

Renata Guizzardi, Anna Perini, Virginia Dignum: Socially Grounded Analysis of Knowledge Management Systems and Processes. In Giorgini, P., Maiden, N., Mylopoulos, J. and Yu, E. (Eds.) "Tropos/i*: Applications, variations and Extensions", Cooperative Information Systems Series, MIT Press (forthcoming), 2008.

Socially Grounded Analysis of Knowledge Management Systems and Processes

Renata S. S. Guizzardi¹, Anna Perini², Virginia Dignum³

¹Computer Science Department, Federal University of Esp rito Santo, Brazil
Email: rguizzardi@inf.ufes.br

²Fondazione Bruno Kessler - IRST, Italy
Email: perini@itc.it

³Institute of Information and Computing Sciences, Utrecht University, The
Netherlands
Email: virginia@cs.uu.nl

1. Introduction

In the struggle to survive and compete in face of constant technological changes and unstable business environments, organizations recognize *knowledge* as its most valuable asset. Consequently, these organizations often invest on Knowledge Management (KM), seeking to enhance their internal processes and available technologies to sustain and disseminate knowledge throughout the organizational setting (Alavi & Leidner, 1999; Fischer and Ostwald, 2001; Nonaka and Takeuchi, 1995; Orlikowski and Gash, 1994; Wenger, 1998).

KM environments are highly influenced by their social dimension, involving intricate relationships among different actors and guided by organizational norms and culture (Dignum and van Eeden, 2005; Nonaka and Takeuchi, 1995; Orlikowski and Gash, 1994; Wenger, 1998). Nevertheless, many current KM systems are developed following a purely techno-centric view, focusing on the

system's functionality rather than on the environment's social dimension. Such approach is bound to fail, as it does not consider the particularities of the social environment in which the system is to be used (Bonifacio & Bouquet, 2002; Guizzardi, 2006; Orlikowski and Gash, 1994).

As a solution to this problem, we must look beyond the usual emphasis on information and on individuals' activities and processes, paying more attention to the rich social networks that add meaning to all forms of work. For that, we claim that more focus should be given to the initial phases of system development, aiming at grasping the *requirements* of the system to be, by understanding in details the social structure of the targeted organization. This analysis should consider both the individual perspective of organizational members and the overall objectives of the organization. Such strategy should be able to generate solution that faces less resistance by organizational members, thus resulting in more appropriate and sustainable KM systems and processes.

In this chapter, we argue that agents are suitable for modeling human and artificial organizations due to their autonomous, reactive and proactive nature. Moreover, agent's cognitive characteristics such as *goals*, *plans*, and *dependencies* are adequate constructs to capture and describe relevant aspects of behavior and organizational norms and processes. Taking advantage of such elements, we propose ARKnowD (Agent-oriented Recipe for Knowledge Management Systems Development). This methodology allows different analysis

methods, such as understanding commitments and vulnerability in the relationships between actors, and grasping the ‘hows’ and ‘whys’ of a particular choice. Moreover, it provides guidelines founded on cognitive sciences to support the analyst in understanding the gaps existing in the organizational setting, paving the way for the proposal of effective KM solutions. The results of the analysis may take different forms. Perhaps the problem may be more effectively solved by proposing changes in the business processes, rather than by applying technology. Here, a case study is used to illustrate the methodology, exemplifying the proposal of changes in the organizational processes to accommodate new KM practices. The relationship of this work with current studies in agent-oriented approaches for organizational KM is also discussed.

The remaining of this chapter is organized as follows: section 2 discusses the appropriateness of the agent-oriented paradigm to support KM; section 3 describes the ARKnowD methodology, along with its activities and lifecycle; section 4 illustrates ARKnowD, analyzing a fictitious scenario with the main purpose of demonstrating the expressivity of the adopted approach to capture organizational aspects specifically related to KM; section 5 discusses how our approach relates to existing work; and section 6 finally concludes this chapter.

2. Agent-oriented Support to Knowledge Management

KM environments can be described as distributed systems where different agents need to interact in order to achieve both their individual and common goals. In

such environments, the ability to communicate and negotiate is paramount.

Furthermore, the number and behavior of participants cannot be fixed a priori and the system can be expected to expand and change during operation, both in number of participants as in amount and kind of knowledge shared.

KM literature often mentions that the efficacy of KM processes and systems are very much impacted by organizational culture (Nonaka and Takeuchi, 1995; Orlikowski and Gash, 1994; Wenger, 1998). Organizational culture is defined by the common sense knowledge, accepted behavior, and cultivated values within the organization. There are often reports on the fact that the adopted KM enabling systems are based on architectures and methods that reinforce old pernicious habits and power structures, instead of creating new and beneficial dynamics (Bonifacio& Bouquet, 2002; Orlikowski and Gash, 1994).

Our work is based on the assumption that technological and social aspects of organizations are tightly interrelated (Bonifacio& Bouquet, 2002; Guizzardi, 2006; Orlikowski and Gash, 1994). The more an organizational process involves high level human activities, the stronger the interdependence between technology supporting that task and the organizational dimensions. In order to guarantee that an adopted system or process is going to fulfill its promises, it is thus necessary to perform a comprehensive analysis of the organizational processes, indicating which changes should occur to better accommodate the use of the adopted solution.

In order to cope with the inherent complexity of more comprehensive solutions for KM, the concept of Agent-mediated Knowledge Management (AMKM) proposes agent-based approaches to deal with collective aspects of the domain in an attempt to cope with the conflict between desired order and actual behavior in dynamic environments (van Elst, Dignum and Abecker, 2004). Inherent to AMKM is a social layer, which structures the society of agents by defining specific roles and possible interactions between them.

The agent paradigm enables to analyze and model the complex environment involving heterogeneous and complex social interactions, autonomous cognitive entities, possibly conflicting requirements, distribution of data and management, and unpredictable business processes. These are all characteristics of KM environments, which lead to an AMKM approach (van Elst, Dignum and Abecker, 2004). The use of agents in KM can be seen in two perspectives. In one hand, agents can be used **to model** the organizational setting where the KM system will operate and, on the other hand, software agents can be used **to implement** the functionality of KM systems. Most existing KM projects involving agent technology concentrate on using agents as implementation tool modeling primitives. Agents are used there to support and extend the activity of (human) users. However, more and more interest is arising about the advantages of agent-based modeling of KM environments, thus viewing both organizations

(and organizational units) and humans as agents in the model of such environments.

Currently, only a few of the existing AMKM approaches provide a comprehensive methodology that enables the development of KM support systems from a software engineering perspective. Agent-Oriented Software Engineering (AOSE) methodologies provide models and methods adequate to represent and support all types of activities of the software lifecycle. These methodologies must be both specific enough to allow engineers to design the system, and generic enough to allow the acceptance and implementation of multi-agent systems within an organization, allowing for the involvement of users, managers and project teams. From an organizational point of view, the behavior of individual agents in a society can only be understood and described in relation to the social structure. Therefore, the engineering of agent societies for KM needs to consider both the *interacting and communicating abilities* of agents as well as the *environment* in which the agent society is situated (i.e. the organization). Moreover, when applied to KM, AOSE methodologies must take into account the characteristics and idiosyncrasies of this particular domain.

The characteristics highlighted in the previous paragraph indicate that AMKM approaches should somehow relate and support the usual KM processes, such those of *creation, integration and dissemination* of knowledge (Fischer and Ostwald, 2001). Knowledge creation, as the name suggests, is the activity that

leads organizational members to generate new knowledge; knowledge integration refers to converting it into a sharable technological format, while also connecting it to existing knowledge within the organization; and finally, knowledge dissemination enables access of specific knowledge to all employees and units that need to apply it in practice.

Given the number of available methods and languages for agent-oriented analysis and design (Henderson-Sellers and Giorgini, 2005), the approach presented in this paper builds on existing work. One of the principles of our methodology is to grant analysts and designers with the freedom to select the appropriate tools from a vast ‘library’ of methods and languages, depending on the specific case at hand. This is inspired by ideas of method engineering¹. Figure 1 illustrates possible development options based on a combination of existing AOSE methodologies and models.

For instance, analysts that are familiar with the Gaia methodology (Wooldridge, Jennings, & Kinny, 2000) could start with the definition of roles and interactions and, then, refine these models respectively into OperA’s roles and scenes (Digum, 2004) (path 1). This would result in a more detailed and formalized analysis model. Another possibility is given by the combination of Tropos (Bresciani, Giorgini, Giunchiglia, Mylopoulos & Perini, 2004) and OperA (path 2). Tropos is the approach that gives more attention to the requirements analysis activity, which is of great relevance for KM settings (Guizzardi, 2006)

(chapters 2 and 3). Combining Tropos and OperA allows the analyst to combine the Tropos's requirements analysis capabilities with the formal semantics of OperA models. However, if a formalized model is not necessary, Tropos can be combined with AORML (Wagner, 2003) (paths 3 and 4), generating a methodology that covers all activities of system development, coming from early requirements analysis to detailed system design. Finally, in path 5, ROADMAP (Juan, Pearce and Sterling, 2002) and AORML are combined. ROADMAP is specifically tailored to enable the development of open systems using AUML (Odell, Parunak and Bauer, 2000) for detailed design. The approach we describe in the next section proposes a methodology for the analysis and design of KM systems, following the combination exemplified in path 4. This methodology takes in account the social aspects of the organization and the possible heterogeneous and conflicting requirements of its individual participants.

3. Proposed Approach

We propose an agent-oriented methodology to develop KM solutions named ARKnowD (reading "Arnold") (Guizzardi, 2006). The ARKnowD methodology can be applied to different application scenarios, ranging from situations in which the problem leads to the proposal of changes in the structure and/or process underlying the organization to accommodate/enhance KM practices, to the case in which the aim is that of adopting/building an enabling tool, technological or not, to support KM. In this chapter, we illustrate the application of ARKnowD to

enable the former, while the latter has been previously exemplified in (Guizzardi and Perini, 2005).

Organizational reengineering is often mentioned as a solution to create more conducive environments for knowledge sharing (Orlikowski and Gash, 1994; Nonaka and Takeuchi, 1995). This can be achieved by *changing the structure of the organization*, creating or extinguishing organizational units or departments. As an interesting example, we cite the case of the Dutch insurance company Achmea reported in (Dignum and van Eeden, 2005), which created a new division (the KM division) to propose projects and take care of all matters related to KM. Besides changes in structure, several are the cases reporting KM enhancements that emerge as a result of *changes in process*. Nonaka and Takeuchi (1995) report a case of this sort, involving a team of technicians from Matsushita that constructed the first fully automatic electronic bread-making machine for home use. After a few frustrating experiences, especially to automate the process of kneading the dough, Matsushita achieved encouraging results after an internship that a senior developer made with a baker. In participating in the activities of making bread, he realized the right way to knead the dough, later embedding a mechanism in the machine that imitated the baker's movements. In this case, no new department was created and no new tool was adopted. Instead, the routine of an employee (the software developer) was radically changed, enabling him to capture tacit knowledge embedded in the practices of a specialist.

The ARKnowD's engineering process is conceived as a chain of activities that may be performed several times, in an iterative process, as illustrated in Figure 2.

We will describe them in the following:

1. *Requirements elicitation.* Requirements elicitation is a basic activity of all software engineering processes, responsible for eliciting the needs and wants of the stakeholders of the system-to-be (Goguen and Linde 1993). In ARknowD, requirements refer to any need for change in terms of *organizational structure, process and tools* (which include information systems and general tools). ARKnowD does not prescribe any specific elicitation technique; however, we recognize benefits may be attained by the observation of people in action (as proposed by ethnographic techniques) in combination with interviews and questionnaires.
2. *Requirements analysis.* Requirements analysis refers to the activity of modeling and reasoning about organizational requirements. Our methodology models requirements as goals, as proposed in Tropos. This view has been largely acknowledged by the Requirements Engineering community (Kavakli and Loucopoulos, 2005) and is also compliant with KM theories. According to Nonaka and Takeuchi (1995), for example, one of the main drivers of knowledge creation is the organization's *intention*, defined as “an organization's aspiration to its goals”.

Nevertheless, these authors mainly focus on the organization top management's intention for facilitating KM initiatives. In contrast, we consider the goals of all stakeholders, trying to understand the relations and possible discrepancies between their goals. This view is aimed at providing autonomy in knowledge sharing, as prescribed by Constructivist KM (Guizzardi, 2006) (chapter 2), and also emphasized in (Dignum, 2004) . In fact, ARKnowD provides guidelines to requirements analysis, based on six principles, also known as *Constructivist KM building blocks*: autonomy, non-hierarchical knowledge sharing, social interaction, physical meaningful artifacts, perturbations, and context. In more detail, ARKnowD particularly supports the analyst to understand (Guizzardi, 2006) (chapter 3):

- how much *autonomy* is given to each organizational member to share (i.e. create, integrate and disseminate) knowledge the way he/she finds appropriate;
- if the creation, integration and dissemination of knowledge happens in a bureaucratic way, obeying hierarchical structures within the organization or if it is rather *non-hierarchical and natural*, motivating each one to contribute with his/her share of knowledge despite organizational position or experience;

- how well organizational processes favor *social interaction*, considered here as an essential ingredient for the disambiguation of tacit knowledge, and thus for the generation of innovation (i.e. creation of new knowledge);
- what kind of *meaningful artifacts* are exchanged among organizational members, cross-cutting divisions and communities and in this way, integrating knowledge which is disseminated throughout the organization;
- how constructive *perturbations* are generated and coped with within the organization, triggering the dynamics that motivate employees to constantly self-improve by sharing knowledge;
- what kind of *contexts* emerge or are actively planned by the organization for knowledge creation, integration and dissemination.

We claim that the presence of the highlighted characteristics within the organization's setting leads to more effective support to KM, favoring the processes of creation, integration and dissemination of knowledge, described in Section 2. Thus, a deeper understanding of how much the analyzed setting complies with these principles gives the analyst the means to assess how well the organization currently supports KM.

Borrowing Tropos's and i* approach (described respectively in chapters [Giorgini, Mylopoulos, Perini, Susi](#) and [Yu – The i* Framework](#)), in ARKnowD, requirements analysis is divided in two subactivities: a) early requirements, focused on analyzing the current state of affairs in the organizational setting; and b) late requirements analysis, targeted on proposing changes to meet the organization's needs (Bresciani et al., 2004) [[crossreference to chapter on Yu-The i* Framework](#)]. During early requirements, while applying the analysis techniques proposed by Tropos, the Constructivist KM principles are used as additional guidance, serving as a kind of *checklist* for the domain analyst. Understanding the presence or absence of such principles allows the analyst to correct limitations in the organization's setting and to propose appropriate KM solutions during late requirements.

3. *Design*. The design activity is responsible for providing the solution in as much details as to enable it to be developed in practice. It can be viewed as two distinct sub-activities: architectural design and detailed design. In architectural design, all agents of the system should be identified, along with their individual goals. In addition, the resources and plans used by the agents to achieve their goals are modeled. In the detailed design, the information structure of the system is detailed, along with processes and agent's behavior. In ARKnowD's life cycle, the use of Tropos ends with

architectural design. For the detailed design, a different notation is applied, namely AORML (Wagner, 2003). Thus, in the transition from architectural to detailed design, we propose a transformation from the notation adopted in Tropos to AORML. This transformation, which can be partially automated, is out of the scope of this paper but it is described in detail in [Guizzardi and Guizzardi].

Note that our conceptualization of ‘system’ is general, including but not being restricted to that of information system. System can be defined as a general set of interacting entities, thus comprehending artificial and non-artificial entities (such as humans, organizations and organizational units). This opens the possibility to consider several outcomes resulting from the application of our methodology, such as: changing organizational structures, modifying processes, and adopting technological or non-technological tools.

The development of a solution commonly requires several cycles, each one performing to a certain extent, some or all the activities described above. The first cycles are characterized by the focus on the clarification, negotiation and agreement of requirements, thus requirements elicitation and analysis are iterated several times. Then, the focus slowly shifts to the development of the solution, although each design cycle may still require the elicitation and further analysis of

new requirements. To sum up, during requirements analysis, ARKnowD provides an agent-oriented methodology enriched with guidelines founded on cognitive sciences to support the analyst on conceiving an appropriate KM solution for the organization. Such solution is structured during architectural design. And finally, a transformation method guides the designer on consistently moving towards a design model which provides all details needed to enable the implementation of the conceived solution.

4. Demonstrating ARKnowD in the Analysis of a Fictitious Scenario: Supporting KM in Communities of Practice

In order to illustrate our proposed methodology, we use here a fictitious scenario, not intended as a universal scenario, but rather focusing on a few illustrative issues of the domain at hand. Although not a real case study, this scenario was carefully tailored to be realistic, taking into consideration the available literature (Alavi & Leidner, 1999; Dignum and van Eeden, 2005; Fischer and Ostwald, 2001; Nonaka and Takeuchi, 1995; Orlikowski and Gash, 1994; Wenger, 1998). Here follows the scenario description.

“BHI is a medium-sized software company, composed of ten branches distributed in three countries. The company develops projects in several Internet-related areas, including e-commerce, e-learning and mobile technology supporting businesses in general. Aiming at providing its workers with a rich environment for knowledge sharing, BHI Management has created a special

division named the Knowledge Management Division. This division is launching an initiative to foster the development of Communities of Practice (CoPs) across the organization. CoPs are self-organizing groups whose members share interests and goals, or perform similar tasks within the organization. They are not necessarily from the same working team or division, and their members are dispersed across BHI branches. One of the main objectives of this division is supporting the CoP on pursuing explicit targets related to the organization's goals. This allows the community members to feel important as a group for the organization at the same time that the CoP's value is more concretely measurable from the Management's point of view.”

In the coming subsections, we apply ARKnowD to analyze the scenario and to evaluate the impact of adopting new organizational practices, particularly supporting the aforementioned KM Division. More specifically, we focus on the activities of Early and Late Requirements analysis, by using i*/Tropos dependency and goal analysis methods (Bresciani et al., 2004)[Yu- i* Framework]. Moreover, we discuss the Constructivist KM principles supported by the analysis results (refer to section 3), highlighting these principles in boldface. This way, we aim at providing guidelines on how such principles can be used in combination with the i*/Tropos methods to enable a coherent and deep analysis of the organization’s KM needs, as well as supporting the analyst on the proposal of an efficient solution for these issues.

4.1. Early Requirements: Understanding the Organization's Current State of Affairs

The Early Requirements analysis starts with an overview of the goal dependencies among the main actors involved in the scenario, that can be identified through *who*, *what*, *how*, *how-else*, and *why* questions. In other words, this analysis activity focuses on the organization's current state of affairs, looking at the present actors and goal dependencies within the organization.

An Early Requirements model of our scenario is depicted in Figure 3², which shows the BHI company's directory board modeled as the Top Management actor, depicted as a circle. The organization has an initial softgoal relative to having the organization's team working well³, which expresses how BHI intends to achieve more general objectives such as pursuing high quality of the products and of the production processes pursuing high quality products/processes goal), as well as innovation (innovating goal) by considering human resources as a main asset.

The analysis of the Top Management's softgoal team working well points out a strategic organizational goal, i.e. CoPs fostering, which is then delegated to the Knowledge Management Division (KM Division actor). In return, the KM Division depends on the organization's Top Management to be legitimized for playing the specific role of motivating and supporting Knowledge Management practices (legitimization getting goal). The initial Top Management's softgoal,

leading to its main goal of supporting CoPs, generates all other goal dependencies between the remaining actors in the scenario.

Taking, for instance, the pair of actors **Employee** and **CoP**⁴, we note that there are goal dependencies in both directions. The **Employee** depends on the **CoP** to get new knowledge and to gain external incentives or motivation in order to share his/her own knowledge (**getting knowledge** and **gaining incentives** goals). On the other hand, the **CoP** aims at taking contributions from the **Employee's** own knowledge and experience (**getting knowledge** goal from the **CoP**). This mutual dependency characterizes what *i** names “*sustainable relationship*”, i.e. a relationship in which two actors depend on each other to achieve one or more of their own goals. Sustainable relationships indicate that there is some kind of balance between the two actors, thus helping them achieve individual goals. On the other hand, if there are dependencies only from one side, this indicates vulnerability by this *dependor* actor towards the *dependee* [Yu-i* Framework]. Such unbalance should be corrected in order to guarantee that both actors are committed to each other.

Focusing on the dependencies between the **KM Division** and the **CoP** actors, note that while the **KM Division** delegates two goals to the **CoP** (having real target and accomplishing **CoP's** goals), the **CoP** does not seem to depend on the **KM Division** for achieving any goal. This can result in a lack of motivation on the part of the **CoP** to target the goals delegated by the **KM Division**.

This kind of dependency analysis is crucial in KM scenarios. In our case, for instance, we realize that while demanding the CoPs involvement with organization's objectives, the KM Division does not provide, as a counterpart, any incentive to the CoP. In fact, the KM literature (Orlikowski and Gash, 1994) indicates that incentive policies are essential to motivate knowledge sharing.

Looking at the Constructivist KM building blocks discussed in section 3, a positive aspect of the present organization is the fact that it encourages the creation of CoPs. According to theoretical and practical findings (Wenger, 1998), these social phenomena may provide an appropriate **context** for effective knowledge creation, integration and sharing. In addition to that, the kind of relationship fostered in these communities are usually **non-hierarchical**, in the sense that each individual is viewed at the same time as a producer and consumer of knowledge, no matter which position he/she occupies within the organization. Since the organization seems to be in the right track, our work on the next analysis step (i.e. Late requirements) is to make a proposal regarding the correction of the unbalanced relationship previously identified between the CoP and the KM Division actors.

In addition to the dependency analysis presented above, careful investigation and constant monitoring is needed within the organization so as to pay attention to external regulations that may prevent the maintenance of sustainable relationships. Besides this, another problem can also arise: the *lock of*

dependencies, where an actor A depends on other actor B to fulfill a goal G1, but the actor B also depends of A for fulfilling the goal G2 in order to satisfy his own goal (i.e. G1). In the case described above, balancing dependencies is the attempted solution to an existing problem. Conversely, in these cases, the balanced relationships represent a problem to be solved unbalancing the relationship. This exemplifies the diversity of issues that can be noted and corrected in an organizational setting by analyzing the dependency between the actors of this organization.

4.2. Late Requirements: Proposing a Solution to the Problem Identified during Early Requirements

During Late Requirements, we shift the focus from analyzing the present situation to the analysis of possible solutions to the problems at hand. This way, the models from now on do not depict the current dependencies between the organization's actors. Conversely, the tendency here is to model the analyst's proposals of how one actor should depend on the other to achieve his as well as the overall organization's goals.

As a result of the analysis described in the previous subsection, we propose to the KM Division the adoption of a set of practices for fostering CoPs, such as those suggested by the *Seduce, Engage, Support (SES)* framework (Dignum and van Eeden, 2005). In this framework, an important role is given to a department such as the KM Division, which provides a set of practical guidelines to create

and sustain CoPs. This makes SES particularly attractive to our scenario, conducting us to the correction of the unbalanced relationship problem formerly analyzed. Considering the model depicted in Figure 4, we analyze the impact of introducing the SES framework in the current organization in a Late Requirements activity.

In order to accomplish the two KM Division's goals (identified during Early Requirements), the CoP needs to get incentives to develop its activities (getting incentives goal), and to have guidance throughout its lifetime (having guidance goal). These two new goals correct the previous unbalance between these two actors, creating a sustainable relationship between CoP and KM Division. Both new goals can be attained by the adoption of the SES framework. Analyzing the perspective of the KM Division, we can understand this in more detail.

Figure 4 allows the analysis of the internal perspective of the KM Division, which adopts the SES framework (adopting SES method goal). This initial goal has been analyzed and decomposed in sub-goals (AND-decomposition), providing us with an overview of the SES framework, which comprises three phases: seduce, engage and support phases (seducing, engaging and supporting goals). During the *seduce* phase, the need of creating CoP motivates the identification of the context and aims of a CoP (clarifying purposes goal), potential members are made aware of their connections and common interests (connecting members goal), and a “marketing campaign” is launched, showing

the added values and benefits of the CoP for the whole organization (marketing goal). In the *engage* phase, both community members and organization are involved in the process of setting up the CoP. The aim is to design a community that is as closely related as possible to the requirements and wishes of the members (linking CoP's and members' requirements goal) and whose tasks and targets are well embedded in the strategic priorities of the organization (supporting establishment of real targets goal). The aim of the *support* phase is to consolidate the CoP, by developing CoP-specific methods and tools for assisting its activities (providing infrastructure goal), besides verifying its progress (monitoring goal) and granting incentives (providing incentives goal).

4.3 Detailing the KM Division Strategy

The following four diagrams present in details how the KM Division may pursue the SES Method. Once more, it is important to emphasize that these diagrams comprehend the analyst's proposal for the organization. Thus, they can be viewed as recommendations to BHI as to how the KM Division should proceed to accomplish each of the phases of the SES method. More precisely, Figure 5 describes how (in the analyst proposal) the KM Division targets the seduce phase, when advertising and clarifying the ideas surrounding the creation of a CoP are essential. Notice that at this stage, the CoP does not yet exist. In fact, the creation of a new CoP is the desirable outcome of the KM Division's guidance at this moment. Next, Figure 6 and 7 deal with the engage phase, assisting the CoP

members to define their targets within the interests of the organization, besides triggering their initial activities. Finally, Figure 8 explicitates how the analyst proposes the KM Division pursue the general goal of providing long-lasting support to CoP activities.

4.3.1. The Seduce Phase

It was detected during early analysis that in the process of developing software projects, BHI's employees apply an increasing number of open source tools in several activities, including project management, system version control and software design and implementation⁵. Having the Constructivist KM principles in mind, we recognize such tools as **meaningful artifacts** shared by employees of different divisions, realizing that BHI may profit from this natural and emergent potential to motivate the formation of CoPs. Thus, according to our proposal, the KM Division proposes a joint project to be undertaken among the organization's divisions (proposing joint open source project goal). In other words, the joint project is the main strategy our analysis proposes to the KM Division as a tool to seduce CoPs.

The joint project consists that an **Employee** of a division turns public all the information regarding open source software he or she uses to the employees of the remaining organization's divisions (sharing information on used open source software goal). Knowledge sharing in this respect may favor the adoption of tools previously unknown within some divisions to speed up and enhance the

quality of their work. In addition to that, the synergy created by the interactions among divisions is hoped to create opportunities presently unforeseen, as reported in KM literature (Nonaka and Takeuchi, 1995). Such synergy is also supported by Constructivist KM, which emphasizes the importance of **social interactions** to enable knowledge creation and sharing.

It is the purpose of the KM Division that in the near future, the joint project gives rise to a CoP specifically focused on this theme. As can be noted in Figure 5, this strategy contributes positively to all sub-goals of the SES seduce phase. On the one hand, the joint project is a way to promote the idea of communities that cross-cut organization divisions and even branches (positive contribution going from proposing joint open source project goal to marketing goal). On the other hand, it assists at the same time to clarify what a CoP is along with its main purposes (positive contribution going from proposing joint open source project goal to clarifying purposes goal). And finally, it initiates the first connections between future CoP members (positive contribution going from proposing joint open source project goal to connecting members goal).

However, the KM Division depends on other agents in the organization to effectively make this plan work. Besides the obvious dependency on the **Employees** to share relevant information, there are two other important dependencies to consider. First, the KM Division depends on the **Top Management** to include the joint project as part of the main organization's

strategy (**adopting strategy goal**). Section 2 pointed out that culture plays a tremendous role in encouraging the adoption of KM practices within organizations. And besides, creating culture that motivates knowledge sharing depends first of all in the firm intention of the organization top management (Nonaka and Takeuchi, 1995). In addition to this, the **Middle Managers** of the organization are potential allies that **KM Division** does not dare to disregard. Everyone in the division works under the coordination of a **Middle Manager**, which makes them the appropriate persons to encourage active participation of all **Employees** in the project (**encouraging workers goal**). Note however that the choice for a *goal dependency* instead of a *plan dependency* in this diagram expresses the fact that the **KM Division** does not specifically prescribes how **Middle Managers** should pursue this goal. In this way, the personal style of each **Middle Manager** and their own relationship to their subordinates are respected, and they may independently choose the system of incentives they shall use to accomplish this goal. Such measure is compliant with the Constructivist KM principle of **autonomy**.

4.3.2. The Engage Phase

In our fictitious scenario, after six months, the objectives of the **KM Division** regarding the joint project have been successfully achieved. Great part of the participants recognized the benefits of knowledge sharing and was willing to create a **CoP** to carry on the activities dealing with the adoption of new open

source tools within the organization. This has given rise to a community named '*OpenS*'.

It is now time for the KM Division to act as motivator, providing the necessary means for the community to grow and mature. In other words, the KM Division and the CoP enter the engage phase. Figure 6 captures a late requirements model for this phase. The analyst proposes that the KM Division takes two main measures: gathering all interested employees in a brainstorming session (organizing brainstorming session plan) to elicit ideas regarding CoP activities and define their specific targets; and starting a discussion about some important initial roles to support the main CoP activities, also assisting in the definition of who among the Members should play these roles (promoting definition of CoP roles goal). Naturally, the KM Division depends on the community's Members to participate in the debate involved in the brainstorming session (participating goal), to volunteer to play specific roles (volunteering for role goal), and to support the decision regarding the chosen ones (supporting decisions goal).

Figure 6 also depicts a contribution analysis regarding the CoP's main goals. The CoP has a general softgoal of developing well its activities. Moreover, it has other two goals: growing in size and importance and gaining visibility within the organization. Positive contribution links going from one goal/softgoal to the other show that although independent, these goals/softgoal have relations among each other (i.e. one contributes positively to the accomplishment of the

other). Such type of analysis is useful in many ways. Here, it provides intuitions on the reasons behind the CoP's involvement in the engage stage⁶. Moreover, although not explored in this chapter, the analyst may also profit from deeper contribution analysis, using qualitative and quantitative methods as described in [Giorgini, Mylopoulos, Sebastiani]. Among other possibilities, these methods explore positive and negative contributions among goals, consider different contribution strengths, and propagate the probability of goal achievement based on contribution analysis.

Figure 7 exhibits the result of the brainstorming sessions promoted by the KM Division, depicting the CoP initial internal structure. Although mediating the session, the KM Division let to the CoP members the decision regarding which roles should be defined to compose such internal structure, thus granting them with **autonomy** in this regard.

Figure 7 shows the CoP is composed of **Members**, who can play two main roles: **Leader** and **Webmaster**. The former is responsible for foreseeing new opportunities of knowledge creation and exchange among the CoP members (proposing new tasks goal), besides motivating knowledge sharing as it is currently targeted (encouraging knowledge sharing goal). The latter should mainly develop and maintain a website to be used as a portal by CoP Members (managing CoP website goal), who particularly felt they could profit from an online newsletter to share information about their activities (creating online

newsletter goal). It was decided in the brainstorming session that the leader goal should be initially played by two individuals, while the webmaster role would be played by one employee with experience on websites development.

The definition of roles is perfectly compliant with Constructivist KM. The fact that such paradigm privileges **non-hierarchical knowledge sharing** does not exclude different functions and positions. On the contrary, we admit that roles and hierarchies may be important in the organization's overall distribution of goals and plans. However, the point we make is that specifically regarding *knowledge processes*, all individuals should be treated as equals. In other words, all organization and community members should be viewed both as consumers and producers of valuable knowledge independently of the roles they play.

The roles of Figure 7 are aimed at facilitating knowledge creation, integration and dissemination, each one having specific goals and plans. However, the definition and refinement of these goals and plans is the focus of the architectural and detailed design and are thus, out of the scope of the present chapter.

4.3.3. The Support Phase

After a CoP has been created and engaged in work, the KM Division work has hardly finished (see Figure 8). Supporting a CoP's work rather than being a phase is a long lasting and important activity to guarantee the well functioning of the community.

The analyst suggested to the KM Division the idea to advertise the achievements of the CoP in a newsletter distributed throughout the organization, aiming at promoting recognition to motivate further involvement of CoP Members in their activities (advertising achievements in newsletter plan contributing positively to providing incentives goal). Moreover, carrying out periodic meetings with the Leaders of the CoP was considered a natural strategy to assist monitoring of the CoP activities (meeting periodically with CoP leader plan), both for finding new support needs and for verifying if the community is still in the right track according to the goals primarily established (positive contribution from this plan to monitoring and providing infrastructure goals).

However, other ideas naturally emerged throughout CoP activities and were reported to the KM Division. The Members early felt the need for an adequate technological support for sharing open source software information. Having gathered experience throughout several months of work, the Members were able to define general relevant properties of an open source tool, creating a kind of template for knowledge sharing. Getting acquainted with this, the KM Division decided to provide the means for the Webmaster to develop a simple system that helped the Members' storing and searching such templates (promoting development of template software plan). To accomplish this (i.e. to accomplish the developing template add-on to website goal), the Webmaster

needs support from the KM Division to dedicate necessary time for the development of the new system (having time and resources for add-on development goal).

In addition to this, one of the CoP Leaders noticed that there was an interest from many Members on meeting people responsible for the development of the open source software they used, proposing the realization of workshops where guests were invited to give talks. The KM Division immediately adopted this idea, foreseeing the benefits of this, both in terms of motivating new knowledge sharing and of contributing to the division's task of monitoring the CoP activities (positive contributions from the organizing yearly workshop for CoP Members plan to the providing incentives and monitoring goals).

The models of Figures 5 to 8 illustrate the shift from Late Requirements analysis to the Architectural Design activity. The Requirements analysis activities end up with the creation of a set of roles that should be assigned to people (more generally, specific agents) in the organization and a set of plans that should be further detailed during the Design activity. In this chapter, we do not go further into the Design, which typically requires a) a few cycles until the solution is clarified in terms of agents, roles, goals, resources and plans (architectural design) and b) yet some other cycles dedicated to describing a conceptual model of the proposed solution, along with a detailed model of the processes and behaviors of the involved agents (detailed design).

5. Related Work

ARKnowD has emerged from the combination of two previous work seeking the proposal of an agent-oriented methodology supporting KM (Guizzardi, Aroyo and Wagner, 2004; Perini, Bresciani, Yu and Molani, 2004). Perini et al. (2004) proposes a methodology for analyzing KM requirements based on intentional analysis, claiming that, in order to develop effective KM solutions, it is necessary to analyze the intentional dimension of the organizational setting, i.e. the interests, intents, and strategic relationships among the actors of the organization. Their methodology is based on the use of the i^* framework, the same used as a basis for the development of Tropos. In our approach, agent-oriented analysis leads to a design activity, not targeted in their initiative. Moreover, having adopted the Tropos methodology allows using a more clear agent-oriented semantics of i^* elements, which enables a smooth transition to our design approach. Guizzardi et al. (2004) propose AORML for KM domain modeling and design. Although it is possible to use AORML for domain modeling, this language lacks concepts and constructs to support requirements analysis. As motivated in section 2, supporting KM depends on a clear understanding of the problem setting, where requirements analysis comes as an essential step. In general, modeling with AORML starts with information modeling (like in UML class diagrams), jumping over the requirements analysis step. The solution proposed by Wagner (2003) for these initial phases is the use of Use Cases, however, we claim that our approach is

more appropriate for focusing on goals, supported by Nonaka and Takeuchi's emphasis on intention (i.e. goals) (Nonaka and Takeuchi, 1995) as the basis of any KM project.

Dignum describes OperA (Dignum, 2004), a recent proposal for an AMKM design framework and methodology. OperA uses the agent paradigm to analyze and model organizations and their knowledge needs, and to provide a reusable architecture to build KM systems. Different knowledge intensive tasks need knowledge from different sources and in different presentation formats. Therefore, the framework distinguishes between application, description and representation of knowledge and provides a common, uniform description of knowledge items (both sources and needs). A community of collaborative agents is responsible for the matching of knowledge supply and demand taking in account the user needs and preferences and the knowledge needs of a task. In our work shares with OperA the use similar concepts, such as actors, roles and goals. However, the modeling constructs applied are completely diverse, for instance, while OperA makes uses of scene scripts and provides a sound formal foundation based on temporal deontic logic, our proposal is much less formal, aiming at the support of the specification of KM systems through the use of a graphical language.

Related work may also be found in (Loucopoulos and Kavakli, 1999), where the authors propose a conceptual modeling approach to support enterprise KM.

This work shares many similarities to ours. It also proposes the analysis of the goals of the stakeholders, allowing the establishment of dependencies and support relationships, which are similar to what Tropos refers to as contribution. It also assigns processes as goal operationalization, as in ARKnowD (starting by the definition of plans to fulfill goals, and then its description with AORML interaction diagrams). However, besides goal, resource and activity dependency (this last one being analogous to Tropos plan dependency), their approach models different kinds of dependency, such as authority dependency and coordination dependency. Another divergence is that for them, the process of acquiring and maintaining knowledge refers to the structure and processes underlying the targeted organization. The authors focus on eliciting and representing this knowledge in a sort of business process analysis. Rather than a KM system, the result of this analysis is the proposal of an information system to automate the organization's processes.

6. Conclusion

This chapter presented a novel approach named ARKnowD to support the analysis and design of KM solutions in organizational settings. ARKnowD is intrinsically agent-oriented, recognizing the suitability of agents to be used as analysis abstractions, representing humans, organizations and systems involved in the organizational setting. Viewing the organization from an agent-oriented perspective enables a deep understanding of the social dimension characterizing

KM settings, thus supporting the analysis of the current practices and the proposal of innovative solutions, both in terms of new structures and processes supporting KM, or regarding the adoption of enabling technology.

Instead of proposing a new set of methodological concepts and constructs, ARKnowD relies on the combination of existing work on AOSE. The main motivation behind our choice of reusing existing work is to profit from their strengths while trying to compensate their limitations by combining two or more approaches, depending on the given situation.

In particular, this chapter focused on organizational reengineering to favor KM. In order to analyze the current setting as well as proposing solutions to the identified problems, we adopt the Tropos methodology, illustrating two of the four activities adopted in ARKnowD. The main contributions of the Tropos approach is supporting domain modeling in terms of the organizational structure and the emergent goals of each human or organizational agent involved in a given context. Other benefits are reached by the adoption of a visually rich and accessible notation, supported by existing modeling tools. We believe that providing such kind of informal visual methodology can be quite beneficial for the KM community, since business analysts and consultants will be able to use it without having to get acquainted with more formal approaches.

We used a fictitious scenario elaborated with basis on KM literature for the purposes of this exemplification. The scenario illustrates some of the main

problems involved in KM setting. We did not discuss the requirements elicitation methods since domain knowledge and information on the fictitious scenario are derived from existing literature. Thus, we rather focused on the Early and Late Requirements analysis, which are subsequent to the Requirements Elicitation activity in ARKnowD's lifecycle.

As aforementioned, this chapter privileged the exemplification of ARKnowD's support on proposing changes in the current organization's structure and process to accommodate and enhance KM. However, the development of the proposed template software (see Figure 7) and other supporting systems comprehend interesting work paths to be followed in the future. In this case, such systems can become actors in the model, to be further analyzed and designed, going through new cycles of the ARKnowD methodology.

Finally, future work on ARKnowD may be viewed according to theoretical and practical aspects. Theoretically, we hope to move forward with the work on the fundamentals behind our methodology, which may result in adding new constructs to ARKnowD's language, possibly also affecting the methodology's life cycle. As for the practical aspects, ARKnowD must undergo experimentation in real organizations. The scenario applied in this chapter is fictitious, thus resulting in a very controlled testing environment. Although we have been careful to be realistic, we are sure that when applied to a real case, some of our assumptions

may be confirmed, but also new insights and ideas should emerge to enhance our methodology.

¹ Method Engineering defines the concept of “method fragments”, meaning coherent portions of a methodology. A repository of methods fragments offers the bricks to set up an appropriate methodology for a specific situation (Harmsen, Brinkkember and Oei, 1994).

² All models in this chapter have been created with the use of the MS Office Visio software, using a Tropos stencil.

³ The reason for modeling team working well as a softgoal is the fact that the **Management** is not monitoring and measuring explicitly the team work quality. In the process of refining the goal analysis from the point of view of the organization's **Management**, the contribution of the **team working well** softgoal to the other goals of this actor can become more explicit.

⁴ At this stage, although considered as an actor in the model, CoP is an *abstract entity* rather than a real agent in the organization. In other words, CoPs do not yet exist in the organization but are rather an idea of the Top Management to enable KM. Conversely, all other actors in the model represent real agents in the organizational setting. The representation of CoPs as actors in the model is important to enable the understanding of the dependencies between this and the remaining actors in case this strategy is indeed adopted by the organization.

⁵ Due to lack of space, the Early Requirements analysis model illustrating this sentence was suppressed from this chapter, as this textual information is enough to motivate the proposal of our solution. Textual descriptions substitute Early Requirements models in several parts of this work from now on. Thus, Figure 3 is the only model of the Early Requirements analysis activity depicted in this chapter.

⁶ Although no goal dependencies between CoP and KM Division are depicted in this diagram, we remind the reader that the ones depicted in Figure 4 still hold.

References

1. Alavi, M. & Leidner, D. (1999). Knowledge Management Systems: Issues, Challenges and Benefits. *Communication of the Association for Information Systems*, 1(2), 1-37.

-
2. Bonifacio, M., & Bouquet, P. (2002). Distributed Knowledge Management: a Systemic Approach. In G. Minati & E. Pessa (Eds.), *Emergence in Complex, Cognitive, Social and Biological Systems*. New York: Kluwer Academic/Plenum Publishers.
 3. Bresciani, P., Giorgini, P., Giunchiglia, F., Mylopoulos, J., & Perini, A. (2004). Tropos: An Agent-Oriented Software Development Methodology. *International Journal of Autonomous Agents and Multi Agent Systems*, 8(3), 203-236.
 4. Dignum, V. (2004). A model for organizational interaction: based on agents, founded in logic. Ph.D. thesis, Utrecht: Utrecht University.
 5. Dignum, V., & van Eeden, P. (2005). Seducing, Engaging and Supporting Communities at Achmea. In W.R.J. Baets (Ed.), *Knowledge Management and Management Learning: Extending the Horizons of Knowledge-Based Management, Integrated Series in Information Systems* (pp. 125-141), 9, Berlin: Springer.
 6. van Elst, L., Dignum, V., & Abecker, A. (2004). Towards Agent-Mediated Knowledge Management. In L. van Elst, V. Dignum & A. Abecker (Eds.), *Agent-Mediated Knowledge Management: Selected Papers* (pp. 1-30). Heidelberg: Springer-Verlag.

-
7. Fischer, G., & Ostwald, J. (2001). Knowledge Management: Problems, Promises, Realities, and Challenges. *Institute of Electrical and Electronics Engineers Intelligent Systems*, 16(1), 60-72.
 8. Goguen, J.A., & Linde, C. (1993). Techniques for Requirements Elicitation. In S. Fickas & A. Finkelstein (Eds.), *In Proceedings of Requirements Engineering 1993* (pp. 152-164). Piscataway, NJ: Institute of Electrical and Electronics Engineers Computer Society.
 9. Guizzardi, R.S.S., Aroyo, L., & Wagner, G. (2004). Agent-oriented Knowledge Management in Learning Environments: A Peer-to-Peer Helpdesk Case Study. In L. van Elst, V. Dignum & A. Abecker (Eds.), *Agent-Mediated Knowledge Management: Selected Papers* (pp. 57-72). Heidelberg: Springer-Verlag.
 10. Guizzardi, R.S.S., & Perini, A. (2005). Analyzing Requirements of Knowledge Management Systems with the Support of Agent Organizations. *Journal of the Brazilian Computer Society - Special Issue on Agents Organizations*, 11(1), 51-62.
 11. Guizzardi, R.S.S. (2006). Agent-oriented Constructivist Knowledge Management. Ph.D. thesis, Enschede: University of Twente.
 12. Harmsen, F., Brinkkember, S., & Oei, H. (1994). Situational method engineering for information system project approaches. In A.A. Verrijn-Stuart

-
- & T.W. Olle (Eds.), *Methods and associated tools for the information systems life cycle* (pp. 169-194). Amsterdam: Elsevier.
13. Henderson-Sellers, B., & Giorgini, P. (Eds.). (2005). *Agent-Oriented Methodologies*. London: Idea Group.
14. Juan, T., Pearce, A., & Sterling, L. (2002). ROADMAP: Extending the Gaia Methodology for Complex Open Systems. In M. Gini, T. Ishida, C. Castelfranchi & W.L. Johnson (Eds.), *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems* (pp. 3-10). New York: Association for Computing Machinery Press.
15. Kavakli, E.V., & Loucopoulos, P. (2005). Goal Modeling in Requirements Engineering: Analysis and Critique of Current Methods. In J. Krogstie, T. Halpin & K. Siau (Eds.), *Information Modeling Methods and Methodologies* (pp. 102-124). London: Idea Group.
16. Loucopoulos, P., & Kavakli, E.V. (1999). Enterprise Knowledge Management and Conceptual Modelling. In *Selected Papers from the Symposium on Conceptual Modeling, Current Issues and Future Directions* (pp. 123-143). Berlin: Springer-Verlag.
17. Nonaka, I., & Takeuchi, H. (1995). *The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation*. New York: Oxford University Press.

-
18. Odell, J., Parunak, H.V.D., & Bauer, B. (2000). Extending UML for Agents.
In G. Wagner, Y. Lesperance & E. Yu (Eds.), *Proceedings of the Agent-Oriented Information Systems Workshop* (pp. 3-17), Austin, TX.
 19. Orlikowski, W.J., & Gash D.C. (1994). Technological Frames: Making Sense of Information Technology in Organizations. *Association for Computing Machinery Transactions on Information Systems*, 12(2), 174-207.
 20. Perini, A., Bresciani, P., Yu, E., & Molani, A. (2004). Intentional Analysis for Distributed Knowledge Management, In L. van Elst, V. Dignum & A. Abecker (Eds.), *Agent-Mediated Knowledge Management: Selected Papers*, (pp. 351-367). Heidelberg: Springer-Verlag.
 21. Wagner, G. (2003). The Agent-Object-Relationship Meta-Model: Towards a Unified View of State and Behavior. *Information Systems*, 28(5), 475-504.
 22. Wenger, E. (1998). *Communities of Practice: learning, meaning and identity*. New York: Cambridge University Press.
 23. Wooldridge, M., Jennings, N. R., & Kinny, D. (2000). The Gaia Methodology for Agent-Oriented Analysis and Design. *Journal of Autonomous Agents and Multi-Agent Systems*, 3(3), 285–312.

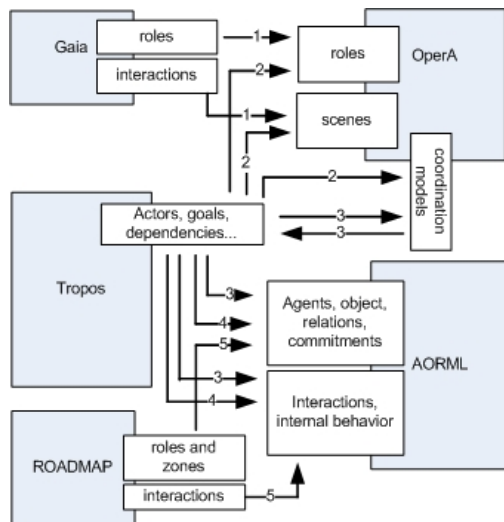


Figure 1: Combining different agent-oriented approaches

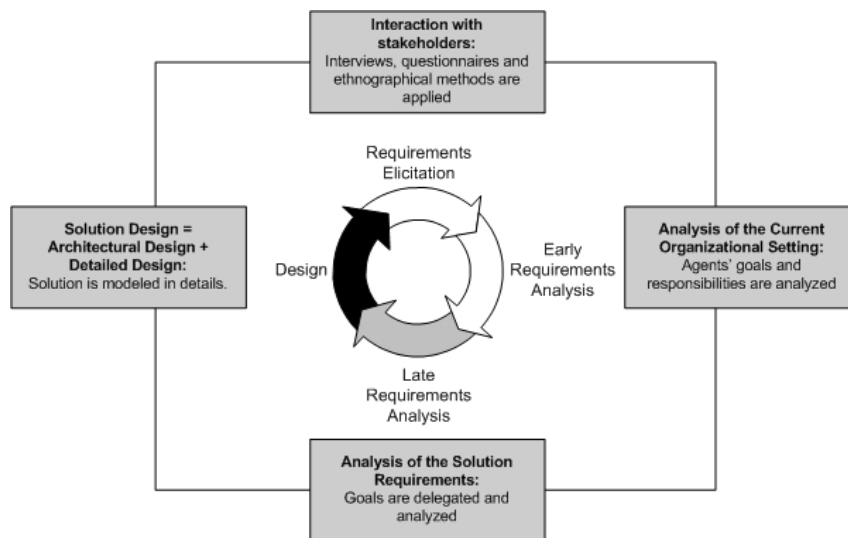


Figure 2. ARKnowD's engineering process

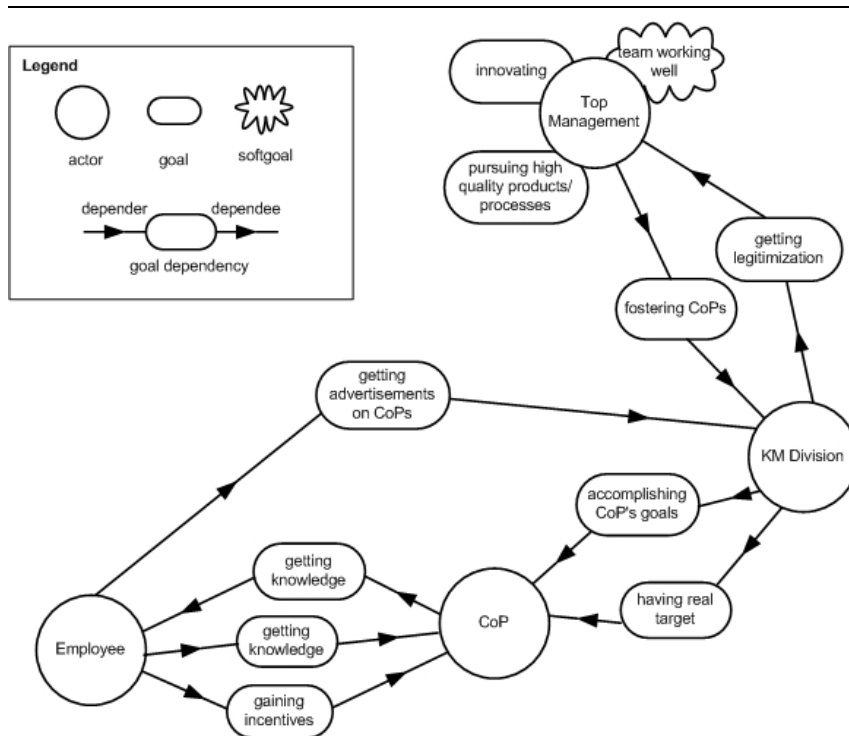


Figure 3. Early Requirements actor diagram showing the main goal delegations between the actors of the scenario

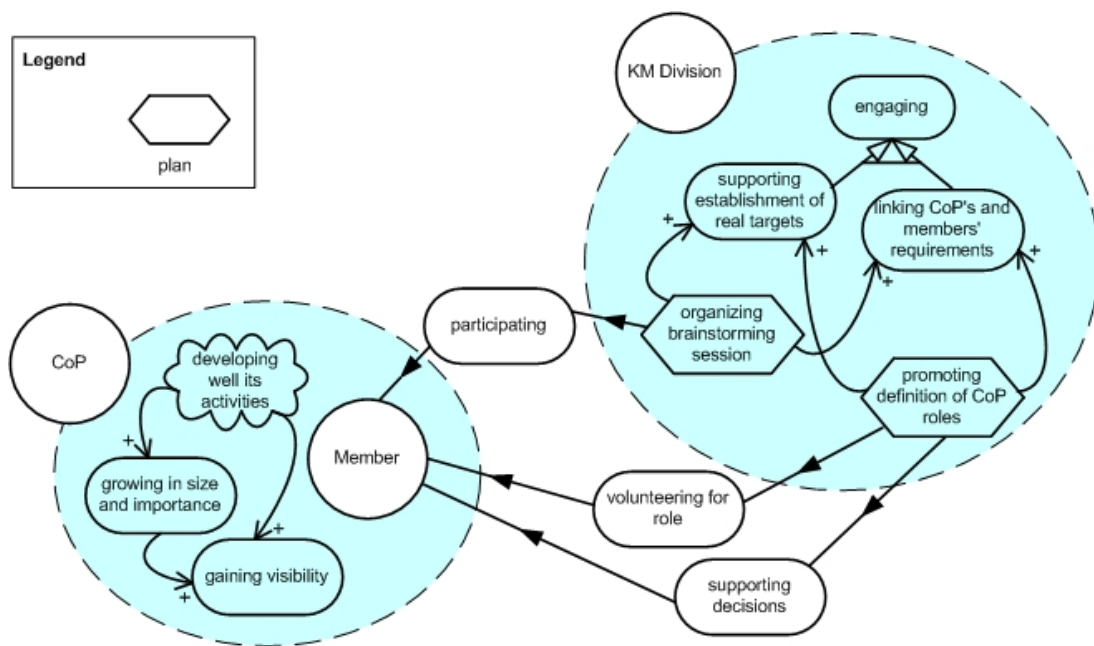


Figure 6. Addressing the SES engage phase

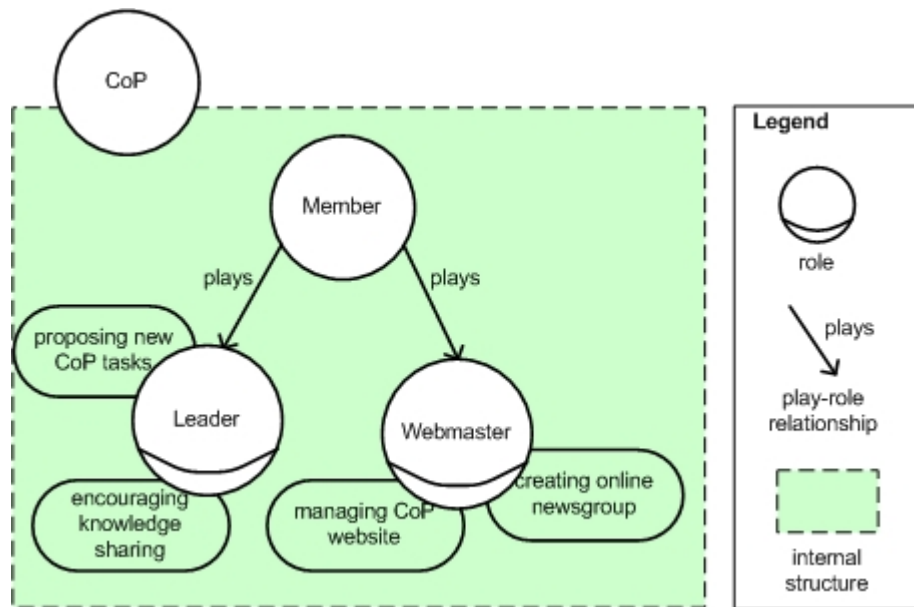


Figure 7. CoP internal structure

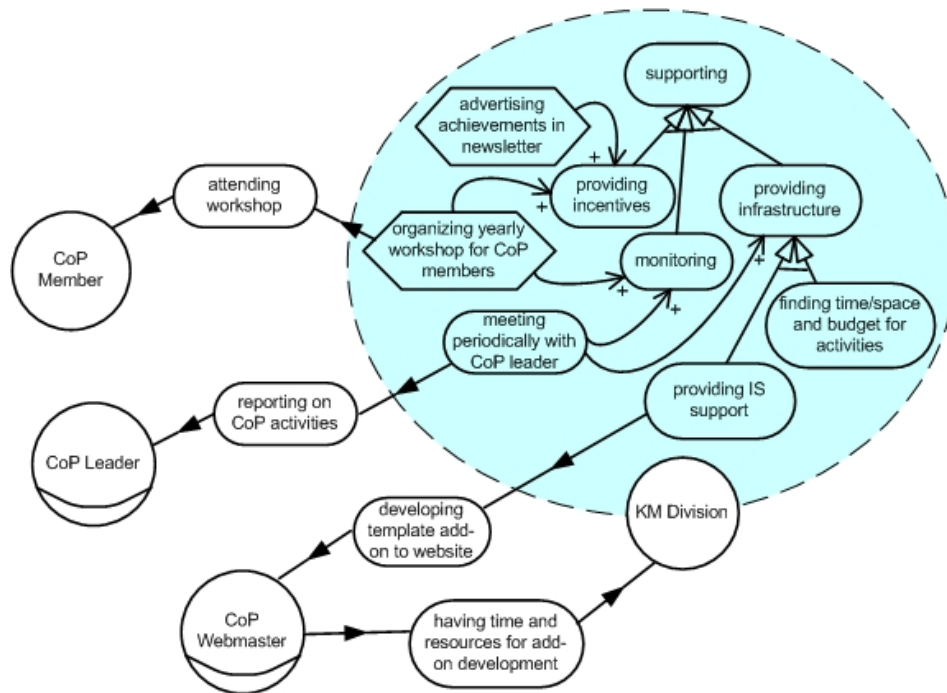


Figure 8. Supporting the CoP's long term work