approach. In: J.Huang et al. (eds.): Service-Oriented Computing: Agents, Semantics, and Engineering. LNCS. Springer-Verlag (2007)
13. Fernandez, A., Vasirani, M., Caceres, C., Ossowski, S.: A Role-based Support Mechanism for Service Description and Discovery. In: J.Huang et al. (eds.): Service-Oriented Computing: Agents, Semantics, and Engineering. LNCS. Springer-Verlag (2007)
14. Giorgini, P., Kolp, M., Mylopoulos, J.: Multi-Agent Architectures as Organizational Structures. Journal of AAMAS. Kluwer Academic Publishers (2003)
15. Hermoso, R.; Billhardt, H. Centeno, R.; Ossowski, S. Effective use of organisational abstractions for confidence models. In: Engineering Societies in the Agents World (O’Hare et al., editors), LNAI 4457. Springer. Pages 368–383 (2007)
16. Huhns, M. N., et al.: Research Directions for Service-Oriented Multiagent Systems. IEEE Internet Computing 9 (6) (2005), pp. 65-70
17. J. F. Hübner, J. Sichman, O. Boissier. Using the moise+ for a cooperative framework of mas reorganisation. Advances in Artificial Intelligence, Springer, págs. 506–515, 2004.
18. J.M. Serrano, S. Ossowski, and A. Fernández. On the Impact of Agent Communication Languages on the Implementation of Agent Systems. Cooperative Information Agents VIII (Klusch, Ossowski, Kargupta & Unland, editores). Springer, 2004.
19. Juan, T., Pierce, A., Sterling, L.: Roadmap: Extending the gaia methodology for complex open systems. Proc. AAMAS-2002, págs 3–10
20. Knobolock C. A., Minton S., Ambite J. L., Muslea M., Oh J. and Frank M. (2001) Mixed-initiative, multisource information assistants. 10th Int. WWW conference. ACM Press.
21. M. Escriva, J. Palanca, G. Aranda, A. García-Fornes, V. Julian and V. Botti A Jabber-based Multi-Agent System Platform. Proc. AAMAS06 pp. 1282-1284. (2006)
22. Omicini, A. 2001. Soda: Societies and infrastructures in the analysis and design of agent-based systems. Agent-Oriented Software Engineering Vol. 1957. pp. 185-193.
23. P. Noriega, C. Sierra. Electronic institutions: Future trends and challenges. Cooperative Information Agents VI, (Klusch, Ossowski, Shehory, editors),LNCS 2446 Springer, 2002
24. Roth, V. Mutual protection of co-operating agents. In Secure Internet Programming 1999, pages 275-285, 1999.
25. SDK WSMO working group http://www.wsmo.org/.
26. T. Bohnenberger, O. Jacobs and A. Jameson (2005). DTP meets user requirements: Enhancements and studies of an intelligent shopping guide, in: Proc. PERVASIVE-05, Lecture Notes in Computer Science, Vol. 3468 (Springer, Berlin, 2005) 279-296.
27. U. Maurer. Modelling a public-key infrastructure. In ESORICS: European Symposium on Research in Computer Security. LNCS, 1996.
28. V. Julián, C. Carrascosa, M. Rebollo, J. Soler, V. Botti. SIMBA: an Approach for Real-Time Multi-Agent Systems. En Actas de V CCIA. LNCS 2504. p. 282-293. 2002.
29. Veloso M., Carbonell J., Perez A., Borrajo D., Fink E. and Blythe J. (1995) Integrating Planning and Learning: The PRODIGY Architecture, Journal of Theoretical and Experimental Artificial Intelligence, 7(1), 1995.
30. Zambonelli, F.; Jennings, N.; Wooldridge, M. 2003. Developing multiagent systems: The GAIA methodology. ACM Trans. on Soft. Eng. and Methodology. V. 12. pp. 317- 370.