Knowledge sharing in change management : a case study in the French railways company

Anne Remillieux, Claire Petitmengin, Jean-Louis Ermine, Christian Blatter

Knowledge Sharing in Change Management, A Case Study in the French Railways Company

Anne Remillieux [a] Claire Petitmengin [a] Jean-Louis Ermine [a] and Christian Blatter [b]
[a] *Institut Télécom, 9 rue Charles Fourier 91011 Evry Cedex, France*
E-mail: {anne.remillieux, claire.petitmengin, jean-louis.ermine}@it-sudparis.eu
[b] *SNCF, 45 rue de Londres, 75 379 Paris Cedex 08, France*
E-mail: christian.blatter@sncf.fr

Abstract. This paper describes a project to represent and share tacit change management knowledge. Based on an ontology, this project enables French Railways Company (SNCF)'s project managers to share their know-how through an IT device called «knowledge server». This paper contains two kinds of contributions. The first contribution is the ontology itself. The second contribution is methodological. By describing the ontology's design process, based on a study of project managers' practices, the paper indicates how to build an ontology that refers to tacit knowledge, to human professional know-how.

Keywords: Knowledge management, Tacit knowledge sharing, Change management, Conceptual graphs, Ontology

1. Introduction

In order to promote a change management approach within the company, the National French Railways Company (SNCF) decided to make available to its staff a tool to share their knowledge and experiences on the domain; an Ontology for Change Management (OCM) was developed in the following context. To satisfy this request, SNCF needs to design a model in which the change managers' knowledge can be formalized, capitalized, and, finally, shared.

Existing models are inappropriate for those requirements, which are situated between goal-oriented modelling (Lamsweerde & Willemet 1998, Lamsweerde 2001, Nurcan & Rolland 2003, Kavakli & Loucopoulos 2006) and argumentation models (Verheij 2005, Echtler 2006). First, similarly as design methodologies and goal-oriented approaches, we aim at facilitating decision making and communication. However, whereas goal-oriented approaches basically deal with *the world to design*, OCM concerns *ways of knowing and designing*, like argumentation models do. However, our model should also be distinguished from argumentation models: whereas the latter model negotiation processes, especially for interfaces of dialog systems, OCM's objective is to allow actors to share their knowledge.

We have chosen a specific model adapted to our aim. This model is an ontology that represents change management knowledge in the formalism of conceptual graphs (Sowa 1984). Once implemented, this ontology is used as a structure for a change management knowledge server. By "knowledge server", we mean "an information system allowing users to improve their practice". In other words, a knowledge server is a system which, instead of simulating human reasoning as an expert system would do, provides the user with some support for reasoning by analyzing the knowledge he needs. As the possible strategies for change management are diverse and strongly context dependent, it is a mean for encouraging users' reasoning and action, instead of guiding them towards a single recommendation resulting from automatic reasoning.

In order to build an ontology of change management knowledge, we could use a foundational ontology, but no existing ontology is adequate because of the particular nature of change management knowledge.

Indeed, the concerned knowledge domain can be considered rather as a *know-how*[1] than as a particular scientific domain, as change management involves taking human factors into account when managing a project. Although change managers can refer to a substantial body of literature,[2] change management remains a practice which is acquired by experience, i.e. repeated confrontation of the practitioner with various problematic situations. This leads to a type of knowledge that is embedded in action and very implicit. Several degrees of implicitness exist, from knowledge which is conscious but not formalized to knowledge which is not conscious.[3] This latter degree of implicitness, inaccessible without some methodical elicitation (Vermersch 2001), is what (Polanyi 2002) calls "the tacit".

Existing ontologies are not adapted to tacit knowledge representation since current research practice on ontology engineering does not meet the issue of tacit knowledge elicitation. Yet, the most often quoted definition of the term "ontology" in Information Science is that of (Gruber 1993): '*An ontology is a formal explicit specification of a conceptualization*". According to this definition, an ontology is a translation of a conceptual model into a formal language. Whatever the level of formalisation is, the ontology design is a process of elicitation: it involves translating information into a form which guides the receiver towards a specific interpretation. More precisely, we identify four stages of elicitation in the ontology design process, i.e. four stages of translation of information into a form which leaves less and less place for ambiguity and implicitness (See Figure 1). Among those stages, only the last three ones are taken into account by current research. The first stage entails translating into linguistic expressions the reality such as it is perceived or experienced by the subject whose knowledge has to be modelled. The second stage consists in identifying the instances, i.e. the expressions which constitute the extension of various concepts within the domain, as they occur in the subject's discourse. The third stage, known as "conceptualization", involves categorizing those instances while bringing out the properties they share. The fourth and final stage, which is evoked by (Gruber 1993), consists in translating the obtained conceptual model into an ontology, which is a structured set of formally defined concepts and relations.

Figure 1. The different elicitation levels involved in the ontology design process

The first stage of this elicitation process seldom raises an issue because, most of the times, the domain to be modelled is easily accessed. Indeed, at first, the domain is often already expressed in a corpus of texts as in the case of Semantic Web or medical ontologies using terminology extracted from texts (Baneyx & Charlet 2005). In this context, the "reality" is immediately accessible in the universe of discourse. The elicitation work then essentially consists in identifying the instances and gathering them into concepts.

Furthermore, when the domain to be modelled is not already expressed in a corpus of texts, it is a technical or physical reality (Bennett 2005) whose structure is already well known, such as chemistry (Fernandez-Lopez et al. 1999) or medicine (Zweigenbaum & Menelas 1995).

Consequently, authors who are interested in the ontology-building process focus their research on stages 2 (identification of instances within a corpus), 3 (progressing from a set of instances to a conceptual model) and 4 (translation of a conceptual model into a formal ontology) of Figure 1. For example, (Uschold & Güninger 1996) consider the level of abstraction of the first concepts constructed (Stage 2); (Bennett 2005) lists the various criteria for instance classification which can underlie the construction of the concepts (Stage 3); finally, (Guarino 1994 ; 1995, Gruber 1993, Bachimont 2000) establish the rules according to which the concepts of an ontology must be defined in order to overcome the ambiguity of the

---

[1] A "know-how" can be defined as an ability of using knowledge in practice.

[2] We can for example quote Brénot and Tuvée (1996), Grouard and Meston (1993), Joule and Beauvois (2003), Watzlawick (1980), Crozier and Friedberg (1977), and Latour (1993).

[3] Later in this paper, these various degrees of implicitness are combined in the generic term "implicit".

linguistic expressions from which they are extracted (Stage 4). One of the key questions not yet treated by those authors is that of the first expression of the perceived reality (Stage 1).

This paper presents two kinds of results:
– First, it describes a domain ontology composed of about fifty concepts (upper levels) and a hundred relations. This ontology enables to represent tacit change management knowledge into conceptual graphs and to specify a knowledge server. But it has a more general scope as it regards the sharing of tacit knowledge within an organization. More precisely, the principles of OCM could be re-used for the modelling of another set of know-how - particularly for the modelling of organizational how-how - or for the building of another knowledge management application.
– Secondly, this paper presents the methodology which has been used to design the OCM. This methodology is composed of two main stages. The first stage consists in of analyzing the knowledge to be modelled by gathering a "know-how expressions corpus" starting from observations and interviews, then extracting and categorizing "knowledge items" from this corpus. The second stage consists of designing the ontology by translating "knowledge items" into hierarchies of concepts and relations. This process probably could be re-used to define the content of any ontology in a domain where most of the knowledge is tacit and refers to know-how.

What follows is organized into five sections: after having described the analysis of change management knowledge which is the basis of the OCM in Section 2, we present the ontology in Section 3. Then, Section 4 describes the way this ontology is implemented in our application - a server for change management knowledge mapped into conceptual graphs. In the last Section, we discuss the status and the possible re-use of the obtained ontology and the design methodology that has been pursued.

2. Analysis of change management knowledge

Change management, conducted by various actors in the organization (project managers, executives, ergonomists, etc.) relies on paying particular attention to the role of users, that is, to the members of the organization the project is intended for. As they will have to give life to the change, the point of view of the users must be listened to, their influence on the success of the project understood and their adaptation to the new situation accompanied. Therefore, change management is a know-how mainly based on relational abilities, which are mostly tacit and acquired by experience.

How to build an ontology starting from this kind of knowledge which is mainly tacit ? As explained in Section 5, many of the concepts suggested by already existing models are either superfluous, or insufficient. Therefore, we cannot use an existing foundational ontology,[4] and we prefer to design an ontology which is specifically adapted to the modelling of "know-how" type knowledge.

In order to build the upper-level concepts capable of representing "know-how" type knowledge, we have to adopt a mainly "bottom-up" strategy - which starts from gathering and analyzing the knowledge used by practitioners. Indeed, the specific case of building an ontology based on tacit knowledge raises the following issue: no existing discourse elicits the know-how to be modelled nor constitutes a sufficient support for the instance identifying work (stage 2 of Figure 1). Therefore, the first step consists of constituting a corpus of "know-how expressions", that is a corpus capable of expressing the required know-how. For that purpose, a gathering work relied on observations and interviews is necessary to complete the existing literature on the domain. Then, "knowledge items" about domain have to be

[4] "A foundational ontology, sometimes also called 'upper level ontology', defines a range of top-level domain-dependent ontological categories, which form a general foundation for more elaborated domain-specific ontologies" (Guizziardi and Wagner 2005). See for example DOLCE (Masolo et al. 2003), SUMO (Niles and Pease 2001),UFO (Guizziardi andWagner 2005), GOL, BWW or OntoClean/Dolce. For a synthetic presentation of these various ontologies, see (Colomb 2002) and (Gomez-Pérez et al. 2004).

extracted from this corpus. “Knowledge items” are “knowledge instances” or “particular expressions of the knowledge of an individual about change management”. Formally, they are propositions empowered with an informative value for someone else. Finally, the categorization of those “knowledge items” allows identifying the higher concepts of the hierarchy. Figure 2 sums up the different kinds of result required to build an ontology starting from mostly tacit knowledge according to the different elicitation levels involved in the ontology design process represented in Figure 2.

Figure 2. The different levels of results involved in the design process of the OCM and any ontology about tacit knowledge

The next sections detail the methodology which has been precisely used for the construction of the OCM. The stages 1, 2 and 3 of Figure 2 are explained in Section 2. The stage 4 is described in Section 3.1. The different steps of this design methodology are summed up in Figure 8 and its contribution for modelling another set of know-how is discussed in Section 5.4.

2.1. Constitution of a corpus of “know-how expressions”

We gathered instances of change management knowledge from three sources, the last two of which made it possible to reach the tacit dimension of the knowledge to be shared:
– The *literature* in change management, within and outside SNCF.
– *Observations in the field*, from two reorganization projects carried out at SNCF. We have analysed these projects by observing, in a non-intrusive way, formal and informal exchanges between actors implied in change management, and have transcribed the content of various exchanges in each dialogue session observed.
– *Interviews*. Interviews conducted with some of the actors observed enabled us to supplement the first type of data with “first person” data, i.e. data expressed using the personal pronoun “I”. These interviews, which belong to the classic semi-directive or elicitation type interviews,[5] led the actors to describe - and sometimes become aware of - their ways of proceeding and of perceiving the project on which they worked, or to represent their environment to themselves.

These various data allowed us to constitute a corpus of “know-how expressions”, i.e. a corpus of documents, on the one hand, and of verbatim records of discussions and interviews, on the other hand.

2.2. Inventory of knowledge items

The analysis of the “know-how expressions” corpus consisted in extracting a typology from the knowledge used by the actors in their practice.[6] For that purpose, we proceeded as follows:
– We listed, and then categorized the *knowledge items* defined in the introduction of Section 2, as discussed below.
– We identified the main *parameters* according to which those knowledge items are determined. In other words, we identified the elements of the context they fit in. In fact, change management knowledge is held by a *person*, raises or solves a *problem*, and relates to a particular *project*. All of these parameters appear in the obtained ontology, presented in Section 4.

[5] The elicitation interview techniques (Vermersch 2001) help the interviewed person to describe the course of his/her subjective experience very precisely. This kind of interview gives access to pre-conscious cognitive processes activated by a subject when realizing an action, i.e. to the internal strategies he implements unawarely.
[6] A second path of analysis consisted in modelling the process whereby the actors co-construct their choices, that is to say the evolution of their knowledge, on the one hand, and the role of interactions between actors in that evolution, on the other hand. As this last type of analysis does not have a direct effect on the nature of the ontology presented, we will not elaborate upon it here. See (Remillieux 2007).

Three kinds of what we call “knowledge items” can be identified in the “know-how expressions” collected:

– Those which are *stated by actors* in a situation of collaboration (verbatim observations) or of written transmission (documents). An instance of this type can be directly extracted from text. For example, the proposition according to which “the users need to be reassured about the change” has been stated by one of the actors in a working session. But a knowledge item can also be an expression written by the modeller in order to synthesize several statements. For example, the statement “the more the users ask questions at the beginning of the project, the more they will commit themselves to the project” synthesises the statements “at the beginning of another project I managed, the users often inconvenienced us by asking many questions. But actually it was a good sign. Thereafter, they commit themselves a lot. Reciprocally, if the users wouldn’t ask questions, we should worry...”. Among the first three types of knowledge items presented, this is the most explicit, in the sense that it requires the minimum of interpretation from the modeller.

– Those which we suppose to underlie *the actors’ behaviour* (verbatim observations). The strategy in this case is to make assumptions concerning the cognitive source of the observed actors’ behaviour. In other words, it is a question of seeking which skills or capabilities underlie their ways of behaving. For example, the way in which one of the observed actors regularly and accurately reformulated his interlocutor’s statements led us to think he had a good listening capability and so “one of the means used by Mr. X to manage the change is to listen to users”.

– Those which can be *induced from what actors say about their subjective experience* (verbatim interviews). For example, starting from the verbatim extract, “I wanted to tell things the way they are in order to know where we were going”, we could induce the knowledge item “to orient himself at the beginning of his mission, Mr. X needs to have a clear vision of the goal of the project”.

2.3. Categorization of knowledge items

Having listed the various knowledge items, we categorized them, i.e. distributed them into groups according to their likeness. Two basic steps of our categorization can be described:

– The first step was to extract “*observation categories*”, i.e. categories intended to guide our work of gathering knowledge items. These first categories were conceived to answer the question “what to observe, what to question?” The same categories resulting from collecting this data thus also served to guide the task of collection. Many classes found at this first level of analysis turned out not to be relevant for the ontology. For instance, the category “item to be further examined at the next gathering stage” is obviously limited to the observation grid it was conceived for. Nevertheless, some initial groupings of knowledge items were carried out at this stage. For example, the distinction between “mission” type actions, which correspond to the tasks successively carried out by actors, and “transverse know-how” type actions, common to several missions, have resulted from the designing of observation categories.

– Only the second stage was carried out with the explicit aim of classifying the knowledge items. The obtained categories are no longer “observation categories” but “*result categories*”. In other words, the objective of this stage was no longer to guide the knowledge observation but to categorize the collected knowledge in a systematic way. As they are preceded by a first task of instance classification, the categories resulting from this second stage are more general than the first ones. The distinction between knowledge of the world and knowledge about action, which we will reconsider below, is an example of a “result category”.

The partitioning criterion which guided this second categorization stage relates to the *object* of knowledge. In other words, we grouped the knowledge items according to “what they related to”. For example, the knowledge items “to gain support for change management, you have to address yourself to the Management Institute of SNCF” and “the Rolling Stock Department has a very hierarchical organization”, both relate to the company, SNCF. That is why they were grouped into the category “knowledge of SNCF”. This kind of partitioning anticipates the supposed logic of the future ontology’s user: it is a question of organizing the base of knowledge so that it constitutes an answer to the question “what would

I like to have information about?" The nature of the knowledge collected, e.g. theoretical *versus* practical, could have been another partitioning criterion. However this kind of criterion would more result in a classic knowledge base than in an ontology, understood as a description of the objects of the world.[7]

2.4. Results of change management analysis

To avoid redundancies with the presentation of the ontology in Section 3, we will not detail here all the distinctions reached by the change management analysis. We will only comment the distinction which was observed between *knowledge about action* and *knowledge of the world*. Indeed, this distinction is crucial not only because it is the foundation of the whole OCM but, as soon explained, because it concerns any know-how modelling.

Each knowledge item gathered by the change management analysis could be ascribed either to the category of *knowledge about action* or to that of *knowledge of the world*. The former relates to the subjects' action. The idea that "you should provide people who will be affected by change with an illustration of that change as soon as possible in the process" is an example of knowledge about action. On the contrary, knowledge of the world describes the environment the agents act upon, without saying how they must or can act within that environment, just as the proposition that "change users especially fear the first implementation of the change".

The distinction between these two kinds of knowledge appeared clearly and each knowledge item collected could easily be ascribed to one or the other. No doubt this can be explained by the specificity of the studied activity: the work of the change management actors consists in constantly adjusting their actions to an unstable context, adapting their decisions to the properties of an environment which represents both an object to be known and an object to be modelled. However, several factors lead us to surmise that this partitioning of knowledge is not limited to the change management domain. Firstly, we can mention the analyses made by (Schön 1994) about other know-how such as architectural design or psychotherapeutical interviewing. These analyses suggest that all professional practices consist in going back and forth between a situation and its transformation, between taking into account the characteristics of the situation and deciding how to act on it. Secondly, it seems that even at its most tacit levels, know-how consists of a succession of actions and information acquisition, with the latter constituting the most implicit aspect of the process being studied (Vermersch 2001).

Nevertheless, the distinction between knowledge about action and knowledge of the world should not lead to the introduction of a strict separation between these two kinds of knowledge. In fact, action and information acquisition are closely related in practice. We refer here to the enaction theory, according to which knowledge does not consist of representations of a world that is independent of us or "given", but is rather deeply determined and structured by our actions. In other words, knowledge and action are co-constituted: "cognition is not the representation of a pregiven world by a pregiven mind but is rather the enactment of a world and a mind on the basis of a history of the variety of actions that a being in the world performs."(Varela et al. 1991) At the same time that "knowledge about action" and "knowledge of the world" are closely related, "action" and "world" are notions both relative to each other. In this sense, the distinction between "knowledge about action" and "knowledge of the world" could also be understood as a distinction between a "plan" and the "target domain of the plan" but not as a distinction between an action in general and a world without action.

However, we deliberately exaggerate the distinction between the two categories of knowledge observed, in order to clarify their links. Indeed, our assumption is that, once visible, these links can be more easily

---

[7] In the case at hand, of the objects of the world *such as they are perceived* by change management actors. For more details about the status of the ontology, see Section 5.3.

enriched, diversified, or even criticized. For instance, the user will be able to evaluate precisely the appropriateness of an action one might propose to him in relation to a given information: first, does he agree with the initial information? (For example, with the idea that "living a change triggers fear of the unknown"). Secondly, does he agree with the suggested answer? (For example, with the recommendation "to reassure those who undergo the changes"). The user will also be able to invent another action to carry out in response to the initial information (for example, "to use the fear as a driver for change"). In other words, the analytical and fragmented representation of knowledge solicits the subject's independence of judgment, criticism and creativity.

The previous section has presented the method by which a classification of change management knowledge can be built, and some of the higher categories obtained. Starting from this classification, still strictly descriptive at this stage, we were able to design the OCM.

3. Presentation of the OCM

3.1. Design method

How can a descriptive model of change management knowledge evolve to a model which determines the structure of the future knowledge server and guides the representation of the change management knowledge? In other words, how can the observed categories be translated into a network of concepts and relations which allow the expression of the type of knowledge which is of interest to us in conceptual graphs? The OCM was in fact designed according to the conceptual graph paradigm (Sowa 1984), which explains why its meta-categories are "concept" and "relation".[8]

The first stage was to adapt, complement and specify the knowledge typology in order to translate it into a *hierarchy of concepts*. To create this hierarchy in detail, it was necessary to:
– *List* the categories resulting from our analysis of change management knowledge and *translate* them into concepts. As we saw in Section 2, two types of content were listed and organized in a hierarchy: knowledge items and their parameters. The parameters (actor, project, etc.) could be directly integrated in the hierarchy of concepts. Knowledge items' categories, on the other hand, first had to be translated into *objects* of knowledge, in order for the ontology to divide the world in such a way that it is perceived by change management actors. In other words, knowledge items' categories have been translated into everything that change management knowledge may have a bearing on. This was done without difficulty since, let us recall it, the knowledge partitioning criterion related to the items' object. Roughly speaking, each kind of knowledge thus has the form "knowledge about X "; and from this, we simply had to translate "knowledge about X" into "X". For example, the category "knowledge about the world" became the concept [element of the world], the category "knowledge about the culture" became the concept [culture] and so on.
– *Organize* the listed concepts in a hierarchy.
– *Complete* the hierarchy so obtained with missing subtypes of each concept and with abstract concepts having the capacity of regrouping the concepts which could be specified by the same relations. For example, the concept [perdurant] have been created to aggregate the concepts [fact] and [action] because of their common capacity to have an (effect). Being connected with the question of relations, this step was carried out in parallel to the one, described later in this article, which consisted in building the hierarchy of relations.

[8] We chose conceptual graph formalism for its flexibility (no predetermined concepts and relations) and its great expressivity (see Section 3.4).

In order to build this hierarchy, we sought to reach the quality criteria of an ontology which have been defined by (Gruber 1993) and (Gomez-Pérez et al. 2004).[9] Among these criteria, the following two points were particularly useful for structuring our conceptual modelling activity:
– *Completeness* (Gomez-Pérez et al. 2004). According to this criterion, the hierarchy must constitute sufficient material for the knowledge representation of the domain. Concretely, this meant that each listed knowledge item could be formalized in the ontology.
– *Reduction of redundancy* (Gomez-Pérez et al. 2004). More specifically, our objective was that each knowledge item can easily be attached to only one category.

Once the hierarchy of concepts has been built, at least partially, it is the time to design the *hierarchy of relationships* or *relations*. This second hierarchy makes it possible to connect concepts in order to express knowledge in the form of conceptual graphs.
For this, we began by listing the relationships we needed in order to represent the collected change management knowledge items, using the concepts previously identified. The first step was thus "bottom-up" and iterative (going back and forth between designing and using the model). We interrupted this process when the new relations that could be added became rare.
The second step, "top-down", consisted in making an inventory of the possible relations between each pair of higher concepts. Once this inventory was made, it was necessary to specify and check the validity of the inherited relations at the level of lower concepts. For example, the relation (compatible stage), defined in order to connect two concepts [perdurant], is not relevant for the sub-type [general principle] of [perdurant] since a [general principle] does not form part of a process.[10]
In addition, we completed the hierarchy of relations listing missing relevant subtypes of each high-level relation, on one hand, and seeking what could be said about each concept considered in itself which was not held by already defined relations, on the other hand. This last stage involved creating not only new relations but also a new concept for each value of the found property. For instance, to state that a [project] is characterized by a certain degree of change, we have to construct both the relation (degree of change) and the concept [value of degree of change] with the instances [evolution] and [revolution]. All the concepts such as [value of degree of change] are grouped together under the concept [quality] and all the relations such as (degree of change) are aggregated under the relation (property).[11]
Finally, we finalised the obtained hierarchies implementing them and harmonizing them as effectively as possible with existing models. This last stage is described more precisely in Section 5.2.

To provide a clearer picture of the ontology, we show the hierarchies of concepts and relations in separate figures.

3.2. Hierarchy of concepts

In this section, we present the hierarchy of the OCM's concepts (Figure 3). Change management can have a bearing on various objects. OCM's concepts represent those various objects. Only higher levels of this hierarchy could probably be applied to another set of know-how.[12] However, for the sake of clarity, Figure 3 also presents some middle level concepts and examples of instances.

Each concept is defined by a *textual description*, an inventory of its *sub-types and instances* (ontological commitment),[13] and finally, the listing of its possible *relations*[14] with other concepts. The listing of a

---

[9] The criteria of (Gruber 1993) are: clarity, coherence, extendibility, minimal encoding bias, minimal ontological commitment.

[10] These exceptions are not mentioned in this article.
[11] Sowa defines a property as "*what cannot exist without some substrat*" (Sowa 2000).
[12] It should be noted that application of OCM to another set of know-how has not been tested.
[13] Bachimont (2000) distinguishes between the "semantic commitment", which consists in defining the meaning of each concept by four differential principles (based on what identifies and differentiates it from its father and what identifies and differentiates it

concept's relations can be found by seeking the signatures of relations this concept appears in (see Figure 13).We do not systematically clarify the differential principles which define a concept by identity and difference relative to its close neighbours (semantic commitment). These principles are implicitly contained in the very mention of its relations.

To complete Figure 3, we propose to define some of the concepts in the hierarchy whose meaning is the least obvious. We only mention the most specific relations which can be mentioned about a concept. Each time we use a term from an existing model, we describe it. Section 5.2 more generally comments on similarities and differences between the presented ontology and others.

First level concept
Object of knowledge: Anything that a knowledge item which is included in the modelled know-how can bear on. It is the universal concept, i.e. the most general concept of the hierarchy, which any other concept of the hierarchy is a type of.
Relations. Any [object of knowledge],[15] except [text] and [quality],[16] can have a [quality] as (property), a [perdurant] as (phase of use), a [project] as (related project), an [agentive social object] as (related actor), a [resource] as (resource), a [problem] as (associated problem) or (solved problem), another [object of knowledge] as (proximate concept) or (equivalent concept), a [proposition] as (required knowledge), and a [text] as (clarification).

Second level concepts
Element of the world: Any component of the subject's environment, i.e. with respect to which the subject positions himself as a "knowing subject". Note that through the concepts [action] and [element of the world], we find again the fundamental distinction between knowledge about action and knowledge of the world established by our fieldwork.
Relations. An [element of the world] can have in addition to the relations inherited from its father [object of knowledge]: a [fact] as (related fact).
Perdurant: Everything that "occurs", unfolds in time. In DOLCE's terms, perdurants are "*entities that happen in time*" (Masolo et al. 2003). In SUMO or ArGon, this concept corresponds to "process" (Niles and Pease 2001, Echtler 2006). When a [perdurant] is started by a change manager, it is an [action], otherwise, it is a [fact].
Relations. A [perdurant] can have in addition to the relations inherited from its father [object of knowledge]: an [experience's narration] as (illustration by a narration), another [perdurant] as (next stage), (previous stage), (sequential stage) or (parallel stage).
Proposition: We borrow this concept from the SUMO ontology, where it is defined as "*semantic or informational content*" (Niles and Pease 2001).
Relations: A [proposition] can have in addition to the relations inherited from its father [object of knowledge]: an [object of knowledge] as (usage) and a [person] as (author).
Text: Textual description which allows commenting a concept.
Relations. A [text] can be a (clarification) for an [object of knowledge] or a (definition) for a [notion].
Project: General framework which surrounds the practitioner's mission. This framework is either a simple political will of change or a plan supported by a project team. As specified by the below relations, this concept is only appropriate for a know-how closely related to change management. We suggest re-defining its relations according to the specific domain to be modelled.

---

from its neighbours) of the "ontological commitment", which determines a concept by the extension of the objects which refer to it.
[14] The relations, defined in 4.3, are organized in a hierarchy. We do not define the relations of kind (property) in this paper.
[15] Concepts are written [concept] and relations are written (relation).
[16] A [text] can only be related to an [object of knowledge] by the relation (clarification). It is indirectly related to the concepts which are linked to the concept it comments. A [quality] can only be related to an [object of knowledge] by a relation (property). It defines a specific [concept] with some specific [properties]. Those two concepts are defined later.

Relations. A [project] can have in addition to the relations inherited from its father [object of knowledge]: an [aspect of organization] as (impacted aspect of the organization), an [organization] as (changed field), or (initiator), a [group] as (impacted population), a [person] as (actor having worked on the project), a [possible case] as (situation aimed at), an [action] or a [chain] as (method), an [object of knowledge] as (related content), a [document] as (built document), an [experience's narration] as (illustration by a narration) and a few [project's qualities], that we do not detail in this article, as (project's properties).[17]

Problem: A situation which is not satisfying for the practitioner and requires a solution. We have discovered that instances of the concepts of the type [problem] could be described by a graph always having the same type of structure. They are thus "contexts", that is concepts whose instance is defined by a "nested graph", or "designator" according to Sowa's terms (Sowa 1984). More specifically, four kinds of structure have been identified and categorized into two types of problems: [dilemma] and [difficulty]. We describe these concepts in a few lines. Representing problems with nested graphs allows users to solve a [problem] not only by consulting the [action] linked to this problem by the relation (solution) but also by consulting the information which concern the concepts its nested graph is built from.

Relations. A [problem] can have in addition to the relations inherited from its father [object of knowledge]: an [object of knowledge] as (arising context) and an [action] as (solution).

Quality: Possible value of a (property). Each [quality] is related to a specific [object of knowledge] with a specific (property). For example, a [degree of change: revolution] is related to a [project] with a (degree of change). No inverse relation of (property) exists because a [quality] does not constitute a relevant starting point for browsing within the graph base. Indeed, a [quality]'s function is to inform about another concept but not to be characterized itself.

Third level concepts

Agentive social object: Human or organizational entity. We borrow a notion defined in DOLCE as a "*non-physical object which is generically dependent a community of agents*" and "*to which we ascribe intentions, beliefs, and desires*" (Masolo et al. 2003).

Relations: An [agentive social object] can have, in addition to the relations inherited from its father [element of the world]: an [object of knowledge] as (related knowledge) or a quality [SNCF's affiliation: yes/no] as (SNCF's affiliation).

Fact: State of things supposed to be real, whether it is verified or not. A [fact] can be a [possible case] in which case it is close to the usual definition of «fact» as «situated event having actually occurred». But it can also be a [general principle], that is a general law *supposed to* describe the world such as it is.
Relations. A [fact] can have, in addition to the relations inherited from its fathers [element of the world] and [perdurant]: a [perdurant] as (cause), another [fact] as (sign) or (signification), an [action] as (recommended action), an [element of the world] as (related element), and a property [kind of fact] as (kind of fact).

Assertion: Proposition stating the existence of a state of things which can be opposed to another one, i.e. opened to argument. Formally, assertions are either a [general principle], or a «context» whose nested graph has five possible forms (see Figure 3 for examples). An [assertion] can be: a [causality] if its nested graph has the form [[action] ↔ (consequence/trigger) ↔ [fact]], a [chain] if its nested graph has the form [[action] ↔ (compatible stage) and/or (other possible mean) ↔ [action]], an [interpretation] if its nested graph has the form [[fact] ↔ (significance/ sign) ↔ [fact]], a [modality] if its nested graph has the form [[action] ↔ (means/goal) ↔ [fact]] or a [solving] if its nested graph has the form [[problem] ↔ (solved problem/solution) ↔ [action]].

Relations. An [assertion] can have, in addition to the relations inherited from its father [proposition]: another [assertion] as (opposition) and a [person] as (modeller).

Unspecified context: An [unspecified context] is a proposition which has the form of a context whose nested graph is unspecified. This concept enables to state things about every graph, especially to mention its author and the environment it is relative to (type of project, phase of use etc.).

[17] The relation (degree of change) is for example a (project's property).

Relations: An [unspecified context] can have, in addition to the relations inherited from its father [proposition]: a [person] as (modeller).
Difficulty: An obstacle faced by an actor carrying out an action. A [difficulty] can be either a [difficult fact:[fact]], a [difficult action: [action]], or the accomplishment of two difficult-to-conciliate actions. This latter case, which can be represented with the standard graph: [Difficulty: [action] ↔ (and) ↔ [action]], is common in change management practice. Indeed, actors are frequently in the situation of seeking "the right balance" between two opposite actions.
Dilemma: A difficult choice between two possible actions. Dilemmas can be described by a graph of the type [dilemma: [action] ↔ (conflicting) ↔ [action]].
Example of instance: take the issue which consists in wondering whether, to communicate to the users, it is better to communicate about changeable information or not; this will be represented by the graph: [dilemma: [action: to communicate about changeable information] ↔ (conflicting) ↔ [action: not to communicate about changeable information]].

3.3. Hierarchy of relations

To define concepts, we have made reference to the relations which could be established between them. Nevertheless, we have not yet systematically defined these relations. Their organization in a hierarchy is shown in Figure 4. Before defining a few of them by way of example, some remarks are in order:
A relation has as its "*signature*" the concepts which it connects. For example, the relation (role) has the signature {person, action}, which means that a concept of the [person] type has a concept of the [action] type as (role). In other words, the signature given as an example allows the graph [person: Mr. X] → (role) → [mission: to manage a project], which will be read as follows "Mr. X has to manage a project as role". The signature of a relation is made up of concepts equivalent to or lower than those which constitute the signature of its father.

Figure 3. Hierarchy of OCM's concepts (high and middle levels) (extract)

Each relation has an *inverse relation*,[18] given that two inverse relations connect the same concepts but each in a different direction.[19] For example, the inverse of the relation (agent): {action, person} is the relation (role): {person, action}.[20] We indicate two inverse relations as follows: (agent)/(role).

Below are some examples of relation definitions. Each of them includes a *signature*, an *inverse relation* and a *textual definition*:
Significance {fact, fact} / Sign: A fact can mean another fact, i.e. signify that this other fact occurred. The relation (significance), by giving the possibility to the users of *interpreting* their environment, calls precisely for the "acquisition of information" component of their know-how (see Section 2.3). For example, [fact: the members of a participative working group have an aggressive behaviour] will have as possible (significance): [fact: there are conflicts within the group].

Figure 4. Hierarchy of OCM's relations (extract)

[18] Unless it is symmetric as the relation (compatible stage), for example.
[19] In Figure 4, inverse relations are arranged the one below the other one, within the same node. Only the signature of the first one is indicated.
[20] The details of the inverse of each relation makes it possible to read any graph starting from one or the other of the connected concepts. For example, the relation (person having worked on the project), which connects a [person] to a [project], having the relation (managed change) as its inverse, the information [project: p] → (person having worked on the project) → [person: Mr x] (read as "a person having worked on the project p is Mr x") could also be expressed by [person: Mr x] → (managed change) → [project: p] (read as "Mr x has managed the project p"). Thus, the same information that we have just expressed in two different ways will be able to appear in the future application whether the concept we are interested in is [project: p] or [person: Mr x].

The following relations allow the establishing of links between *knowledge about action* and *knowledge on the world*:
Possible action {possible case, action} / Condition: It is a question of specifying which preliminary situation conditions the accomplishment of an action.
Recommended action {fact, action} / Circumstance (if the target [fact] is of the type [possible case]), or Reason (if the target [fact] is of the type [general principle]): For an [action], "to be recommended given a fact" means "to have to be carried out in case the target fact actually occurs".
Required knowledge {object of knowledge, proposition} / Use: Reality to be observed or taken into account to understand an object of knowledge, notably to understand how to carry out an action.

3.4. Graph example

Figure 5 shows the kind of conceptual graph which can be constructed from the OCM. Such graph is validated by the knowledge server's users themselves. Indeed, the interface of the future application's prototype provides the user with the possibility to model his own knowledge, therefore to complete or criticise the graph base.[21]

Figure 5. Example of a conceptual graph based of the OCM

The representation of change management knowledge - or other mostly tacit knowledge – through conceptual graphs offers two kinds of advantages. First, it is a good means to obtain a highly capable knowledge server. Indeed, on top of the kinds of operation allowed by any database (to search the different arcs linked to some concept within the graph base, to browse within the concepts' hierarchy, to identify some concept's instances, etc.), conceptual graph formalisms (Sowa 1984), (Sowa 2000) are a means for:
– considering modelled knowledge as being a "context", i.e. a concept whose the referent is a nested graph and which can itself be specified by relations with others concepts.[22] Such capacity provides conceptual graphs with a great expressiveness.
– executing some generalization operations on data. This capacity could enable to identify recurrent data starting from particular graphs of the base.[23]

Secondly, a conceptual graph provides a visual representation of the knowledge to be shared. Such representation facilitates the construction of a clear and general view of information by the future knowledge server's user. However, the interface of the application prototype, which is presented in Section 4, does not use this kind of representation yet.

4. Prospects for application

The OCM was designed to structure a knowledge server in the domain of change management. A prototype of this server is in the pipeline. Before presenting an example of this prototype's interface, we will illustrate the evolution from the ontology to the server by a schematic diagram of the interface (Figure 6). In this example, the tool informs the user about the possible case [the members of a participative group are aggressive] by a list of headings ("signification" "sign" "recommended action"). It is important to note that the headings correspond to *relations* of the ontology and their contents to its *concepts*.

In other words, the interface represented in Figure 6 is based on the following conceptual graphs:

[21] See the input "Share my knowledge" in Figure 7.
[22] See also the definitions of "Assertion" and "Unspecified context" in Section 3.2.
[23] See (Remillieux et al. 2007), p.192.

- [possible case: the members of a participative group are aggressive] → (signification) → [possible case: there are conflicts inside the group];
- [possible case: the members of a participative group are aggressive] → (sign) → [possible case: criticisms are systematic and closer to each others];
- [possible case: the members of a participative group are aggressive] → (recommended action) → [action: to write down their notices on the paper-board].

Figure 6. From the ontology to the interface

As a result, you have a system in which the user can indefinitely "zoom in", on each content, as both the answer elements and the query elements can be described in terms of the ontological structure to which they all belong to. In other words, the result is a system in which it is possible to break up each item of information into increasingly detailed units.

This kind of system is particularly relevant for sharing tacit knowledge. For example, thanks to Vermersch's elicitation techniques (Vermersch 2001), it is possible to describe very precisely the stages which compose an action. Thus, we know that in order to write down participants' notices, you need to understand what they want to say; that in order to understand what the participants want to say; you have to observe and read their movements etc. Our system permits the user to access these increasingly specific descriptions by zooming successively on each of the proposed stages.

To illustrate Figure 6, Figure 7 provides a more realistic picture of the future application's interface which will allow the user not only to *collect* some information but also to *share* his own knowledge and experiences.

Figure 7. Interface of the change management knowledge server prototype

5. Discussion

This paper presents two kinds of contributions: the OCM itself and the methodology used for its design. In this Section, we detail the originality and the scope of each of those contributions. Sections 5.1, 5.2 and 5.3 regard the first contribution; Section 5.4 deals with the second.

5.1. Comparison with existing conceptual modelling approaches

The originality of the OCM firstly lies in its purpose - which is to allow the sharing of change management know-how through the medium of a knowledge server. Then OCM differs from other similar models like those which are built in design research and goal-oriented or argumentation modelling.

Change management may be considered as a practice consisting in solving design problems, that is to say problems which can be solved in many different ways. In this sense, our question is related to design research. As shown by (Dorst 2003), this field of research includes two ways of understanding what a design problem is. On the one hand, according to the "Rational paradigm" (Newell and Simon 1972), a design problem deals with a rational activity which can be supported by a design rational methodology. Goal-oriented modelling approaches belong to this paradigm (Lamsweerde and Willemet 1998, Lamsweerde 2001, Nurcan and Rolland 2003, Kavakli and Loucopoulos 2006). By eliciting and modelling goals and sub-goals which have to be reached in order to solve a problem, these approaches contribute to the design process. In this perspective, we have to mention a line of research on organizational change modelling realized in order to facilitate the decision making by the stakeholders of an organization (Nurcan and Rolland 2003, Kavakli and Loucopoulos 2006). However, those works differ from ours in two main points. First, whereas we focus on how to take into account human factors in change management in order to make easier the psychological process of change, they are mainly interested in change engineering at a strategic level. Secondly, in a more general way, the objective of goal oriented research - just like for argumentation models (Verheij 2005, Echtler 2006) - is to identify requirements and *specify systems* rather than to *share knowledge*.[24] In other words, our approach is more descriptive insofar as we prefer to focus on how the change managers *proceed* instead of focusing on what they *want to do*. Paying attention to the subjective process of change management is linked to the second paradigm in design research identified by Dorst (2003).

On the other hand, the "Reflective Practice paradigm" proposed by Schön (1994) insists on the tacit dimension of the design process: practitioner's knowledge cannot be reduced to a determined list of possible solutions related to a given problem. It's a "knowing in action", which cannot be easily shared. Simon (1973) himself recognized the imprecise nature of certain problems. Those "ill-structured" problems cannot be completely described. In the course of instigating changes, actors regularly meet a number of such relational and imprecisely defined problems. The OCM aims to explicit and make 'shareable' the know-how which enables these problems to be solved.

5.2. Comparison with existing models

The original purpose of the OCM, previously defined in Section 5.1, involves an original content - as the comparison with other models has confirmed.

As explained in Section 2, we built the OCM according to a bottom-up strategy, without reusing any existing ontology. Nevertheless, once the ontology was constructed, we have compared it with others, notably for wider interoperability. More specifically, we have taken into account the three following fields of research (see previous Section for the two first fields): goal-oriented modelling, argumentation based negotiation and foundational ontologies (DOLCE,[25] SUMO,[26] KR Ontology,[27] etc.[28]). First, some notions of the OCM obviously overlap with other models, like that of "goal" used in the goal oriented modelling. Sometimes, we changed our first choices to harmonize our terms with others. For example, "event" became "perdurant"(DOLCE) and "attribute" became "property" (KR Ontology).

However, three kinds of differences between OCM and the existing models confirmed how difficult it would be to re-use one of them to obtain an ontology which was both sufficient and relevant for our goal. First, many distinctions were not useful for our objective. A high-level example is the one which is often made between abstract and physical objects (SUMO, KR Ontology). Based on the criterion of the location in space, this distinction is irrelevant for the OCM, which does not need to characterize things according to by their location. On the contrary, we have not found in the literature distinctions like the one we have observed between dilemma and different kinds of difficulties to explain what a [problem] is. Third, we have often found in other models notions which, although similar, were not exactly equivalent to ours. For instance, the concept [occurent] (KR Ontology) is not equivalent with [fact] although it is close to it. Indeed, defined by its possible decomposition in stages, this concept cannot be the supertype of [general principle].

More generally, the ontology designed is based on a specific logic which explains these differences: more than distinguishing concepts according to the intrinsic qualities of what they refer to (like the capacity to be located or not), it differentiates among them according to their meaning for an actor's point of view (like the fact of being started by the actor or not for a given perdurant). In the same way, more than gathering together relations according to their general nature of "prehended" or "prehending entity" for

[24] Even if one of its benefits is to facilitate communication and negotiation between stakeholders.
[25] (Masolo et al. 2003)
[26] (Niles and Pease 2001)
[27] (Sowa 2000)
[28] (Gomez-Pérez et al. 2004, Colomb 2002)

example (Sowa 2000), we grouped relations according to the kind of information they bring to the user of the knowledge server (description, environment, consequence etc.).

5.3. The status and application range of the OCM

According to the *foundational* vs. *domain* ontology typology, the OCM is a domain ontology that is an ontology designed for a specific knowledge domain. Indeed, this ontology has been built for the domain of change management and a particular sort of application (knowledge server). In this sense, its relevance is limited to the change management task. Furthermore, the OCM is strictly speaking an *epistemology* rather than an *ontology* insofar as it concerns the change management's actors ways of knowing. In other words, it deals with the world as it is perceived by subjects in a specific context instead of the world in itself.

However, even if it is built from a particular domain, most of OCM's high levels are nonetheless as generic as those of any foundational ontology. In addition, even if the world it concerns is not an absolute world, it is a world anyway. In this sense, the OCM could be relevant for another set of know-how, notably those related to organization management like project management, risk management or strategy. Some initial elements confirm this idea. Indeed, as explained in Section 2.4, the structure of change management know-how, which results from our analysis and on which the OCM is based, shares characteristics with that of any know-how. For example, the importance of establishing permanent interactions between decision actions and world knowledge can be just as easily observed in change management as in psychotherapeutical interviewing described by Schön (1994). Furthermore, the OCM could be re-used for other applications related to knowledge management. Indeed, as explained in Section 5.1, the OCM's originality also rests with its purpose: to allow the sharing of mostly tacit knowledge through the medium of a knowledge server.

More precisely, we consider that every concept presented in 3.2 could be relevant for another set of know-how, except [project], [project quality], [project phase], [group], [aspect of organization], [organization] and their subtypes - which specifically regard professional know-how related to organizational management. However we cannot be certain of OCM's domain of application before having used it to model another set of know-how than change management.

5.4. Remarks on the design methodology used

This Section has two distinct goals. First, it analyses the design methodology which has been used for the case-study presented in this article. Secondly, it considers what of this methodology can be re-used for other expertise based-domains where knowledge is tacit .

As explained in Sections 3.1 and 3.2 and summed up in Figure 8, the design methodology of the OCM has two main characteristics: first, starting from operational work, it does not use existing generic ontologies; second, it contains many iterative stages, especially between bottom-up and top-down approaches. This method presented both advantage and disadvantage. On the one hand, as shown in the previous Section, it made it possible to build a particularly exhaustive ontology, adapted to its goal - sharing of the change management know-how. On the other hand, it required an important amount of work, notably of verification such as specifying and checking the validity of the inherited relations, and of comparison with other models. We do not have any proof that adapting a generic ontology to our bottom-up analysis would not be more efficient than designing it from scratch.

Figure 8. The methodology used to build the OCM

Nevertheless, this work has to be regarded as a first attempt to building an ontology starting from tacit knowledge which could be both re-used and improved. More precisely, this methodology could be re-used in two ways. The first way consists in fully applying it in order to define every level of a new ontology about another expertise-based domain where knowledge is tacit . The second way assumes that OCM's higher levels can be applied in another set of know-how and restricts the re-using of the design methodology for building the new ontology's low levels. In this second case, we suggest to follow a simplified methodology, composed with the following stages, (most of those stages are represented in Figure 8 where they are framed by a bold surround):

1. Constitution of a "know-how expressions corpus". This stage and the following one are required for any mostly tacit knowledge modelling. (See Section 2.1)
2. Inventory of "knowledge items".
3. Check of the OCM's capacity to express the previously identified "knowledge items". Rather than using "knowledge items" in order to categorize and translate them into concepts - as we did for designing a whole ontology - we suggest here to use them in order to verify the suitability of the OCM for the domain in question.
4. If the preceding check shows that the OCM cannot express every gathered "knowledge items", completion of the ontology.
5. Completion of the concept's hierarchy according to the domain to be modelled with a top-down strategy.

6. Conclusion

We have presented the Ontology for Change Management, constructed in order to allow SNCF actors to share their knowledge in change management. We have created its higher concepts rather than borrowing them from an other ontology, because our problems are quite specific and cannot be solved by an existing foundational ontology.

The OCM formalizes knowledge related to a practice, i.e. to knowledge which is mostly tacit and embedded in actors' practice. Our aim was to design an ontology perfectly adequate to model such knowledge and we tried to do that by designing its higher levels starting from the study of knowledge related to a particular know-how, that of change management. Our study involved case study work, consisting in observations and interviews, necessary to reach the tacit dimension of that knowledge. We thus developed abstract categories constructed *a posteriori* from the domain knowledge, as opposed to most foundational ontologies which are, on the contrary, built starting from an *a priori*, i.e. purely intellectual, approach.
Another specificity of the OCM lies in its objective: design a knowledge server for the change management domain which supports the sharing of the tacit dimension of knowledge. Even if today, we still have to work on the terminology of the ontology in order to translate it into an interface which is perfectly adapted to the user, the application this ontology is intended for has influenced its whole design. This is why the way in which the OCM represents reality intrinsically depends on the task it is dedicated to: facilitating the elicitation and the transmission of know-how.

The nature of our contribution follows from the previously described specificities which characterize OCM. The paper's first result is the OCM itself as an ontology that refers to change management knowledge. Furthermore, although the OCM was designed for a particular knowledge domain, it could probably be re-used for sharing other types of know-how, in particular those which are related to the organizational management sciences - insofar as it includes higher-level concepts which are not specific to change management. Secondly, the methodology used for the OCM design provides a first attempt to build an ontology that relates to tacit knowledge.

In conclusion, even if our knowledge server has been built *for* and *starts from* knowledge managers, we must be careful about its actual use when it will be put in operation. As it is the case of every knowledge server, its success depends on many parameters which are not directly linked to the quality of the system.

First, we have to mention that many organizational obstacles can stand in the way of its real use (Prax 2003). These obstacles, such as intra-organizational conflicts or users' reluctance to share their knowledge, raise problems which transcend our domain. Another area of uncertainty, related to the problem of knowledge appropriation, is particularly important for the tacit nature of the knowledge we expect to share: once the knowledge is elicited from subjects and modelled, how can we ensure that it will be assimilated by others, i.e. become again actual knowledge which is usable in practice?

7. Acknowledgments

We thank every SNCF's change manager we have observed or interviewed, for having contributed to our work, Mondeca and KnowledgeConsult companies for having collaborated on the prototype's buildind and Claire Goosens for having designed its interface.

References

Bachimont, B. (2000). Ingénierie des connaissances, évolutions récentes et nouveaux défis. In J. Charlet, M. Zacklad, G. Kassel, and D. Bourigault (Eds.), *Engagement sémantique et engagement ontologique : conception et réalisation d'ontologies en ingénierie des connaissances*, pp. 305–323. Eyrolles.

Baneyx, A. and J. Charlet (2005). Construction d'ontologies médicales à partir de textes : propositions méthodologiques. In *Actes de la conférence IC 2005 - 16ème journées francophones d'Ingénierie des Connaissances - Nice*, pp. 37–47.

Bennett, B. (2005). Modes of concept definition and varieties of vagueness. *Applied Ontology 1*(1), 17–26.

Brénot, J. and L. Tuvée (1996). *Le changement dans les organisations*. PUF.

Colomb, R. M. (2002). Formal versus material ontologies for information systems interoperation in the semantic web. Technical report, National Research Council - Institute of Biomedical Engineering - ISIB-CNR.

Crozier, M. and E. Friedberg (1977). *L'acteur et le système - Les contraintes de l'action collective*. Paris: Editions du Seuil.

Dorst, K. (2003). The problem of design problems. In E. Edmonds and N. Cross (Eds.), *Expertise in design, design thinking research symposium 6*, Sydney, Australia. Creativity and Cognition Studios Press.

Echtler, F. (2006). Using semantic web languages in argumentation models. Technical report, AI/Cognition group/Technical University of Munich.

Fernandez-Lopez, M., A. Gomez-Pérez, and J. Pazos-Sierra (1999). Building a chemical ontology using methontology and the ontology design environment. In *IEEE Intelligent Systems*, pp. 37–46.

Gomez-Pérez, A., O. Corcho, and M. Fernandez-Lopez (2004). *Ontological Engineering - With examples from the areas of Knoweldege Management, e-Commerce and the Semantic Web*. Springer-Verlag.

Grouard, B. and F. Meston (1993). *L'entreprise en mouvement, conduire et réussir le changement*. Dunod.

Gruber, T. R. (1993). Towards principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli (Eds.), *Formal Ontology in Conceptual Analysis and Knowledge Representation*. Kluwer Academic Publishers.

Guarino, N. (1994). The ontological level. In B. S. R. Casati and G. White (Eds.), *Philosophy and the Cognitive Sciences*, pp. 443–456. Vienna : Hömder-Pichler-Tempsky.

Guarino, N. (1995). Formal ontology, conceptual analysis and knowledge representation. *International Journal of Human and Computer Studies 43*(5/6), 625–640.

Guizziardi, G. and G.Wagner (2005). Towards ontological foundations for agent modeling concepts using ufo. In *Agent-Oriented Information Systems II*, Volume 3508, pp. 110–124. Springer Berlin/Heidelberg.

Joule and Beauvois (2003). *La soumission librement consentie. Comment amener les gens à faire librement ce qu'ils doivent faire ?* Paris: PUF.

Kavakli, E. and P. Loucopoulos (2006). Experiences with goal-oriented modeling organizational change. *IEEE Transactions on Systems, Man, and Cybernetics. Part C : Applications and Reviews 36*, 221–235.

Lamsweerde, A. V. (2001). Goal-oriented requirements engineering: a guided tour. In *5th IEEE International Symposium Requirements Engineering*, pp. 249–262.

Lamsweerde, A. V. and L. Willemet (1998). Inferring declarative requirements specifications from operational scenarios. *IEEE Transactions on Software Engineering 24*(12), 1089–1114.

Latour, B. (1993). *Aramis ou l'amour des techniques*. Editions la découverte.

Masolo, C., S. Borgo, A. Gangemi, N. Guarino, and A. Olramari (2003). Wonderweb deliverable d18. Deliverable Version 1.0, Laboratory For Applied Ontology - ISTC-CNR.

Newell, A. and H. Simon (1972). *Human Problem Solving*. Prentice-Hall. Quoted by Dorst (2003).

Niles, I. and A. Pease (2001). Towards a standard upper ontology. In *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems*.

Nurcan, S. and C. Rolland (2003). A multimethod for defining the organizational change. *Inform. Softw. Technol. 25*(2), 61–82.

Polanyi, M. (2002). *Personal Knowledge*. Routledge.

Prax, J.-Y. (2003). *Le manuel du knowledge management*. Paris: Dunod.

Remillieux, A. (2007). Modélisation et formalisation des connaissances en matière de conduite du changement à la sncf, l'analyse cognitive au service de la conception d'un outil de partage des connaissances. In RESEO (Ed.), *Changement technique, changement social*, pp. 77–90. Paris: L'Harmattan, Collection Dossiers Sciences Humaines et Sociales.

Remillieux, A., C. Petitmengin, J.-L. Ermine, and C. Blatter (2007). Construction d'une ontologie pour le partage des connaissances implicites de conduite du changement (OCM). In C. de Publication Universitaire (Ed.), *Les ontologies, mythes, réalité et perspectives,JFO 2007*, pp. 175–194.

Schön, A. (1994). *Le praticien réflexif - A la recherche du savoir caché dans l'agir professionnel*. Les éditions logiques.

Simon, H. (1973). The structure of ill-structured problems. *Artificial Intelligence 4*, 108–201. Quoted by Dorst (2003).

Sowa, J. (1984). *Conceptual Structures : Information Processing in Mind and Machine*. Addison-Wesley.

Sowa, J. F. (2000). *Knowledge Representation: logical, philosophical, and computational Foundations.* Brooks Cole Publishing Co., Pacific Grove, CA.

Uschold, M. and M. Güninger (1996). Ontologies: Principles, methods, and applications. *Knowledge Engineering Review 11*(2), 93–155. Quoted by (Gomez-Pérez and al. 2004).

Varela, F. J., E. Thompson, and E. Rosch (1991). *The embodied Man*. MIT Press.

Verheij, B. (2005). An argumentation core-ontology. In *Agentlink technical Forum Group*. Quoted by Echtler (2006).

Vermersch, P. (2001). *L'entretien d'explicitation*. Paris: ESF.

Watzlawick, P. (1980). *Le langage du changement, éléments de communication thérapeutique*. Editions du Seuil.

Zweigenbaum, P. and C. Menelas (1995). Menelas : Coding and information retrieval from natural language patient discharge summaries. In M.-F.Laires, M. Ladeira, and J. Christensen (Eds.), *Advances in Health Telematics*, pp. 82–89. Ios Press, Amsterdam. Quoted by Bachimont (2000).

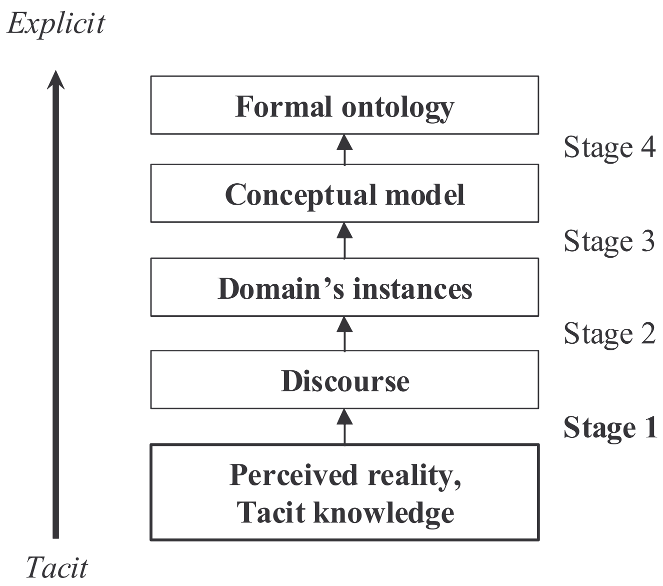

Figure 1. The different elicitation levels involved in the ontology design process

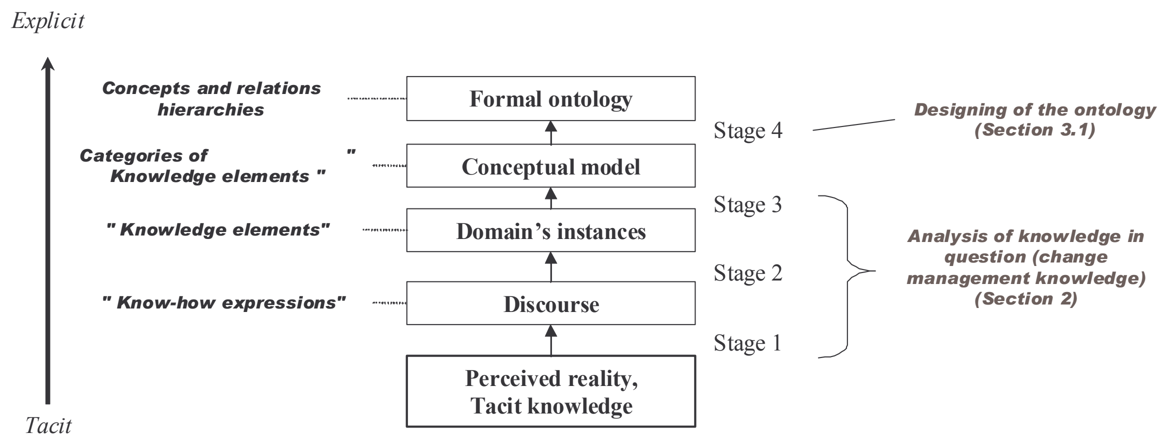

Figure 2. The different levels of results involved in the design process of the OCM and any ontology about tacit knowledge

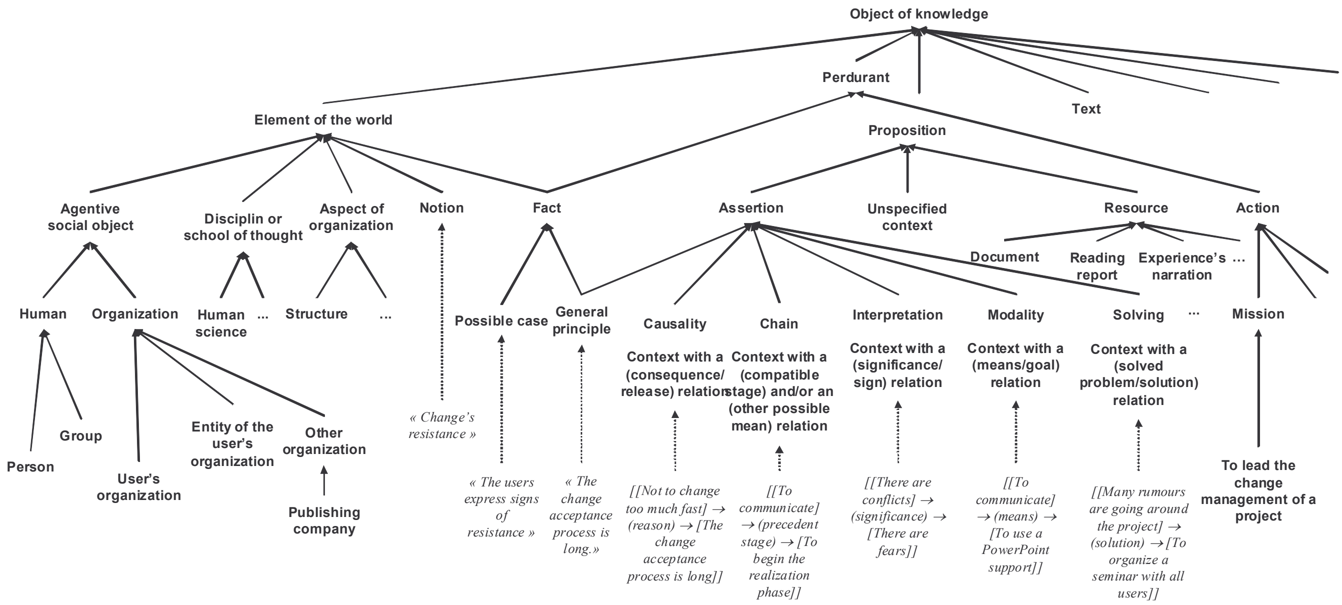

Figure 3. Hierarchy of OCM's concepts (high and middle levels) (extract)

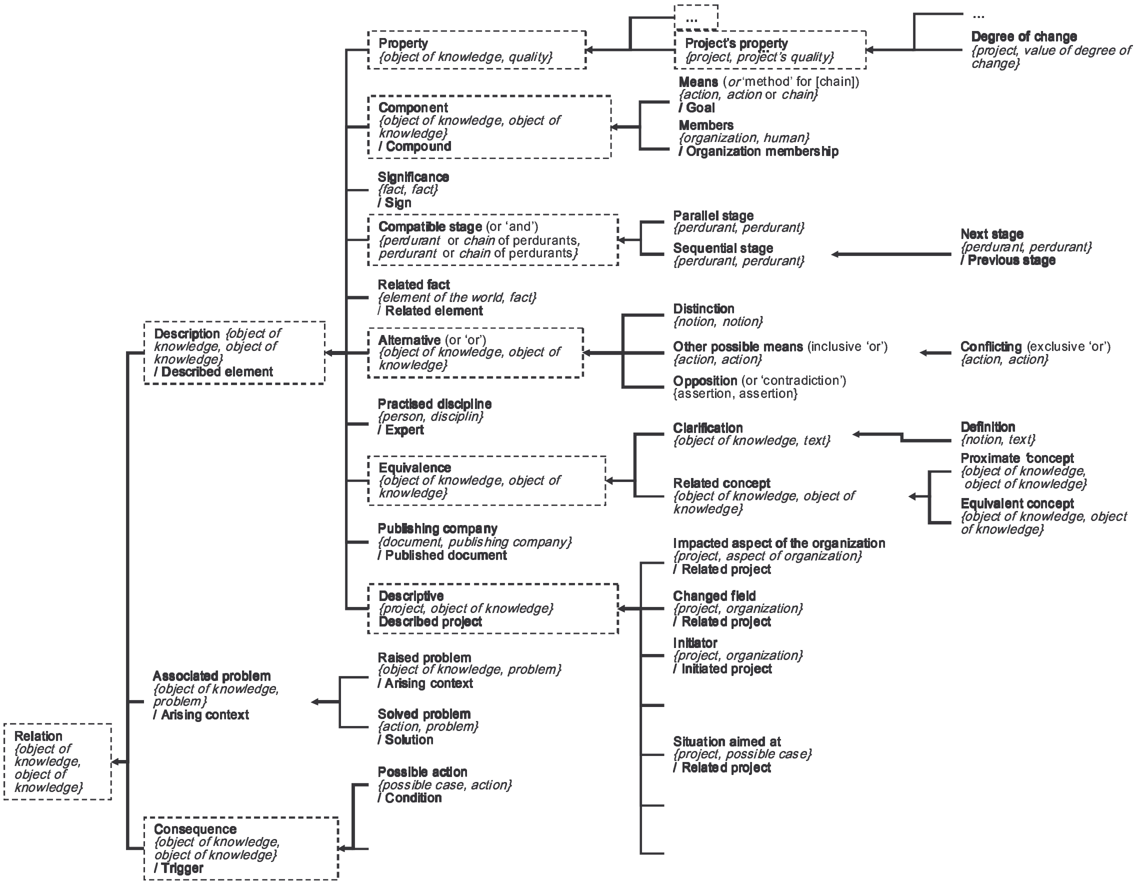

Figure 4. Hierarchy of OCM's relations (extract)

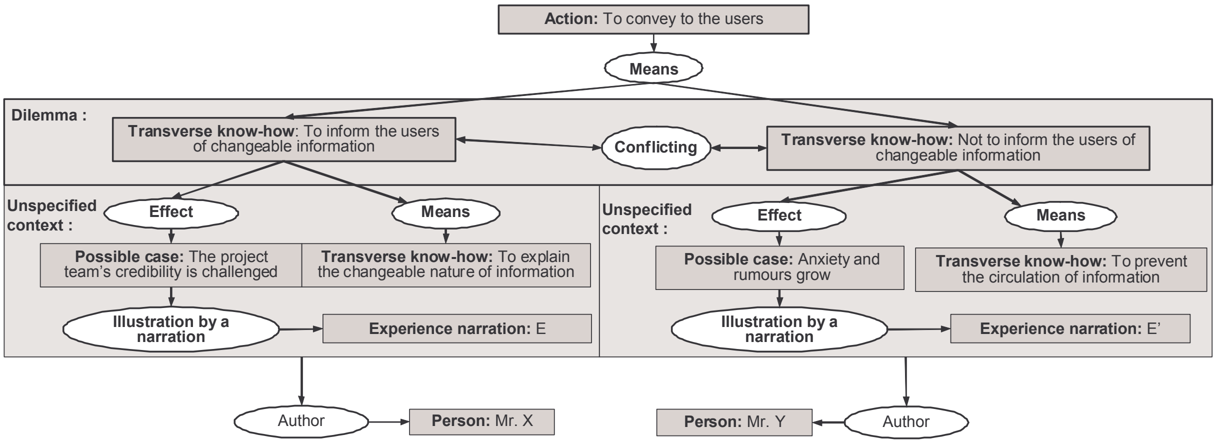

Figure 5. Example of a conceptual graph based of the OCM

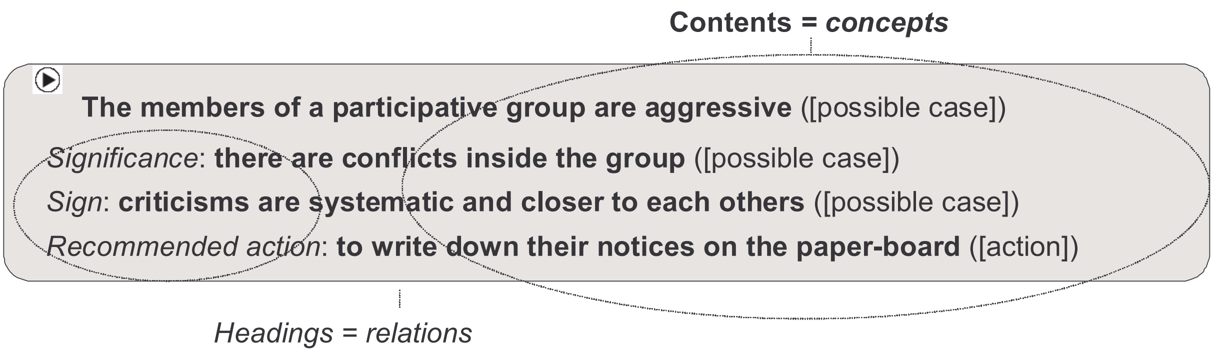

Figure 6. From the ontology to the interface

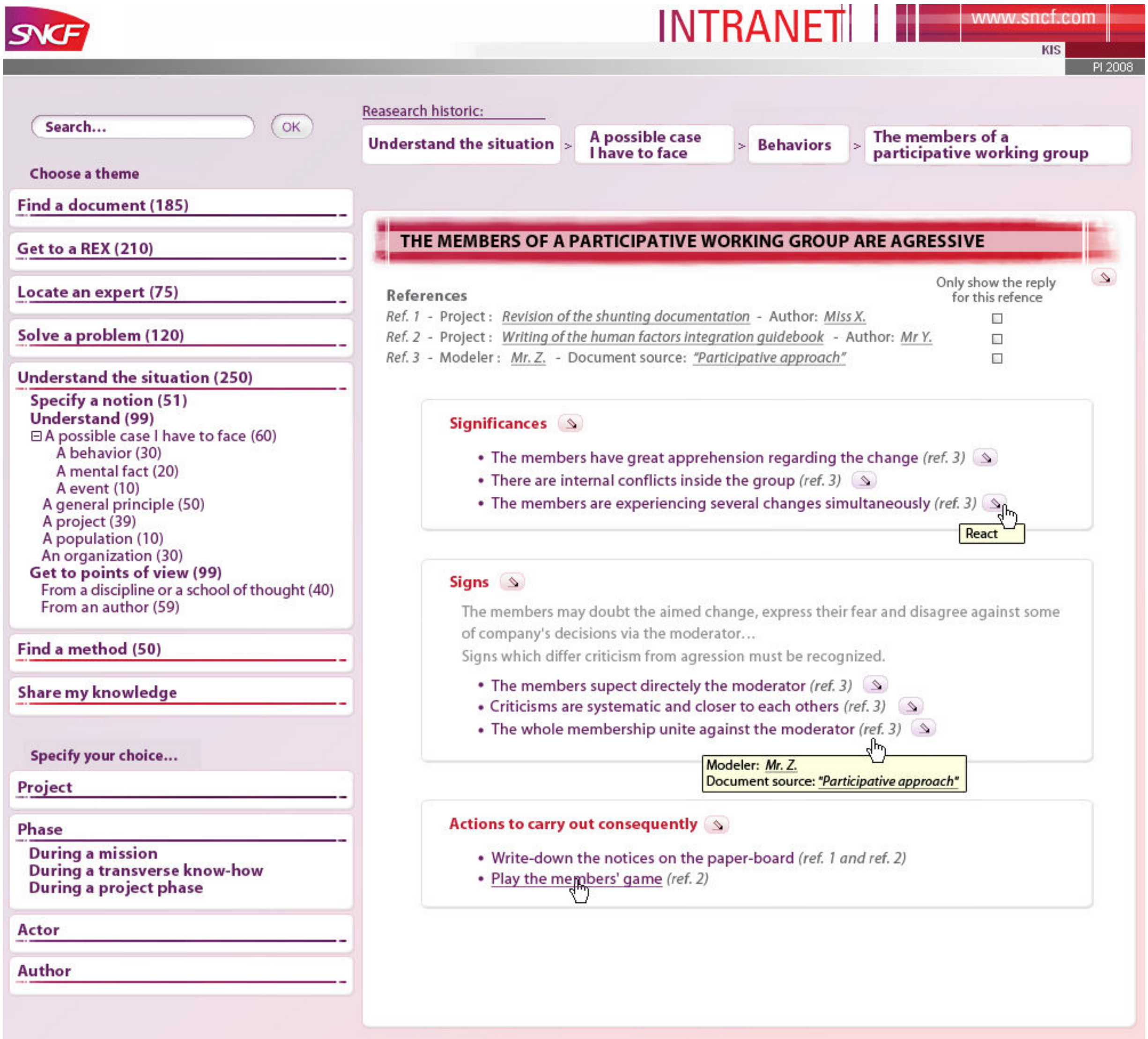

Figure 7. Interface of the change management knowledge server prototype

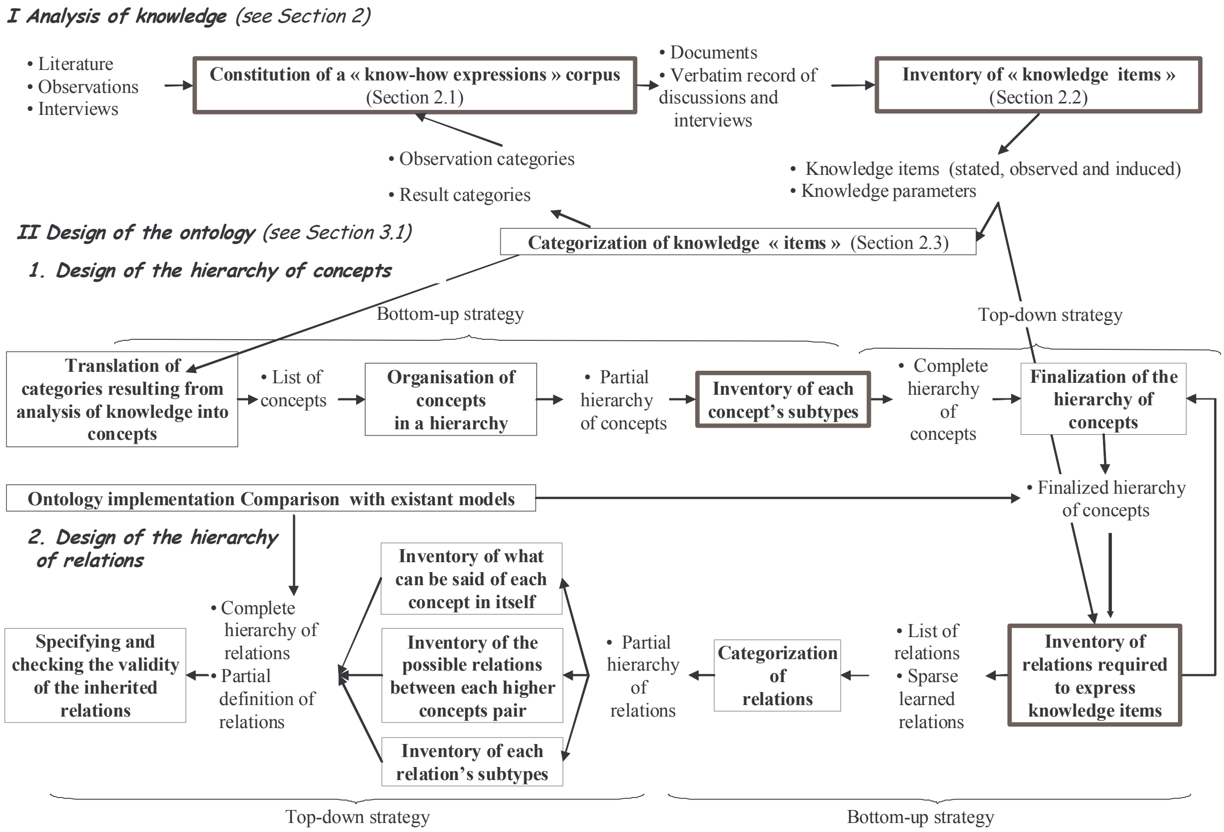

Figure 8. The methodology used to build the OCM

Anne Remillieux has a Research Master's degree in Philosophy (University of Creteil) and of a professional Master's degree in Knowledge Management (Paris IV - Sorbonne), she is a PhD student in Institut Télécom (Télécom Business School), she leads a research project on modelling and formalization of change management's knowledge in the National French Railways Company and on how to apply techniques of knowledge elicitation to the knowledge management's problems.

Claire Petitmengin is Associate Professor at Institut Télécom (Télécom Business School) and Member of the Centre de Recherche en Epistémologie Appliquée in Paris. Since her doctoral thesis of 1998 (under the direction of Franciso Varela), her research has focused on pre-reflective knowledge, the methods enabling us to become aware of it and describe it, and the methods enabling the detection of experiential generic structures. Her research evaluates the reliability of these methods, and their educational and therapeutic applications. She is also interested in the process of mutual guidance and refinement of "first person" and "third person" analyses in the context of "neuro-phenomenological" projects.

Jean-Louis Ermine obtained a PhD in fundamental mathematics and the title of National Research Director in computer science. For more than ten years, at the University of Algiers, then Bordeaux, he carried out research in mathematics, on algebraic geometry applied to functions of several complex variables. In 1985, he switched to Artificial Intelligence. In 1991, he joined the French Atomic Energy Commission (CEA) in Saclay, near Paris; from 1994 to 2000 he was responsible for the Knowledge Management Group in the CEA. In 2003, he joined the Business School of the French "Institut Télécom" as head of the Information Systems Department, to develop research and teaching activities in Knowledge Management. He is now Associate Dean for Research in this school. Jean-Louis Ermine is the inventor of the MASK method, a Knowledge Management methodology used in the CEA and in a great number of French and foreign companies since 1993. He has written more than 100 scientific articles and four books on Knowledge Management and Engineering, and has supervised 22 PhDs in these domains. He was named as "CEA senior expert" in 1996, and "CEA expert consultant" for external companies in 1997. Since 2002, he has acted as a KM expert at the United Nations (International Atomic Energy Agency, International Telecommunications Union). In 1999, he founded the French Knowledge Management Club, an association grouping a large number of French speaking companies. He is now the president of this club. Since 2006 he has been the invited researcher at the « Centre d'Études Francophones pour l'Informatisation des Organisations » (CEFRIO) in Canada. In 2007 he was named by the French government as KM expert for the UN/IAEA. Jean-Louis Ermine has been project leader or advisor in numerous research or industrial KM projects in public or private companies in France (Industry, Energy, Transport, Defence, Banking, SMEs etc.) and abroad (Sonatrach (Algeria), Hydro-Québec, Public Administration (Canada), Nuclear Research Centre (Brazil), National Nuclear Safety Authorities (IAEA Asia) etc.

Christian Blatter has two masters' degrees, in Ergonomics and in Psychology. He has worked in the Applied Psychology Service in SNCF (French Railways Company), in the Ergonomics section of the Human Resources Department. Then he was responsible for the change management in the EOLE project (a Paris metro line). He is now head of the Human and Social Science Unit in the Research and Innovation Direction of SNCF. He is a member of the French Society of Ergonomics, and participates to scientific committees of numerous seminars and conferences on his domain of expertise. He makes a lot of contributions (articles, communications, professional works, training) in the domains of socio-technical approaches, integration of human factors in projects, ergonomics in design (software, tools, workplaces …).