

Knowledge Sharing In Change Management: A Case Study In The French Railways Company

Anne Remillieux ¹, Claire Petitmengin ¹, Jean-Louis Ermine ¹, Christian Blatter ²

¹ Institut Télécom, France, ² SNCF, France

ABSTRACT:

This paper describes a project to represent and share tacit change management knowledge. Based on ontology, this project enables French Railways Company (SNCF)'s project managers to share their know-how through an IT device called «knowledge server».

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; 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 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 *know-how* (defined as an ability to use knowledge in practice) 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 e.g. Brénot and Tuvée (1996), Grouard and Meston (1993), Joule and Beauvois (2003), Watzlawick (1980), Crozier and Friedberg (1977), Latour (1993). 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. Later in this paper, these various degrees of implicitness are combined in the generic term "implicit". 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, 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).

This paper presents the domain Ontology for Change Management, which is built starting from the analysis of change management knowledge and composed of about fifty concepts (upper levels) and a hundred relations. OCM 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.

What follows is organized into five sections: after having described the methodology used to build 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.

2. Methodology

The methodology which has been used to design the OCM is composed of two main stages.

The first stage consists in analyzing the knowledge to be modelled by gathering a “know-how expressions corpus” starting from observations and interviews - classic semi-directive or elicitation type interviews. 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 unawares. This provides a corpus capable of expressing the required know-how - then extracting and categorizing “knowledge items” 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. The categorization of those “knowledge items” allows identifying the higher concepts of the hierarchy.

The second stage consists of designing the ontology by translating the previous results into hierarchies of concepts and relations.

First, we have built the *hierarchy of concepts* by translating the categories resulting from our analysis of change management knowledge into concepts. This was done without difficulty since 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] and so on. This was followed by organizing the listed concepts in a hierarchy and completing 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.

Once the hierarchy of concepts has been built, at least partially, we have designed 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.

Finally, we finalised the obtained hierarchies implementing them and harmonizing them as effectively as possible with existing models.

3. Presentation Of The OCM

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

3.1. Hierarchy Of Concepts

In this section, we present the hierarchy of the OCM’s concepts (Figure 1). 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. It should be noted that application of OCM to another set of know-how has not been tested. However, for the sake of clarity, Figure 1 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). 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 from its neighbours) of the “ontological commitment”, which determines a concept by the extension of the objects which refer to it. This is followed by the listing of its possible *relations* with other concepts

(note the relations, defined in 4.3, are organized in a hierarchy and we do not define the relations of property in this paper.). The listing of a concept's relations can be found by seeking the signatures of relations. 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 1, 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.

3.1.1. First Level Concept

Concepts are written [concept] and relations are written (relation); 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).

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], except [text] and [quality], 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).

3.1.2. 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.

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

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.

3.1.3. 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 1 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.).

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]].

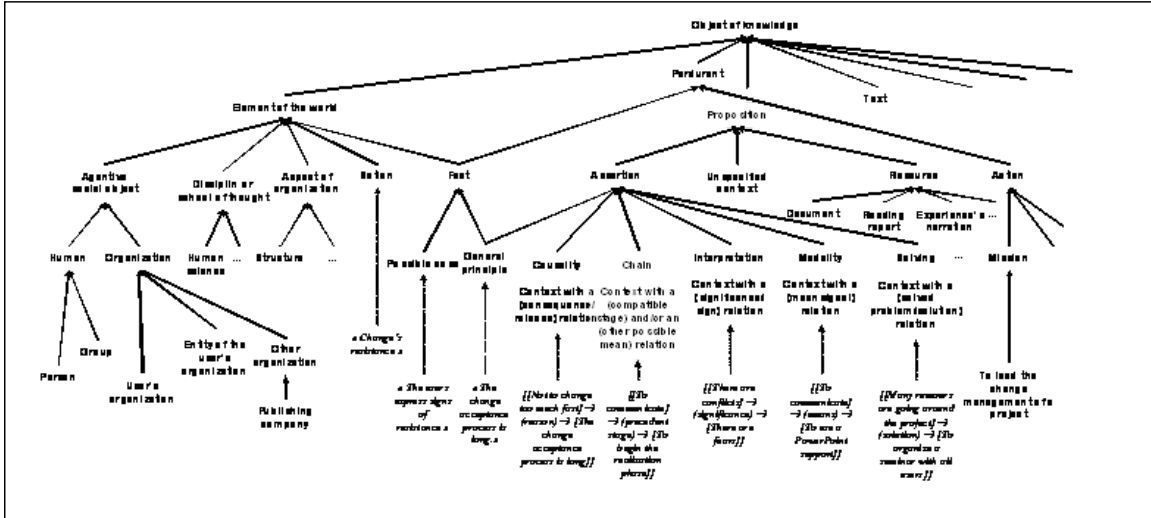
Example of instance: take the issue which consists in wondering whether, to communicate to the users, it is better to communicate about changeable information or not; this will be represented by the graph: [dilemma: [action: to communicate about changeable information] ↔ (conflicting) ↔ [action: not to communicate about changeable information]].

3.2. 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 2. 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 1: Hierarchy Of OCM’s Concepts (high and middle levels) (extract)



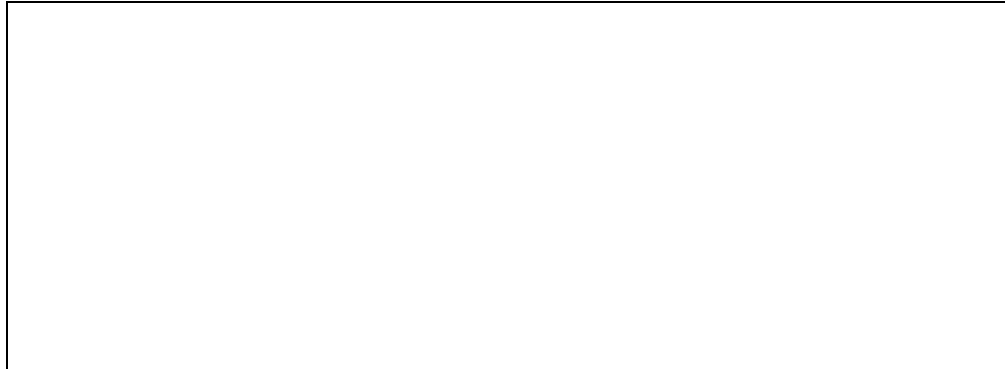
two inverse relations connect the same concepts but each in a different direction. In Figure 2, inverse relations are arranged the one below the other one, within the same node. Only the signature of the first one is indicated. For example, the inverse of the relation (agent): {action, person} is the relation (role): {person, action}. 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].

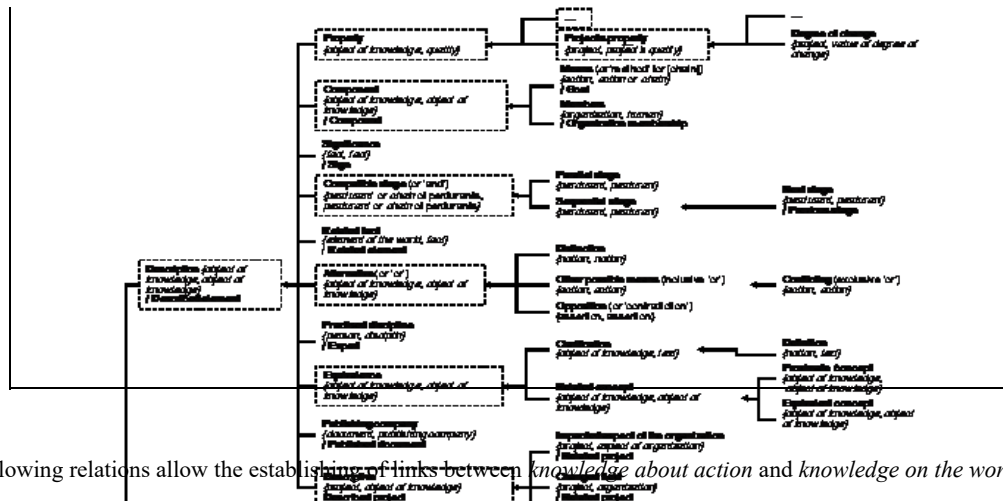
We indicate two inverse relations as follows: (agent)/(role).

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 2: Hierarchy Of OCM’s Relations (extract)





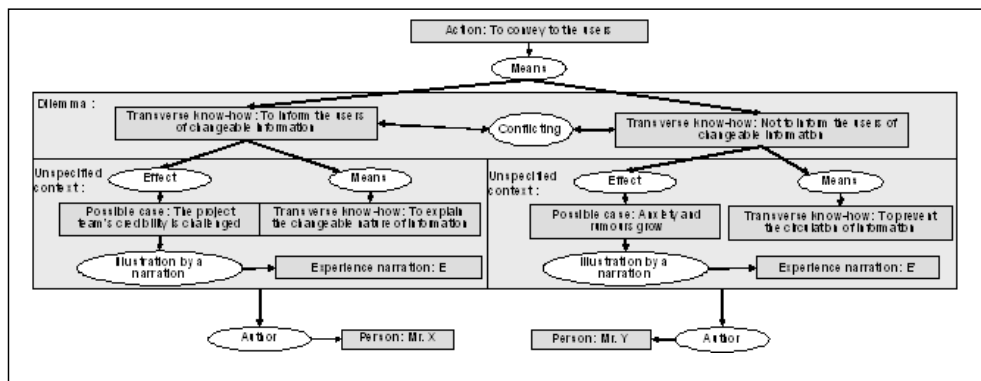
The following relations allow the establishing of links between *knowledge about action* and *knowledge on the world*:

- Possible action {possible case, action} / Condition: It is a condition of specifying which preliminary situation conditions the establishment 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 [assertion 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.

3.3. Graph Example

Figure 3 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 (see the input “Share my knowledge” in Figure 5).

Figure 3: 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 (See also the definitions of “Assertion” and “Unspecified context” in Section 3.2.) 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 (Remillieux et al, 2007; p.192).

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.

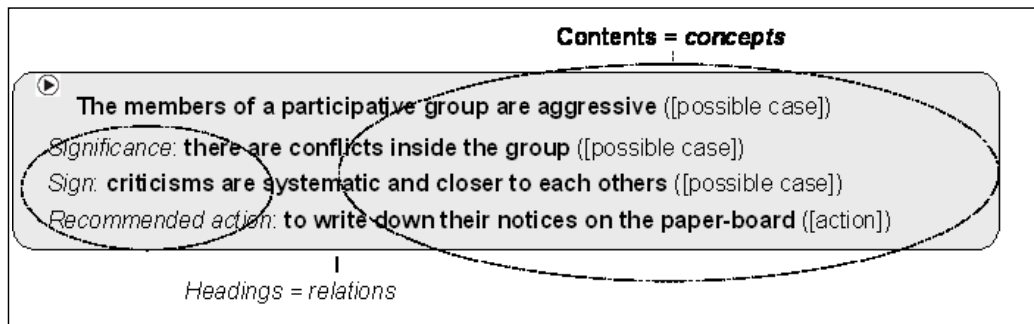
4. Prospects For Application

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 4). 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 4 is based on the following conceptual graphs:

- [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 4: 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 4, Figure 5 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 5: Interface Of The Change Management Knowledge Server Prototype



The screenshot shows the INTRANET interface with the following elements:

- Header:** SNCF logo, INTRANET | WWW.SNCF.COM, NIS, P.2008
- Search:** Search... OK
- Research historic:** Understand the situation > A possible case I have to face > Behaviors > The members of a participative working group
- Choose a theme:** Find a document (185), Get to a REX (210), Locate an expert (75), Solve a problem (120), Understand the situation (250), Find a method (50), Share my knowledge, Specify your choice...
- Understand the situation (250):**
 - Specify a notion (51)
 - Understand (99)
 - A possible case I have to face (60)
 - A behavior (30)
 - A mental fact (20)
 - A event (10)
 - A general principle (50)
 - A project (39)
 - A population (10)
 - An organization (30)
 - Get to points of view (99)
 - From a discipline or a school of thought (40)
 - From an author (59)
- Project
- Phase
 - During a mission
 - During a transverse know-how
 - During a project phase
- Actor
- Author

- Main Content Area:**
- THE MEMBERS OF A PARTICIPATIVE WORKING GROUP ARE AGGRESSIVE**
- References:**
 - Ref. 1 - Project : Revision of the shunting documentation - Author: Miss X.
 - Ref. 2 - Project : Writing of the human factors integration guidebook - Author: Mr Y.
 - Ref. 3 - Modeler : Mr. Z. - Document source: "Participative approach"
- Significances:**
 - The members have great apprehension regarding the change (ref. 3)
 - There are internal conflicts inside the group (ref. 3)
 - The members are experiencing several changes simultaneously (ref. 3)
- Signs:**
 - The members may doubt the aimed change, express their fear and disagree against some of company's decisions via the moderator...
 - Signs which differ criticism from aggression must be recognized:
 - The members suspect directly the moderator (ref. 3)
 - Criticisms are systematic and closer to each others (ref. 3)
 - The whole membership unite against the moderator (ref. 3)
- Actions to carry out consequently:**
 - Write-down the notices on the paper-board (ref. 1 and ref. 2)
 - Play the members' game (ref. 2)

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, Echter 2006) - is to identify requirements and *specify systems* rather than to *share knowledge*. Even if one of its benefits is to facilitate communication and negotiation between stakeholders. 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 (Masolo et al, 2003), SUMO (Niles and Pease, 2001), KR Ontology (Sowa, 2000)]. 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 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.

6. Conclusion

We have presented the Ontology for Change Management, constructed in order to allow SNCF actors to share their knowledge in change management.

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.

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.

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 building and Claire Goosens for having designed its interface.

8. 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.
- 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.
- Grouard, B. and F. Meston (1993). *L'entreprise en mouvement, conduire et réussir le changement*. Dunod.
- 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. Oramari (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., 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.
- Verheij, B. (2005). An argumentation core-ontology. In Agentlink technical Forum Group. Quoted by Echter (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).

About the Authors:

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). Her research project concerns the modelling and formalization of change management's knowledge in the National French Railways Company and 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 Francisco 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 ...).

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; Email: *christian.blatter@sncf.fr*
