Robustness of know-how within a Product Lifecycle Management system

Mathieu Barcikowski* — Philippe Pernelle* —
Arnaud Lefebvre* — Michel Martinez*

** Laboratoire PRISMA - Université Lyon 1*
17 rue de France, F-69 Villeurbanne cedex
mathieu@sfr.fr, {philippe.pernelle, arnaud.lefebvre, michel.martinez}@iutb.univ-lyon1.fr

ABSTRACT. *Product Lifecycle Management systems have vocation to capitalize know-how around the development of manufactured goods. Knowledge is difficult to formalize. Therefore it seems necessary to provide the tools for the evaluation of its quality. This paper propose an approach to evaluate the robustness of knowledge formalized within a Product Lifecycle Management (PLM) system. Thus we propose to evaluate various qualitative and quantitative criteria on knowledge formalized with conceptual graphs. These criteria are based on iterative process of validation of expert and on the management of the traceability of a PLM system (temporal evolution, frequency of use, rate of modification. We thus seek to obtain a robust knowledge which allows whatever the context of use to maintain the quality of the provided answers which satisfy criteria of exhaustiveness, coherence, consistency and precision. After the presentation of these various criteria, we propose a software prototype coupled with a PLM as well as a step of validation around knowledge on a manufacturing process.*

RÉSUMÉ. *Les Systèmes d'Information Produit (ou PLM) ont vocation à capitaliser le savoir-faire et la connaissance métier autour du développement de produits manufacturés. Cette connaissance étant difficile à formaliser, il semble nécessaire de fournir les outils pour son évaluation en terme de qualité. Cet article propose une approche pour évaluer la robustesse des connaissances formalisées au sein d'un PLM. Pour cela nous proposons d'évaluer différents critères qualitatifs et quantitatifs sur des connaissances formalisées avec des graphes conceptuels. Ces critères s'appuient sur des un processus itératif de validation d'expert et sur la gestion de la traçabilité d'un système PDM (évolution temporelle, fréquence d'utilisation, taux de modification …). Nous cherchons ainsi à obtenir une base de connaissance robuste qui permet quel que soit le contexte d'utilisation de maintenir la qualité des réponses fournies qui satisfassent des critères d'exhaustivité, de cohérence, de consistance et de précision. En plus de la présentation des différents critères, nous proposons un prototype logiciel couplé avec un PDM ainsi qu'une démarche de validation autour de connaissances sur un procédé de fabrication.*

KEYWORDS: *Product Information System, PLM/PDM, Knowledge*

MOTS-CLÉS : *Système d'Information Produit, PLM / PDM, connaissance métier, robustesse.*

1. Introduction

The capitalization of knowledge is a crucial stake for the companies which want to ensure their perenniality and their capacity of innovation. For that, the knowledge engineering provides a set of tools and methods meeting this need for capitalization. We wish to expose the problem of the relevance of knowledge through this paper. Is it exploitable for fields or a context different from those where it was formalized? In order to answer this question, we will introduce the concept of robustness of knowledge and will show how to evaluate this robustness while being based on a PLM system. PLM Systems (Product Lifecycle Management) are in theory able to deal with the totality of the process of development of the product. The strategic choice to center its Information System on a PLM can be justified by the reduction in the costs and of the deadlines. But a PLM remains finally a system of capitalization and perpetuation of industrial know-how. The interest of a PLM system to evaluate the robustness of knowledge is double: the first interest relates to its capacity of modeling. Indeed, the placement of Information System around a PLM requires a work of modeling. However these models are compatible with some formalism used in knowledge engineering. The second interest relates to the capacities of a PLM system to manage the traceability of the access, modification and control around knowledge.

In the first part, we define the concept of robustness applied to knowledge and various criteria to evaluate this robustness. In the second part, we detail the elements of modeling within a PLM to integrate a particular type of know-how. Finally in the last part, we present the mechanisms of evaluation of knowledge within the PLM system.

2. Concept of robustness of knowledge

2.1. *Knowledge within a PLM system*

Knowledge handled within the company is varied and complex. Its classification depends on the point of view of the actors. We will make a usually allowed distinction which separates explicit knowledge and tacit knowledge. Explicit knowledge is dedicated to a particular problem and relates to the heuristic ones which reflect the experiment of the experts. It’s sometimes difficult to reach and its mode of formalization always does not meet the expressed need. Tacit knowledge (Grundstein, 2001) corresponds to know-how, formalizables more or less easily. Knowledge can be obtained mainly in two manners. Methodologies (MASK (Ermine, 2001)...) propose to use the experts of the company as source of knowledge. But it is also possible to obtain knowledge starting from documents resulting from the activity of the company (Bourigault *et al.,* 1996) (Assidi, 1998) (Biébow *et al.*, 1999).

Knowledge is modelled starting from a point of view on the field of knowledge at a given moment and according to a methodology. So a divergence between the field of knowledge and its model (the knowledge base) can appear.

The capitalization and the perpetuation of knowledge can be difficult tasks because the evolutions of technologies, know-how, environment, manpower and product must be taken into account. This difficulty is reinforced by the constraints due to the access, the selection and the modeling of crucial knowledge (Grundstein, 1995) of the company. It would be interesting to know in which proportions knowledge resists the evolutions of the field, i.e. with the technical, social and economic evolutions. It would be also interesting to know how knowledge, once modelled, resists the divergences due to the problems of selection of crucial knowledge.

The evaluation of robustness of knowledge vis-a-vis to these evolutions and these difficulties of modeling would make it possible to consider risks taken by the people using knowledge modelled within a PLM. One can also imagine a better anticipation and detection of the problems relating to the use and the update of knowledge of a PLM and thus even improve to reduce the needs for maintenance. The maximization of the level of robustness of knowledge would make it possible to increase the confidence which one can have during the use of this knowledge within the PLM.

2.2. *Notion of robustness of knowledge*

The robustness is a concept used in many fields (automatic, system...) which can be adapted to knowledge by analogy. Intuitively, one can say that the more robust one knowledge will be and the more it could be used in a different or dubious context compared to the initial context where it was created. More formally, we approach this definition according to two points of view:

– The temporal point of view makes it possible to take into account the divergences between the knowledge modelled at a given moment and the natural evolutions of the field,

– The contextual point of view makes it possible to take into account the divergence between the base of knowledge and the field because of the errors of modeling.

2.2.1. *Contextual robustness*

We use a knowledge base exploited through inference engine. We suppose here that the inference engine is of sufficient quality. Thus, the quality of the provided answers is dependent on the quality of the knowledge base (Groot *et al.*, 2000). So a base of knowledge is robust if it makes it possible to maintain the quality of the provided answers, some is the context of use.

2.2.2. *Temporal robustness*

One studies here the capacity at the knowledge base to resist the evolutions of the field. The field of knowledge to model within the knowledge base is depend on the evolution of the real world. The evolutions of the technical, social and economic environment can call into question the current knowledge and know-how. Ideally, a modification of the field should result in a modification of the knowledge base. This dynamics can be difficult to set up because it can be difficult to have access and to detect these evolutions.

A robust knowledge base makes it possible to limit the effects of the modifications of the field. It can preserve a sufficient quality despite of the evolutions of the field and without maintenance of the knowledge base.

2.3. ***Evaluation of the robustness of knowledge***

The robustness of knowledge is dependent from the point of view approached. For that, we wish to define and set up a certain number of criteria of evaluation organized according to the level of abstraction. Each criterion evaluates an aspect of the robustness then.

2.3.1. *Criteria dependent on the level information*

The criteria dependent on the level information are evaluated and are based on the activity around the knowledge based system (KBS).

The level information being dense, it is easier to define criteria. In the other hand, these criteria do not make it possible to define the robustness of knowledge directly, but simply to identify the evolution or the localization of the zones of robustness.

The evaluation is carried out starting from the histories of use of the systems related to the knowledge base. It is a question of obtaining traces on the use of the knowledge base and of calculating statistics.

2.3.2. *Criteria dependent on the level knowledge*

The level knowledge asks for a higher level of abstraction. Knowledge necessary for the evaluation of the criteria of robustness of this level can be provided by an expert. Knowledge thus obtained forms a model partial of the field which it will be a question of comparing with that contained in the knowledge base.

The criteria of this level make it possible to obtain an evaluation based on the meaning with contrario of the criteria of level information which offer a statistical evaluation. However the evaluation of these criteria can be less controllable because of intervention of the expert. Moreover considering the possible size of a knowledge base, it would be illusory to entrust to only one expert the task to evaluate the whole of the criteria, especially if this task must be often repeated

It would thus be necessary to choose several experts for this task of evaluation, which poses problems of methodology for the choice of the experts and the distribution of the work of evaluation between them.

2.3.3. *Criteria dependent on the level meta-knowledge*

This meta-knowledge can be stated is forms of rules to check in an automatic way the knowledge base. They can be provided by the experts who have enough retreat on their field to provide this type of rule.

The analysis on the level meta-knowledge makes it possible to evaluate robustness in the automatic and systematic way on the whole knowledge base. However obtaining the rules can be difficult considering the level of knowledge which they ask on the field.

The analysis of the robustness with these rules consists in analyzing the knowledge base how complies with these rules (example: frequency of errors or frequencies of error compared to a given rule).

3. Integration of knowledge within a PLM

In this paragraph, we will specify how knowledge is integrated into a Product Lifecycle Management system.

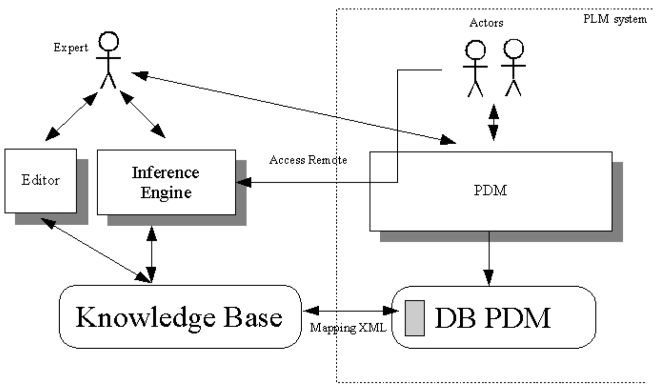

Figure 1. ***PLM and Knowledge Tools***

3.1. *Procduct/Process/Organization Model*

The definition of Product/Process/Organization models constitutes the last stage before the deployment of a PLM solution. The choice in the various formalisms of representation depends on complexity and the activity of company (STEP, IDEF, SADT, UML...). This choice remains however limited to the capacities of the data-processing platform selected. In this article, we will use integrated a Product/Process/Organization model. This model is based on:

– A description of product with meta-article (entity, link, state …)

– A description of processes of the workflow supplemented with specific objectives and operators.

– A description of the organizations by using functional units.

We will in addition use a integrated modeling language built on XML (extensible Markup Language). This step makes it possible to represent and structure all the elements (article, nomenclature, activity, organization, etc.) with a single formalism. This language (Product Information System Markup Language) (Pernelle, 2002) (Pernelle, 2003) described, on the one hand generic objects allowing to conceive an information system PLM, other instances of objects circulating in the PLM. We will define with this language the elements for formalization of knowledge.

3.2. *Formalization of knowledge within the PLM*

3.2.1. *Conceptual graphs*

Knowledge will be modelled using the conceptual graphs (Sowa, 1984). The model of the conceptual graphs is a model of knowledge representation based on the existential graphs of C PEIRCE (Peirce, 1933) and the semantic networks. A conceptual graph is a graph composed of two types of nodes which are respectively of concept type and relation type.

The model of the conceptual graphs is defined formally by means of an abstract syntax which allows the representation of the graphs according to various notations. The Display Form makes it possible to the users to understand and modify the conceptual graphs more easily than with a representation in the form of logical formulas for example. It is also possible to represent conceptual graphs by using XML such as the editor of conceptual graphs CharGer uses it (Delugach, 2001).

Figure 2 represents the sentence "John takes the bus in direction of Boston" by a conceptual graph in its display form.

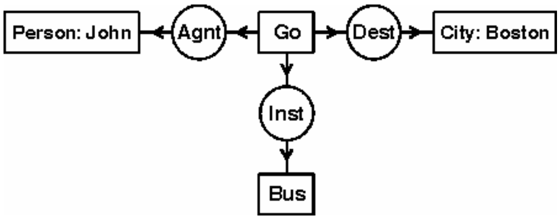

Figure 2. ***Example of conceptual graph***

3.2.2. Modeling

In our model PPO, the elements will be to model in the following way:

Element	***Description***	***PPO Type***
Graph	Set of concepts and relation	Entity
Concept	Identifiable representation of an element of a field	Entity
Relation	Typed set binding concepts and graphs	Link
Taxonomy	Set of relations of specification between concept	Entity
Expert	Actor having the expertise to create and validate concepts	Organisation

Figure 3. ***Elements of knowledge of model PPO***

From these elements we identify formalization compatible with the PLM and described in PisML. Following files XML (Fig. 4,5,6) presents various modelings associated with the elements of knowledge modélisables:

```
<!DOCTYPE Entity SYSTEM "Entity.dtd">
<ClassItem>
  <ClassObject idname="Graph" level="">
    <Items>
```

```
      <Item idname="nom" required="yes" formatted="NONE">
      <Item idname="type" required="yes" formatted="NONE">
      </Item>
    </Items>
  </ClassObject>
</ClassItem>

<ClassItem>
    <ClassLink idname="CompositionGraph" lib="">
       <Inputs>
          <Library>
             <Out>
                <PismlRef pref="PISREF:ClassItem//.@./Graph()"/>
                <PismlRef pref="PISREF:ClassItem//.@./Concept()"/>
                <PismlRef pref="PISREF:ClassItem//.@./Relation()"/>
             </Out>
          </Library>
       </Inputs>
    </ClassLink
</ ClassItem >
```

Figure 4. ***Extract from entity GRAPH***

```
<!DOCTYPE Entity SYSTEM "Entity.dtd">
<ClassItem>
  <ClassObject idname="Concept" level="">
    <Items>
      <Item idname="Nom" required="yes" formatted="NONE">
        <lib>Concept label</lib>
      </Item>
    </Items>
  </ClassObject>
</ClassItem>
```

Figure 5. ***Extract from entity CONCEPT***

```
<!DOCTYPE Entite SYSTEM "Entite.dtd">
<ClassItem>
    <ClassLink idname="Relation" lib=" Relation ">
       <Inputs>
          <Input card="1" required="YES">
```

```
      <In>
        <PismlRef pref="PISREF:ClassItem//.@./ Concept ()"/>
      </In>
    </Input>
    <Library>
      <Out>
        <PismlRef pref="PISREF:ClassItem//.@./Concept()"/>
      </Out>
      </ Library >
```

Figure 6. ***Extract from link RELATION***

3.2.3. *Traceability management*

The traceability management makes it possible to preserve the activities history as well as the whole of the modifications. This point is essential to evaluate the temporal robustness. Indeed, it is starting from the access and modification monitoring that we will calculate the criteria of temporal robustness. For that we define states and states spaces. A State is an entity which characterizes the status of a technical object. For example, a document will be able to have several valid revisions, with for each revision, one dégré of maturity different. This diversity, leads us to define a state as a whole of state vectors. A vector consists of conditions of entry and transitions towards other states.

Thus, the possibility of defining these states spaces it possible each company versionnement to determine the statutes necessary to these processes (, maturity, and Classically we identivions a space on related to the versionnemnt objects. Moreover we can define a state related to the maturity of the concept.

```
<!DOCTYPE Entity SYSTEM "Entity.dtd">
<ClassItem>
 <ClassStateSpace id="Maturite" >
    <Vectors>
      <Vector id="preliminaire">
        <On>
          <PismlRef pref="PISREF:ClassItem//.@./ Concept ()"/>
          <EntryCondition>
                 <IsExist on="INSTANCE">
              <PismlRef pref="PISREF:ClassItem//.@./ Concept ()"/>
                 </IsExist>
          </EntryCondition>
          <Transitions>
           <PismlRef pref="PISREF: ClassStateSpace //.@./maturite(valide)"/>
          </Transitions>
```

```
    </On>
   </Vector>
…
```

Figure 7. ***Extract from State Space : MATURITE***

We define the monitoring of the knowledge elements which we wish observes. From this observation we will be able to evaluate a set of robustness criteria

4. Evaluation of the robustness

4.1. *Criterion of robustness*

The level of robustness of knowledge is not evaluated in a global but it is evaluated by a set of robustness criteria. Each one of these criteria makes it possible to describe an aspect of the robustness of knowledge of the PLM. It is thus necessary to take into account each one of these criteria to have a global vision of the robustness of knowledge. Among these criteria, we propose the criteria of robustness related to the activity of update of the knowledge base. These criteria will be clarified in a more concrete way through the example below.

The criteria related to the activity of update of the knowledge base make it possible to observe the stability of the knowledge base in the course of time. They allow observing zones of the knowledge base which are stabilized because of the maturity reached by knowledge which contain. It also makes it possible to observe zones which were not updated and which requires possibly a modification because of evolution of the field. They finally make it possible to put forward the zones of the base of knowledge which requires a monitoring deepened because of the great activity related to these zones.

These criteria are placed in temporal dimension. Indeed, they make it possible to evaluate the quantity of modification operated in the course of time, and thus indicate the temporal evolution of the knowledge base. They are also placed in dimension information because it is not possible via these criteria to deduce from the activity of modification the incidences from the semantic point of view between the field and the knowledge base.

The evaluation of these criteria is carried out starting from the traces of the activity maintenance generated by the knowledge base editor and captured by PLM system. These criteria are evaluated for each element of the knowledge base. For each element, the sum of the quantity of modifications operated on the elements makes it possible to define the stability of these elements.

The goal of the algorithm presented in figure 8 is to evaluate in a local way the criteria of update of the knowledge base. For this, we have a list of elements to treat coming from the knowledge base LV, the trace of the actions carried out on the knowledge base LA. The first part of the algorithm makes it possible to calculate the raw value of the criteria. The second part makes it possible to standardize the values obtained by real numbers ranging between 0 and 1. The elements of the knowledge base contain their identifier like their value for the criteria of robustness, i.e. the addition, the modification and the removal of elements. The actions are described by their type and the element on which they acted.

```
LV  is the list of the elements of the knowledge base to be treated
LA is the list of the actions carried out the knowledge base
First indicate if the first element is treated
Min is minimum of action carried out of the studied type
Max is maximum of action carried out of the studied type
TypeAction is the type of action carried out on the element (addition, suppression,
modification)
First := True
For Each NodeSel in LV Do
| For Each ActionSel in LA Do
| | If ActionSel.IdRef = NodeSel.Id Then
| | | If ActionSel.Type = TypeAction Then
| | | | NodeSel.TypeAction := NodeSel.TypeAction+1
| | | EndIf
| | EndIf
| EndDo
| If First Then
| | Min := NodeSel.TypeAction
| | Max := NodeSel.TypeAction
| | First := False
| Else
| | Min := Minimum(NodeSel.TypeAction,Min) ;
| | Max := Maximum(NodeSel.TypeAction,Max) ;
| EndIf
EndDo

For Each NodeSel in LV Do
| NodeSel.TypeAction:= (NodeSel.TypeAction - Min) / (Max-Min);
EndDo
```

Figure 8. ***Algorithm of evaluation of the local criterion of quantity of addition***

This algorithm consists in for each element of the knowledge base treating, to traverse the list of the actions carried out to count the number of actions of addition, suppression or modification which were carried out on the treated element of the knowledge base. Once the number of actions carried out on the entered element, one evaluates the local value of the various criteria of the elements of the knowledge base in order to preserve smallest and the greatest value of these criteria among all the treated elements of the knowledge base.

Then each element of the knowledge base to be treated is standardized: the value of the element is modified according to the maximum and minimum value criteria of addition, modification and suppression in order to obtain values ranging between 0 and 1 for each element of the knowledge base treated.

4.2. ***Example***

The example suggested relates to the modeling of the cylinder head camshaft bearing line. Figure 9 presents a cylinder head. Camshaft bearing line is made up of two types of product entities: a *camshaft bearing* feature and a *seal housing* feature.

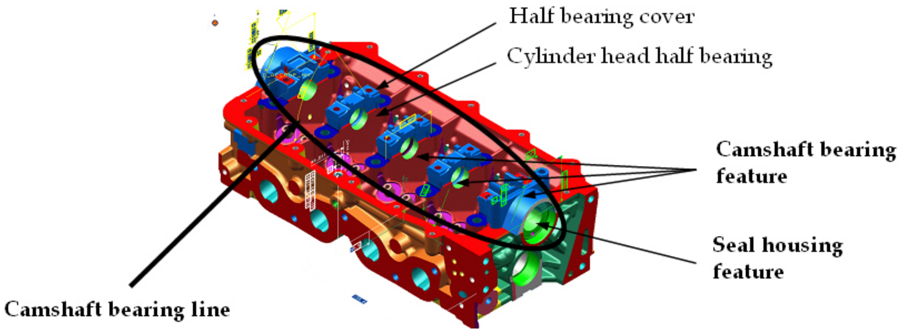

Figure 9. ***View from the top of assembled cylinder head***

The model of conceptual graph of figure 9 results from a know-how relating to the machining of a *Camshaft bearing* feature. This feature is characterized by *product* data identified by the Camshaft_bearing_PRODUCT graph and of the *process* data identified by the Camshaft_bearing_PROCESS graph.

For a *Camshaft bearing* feature, the parameter *process number operations* strongly depends on the following parameters produced:

– Tolerance on the diameter,

– Roughness,

– Coaxiality,

– Localisation,

– Cylindricity,

The model conceptual graph of figure 10 retranscribed the relation between these parameters by the intermediary of the relation *link*.

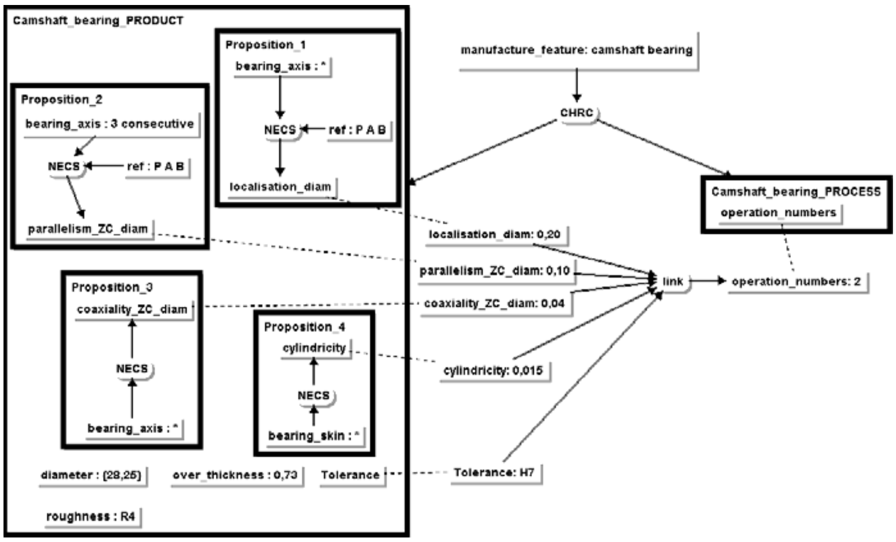

Figure 10. ***Conceptual Graph modelized at moment t***

The expertise having evolved in time, we replace the static relation *link* connecting the parameters *product* and *process* (Fig. 11) by a dynamic relation which characterizes a logical process and which represents the function of know-how G exit of a design of experiments for example.

Figure 11 presents the expertise machining model of the *camshaft bearing* feature. As we integrated the major knowledge represented by the function G into the model of figure 10, the model becomes generic then and the markers of the concepts become nondefined.

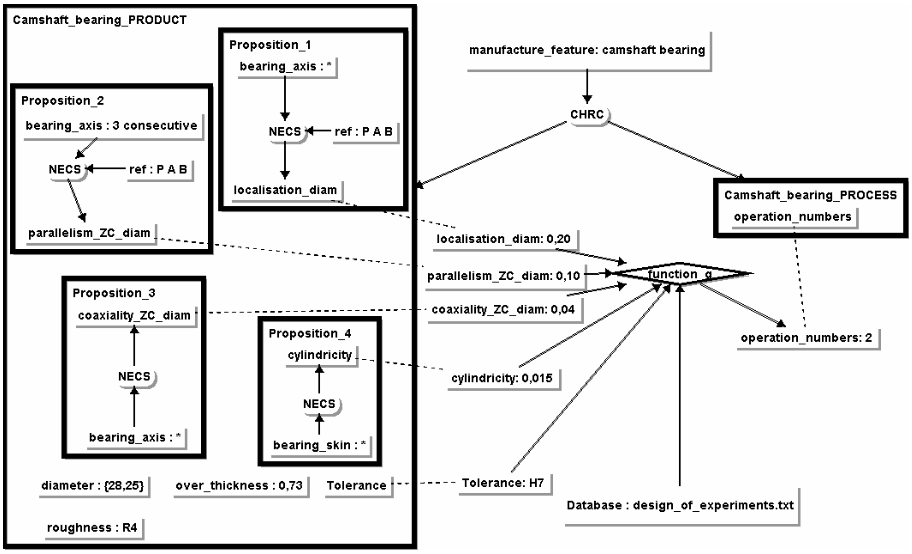

Figure 11. ***Conceptual Graph modelized at moment t+1***

Figure 11 presents the list of the modifications which were made between the two versions of the conceptual graph. PLM System maintains up to date this list via its system of management of the traceability.

Graph Element	Type of action	Detailed description	Date of detection of the change
Relation « link »	addition	Addition of a concept « Database » to the relation « link »	t + 1
Relation « link »	modification	Modification of the type of relation of link towards fonction_g	t + 1

Figure 12. ***List modifications made to the conceptual graph***

By the use of the algorithm of figure 8, the values obtained for the criteria of robustness related to the update of the knowledge base is zero for all the elements of the conceptual graph of figure 11, except for the relation « link » which obtains a score of 1 for the criterion taking into account the number of addition and 1 for the criterion taking into account of modification.

One can deduce from it that the knowledge base was robust over the period of time between the two versions of the conceptual graph. The least robust element from the point of view of the update is the relation « link » which, relative with the other elements, has undergoes much modification. The difference in quantity of modification between the relation « link » and the other elements explains the variations of robustness between the various elements of the conceptual graph.

The criteria evaluated here do not make it possible to deduce the reasons from this low robustness of the relation « link » and from these consequences on the robustness from the contextual point of view. In order to obtain other information of higher level on the robustness of these elements, we wish to connect the criteria of robustness presented here with other criteria of robustness. These other criteria, which are not presented in this paper, define for example the robustness of knowledge according to the activity related to the use of information of the PLM and according to the activity related to the use of the knowledge base. In conjunction with the criteria related to the activity of update of the knowledge base, it is possible for example to deduce that a nonrobust element from the point of view of the update of the knowledge base didn't have a notable consequence on the total robustness of the knowledge base, had with the fact that the element is not used by the people using system PLM.

5. Conclusion

The capitalization of knowledge improve an optimal exploitation of the technical inheritance of the company. To be effective, it must be based on robustness environement i.e. not very sensitive to not very sensitive to its evolution. The approach proposed is to integrate the knowledge management into a PLM of which the goal is to make it possible to evaluate this robustness. It would be abusive to regard a PLM as a system of knowledge management However we wanted to show that certain knowledge can be formalized and that it's possible to use PDM tools to evaluate the temporal robustness of knowledge. Our next step is to propose the management of the expertise protocol within the PLM processes management.

Bibliographie

Assidi, H., Construction d'ontologies à partir de textes techniques. Applications aux systèmes documentaries, rapport de thhèse, 1998, Paris, Université Pierre et Marie Curie.

Biébow, B. and Szulman S., TERMINAE: A linguistic-based tool for the building of a domain ontology, *Knowledge Acquisition, Modeling and Management, Proceedings of the 11th European Workshop*, 1999, Allemagne, Springer-Verlag.

Bourigault, D. and Lépine P., Utilisation d'un logiciel d'extraction de terminologie (LEXTER) en acquisition des connaissances, *Acquisition et ingénierie des connaissances, tendances actuelles*, 1996, N. Aussenac-Gilles, P. Laublet and C. Reynaud, Editions Cépaduès.

Delugach, H. S., CharGer: A Graphical Conceptual Graph Editor. *Workshop on Conceptual Graphs Tools, 9th International Conference. On Conceptual Structures*, 2001, Stanford University.

Ermine, J.-L., Capitaliser et partager les connaissances avec la méthode MASK. *Ingénierie et capitalisation des connaissances*, 2001, M. ZACKLAD and M. GRUNDSTEIN, Paris, Hermès: 66-105.

Grundstein, M., La capitalisation des connaissances de l'entreprise, système de production des connaissances. *L'entreprise apprenante et les Sciences de la Complexité*, 1995, Aix-en-Provence.

Grundstein, M., De la capitalisation des connaissances au renforcement des compétences dans l'entreprise étendue, *colloque Gestion des Connaissances et des Compétences en Génie Industriel*, 2001, Nantes.

Groot, P., Harmelen F. v., et al., Torture tests: a quantitative analysis for the robustness of Knowledge-Based Systems, *12th International Conference on Knowledge Engineering and Knowledge Management*, 2000, Springer-Verlag.

Peirce, Charles S., "Collected Papers." Edited by Charles Hartshorne and Paul Weiss. Cambridge: Harvard University Press, In Edward Moore and Richard Robin (Eds.), *Studies in the Philosophy of Charles S. Peirce*., 1933, Amherst, University of Massachusetts Press.

Pernelle P., Système d'Information Produit pour les PME/PMI : modélisation multi-niveaux d'entreprises engages dans un travail cooperative, rapport de thèse, décembre 2002, Université de Savoie.

Pernelle P., Courois A., "Pernelle P., Courtois A., 2002. Formalisation des structures organisationnelles pour le partage des informations techniques. Revue Internationale de Génie Industriel et de Production Mécanique.

Sowa, J. F., Conceptual Structures: Information Processing, *Mind and Machine*, 1984, Addison-Wesley, London.