

How To Design a Multi Agent System Dedicated to Knowledge Management; the DOCK Approach

Julien Girodon¹, Davy Monticolo² and Eric Bonjour¹

¹Essilor, Paris, France

²ERPI laboratory, University of Lorraine, Nancy, France

Abstract: *Knowledge-Based Engineering approaches (KBE) are designed to reduce the time and cost of product development by capturing, retaining and re-using design knowledge. They currently focus on repetitive design tasks where knowledge is considered as a resource. However, knowledge is intrinsically linked to the people and organizations. Thus, to be efficient, these knowledge-based systems (KBS) have to be able to take into account all the mechanisms of knowledge creation, sharing and evaluation made by the users. Using the agent paradigm, new knowledge-based systems can be designed in order to address this research issue. Indeed the agents have social abilities and are able to achieve very complex tasks. These two features are necessary to make a knowledge-based system efficient. However, there still exists today a lack of approaches or methodologies to help to design such applications. This paper presents DOCK, a methodology to design an intelligent knowledge-based system that aims to support the knowledge management process, as well the decision making process throughout engineering projects. In order to take into account all the mechanisms of knowledge generation, sharing and re-use, DOCK is based on the hypothesis that efficient modeling of the human organizations by highlighting their roles, collaborations, skills, goals and knowledge will help the designer to specify an adapted knowledge-based system.*

Keywords: *Organizational Modeling, Multi-agent Systems, Knowledge-based system*

1. Introduction

Companies are engaged in continuous performance improvement processes to stay competitive in their core business. Performance improvement levers can be found at every level of the company and can be summarized around three main axes. The first axis has the purpose of defining how the performance of the products and the design processes can be improved. This area is well recognized and investigated by enterprises. It embeds methodologies, methods and tools such as functional analysis, dependability, statistical process control and, modeling & simulation. The second axis is oriented towards an agreement on how to improve the performance of organizational processes. This area embeds, for instance, all the methodologies and tools of project management, agile methodologies, system engineering, or quality management system. Finally, the third axis deals with the improvement of people performances. It embeds all the elements dealing with the ways to better manage and valorize the knowledge and competences. This paper focuses on this last axis in order to present a methodology to design a support system to enhance knowledge sharing. The research field of Knowledge-Based Engineering (KBE) proposes many approaches that allow one to capture and re-use knowledge. These approaches currently focus on repetitive design tasks where knowledge is considered as a resource. However, knowledge cannot be simply reduced to a static resource based view, because of its intrinsic nature. People interact together during meetings or workshops in a collaborative way to share their expertise and experiences in order to carry on a common objective. Thus they generate organizational groups where the knowledge is created, shared, evaluated and updated **Error! Reference source not found.** In order to take into account this complexity, the KBE approaches need to rely on knowledge models having richer semantics and better traceability **Error! Reference source not found.** According to this statement, and contrary to other KBE approaches, an organizational approach makes it possible to highlight the mechanisms of knowledge sharing and evolution, rather than focusing on the formalization and the modeling of knowledge **Error! Reference source not found.**

Recently, the Multi-agent Systems (MAS) research field has proposed a metaphor of an organizational model in order to design and implement intelligent systems. Multi-agent systems working in a decentralized way are able to use distributed and incomplete sources of information and knowledge **Error! Reference source not found.** However the existing methods to design and implement these systems are not knowledge based oriented i.e. they do not take into account all the mechanisms of knowledge creation, sharing and evaluation made by the users. Thus based on these approaches, this paper presents DOCK, a methodology to design an intelligent knowledge-based system, which aims to support the knowledge management process.

2. Literature review of the MAS design

The design methodologies of MAS cover each step in the life cycle of an agent-oriented application **Error! Reference source not found.** These methodologies can be split into two main categories: agent-oriented approaches and organizational approaches **Error! Reference source not found.** In the former of the two categories, designers focus on the description of the agents' individual actions, whereas the latter category focuses on the description of the agents' organizations.

By analyzing an organization, one can define how members of a society act and interact with one another. This is important as the effectiveness of an organization depends on several factors, including the coordination between the members and their ability to solve problems. Several published works focus on the effects of organizational structures on performances and knowledge sharing inside a group **Error! Reference source not found.,Error! Reference source not found.,Error! Reference source not found.,Error! Reference source not found.** However, it is difficult to take these structures into account in standards KBE approaches because they currently deliver centralized systems **Error! Reference source not found.**

To effectively model the dynamic aspect of the organizational environment, the agent paradigm can be used. Agents are by definition, autonomous entities, proactive and capable of social interaction in dynamic environments **Error! Reference source not found.** They are thus, able to handle this issue when they are considered as components of groups of agents that interact together in a system in order to fulfill common goals.

In addition, the increasing complexity of the MAS (number of entities, number of complex tasks to fulfill, dynamical environment, etc.) has led researchers to work on an individual point of view of the agents and their interactions. This research is dedicated to the specific field of agent organizations and has increased the flexibility of agents architectures **Error! Reference source not found.Error! Reference source not found.** In order to give a synthetic view of these works, Dignum and Meyer in **Error! Reference source not found.** provide the following definition: "The agent organizations can be seen as sets of entities and their interactions, regulated by mechanisms of social order and created by autonomous actors to achieve a common goal". This definition helps to emphasize the social aspect of agent organizations. In addition, Boissier and Demazeau **Error! Reference source not found.** assert that: "An organization of agents can be seen in a simple way as a set of constraints adopted by a group of agents with the aim to facilitate the achievement of their goals". From these two statements it is clear that within an organization, agents must take into account the objectives of the group so that they have the ability to achieve their own goals. Guizzardi **Error! Reference source not found.** proposes to include the concept of knowledge within organizations: "An organization of agents is a community of knowledge sharing in which agents collaborate and exchange knowledge to carry out their activities." This definition is important because it highlights the effectiveness of knowledge sharing within organizations of agents.

As a result, knowledge based engineering methodologies can provide good basics for MAS design so long as they enable to model the mechanisms of knowledge sharing within a community. There are currently two relevant approaches that address this topic in KBE field: the MAS-CommonKADS approach **Error! Reference source not found.** and the MASINA methodology **Error! Reference source not found.** which is an extension

of the former one. These two approaches belong to the agent-oriented methodologies to design MAS, because they focus on the description of process elements such as tasks and relationships by a collection of rules. By a set of models, they guide the MAS designers through a bottom-up analysis of the system starting from the agents' specifications to the organization's architecture description. Compared to the MAS-CommonKads approach, MASINA strengthens the communication between agents by modifying the meaning of the coordination and communication models of MAS-CommonKads and introducing a new model. But in order to efficiently cover the mechanisms of knowledge sharing through organization, a top-down analysis of the system seems better suitable. Starting from the agents' organization description to the agents specifications enables to identify roles and interactions [21] that are crucial to describe knowledge sharing mechanisms.

According to the descriptions of the agents' organizations, Carley **Error! Reference source not found.** explains that the analysis and the study of human organizations allow one to design computational models which can be used to enhance three perspectives in organizations: the structure, the information sharing, and the social cooperation. The agent paradigm, together with, coordination, stigmergy, adaptivity and evolution enables the development of knowledge engineering applications to enhance the performance of human organizations.

3. Design Methodology of a Knowledge-based system: the DOCK Approach

This section presents an organizational approach to design MAS oriented to Knowledge Engineering applications. This approach is called DOCK for **D**esign based on **O**rganization, **C**ompetence and **K**nowledge, and focuses on the organizational modeling of the future Multi-agent system in order to specify its architecture.

The approach begins by modeling human organisations and highlights the skills of the professional actors, the description of the knowledge that they create, use and share, and the formalization of the organizational structure that they enforce. The fundamental elements of the modeling can be divided into four models (Fig. 2):

- The Organizational structure model that embeds a set of elements that can characterize an organization such as its missions, goals, etc.
- The Process model which describes the sequence of activities made by actors.
- The activity model which describe in detail the feature of the activities with elements such as the competences mobilized (skills in the model), the knowledge, the roles played and the responsibilities. This part allows one to formalize a role-based view of the organization's human resources.
- The role model that displays the set of items related to the knowledge generating within an organization by a role, such as, the activity performed, the expected deliverables, the mobilized knowledge, as well as the procedures and operating modes. Identify these elements allow one to characterize the knowledge according to a situation (the activity).

This set of models allows one to create a complete organizational modeling taking into account the notions of organization, knowledge and competences. Finally, this organizational modeling is complemented within DOCK by the specification of the MAS general architecture through the agents' model and interactions model. This part is performed more closely to agent's paradigm requirements.

3.1. The organizational structure model

Modeling the agents' organizational framework aims to describe their general environment, their macro interactions, and their activities. This modeling relies on the organizational structure model and the process model. This model is structured around several concepts: the organization, the missions, the goals and the processes. The aim of this model is to highlight the main missions of an organization, what goals it fulfills and what are the processes on which it can rely on. The goals, missions and processes are depicted as boxes and linked with arrows (Fig. 1). There is one and only one Organizational structure model per organization. However, a MAS can be built upon several organizations. In that case, by identifying the overlapping areas with the Organizational structure models, designers can better identify the interactions to formalize in the following design steps.

3.2. The Process model

A process is defined by the organization (as seen above). It can be split into several activities that are of a lower complexity level. The role is also defined by the organization and is assumed by an actor.

Based on these elements, each process identified above, is formalized through a “Process model” in order to identify the contribution of each of the roles in the organization. The model is based on the integrated computer aided manufacturing definition methodology (IDEF0) **Error! Reference source not found.**, customized with the responsibility alignment matrix methodology, RACI **Error! Reference source not found.**, in order to take into account the responsibility levels of the roles (Fig. 1). The boxes display the activities; the horizontal arrows are the inputs and outputs of each activity (deliverables) and the field under each box is dedicated to the roles involved in the realization of the activity. Next to the role name, in brackets, is its degree of involvement following the RACI methodology:

- Responsible: The one who is in charge of the realization of the activity.
- Accountable: The one to whom "R" is accountable / Who must sign off (approve) on work before it is operative.
- Support: The one who can provide resources or can play a supporting role in the implementation.
- Consulted: The one who has information and/or the capabilities necessary to complete the work.
- Informed: The one who must be notified of results, but does not need to be consulted.

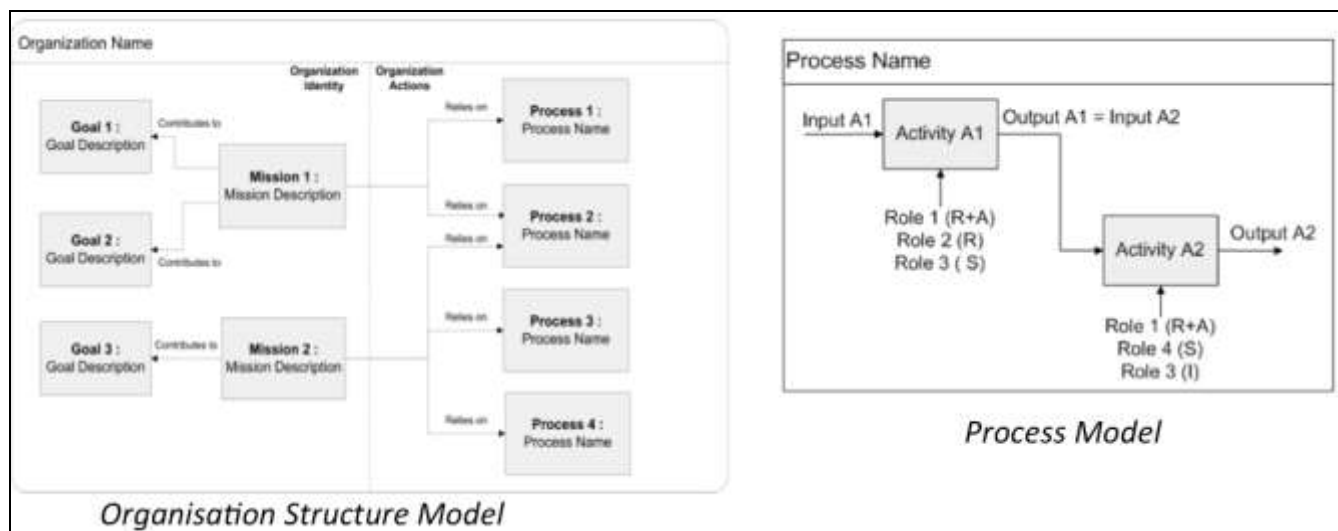


Fig. 1: The Organizational structure model And the Process models

To summarize, these two models allow one to define the general framework of the studied organization. This step is crucial to highlight the organizational nature of knowledge sharing and combining. For a given organization, the rule within DOCK is to formalize only one “Organizational structure model” and as many “Process model” as required covering the whole organization’s action domain. Based on these two model types, the next step consist of the formalization of each activity identified in order to highlight the knowledge mobilized, used and shared.

3.3. The Activity model

This model is formalized for all the activities identified in the process models seen above. It assumes that the realization of an activity generates deliverables and knowledge, mobilizes competences, and can lead to the update of action schemas. This is the result of interactions between several roles, assumed by agents or human actors according to the studied process.

The model is an adaptation of the RIOCK formalism (Role, Interaction, Organization, Competence, and Knowledge) **Error! Reference source not found.** The main goal of this modeling is to highlight the interactions between roles for each activity and to identify the elements used and generated through it (Fig. 2).

Roles are defined by their competences, knowledge, responsibilities and actions schema. Each role is represented in the model by a box embedding all these elements. The interactions between roles are formalized by a circle linked to the roles' boxes by connectors. The distinction is made between three types of interactions; the alliance, the coalition or the leadership.

These circles are also linked to the elements generated through the activity. These elements are pictured by rectangles with rounded corners. The model displays the activity's deliverables, the knowledge generated, the competences implemented and the action schema updated (Fig. 4).

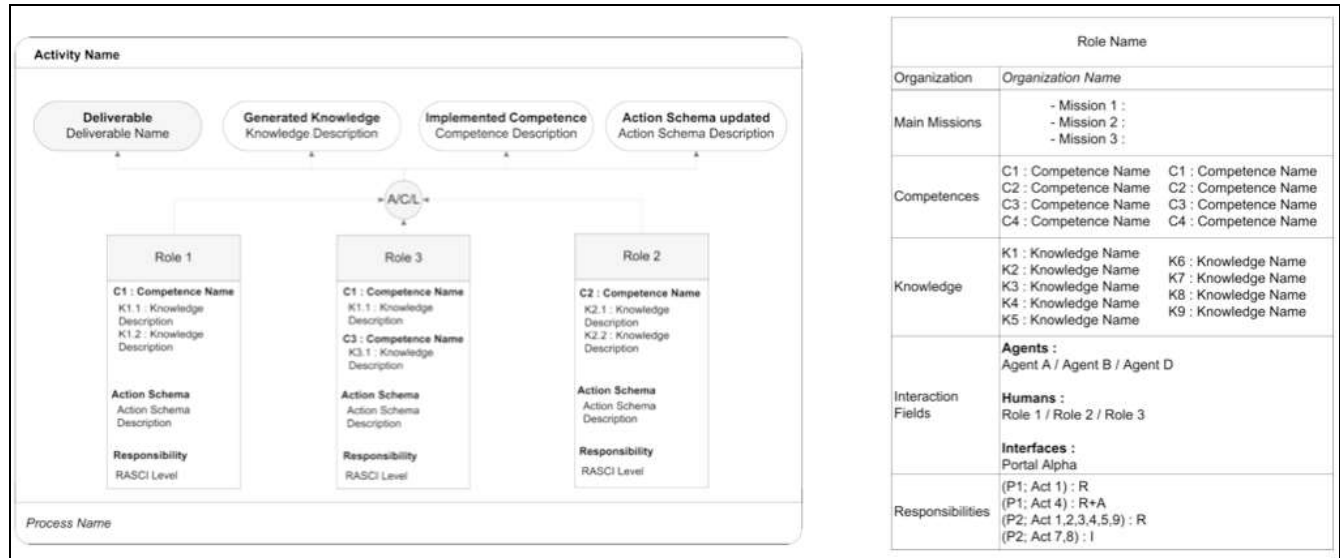


Fig. 2: The activity model and the role model

This model is the cornerstone of the organizational modeling. It highlights that knowledge is linked to a working situation and to concepts belonging to the Organizational structure and the Competence Management field. According to the number of processes implemented by an organization and their complexity, the model's instantiation number can increase quickly. Therefore, it allows one to formalize a complete knowledge mapping for the organization. Based on these elements, the modeling can focus next on the consolidation of the roles.

3.4. The Role model

The Role model allows one to consolidate elements that are spread across several activity models around a same role. Indeed, in the modeling, a role is defined by several attributes that can be retrieved through the previous models. These elements are the organization, the mission, the knowledge, the competences, the interaction field and the responsibilities. The model gathers all these elements in a single formalism (Fig. 2).

The Role models allow the designers to formalize each function that need to be performed by the system. A function includes the achievement of a set of activities targeting a common goal. Based on these models, the designers can afterward allocate these roles to the agents in order to design them with the required abilities for the future system. Moreover, by considering the interactions between the different roles in the activity models, the links between the agents' organizations and the humans' organizations are highlighted. This general understanding may avoid the design of virtual organizations that disconnected from human activities.

4. Defining the general architecture of the Multi-agent system

The four organizational models allow one to design the whole action framework of the future intelligent knowledge-based system.

4.1. Identification of the agents' roles : the agent model

In DOCK the roles identified in the organizational modeling are distributed and instantiated among the agents and humans actors. There are two types of agents needed to design a knowledge-based system; the cognitive agents that monitor the actions of the human actors and the reactive agents that support the knowledge management process.

Firstly, the reactive agents are those which receive information and knowledge from their environment (from another agent, human or interface) and achieve exploitation tasks. They do not simulate a behavior. The reactive agents are dedicated to managing the knowledge and its lifecycle.

Secondly, the cognitive agents are those which have roles close to humans' roles. They monitor the humans' roles or simulate the behavior of a human. Thus, their roles are deduced from the modeling of the organization of human actors with the KROM approach. The Activity model lists the roles used in human organizations and the "Role model" details all the characteristics of each role.

4.2. Identification of the agents' organization

The agents architecture design is crucial for the efficiency of the future system. Several agents' structures can be found according to the degree of interactions between agents. With DOCK, agents are organized in teams in order to ease the knowledge capitalization and sharing (Fig. 3). This organization relies on three layers. The first one is the human organization studied (1). The second layer is the cognitive agents' organization (2) and the last one is the reactive agents' organization based on the KM process described above (3).

At this step, designers identify the macro interactions between the reactive agents' model, the cognitive agents' model and the human organization that the system should support. In order to realize that, the elements identified in the previous models and especially in the activity models must be used to have a global view of the system architecture (Fig. 3).

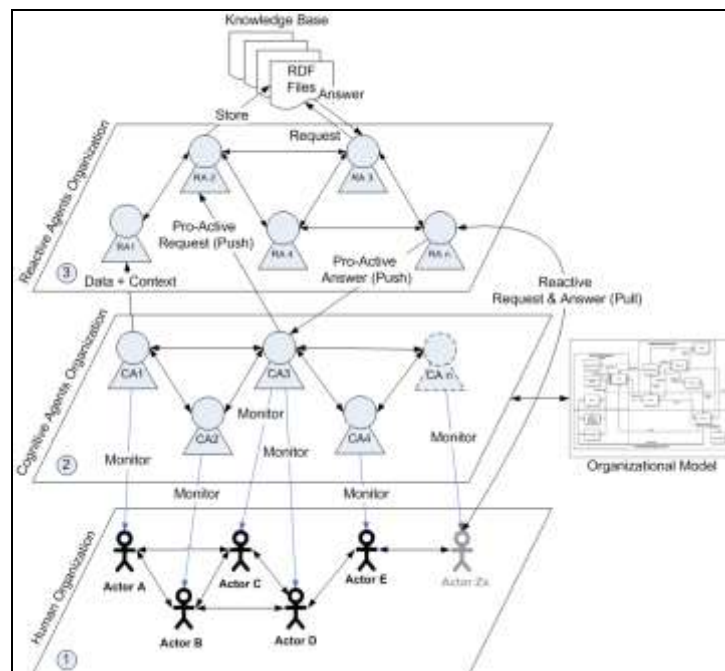


Fig. 3: General Organization of the agent system

This view of the system is now ready to be further developed. The last part of DOCK is focused on the description of the communications between agents.

4.3. Defining the general communication scheme

After identifying the agents' organization, the designers can specify the general interactions between the agents based on the previous models, such as the process models, the roles models and the agent models. This specification allows one to identify the information or knowledge shared between agents and to formalize the communications with and inside the system. In order to keep the sequence of exchanged information and knowledge, the sequence model of the UML approach is used to formalize these communications (Fig 4).

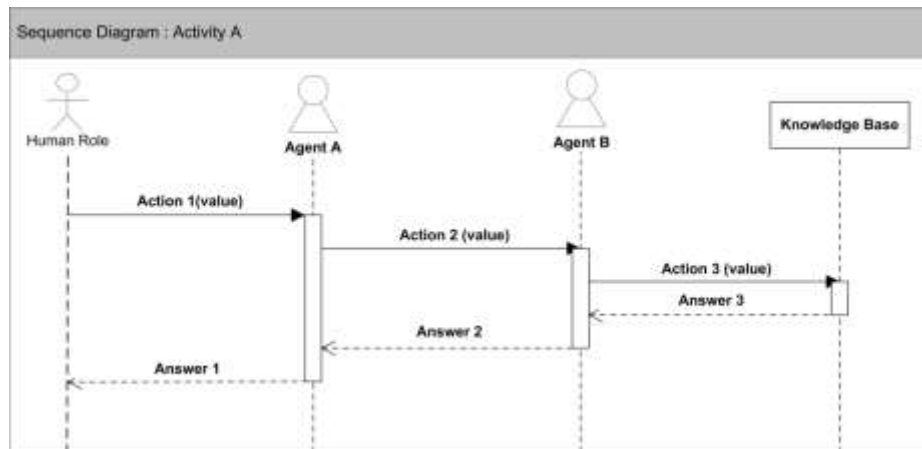


Fig 4: Sequence diagram model

On the sequence diagrams, agents are represented by a specific symbol. Other elements of the system, for example the knowledge base, are represented by the classic representation of an object in the UML sequence model. The “values” presented in Fig. 9, represent the information and knowledge exchanged between agents. In this approach each activity coming from the process and activity models must translate into one or several sequence diagrams. This last element of the DOCK approach allows designers to identify the interfaces and the general architecture of the future MAS.

5. Conclusion and future work

The DOCK methodology is an organizational approach to the design of Multi Agent Systems applied to Knowledge-based systems. Compared to the major MAS design approaches, DOCK aims to take into account the organizational aspect of the knowledge in order to design intelligent knowledge-based systems.

At this stage, the approach still shows some areas where additional research is required. Firstly, the DOCK approach does not yet cover the agents' lateral concepts such as the beliefs, desires and intentions. Agents' lateral concepts have to be taken into account by the DOCK approach in order to propose upgradeable systems and be able to address more complex developments. Secondly, the approach is not yet consolidated enough to take into account the implementation and testing steps of the software development process. Finally, in line with the previous point, the approach cannot yet rely on supporting tools to help the designer to go through the development steps and particularly the implementation and testing steps. These points will be addressed in future works in order to provide a design methodology adapted to the development of KBE systems and usable by the largest number of designers.

6. References

- [1] D. Monticolo, J. Badin, S. Gomes, E. Bonjour, D. Chamoret, A meta-model for knowledge configuration management to support collaborative engineering, *Comput. Ind.* 66 (2014) 11–20. doi:10.1016/j.compind.2014.08.001.

<http://dx.doi.org/10.1016/j.compind.2014.08.001>

- [2] M.-T. Chu, P. KrishnaKumar, R. Khosla, Mapping knowledge sharing traits to business strategy in knowledge based organisation, *J. Intell. Manuf.* 25 (2012) 55–65. doi:10.1007/s10845-012-0674-1.
<http://dx.doi.org/10.1007/s10845-012-0674-1>
- [3] W.J.C. Verhagen, P. Bermell-Garcia, R.E.C. van Dijk, R. Curran, A critical review of Knowledge-Based Engineering: An identification of research challenges, *Adv. Eng. Informatics.* 26 (2012) 5–15. doi:10.1016/j.aei.2011.06.004.
<http://dx.doi.org/10.1016/j.aei.2011.06.004>
- [4] A. Abecker, A. Bernardi, L. van Elst, Agent technology for distributed organizational memories, in: *Proc. 5th Int. Conf. Enterp. Inf. Syst., Angers, FRANCE, 2003*: pp. 3–10.
- [5] L. Monostori, J. Váncza, S.R.T. Kumara, Agent-Based Systems for Manufacturing, *CIRP Ann. - Manuf. Technol.* 55 (2006) 697–720. doi:10.1016/j.cirp.2006.10.004.
<http://dx.doi.org/10.1016/j.cirp.2006.10.004>
- [6] C.A. Iglesias, M. Garijo, J.C. Gonzalez, A Survey of Agent-Oriented Methodologies, in: J. Müller, M.P. Singh, A.S. Rao (Eds.), *5th Int. Work. Intell. Agents V Agent Theor. Archit. Lang.*, Springer Verlag, Paris, France, 1999: pp. 317–330.
http://dx.doi.org/10.1007/3-540-49057-4_21
- [7] G. Cristina, D. Isern, A. Moreno, RESEARCH REPORT Software Engineering Methodologies to Develop Multi-Agent Systems : State-of-the-art, Tarragona, Espagne, 2007.
- [8] C. Liao, S.-H. Chuang, P.-L. To, How knowledge management mediates the relationship between environment and organizational structure, *J. Bus. Res.* 64 (2011) 728–736. doi:10.1016/j.jbusres.2010.08.001.
<http://dx.doi.org/10.1016/j.jbusres.2010.08.001>
- [9] W. Zheng, B. Yang, G.N. McLean, Linking organizational culture, structure, strategy, and organizational effectiveness: Mediating role of knowledge management, *J. Bus. Res.* 63 (2010) 763–771. doi:10.1016/j.jbusres.2009.06.005.
<http://dx.doi.org/10.1016/j.jbusres.2009.06.005>
- [10] B. Michaelis, J.D. Wagner, L. Schweizer, Knowledge as a key in the relationship between high-performance work systems and workforce productivity (IN PRESS), *J. Bus. Res.* (2014). doi:10.1016/j.jbusres.2014.10.005.
<http://dx.doi.org/10.1016/j.jbusres.2014.10.005>
- [11] J. Rašula, V.B. Vuksic, M.I. Stemberger, The impact of knowledge management on organisational performance, *Econ. Bus. Rev.* 14 (2012) 147–168. <http://www.ebrjournal.net/ojs/index.php/ebr/article/view/85>, 2015.
- [12] N.R. Jennings, M.J. Wooldridge, *Agent Technology Foundations, Applications*, Springer, New York, 1998.
<http://dx.doi.org/10.1007/978-3-662-03678-5>
- [13] L. Gasser, Perspectives on organizations in multi-agent systems, *Multi-Agent Syst. Appl. Lect. Notes Comput. Sci.* 2086 (2001) 1–16.
http://dx.doi.org/10.1007/3-540-47745-4_1
- [14] B. Horling, V. Lesser, A survey of multi-agent organizational paradigms, *Knowl. Eng. Rev.* 19 (2005) 281–316.
<http://dx.doi.org/10.1017/S0269888905000317>
- [15] L. Shan, H. Zhu, CAMLE : A Caste-Centric Agent Modelling Language and Environment, in: *3rd Int. Work. Softw. Eng. Large-Scale Multi-Agent Syst. SELMAS'2004*, Edinburg, United Kingdom, 2004: pp. 1–13.
- [16] V. Dignum, F. Dignum, J.-J. Meyer, An agent-mediated approach to the support of knowledge sharing in organizations, *Knowl. Eng. Rev.* 19 (2004) 147–174. doi:10.1017/S0269888904000244.
<http://dx.doi.org/10.1017/S0269888904000244>

- [17] O. Boissier, Y. Demazeau, ASIC: An architecture for social and individual control and its application to computer vision, *Distrib. Softw. Agents Appl. Lect. Notes Comput. Sci.* 1069 (1996) 135–149.
http://link.springer.com/chapter/10.1007/3-540-61157-6_27 (accessed October 31, 2014).
- [18] R.S.S. Guizzardi, *Agent-oriented Constructivist Knowledge Management*, University of Twente, the Netherlands, 2006.
- [19] C. a Iglesias, M. Garijo, J.R. Velasco, *Analysis and Design of Multiagent Systems Using MAS-CommonKADS**, *Intell. Agents IV Agent Theor. Archit. Lang. Lect. Notes Comput. Sci.* 1365 (1998) 313–327.
doi:10.1007/BFb0026768.
<http://dx.doi.org/10.1007/BFb0026768>
- [20] A. Jose, C. Mariela, H. Francisco, *A Methodology to Specify Multiagent Systems*, *Agent Multi-Agent Syst. Technol. Appl. Lect. Notes Comput. Sci.* 4496 (2007) 92–101.
http://dx.doi.org/10.1007/978-3-540-72830-6_10
- [21] P. Gruer, V. Hilaire, A. Koukam, K. Cetnarowicz, *Short Survey A formal framework for multi-agent systems analysis and design*, *Expert Syst. Appl.* 23 (2002) 349–355.
[http://dx.doi.org/10.1016/S0957-4174\(02\)00070-2](http://dx.doi.org/10.1016/S0957-4174(02)00070-2)
- [22] K.M. Carley, *Computational organization science: a new frontier.*, *Proc. Natl. Acad. Sci. U. S. A.* 99 Suppl 3 (2002) 7257–7262. doi:10.1073/pnas.082080599.
<http://dx.doi.org/10.1073/pnas.082080599>
- [23] C.L. Ang, M. Luo, L.P. Khoo, R.K.L. Gay, *A knowledge-based approach to the generation of IDEF0 models*, *Int. J. Prod. Res.* 35 (1997) 1385–1412. doi:10.1080/002075497195380.
<http://dx.doi.org/10.1080/002075497195380>
- [24] Project Management Institute, *Project Management Body of Knowledge (PMBOK Guide), Fourth Edition*, Project Management Institute, Inc., 2008.
- [25] D. Monticolo, V. Hilaire, S. Gomes, A. Koukam, *A multi-agent system for building project memories to facilitate the design process*, *Integr. Comput. Aided. Eng.* 15 (2008) 3–20.