Towards an Ontology of Crucial Knowledge Identification to Improve the K-DSS

Sabine Bruaux
Laboratory of Modelisation, Information, System
Picardie Jules Verne University
Amiens School of Management
Amiens, France
sabine.bruaux@u-picardie.fr

Inès Saad
Laboratory of Modelisation, Information, System
Picardie Jules Verne University
Amiens School of Management
Amiens, France
ines.saad@u-picardie.fr

***Abstract*— In this paper, we propose a characterization of the main classes contained in the database of the system K-DSS and related to the domain of identification of the crucial knowledge for which a capitalizing operation is required. We exploit ontological categories existing in the literature to define the notions of Knowledge, Actor, Support and Criteria of knowledge vulnerability. The objective is to improve the process of the crucial knowledge evaluation providing to different decision makers a unified semantic of these entities. Such approach brings us a preliminary analysis for the construction of an application ontology that we aim to integrate as a new component of the decision support system K-DSS.**

Keyword: Knowledge management; Ontology, Crucial Knowlegde, K-DSS, Multi-criteria Decision Aid

I. INTRODUCTION

The necessity to create and to use knowledge mobilized and produced in firms has increased rapidly these last years. Firms become aware of the importance of the immaterial capital owned by their employees which corresponds to their experience and accumulated knowledge about the firm activities. Maintaining this capital is powerful mean to improve the level of performance of the firm. In order to create, preserve and share knowledge in firms, Knowledge Management has been occupying since the beginning of the nineties a more and more important place within organizations. Thus, companies should invest in engineering methods and tools [11] in order to preserve knowledge especially those of *tacit* nature. Researchers in knowledge engineering and knowledge management have been focusing on the problems of acquisition, preservation and transfer of knowledge. However, considering the large amount of knowledge to be preserved, the firm must first determine knowledge that should make the object of capitalization. We should focalize on only the so called "crucial knowledge", i.e. the risk of their lost and the cost of their (re)creation is considered important; their contribution to reach the project objectives is very important and their use duration is long. Our previous research works also revealed the interest of the identification of crucial knowledge [34]. Not enough works exist concerning the identification of knowledge on which it is necessary to capitalize [18] [34] [41]. Thus, we have proposed a multicriteria method based on dominance rough set approach to identify and qualify crucial knowledge in order to justify a situation where knowledge capitalization is advisable. The value added of our methodology is to elicit the preference of the decision makers. The proposed method was conceived and validated in the French car Company [33]. This method is supported by a decision support system called K-DSS [34]. Our system K-DSS is based on two types of tasks: automation task and human task.

The K-DSS system implements a database in the form of a UML diagram of classes, which models the process of the knowledge assessment on a criteria family. However, this database has been designed without to give some meaning to the classes that it contains (e.g., the classes of *knowledge, process, actor*) [8]. Currently, the different criteria of evaluation (e.g., *scarcity, complexity, portability*) are the attributes of class knowledge. However, the notion of knowledge doesn't need the notions of scarcity, complexity or portability to be defined. We think that lack of semantic

In order to improve the performance of K-DSS, a first work is to specify the semantics of the UML classes in an ontology of the domain of the potentially crucial knowledge assessment. The construction of a such ontology in the context of the knowledge management system K-DSS, will define a shared vocabulary about the knowledge evaluation on the vulnerability criteria. This involves to define the elements of knowledge to which it is referred in the database (such as knowledge, tacit knowledge, explicit knowledge, individual knowledge, collective knowledge, actors, criteria of vulnerability, etc.), the relations between these elements and the semantics that they should be interpreted.

This article presents the first step of our process of the construction of an ontology reflecting the process of the knowledge potentially crucial evaluation.

In the following sections of this article, we first present the functional architecture of the decision support system K-DSS. We describe in particular the UML classes and relations involved in the process of identification of the knowledge (section 2). In the section 3, we expose a literature review to explain and justify our methodology of the construction of an ontology covering the domain of the evaluation of crucial knowledge. Then, we present the result of the first phase of the construction of an ontology, which means the conceptualization of the domain of the evaluation of crucial knowledge. We define in an informal language the concepts of Knowledge, Actor, Support and Criteria (section 4). Finally, we present our conclusions and perspectives (section 5).

II. Related Work

The Knowledge-Based Systems (KBS) are defined to reuse and share all or parts of the knowledge bases in order to extend the class of problems to be solved (e.g., car repairs, medical diagnostics, etc..) or to rely on skills of other systems (e.g., obtain an advice about a rare disease).

“Building knowledge-based systems today usually entails constructing new knowledge bases from scratch. It could be done by assembling reusable components. System developers would then only need to worry about creating the specialized knowledge and reasoners new to the specific task of their system. This new system would interoperate with existing systems, using them to perform some of its reasoning. In this way, declarative knowledge, problem-solving techniques and reasoning services would all be shared among systems. This approach would facilitate building bigger and better systems cheaply.” [39]

The principle of KBS relies on its internal structure. Since the middle of 80s, the modeling of knowledge for the development of a KBS differentiates the representation of the terminological knowledge of a domain from the modeling of treatments we want done on these knowledge, that we call the inferential knowledge. Basically, different kinds of knowledge are exploited by the KBS:

- the domain knowledge. For example, the knowledge: “a meningitis is common severe headaches” focuses on the disease meningitis, it helps to define by precising one of its frequent manifestation[1].
- the control knowledge, which details a method of use of the domain knowledge to solve a problem. For example, the knowledge: “if the patient has a sign corresponding to the frequent manifestation of a disease, then mention this disease as a hypothesis of diagnosis” exploits the facts to provide a method and discuss the hypothesis of diagnosis.
- the rules, which are formulated in the form of empirical associations between the characteristics of the problem and the possible solutions. For example, the knowledge: “if the presence of severe headaches, then think about meningitis” domain knowledge and control knowledge.
- the constraints, which allow to specify impossibilities or obligations, for example: “a meningitis can affect anyone at any age, from newborns to seniors”.

Thus, the KBS is able to solve a problem through a series of deductions (inferences). The construction of the KBS has asked the question of the construction of models of knowledge (e.g., domain model, reasoning model) and has highlighted the fact that during this construction takes place a process of creation of new knowledge. Such knowledge is not a knowledge already present in the head of the expert (s). The term "knowledge-based system" reflects this new approach of knowledge acquisition. Thus, from the initial practice of knowledge acquisition (do precisely the reasoning human), we have moved progressively to a practice of the structuration and the formalization of knowledge, in other words, a practice of the construction of models.

The researches are oriented now to the activity of the construction of a model of knowledge, which is no more focused on the problem-solving performance of the system (similar to those of the expert), but on how the problem-solving knowledge are used in interaction with the user into the cooperative systems. The research on the sharing, the reuse of knowledge bases and the semantic interoperability of KBS (ARPA Knowledge Sharing Effort project [39]) requires the use of ontologies to express knowledge by using the primitive of specification defined at a conceptual level, independently of any formal representation.

An ontology is an explicit specification of a conceptualization covering a domain of knowledge [17]. The term “explicit specification” means that the design is represented in a natural language or formal. The term "conceptualization" refers to a system of concepts. An ontology defines the central terms of a domain of knowledge and the consensual semantics associated with these terms, in the form of concepts related to each other by taxonomic (hierarchical) and semantic relations [42]. In the medical domain, for example, the knowledge will focus on the function of an organ, the effects of an antibiotic, or the manifestations of a disease. The domain is divided into categories of entities such as: "body", "pathophysiological process," "disease," "physiological function," linked by relations: "causes", "manifested by", "provides the function of ".

A concept can be defined as an entity composed of a term (e.g., the term "star"), an intension, which is the set of properties reflecting the meaning of the concept (e.g., a bright spot in the sky at night) and an extension that is the set of the objects (called instances of the concept) denoted by the concept (all bright spots). This method of definition is a long tradition that can be traced back to the Greek philosopher Aristotle [4]. By convention, a relation is also characterized by a term (e.g., "to be the author of"), an intension, which helps to express the meaning of the relation by specifying the concepts that it connects (e.g., “R is a relation between a person or a group who created a document, and its intellectual content, its arrangement or its shape") and an extension that is the set of the tuples of instances linked by the relation (e.g.: (Hugo, Notre Dame Paris)). The relations have in addition a "signature", a list specifying the types of instances that they connect, or for our example: (person, Document).

We have seen that the properties (or intensions) of concepts and relations involved in the definition of the semantics of a domain of knowledge. More generally, all the properties specific to the domain of knowledge, which

[1] We take the example of a KBS used to generate diagnostic hypotheses proposed in [23].

contribute to express the meaning of concepts and relations, and how to use them in the application, are represented in the ontology by axioms. Since the objective is to share meaning, these primitives should get the meaning of the term as objectively as possible, i.e., independently of the use that we want to do of these knowledge [21]. “In order to integrate an ontology in a KBS, it should be translating into a form suitable for the use of the KBS, i.e., it should be specified the semantic of the manipulation of axioms. Thus, an axiom can be used to infer new knowledge or to validate the adequacy of a knowledge in relation to the semantic of the domain” [29].

As the ontologies are particular knowledge bases, the methods of the construction of the ontologies incorporate the main principles of the construction of the KBS. In particular, the construction of an ontology is done by successive transformations of ontological models. It is customary to distinguish three main stages in the construction of an ontology in a formalism that allows the manipulation of knowledge in an domain with a KBS [29]:

1. **The knowledge acquisition.** This process consists in identifying a corpus (which may contain for example terminological bases, technical documentation, summaries of interviews, questionnaires) covering all the documents of a given domain, knowledge for the operational needs in terms of concepts, relations, instances and axioms (i.e., the semantics of the domain). This process, which, from raw data, leads to a conceptual model informal (e.g., in a natural language) or semi- informal (ex : CML [35] and UFO [17] languages), is called conceptualization.
2. **The knowledge modeling.** This is to structure all conceptual entities, identified during the step of acquiring knowledge, and to formalize them in a language of representation of ontologies (e.g., the languages based on frames, the description logics, Conceptual Graphs [36], Ontolingua [19], RDF [22]). This process, which, from a conceptual model leads to an ontology (semi-formal) is called ontologization.
3. **The knowledge representation.** This is to clarify the semantic of the manipulation of the axioms in order to allow the KBS to reason about the knowledge of the domain (depending on the scenario for use by the application, like enable the KBS to take decisions). The ontology obtained in the previous step must therefore be specified in an operational representation (e.g., FLogic, KIF, OCML, RDFS, DAML, OIL, OWL), i.e., a formal language that has inferential mechanisms (facts, rules and constraints). This process of specification of a semi-formal ontology into a model executable by a machine (operational ontology or computational ontology) is called operationalization.

Several methodologies have been proposed for building ontologies. Some methodologies are planning to take over the whole process of the specification at the conceptual level of an ontology to its formalization (e.g., METHONTOLOGY [21], TERMINAE [17], the method of Gruninger and Fox [19], On-To-Knowledge [37]). Thus, they distinguish two levels of modeling: the modeling to establish the meaning and the modeling to implement a KBS. Other methods focus on one phase of the process (conceptualization, ontologization, operationalization). The methods Cyc [44], SENSUS [40], the approach KACTUS [27] and the method of Uschold and King [42] for example insist on the stage of conceptualization. The methods OntoSpec [27], Archonte [29] and OntoClean [22] provide a help for the structuration of the hierarchies of concepts and relations during the phase of ontologization.

Like the methodologies, many tools to build the ontologies have been developed. These include: KAON [39], OntoEdit [39] based on the methodology On-To-Knowledge, Protégé-2000 [30], Oiled [5], WebODE [1] that implements the methodology METHONTOLOGY.

The construction of a KBS begins, before the implementation (not to constraint the representations with the criteria of performance or computability), by the abstract description of the system, by using the primitive of specification of the conceptual model at the knowledge level (KADS method [23], method MACAO method [33]). It is possible to take advantage of the existence of the repeated structures in the conceptual models. The reuse of the generic components is a process of specialization which consists in adapting the generic problem-solving method the most appropriate to a class of problems in the application domain. The model of reasoning of the application is a specialization of the generic problem-solving methods selected. This principle can also be applied to the elements of a domain, by reusing a generic domain ontology that contains generic concepts from the systemic (e.g., state, function, system). A help to the modeling of knowledge consists in reusing ontologies already built.

In the next section, we present the result of the first phase of the construction of an ontology relative to the process of crucial knowledge evaluation. The phase of the conceptualization of the domain of crucial knowledge evaluation is the most long and the most difficult. This phase consists in identifying the terms structuring the domain of the potentially crucial knowledge evaluation in terms of concepts, relations, instances and axioms (e;g., define the minimum and sufficient conditions to to say that an object belongs to a given class), from available resources. Here, the resources are the interviews with experts modeled as a diagram of UML class [33]. This diagram models the process of critical knowledge evaluation. We apply a “middle-out” approach for the identification of the central concepts of the ontology, we will generalize and specialize to complete the ontology [43]. It is recognized that this approach promotes the modularity and the stability of the resulting ontology. We also exploit the ontological frameworks existing in the literature (parts of high-level ontologies and domain ontologies) to clarify the definitions of concepts and relations of ontology.

III. FUNCTIONAL ARCHITECTURE OF K-DSS

This section describes the functional architecture of the system K-DSS.

First, it is important to identify the specialized roles played by the persons concerned by the knowledge identification decision system (Figure 1).

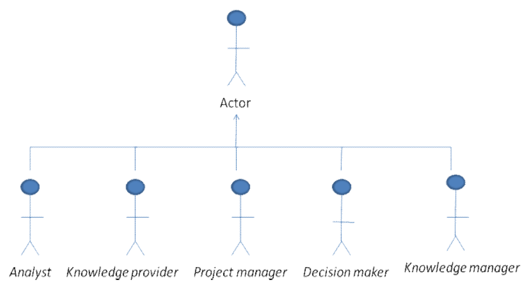

Figure 1. the specialized roles played by the persons concerned by the knowledge identification decision system

The following list enumerates the main involved internal and/or external actors of the organization:

- *Knowledge provider*. An important role in the crucial knowledge identification decision process is played by the knowledge “owner” or provider. The knowledge provider is generally an expert in the project under study but can also be a different person in the organization who is not considered to be an “expert” .
- *Project manager*. The project manager is responsible for running the project considered by the crucial knowledge identification decision process. So, she or he is involved in all the phases of the decision process.
- *Decision maker*. A decision maker is an individual or a group of individuals who, because of their value system, directly influence the final recommendation.
- *Knowledge manager*. The knowledge manager formulates knowledge identification, preservation, distribution and actualisation.
- *Analyst*. An analyst is not involved in development project. He formulates criteria and preference model to help decision makers for using the system and identifying crucial knowledge.

Two phases may be distinguished. The first phase is relative to the construction of the preference model. The preference model is represented in terms of decision rules. The second phase concerns the classification of “potential crucial knowledge" by using the rules collectively identified by all the decision makers during the construction of the preference model.

A. *Construction of the preference model*

This phase consists in identifying, from the ones proposed, an algorithm for computing the contribution degrees. The selection is collectively established by all the decision makers with the help of the analyst. Whatever the selected algorithm, it uses the matrices Knowledge-Process (K-P), Process-pRoject (P-R) and pRoject-Objective (R- O) extracted from the database more specifically from the three association classes “Evaluate-K-P", “Evaluate-P-R "and “Evaluate-R-O " to compute the contribution degree of each piece of knowledge into each objective. To avoid data redundancy, these matrices are not explicitly stored in the database but generated during processing. Only their intentional definitions are permanently stored in the system.

Once these matrices are generated, the contribution degrees are first stored temporally in a decision table and then introduced in the database. As for matrices, only the intentional definition of the decision table is maintained in the system.

The decision table (Table 1) contains also the evaluation of the “Reference crucial knowledge" concerning the vulnerability and use duration criteria extracted from the class “Knowledge" precisely. These evaluations are collectively defined and introduced by the analyst into the database. The analyst should introduce in the decision table, and for each decision maker, the decisions concerning the assignment of “Reference crucial knowledge" into decision classes Cl1: “Not crucial knowledge” and Cl2:”Crucial Knowledge”.

K_i	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	g_{10}	g_{11}	g_{12}	g_{13}	g_{14}	g_{15}	Decision class
K_8	2	2	3	3	1	2	4	4	5	2	4	5	5	5	2	Cl1
K_9	3	3	2	2	3	3	4	4	4	2	4	4	3	4	2	Cl1
K_{16}	2	3	3	2	2	2	3	4	5	2	5	5	5	2	2	Cl2

Table 1. An extraction from the decision table for one decision maker

The decision table contains, in addition to the columns relative to vulnerability and those relative to contribution degree and use duration criteria, as many columns as decision makers. Once the decision table is generated, it will be used as the input of the induction algorithm selected by the decision makers.

This algorithm permits to generate the list of the initial rules for each decision maker. It is important to mention again that only rules relative to class Cl2 are stored. Then, each decision maker should select a subset from these initial rules. The next step in this phase consists to collectively select, from the set of decision rules individually identified by the different decision makers, a subset of decision rules that will be used latter by JESS for the classification phase.

B. *Evaluation of “potential crucial knowledge"*

The second phase consists in classifying the new knowledge called “potential crucial knowledge". As the previous one, this phase starts by identifying the algorithm to be used for computing the contribution degree of each piece of knowledge into each objective. This algorithm uses as input the information relative to the performances of "potential crucial knowledge" previously introduced in the matrices K-P, P-R and R-O. The results are stored in a performance table. The information contained in the performance table are then transformed into facts. The inference engine incorporated in JESS verifies first if exists at least one rule that verifies the different facts and if this holds, the piece of knowledge is classified as crucial; otherwise the piece of knowledge is considered non crucial.

C. Database

The UML-based conceptual schema of the database is shown in figure 3. The central class in the model is the class "Knowledge". It is described with an unique number (K-Num), a name (K-Name), a description (K-Description), eight attributes (Complexity-Level, Substitutability-Level, Validation-Level, Transferability-level, Scarcity-Level, Acquisition-Cost, Production-Time, and Accessibility-Level) corresponding to the eight criteria $g_1, g_2, \ldots,$ and g_8 composing knowledge vulnerability family, use duration (Use-Duration) corresponding to the only criterion, g_{15}, of use duration family, (Knowledge-Type) (i.e. "reference crucial knowledge" or "potential crucial knowledge").

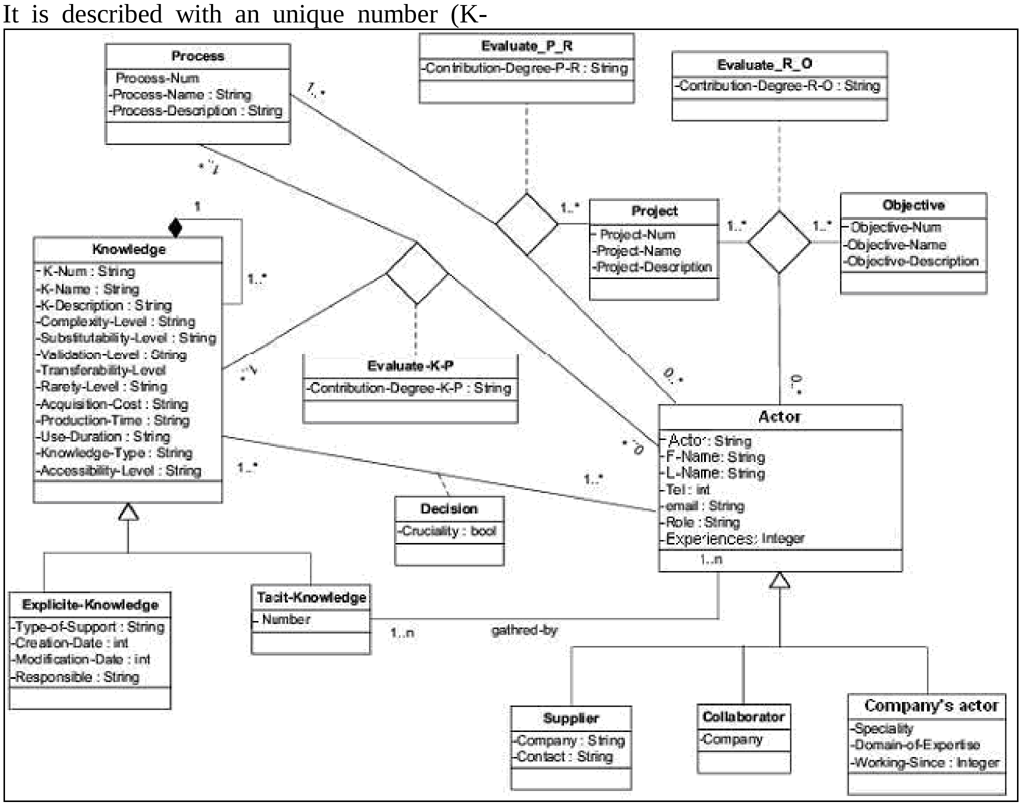

Figure 3. Database [Saad, 2005]

Below we quickly specify the content of the criteria used. These criteria are constructed based on a real context and a real-world case study conducted in an automobile company. We believe that these criteria are generally valid for the entire problem of identification of knowledge requires an operation funded through a transfer to similar projects[2]:

- *Scarcity* represents the number of person (internal or external to the organization) who own the knowledge.
- *Transferability* is the degree of the transfer of individual or collective knowledge. We have based our analyze on the definition given by Davenport and Prusak [10] "*knowledge transfer involves two actions: transmission and absorption by that person or group*" to measure the degree of transferability of knowledge. So we distinguish two states for measuring the knowledge transferability :
    - o *Transmission* represents the degree to which an individual or group of individual can transmit his knowledge to other person. It is difficult to transmit individual knowledge because it is fundamentally tacit. Knowledge incorporates so much embedded learning that its rules may be separate from how individual acts.
    - o *Absorption* is the degree to which individual can appropriate the knowledge by either studying technical document or talking to her predecessor skilled individual.

[2] So far, we suggest to the knowledge manager an update of the definitions of criteria and scales if necessary, according to the needs of actors.

- *Imitability* represents the degree to which competitors can copy knowledge by analyzing patent, by experiencing product, etc.
- *Accessibility* represents the time needed to access to the knowledge. The accessibility notion is relative because it has to be compared with the length of time the person actually has to access to the knowledge.
- *Complexity* represents the degree to which multiple kind of knowledge domain are needed to create a knowledge in the process or to adapt it to another context.
- *Validity* represents the validation state of knowledge. We distinguish two types of validation :
    - knowledge validated by macroscopic or microscope experiments
    - knowledge validated by experts ; for example; thesis, patent...
- *Substitutability* is the degree to which knowledge can be replaced by an other knowledge to take the same task with the same performance.
- *Cost and time of knowledge production* represents the number of persons and the period needed to create the knowledge.
- *Use duration :* The evaluation concerning criterion g_{15} is provided by experts. For example, the knowledge relative to "the measurement of the additive" has an "average use duration" because is related to the use duration of the first generation of depollution system; new generations of the depollution systems are without additive.

Note finally that a piece of knowledge may be composed of several elementary pieces of knowledge. This is modelled by the aggregation relation defined on the class "Knowledge".

The classes "Explicit-Knowledge" and "Tacit-Knowledge" are specializations of the class "Knowledge". The "Explicit- Knowledge" class permits to identify for each explicit knowledge the set of supports (documents, database, knowledge base system) on which this knowledge is represented. If the knowledge is tacit, it is characterized with the person who gathers it. This information is deduced from the relationship between "Tacit-Knowledge" and "Actor". The class "Actor" contains the information relative to the different actors (Id, Name, Telephone, Email, Role, Experience). The class "Actor" is specialized into three classes: "Supplier", "Collaborator" and "Company's Actor".

The three classes: "Process", "Project" and "Objective" permit to handle the information relative to the names and descriptions of processes, projects and objectives, respectively. The association class "Evaluate-K-P" between "Actor", "Process" and "Knowledge" stores the contribution degree of a knowledge into a process (Contribution-Degree-K-P) attributed by a given actor.

IV. Conceptualization of the Domain of Crucial Knowledge Evaluation

The various experiments "in situ" revealed that it was difficult for an expert to assess directly knowledge on certain criteria. Therefore, we propose to conduct a thorough analysis of this entities intervening in the knowledge assessment. In this section we define the concepts of Knowledge, Actor, Support and Criteria extracted from the database (figure 1) by reusing ontological categories defined in the literature.

A. Domain analysis

1) Knowledge

The notion of knowledge identified in dictionaries means the ability of an individual (according to his learning faculty and memory) to analyze and understand information in order to assimilate and generate an interpretation and an own representation (tacit or explicit) with the intention to act in a given context. Each individual has his own knowledge. Each one represents the world in its own way and this representation determines how it addresses the problems (in accordance with the interests of time, mood, etc.). As such, we consider knowledge as a set of beliefs held by an individual (or several). In reference to the Belief-Desire-Intention paradigm, beliefs reflect the knowledge that can have an individual on the universe to which he/she belongs.

This acceptation of knowledge is closer to the one of the notion of *Computed Belief* which is defined in the COM ontology [12]. The principle is that an intentional agent have a *Mental State* (e.g., a *Belief*) about a *Mental Object* (respectively, a *Computed Belief*) at a certain time. Our notion of the knowledge rejoins that of *Proposition* defined in the I&DA ontology [29] or that of *Description* of the D&S ontology [35]. A *Proposition/Description* represents a mean for an individual to describe situations that he/she considers as existing in the world. In particular, a *Proposition* may correspond to the content of a document (this is important for the follow).

To lead a more effective analysis, we require to characterize and locate knowledge. Thus, in the context of activities within the car company, we mainly distinguish three different types of knowledge needed to control processes and which can be sensitive and crucial to the organization in question. They are:

- knowledge about the development and the adaptation of material resources necessary to lead the activity (e.g., knowledge about the adaptation of a chemical model, knowledge about the development of a simulation tool able to predict the rate of diesel dilution in oil)
- knowledge necessary to lighten some technical constraints of the activity: it is used indirectly in the activity and produced outside of the project;
- knowledge produced or used during the activity (e.g., knowledge about the improvement of strategies related to the supervisor).

We can differentiate more specifically two main categories of knowledge necessary for the control of sensitive process:

- the knowledge *produced during an activity* may be produced either intentionally or not. They may therefore be a desired outcome or result of a "side effect" of the activity, not predictable a priori. They were produced (in the sense that they are new knowledge) or processed during this activity (e.g., an updating of knowledge).
- the knowledge *used during an activity* (directly or indirectly such as the knowledge necessary to lighten some technical constraints of the activity, the knowledge about the development and the adaptation of material resources necessary to lead the activity) provide a help for an agent (or several) to carry out an action (to reach a goal).

This distinction allow us to precise our definitional framework by assimilating knowledge *produced during an activity* and knowledge *used during an activity* respectively to *artificial entities* and *functional entities* in the broadest sense of the definitions given in a recent study [37].

Finally, the UML model (see figure 1) distinguishes two classes of knowledge: the *Tacit-Knowledge* class (linked to the *Actor* class) and the *Explicit-Knowledge* class. We show in the following sections that we are doing the distinction between knowledge held by an individual (or several) (the *Tacit-Knowledge* class) (§2) and those inscribed on a support (the *Explicit-Knowledge* class) (§3). If we go back to the UML definition of the *Knowledge* class, a knowledge is defined by eight particular attributes: the criteria of scarcity, transferability, imitability, accessibility, complexity, validity, substitutability and the cost and time of knowledge production. In the section 4, we try to clarify the nature of these criteria necessary to classify the potential crucial knowledge.

2) Actors

When we are interested in the knowledge assessment, we consider the organization where knowledge is mobilized and used by different actors.

According to [26], the fact to know is similar to fact to be likely to act. A *knowledge* is therefore "actionable" and "to be likely to act" joined the concept of ability (or faculty) to perform an action. From a consensual point of view in AI, the notion of ability implies that knowledge is in an “ideal” level (it's a "private" experience) and belongs to mental world proper to an individual. It therefore does not coincide with any of the actions carried out:

As a potential (or ability, talent), the ability exists independently of the action to which it relates and whether that action succeeds or fails, then regardless of whether the result exists or not. [37]

We talk more specifically about competence, ability or talent of an individual. The knowledge is therefore "owned" by an individual (or several) giving him/her the ability to perform (and to repeat) an action (to reach a goal).

This faculty to perform an action is embodied in an entity defined in [39], the *Agentive*. An *Agentive* could be a human being, a robot, a knowledge-based system or an organization. He/she acts with the intent to achieve a goal and implements the appropriate means to achieve his goals. An *Agentive* plays the role of *Agent* (in the sense of [32]) during the *Action* in which it participates (according to the relationship of participation of the DOLCE ontology [28]). This means that an individual intern or extern to the organization (e.g., the French car company) could be both a decision maker and may be also a knowledge provider and/or a project manager. The notions of actor presented in the section 2 (*Knowledge provider, Project manager, Decision maker, Knowledge manager* and *Analyst*) are therefore specialized roles of *Agent* in the context of the process of knowledge evaluation.

3) Support

In our analysis, the knowledge results from the interpretation (the sense) given to any entity (an object, a process) by an individual or a social group (i.e., a community of intentional agents) in an organization. This knowledge is either owned by an individual or a social group (in the form of a mental inscription), or included on a support (e.g. a sheet of paper, an audio-visual document, a CD, the computer’s memories, etc.). The organizations have the “capacity” to give a status to certain objects (for example, a piece of paper can acquire the status of a bill because members of the organization recognized as such) [34].

According to the theory of the support of [44], a *Support* is a physical object having a semiotic inscription of knowledge (e.g. a text manuscript or printed materialized by some ink and formatted) intelligible for a cognitive agent (e.g. a human being, a software, a robot, etc.). This implies that the agent have the competences for interpreting the form perceived and give it meaning. This also implies that this form has been apprehended internally by the agent in the form of a mental inscription.

Therefore, the entity which makes sense is neither the document nor the object but an mental inscription resulting from the perception of the document by an agent. This can reflect the fact that objects that are not documents (which were not intentionally created as such) are not intrinsically sense but that agents can make meaningful perception of these objects. This can also reflect the fact that the nature of these objects can be any and it can include practices or temporal objects (like *Perdurants* in the meaning of DOLCE: which “happens in time” like processus, events, actions, etc).

For our needs, we restrict to the supports specially designed by the human to vehiculate and communicate meaning (database, knowledge base system). In other words, we only take into account neither natural objects nor artificial objects communicating accidentally sense (for example, the location of the moss on tree trunks informs on the direction of the wind in a geographical area).

4) Criteria of knowledge vulnerability

The UML definition of the *Knowledge* class is defined by a family of criteria (scarcity, transferability, imitability, accessibility, complexity, validity, substitutability and the cost and time of knowledge production) whose aims to influence the opinion of the decision makers about the cruciality of knowledge. However, within the meaning of the DOLCE ontology, these criteria could not be defined as properties (specifying the concept *Quality* defined in DOLCE) of the concept of knowledge because it is obvious that they are not "inherent" to a knowledge. This modeling choices have sense (i.e., they depend on the knowledge to which they assign) only for computing the contribution degree of knowledge in the studied context. This is confirmed by the definitions of the very heterogeneous criteria reminded in the section 2.3 of this paper: the *Scarcity* of the knowledge in the context of the knowledge evaluation amounts to assess the number of persons owning a certain knowledge, the *Cost and time of knowledge production* criterion represents the evaluated number of persons and the evaluated period needed to create the knowledge, and so on.

This definitional framework is that of a meta-level because it reflects the idea that a decision maker has about a given knowledge. In this consideration, we are interested in the work done by A. Gangemi in a technical report [16] about the ontology evaluation design pattern. In our turn, we could consider a quality-oriented knowledge description in which we would define parameters for the quality of knowledge (e.g., scarcity, complexity). However, for the same reasons evocated above, we cannot propose definitions of these criteria at a meta level.

A fact is that each definition of criteria assumes an action of knowledge assessment on certain criteria leading to transform the knowledge state of a decision maker. We assimilate the knowledge assessment on the different criteria to a reasoning which is decomposed into several sub-reasonings (for example, *Assessing knowledge scarcity* consisting in the assessment of the number of person who own the knowledge; *Assessing knowledge accessibility* consisting in the assessment of the time needed to access to the knowledge; and so on). This reasoning have for data a given knowledge to assess (for example, knowledge relative to material of filter support).

B. *Conceptual model*

On the basis of this previous analysis, we propose a modeling framework which consists in reusing the ontological resources defined by the team of G. Kassel in the MIS laboratory, extending the DOLCE ontology. These resources are available at the URL: http://www.laria.u-picardie.fr/IC/site/spip.php?article53. More precisely, our modelling framework exploits:

- the core ontology of *Actions*;
- the core ontology of *Participant roles* (also called "casuals roles" or "thematic roles" in the literature), which cover the domain of the "modes of participation" of the entities intervening in the evaluation of the crucial knowledge;
- the core ontology I&DA, which cover the domain of semiotics, initially built to classify documents by their contents.

By admitting that all knowledge is knowledge about "something", about an "object", we can schematically distinguish between two categories of knowledge, depending on the nature of the objects (physical or mental) with which it deals:

- *practical* knowledge (i.e. know-how "to act") deals with *physical* objects and enables action in the real world (e.g. banging in a nail, riding a bicycle)
- *theoretical* knowledge (i.e. know-how "to think") deals with *theoretical* objects (mental objects) and enables action in the mental world (e.g. calculating, deciding).

According to this definition and assimilating *Actions* to transformations of a world (entities), the core ontology of *Actions* divides actions into *Doings* (*Physical Actions*), which are actions on the physical world, and *Non-Physical Actions*, which aim at transforming the agent reasoner's mental world. It means that the modification does not concern the real world but the representation that the reasoner makes of the real world. Among the *Non-Physical Actions*, we distinguish the *Conceptual Human Actions* which transform *Conceptualizations* of a *Human agent* (e.g., *Assessing a hypothesis*, *Diagnosing a car's breakdown*).

A *Conceptualization* is defined in the core ontology of I&DA. It is a mean by which agents can reason about a world. They are expressed in the form of *Expressions* and are physically realized by the *Inscriptions.* An *Inscription is* a knowledge form (e.g., printed texts, pictures) materialized by a substance (e.g., some ink, an electronic field) and inscribed on a Support, i.e. a material object (e.g., paper, hard disk, ambient air in the case where a text is read). An *Expression* is a non-physical knowledge form expressed in a communication code and for which an agent assigns some meaning. Among *Conceptualizations*, a functional distinction is made between *Propositions*, which are descriptions of situations, and *Concepts*, which allow for classifying entities in a world.

We define the action of evaluation of the crucial knowledge as a *Conceptual Human Action* which is an evaluation bearing on *knowledge produced during an activity* and knowledge *used during an activity*, and having for agent a *Decision maker*. This action of knowledge assessment is decomposed into several sub-actions (*Assessing knowledge scarcity*; *Assessing knowledge accessibility*, and so on) consisting in the evaluation of crucial knowledge on different criteria leading to transform the knowledge state of a decision maker.

The action of evaluation of crucial knowledge has for specific data a given knowledge to assess (for example, knowledge relative to material of filter support) which is a *Proposition* in accordance with the principle of modeling of I&DA. The ways of participation to an action are defined in the core ontology of *Participant roles* by introducing specialized relations of participation in the sense of DOLCE: only *Endurants* (i.e., an entity "enduring in time" as a pen, a car company, some water, human rights) participate in *Perdurants* (an entity which "happens in time" as the Olympics games, your reading of this article) and, furthermore, any *Endurant* participates necessarily to a *Perdurant*. For example, a *knowledge to assess* is a *Knowledge used during an activity* and an *Assessing Data*. The roles of *Assessing Data* and *Assessing Result* specialize classes of the *Data* and *Result* roles (figure 4).

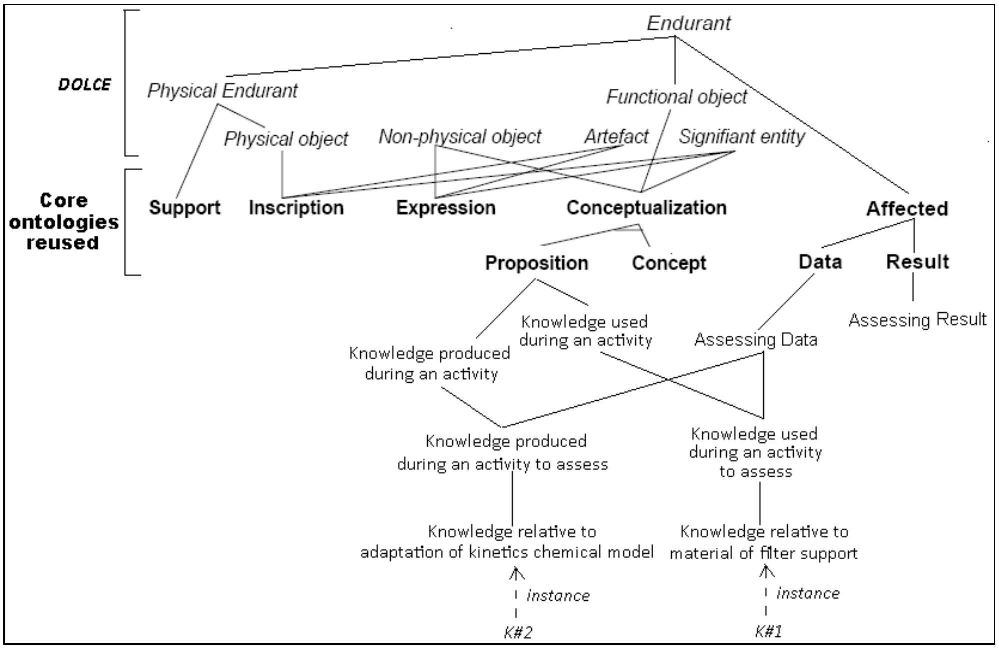

Figure 4. An excerpt of the ontology relative to the evaluation of crucial knowledge

The notion of *Agent* akins to a way for an *Endurant* to participate temporally in an *Action*. We specialize the role of *Agent* defined in the core ontology of *Participant Roles* to define the different contextual roles played by the members of the organization: the *Knowledge provider*, the *Project manager*, the *Decision maker*, the *Knowledge manager* and the *Analyst* (figure 5). For example, a *Decision maker* plays the role of agent in the action of *Making a decision*. It is important to mention that the same person may have different roles. For instance, a *Decision maker* may be also a *Knowledge provider* and/or a *Project manager*.

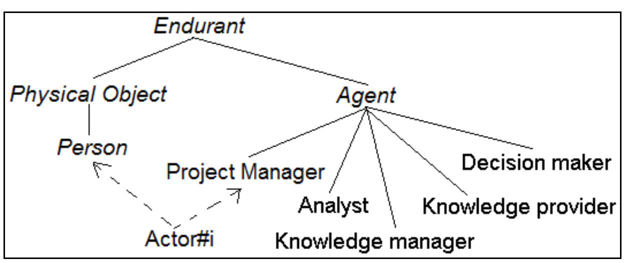

Figure 5. The sub-ontology of the persons concerned by the knowledge identification decision process; Actor#i *have for type* Person and *plays the role of* ProjectManager

V. CONCLUSION AND FUTURE WORK

In this paper, we have done the first phase of the construction of an ontology covering the domain of the crucial knowledge identification. We have exploited the ontological categories existing in the literature to precise the definitions of the UML classes by proposing a coherent modeling framework linking the notions of Knowledge, Actor, Support and Criteria. In particular, we expressed that knowledge produced or used during an activity in the context of the car company is a *Proposition* participating as data and result in the evaluation of the cruciality of knowledge which is realized by a decision maker - an *Agent*.

Our work is currently continued to formalize the knowledge that we have defined in this article, the relations and the associated semantics. We aim to integrate into the system K-DSS the ontology of the domain of the crucial knowledge assessment to automatically reasoning from only a sample of crucial knowledge.

For example, assessing a knowledge *K* based on the complexity criteria is to study the number and the degree of dependency between the knowledge needed to maintain K. If the complexity of K is important, then K requires knowledge of at least four other knowledge, i.e., the different expertise or domain of knowledge of different businesses, used by an actor in a given activity. This process of evaluation on the complexity criteria is based on the tacit knowledge of decision makers. The idea is to make explicit such dependences (in the form of properties or relations in the ontology) between the crucial knowledge to enable the K-DSS system to automatically deduct the cruciality of a knowledge *Ki* knowing the cruciality of the knowledge *Kij*.

Finally, in order to optimize the collaboration between the system K-DSS and the end-users, and to use the system remotely, we will propose new features based on the principles of the technologies related to the Semantic Web (ontologies, reasoning, Web services, etc.). We will use the ontology editor Protege-2000, developed at Stanford University [30], which benefits of the development of “plug-in” for the languages RDF, DAML + OIL and OWL to specify an ontology in different languages on the Semantic Web. In terms of implementation, the plug-in JessTab integrated into Protege-2000, allows to introduce the knowledge stored by Protege-2000 in a database of facts for the application of rules by the inference engine Jess, to the instances of the ontology and the ontology itself (meta-reasoning).

REFERENCES

[1] J. Arpirez, O. Cororcho, M. Fernandez-Lopez, and A. Gómez-Pérez, WebODE in a nutshell, AI Magazine, 24(3), pp 37-48, 2003.

[2] N. Aussenac, Conception d'une méthodologie et d'un outil d'acquisition des connaissances expertes. Thèse de Doctorat en informatique, Université P. Sabatier, Toulouse, Octobre 1989.

[3] N. Aussenac-Gilles, B. Biébow and S. Szulman, D'une méthode à un guide pratique de modélisation de connaissances à partir de textes. In F. Rousselot, éditeur, Actes des 5e rencontres Terminologie et IA (TIA 2003), pages 41–53, Avril 2003.

[4] B. Bachimont, 2004. Art et sciences du numérique : ingénierie des connaissances et critique de la raison computationnelle. Mémoire d'habilitation à diriger des recherches, Université Technologique de Compiègne.

[5] S. Bechhofer, I. Horrocks, C. Goble, and R. Stevens, OilEd: a Reasonable Ontology Editor for the Semantic Web, In Proceedings of the Joint German/Austrian Conference on Artificial Intelligence (KI'2001), volume 2174, pp 396–408, Springer-Verlag LNAI, 2001.

[6] E. Bottazzi, and R. Ferrario, "Preliminaries to a DOLCE Ontology of Organizations", In International Journal of Business Process Integration and Management, Special Issue on Vocabularies, Ontologies and Business Rules for Enterprise Modeling. C. Atkinson, E. Kendall, G. Wagner, G. Guizzardi, M. Spies (Eds.), 2008.

[7] J. Breuker and B. Wielinga. KADS : Structured Knowledge Acquisition for Expert Systems. In Actes des 5èmes journées internationales "Les systèmes experts et leurs applications", Avignon, France, 1985.

[8] S. Bruaux S. and I. Saad, Improving semantic in the decision support system K-DSS. In Proceedings of the International Conference on Information, Process, and Knowledge Management. IEEE Computer Society Press p.p. 66-71, Cancun (Mexico), 1-7 February 2009.

[9] S. Bruaux, G. Kassel, and G. Morel, "A clarification of the ontological status of knowledge roles", In Proceedings of the Workshop on Advances in Conceptual Knowledge Engineering, co-located with the 18th International Conference on Database and Expert Systems Applications, Germany, Septembers 2007.

[10] T. H. Davenport, and L. Prusak, "Working Knowledge: How Organisations Manage What They Know" *Harvard Business School Press*, Boston, MA, 1998.

[11] R. Dieng, O. Corby, A. Giboin, and M. Rybière, Methods and tools for corporate knowledge management, Rapport technique, INRIA, projet ACACIA, Sofia, 1998.

[12] R. Ferrario and A. Oltramari, "Towards a Computational Ontology of Mind", Formal Ontology in Information Systems, Proceedings of the International Conference FOIS 2004, A.C. Varzi, and L. Vieu (Eds.), IOS Press Amsterdam, November 2004, pp. 287-297.

[13] M. Fernandez, A. Gomez-Perez, and N. Juristo, METHONTOLOGY: From Ontological Art Towards Ontological Engineering. AAAI-97 Spring Symposium on OntologicalEngineering, Stanford University, March 24-26th, 1997.

[14] J.Y. Fortier and G. Kassel, "Managing Knowledge at the Information Level: an Ontological Approach", In Proceedings of the ECAI'2004 Workshop on Knowledge Management and Organizational Memories, Valencia (Spain), August 2004, pp. 39-45.

[15] Fürst F., 2006. L'opérationalisation des ontologies : une méthodologie et son application au modèle des Graphes Conceptuels. In Journal électronique d'intelligence artificielle, Vol. 5, number 38, 2006.

[16] A. Gangemi and P. Mika, "Understanding the Semantic Web through Descriptions and Situations", Proceedings of the International Conference on Ontologies, Databases and Applications of Semantics (ODBASE 2003), R. Meersman, and al. (Eds.), Catania (Italy), November 2003.

[17] T.-R. Grüber. Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In Nicola Guarino et Roberto Poly (eds.), editor, Proceedings of the International Workshop on Formal Ontologies, Padova, Italy, 1993. Kluwer Academic Publishers.

[18] M. Grundstein, C. Rosenthal-Sabroux and A. Pachulski, Reinforcing Decision Aid by Capitalizing on Company's Knowledge, European Journal of Operational Research, 145, pp. 256-272, 2003.

[19] M. Gruninger and M. S. Fox, Methodology for the design and evaluation of ontologies. In Proceedings of the Workshop on Basic Ontological Issues on Knowledge Sharing, IJCAI'95, 1995.

[20] N. Guarino and C. Welty, A formal ontology of properties. In R. Dieng et O. Corby, éditeurs, 12th International Conference in Knowledge Engineering and Knowledge Management (EKAW'00), pages 97–112. Springer Verlag, 2000.

[21] N. Guarino and P. Giaretta, Ontologies and knowledge bases: towards a terminological clarification. In N. Mars (Ed.), Towards very large knowledge bases (p. 25 - 32). Amsterdam IOS Press, 1995, http:// ontology.ip.rm.cnr.it/Papers/KBKS95.pdf.

[22] J. Kahan, M. Koivunen, E. Prud'Hommeaux, and R. Swick, Annotea: An Open RDF Infrastructure for Shared Web Annotations, In Proceedings of the 10th International World Wide Web Conference, pp 623-632, 2001.

[23] G. Kassel. Le projet AIDE : une contribution aux systèmes experts de seconde génération. Mémoire d'Habilitation à Diriger des Recherches, Dauphine, 1995.

[24] G. Kassel, P. Lando, A. Lapujade, and F. Fürst, "Des Artefacts aux Programmes", In Proceedings of the 1ères Journées Francophones sur les Ontologies : JFO 2007, Sousse (Tunisia), 18-20 October 2007, pp. 281-300.

[25] G. Kassel, Integration of the dolce top-level ontology into the ontospec methodology, 2005. LaRIA Research Report 2005-08, Université de Picardie Jules Verne. Available at <http ://hal.ccsd.cnrs.fr/ccsd-00012203>.

[26] D. Kayser, La représentation des connaissances, Collection informatique, Hermès, Paris, 1997.

[27] DB. Lena and RV. Guha, Building Large Knowledge-based Systems: Representation and Inference in the Cyc Project. Addison-Wesley, Boston, Massachusetts, 1990.

[28] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, A. Oltramari, and L. Schneider, "The WonderWeb Library of Foundational Ontologies and the DOLCE ontology", WonderWeb Deliverable D18, Final report (vr 1.0, 31-12-2003), 2003.

[29] F. N. Noy and D.L. McGuinness, Ontology Development 101: A Guide to Creating Your First Ontology, Technical Report SMI-2001-0880, Stanford Medical Informatics,Stanford University, Stanford, CA , USA.

[30] N. Noy, R. Fergerson, and M. Musen, The knowledge model of Protégé-2000: Combining interoperability and flexibility. In R. Dieng and O. Corby, editors, Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management: Methods, Models, and Tools (EKAW 2000), volume 1937 of Lecture Notes in Artificial Intelligence (LNAI), pages 17–32, Juan-les-Pins, France, 2000. Springer.

[31] R. Neches, R. Fikes, T. Finin, T. Gruber, R. Patil, T. Senator and W.-R. Swartout, Enabling technologies for knowledge sharing. In AI Magazine, 12(3):36--56, Fall 1991.

[32] R. Volz, D. Oberle, S. Staab, and B. Motik: KAON SERVER - A Semantic Web Management System. WWW (Alternate Paper Tracks) 2003

[33] I. Saad, Une contribution méthodologique pour l'aide à l'identification et l'évaluation des connaissances nécessitant une opération de capitalisation. Ph.D. thesis, Université Paris-Dauphine, Paris, France, 2005.

[34] I. Saad and S. Chakhar, A decision support system for identifying crucial knowledge requiring capitalizing operation, European Journal of Operational Research (EJOR),Volume 195, n° 3, pp. 889-904, June 2009.

[35] G. Schreiber, B.J. Wielinga, H. Akkermans, W. Van de Velde, and A. Anjewierden. CML: The CommonKADS Conceptual Modelling Language. In Proceedings of the 8th European Knowledge Acquisition Workshop: EKAW'94, Springer-Verlag, 1994, p. 283-300.

[36] Sowa J., Conceptual structures : information processing in mind and machine, Addison-Wesley, 1984.

[37] S. Staab, H.-P Schnurr, R. Studer, and Y. Sure, Knowledge processes and ontologies. IEEE Intelligent Systems, Special Issue on Knowledge Management, 16(1). Staab S, Schnurr HP, Studer R, Sure Y (2001) Knowledge Processes and Ontologies. IEEE Intelligent Systems 16 (1): 26-34.

[38] L. Steel, Corporate knowledge management, Proceedings of the International Symposium on the Management of industrial and corporate knowledge (ICMICK'93), Compiègne, octobre 1993, pp.9-30.

[39] Y. Sure, S. Staab, and J. Angele, OntoEdit: Guiding ontology development by methodology and inferencing. In Proceedings of the International Conference on Ontologies, Databases and Applications of SEmantics ODBASE 2002, University of California, Irvine, USA, 2002.

[40] B. Swartout, R. Patil, K. Knight and T. Russ, Towards Distributed Use of Large-Scale Ontologies. Spring Symposium Series on Ontological Engineering, pp.138-148.

[41] B. Tseng and C. Huang, "Capitalizing on Knowledge: A Novel Approach to Crucial Knowledge Determination," IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans, Volume 35, Issue 6, 919-931, 2005.

[42] M. Uschold and M. King, "Towards a methodology for building ontologies" In Workshop on Basic Ontological Issues in Knowledge Sharing: International Joint Conference on Artificial Intelligence. (Also available as AIAI-TR-183 from AIAI, The University of Edinburgh.), 1995.

[43] M. Uschold and M. Grüninger. Ontologies: Principles, Methods and Applications. In Journal of Knowledge Engineering Review, 11(2), 1996.

[44] B. Wielinga, J. Benjamin, W. Jansweijer, G. Schreiber, E. Meis, G. Willumsen, J. Eggen, P. Gobinet, N. Modiano, A. Bemaras, I. Laresgoiti, and F. Persson, Principles and Guidelines for Domain Ontology Library Design, v. 2, ESPRIT Project 8145 KACTUS, deliverable DO5a.2, 1996.