Knowledge Management for Informally Structured Domains: Challenges and Proposals

Karla Olmos-Sánchez and Jorge Rodas-Osollo

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.70071

Abstract

Eliciting requirements of products or solutions in informally structured domains is a highly creative and complex activity due to the inherent characteristics of these domains, such as the great quantities of tacit knowledge used by domain specialists, the dynamic interaction between domain specialists and their environment in order to solve problems, the necessity of these solutions of products to be developed by teams of specialists and the asymmetry of knowledge between domain specialists and requirements engineers. The knowledge management discipline promotes an integrated approach in order to face these challenges; therefore, a strategy for addressing requirements elicitation that incorporates techniques and methods of this discipline has been proposed as a serious approach to deal with those challenges. The valuable results of the application of the strategy in real cases prove empirical insights about its utility.

Keywords: knowledge management, informally structured domains, requirements elicitation, tacit knowledge, knowledge creation spiral

1. Introduction

Requirements elicitation (RE) is a valuable process for the identification of *solution* requirements according to the need of clients of users [1]. In this chapter, the concept of *solutions* includes products, such as software systems or intangible solutions, such as data analysis. According to several authors, application domain knowledge is essential to obtain the correct and appropriate requirements. The application domain is an area where a solution is or will be used. Consequently, requirements engineers must understand, as soon as possible, the structure, the processes and the restrictions of a domain in which they are generally neophytes. This knowledge belongs to domain specialists, any person possessing application domain knowledge and/or having a role in the domain. Therefore, requirements engineers must elicit the application domain knowledge from domain specialists in order to include it into a set of solution requirements. It is a complex and highly creative activity that involves intensive cognitive activities, especially when the application domain has a high degree of informality where knowledge is informally stated, partially complete, implicitly assumed, tacit and unstructured [2].

This phenomenon is presented in many disciplines such as *intelligent tailored solutions for ill-structured domains, software for complex domains, intelligent tutoring systems, knowledge based systems, industrial design, among others*. In general, every necessity that requires a complex, highly creative solution, in which the requirements engineers are not a part of the application domain and need eliciting sufficient high-quality knowledge to understand the clients' need and expectations, faces this challenge [2]. Therefore, instead of focusing on the challenges of developing a requirements elicitation proposal for each of these complex areas, we have expanded the vision and generalized these domains as *informally structured domains* (ISD) [3], which is widely explained in Section 2.

In addition, solutions in ISD usually respond to clients and users' specific needs. As a result, they are diverse, consensus and unverifiable, and there are not fully defined processes to develop them. Therefore, these solutions or products must be developed according to the experience of domain specialists. These characteristics hamper the requirements elicitation process because the implications of knowledge transfer and transformation, the appropriate management of tacit knowledge and the issues of knowledge exchange must be considered.

In this context, we assume that a perspective of requirements elicitation that emphasizes the importance of knowledge management (KM) is a useful approach for addressing ISD inherent problems. KM is a discipline with the aim of enhancing an organization by sharing and managing knowledge flow among the people, taking advantage of information technologies [4]. Regarding KM in requirements elicitation is not new, but only few efforts offer a full knowledge management perspective [5].

The knowledge management strategy for requirements engineering (KMoS-RE©) [6] is a high-level plan oriented to the transfer or transformation of knowledge. The strategy has the aim of eliciting, structuring and creating knowledge that can be incorporated into a specification closest to the needs and expectations of clients. It is especially design from a full KM perspective in order to be applied in the context of ISD. The goal of this chapter is to describe the challenges of ISD and make a critical analysis of the KMoS-RE© strategy as a serious requirements elicitation proposal to face them. The analysis is based on the experience of applying the strategy in several ISD real cases. According the valuable results, the KMoS-RE© strategy promises to be a useful tool in the requirements elicitation of solution or products, especially in disciplines that share ISD characteristics [8].

The remainder of this chapter is organized as follows: Section 2 presents a characterization of ISD in order to explain the challenges of eliciting requirements in these domains. This section also includes a wide explanation of tacit knowledge. Section 3 describes fundamental concepts of KM in requirements elicitation. Section 4 discusses the utilization of KMoS-RE© as a serial proposal to face the challenges in ISD. Finally, in Section 5, the conclusions and future works are presented.

2. Informally structured domains

2.1. Tacit knowledge

As discussed above, a key element in a successful requirements elicitation process in ISD is *knowledge*. But, *what is knowledge?* Despite the widely recognised importance of knowledge as the main asset in today's society, defining it is an unresolved issue. In order to establish a baseline, this work supports the idea that knowledge has a subjective and personal quality. This view is based on the traditional definition of knowledge as *justified true belief*. However, as in Ref. [9], the focus is on the *justified* rather than the *true* aspect of belief. The *justified* view of knowledge makes it as dynamic, context-specific, humanistic, deeply rooted in individuals' value system and created in social interactions among individuals as opposed to the *true* view in which knowledge is absolute, static and non-human.

According to Ryle, knowledge can be classified in *knowing-that* and *knowing-how. Knowing-that* means storing and recalling facts. *Knowing-how* is a practical knowledge. This distinction carries through Polanyi's theory of personal knowledge, which classifies knowledge in *explicit* and *tacit* [10]. *Explicit knowledge* is transmitted through any language or formal representation: from text written in natural language to complex formalism as ontologies. On the other hand, *tacit knowledge* is personal and context-specific, generated by experience and therefore difficult to communicate and formalize.

Polanyi was interested in '… to show that complete objectivity, as usually attributed to the exact science, is a delusion and is in fact a false idea'. Thus, he examined how individuals gain and share knowledge. He concluded that knowledge is highly personal and questioned the commonly held view of the dispassionate objective scientist. He also emphasized that people can often know how to do things without either consciously knowing, or being able to articulate to others why what they do works.

According to Polanyi, *tacitness* is something personal, usually abilities or skills that people use to solve a problem or to do something valuable. *Tacitness* depends on people's experiences and learning. Polanyi suggested that all knowledge has a tacit component and discussed the process of how the tacit cooperates with the explicit. He also argued that language is a vital tool that people use to share knowledge, and that with the appropriate use of it, much, but not all, of this knowledge can be transmitted among individuals who share a mutually agreed language. When *tacitness* predominates, this articulation is not possible. However, it does not prevent knowledge from being transmitted by other means, such as observation or task repetition. This is what people do when learning to ride a bicycle or when an art master transfers knowledge to his or her apprentices. We should keep in mind that Polanyi's theory was generated in the field of psychology and his work was addressed towards perception. Thus, from Polanyi's perspective, any attempt to convert tacit knowledge to explicit will be unfruitful because it cannot be articulated at all.

Grant [11] provides a graphical representation of knowledge degradation as it is expressed by Polanyi's work (**Figure 1**). The bar represents how the knowledge is flowing in a continuum between tacit and explicit. The continuum ranges from knowledge inherently tacit to knowledge that can be easily expressed by words. The knowledge that can be expressed by words ranges from explicit to experts to explicit to most people. The knowledge explicit to experts requires specialized language. Most of this knowledge is also implicit, i.e. knowledge that can be expressed by words, but that for some reason it has not made explicit. The tacit knowledge ranges from ineffable to highly personal. Much of this knowledge is related to the use of instruments, such as playing piano or using a specialized machine.

To Gourlay [12], Polanyi's work has been misunderstood. He argued that some tacit knowledge does become amenable to analysis and decomposition, allowing recording it in an explicit form. Likewise, tacit knowledge in requirements elicitation has been misused. For example, Janik [12] has identified that the concept of tacit knowledge is used in two ways:

1. Concerning to knowledge that can be expressed, but for some reasons, it remains hidden. Janik identified three reasons why knowledge tends to remain tacit: (1) concern for secrecy and power, (2) because no one has bothered to recognize it or tried to explain it and (3) because it concerns presuppositions, we all generally hold. These situations can be aware, as the first one, or unaware, as the second and third ones. However, there are non-insuperable barriers to make this knowledge explicit.

2. Concerning to knowledge gained through familiarity and practice, which is inexpressible in words, or knowledge gained by perception as sight, smell or know-how. A wine taster or identifying an instrument when listening to a sound, are some examples of this knowledge.

What is really important in requirements elicitation is making the most possible quantity of knowledge explicit. Whether it is tacit, implicit or that for some reason remains hidden, even because nobody asks.

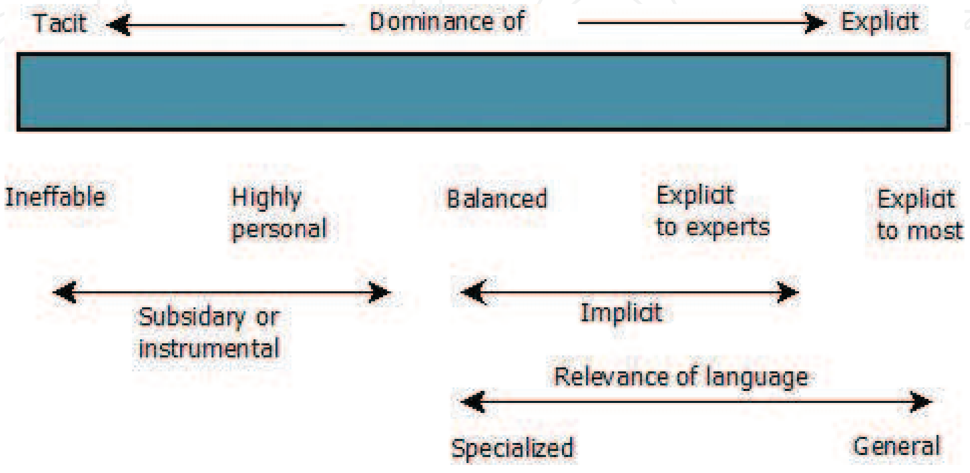

Figure 1. Granularity of knowledge.

The problem of tacit knowledge in requirements elicitation is not new. Goguen [13] did an extensive analysis of the term from a social perspective. He analysed several methods to elicit requirements such as introspection, questionnaires, interviews, focus group and even protocol analysis. He argued that these methods have limitations to manage tacit knowledge. To Goguen, it is indispensable considering a social perspective to attend this problem; thus, he suggests using combinations of these methods besides including discourse, conversations and interactions analysis.

Later, Nuseibeh [14] emphasized the importance of tacit knowledge and how it may affect the requirements of elicitation process. For him, the responses of the domain specialists to direct questions about their domain of expertise do not reflect, neither their current behaviour nor the reality, for the large amounts of tacit knowledge that is handled by them. Thus, product developers or solution solvers should consider theoretical and practical techniques of *cognitive psychology, anthropology, sociology* and *linguistics* to have better results.

The importance of sharing tacit knowledge to improve the problem-solving processes or as a strategy to gain competitive advantage in organizations is undeniable. For example, Wyatt [15] argues that much of the medical progress in modern times has been attributed to an evolution from tacit to explicit knowledge. Despite that, nowadays, tacit knowledge remains as an ambiguous and inconsistent concept. We are aware that not all knowledge of specialists is susceptible to becoming explicit; however, it is essential trying this transformation with a well-founded strategy for the requirements as close as possible to the reality of the application domain.

2.2. Formality and informality

Intuitively, a domain is a well-defined area of human activity with formal and informal issues in which a *universe of discourse* occurs. According to Webster's Dictionary, 'formal' means definite, orderly and methodical. In computer science, to be formal does not necessarily require the use of formal logic, or even mathematics, but the use of a formal notation to represent system models. Everything that computers do is formal because the syntactic structures in a program are manipulated according to well-defined rules [13]. In domains with a significant social context, much of the information is embedded in the social world of domain specialists; it is informal (not susceptible to be formalized or not yet formalized) and depends on the context for its interpretation [16]. These kinds of domains share characteristics such as informally defined concepts and lack of absolute verification of the processes; therefore, the domains specialists should use a great quantity of tacit knowledge to solve everyday situations.

Every domain is susceptible to be formalized to a certain level, but there will always be issues that remain informal. If a domain is mainly formal, the domain specialists can build, in a relative easy way, formal structures to solve problems. On the other hand, if a domain is mainly informal, it does not mean that domain specialists cannot build a structure; definitely they do. In some way, it is possible to solve diagnostic or design problems; however, these structures are informal, i.e. they depend on the context and the domain specialists' experience and knowledge. When informal characteristics prevail, the process and effort to solve problems can be extremely costly and time consuming.

Nguyen and Shanks [17] describe requirements elicitation as an ill-structured problem due to its openness, its context poorly understood and the existence of multiple domains. For Nguyen, solving problems in requirements elicitation requires a complex and dynamic social interaction between domain specialists and developers. The knowledge of both actors evolves as the project advances: the domain specialists get involved with software-solution and the developers with the organizational structure and business processes, i.e. the application domain. According to Nguyen, to solve ill-structured problems, understanding the problem and the structure of the solution are interleaved. The problem solvers, i.e. the requirements engineers, must explore different areas of the problem to find a solution. To accomplish this task, they communicate with the diverse actors who have domain knowledge or another perspective of the possible solution. By performing this task, their domain knowledge increases and they can return to previous stages of the problem, but with additional knowledge that allows them to explore new possibilities of solution. Therefore, the knowledge of the problem and its solution gradually evolves as the requirements engineers gain more knowledge of the domain, mainly due to social interactions and involvement of business processes.

We go further and assume that the grade of informality of the domain application influences the complexity of the process, as **Figure 2** depicts. Our focus is on the requirements elicitation process for domains with a high degree of informality, where knowledge is informally stated, partially complete, implicitly assumed, tacit and unstructured.

2.3. Characteristics of ISD

In order to effectively deal with ISD, we assume that they are located in the intersection of knowledge engineering and requirements engineering, and have the following characteristics [2]:

- Presence of multiple *domain specialists* who have different backgrounds, perspectives, interests and expectations, and whose knowledge, either tacit or explicit, of the application domain varies depending on their experience and their role in the domain. Usually, domain specialists are not aware of the details of the product or solution and only have a vague idea of its general functionality.

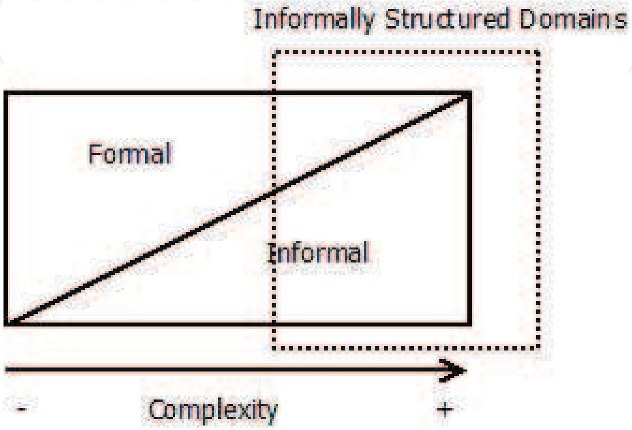

Figure 2. Complexity of domains.

- Presence of a group of *requirements engineers*, responsible for eliciting the requirements, who generally are not involved in the application domain. They have general technical knowledge about the development of the product or solution; however, they must elicit the application domain knowledge in order to understand the details of it and derive the correct and appropriate solution requirements.

- The solution solves or addresses a particular and unrepeatable situation. Thus, it must have its own design. However, there could be an infinite number of alternative solutions. In addition, the solution could be a tangible or intangible product and must be developed according to a requirements specification.

- A *requirements specification* is a document that contains the set of solution requirements. A requirement is a natural language statement to be enforced by the solution, possibly in cooperation with other system components, and formulated in terms of the application domain. The development of the requirements specification requires eliciting, synthesizing, validating, sharing and creating great quantities of application domain knowledge and solution knowledge in order to reach an acceptable solution. In addition, in order to develop the requirements specification, a dialectical thinking is necessary among all involved in the project.

Figure 3 depicts the characteristics described above; the figure represents *explicit knowledge* by puzzle pieces and *tacit knowledge* by clouds. The *requirements specification* is formed by pieces of knowledge of the domain specialists and requirements engineers. The requirements engineers must make the greatest possible amount of tacit knowledge explicit, synthesize the

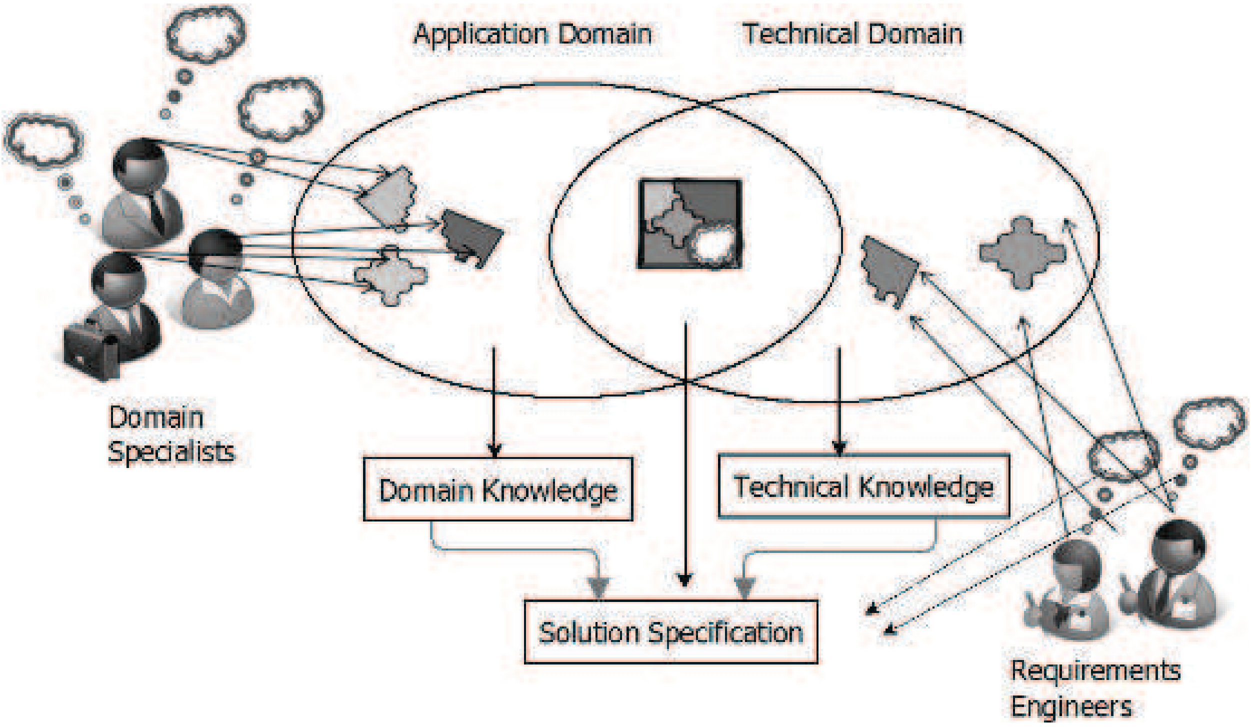

Figure 3. Informally structured domains.

disperse knowledge and reconcile the diverse beliefs and necessities of the domain specialists. In addition, they need to incorporate their own technical knowledge in order to generate the set of requirements of the solution. In the figure, this process is represented by the solved puzzle. The cloud in the solved puzzle means that there will always be knowledge that is not susceptible to be formalized.

2.4. Challenges in ISD

Some challenges of ISD are described as follows:

2.4.1. Tacit knowledge

Tacit knowledge can cause critical knowledge, goals, expectations or assumptions to remain hidden. In consequence, the emergent requirements will appear incomplete and inappropriate, which can cause poor systems or costly effects [18].

2.4.2. Situatedness

Situated actions involve a dynamic interaction with the actor and its environment; they only acquire meaning through interpretation in a specific context [19]. Situated actions involve conscious reference to the context and choice of action. An action is not situated if it takes the form of a prescribed response or if it is an unconscious automatic response. In ISD, situated actions occur frequently; in consequence, requirements are mostly situated and depend on a process of negotiation. In this situation, domain knowledge is fundamental in order to understand the rationality behind requirements, facilitate the negotiation process and propose technological aspects of the solution according to the real necessities of the domain specialists.

2.4.3. Disperse knowledge

Products or solutions in ISD are so complex that the human knowledge required to develop them generally is vastly larger than the maximum individual human capacities [20]. Therefore, organized teams formed of specialists must develop them. In order to cooperate in the solution, domain specialists must share some knowledge about the domain. However, they always have different backgrounds, perspectives, interests and expectations, and their knowledge and experience vary depending on their own practice and role in the domain. Sometimes even inconsistent and incompatible beliefs can exist. Product developers or solution-solvers should reconcile and prioritize the diverse beliefs and knowledge about the application domain in order to incorporate it to the solution.

2.4.4. Asymmetry of knowledge

Domain knowledge is the knowledge of the area to which a set of theoretical concepts is applied. In ISD, the concept of domain knowledge has two meanings. Firstly, solution domain knowledge corresponds to methods, techniques and tools that form the basis for the development of the product or solution. Secondly, those products or solutions are developed to necessities of real-word problems that exist in an application domain. Thus, both solution domain knowledge and application domain knowledge are necessary to develop the product or solution [20].

Asymmetry of knowledge, or symmetry of the ignorance, refers to the knowledge gap that exists between domain specialists, owners of the application domain knowledge, and requirements engineers, owners of the solution domain knowledge [5]. In ISDs, this phenomenon is increased because of the large amount of tacit knowledge involved. When the gap is big, there is not cognitive empathy and the communication process is not effective. Therefore, requirements engineers could produce models that do not represent the reality.

3. Knowledge management

Knowledge management (KM) is a discipline with the aim of enhancing an organization by sharing and managing the knowledge flow among the people [5]. KM is much more than just the use of information technology to manage knowledge. Due to the complexity of deal with knowledge, this discipline has developed theoretical concepts in order to explain and face the underlying problem of elicitation, creation, exchange and validation of knowledge. According to Pilat and Kaindl [5], there are three fundamental concepts in KM closely related to requirements elicitation such as the knowledge transfer and transformation process, the distinction between explicit and tacit knowledge and the issues of knowledge exchange. In addition, we consider that a creation knowledge process, where the knowledge of all involved in the project evolves, is also present in ISD.

3.1. Knowledge transfer and transformation process

The knowledge transfer process is carried out when the knowledge of a person is transformed into natural language, and in non-verbal channels of human communication in order to be transferred to another person, who then decodes this knowledge according to their own interpretation. Any transfer of knowledge is inherently bound to acknowledge transformation, so there will always be some degree of ambiguity. Ambiguity affects the elicitation of correct requirements because people involved in the project could build different and possibly incompatible interpretations of the concepts, relations and processes of the domain. Linguists point to several sources of ambiguity such as lexical, syntactic, semantic and pragmatic. ISD produce an additional kind of ambiguity named nocuous when two people mutually ignore that they have their own different interpretation. In this situation, they end up talking about different concepts while they think that they are talking about the same topic. According to Gacitua et al. [18], any person involved in the process can be aware of this phenomenon, because they do not have access to the tacit knowledge of each other.

3.2. Conversion of tacit knowledge to explicit

Nonaka and Takeuchi [7] propose a model of conversion of knowledge in organizations based on Polany’s theory of tacit knowledge. For them, knowledge creation in an organization is the result of social interaction where tacit and explicit knowledge is transferred. The model postulates four iterative conversion modes such as socialization, externalization, combination and internalization (SECI) which are described as follows:

- **Socialization** is the process of transferring tacit knowledge among individuals by sharing mental models and technical skills.
- **Externalization** is the process of converting tacit knowledge to explicit through the development of models, protocols and guidelines.
- **Combination** is the process of recombining or reconfiguring existing bodies of explicit knowledge to create new explicit knowledge.
- **Internalization** is the process of learning by repetition of tasks that apply explicit knowledge. Individuals will absorb the knowledge as tacit knowledge again.

According to Nonaka, if this cycle is done consciously, looping through this knowledge spiral may evolve the overall knowledge held collectively. The spiral of knowledge can be applied to requirements elicitation in order to face the inherent knowledge management challenges of the process. Despite that, we found just a few researches that explore this possibility. Wan et al. [21] proposed a model of knowledge conversion to the requirements elicitation process with the aim to minimize the symmetry of ignorance between developers and domain specialists. The authors base their model on the SECI model and consider the knowledge flowing between domain specialists and developers. They introduced a new agent in the process: the *requirements specialist*. This person would act as an intermediary between the domain specialists and the developers, so he or she must earn the trust of those involved in the process. The authors use their model to analyse a requirements elicitation process of a real software development project. In conclusion, the authors argue that the proposed model can reduce the symmetry of ignorance and facilitate the elicitation of tacit requirements. Nevertheless, to be successful in the process they suggest that the requirement specialists must have enough domain knowledge. We consider that it is difficult, if not impossible, that a person knows about every domain, so the incorporation of this new agent could hinder the, complex by itself, elicitation process. On other hand, Vásquez-Bravo et al. [22] proposed a classification of elicitation techniques to facilitate their selection in an RE process based on the phases of Nonaka’s model. However, they do not propose how to use these techniques and how to elicit tacit knowledge.

3.3. Knowledge sharing

In order to implement the knowledge spiral property, it is crucial to facilitate the exchange of knowledge among all involved in the project [16]. It implies focusing on the knowledge holders, especially in ISD where knowledge is mostly tacit. This task may become difficult to handle because requirements engineers can be confronted with several persons whom they do not know. KM offers the concept of knowledge map [5], an artefact that points to knowledge but does not contain it. The artefact could be a table or a matrix indicating which person has what knowledge. The knowledge map should be created and initialized at the beginning of the process and continually be updated as the spiral of knowledge evolves. A knowledge map is also useful to discover for which knowledge a knowledge holder might be missing. In

ISD, we assume that a knowledge map would also be useful for indicating the tacitness level of the knowledge holders. Thus, we propose the piece of knowledge (PoK) matrix, an artefact to fulfil the functions mentioned above and to be used in the KMoS-RE© strategy.

The PoK matrix is a data structure that stores the relation of every individual (solution solver or domain specialist) involved in the project with every piece of knowledge about the domain. A piece of knowledge can be a concept, a relationship or behaviour. The PoK matrix is used as a reference to figure out which concepts, relationships or behaviours had been made explicit and which of them remain tacit. The aim of the KMoS-RE© strategy is to look for the transformation, from 0 to 1, of the most possible values in the PoK matrix. This is in order to make explicit the most possible quantity of tacit knowledge. It would be ideal if the requirements engineers could make explicit all pieces of knowledge. However, there will always be knowledge that it cannot be converted to explicit; therefore, the requirements engineers must propose the most suitable solution with the explicit knowledge obtained at a particular moment.

3.4. Knowledge evolution spiral

As was mentioned above, in ISD, understanding the problem and the structure of the solution are intertwined. The problem solvers, that is, the requirements engineers, must explore different areas of the problem to find a solution. In order to accomplish this task, they should dialogue with the domain specialists, who have their own domain knowledge and perspective of the possible solution. By performing this task, the knowledge of the problem solvers about the application domain evolves. If were necessary, they can return to previous states of the project, but their knowledge is not the same, they will have additional knowledge that allows them to explore new possibilities of solution. In summary, the knowledge of the problem and its solution gradually evolves as requirements engineers gain more knowledge of the application domain due to social interaction and their involvement with the business processes. In order to model that behaviour, the knowledge evolution model for requirements elicitation (KEM-RE) was developed based on the SECI model. The KEM-RE is an iterative cycle (**Figure 4**) that consists of four stages that include the four kinds of knowledge processes in the innovation of complex problem solving:

- **Knowledge elicitation (KE) stage**. The requirements engineers elicit knowledge from domain specialists and vice versa. The socialization mode predominates.
- **Knowledge integration and application (KI & A) stage**. The requirements engineers integrate the acquired knowledge and their own experience into models. This is a complex activity in which combination and externalization modes are presented. In addition, as the requirement engineers develop models they internalize the domain knowledge.
- **Knowledge sharing and exchange (KS & E) stage**. The models developed by requirements engineers will be shared with the domain specialists. This phase takes place through socialization.
- **Knowledge validation (KV) stage**. The domain specialists validate the models. In order to develop this activity, they must internalize the knowledge behind the models through a cognitive dialogue. This process leads to the elicitation of new knowledge. Then the cycle starts again.

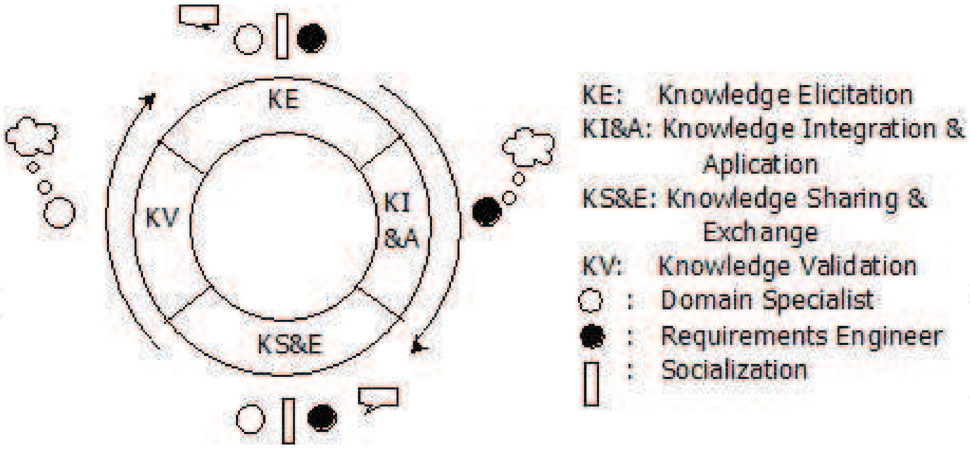
KE
KI &A
KS&E
KV
KE: Knowledge Elicitation
KI&A: Knowledge Integration & Aplication
KS&E: Knowledge Sharing & Exchange
KV: Knowledge Validation
○ : Domain Specialist
● : Requirements Engineer
▯ : Socialization

Figure 4. Knowledge evolution model for requirements elicitation.

4. KMoS-RE©: an approach from knowledge management discipline

The KMoS-RE© strategy [6] is a high-level plan to achieve a set of requirements of a solution or product through the eliciting, structuring and creating of knowledge. Following the work of [24], the strategy consists of three phases: *domain modelling (DM), system modelling (SM)* and *specification developing (SD)* and structures its flow of activities according the KEM-RE. Furthermore, it includes transversal activities to identify and make explicit the most possible quantity of tacit knowledge. Those activities are conducted by the identification of presuppositions [18] and classification of verbs according the Blooms' taxonomy [23]. The strategy also includes artefacts to facilitate the sharing knowledge: a record of wrong beliefs and the PoK matrix (Section 4.3). A brief explanation of each phase is provided as follows:

- **Domain modelling phase (DM)**. In this phase, the terms, i.e. the concepts, attributes and relationships, and the basic integrity restrictions are formalized through a consensus, in order to understand the application domain without worry about the solution. The terms are recorded in the Knowledge of Domain on an Extended Lexicon (KDEL); a lexical that classifies them into objects, subjects and verbs. The KDEL is used to facilitate the building of a graphical conceptual model. The externalization of this knowledge will enable achievement a consensus among the stakeholders; hence to minimize the symmetry of ignorance. The concepts and relationships identified in this phase will generate the first version of the piece of knowledge (PoK) matrix. In addition, a graphical conceptual model is required in order to facilitate the cognitive dialogue with the domain specialists. Requirements engineers will decide what kind of conceptual model use, from entity-relationship model to ontologies, depending on the characteristics of the domain.
- **System modelling phase (SM)**. In this phase, the current and future system processes are formalized. The current system corresponds to the system, as it exists at present. The future system represents the system after the deployment of a solution or product. The Use Cases technique was selected to model the system, both current and future, because its usefulness

has been demonstrated through the time. The system model is obtained from the KDEL and the conceptual model. The behaviours identified in this phase will also change the values of the PoK matrix.

- **Specification development phase (SD)**. In this phase, the requirements are derived from the Uses Cases' scenarios of the future system and incorporated into the solution requirements specification (SlRS).

Figure 5 depicts a general view of the KMoS-RE© strategy in a unified modelling language (UML) activity diagram. Every activity of the strategy corresponds to one stage of the KEM-RE: model validations (MV) is related to knowledge validation (KV), knowledge elicitation (KE) is related with the stage of the same name, model discussion (MD) corresponds to knowledge sharing and exchange (KS & E) and domain modelling (DM), system modelling (SM) and specification development (SD) correspond to knowledge integration and application (KI & A). The swim lanes in the figure represent the activities developed by each type of actor.

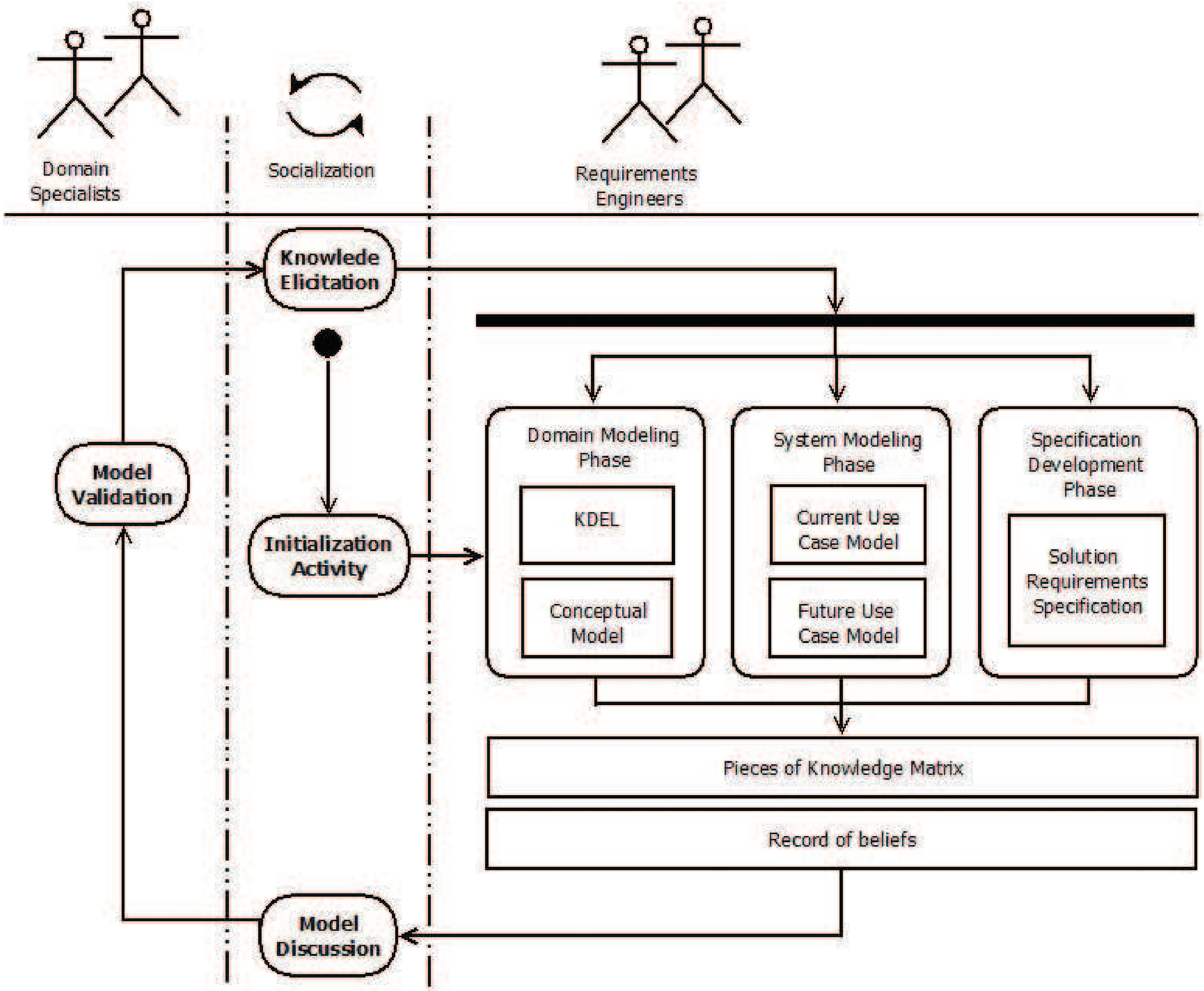

Figure 5. Knowledge management on a strategy for RE.

4.1. KMoS-RE© analysis

According to Maalej and Thurimella [24], managing requirements knowledge is about efficiently identifying, accessing, externalizing and sharing domain and technical knowledge by and to all involved in the project, including analysts, developers, and domain specialists, which is closely related to a full perspective of KM. The rationality of the KMoS-RE© strategy is based on the fundamental issues described as follows:

- The flow of activities in the KMoS-RE© strategy is based on the knowledge evolution model KEM-RE, which is based on the knowledge evolution spiral proposed by Nonaka. The evolution spiral knowledge has the aim of facilitating the conversion of tacit knowledge to explicit. In addition, the incorporation of techniques such as the identification of presuppositions and the classification of verbs according the Bloom's taxonomy make easier-identifying knowledge that could be tacit, and hence hidden.

- Representing requirements knowledge targets an efficient information access and artefact reuse within and between projects. The KMoS-RE© strategy proposes several artefacts in order to represent different views of the system. They can be accessed and shared by all involved in the project. Although several requirements elicitation proposals use lexical, conceptual models, use cases models and scenarios, few of them combine those techniques in a strategy. Besides, the KMoS-RE© strategy proposes two innovative artefacts: the record of belief and the PoK matrix.

- Sharing requirements knowledge improves the collaboration among all involved in the project and ensures that their experiences do not get lost. The knowledge spiral in which the activities of the KMoS-RE© strategy are based compels to sharing the knowledge among solution-solvers and domain specialists through socialization.

- Reasoning about requirements and their interdependencies aims at detecting inconsistencies and deriving new knowledge. Externalizing the knowledge through the development of the different artefacts let the solution-solver reason and internalized the domain knowledge.

4.2. KMoS-RE© applied to real ISD cases

The KMoS-RE© strategy has been applied in the development of solution of several real ISD cases:

- **Software development for complex domains**. This is a complex and creative activity in which software developers should understand, as soon as possible, the knowledge of a domain in which generally are neophytes. Then, combine this knowledge with their own technical knowledge in order to reach a solution that meets clients' expectations. The KMoS-RE© strategy has been used to develop a cognitive rehabilitation system for sclerosis multiple patients [6].

- **Soft computing**. This artificial intelligence (AI) subarea includes several techniques that are suitable for solutions in ISD, since it is tolerant of imprecision, uncertainty, partial truth and approximation. A complex problem in soft computing is how to elicit the knowledge of specialists in order to incorporate it in an appropriate representation and to reach correct solutions. A case-based reasoning system to support heating ventilation and air conditioning (HVAC) design decisions was developed using the KMoS-RE© strategy [8].

- **Intelligent tutoring systems**. Over the past decade, intelligent tutoring systems have become increasingly accepted as viable learning tools in academia and industry. However, most of these solutions had been developed for well-defined domains. Informally structured domains, such as computer programming, laws and ethics, present a number of unique challenges for researchers in intelligent tutoring, especially to represent and evaluate the knowledge. We are currently exploring the adaptation of the KMoS-RE© strategy with the aim of getting a method to develop Bayesian Networks for evaluation in intelligent tutoring systems in the context of ISD.
- **Industrial design**. The KMoS-RE© strategy had been used as a HVAC requirements process in a real company [8]. The HVAC design is a difficult task because the information necessary for solving the problem is incomplete and vague. This knowledge belongs to the domain specialists, generally a set of specialists from different fields, such as mechanical engineers, control engineers, electrical engineers and architects. In addition, there could be multiple and controversial solutions and the criteria that determine the best design solution is complex and imprecise.

4.3. Discussion

Nowadays, the negative effects of inappropriate, incorrect and ambiguous requirements have been widely studied and are well known. Despite the vast quantity of proposals, methods, techniques and tools, requirements elicitation is still an open problem, as shown by many projects that do not fulfil clients' expectations or that exceed the development time due to bad elicited requirements. Thus, there are still clear opportunities to improve.

The application of the KMoS-RE© strategy in real ISD cases has showed that its characteristics are clear contributions to the requirements elicitation area, as it is described as follows:

- **Emphasis on application domain knowledge**. The importance of the application domain knowledge in order to improve the requirements elicitation process is widely accepted. However, currently the most of the methods and tools of requirements elicitation are designed for general problem domains, where problem-specific domain knowledge is not completely necessary [25]. The KMoS-RE© strategy emphasizes the importance of the application domain knowledge, either tacit or explicit, besides of proposing techniques and methods to facilitate its discovery, representation, sharing and appropriation among all involved in the project. Thus, the strategy can be applied in knowledge-intensive projects.
- **Generality and adaptability**. The theoretical concepts of knowledge management, in which the KMoS-RE© strategy is supported, allow it to be applied to domains with different levels of informality. It also has the advantage of being considered as a high-level plan; therefore, the requirements engineers have the authority to decide which phases are necessary. In some cases, they can even choose between different techniques and methods.
- **Evolutive**. The strategy is not limited to the proposed so far. The model allows the incorporation of new proposals from methods and techniques to knowledge management models or perspectives of elicitation of requirements. For example, we are currently analysing the

adaptation that is based on goals elicitation approach [26], the knowledge audit model [25] and the social network analysis [27].

- **Algorithmic**. Despite its generality and adaptability, the KMoS-RE© strategy is algorithmic in the sense that the process of its implementation is well defined and limited, so the requirements engineers do not need a deep knowledge about the theoretical concepts of knowledge management. Most of the cases showed in the previous section were led by undergraduate engineering students [8]. Although a process of awareness of the issues of informal structured domains is always recommended.

Finally, we would like to emphasize that the most important contribution of the KMoS-RE© strategy is that it does not try to work against human nature; it recognizes its capabilities and limitations and builds the best proposal based on that. Thus, according to the below and the valuable results of the application of the KMoS-RE© strategy in several and different contexts, it can already be considered as a serious approach for requirements elicitation knowledge.

5. Conclusions and future works

The KMoS-RE© strategy is a novel approach from KM in order to face the challenges of eliciting knowledge and creatively transform it into a set of requirements of a product or solution in order to satisfy the needs and fulfil whole expectations of clients and users. The strategy is focused on dealing with ISD. Due to the characteristics of these domains, the strategy has a full KM perspective, i.e. it incorporates knowledge engineering techniques in order to properly manage tacit knowledge. The *domain modelling phase* handles the problem of formalizing the concepts and relationships; at least a consensus about it is reached. The *system modelling phase* deals with the problem of structuring the processes in the domain. Thus, the problem of handling tacit knowledge has addressed properly by KMoS-RE© strategy.

The strategy was applied to several ISD real cases in different and diverse areas. The solutions achieved provide evidence about the usefulness, the value and the generalization of it. Therefore, the application of the KMoS-RE© strategy in several real cases shows that it is a useful approach in order to elicit the requirement of solutions or products especially in ISD.

Finally, the challenge of managing the tacit knowledge requires to analyze more cases in order to improve all the KM approaches including the KMoS-RE© strategy.

Author details

Karla Olmos-Sánchez* and Jorge Rodas-Osollo

*Address all correspondence to: kolmos@uacj.mx

Autonomous University of Ciudad Juárez, Ciudad Juárez, Chihuahua, México

References

[1] Hansen S, Berente N, Lyytinen K. Requirements in the 21st century: Current practice and emerging trends. In: Design Requirements Engineering: A Ten-Year Perspective. Springer Berlin, Heidelberg; 2009. pp. 44-87

[2] Olmos-Sánchez K, Rodas-Osollo J. A strategy of requirements engineering for informally structured domains. The International Journal of Combinatorial Optimization Problems and Informatics. 2016;**7**(2):49-56

[3] Olmos K, Rodas J. Requirements engineering process model for informal structural domains. International Journal of Computers Communications & Control. 2013;**2**(1):75-77

[4] Chen L, Fong P. Evaluation of knowledge management performance: An organic approach. Information and Management. 2015;**52**(4):431-453

[5] Pilat L, Kaindl H. A knowledge management perspective of requirements engineering. In: Proceedings of the Fifth International Conference on Research Challenges in Information Science (RCIS), 2011; 19-21 May 2011; Gosier, France. IEEE; 2011. pp. 1-12

[6] Olmos K, Rodas J. KMoS-RE knowledge management on a strategy to requirements engineering. Special Issue Requirements Engineering Software Product Line, Requirements Engineering Jornal 2014;**19**(4):421-440

[7] Nonaka I, Takeuchi H. The knowledge-creating company: How Japanese companies foster creativity and innovation for competitive advantage. Harvard Business Review. 1995;**69**(6):96-104

[8] Olmos-Sánchez K. Knowledge Management on a Strategy for Requirements Engineering. Universidad Autónoma de Ciudad Juárez; 2015

[9] Nonaka I, Toyama R, Konno N. SECI, Ba and leadership: A unified model of dynamic knowledge creation. Long Range Planning. 2000;**33**(1):5-34

[10] Polanyi M. The Tacit Dimension. Chicago, United States of America: University of Chicago Press; 1966

[11] Grant KA. Tacit knowledge revisited—We can still learn from Polanyi. Electronic Journal of Knowledge Management. 2007;**5**(2):173-180

[12] Gourlay S. Towards conceptual clarity for 'tacit knowledge': A review of empirical studies. Knowledge Management Research & Practice. 2006;**4**(1):60-69

[13] Goguen JA. Formality and informality in requirements engineering. In: Proceedings of the 4th International Conference on Requirements Engineering; IEEE Computer Society, Washington, DC, USA 1996, pp. 102-108

[14] Ma L, Nuseibeh B, Piwek P, De Roeck A, Willis A. On presuppositions in requirements. In: Proceedings of the Second International Workshop on Managing Requirements Knowledge (MARK), 2009; 1 Sept. 2009; Atlanta, GA, USA. IEEE; 2009. pp. 27-31

[15] Wyatt JC. Management of explicit and tacit knowledge. Journal of the Royal Society of Medicine. 2001;**94**(1):6

[16] Sutcliffe A, Sawyer P. Requirements elicitation: Towards the unknown unknowns. In: 2013 21st IEEE International Requirements Engineering Conference, RE 2013—Proceedings; 15-19 July 2013; Rio de Janeiro, Brazil. IEEE; 2013. pp. 92-104

[17] Nguyen L, Shanks G. A framework for understanding creativity in requirements engineering. Information and Software Technology. 2009;**51**(3):655-662

[18] Gacitua R, Ma L, Nuseibeh B, Piwek P, de Roeck AN, Rouncefield M, Sawyer P, Willis A, Yang H. Making tacit requirements explicit. In: Proceedings of the Second International Workshop on Managing Requirements Knowledge (MARK) 2009; 1-1 Sept. 2009; Atlanta, GA, USA. IEEE; 2009. pp. 40-44

[19] Fernandes KJ. Interactive situation modeling in knowledge-intensive domains. International Journal of Business Information Systems. 2009;**4**(1):25-46

[20] Vessey, I. The Effect of the Application Domain in IS Problem Solving: A Theoretical Analysis. In: Information Systems Foundations: Theory, Representation and Reality, edited by Hart Dennis N. and Gregor Shirley D., 25-48. ANU Press, 2007

[21] Wan J, Zhang H, Wan D, Huang D. Research on knowledge creation in software requirement development. Journal of Software Engineering and Applications. 2010;**3**(5):487-494

[22] Vásquez-Bravo D-M, Sánchez-Segura M-I, Medina-Domínguez F, Amescua A. Combining software engineering elicitation technique with the knowledge management lifecycle. International Journal of Knowledge Society Research. 2012;**3**(1):1-13

[23] Bloom, BS. Taxonomy of educational objectives. Vol. 1: Cognitive domain. New York: McKay. 1956:20-4

[24] Maalej W, Thurimella AK. An introduction to requirements knowledge. In: Managing Requirements Knowledge. Berlin, Heidelberg: Springer; 2013. pp. 1-20

[25] Taheri L, Che Pa N, Abdullah R, Abdullah S. A knowledge audit model to assess the knowledge in requirement elicitation process. In: 2015 9th Malaysian Software Engineering Conference (MySEC); 16-17 Dec. 2015; Kuala Lumpur, Malaysia. IEEE; 2015. pp. 106-111

[26] Franch X, López L, Cares C, Colomer D. The i* framework for Goal-Oriented modeling. In: Domain-Specific Conceptual Modeling. Cham: Springer International Publishing; 2016. pp. 485-506

[27] Lim SL, Ncube C. Social networks and crowdsourcing for stakeholder analysis in system of systems projects. In: 2013 8th International Conference on System of Systems Engineering; 2 Jun-06 Jun 2013; Wailea-Makena, HI, USA. 2013. pp. 13-18