Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 9 (2011) 251–260

Engineering Procedia

TRIZ Future Conference 2009

An ontology for TRIZ

Denis Cavallucci *, François Rousselot, Cécilia Zanni

INSA Graduate School of Science and Technology, 24, Boulevard de la Victoire 67084 Strasbourg Cedex - France

Abstract

This paper discusses the usefulness of an ontology for TRIZ and is an attempt to provide its readers (researchers, industrialists, educators, students) with answers regarding elementary questions they may have concerning TRIZ groundings and potential uses. These answers have the main objective to clarify how they may locate their perceptions of TRIZ and more easily find a way to contribute to its corpuses' evolution. An additional section will discuss about a software implementation of these elements through a software prototype built using this ontology. On a longer scale, we are aiming at sharing this ontology with a broader spectrum of research units in various disciplines, debate on its relevance through experiments with industry and further develop it in a collaborative way. A discussion section will then highlight the assets, the limits and the perspectives of having a shared ontology for TRIZ future developments.

Keywords: TRIZ; Ontology; Software development; Knowledge representation;

1. Introduction

Since it arrives in highly industrialized countries, TRIZ has both fascinated and discouraged its observers. Several reasons are behind this paradox and have encouraged few tempetuous researchers in design to spend many years in finding roots to this problematic. After a decade and a half to study TRIZ and Engineering Design Science findings, we highlighted several incomplete elements in the corpus constituted by the theory [1]. We also noted that most of these elements were tacitly present in experts mind but have never been decomposed in order to be easily understood and taught to newcomers in an efficient manner as noted in [2].

Our postulate is that a first and important way to understand how TRIZ can be useful to engineering Design science is to understand at which point it differs from classical engineering way of understanding design and on which basis a relevant use of TRIZ can be organized.

These observations were both considered from an industrial, academical and educational perspective and have helped us to build a coherent corpus of elements described through an ontology which currently serve several

* Corresponding author.
E-mail address: denis.cavallucci@insa-strasbourg.fr .

doi:10.1016/j.proeng.2011.03.116

research programs and plays a crucial role in moving towards acceptance of Inventive Design concept. Our work is a contribution in the direction of the building and the acceptance of an ontology for TRIZ and proposes a framework aiming at being shared and discussed in various scientific communities.

2. What are ontologies?

In recent years the development of ontologies has been moving from the realm of Artificial-Intelligence laboratories to the desktops of domain experts. Ontologies have become common on the World-Wide Web. The ontologies on the Web range from large taxonomies categorizing Web, to categorizations of products for sale and their features. The WWW Consortium (W3C) has developed RDF - the Resource Description Framework [3], a language for encoding knowledge on Web pages to make it understandable to electronic agents searching for information. The Defence Advanced Research Projects Agency (DARPA), in conjunction with the W3C, has developed the DARPA Agent Mark-up Language (DAML) by extending RDF with more expressive constructs aimed at facilitating agent interaction on the Web [4]. Many disciplines now develop standardized ontologies that domain experts can use to share and annotate information in their fields. Medicine, for example, has produced large, standardized, structured vocabularies.

An ontology defines a common vocabulary for researchers who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them.

Although there is a lack of unanimity in the exact definition of the term *ontology,* it is generally regarded as a formalized representation of the knowledge in a domain taken from a particular perspective or conceptualization. The main use of ontology is to share and communicate knowledge, both between people and between computer systems. A number of generic ontologies have been constructed, each having application across a number of domains, which enables the reuse of knowledge. In this way, a project need not start with a blank sheet of paper, but with a number of skeletal frameworks that can act as predefined structures for the knowledge being acquired. Ontologies also provide guidance to the knowledge engineer regarding the types of knowledge to be investigated [5].

This formal explicit description of concepts in a domain of discourse includes classes called **concepts**, properties of each concept describing various features and attributes of the concept or **slots** (sometimes called **roles** or **properties**), and restrictions on slots , called **facets** (sometimes called **role restrictions**). An ontology together with a set of individual **instances** of concepts constitutes a **knowledge base**. In reality, there is a fine line where the ontology ends and the knowledge base begins [6].

[7] identifies three layers of knowledge, corresponding to three different types of ontologies, based on their levels of generality, namely:

- Generic or foundational ontologies, which capture general, domain independent knowledge (e.g. space, time, etc).
- Domain ontologies, which capture the knowledge in a specific domain (such as automobile industry).
- Application ontologies, which capture the knowledge necessary for a specific application.

3. Why use ontologies?

Knowledge acquisition during the problem formulation phase will lead to the creation of a shared model and, therefore, the model concepts must then be clearly defined. The knowledge acquisition process is carried out without a precise ordering, a parameter will appear perhaps initially without belonging to a contradiction, a contradiction will perhaps appear without mentioning its dependence to a problem, and so on….

It is necessary to clearly define the concepts implemented and the relations that they present, it consists in providing assistance to the detection of possible inconsistencies and eventual missing items. To clarify relations between TRIZ associated concepts is a need; the elaborated model must be sharable, targeting a semantic integration of all useful sources of information. These pieces of information might be extracted either from the speech of an expert (interviews, working sessions…) or captured in texts (patents, list of requirements, norms).

In general, some of the reasons why it is advisable to develop an ontology are:

- to share common understanding of the structure of information among people or software agents,

- to enable reuse of domain knowledge,
- to make domain assumptions explicit,
- to separate domain knowledge from the operational knowledge,
- to analyze domain knowledge.

4. Ontological framework of our works

Application of “foundational” ontologies as a basis for this development has appeared a judicious choice to us, so as to take profit of the justification and formal organization characteristics of this type of ontology.

Only some foundational ontologies have been developed at a satisfactory level in the literature, in particular, DOLCE (the Descriptive Ontology for Linguistic and Cognitive Engineering) [8], GFO (the General Formal Ontology) [9], OCHRE (the Object-Centered High-level Reference Ontology) [8], SUMO (the Suggested Upper Merged Ontology) [10] and BFO (the Basic Formal Ontology) [8].

As foundational ontologies are complex systems, two crucial elements have to be taken into account to choose the right one to work on: the ontology has to provide a rich set of conceptual distinctions regarding the domain application, and all the needed characteristics have to be clearly characterized. We have decided to work with DOLCE, because this ontology provides a certain number of high level concepts that are useful for our application. DOLCE makes a distinction among objects (such as substances) and events, and also provides definitions concerning objects characteristics, such as attributes, values, qualities.

The entire development of our ontology along with its formal characterization can be found at [11].

5. The ontology we have developed

The ontology we present aims to be a domain ontology of TRIZ in specifying its base notions. Current literature review indicates that these base notions are unevenly perceived by TRIZ world. Nevertheless it is unavoidable for a theory to evolve, to ensure that experts have a common and precise understanding of those notions and that the complementary ones are coherent. This ontology will therefore evolve, once we will add new elements of knowledge that appear, more or less explicitly, in acknowledged published sources.

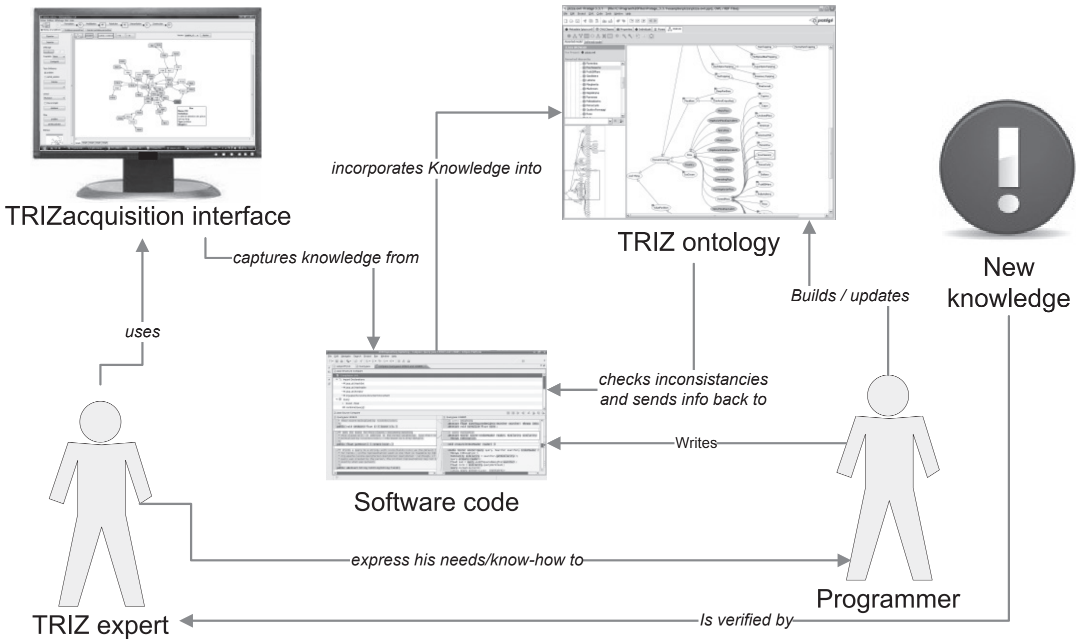

TRIZacquisition interface
incorporates Knowledge into
TRIZ ontology
New knowledge
captures knowledge from
Builds / updates
uses
checks inconsistancies and sends info back to
Writes
Software code
express his needs/know-how to
Programmer
TRIZ expert
Is verified by

Figure 1: Graphical representation of interactors.

A Schematic illustration of the relation between our ontology and the software prototype is given figure 1.

Our ontology formalizes the main concepts of TRIZ for operating inventive design. Knowledge acquisition and problem formulation will then consist in populating this ontology. Ontology concepts are schematized as a set of objects, illustrated in "UML-like" diagrams [12]

The following notation will be used:

- Boxes represent concepts.
- Single links represent associations. Associations may be named and may include cardinality at the ends. Bidirectional associations are represented by a single line while unidirectional associations are represented by a simple arrow.
- A clear diamond shaped arrow represents an aggregation, that is more specific that an association. It is an association that represents a part-whole relationship. As a type of association, an aggregation may be named and have a cardinality. Aggregation occurs when a concept is a container of other concepts. If the container is destroyed, its contents are not.
- A solid diamond shaped arrow represents a composition, a stronger variant of an association. It is a more specific part-whole relationship than an aggregation. In this case, if the container is destroyed, its contents are also destroyed.
- A hollow triangle shaped arrow indicates a generalization or is-a relationship. The arrow points to the super-concept.

With the goal of formalizing the theory's main concepts, we have compiled partially the vocabulary that is used by TRIZ experts. During knowledge acquisition, we have isolated the concepts, the predicates, the relationships and the class attributes; and have specified the meaning of every term that appears during the acquisition process.

Since TRIZ (observed from a classical viewpoint) was not featuring a complete and coherent set of concepts and relations, we completed this ontology both in reading available literature on further development of TRIZ like OTSM [13] and some existing publications on the subject.

An example of this ontology's role is the concept of Laws of engineering systems evolution. For many years, laws were used differently, sometimes forgotten or neglected. Not because people do not understand them or do not accept them, but just because it was unclear how to use them within a study case. This subject has already been discussed in [14]. As a result, the URL diagram (figure 1) of TRIZ ontology clearly shows that the reason was probably because nothing was linking them to other concepts. This situation was indeed a strong invitation to discuss about the creation of a new concept aiming at linking laws to the rest of the concepts.

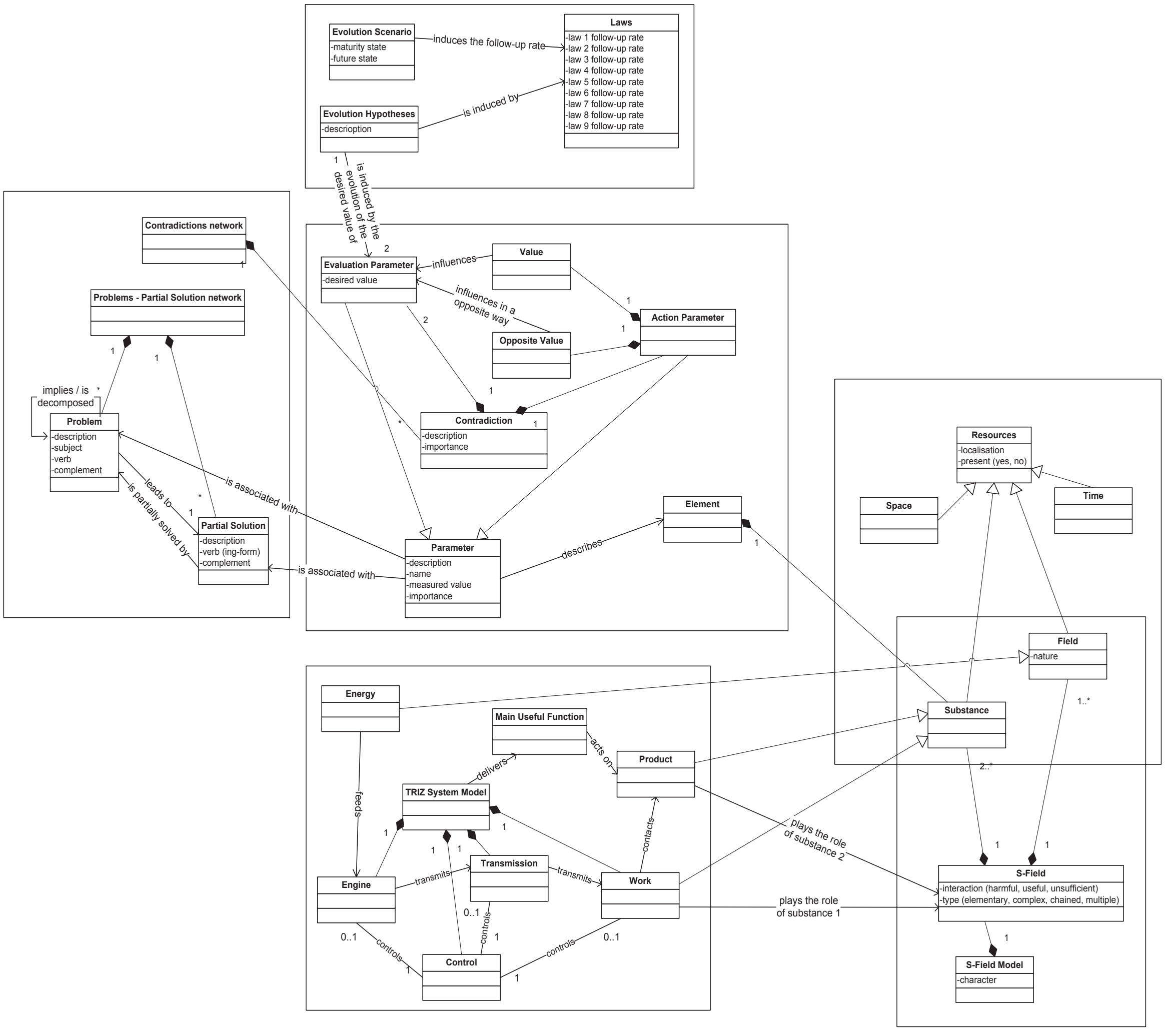


Figure 2: Domain ontology of TRIZ (UML representation).

6. A TRIZ related software prototype to operate and use this ontology

As a logical implementation of this ontology, we have targeted the objective of framing TRIZ expert practices with the objective of being exhaustive and clear when using concepts. In that sense, it forced the process to follow a path which has some degree of freedom but also some obligation of completing and populating the ontology.

Our software prototype can be decomposed in the following form:

- Initial situation description (beyond classical TRIZ concepts).
- Contradiction disclosure (Within classical TRIZ concepts).
- Solution concept generation (Within Classical TRIZ Concepts).
- Decisions for R&D activities (beyond classical TRIZ concepts).

The navigation system follows the main steps of TRIZ and beyond; from Initial Situation analysis to Solution concepts evaluation.

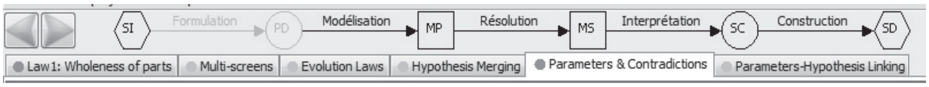


Figure 3: TRIZ acquisition navigation system – first level and formulation sub-level.

A screenshot of the software interface between the TRIZ expert and program is represented figure 4. The main idea of such interface is both to compile experts gathered data, but also to check their coherence and perform some functionalities useful for TRIZ study.

As an example, in the contradiction extraction step, we need to be respectful to the links between system's parameters. Parameters qualify elements while allocating them a specificity which, associated to elements, represent an explicit knowledge of the field observed. The forms of their expression are multiple; they are mainly names, complements of objects or adverbs and. They are divided in two categories:

- Action Parameters (AP): they represent parameters on which the designer has a capacity of state modifications (the designer can make a design choice, an anvil of large volume or a small one, in this case volume = AP).
- Evaluation Parameters (EP): Their nature lies in the capacity to evaluate the positive aspect resulting from a choice of the designer. The consequence to design an anvil with an important mass is that an ease of insertion is a logical consequence; (in this case Ease of driving = EP).

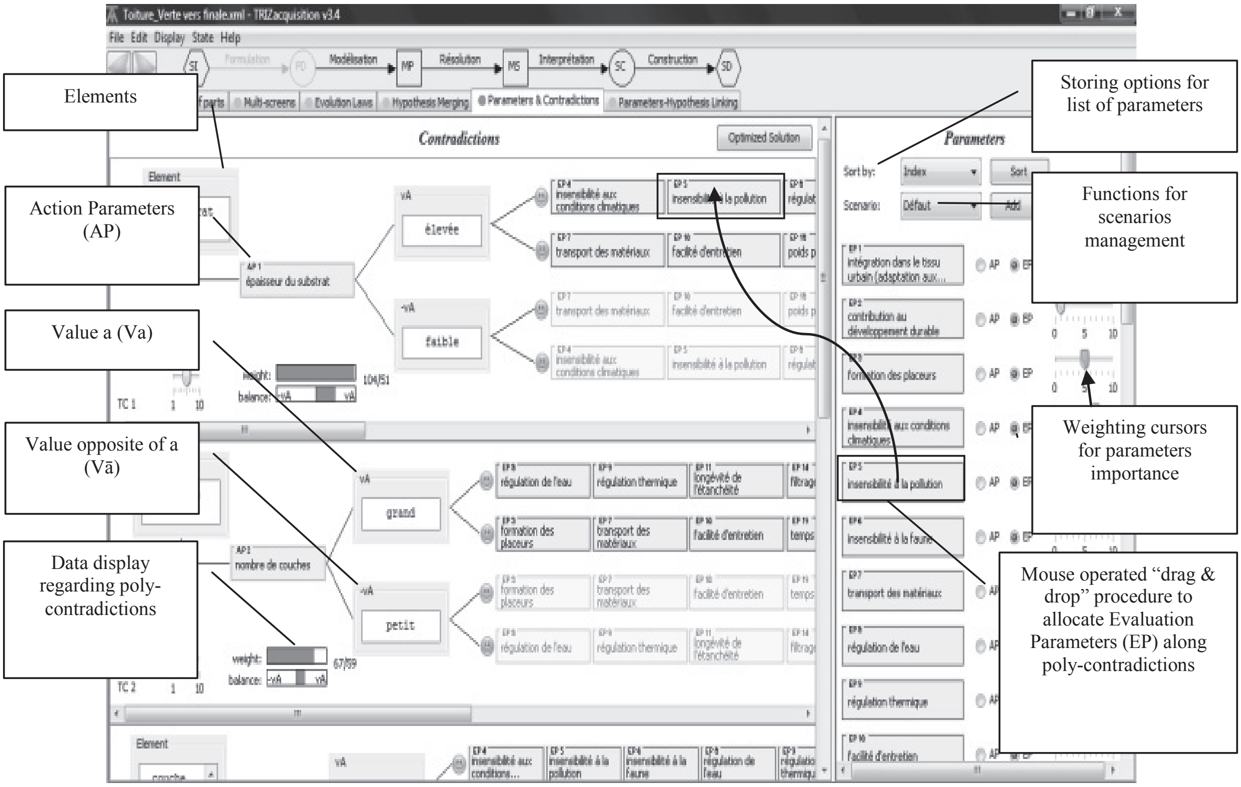
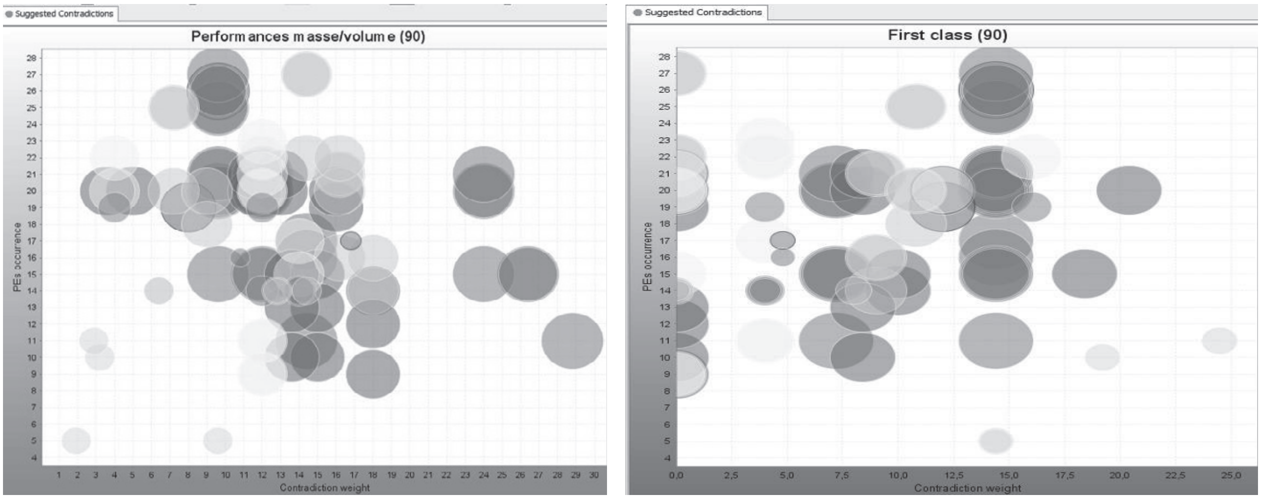

Toiture_Verte vers finale.xml - TRIZacquisition v3.4
File Edit Display State Help
Contradictions
Optimized Solution
Parameters
Sort by: Index
Sort
Scenario: Défaut
Add
AP 1 épaisseur du substrat
vA élevée
-vA faible
AP 2 nombre de couches
vA grand
-vA petit
EP 4 insensibilité aux conditions climatiques
EP 5 insensibilité à la pollution
EP 7 transport des matériaux
EP 10 facilité d'entretien
EP 8 régulation de l'eau
EP 9 régulation thermique
EP 11 longévité de l'étanchéité
EP 3 formation des placeurs
EP 1 intégration dans le tissu urbain (adaptation aux...
EP 2 contribution au développement durable
EP 6 insensibilité à la faune
weight: 104/51
balance: -vA vA
weight: 67/89
TC 1
TC 2
AP EP
Elements
Action Parameters (AP)
Value a (Va)
Value opposite of a (Vā)
Data display regarding poly-contradictions
Storing options for list of parameters
Functions for scenarios management
Weighting cursors for parameters importance
Mouse operated "drag & drop" procedure to allocate Evaluation Parameters (EP) along poly-contradictions

Figure 4: User interface of the poly-contradiction scenarios construction and weighting screen.



5a 5b

Figure 5: Weighted set of Contradictions based on cost reduction (a) and breakthrough (b) scenario.

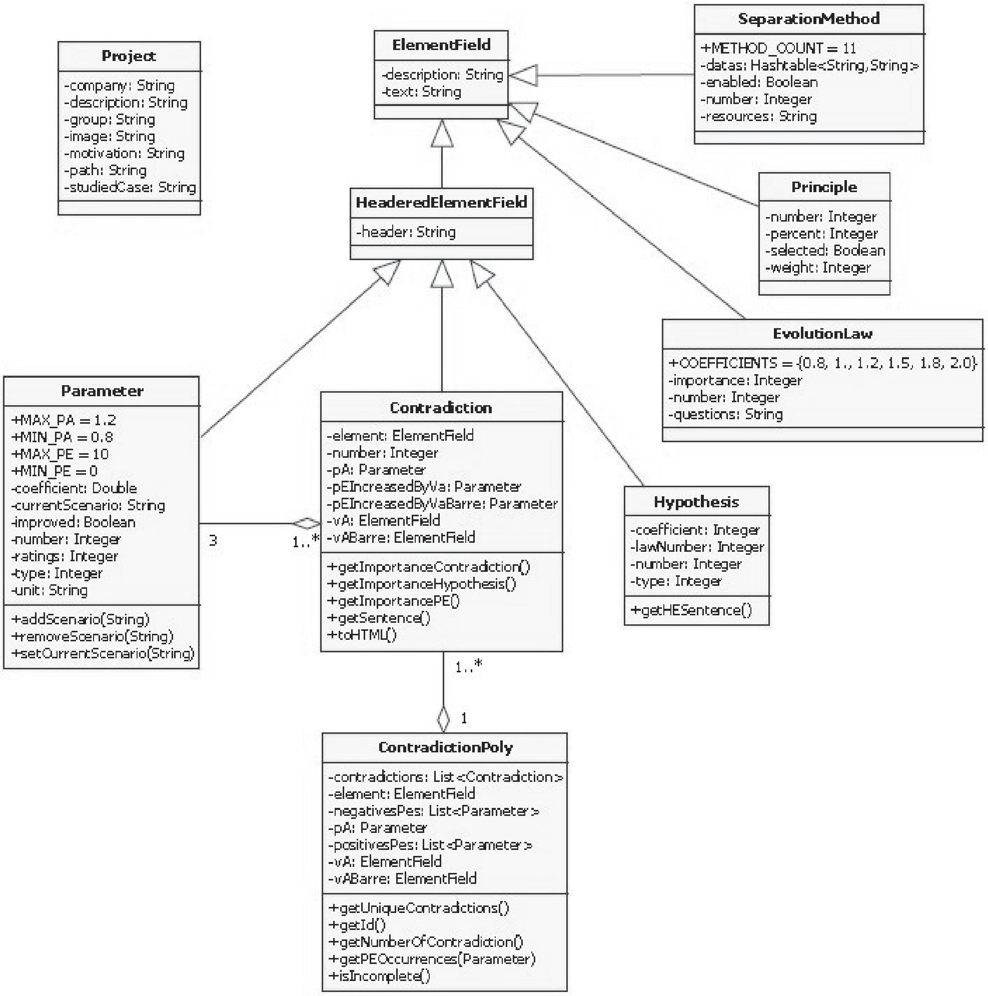


Figure 6: Concepts to be instantiated both in the code and in the ontology – formulation sub-level.

If we want to avoid any wrong construction of a contradiction, while the ontology is made, it will also check the coherence of the contradiction performed by the user and either accept the construction or refuse it. Rules for building relationships between EP and AP (and many other rules) are stored in the ontology.

Another example is related to a new notion we have observed during case studies and absent from classical TRIZ: the notion of viewpoints. The question is the following: “how to capture the fact that each value associated with the importance of an EP and an AP is tightly linked with the adopted viewpoint of the observers?”

An answer we provide to this question is the building of several ranking scenarios depending on the viewpoint the study team wants to adopt using tables. In coherence with the company’s strategy, the study team might want to investigate both scenarios of cost reduction or breakthrough strategy in parallel. Of course we assume that depending on the strategy the company wants to adopt, the ranking of EP and AP will vary and provoke a different layout of contradictions to be solved in the rest of the studyIn the figures 5a & 5b, the layout of contradictions presented as a cloud, further details regarding this representation can be found in [15].

As case study data are discussed through the relation between domain experts and TRIZ expert, they are compiled in the ontology through the software interface.

The ontology is instantiated by the expert inputs through an established relationship between software code and protégé ontology model. To clarify the concepts attributes in detail, we compiled the elements using UML diagrams and we iterate them as new elements of knowledge appear out of our research findings. In figure 6, the example of UML is limited to the screenshot figure 4: problem formulation and contradiction synthesis sub-level.

7. Discussions

Several analyses of our software prototype in operation have been conducted through case studies. They lead us to the following items of discussions:

An incomplete model tacitly creates confusion in engineers mind. As engineers are sometime intuitively acting when designing in projects, the fact that TRIZ way of conducting design differs from classical design, disturb the progress of the study. While engineers are expecting the model we build to be respectful to the model they have in mind, we face with a lot of questions as the TRIZ model seemed to be incomplete for them. The most predominant questions were oriented towards clarifying why customers needs were absent from the model. This is mostly due to lack of understanding of TRIZ thinking, which is law of evolution oriented. At this point, our ontology model allows us to highlight that customer’s needs were present within evaluating parameters.

On the contrary, expressed completeness of the ontology positivizes people’s attitude and confidence. We have observed that it was not traditional in their way of designing to automate the evaluation of solution concepts impact of the expressed problematic. This stage was mostly intuitive and based upon grid of evaluation (level of satisfaction to fulfilling criteria). Due to the completeness of our model, the feedback of the impact is thoroughly built all along the study and when solution concepts appear (based on contradiction’s resolving) their impact of problem graph is rapidly made and displayed to the study group.

Sharing and co-working on a given ontology is essential for a domain to progress. If we assume that an ontology is an unavoidable step in the building of a theory it is even more crucial for the communication and understanding around its groundings. The findings are multiple, from easy building of a teaching curriculum to automating tasks in the process but moreover new research openings and findings for its evolution.

As an opening to other research communities and to assume what has been described through this paper, we have made available our ontology through web-protégé framework and hope to start a discussion around the concepts, their interrelations and that the result of such collaboration will be useful for all.

The software prototype we built (TRIZacquisition) is now in its third generation and has also opened its internal procedures to elements beyond the scope of classical TRIZ. We have implemented the notion of Problem Graph as a mean to build a multidisciplinary problematic model and tested the coherence of these new elements with the current TRIZ based ontology.

8. Summary

As we clarify TRIZ major concepts and their interrelations, we observe a reduction of complains from its observers or newcomers relatively to its complexity. Indeed, if TRIZ remains mastered intuitively by a reduced amount of highly skilled experts, it will always appear fuzzy to outsiders or newcomers and complex to learn. What we are targeting through this article is to contribute to TRIZ progresses in using standard procedures in science: the building of an ontology.

Experiences of reduction of TRIZ's complexity through SIT, ASIT & others are valuable in certain sub-activity of classical design like "creative or ideation stage", but cannot support the complete driving of a design activity. Therefore, we shall distinguish reoriented uses of TRIZ for creative stage enhancement from exhaustive use of TRIZ for supporting inventive design practices. This ontology is not for SIT - derivate methods and tools but will certainly be helpful for positioning TRIZ and SIT as this question is recurrent in TRIZ societies.

In the coming future, what is known as TRIZ will certainly be less visible as tools for its implementation will represent its concepts on reliable basis. It will remain a theoretical grounding for methods and tool's construction, as any theory is logically proposing. This is nevertheless not avoiding industry to organize inventive practices from an organizational perspective: organizing innovation in the corporate structure. But again, since clearly defined, accepted and shared ontology will give a reliable rise of efficient computer supports, these tools will certainly assist the successful implementation and the soaring, not of TRIZ itself, but of its forthcoming derivate methods and tools.

References

[1] Martin R., 2007, "TRIZ and the optimization conjecture", TRIZconference 2007, Saint-Petersbourg - Russia.

[2] KUROSAWA S., 2008, "Contemporary Issues of TRIZ to be a Scientific Method", The Fourth TRIZ Symposium – Japan.

[3] Brickley D., Guha, R.V., 1999, Resource Description Framework (RDF) Schema Specification. Proposed Recommendation, World Wide Web Consortium: http://www.w3.org/TR/PR-rdf-schema.

[4] Hendler and McGuinness, 2000, Hendler, J. and McGuinness, D.L. (2000). "The DARPA Agent Markup" Language. IEEE Intelligent Systems 16(6): 67-73.

[5] Milton N.R., 2007, "Knowledge Acquisition In Practice: A Step-By-Step Guide" Springer-Verlag London Ltd.

[6] Noy N. F., Mcguinness D. L., 2001, "Ontology development 101: A guide to creating your first ontology". Online URL http://www.ksl.stanford.edu/people/dlm/papers/ontology101/ ontology101-noy-mcguinness.html

[7] Guarino N., 1998, "Formal Ontology and Information Systems". In N. Guarino (ed.). "Formal Ontology in Information Systems". Proceedings of FOIS'98, Trento, Italy, 6-8 June 1998.Amsterdam, IOS Press, pp. 3-15.

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

[9] Heller B., Herre H., 2003, "Ontological Categories in GOL" Axiomathes, Volume 14, Numbers 1-3, pp. 57-76(20)

[10] Niles I., Pease A., 2001, "Towards a Standard Upper Ontology", FOIS 2001, pp. 57-7.

[11] Zanni-Merk C., et al., 2009, "An ontological basis for computer aided innovation", Comput. Industry, doi:10.1016/j.compind.2009.05.012

[12] Rumbach J., Jacobson, I., Booch, G., 2004, "The Unified Modelling Language Reference Manual (2nd edition)", Addison Wesley Professional, ISBN 978-0321245625

[13] Khomenko N., De Guio R., Lelait L. and Kaikov I., 2007, "A framework for OTSM-TRIZ based computer support to be used in complex problem management", International Journal of Computer Applications in Technology, Vol. 30, No. 1/2, 2007, pp 88-104.

[14] Cavallucci D., Rousselot F. and Zanni C. “Linking contradictions and laws of engineering system evolution within TRIZ framework” , Creativity and Innovation Management, ISSN 0963-1690, Blackwell Publishing.

[15] Cavallucci D., Rousselot F. and Zanni C., 2008, “On contradiction clouds”, 8th ETRIA’s International TRIZ Future Conference (TFC2008), Nov. 5-7, Twente – The Netherland.