Knowledge-Based Systems 68 (2014) 88–102

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

An agent-based system to build project memories during engineering projects

Davy Monticolo [a,*], Simona Mihaita [a], Hind Darwich [a], Vincent Hilaire [b]

[a] Université de Lorraine, ERPI Laboratory, 8 Rue Bastien Lepage, 54000 Nancy, France
[b] University of Technology UTBM, 90010 Belfort, France

ARTICLE INFO

Article history:
Available online 24 January 2014

Keywords:
Organizational model
Multi-agent system
Knowledge engineering
Ontology
Experience reuse

ABSTRACT

Engineering projects are organizations where several actors with different professional fields and know-how work together to carry out the same aim: to develop a new product. Inside these organizations, heterogeneous and distributed information has to be managed in order to create project memories that will be useful in future projects. In this paper we describe a Multi-Agent System (MAS), which is based on the social and cooperative approach to support the knowledge management process all along mechanical design projects. Indeed, this multi-agent system, called KATRAS, aims to capitalize and reuse knowledge according to the roles involved in the design projects. We will present in this paper how the agents capitalize six different types of knowledge (professional vocabulary, process, expertise, project evolution, and return of experience) and how they help the professional actors to reuse knowledge.
© 2013 Elsevier B.V. All rights reserved.

1. Introduction

Current competitiveness has led companies to fast product renewal coupled with lower costs. Today, manufacturers are creating more and more efficient products while meeting shorter deadlines in order to satisfy the customer's needs and target sales.

These companies have to break into new markets by showing how creative they are in order to grow more profitable. Such creativity requires an optimized organizational mastery, a control of the industrial process and the development of a 'learning company' in which getting the innovation knowledge represents an asset for the ongoing projects. Learning within the company has now become the best way to be competitive. The 'learning culture' implies that every co-worker, every team and even the whole company will be able to optimize their capacities by continuously sharing their knowledge and their know-how and to learn about the best actions and the failures encountered in the past.

In this article we present a knowledge management approach based on a multi-agent system, which captures knowledge according to the roles of the professional actors. We will focus on the various roles played by the actors and their cooperation. Our approach will dwell on the study and the modeling of different design activities from an organizational point of view.

Organizations can be defined as a set of connected entities run by social interactions between independent actors in order to achieve a common goal [14,9]. Every actor is autonomous thanks to his/her knowledge and know-how, but also every co-worker communicates with every other so that their common activities will be performed successfully. Due to the sustained interactions between actors, knowledge is shared and experiences are created inside the organization. These organizations need to be modeled in order to create a system that is able to capitalize knowledge and build up project memories including experience reuse. The system will also enable the reuse of knowledge every time the actors involved in the project demand it.

In the first part of this paper, we will introduce a description of the various project memory types. Section 2 presents an analysis of knowledge engineering research using the agent paradigm. This paper will lead us to a thorough analysis of the proposed multi-agent system regarding knowledge management. Finally, some knowledge management packages, especially the experience reuse module, will be presented.

2. Heterogeneous and distributed project memories

Knowledge is an organization's most powerful resource used to improve profitability and maintain competitiveness, an aspect which determines the knowledge management to become critical for organizational success [34]. The need to store knowledge inside the companies has led to the creation of organizational memories, often called project memories [48].

* Corresponding author. Tel.: +33 (0) 3 83 19 32 49, fax: +33 (0) 6 15 75 34 44.
E-mail addresses: davy.monticolo@univ-lorraine.fr (D. Monticolo), adriana-simona.mihaita@univ-lorraine.fr (S. Mihaita), vincent.hilaire@utbm.fr (V. Hilaire).

http://dx.doi.org/10.1016/j.knosys.2013.12.022
0950-7051/© 2013 Elsevier B.V. All rights reserved.

A project memory can be defined as an explicit and continuous representation of knowledge, data or data source within an organization [27,45]. Moreover Abecker et al. [2] adds the fact that the project memories also contain the context in which the knowledge has been created, as it allows the annotation of the information and facilitates the reuse of information.

Therefore, the stake of the project memory management is to allow professional actors to reuse and share knowledge, which has been capitalized from previous projects in order to carry out a new one [5]. Our design project analysis in different companies led us to the conclusion that:

- A project memory is by nature a source of heterogeneous and distributed information coming from software programs, technical documentation or staff meeting reports [32].
- The project memory users are heterogeneous and distributed. They have specific qualities and play different roles all throughout projects, such as design engineers, mechanical engineers, automatic engineers, and assembly technicians. These professional actors have to collaborate during the project and are in different geographic locations.

In order to do this, the creation of a project memory, by taking into consideration these heterogeneous and distributed information sources, is based on different stages, as expressed by [18]: the needs, the detection, the construction and structuring, the diffusion and the contents, the exploitation of the contents, the evaluation of the objectives, the maintenance and the evolution of the contents.

In the literature, different works propose new methods and models for data acquisition and construction of project memories. For example, Ribière et al. [50] proposes a project memory based on conceptual graphs for knowledge representation, Matta et al. [42] develops a project memory by capitalizing the trace of old projects, while others focus on the design of novel methods for tracking the design rationale: QOC [52], DRCS [38] or on the evolution of decisions during a project, like DyPKM [6]. Recent studies have been building project memories by exploiting the connection between knowledge sharing and employee service performance [40], or by creating a transactive memory system [36].

The main objective of a project memory is to help the involved actors to accomplish their activities or to solve new problems. To achieve these activities, the actors need to use a common terminology, especially when they are geographically distant; this explains why organizational memories are often based on ontologies, an aspect which will be discussed in the following section.

Thus we propose to design a knowledge-based system that enables to manage heterogeneous and distributed information and to take into account the social and collaborative aspects concerning professional actors. The distributed artificial intelligence domain, and more particularly the Multi-Agent Systems (MAS) facilitate the modeling of increasingly sophisticated systems. Studies have shown that the agent paradigm has turned out to be well adapted to the software structure design that ensures heterogeneous and distributed information management.

Indeed, a MAS can provide a flexible organizational memory, i.e. to adapt the structure of the memory according to the role of the actors and the knowledge they can create, share or use.

The next section will present the benefits of using software agents in the knowledge management domain.

3. The agent paradigm in knowledge engineering

Knowledge engineering is meant to gather, study, organize and represent knowledge. Multi-agent systems seem to be able to perform such a task. Klusch made a list of the services that a multi-agent system can offer in a knowledge management approach [39]:

- Knowledge search, acquisition, analysis and classification from various data sources.
- Information given to human and computing networks once usable knowledge is ready to be consulted.
- Negotiation on knowledge integration or exclusion into the system.
- Explanation of the quality and reliability which are related to the system integrated knowledge.
- Learning progressively all along the knowledge management process.

Such services are mostly implemented to create two MAS categories devoted to knowledge management. The first MAS type is based upon an agent cooperation to solve complicated problems related to the knowledge types. The second MAS category gathers management assistant agents depending on the actors' needs. We describe these categories in the following subsections.

3.1. MAS used in knowledge engineering

In this range, agents are expected to be flexible, proactive and reactive regarding user requirements [56,14]. In other cases, this feature is completed with the agent's ability to run distributed data and solve difficulties such as knowledge distribution cooperation in a community of practice [32].

Some of these MAS were created as complementary tools in information management (workflows, ontologies, information research systems, and so on) to design platforms such as FRODO [1], CoMMA [24], Edamok [8] and KRAFT [46]. All these works have pointed to the 'Multi-Agent Information System' or MAIS. A MAIS is a multi-agent system whose functions manage and use distributed information [15,16,24]. Moreover, access authorization, data upgrading and compiling heterogeneous information are some of the MAIS capacities.

In addition, Van Elts in [58] suggests to take into account the collaborative dimension of a domain along with the actors' needs and goals in the same domain. This approach is known as 'Agent Mediated Knowledge Management' or AMKM. AMKM agents are defined in *agent organizations* with a specific description of their roles and configurations that enable interactions. These organizations make knowledge management easier in dynamic environments. It is therefore the first contribution towards the importance of collaborative and social aspects in a domain for MAS specification dedicated to knowledge management.

Thanks to this, the system is able to calculate how much knowledge to capitalize, and to anticipate the actors' needs when they carry out their professional activities. Agent organization modeling is one approach to the MAS specifications. We propose to use this approach in order to define our knowledge-based system.

However, even if agent organizations make it possible to take into account the social and collaborative aspects of the project teams, we have to provide these agents with the capability to handle knowledge. This functionality can be performed if the MAS uses a ontologie to understand the knowledge world. In the next section we will present some research works using the ontologies for knowledge exploitation.

3.2. Ontologies used in MAS to handle knowledge

3.2.1. Ontologies to help knowledge modeling

Knowledge created and used in engineering projects comes from the interpretation of the professional actors having a collection of technical data in a given activity [59].

Ontology is an object of Artificial Intelligence that recently came to maturity, and a powerful conceptual tool of Knowledge Modelling [7,25]. It provides a coherent base to build on, and a shared reference with which to align, in the form of a consensual conceptual vocabulary on which one can build descriptions and communication acts.

Thus, knowledge created in engineering projects needs to be defined precisely in order to be useful in an information system. Ontology provides a vocabulary and semantics that make it possible to process knowledge related to a specific domain. Ontology gives definitions and indicates how concepts are connected to each other. These connections form a structure on the defined domain and clarify the possible meanings of the concepts [57]. Therefore, domain ontology has the specific concepts of a given domain. The ontology describes their entities, properties and the way they can be related to each other. These ontologies are meant to be reused in the same domain, but in new and different applications. They are said to be contextual [10] when the concept properties evolve according to the situation.

3.2.2. Interests of the Ontologies in MAS

The idea of using domain ontologies in an agent system aims at reusing parts of the domain knowledge in order to lead agents to share their information. In a MAS, several agents interact or work together to carry out common goals [55]. The coordination between agents is successful if each agent possesses the knowledge useful for achieving global goals, and provides a section of the knowledge world which is essential for the agent carrying out his tasks [35].

Some research projects, such as those of Buccafurri et al. [13] and Wooldridge [60], use the ontology for providing the agents with an internal representation of the interests and the behaviors of their associated human users. Other research works use ontology to help agents to choose the most promising agents to be contacted for knowledge-sharing purposes [11,12]. Generally, these systems are designed not to allow some agents to have access to the ontologies of other agents; they ensure an individualistic view of the agent societies. This is the viewpoint of most of the so-called BDI (Belief, Desire and Intention) approaches [49,29].

Another interesting approach is to design agents that are able to automatically build their ontologies by monitoring the users' actions [4]. In this approach, the agents are able to automatically extract logical rules which represent the user's behavior from the user interests that are described in the ontology.

In addition, Guerin and Pitt [30] and Singh [53] designed their MAS by adopting a 'social' view of the agent communities, where it is assumed that the ontology of each agent is, even partially, accessible to the other agents. In this paper, the proposed MAS called KATRAS is composed of a common ontology used by the agents to perceive the whole knowledge of the user's domain. In the next section we describe the overall structure and the components of KATRAS.

4. Overview of our approach

As explained above, information and knowledge used by the professional actors in engineering projects have to be managed (capitalized, structured, evaluated, reused) to assist professional actors in putting forward good practices or avoiding repeating mistakes. We propose to use a social and collaborative approach based on the modeling of the roles and the collaboration between actors throughout the project in order to design a new knowledge-based system. We believe that knowledge is created, shared and used through the professional activities and the interactions between the actors when they are playing specific roles. The study of these roles and interactions enables the building of an organizational model to favor knowledge capitalization and reuse throughout all the project.

We have created an organizational model of the product development process (called **OrgaDesign**) [43], which describes the role of the agents, their interaction and competences, as well as their knowledge. This knowledge is updated in all the professional activities taking place during an engineering project. Thus, OrgaDesign allows the agents to have a map of knowledge being generated, used and shared.

While using this knowledge map, the agents are able to give context to the knowledge, according to the characteristics of the collaborative professional activities (professional actor roles, activity, objectives, needs, competences). We propose to structure the domain concepts through a project memory model called **MemoDesign** [19]. This model helps agents to organize and to represent knowledge. In the following, we will explain how agents are able to generate the knowledge book from the project memory MemoDesign.

Our approach is completed by a definition of the concepts, their relations and attributes, which constitutes the domain ontology of a design project called **OntoDesign** [44]. The semantics and the vocabulary described in the ontology allow the agents to handle and to deduce some inferences with the capitalized knowledge.

Thus the proposed knowledge-based system (Fig. 1) concerns the design of a multi-agent system which is able to ensure the knowledge management process by using the three components presented above:

OrgaDesign: The organizational model of the engineering projects, which describes the roles, competences and interactions between professional actors within the professional activities. This model helps the agents to capitalize, annotate and reuse knowledge;

MemoDesign: The project memory model, which provides a structure for the knowledge representation. The agents use this model to give a representation of the knowledge based on a project memory;

OntoDesign: The domain ontology, which allow the agents to exploit and handle knowledge;

The multi-agent system called ***KATRAS*** (Knowledge Acquisition Traceability by Agents System) aims at capitalizing the structure, updating and evaluating the knowledge. Before describing the MAS architecture, let us introduce more information about the three above-mentioned models in the following sections.

4.1. Modeling the project organization

4.1.1. An organizational metaphor

In order to understand the knowledge management process during a engineering project, we have used an organizational approach. The notion of organization is not recent. Galbraith [23] explains that an organization is composed of entities working together to carry out a shared objective, to distribute their tasks and to set up decisional processes in time. Dignum et al. [17] highlights the social aspect of organizations by adding that it is represented by a set of entities and their interactions, which are regulated by mechanisms of social order and created by more or less autonomous actors in order to achieve common goals. Therefore an organization is built around steady behaviors and interactions. Thus the modeling of an organization makes it possible to highlight the interactions between professional actors and knowledge they have to use in order to carry out their objectives.

We focus on the professional actors' roles within the organizations. Professional actors may have one or several roles at a time in the projects. When they play their roles, they apply a knowledge management process, i.e. they create, use and share their professional knowledge. Thus each role located in an organizational context has one or several sub-roles dedicated to managing its knowledge. Consequently, an organizational model of the

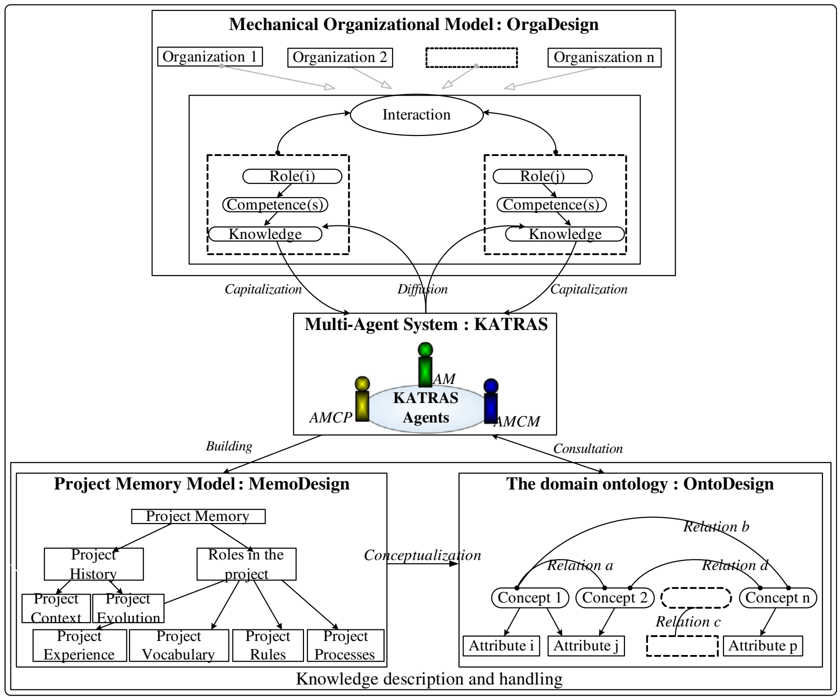

Fig. 1. The four components of the proposed knowledge-based system.

engineering projects allows us to highlight the knowledge used by the professional actor.

4.1.2. The RIOCK organizational model

The first step of our work is to understand, to analyse and to model the engineering projects. Thus we have followed several projects in an experiment of knowledge management deployment, during more than one year, in an SME company specialized in the development of rolling shutter mechanisms.

Our first experiment of knowledge management deployment was based on the MASK (Method for Analysing and Structuring Knowledge) method [41] to formalize knowledge. This method is a guide to structure the knowledge from the interviews of the professional experts. However, after a few months the experts explained to us that they do not have enough time to formalize their knowledge by using the MASK method during the project. Thus we thought about a new approach to assist experts in formalizing their knowledge and extracting it in a semi-automatic way by using a modeling approach to their activities.

With regard to this experience, we have analysed several design activities and we have validated the product development lifecycle with four phases: the feasibility study, the preliminary study, the detailed study and the manufacturing engineering. Each phase is structured according to some usual recurring activities for all projects led in the company. Each phase requires professional actors from different specialties and each activity can be carried out simultaneously.

In order to understand and model the lifecycle, we use the RIO formalism [33]. It is based on three concepts: Roles, Interaction, and Organization. We consider the project and its stages as RIO organizations. Inside, roles are generic behaviors. These behaviors can mutually interact according to an interaction pattern. Such a pattern, which groups generic behaviors and their interactions, constitutes an organization. Agents (human in this case) instantiate an organization (roles and interactions) when they exhibit behaviors defined by the roles of the organization, and when they interact by following the organization rules [1].

Moreover, the RIO formalism proposes a heritage of roles and organizations. An organization can also be seen as a participant in an interaction which is using other entities. Singh [54] suggests abstracting an organization and considering it as a role in another organization.

The project with its lifecycle is seen as an organization containing several sub-organizations, called phases and activities. Thus, sub-organizations are dependent upon each other since they belong to the same organization. Consequently, each lifecycle activity is an organization, which is able to be divided into subsidiary organizations.

In the design process, engineers do indeed have and share their knowledge to achieve tasks in a collaborative way and also develop learning issues by the capitalization process. Consequently, with RIO we attempt to identify the knowledge used by professional actors.

From different experiences and observations made inside the company, we define several roles for each organization.

We attribute to those roles the competences they use to fulfil tasks of the stage. The competence is defined at the individual level [37]: “it is the capacity of an individual to implement his knowledge and to develop his know-how within a professional framework”.

The concept of competence associated with a role allows the selection of Knowledge by professional fields, since a role can belong to two different professional fields.

Table 1 presents different types of Knowledge corresponding to two different roles belonging to two different professional fields. These roles have the same competence for a specific stage of the project. Competences are related to professional fields even if one competence can belong to two different professional fields.

Each competence is described with a set of knowledge. The interaction between several roles highlights two types of results: exchanges between professional actors and the emergence of knowledge. Thus, in an organization, a role uses one or more competences, which require one or more items of knowledge. A role interacts with other roles in order to achieve a task, develop the collaborative work and thus create its result. We extend the RIO

Table 1
Association between professional fields and roles.

Prof. fields	Competence	Role	Knowledge
Engineering and design department	To choose the good material for the product	Plastic injection mold engineer	(i) Design of the plastic injection mold (ii) Constraints of the product to be injected
Plastic injection unit	To manufacture the product	Plastic injection unit technician	(i) Use of plastic material
...	...	...	...

formalism by adding the concepts of competence and knowledge and we obtain the **RIOCK formalism** (Fig. 2).

In the activity (which can be modeling as an organization) 'to industrialize the selected concept' we observe three roles (Fig. 3). The 'Mechanical engineer' role uses two competences; we read it like the capability to 'To design the concepts' and 'To sketch the concepts'. Each of these competences requires a knowledge element ('Specifications of the concepts' and 'Geometry of the concepts'), which is used to satisfy the organization.

In the RIOCK diagram the knowledge type is read as "Knowledge on"; for example, the 'Plastic injection mold engineer' role has the Knowledge on 'Constraints related to the material and the product'. In addition, RIOCK presents the knowledge resulting from the collaboration among these three roles; here this is the industrial process to manufacture the new product.

The organizational model specifies for each activity the roles played by the professional actors, their interactions, the competences and the knowledge they have used in order to carry out their tasks. OrgaDesign allows the agents to identify, capitalize and reuse the knowledge during the projects.

4.2. Managing knowledge with the project memory model MemoDesign

In order to present a knowledge classification, we have regrouped of the knowledge identified in the organizational model in six groups: the ProjectContext, the ProjectEvolution, the ProjectProcess, the ProjectGlossary, the ProjectExpertise and the ProjectExperience. Each group represents a type of knowledge described in Table 2.

On the other hand, in Fig. 4 we have represented the UML (Unified Modelling Language) model of the project memory MemoDesign with the six knowledge elements regrouped in two categories: the characteristics of the project and the professional competences. This last category exists for each competence used in the project.

4.3. Making comprehensible knowledge with the domain ontology OntoDesign

In order to make the information and the knowledge (stored in the project memories) understandable and easy to handle by the agents, we have developed an ontology for the mechanical design projects. It is a conceptualization of the elements of the project memory model, with a definition of a taxonomy, and a specification of their relations and attributes.

We give a unique name for the relations (i.e. the link between two concepts) by specifying domain and range.

The second part of this work is to associate a set of attributes with each concept. Fig. 5 presents a graphical view of the ontology OntoDesign with the relations, the attributes and the concepts associated with the project evolution and the project process.

We have specified the concepts of the project memory and their relationships in the ontology OntoDesign with Protégé 2000 [47]; the purpose was to visualize, validate and build our ontology in the OWL language in conformity with the W3C recommendations.

The Protégé OWL editor supports OWL-DL language except for anonymous global class axioms, which need to be given a name by the user. Thus we have developed our ontology in OWL-DL with this tool. OWL-DL is based on Description Logics (hence the suffix DL). Description Logics are a decidable fragment of the First Order Logic and are therefore amenable to automated reasoning. It is therefore possible to automatically compute the classification hierarchy and check for inconsistencies in an ontology that conforms to OWL-DL.

Consequently, OntoDesign provides an integrated conceptual model for sharing information related to a mechanical design project. An OWL property (Fig. 5) is a binary relation to relate one OWL Class (Concept in OntoDesign) to another, or to the RDF literals and XML Schema data types. For example, the "infoInput" property relates the Document class to the Activity class. Described by these formal, explicit and rich semantics, the domain concept of Activity, its properties and relationships with other concepts can be queried, reasoned or mapped to support knowledge sharing across the design projects.

5. Architecture of multi-agent system KATRAS

In this section we present the agent model that supports the KATRAS system. We then proceed with the design process and the final architecture of the system, starting from the societal level down to the agents' behaviors.

5.1. Design of the MAS supporting the knowledge management process

We have seen previously that an organization is a set of entities and interactions between entities; these interactions are regulated

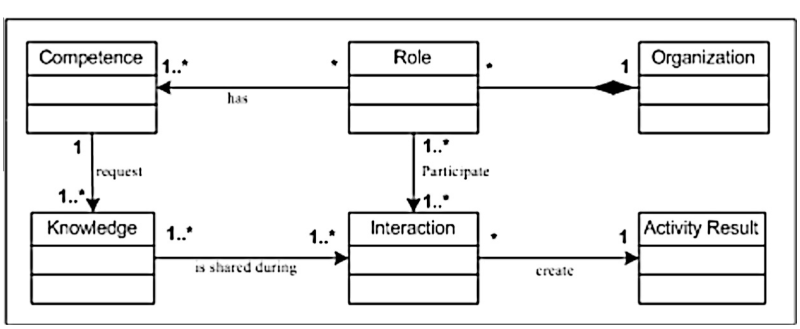

Fig. 2. RIOCK model.

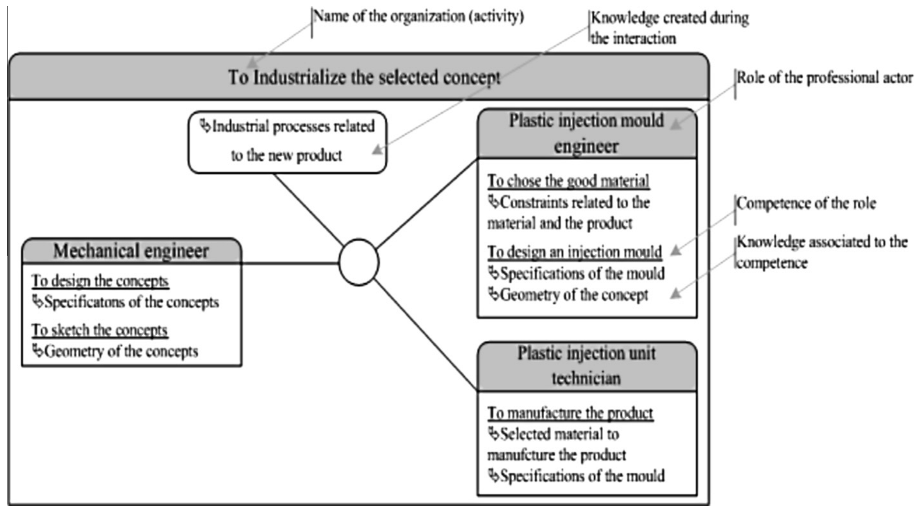

Fig. 3. Modelling of the organization "To industrialize the selected concept".

Table 2
Knowledge regrouping.

Name of the knowledge type	Knowledge
Context of the project (*ProjectContext*)	– Knowledge presenting the origin of the project – Knowledge describing the organization of the project
Evolution of the project (*ProjectEvolution*)	– Knowledge related to the history of the evolution of the project
Professional processes set up in the project (*ProjectProcess*)	– Knowledge presenting the activities carried out, the interventions of the professional actors and the information handled for each activity
Glossary of the project (*ProjectGlossary*)	– Knowledge defining the vocabulary used during the project
Expertise in the project (*ProjectExpertise*)	– Knowledge related to the professional rules used to develop the product
Experience developed in the project (*ProjectExperience*)	– Knowledge describing the errors, failures and difficulties encountered in the project

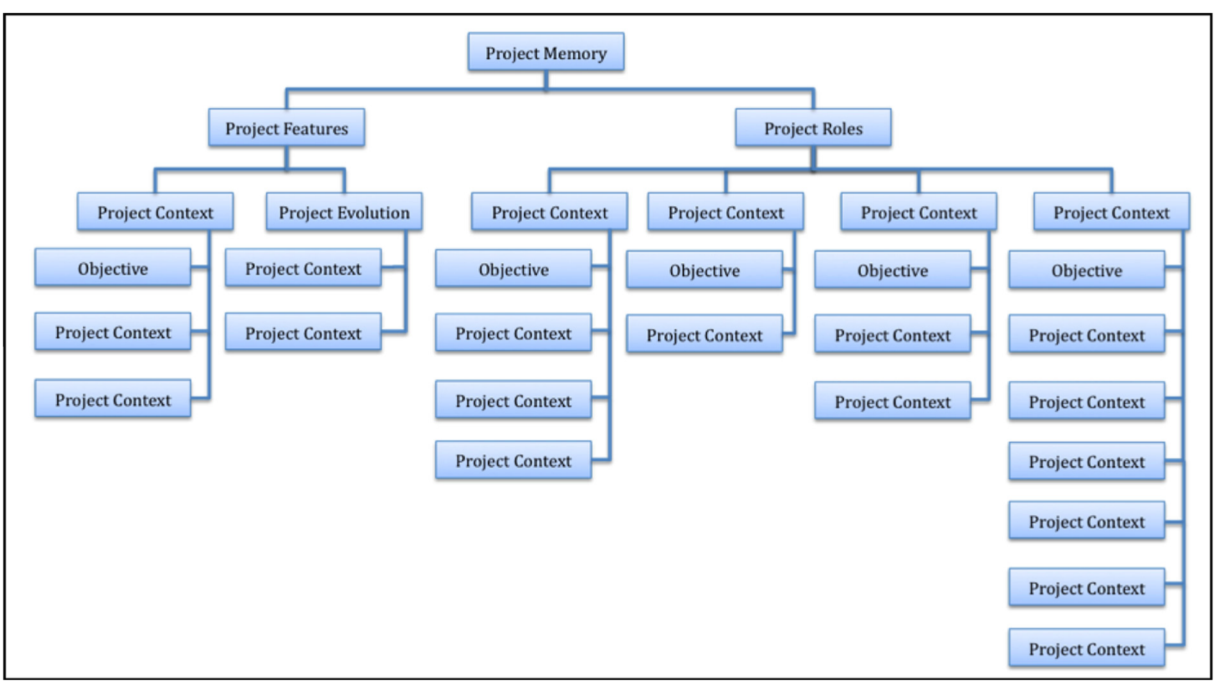

Fig. 4. The project memory model MemoDesign.

by mechanisms of social order and created by autonomous actors in order to achieve common goals. Thus, professional activities can be seen as organizations where engineers from different professional fields work together to reach the same objective: to develop a new product.

Consequently, we use an organizational approach to design the MAS, i.e. a specification of the agents' roles and their interactions inside the organizations.

In order to define the roles of the agents dedicated to the knowledge management process, we have studied the cycle proposed by

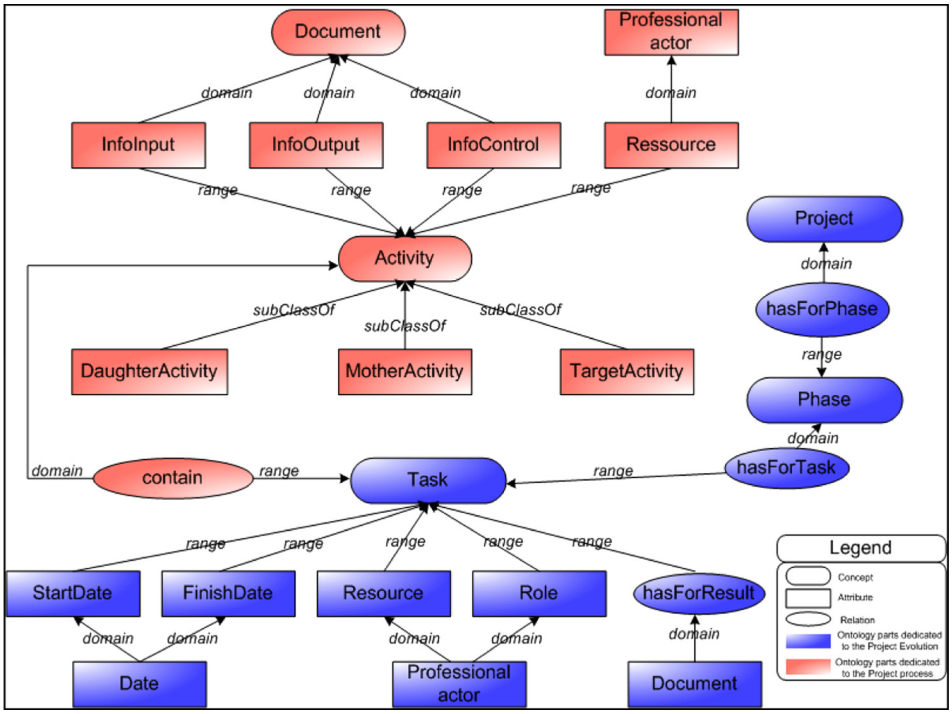

Fig. 5. An extract of the ontology OntoDesign.

Grundstein [31], which includes four steps: *knowledge identification, knowledge updating, knowledge valuing and knowledge preservation*. Each step has its own objectives. Thus we propose to associate with each step some roles dedicated to the achievement of these objectives (identification, updating, etc.). In addition, we associate with each role a set of activities to carry out the knowledge management task. For example, the role of 'knowledge user' has to reach, diffuse and share the information and the knowledge. Fig. 6 describes the different roles of the agents with their actions to realize.

In order to help professional actors during their projects we have to give them assistance in reusing related knowledge from previous projects and from the current project. Thus we propose three types of agents:

- The agents dedicated to the knowledge identification and to the user assistance, called **Professional Agents** (**PA**). They have the roles of 'knowledge identifier' and 'knowledge user'.
- The agents who manage the knowledge of the project, called **Project Knowledge Manager Agents** (**ProjKMA**). They have the roles of 'mediators', 'knowledge reasoners' and 'project memory creators'.
- The agents responsible for the knowledge management from previous projects, called **Professional Knowledge Manager Agents** (**ProfKMA**). These agents have the same roles as the ProjKMA.

We detail these three types of agents in the next section.

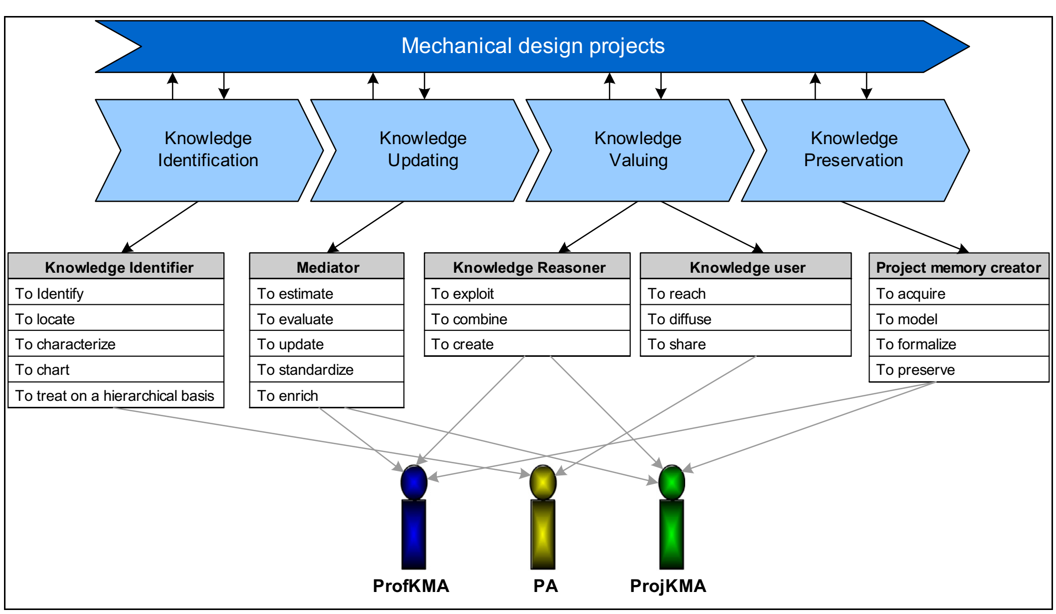

Fig. 6. Agents' roles dedicated to the knowledge management process.

5.2. Roles, organizations, interactions and behaviors

From the analysis of the knowledge management cycle, we have derived the characteristics of the agent roles that have been identified; we therefore implement the corresponding behaviors in three types of agents (Section 5.1). We consider that the role is the abstraction of a behavior in a certain context and confers a status within the organization.

The structure of the organization determines important autonomous activities that must be explicitly organized into autonomous entities and relationships in the conceptual model of the agent society. KATRAS agents have to constitute organizations that will ensure the knowledge management used by the professional actors.

Consequently, agents have to monitor the organization of the professional actors in order to identify, capitalize and propose a knowledge reuse. They use the organizational model to carry out this aim. Thus, for each organization of the professional actor (i.e. professional activity in a mechanical design project), which was defined in the organizational model OrgaDesign, the agents build two sub-organizations. Fig. 7 describes these two organizations.

The first sub-organization is related to the knowledge management by the professional role inside the current project. The second sub-organization concerns the knowledge management resulting from all projects. Therefore these two sub-organizations highlight the knowledge management process.

Software agents have to monitor the activities of the professional actors when playing roles that have been modeled in the organizational model; this would ensure the knowledge management process. Fig. 7 shows the overview of the agent system:

- The human level where professional actors interact together in order to carry out the design activities.
- The dynamic organizational model related to the human interactions.
- The two sub-organizations describing the agents' properties (role, interaction, etc.).
- The agent level specifying the type of agents which support the knowledge management process during each professional activity.

In the agent organizations we observe the five roles which ensure the knowledge management process. These five roles are played by the three types of agents: ProjKMA, ProfKMA and PA.

For each human organization, the three types of KATRAS agent interact together inside two sub-organizations in order to manage knowledge. In Fig. 7 we observe that the Professional Agents interact with both other types of agents and take part in the two

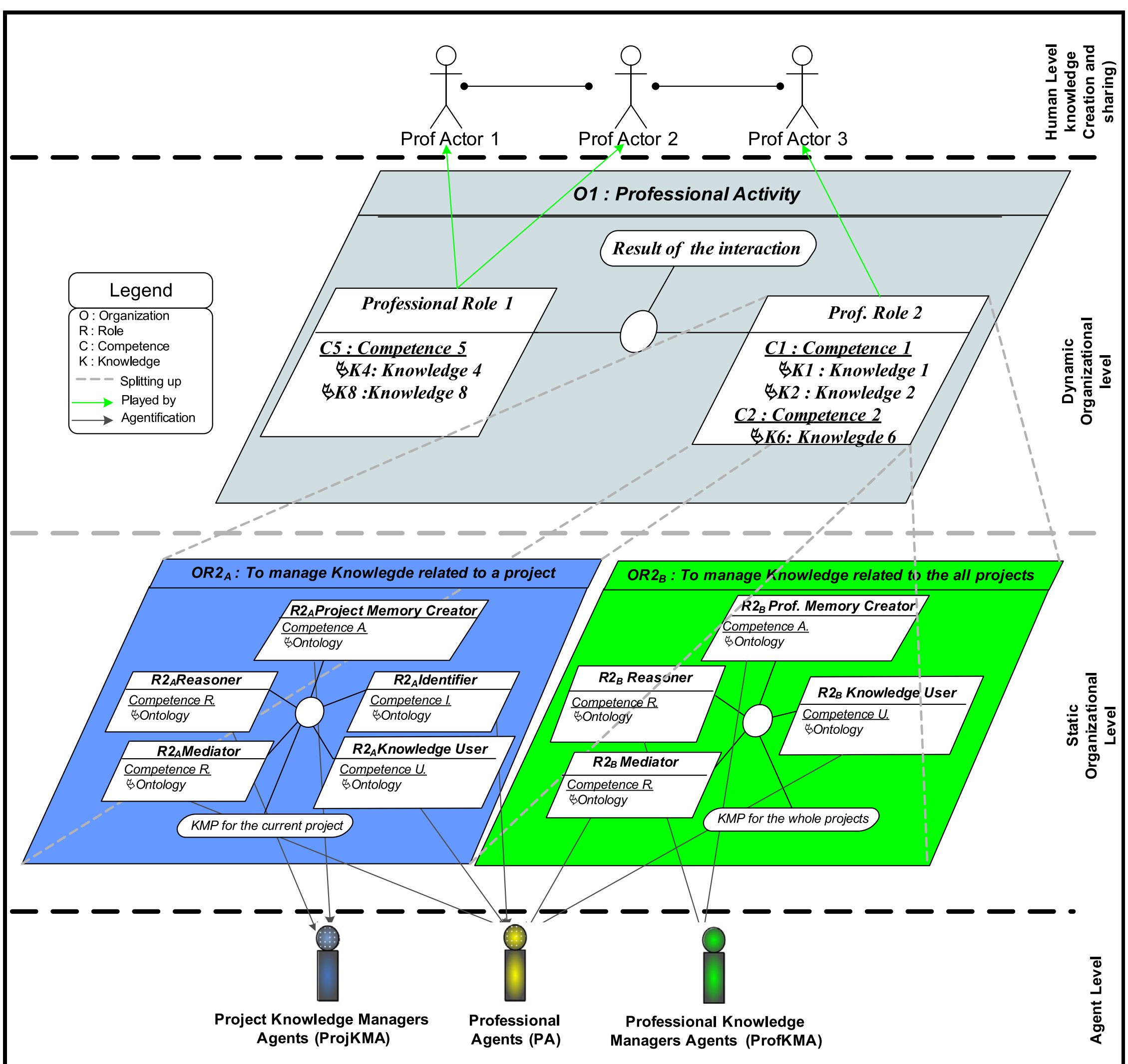

Fig. 7. Agents' organizations and human organizations.

sub-organizations. There is one PA for one professional actor. There are six ProjKMA for one project, each one corresponding to a knowledge type defined in the project memory model. These agents are deleted at the end of a project, and the project memory is sent to the ProfKMA to be managed with the other project memories of previous and finished projects. There are six ProfKMA for all the projects, each one corresponding to a knowledge type defined in the project memory model. In Table 3 we detail the profile of the agents.

In the next section we describe how the agents use the three models (OrgaDesign, MemoDesign and OntoDesign) in order to manage knowledge during the projects.

6. Agents in a knowledge world

6.1. Knowledge identification & annotation

The Professional Agents (PA) are able to identify the knowledge to acquire thanks to the design organizational model, OrgaDesign. Indeed, this model describes the roles of the professional actors, their interaction, their competences and the knowledge they use and share for every professional activity. The PA seeks the six knowledge types for each activity carried out by their corresponding professional actors. Fig. 8 presents three human actors carrying out an activity. In this figure, we have represented the three Professional Agents that monitor the actions of the three human actors. Those agents identify the knowledge created in the activity (step ❶, Fig. 8). They have the "Identifier" role. For example, if the three humans work on the activity "to industrialize the selected concept", and they have the roles of Mechanical Engineer, Plastic injection unit technician and Plastic Injection mold engineer, they constitute the organization described in Fig. 3. Thus there will be three Professional Agents (corresponding to the three roles) which follow the actions of the human actor by using the OrgaDesign Model. This model defines for the agent the knowledge to extract in the database. Here the PA which monitors the role of "Mechanical Engineer" will search the knowledge related to the "Specification of the concept" and "the geometry of the concept" since this knowledge is specified in the OrgaDesign model for this organization and this role. As soon as the PA identifies the knowledge, they annotate it (step ❷). The annotation defines the organizational context of the knowledge, i.e. the role which, and the activity and the project where the knowledge was created. An example of organizational context is given in Fig. 10. Afterwards, the knowledge with the annotations is sent (step ❸) to the Project Knowledge Manager Agents (ProjKMA). In the example of the organization "to industrialize the selected concept", the three PA send the knowledge that they have captured to the ProjKMA.

Knowledge is distributed according to the competence of each ProjKMA. There are six ProjKMA and each of them is able to manage the knowledge from one of the six knowledge types. The ProjKMA build the project memory progressively during the project with the annotations sent by the PA. The project memory is created according to the MemoDesign model described in the OntoDesign ontology.

Moreover, during an activity, the PA are able to provide knowledge reuse assistance by proposing knowledge already capitalized. Indeed, PA consult the OntoDesign ontology in order to create the queries they send to the ProjKMA (step ❹). Those agents respond to the request by sending knowledge capitalized in the project (step ❺).

In the next section we describe some of the agents' knowledge exploitation mechanisms.

6.2. Knowledge exploitation

The KATRAS agents use the OntoDesign ontology for annotating the knowledge according to the definition of the concepts, relations and attributes. The Professional Agents build the annotations with the concepts 'Project', 'Activity' and 'Role' in order to specify for each item of knowledge the organizational context where it was identified. Afterwards this context is described with the triplet Project–Activity–Role. Fig. 9 presents an extract of the OntoDesign where these concepts and their relations are specified, allowing the knowledge annotation by the agents.

Using the RDF format, the Professional Agents build the knowledge annotations, which are some instances of the ontology. These annotations allow the description of the knowledge by giving the context where the knowledge was created. Each annotation is composed of two parts. The first part is an address (URI) where the information was capitalized with its description. The second part describes the organizational context defined with the name of the project, the activity where the knowledge was used and the role of the professional actor. Fig. 10 describes the annotation of the professional rule "MechanicalDesignRule65" with a URI in the first line where the knowledge was captured ("http://utbm.acsp.fr/SIAChallenge/MechanicalDesignRule65"). And the end of

Table 3
KATRAS agents' profiles

Agent type	PA	ProjKMA	ProfKMA
Roles	– Knowledge identifier – Knowledge user	– Mediator – Knowledge reasoner – Project memory creator	– Mediator – Knowledge reasoner – Project memory creator
Interactions	– PA to research and share knowledge – ProjKMA to store knowledge – Professional actors to reuse knowledge	– PA to store the knowledge annotations – PA to propose solutions to the human actors – Human actors to validate knowledge	– ProjKMA to store the project memory of the finished projects – PA to propose solutions to the human actors – Human actors to validate knowledge
Responsibilities	– To seek the 6 knowledge types – To annotate knowledge – To propose knowledge reuse assistance	– To store knowledge according to the project memory model – To ensure knowledge reliability	– To ensure knowledge reliability – To infer with knowledge detains in the whole project memories of finished projects
Expertise	– OrgaDesign model to identify knowledge – Information network to seek knowledge – OntoDesign ontology to annotate knowledge	– OntoDesign ontology to know the project memory structure – knowledge validation cycle – inference rules to handle knowledge	– OntoDesign ontology to know the project memory structure – knowledge validation cycle – inference rules to handle knowledge
Number of agents	One agent for one professional actor	Six agents for each project; one agent by knowledge type	Six agents for all projects, one agent by knowledge type

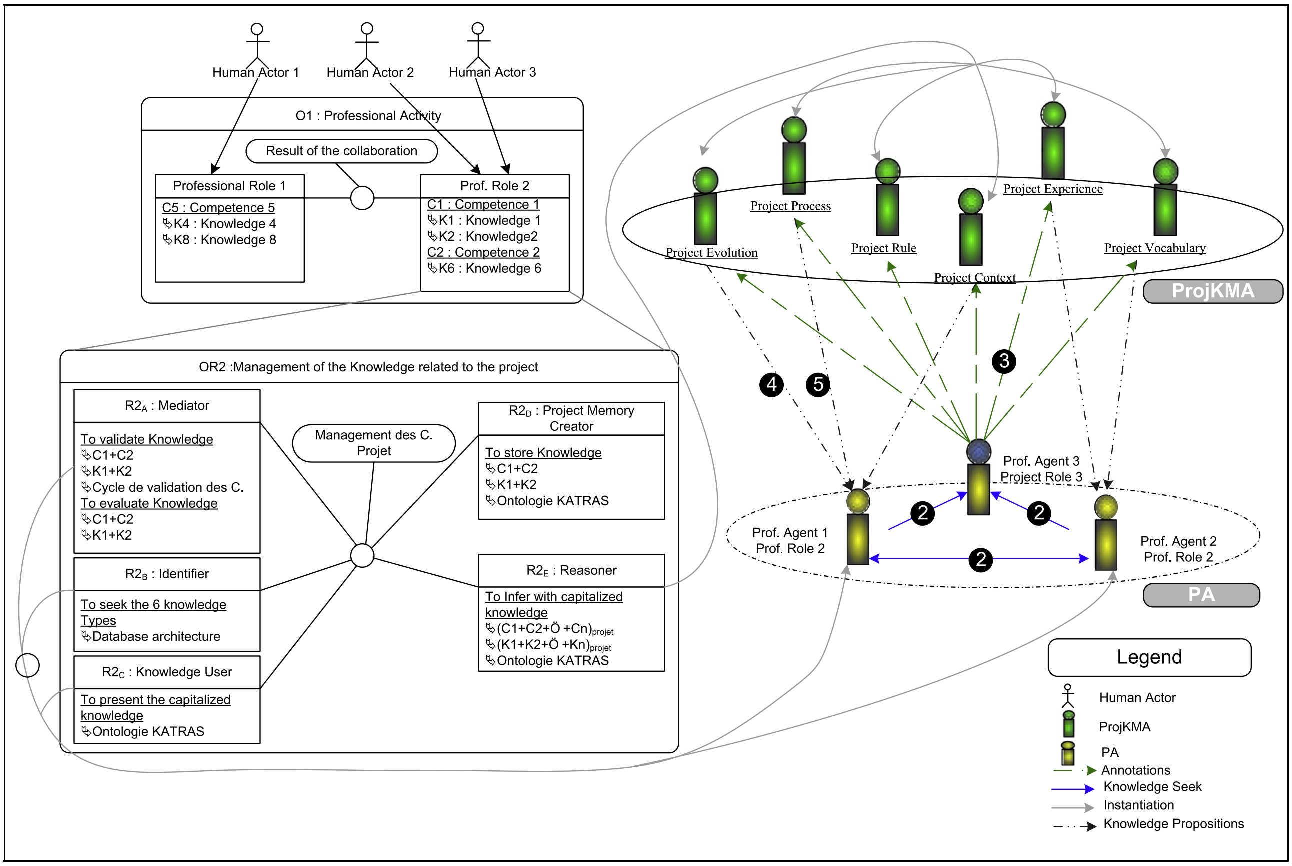

Fig. 8. Communication between agents.

the annotation is defined by the knowledge and the organizational context describing the knowledge origin.

6.3. Active assistance for knowledge reuse

The design of the MAS KATRAS was carried out from an organizational approach based on the mechanical design process (Section 5.2). As a matter of fact, the professional agents ensure the knowledge management process by rebuilding organizations during the professional activities.

They know the organizational context (Project/Competence/Role/Activity) where the actors work. Thus, the agents are able to assist the professional actors by proposing knowledge already capitalized in similar organizational contexts during previous projects.

The search of the organizational context is based on the OntoDesign ontology. Fig. 11 presents the assistance protocol applied by the agents.

When a professional agent perceives the action of this actor (i.e. the activities he carries out) and his role, the agent sends a request to the Project Knowledge Manager Agents in order to search the knowledge stored for similar activities and roles.

In the case that the ProjKMA agents find some knowledge annotated with the same organizational context, the Professional Agent is able to provide assistance to the user by proposing the capitalized knowledge.

The active assistance for the knowledge reuse is ensured according to the following protocol:

❶ The professional agent perceives the actions of his professional actor during the project. The PA knows the activity realized by his actors and their role.

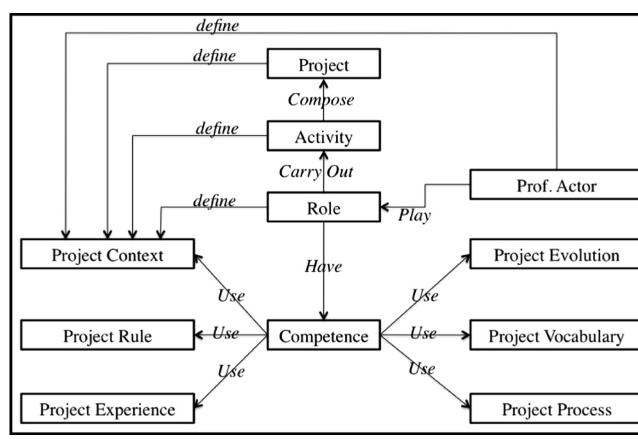

Fig. 9. Extract of the OntoDesign with the concepts used in the knowledge annotation.

❷ The Professional Agent takes on the role of "knowledge user" and sends a request to the six Project Knowledge Manager Agents. The PA transmits the organizational context (Role/Competence/Activity) to the six agents who are managing the six types of knowledge stored in the project memory. The ProjKMA take on the role of "Knowledge Reasoner".

❸ Every ProjKMA builds the query corresponding to the request of the PA according to its knowledge type. The queries are built in using the SPARQL language and the OntoDesign ontology. For example, the ProjKMA-

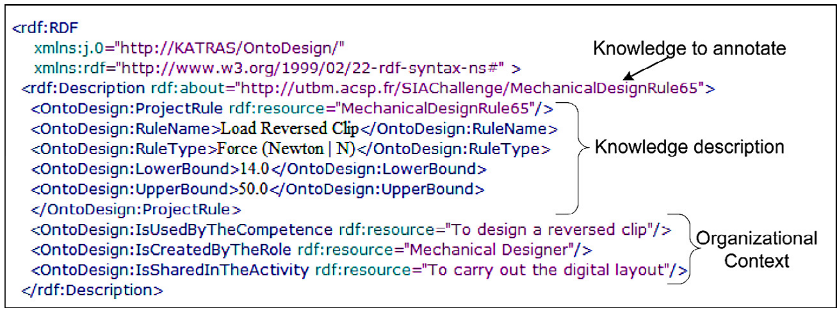

Fig. 10. Knowledge annotation generated by the agents.

ProjectRule searches possible professional rules which are capitalized in similar organizational contexts during past projects. The ProjKMA-ProjectExperience searches the capitalized experience, the ProjKMA-ProjectProcess searches the capitalized processes, and so on for the three other ProjKMA.

❹ The Professional Agent notifies the actor that there is some capitalized knowledge that he can use in his activity.

7. Industrial experimentation

7.1. Context

Our works are deployed in an SME company called Zurfluh–Feller of four hundred employees in the domain activity of window rolling shutters. The research and development department is constituted by fifty technicians. The method department, the laboratory department and the design and engineering department work together through a project organization in a concurrent engineering way.

One of the problems we have tackled is to enable professional actors to reuse their collaborative professional experience from previous projects. The directors of the company have decided to develop a knowledge engineering approach in building project memories to solve this problem.

In order to deploy a knowledge management approach, we have chosen to connect the proposed KATRAS system to a collaborative environment (e-Groupware) called PCW (Project Cooperative Workshop). Today, engineers in the company's Research & Development department are using this software to manage their projects. PCW is a PLM (Product Lifecycle Management) platform. Indeed, the engineers are using PCW to share information about three domains:

- The project with the management of the delay (Gantt), of the human resources and of the costs.
- The product with the management of the information about the product (design rule, calculus analysis, FMEA analysis etc.).
- The process analysis with the management of the production range and the production means.

All the information about the product and the professional activities to develop and industrialize it is saved in PCW so it is relevant to deploy KATRAS in this embedded environment.

7.2. Integration of KATRAS into the PCW collaborative e-groupware

The MAS is developed with the java platform Madkit [21] which allows us to implement agents with the notion of agent, group and

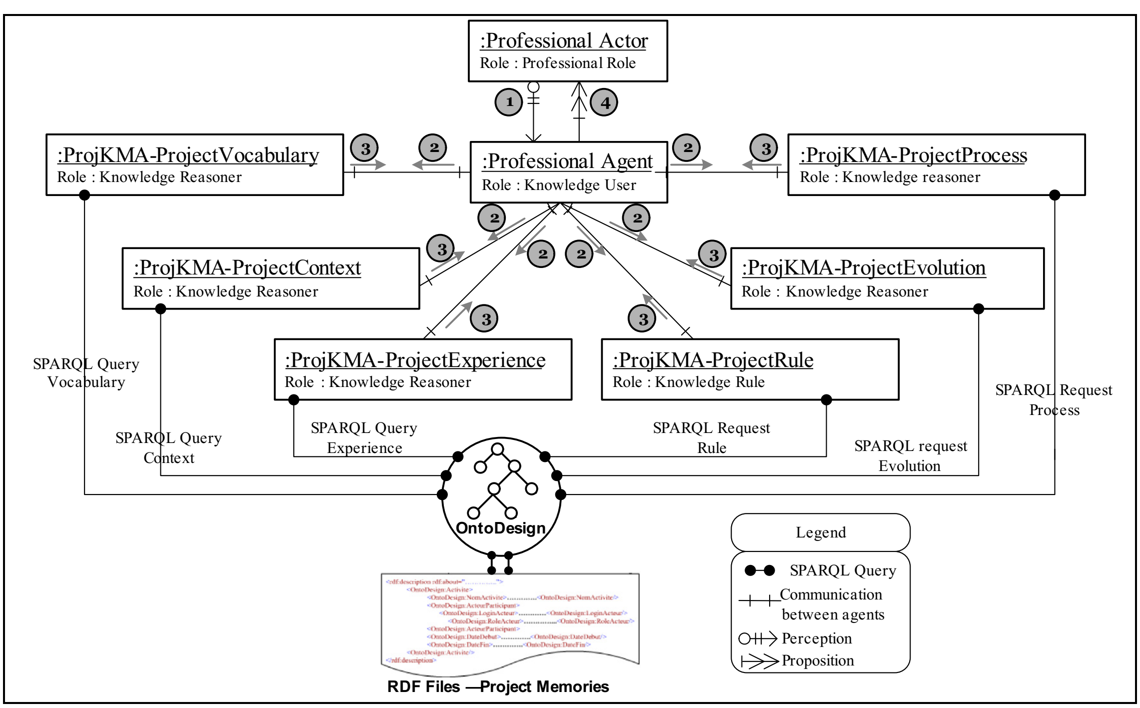

Fig. 11. Knowledge search protocol for users' assistance.

role. The Madkit platform manages the communication between agents by using the FIPA specification. Thus an agent can send a message to a group of agents or to the agents that have a specific role.

To handle knowledge, we use the Jena java API [51] which provides our agents with the possibility to perceive an ontology in OWL format. Moreover, Jena includes a rule-based inference engine based on the SPARQL language.

To test our approach, we have integrated KATRAS into the e-Groupware PCW. The professional actors use this platform to carry on their mechanical design project activities. It is a Web-based collaborative engineering environment, using a multi-domain and multi-viewpoint design model [28]. This Web-based tool was developed as an environment for computer-supported cooperative work (e-Groupware), in order to organize and structure the collaborative activities of designers from anywhere in the world. Thus the team project can use synchronous communication through PCW chat to organize remote meetings. In addition, designers can use the PCW Forum to declare the problems they encounter.

The PCW software interface, connected to a relational Database Management System, is divided into four main sub-modules managing information from the project, product, process and the usability design domains.

Each design domain includes various design data and files describing functional, structural and dynamic aspects of the studied domain. PCW features are used for the design chain data management: product data and information, documents and their associated content (all types, formats and media), requirements (functional, performance, quality, cost, physical factors, interoperability, time, etc.), etc.

Consequently, the e-Groupware makes it possible to capture the results of the professional activities in its database, including information on the project, product and process management. It offers a centralized environment with heterogeneous information (documents, data, digital mock-up, etc.).

We chose this collaborative environment to deploy the proposed knowledge-basedsystem KATRAS. There are two advantages to using the PCW platform: the first is that engineers are accustomed to working with it; the second is that the agents do not need to treat distributive information, since all the data is centralized in the database. Thus we have developed certain interfaces integrated in this environment in order to allow professional actors to consult the knowledge capitalized by KATRAS.

PCW was developed in ASP.Net technology with the MYSQL database. The KATRAS engine is deployed in the same server and communicates with PCW through two types of communications:

- The use of queries to detect new information stored in the MYSQL database and to send new information.
- The sending of information and knowledge by using the ASP.Net classes. Inside the e-groupware we have developed ASP.Net interfaces to broadcast knowledge managed by the KATRAS agents.

Fig. 11 shows the KATRAS Interface inside the PCW platform. The professional actors can search knowledge by typing in keywords and choosing the knowledge type (Experience, Context, Evolution, Rule, Vocabulary and Process). When the actor has read the knowledge, he can give a positive or negative evaluation. Thus the agents are capable of managing the quality and the reliability of the knowledge. Knowledge with less than 20% of positive evaluation will be deleted.

7.3. Knowledge reuse inside the e-Groupware

The integration of the knowledge-based system inside the e-Groupware is materialized by the addition of a "knowledge engineering module". This module makes it possible for the users to seek and consult the knowledge, which is capitalized by the KBS. The module is composed of interfaces dedicated to the knowledge consultation. Professional actors are able to consult knowledge of the current project (i.e. its project memory) as well as knowledge of all the projects (i.e. all the project memories).

There is one interface for each knowledge type (Fig. 12):

- The *project context* is represented by a form in which are described the objective, the environment and the organization of the project.
- The *project evolution* of the project is described by a planning.
- The *project rules* are represented inside a rule editor.
- The *project process* is described with an IDEF0 diagram [20].
- The *project terms* are represented in a glossary.
- The *project experiences* are represented in a form in which are mentioned the successes, the difficulties and the failures encountered during the projects. For this type of knowledge, the agents provide proactive assistance. Indeed, they inform the professional actors if there is a return of experience corresponding to a similar activity when they carry out one task.

7.4. Results of the experimentation and discussion

Nowadays, KATRAS is implemented inside the e-Groupware PCW. We observe that the different roles during the project use KATRAS differently:

- The project leaders use knowledge related to the 'Project Evolution' to begin a new project. This allows us to estimate the delay of a new product development from past projects in which the company has industrialized similar products.
- The production technicians use knowledge related to the 'Project process' to optimize their industrial processes. They appreciate the time this saves.
- Knowledge related to the 'Project terms' is only used by novice employees. It allows them to learn about the technical terms, the product references and the technical methods used in the company.
- The most important result is about knowledge related to the 'project rule'. Also, engineers reuse the professional rules associated with developed product parts in order to design a similar product. This functionality yields important time saving. To obtain this result the engineering development times (requirement analysis, feasibility study and design of the product) of six similar projects were compared. These projects were about the design of two simple rolling shutters. The engineers in the company used to develop rolling shutters. Thus, from this analysis the project leaders have estimated that in the case of routine projects, our knowledge management approach yields a gain of about 20% in time saving.

7.5. Related work and discussion

The contribution of the use of a Multi-Agent System in a Knowledge-Based System concerns proactivity. Indeed, the KATRAS agents can monitor the actions for the professional actors and can propose that they reuse knowledge by interpreting the context where the information was captured. Thus these agents are reactive since they knowledge capitalization tasks as soon as they have detected a new information stored by a professional actor. The reactivity is also the aim feature of the personal assistants proposed by Tacla [56].

Moreover, the KATRAS agents detect the organizational context of an actor and use the OrgaDesign model to propose knowledge already capitalized in a same context. These agents are pro active.

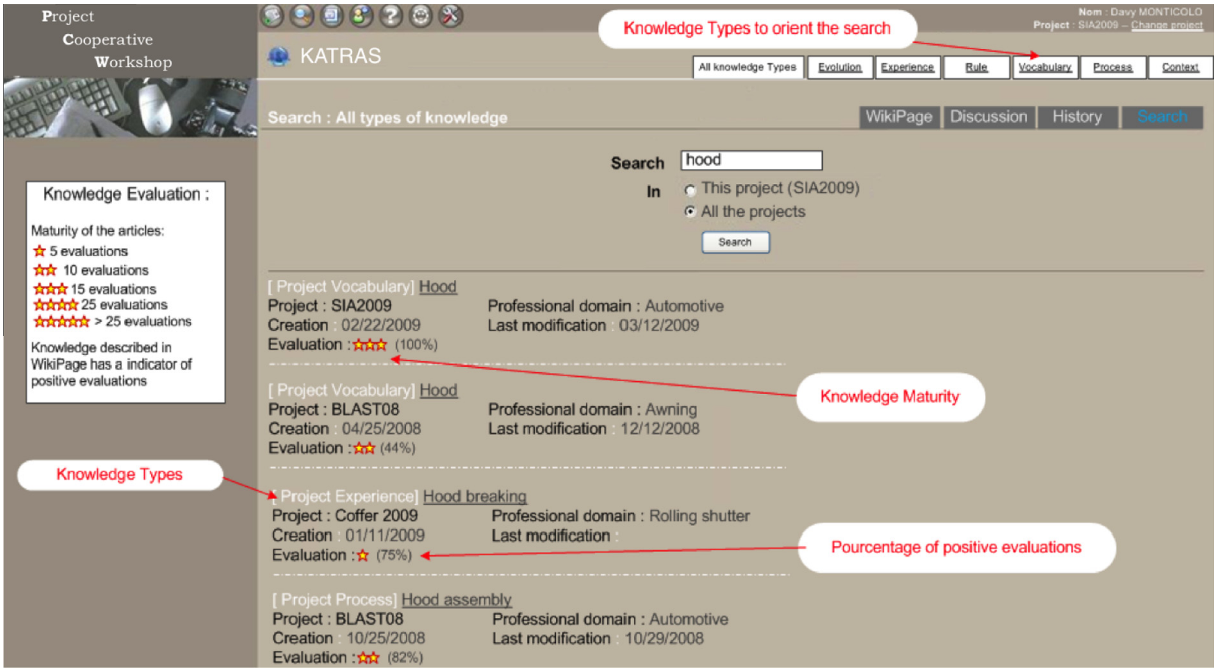

Fig. 12. Interface of the KATRAS knowledge search engine.

We observe these kind of agents in the FRODO [1,46] and KRAFT systems.

The knowledge management is a complicated process, which is support by the KATRAS agents by using an ontology. Indeed, the ontology allows them to annotate knowledge and to infer in order to find the good knowledge that the actors need. This feature provides the intelligence for these agents.

Other recent systems such as ONTOMADEM [26], KDSS [22] and REA [3] also use a MAS with an ontology to apply a knowledge management approach. These systems provide efficient knowledge treatment but do not focus on the proactive assistant to knowledge reuse (see Fig. 13).

The last feature of the KATRAS agents is to act by using a conceptual model of the behavior of the human actor. Indeed the OrgaDesign model describes the roles, the interaction between the actor and the competences and knowledge that they share. Thanks to this model, the KATRAS agents increase their intelligence in the knowledge sharing. They propose to re-use knowledge in a proactive way not only to one actor but also to all the actors who are likely concerns by the knowledge.

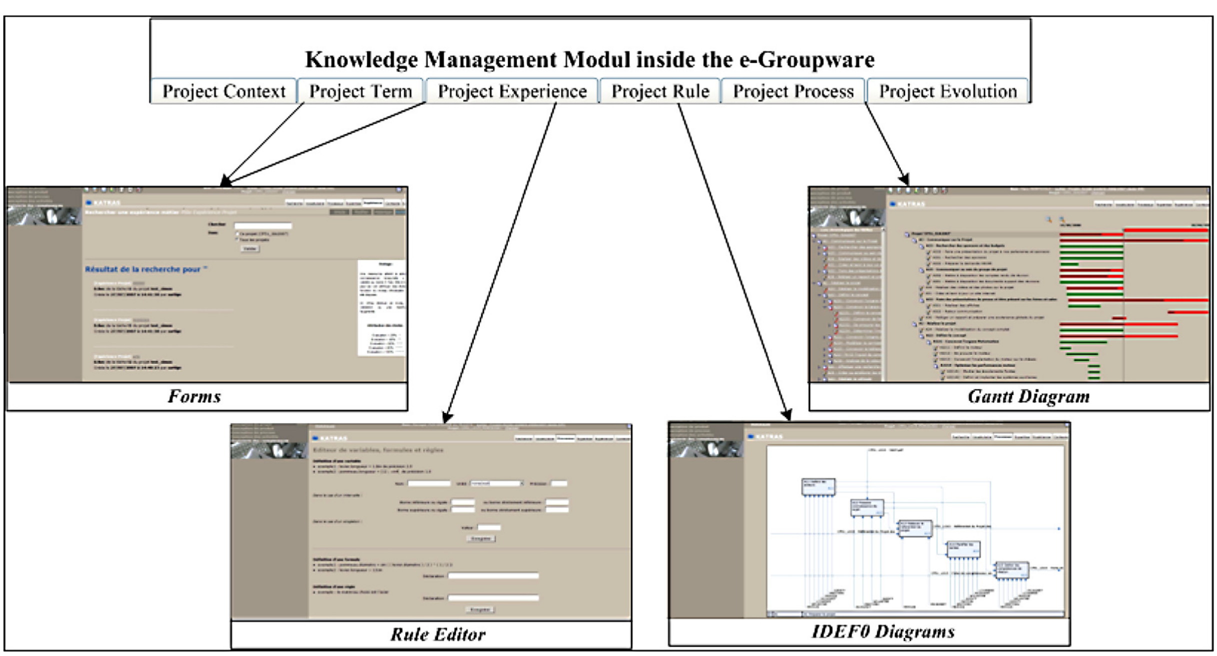

Fig. 13. Interfaces to consult Knowledge inside the e-Groupware.

All these functionalities make KATRAS like a complete intelligent multi agent system which is flexible and autonomous.

8. Conclusion & future work

Nowadays, KATRAS along with the e-Groupware is used in several companies. The agents assist the professional actors in reusing knowledge by proposing to consult and to visualize the project memory pages. The agents propose some links to the project memories. The knowledge research is done by keywords.

Now, we have to think about proactive assistance to guide professional actors throughout their project and provide a decision aid.

In order to do this, we have to implement the role of the "knowledge user" agent with the objective of learning knowledge needed by the users; we need to extend its knowledge world. In this way, we will work on the possibility of using several domain ontologies with KATRAS in order to increase the agents' assistance. We also work on the possibility that KATRAS will be plugged into several software applications, in order to capitalize all the concepts of the domain ontology. One solution to carry out this functionality seems to use the Semantic Web Services. They can use the semantics related to a knowledge domain in order to communicate between different software applications, to capitalize and broadcast the relevant information.

Acknowledgments

We would like to thank all the engineers of the research department of the Zurfluh Feller Company (SOMFY Corporation). Their uses of the KATRAS system and their reviews, comments and suggestions have helped us to improve the proposed knowledge-based system.

References

[1] A. Abecker, A. Bernardi, L. van Elst, Agent technology for distributed organizational memories: The Frodo Project, in: 5th International Conference on Enterprise Information Systems, vol. 2, 2003, pp. 3-10.
[2] A. Abecker, A. Bernardi, K. Hinkelmann, O. Kühn, M. Sintek, Towards a technology for organizational memories, IEEE Intell. Syst. 13 (1998) 40–48.
[3] K. Amailef, J. Lu, Ontology-supported case-based reasoning approach for intelligent m-Government emergency response services, Dec. Supp. Syst. 55 (1) (2013) 79–97.
[4] T. Atanasova, H.-J. Nern, M. Hamalainen, H.N. Eldin, Distributed heterogeneous knowledge database for control system design: multiagent development and support, in: Proceedings of the 2000 IEEE International Symposium on Intelligent, Control, 2000, pp. 363–368.
[5] F. Belkadi, E. Bonjour, M. Camargo, N. Troussier, B. Eynard, A situation model to support awareness in collaborative design, Int. J. Hum.–Comp. Stud. 71 (1) (2013) 110–129.
[6] S. Bekhti, DYPKM, Un processus dynamique de définition et de réutilisation de mémoires de projets, Ph.D. Thesis, Technological University of Troyes, 2003.
[7] G. Beydoun, N. Tran, G. Low, B. Henderson-Sellers, Foundations of ontology-based MAS methodologies, in: Agent-Oriented Information Systems III, Lecture Notes in Computer Science, vol. 3529, Springer, Berlin Heidelberg, 2006, pp. 111–123.
[8] M. Bonifacio, P. Bouquet, P. Traverso, Enabling distributed knowledge management: managerial and technological implications, Novatica and Informatik/Informatique 3 (1) (2002) 23–29.
[9] E. Bonjour, F. Belkadi, N. Troussier, M. Dulmet, Modelling interactions to support and manage collaborative decision-making processes in design situations, Int. J. Comput. Appl. Technol. 36 (3–4) (2009) 259–271.
[10] P. Bouquet, F. Giunchiglia, F. Van Hermelen, L. Serafini, H. Stuckenschmidt, C-owl Contextualizing ontologies, in: Proceedings of the 2nd International Semantic Web Conference (ISWC2003), Sundial Resort, Sanibel Island, FL, USA, 2003, pp. 164–174.
[11] R.M.M. Braga, C.M.L. Werner, M. Mattoso, Odyssey-search: a multi-agent system for component information search and retrieval, J. Syst. Softw. 79 (2) (2006) 204–215.
[12] M.C. Bravo, J. Pérez, V.J. Sosa, A. Montes, G. Reyes, Ontology support for communicating agents in negotiation process, in: Fifth International Conference on Hybrid Intelligent Systems HIS'05, 2005.
[13] F. Buccafurri, D. Rosaci, G.M.L. Sarne, D. Ursino, An agent-based hierarchical clustering approach for e-commerce environments, Proceedings of the Third International Conference on e-Commerce and Web Technologies 2455 (2002) 109–118 (EC WEB 2002).
[14] C. Castelfranchi, Engineering social order, in: Engineering Societies in the Agents World, Lecture Notes in Computer Science, Springer Verlag, 2000, pp. 1–18. 1972.
[15] P.A. Champin, Y. Prié, A. Mille, MUSETTE: modelling USEs and tasks for tracing experience, in: Proceedings of From Structured Cases to Unstructured Problem Solving Episodes – WS 5 of ICCBR'03, Trondheim (NO) NTNU, 2003, pp. 279–286.
[16] K. Chen, H. Wang, H. Lai, A general knowledge mediation infrastructure for multi-agent systems, Expert Syst. Appl. 38 (1) (2011) 495–503.
[17] V. Dignum, F. Dignum, J.J. Meyer, An agent-mediated approach to the support of knowledge sharing in organizations, Knowl. Eng. Rev. 19 (2) (2004) 147–174 (Cambridge University Press).
[18] R. Dieng, O. Corby, A. Giboin, M. Ribière, Methods and tools for corporate knowledge management, Int. J. Hum.–Comput. Stud., Spec. Iss. Knowl. Manage. 51 (3) (1999) 567–598.
[19] C. Djaiz, D. Monticolo, N. Matta, Project memory decision making, Int. J. e-Collab. Creativ., Innov. eCollab. 2 (3) (2008) 12–28.
[20] J.M. Dorador, R.I.M. Young, Application of IDEF0, IDEF3 and UML methodologies in the creation of information models, Int. J. Comput. Integ. Manuf. 13 (5) (2000) 430–445.
[21] J. Ferber, O. Gutknecht, F. Michel, From agents to organizations: an organizational view of multi-agent systems, in: Agent-Oriented Software Engineering – the 4th International Workshop, (AOSE-2003@AAMAS 2003), LNCS, vol. 2935, Melbourne, Australia, 2003, pp. 214–230.
[22] A. Fiannaca, M.L. Rosa, A. Urso, R. Rizo, S. Gaglio, A knowledge-based decision support system in bioinformatics: an application to protein complex extraction, BMC Bioinform. 14 (1) (2013) 1–14.
[23] J.R. Galbraith, Organization Design, Addison-Wesley, Reading, MA, 1977.
[24] F. Gandon, L. Berthelot, R. Dieng-Kuntz, A multi-agent platform for a corporate semantic web, in: Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agents Systems, USA, 2002, pp. 1025–1032.
[25] R.J. Gil, M.J. Martin-Bautista, A novel integrated knowledge support system based on ontology learning: model specification and a case study, Knowl.-Based Syst. 36 (2012). 340–252.
[26] R. Girardi, A. Leite, A knowledge-based tool for multi-agent domain engineering, Knowl.-Based Syst. 21 (7) (2008) 604–611.
[27] J.P. Girard, Building Organizational Memories: Will You Know What You Knew? Idea Group Inc Global, 2009.
[28] S. Gomes, J.C. Sagot, A concurrent engineering experience based on a cooperative and object oriented design methodology, in: 3rd International Conference on Integrated Design and Manufacturing in Mechanical Engineering, IDMME 2000, Montreal, 2000.
[29] N. Griffiths, M. Luck, Cooperative plan selection through trust, in: Proceedings of the 9th European Workshop on Modelling Autonomous Agents in a Multi-Agent World: Multi-Agent System Engineering (MAAMAW-99), vol. 1647, Springer, Berlin, 1999, pp. 162–174.
[30] F. Guerin, J. Pitt, Denotational semantics for agent communication language, in: AGENTS '01: Fifth International Conference on Autonomous Agents, ACM Press, 2001, pp. 497–504.
[31] M. Grundstein, C. Rosenthal-Sabroux, A. Pachulski, Reinforcing decision aid by capitalizing on company's knowledge: future prospects, Euro. J. Operat. Res. 145 (2) (2003) 256–272.
[32] R.S.S. Guizzardi, L. Aroyo, G. Wagner, Agent-oriented knowledge management in learning environments: a peer-to-peer helpdesk case study, in: Agent-Mediated Knowledge Management, AAAI Press, Stanford, CA, USA, 2003, pp. 57–72, ISBN 1-57735-178-9.
[33] V. Hilaire, A. Koukam, P. Gruer, J.-P. Müller, Formal specification and prototyping of multi-agent systems, in: Proceedings of the First International Workshop on Engineering Societies in the Agents World: Revised Papers, vol. 1972, Springer Verlag, 2000, pp. 114–127.
[34] M. Ipe, Knowledge sharing in organizations: a conceptual framework, Hum. Resour. Develop. Rev. 2 (4) (2003) 337–359.
[35] K. Jee, J.-J. Yang, Knowledge description model for MAS utilizing distributed ontology repositories, in: Proceedings of the 9th Pacific Rim International Conference on Agent Computing and Multi-Agent Systems, vol. 4088, Berlin, 2006, pp. 773–780.
[36] P. Jackson, Transactive directories of organizational memory: towards a working data model, Inform. Manage. 49 (2) (2012) 118–125.
[37] J.J.J. Kasvi, M. Vartiainen, M. Hailikari, Managing knowledge and knowledge competences in projects and project organisations, Int. J. Project Manage. 21 (8) (2003) 571–582.
[38] M. Klein, Capturing design rationale in concurrent engineering teams, Computer 26 (1) (1993) 39–47.
[39] M. Klusch (Ed.), Intelligent Information Agents: Agent-based Information Discovery and Management in the Internet, Springer Verlag, New York, 1999.
[40] M.-C. Lai, H.-C. Huang, L.-H. Lin, M.-C. Kao, Potential of organizational memory for creating service performance. A cross-level analysis, Expert Syst. Appl. 38 (8) (2011) 10493–10498.
[41] N. Matta, J.L. Ermine, G. Aubertin, J.-Y. Trivin, Knowledge Capitalization with a knowledge engineering approach: the MASK method, in: Knowledge Management and Organizational Memory (IJCAI'2001 Workshop), 2002, pp. 17–28.

[42] N. Matta, M. Ribière, O. Corby, M. Lewkowicz, M. Zacklad, Project memory in design, in: Rajkumar Roy (Ed.), Industrial Knowledge Management – A Micro Level Approach, Springer-Verlag, 2000.

[43] D. Monticolo, V. Hilaire, A. Koukam, S. Meunier, An approach for building project memories to facilitate design process in a concurrent engineering context, in: International Conference of Concurrent Engineering (CE), Antibes-France, 2006, pp. 279–287.

[44] D. Monticolo, V. Hilaire, S. Gomes and A. Koukam, An ontological approach to manage project memories in organizations, chapter book in: P. Ceravolo, E. Damiani, A. Corallo, G. Elia, A. Zilli (Eds.), Semantic Knowledge Management: An Ontology-Based Framework, 2009, doi:http://dx.doi.org/10.4018/978-1-60566-034-9.ch011.

[45] M.V.N. Prasad, E.P. Ilia, Corporate Memories as Distributed Case Libraries, Technical Report, 1996.

[46] A.D. Preece, K. Hui, W.A. Gray, P. Marti, T.J.M. Bench-Capon, D.M. Jones, Z. Cui, The KRAFT architecture for knowledge fusion and transformation, Knowl. Based Syst. 13 (2–3) (2000) 113–120.

[47] Protégé 2000, 2000 http://protege.stanford.edu.

[48] A. Rabarijaona, R. Dieng, O. Corby, Building a XML-based corporate memory, in: Proceedings of the IJCAI '99 Workshop on Knowledge Management and Organizational memories Stockholm, Sweden, 1999.

[49] A. Rao, M.P. Georgeff, Decision procedures of BDI logics, J. Logic Comput. 8 (3) (1998) 293–343.

[50] M. Ribière, N. Matta, C. Cointe, A proposition for managing project memory in concurrent engineering, in: International Conference on Computational Intelligence and Multimedia Applications (ICCIMA'98). Churchill, Australia, 1998.

[51] A. Seaborne, E. Prud'hommeaux, SPARQL Query Language for RDF. Technical Report, W3C, 2006 http://www.w3.org/TR/2006/CR-rdf-sparql-query-20060406/.

[52] S.J.B. Shum, A. MacLean, V.M.E. Bellotti, N.V. Hammond, Graphical argumentation and design cognition, Hum.–Comput. Interact. 12 (3) (1997) 267–300.

[53] M.P. Singh, A social semantics for agent communication languages, in: Issues in Agent Communication, Springer, Berlin, 2000, pp. 31–45.

[54] B. Singh, Interconnected Roles (IR): A Coordination Model, Technical Report CT-084-92, MCC, 1992.

[55] K. Sycara, In-context information management through adaptive collaboration of intelligent agents, in: M. Klusch (Ed.), Intelligent Information Agent: Agent-Based Information Discovery and Management on the Internet, Springer, 1999, pp. 53–77.

[56] C. Tacla, De l'utilité des systèmes multiagents pour l'acquisition des connaissances au fil de l'eau, thèse soutenue en, 2003.

[57] M. Uschold, Building ontologies: towards an unified methodology, in: 16th Conference of the British Computer Society Specialist Group and Expert Systems, Cambridge, UK, 1996.

[58] L. Van Elst, V. Dignum, A. Abecker, Agent-mediated knowledge management, in: International Symposium AMKM, Standford CA, Selected Papers, LNAI 2926, 2004.

[59] M. Weggeman, Knowledge management: the modus operandi for a learning organization, in: J.F. Schreinemakers (Ed.), Knowledge Management: Organization, Competence and Methodology, Proceedings of ISMICK'96 Advances in Knowledge Management, vol. 1, Ergon Verlag, Wurzburg, Rotterdam, The Netherlands, 1996, pp. 175–188.

[60] M. Wooldridge, Reasoning About Rational Agents, MIT Press, Cambridge, MA, 2000.