International Journal of Advance Research in Science and Engineering
Vol. No.4, Special Issue (01), August 2015
www.ijarse.com
IJARSE
ISSN 2319 - 8354

MODELING A COLLABORATIVE MULTI-AGENT SYSTEM USING A PROCESS FOR COMPLEX ENGINEERING SYSTEMS

Amaigarou Noureddin[1], Mohamed Khaldi[2]

[1]Laboratory of Computer Science Operational Research and Applied Statistics,
Abdelmalek Essaadi University, Faculty of Science, Tetouan, Morocco B.P. Martil, Morocco
[2]Laboratory of Computer Science Operational Research and Applied Statistics,
Abdelmalek Essaadi University, Higher Normal School, Tetouan, Morocco B.P. Martil, Morocco

ABSTRACT

In this paper, we propose a methodological approach based on ASPECS (Agent-Oriented Software Engineering Process for Complex Systems) dedicated to the analysis, the design and deployment of complex systems. This analysis will allow us to highlight the objectives of our Knowledge Management System (KMS) and the main mechanisms of its functioning. Among the activities ASPECS, identifying needs is through an approach that will allow our but[1] modeling objectives of collaborative system and users of the system and parts of the Multi-Agent System (MAS) support different goals. The idea behind this methodology is based on a needs analysis, to define the main elements of the problem and the organizational structure of MAS that will meet the needs.

Keywords: ASPECS, Multi-Agent System, Knowledge Management System, Modelling, Collaborative

I. INTRODUCTION

CMS can be described as distributed systems where different actors (business actors), act autonomously to achieve a specific goal and interact to achieve a common goal.We propose a model of knowledge management system (KMS) which is based on the methodology ASPECS[2] dedicated to the analysis, design and deployment of complex systems. This analysis helps to identify the objectives of a MAS and the main mechanisms of its functioning. Among the ASPECS activities, identifying needs is through an approach that will allow[3] goals modeling KMS objectives and the actors involved and their dependencies to achieve each goal contributing to the creation of knowledge.

Knowledge management is a very broad field, raising several issues for which have been proposed various methods and techniques. Many studies (Ruggles, 1998)[4] and (Prax, 2003)[5] define knowledge management as a lifecycle consisting of process / phases that highlight the issues of management knowledge in a temporal perspective, such as the life cycle of Figure 1. This cycle of life, proposed in figure1 is intended generic and is organized into four processes.

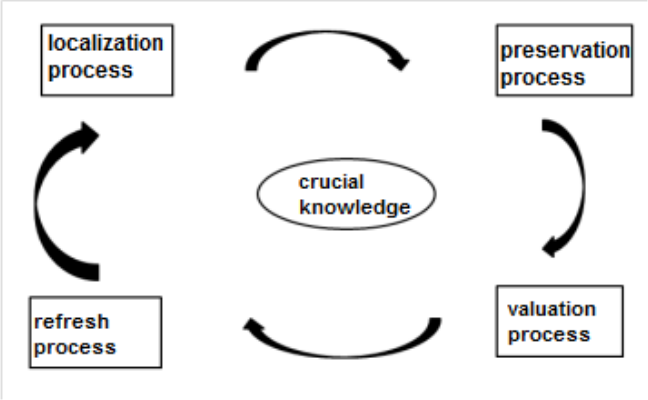

Figure 1: lifecycle knowledge

On this basis, ASPECS proposes to use MAS to facilitate the development of complex software systems. For this ASPECS introduced a particular type of agents, holons, which have the particularity to be composed of (sub) holons interact. ASPECS the development cycle is based on an iterative process. ASPECS provides a comprehensive guide, from needs analysis to implementation and deployment, allowing modeling of a system different levels of detail, proceeding by successive refinements. The approach offers the possibility for the designer to model a system with entities of different granularities. He can recursively decompose a system into subsystems, until reaching a level where the complexity of the tasks identified is MAS enough to be carried by entities considered atomic and easy to implement.

II. ASPECS, A MULTI-AGENT ENGINEERING PROCESS

2.1 Description

ASPECTS is a software engineering process that describes step by step the steps for software development, from requirements analysis through production code and deploy it on a specific platform . It is based on the metamodel CRIO [6], which defines the main concepts for the analysis, design and implementation of MAS.

The phases of the process are illustrated by the figure above :

- The needs analysis aims to provide a description of the organizational system (hierarchical decomposition of the system). It must also collect the available knowledge about the problem domain and organize within an ontology.
- The design of an agent company seeks to build the model of MAS, whose overall behavior should be able to provide a solution to the problem described in the previous phase. Knowledge of the system are refined and incorporate elements specific to the proposed solution.
- The implementation of the solution describes the architecture of the agents involved in the solution and must provide the source code of the application.
- The deployment of the solution is the final phase in charge of deploying the application on the platform chosen.

The modeling language UML is adopted. To fully satisfy the objectives and specific needs-oriented agent approach, semantics and UML notations were extended, and new UML profiles have been introduced including.

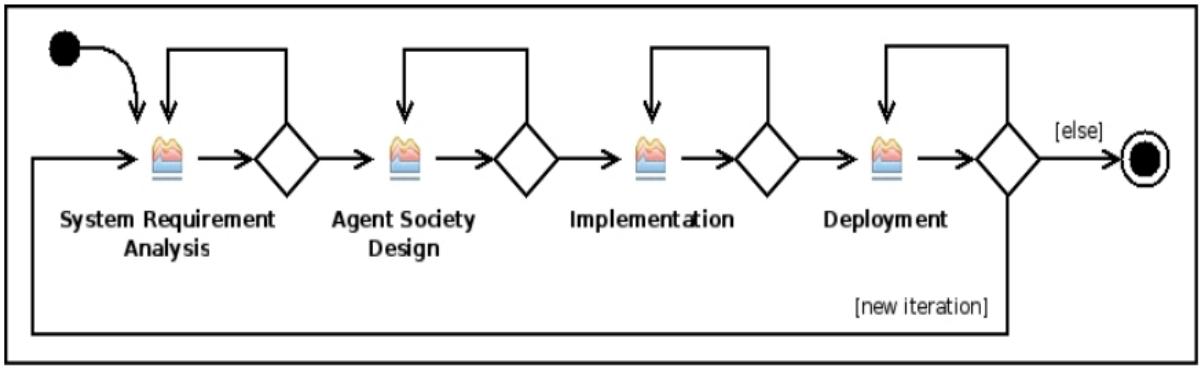

Fig 2: Phases du processus ASPECS[1]

2.2 ASPECS Activities

Each phase of ASPECS consists of activities that keep coming in the order shown schematically in Figure 1.3. An activity is "all the basic tasks performed by an individual or group that lead to the realization of goods or services» [7].

In this figure, the implementation and deployment phases were combined for clarity. Each activity is represented by a rectangle that contains the one hand the name of the activity at the top and the other, the objective of the activity in the lower part.

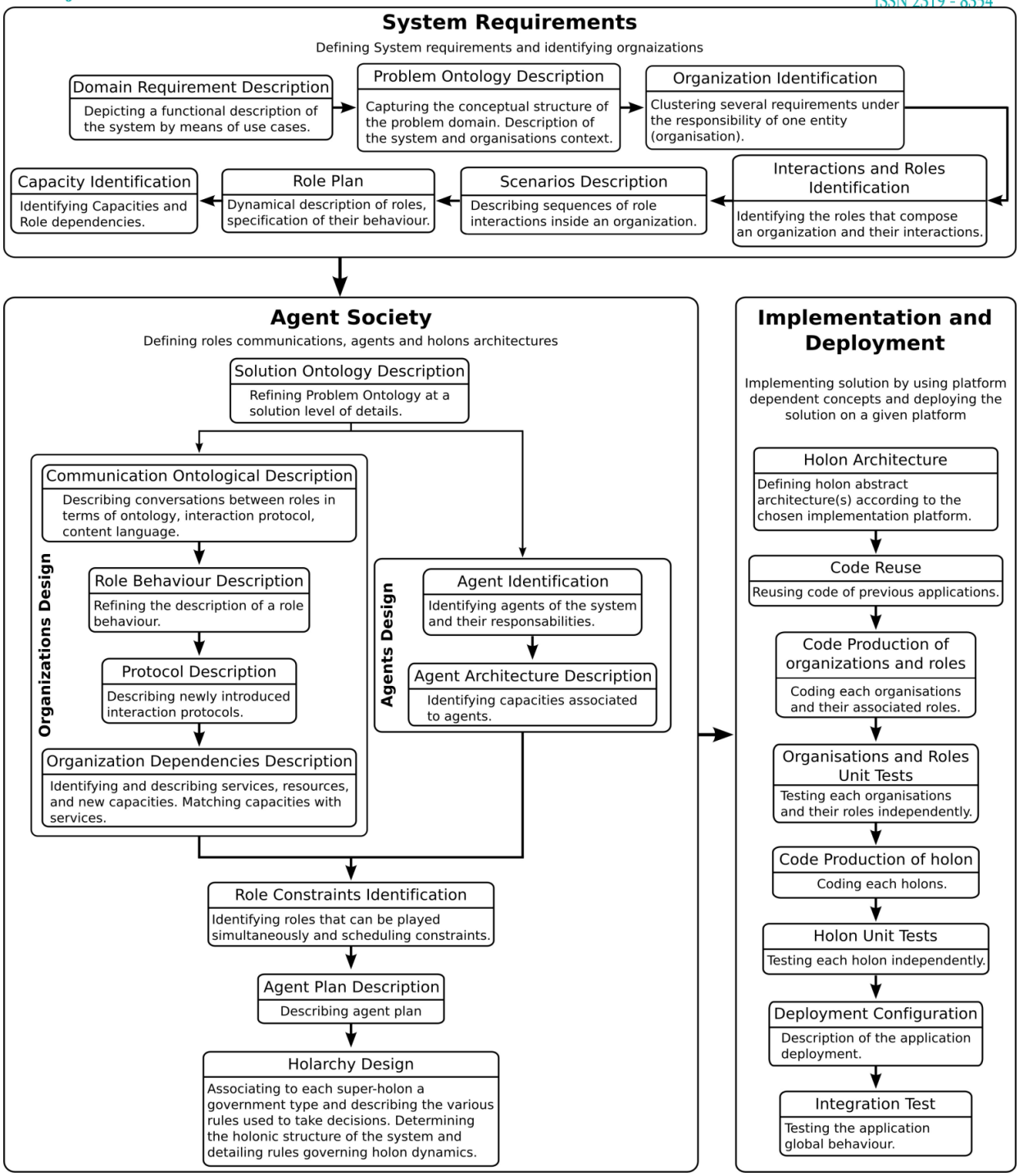

Fig 3: Details of activities ASPECS

III. SYSTEM ANALYSIS

3.1 Introduction

S@J-wiki[8] is a wiki system prototype under development as part of our research. S@ J-wiki serves several purposes:

1. can be used to annotate existing data with terms to improve search and navigation;
2. it can be used to create instance data based on an existing ontology;
3. can be used as a tool for creating and editing ontologies.

All three goals can be monitored simultaneously, perhaps by users with different roles and different levels of experience in knowledge engineering. Indeed, many engineering knowledge more complex tasks probably require this kind of collaboration.

In addition, S@J-Wiki refers to the following objectives:

- The syntax compatibility and appearance with existing systems (currently Wikipedia); allowing users to take existing knowledge (eg Wikipedia), import in S@J-Wiki, and begin the formalization of knowledge right away.
- Compatibility with existing Semantic Web technologies; Currently, S@ J-Wiki uses RDF and OWL to store and reason with formal knowledge.
- Immediate exploitation of formal knowledge for navigation and existing publishing; Users should get an instant reward for the extra effort they put into formalizing their knowledge.
- Easy access to common tasks; but still give users the functionality and complexity if they wish.
- feeling of an application, not a website; User interface should support the user beyond "wiki syntax" by providing a modern graphical interaction with the system (eg, WYSIWYG editing).

3.2 Identification of needs

S @ J-WIKI is a knowledge management system that has the main purpose of managing multi-source knowledge during the product design process.

The analysis of the main work on knowledge management highlights four principauxis process (Figure 1) :

- definition of knowledge.
- extraction of knowledge.
- validation of knowledge.
- reuse of knowledge.

We can equate these processes to global goals, such as soft-goal while KMS must reach (at least partly). Indeed, it seems difficult to define precise criteria for determining whether these aims are achieved or not.

These four soft goals therefore contribute positively to achieving the soft goal of knowledge management system "knowledge management". In addition, they are linked by the AND decomposition (Figure 8) because it is through the combination KMS may tend to the overall goal.

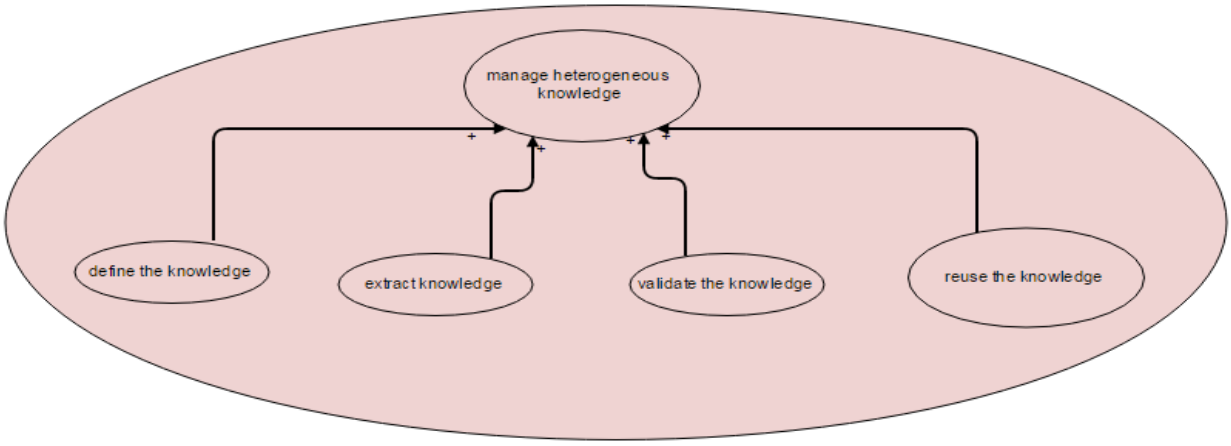

Figure 4 : Global Architecture of the KMS

3.3 Ontology Problem Description

The overall objective of the description of the ontology of the problem is to provide a conceptual overview of the problem under study. This activity deepens the understanding of the problem with a description of the concepts that make the problem domain.

The ontology of the problem is modeled using a class diagram where the concepts, attributes and actions are identified by specific stereotypes [9].

These stereotypes are:

- "concept" : to designate an entity of the domain,
- "action" : to describe a transformation of a concept,
- "predicate" : to designate a predicate on a set of concepts.

The UML diagram in Figure 9 shows our proposed ontology on the area of interest, namely our S @ J-Wiki system.

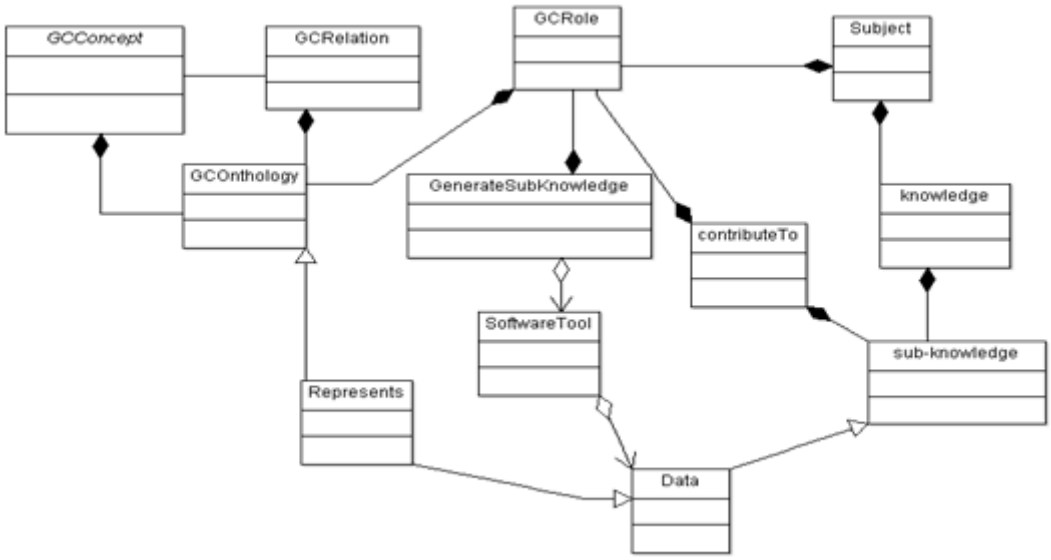

Figure 5: The UML Diagram

3.4 Identification of Organizations

In the activity of identification of organizations, each object is assigned to an organization that will aim, through interaction roles that will compose the organization to meet this goal. The goals identified, Figure 8, are allocated to organizations according to the relationships described in Figure 10. A global organization called J-S @ WIKI represents the interactions for the entire MAS. This organization is divided into four sub-organizations :

- **Definition**, aims to meet the objective of defining the knowledge that will be extracted for reuse.
- **Extraction**, aims, based on ontologies defined by the organization to provide knowledge Definition taken from existing projects to designers.
- **Validation**, aims to meet the goal Validate knowledge.
- **Reuse** allows reuse of this knowledge organizational memory

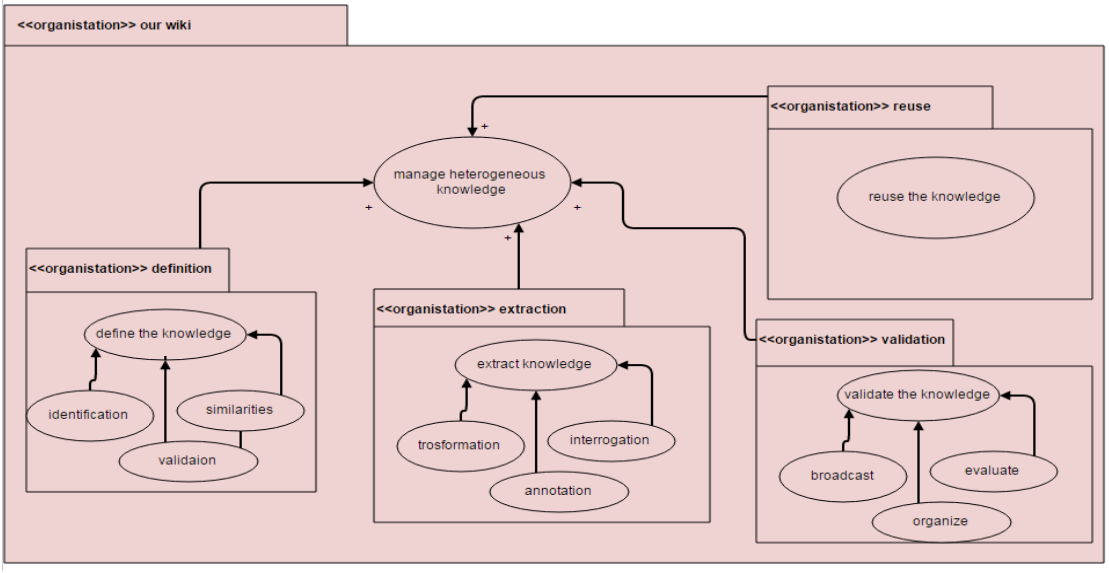

Figure 6: Global Organization of Our System

3.5 Identifying Roles and Interactions

The context and objectives of each organization are now known. Identifying the interactions and roles is to decompose the overall behavior embodied by an organization into a set of roles in interaction.

According Dieng [10]: « A role is an abstraction of a behavior in a specific context and confers status in the organization. The role the entity that gives the right to exercise these capabilities interpreter ». A role interacts with other roles in the organization to accomplish their tasks. This activity must describe the responsibilities of each role in meeting the needs associated with their organizations. Each role is associated with a set of concepts in the ontology.

In this article we'll just present the roles and interaction of the organization **definition**.

Expert adds the role of concepts and relationships GCOntologie ontology. Each new addition or modification, the role ChercheurSimilitude triggers a search mechanism concepts or similar relationships by synonyms, and offers the results to the expert for validation.

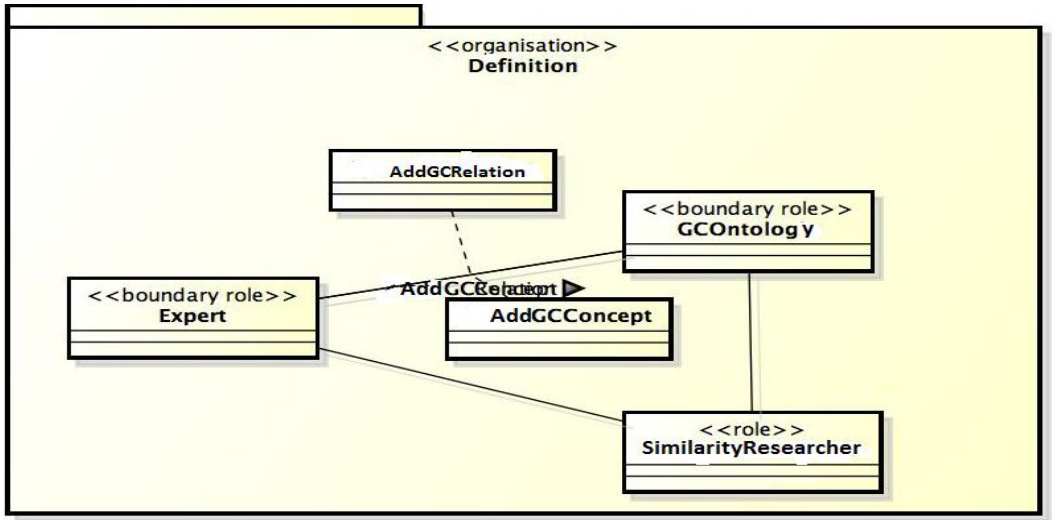

Figure 7: Sub-organization "definition"

IV. CONCLUSION

This chapter introduces the ASPECS software development process. ASPECS covers the entire development process, from requirements analysis to deployment.

This section briefly summarizes the contributions related to the methodology that will be used later in this manuscript.

Reuse the models and use of organizational design patterns are key points ASPECS. The definition of the behavior of roles on the basis of capacity, increases generics organizations and thus promote their re-use in future applications. In addition, the organizational approach adopted is particularly advantageous for the development of complex applications and promotes modularity, scalability and reuse patterns.

Reuse is also encouraged by the ontology, considered the basic common and transversal knowledge modeling process. Indeed, knowledge of the problem domain are grouped and classified in this database.

REFERENCES

[1] Cossentino M, Gaud N, Hilaire V, Galland S, Koukam A (2007), ASPECS: an Agent-oriented Software Process for Engineering Complex Systems, In Proc. Of the Fifth Agent Oriented Software Engineering Technical Forum (AOSE-TF5), Hammamet, Tunisia.

[2] Cossentino M, Gaud N, Hilaire V, Galland S, Koukam A (2007), ASPECS: an Agent-oriented Software Process for Engineering Complex Systems, In Proc. Of the Fifth Agent Oriented Software Engineering Technical Forum (AOSE-TF5), Hammamet, Tunisia.

[3] FIPA ACL, 2002. FIPA ACL Message Structure Specification (Standard No. SC00037J),Foundation For intelligent Physical Agents.

[4] Ruggles, R (1998), « The state of the notion : knowledge management in practice », California Management Review 40 (3): 80-89.

[5] Prax J.-Y Manuel de knowledge management. - Paris, Dunod, 2003.

[6] Systèmes Multi Agents Holoniques: de l'analyse à l'implanatation. Métamodèle, méthodologie et simulation multi-niveaux. Thèse soutenue en 2007.

[7] Pahl G., Beitz W., "Engineering Design: a Systematic Approach", 2nd Springer-Verlag, London, 1996.

[8] AMAIGAROU Noureddin, KHALDI Mohamed, (2014); "S@J-Wiki A collaborative semantic wiki to manage juridical knowledge "; International Journal of Computer Applications (0975 – 8887) Volume 104__2014

[9] FIPA ACL Message Structure Specification. Foundation For intelligent Physical Agents, 2002. Standard, SC00037J

[10] DIENG, R., CORBY, O., GIBOIN, A., RIBIÈRE, M., 1999. Methods and tools for corporate knowledge management. International Journal of Human-Computer Studies 51, 567‑598.