# Case Retrieval in Ontology-Based CBR Systems

Amjad Abou Assali, Dominique Lenne, and Bruno Debray

University of Technology of Compiègne, CNRS
HEUDIASYC
{aabouass,dominique.lenne}@utc.fr
INERIS
bruno.debray@ineris.fr

**Abstract.** This paper presents our knowledge-intensive Case-Based Reasoning platform for diagnosis, COBRA. It integrates domain knowledge along with cases in an ontological structure. COBRA allows users to describe cases using any concept or instance of a domain ontology, which leads to a heterogeneous case base. Cases heterogeneity complicates their retrieval since correspondences must be identified between query and case attributes. We present in this paper our system architecture and the case retrieval phase. Then, we introduce the notions of similarity regions and attributes' roles used to overcome cases heterogeneity problems.

**Keywords:** Case-based reasoning, Ontology, Heterogeneous case base, Similarity measures, Similarity regions.

## 1 Introduction

In industrial sites concerned by the SEVESO directive, risks exist due to the presence of hazardous substances and processes. To reduce such risks, safety barriers are set up depending on the type of process and on the hazardous situation to control as well as on the local environmental conditions. However, barriers may not work properly, and thus hazards may arise. In such a case, industrial experts intervene to diagnose the barriers failure basing mostly on past failure experiences occurred in similar situations. The hypothesis of experts here is that "if a barrier did not work correctly in a past similar situation, it is strongly probable that it does not work, in the current situation, *for similar reasons*". To simulate the experts' activity, we use a Case-Based Reasoning (CBR) approach [1]. CBR is a problem-solving approach used to solve a new problem (target case) by remembering a previous similar situation (source case) and by reusing information and knowledge of that situation [2]. In the diagnosis activity, when no obvious cause of failure is found, experts may realize that more information is needed to find the right cause of failure. This led us to apply a conversational CBR approach where cases are enriched as experts advance in the diagnosis.

The system was first applied to the diagnosis of the failure of gas sensors installed in industrial plants. Such sensors are intended to detect certain gases

B. Mertsching, M. Hund, and Z. Aziz (Eds.): KI 2009, LNAI 5803, pp. 564–571, 2009.

so that if there is a leak somewhere in the site, a safety action can be undertaken. In this context, a case represents a diagnosis of the failure of a gas sensor in a given industrial environment.

In this paper, we introduce our platform COBRA (Conversational Ontology-based CBR for Risk Analysis), an ontology-based CBR platform. It allows to capitalize and reuse past failure experiences based on ontological models describing the domain and case structures. One of the interesting features of COBRA is that it allows a dynamic representation of cases; *i.e.,* users can define their cases' attributes at run time, which leads to a heterogeneous case base. However, this heterogeneity complicates the case retrieval phase. We describe in this paper the case retrieval phase as well as the similarity measures used. In addition, we present our approach to overcome the cases heterogeneity problems.

## 2 Related Work

The integration of general domain knowledge into knowledge-intensive CBR systems was an important aspect in several projects. In the CREEK architecture [3], we find a strong coupling between case-based and generalization-based knowledge. Thus, cases are submerged within a general domain model represented as a densely linked semantic network. Fuchs & Mille proposed a CBR modeling at the knowledge level [4]. They distinguished four knowledge models: 1) the conceptual model containing the concepts used to describe the domain ontology regardless of the reasoning process; 2) the case model that separates the case into three parts: problem, solution, and reasoning trace; 3) the models describing the reasoning tasks; 4) and the reasoning support models. Dìaz-Agudo & Gonzàlez-Calero [5] proposed a domain-independent architecture that helps in the integration of ontologies into CBR applications. Their approach is to build integrated systems that combine case specific knowledge with models of general domain knowledge. So, they presented CBROnto [6], a task/method ontology that provides the vocabulary for describing the elements involved in the CBR processes, and that allows to integrate different domain ontologies. CBROnto was reused later by jCOLIBRI, a powerful object-oriented framework for building CBR systems [7]. It splits the case base management into two concerns: persistence and in-memory organization, which allows different storage mediums of cases (text/XML files, ontology, *etc.*) accessed via specific connectors. However, jCOLIBRI does not allow the treatment of dynamic and heterogeneous case bases. In this work, we decided to keep the interesting aspects of jCOLIBRI, and to add our own layer that allows to work with heterogeneous case bases.

## 3 COBRA Architecture

Several CBR architectures have been presented in the literature [2,8]. Inspired by these architectures, COBRA architecture is composed of two main layers [9]:

– *Processes* layer: it contains the off-line process, *case authoring*, along with the reasoning processes: ELABORATE, RETRIEVE, DIAGNOSE, ENRICH, REVISE,

and RETAIN. In the "diagnose" phase, the system tries to identify failure causes from the retrieved similar cases. If no cause is found, or even if the diagnosis proposed by the system was not validated by the user, the system asks the user to "enrich" his case description, and thus new similar cases can be retrieved. This cycle is repeated until a relevant diagnosis is proposed by the system, or no solution is found.

– *Knowledge containers* layer: it contains three main containers: vocabulary, case base, and similarity measures. In knowledge-intensive CBR systems, ontologies play an important role in representing these containers. They can be used as the vocabulary to describe cases and/or queries, as a knowledge structure where the cases are located, and as the knowledge source to achieve semantic reasoning methods for similarity assessment [7]. The vocabulary and the case base rely on two knowledge models, the domain and case models respectively.

### 3.1 Domain Model

This model represents the knowledge about safety barriers, our current domain of application, in an ontological structure. Following the classification proposed by Oberle [10], we have developed two types of ontologies : a core ontology that contains generic concepts about industrial safety such as EQUIPMENT, DANGEROUS PHENOMENON, SAFETY BARRIER, EFFECT, *etc.* A domain ontology describing the domain of safety barriers, in particular gas sensors. Its concepts are specializations of other concepts of the core ontology, and contains concepts such as: GAS SENSOR, SAFETY FUNCTION, ENVIRONMENTAL CONDITION, *etc.* Ontologies are represented in OWL Lite and have been developed by several experts of the INERIS, the French national expertise institute on industrial safety, with the help of an ontology expert [11].

### 3.2 Case Model

A case in our system represents a diagnosis experience, and thus consists of three main parts: a *description* part describing the context of the experience, a *failure mode* part describing the type of failure, and a *cause* part describing the different possible causes of this failure. A case in general is described by a pair (*problem*, *solution*). Accordingly, the *description* and *failure mode* parts of our model correspond to the *problem* part, and the *cause* part corresponds to the *solution* part. Inspired by the approach of JCOLIBRI, in order to enhance the communication between the case base and the domain model, the case model is represented within an ontology that integrates the domain model. This ontology contains the main following concepts (Figure 1): 1) CBR-CASE that subsumes the various case types that may exist in the system; 2) CBR-DESCRIPTION that subsumes the case main parts, *failure mode* and *cause*; and 3) CBR-INDEX allowing to integrate the domain model concepts used to describe cases.

Cases are thus represented by instances of the ontology and can have two types of attributes: *Simple attributes* corresponding to data-type properties of

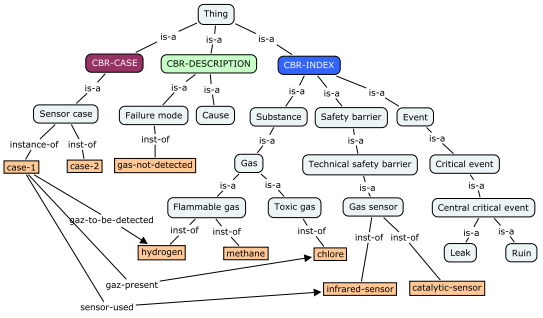


**Fig. 1.** Case model

the ontology, and *Complex attributes* corresponding to instances of the ontology that have, in turn, their own simple properties. In the case authoring phase, we realized that cases do not share always the same attributes. Thanks to the developed platform, experts were allowed to use any concept from the domain model to describe their cases [9]. This led to a heterogeneous case base, which complicated the case retrieval phase.

## 4 Case Retrieval

In this phase, similarity measures are employed to retrieve similar cases for a user query. Generally, with object-oriented case structures, similarity measures follow the "local-global principle" [12,13]. This principle is followed in our work since cases are represented by instances of the ontology. The similarity computation of two ontology concepts can be divided into two components [12,7]: *a concept-based similarity* that depends on the location of concepts in the ontology, and *a slot-based similarity* that depends on the fillers of the common attributes between the compared objects.

### 4.1 Weighted Similarity Measures

Sometimes, query attributes may not have the same importance (weight) in the similarity computation. In this work, weights can be assigned to simple attributes (IGNORE or EXACT), and/or to complex attributes (in the interval [0, 1]).

Let $Q = \{q_i : 1 \leq i \leq n, n \in \mathbb{N}^*\}$ be the user query, where $q_i$ is a simple or a complex attribute, and let $\Omega = \{C_j : 1 \leq j \leq k, k \in \mathbb{N}^*\}$ be the case base, where $C_j = \{c_{jl} : 1 \leq l \leq m_j, m_j \in \mathbb{N}^*\}$. The concept-based similarity measure, $sim_{cpt}$, is defined as follows: For each complex attribute, $q \in Q$ and $c \in C$,

$$sim_{cpt}(q, c) = w_q * \frac{2 * prof(LCS(q, c))}{prof(q) + prof(c)} \quad (1)$$

where $w_q$ is the weight assigned to $q$, $prof$ is the depth of a concept (or an instance) in the ontology hierarchy, and $LCS$ is the Least Common Subsumer

concept of two instances. In a special case, when $q$ and $c$ represent the same instance in the ontology, we have: $prof(LCS(q,c)) = prof(q)$.

The slot-based similarity measure, $sim_{slt}$ is defined as follows:

$$sim_{slt}(q,c) = \frac{\sum_{s \in CS} sim(q.s, c.s)}{|CS|} \tag{2}$$

where $CS$ is the set of common simple attributes of $q$ and $c$, $|CS|$ is the set cardinality, $q.s$ (or $c.s$) represents the simple attribute $s$ of $q$ (or $c$), and $sim(q.s, c.s)$ is the similarity measure between the two simple attributes. For the moment, we consider only the first two weights (IGNORE, EXACT), and thus $sim(q.s, c.s)$ can be defined as:

$$sim(q.s, c.s) = \begin{cases} 1 & \text{if} \quad (w_{q.s} = exact) \wedge (v_{q.s} = v_{c.s}) \\ 0 & \text{otherwise} \end{cases}$$

where $w_{q.s}$ is the weight of the simple attribute $q.s$, and $v_{q.s}$ is its value.

Now, it is time to define the global similarity measure of the two complex attributes, $q$ and $c$, which is given by the following formula [14]:

$$sim(q,c) = (1-\alpha) * sim_{cpt}(q,c) + \alpha * sim_{slt}(q,c) \tag{3}$$

where $\alpha$ is an experience parameter (at present, $\alpha = 0.4$).

To compute the similarity between a case and a given query, each complex attribute of the query is compared to its corresponding attribute in the case. In *homogeneous* case bases, all cases share the same predefined structure, and thus, corresponding complex attributes are already identified. On the other side, *heterogeneous* case bases contain cases with different structures. Therefore, before computing the similarity, we need to determine the corresponding complex attributes.

For each complex attribute $q' \in Q$, let $c' \in C_j$ be the corresponding complex attribute in the case $C_j \in \Omega$. To determine $c'$, we considered first that it is the attribute with which $q'$ has the maximum similarity in the case $C_j$:

$$sim(q', c') = \max_{1 \leq l \leq m_j} (sim(q', c_{jl})) \tag{4}$$

However, we realized that this definition of corresponding attribute is not sufficient, and that more conditions must be satisfied. The problem is that $c'$ may have a maximum similarity with $q'$ while, in reality, $q'$ may have no corresponding attribute in $C_j$. The first improvement that we propose is to compare the obtained similarity $sim(q', c')$ with the maximum similarity obtained for $q'$ within the whole case base. This leads to the following condition:

$$\frac{sim(q', c')}{\max_{1 \leq j \leq k, 1 \leq l \leq m_j} (sim(q', c_{jl}))} \geq \beta \tag{5}$$

where $\beta$ is a specific threshold calculated after experimentations ($\beta = 0.6$).

This condition led to better results, but it was not however sufficient in special cases. Thus, we propose to consider the following interesting notions:

### 4.2 Similarity Regions

We consider that the similarity measures are not always applicable to each pair of concepts (or instances) of the ontology. For example, a gas instance must not be compared at all with an equipment instance. To prevent such comparisons, we propose to define the notion of *similarity regions*. A similarity region is a sub-branch of the ontology hierarchy where concepts and instances are comparable with each other (Figure 2). The definition of such regions is manual and depends on the target application. To compute the similarity between a query attribute and a case attribute, it must be verified first if these two attributes belong to the same similarity region.

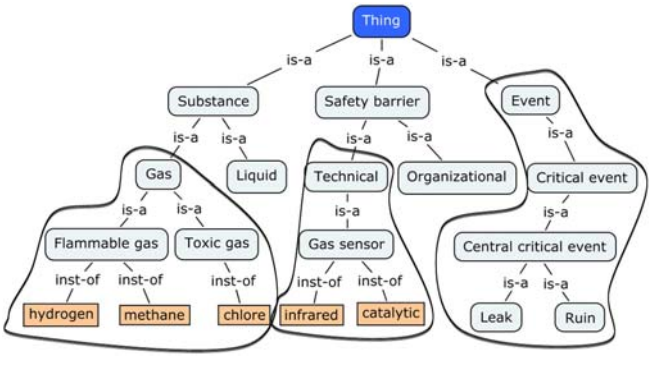


**Fig. 2.** Example of similarity regions

### 4.3 Roles of Attributes

For a given query attribute, several corresponding attributes may be found sometimes. Let's take, for example, the following excerpt of a case description: "At an industrial site, an infrared sensor was used to **detect** the *methane*. Other gases were **present** at the site such as the *hydrogen*. The sensor did not work as expected, and there was an explosion consequently". Now, let's suppose a query looking for cases where a gas sensor was used to **detect** the *hydrogen*. Following the approach presented till now, we find that the hydrogen of the case is the attribute corresponding to the hydrogen of the query. However, it is not the aimed result since we look for the hydrogen when it is the gas to be detected, which means that its actual corresponding attribute must be the methane.

To solve this ambiguity, we propose to describe the *role* of case attributes that may give rise to ambiguous situations. For example, the hydrogen of the case is a ***gas present*** at site, and the methane is the ***gas to be detected***. Thus, attributes that have same roles are identified first. Then, for the other attributes, we follow the proposed approach (Formulas (4), (5) along with similarity regions).

## 5 Results and Conclusion

This work led to the development of COBRA, a domain-independent CBR platform, as a JAVA application (Figure 3). It is based on jCOLIBRI, but it extends it with a layer that allows the treatment of dynamic and heterogeneous case bases. Thanks to the platform architecture, developing a new CBR system is made by supplying its domain ontology and reconfiguring some XML files. COBRA is used currently for two parallel objectives: to capitalize knowledge about gas sensor failures, and to provide support to experts and safety engineers to diagnose failure causes of gas sensors in industrial conditions. Experts can use any concept or instance of the domain model to describe their cases, which leads to a heterogeneous case base. Cases heterogeneity implicated that corresponding complex attributes, between a query and other cases, are complicated to identify. In this paper, we introduced the notions of similarity regions and attributes' roles to overcome this problem.

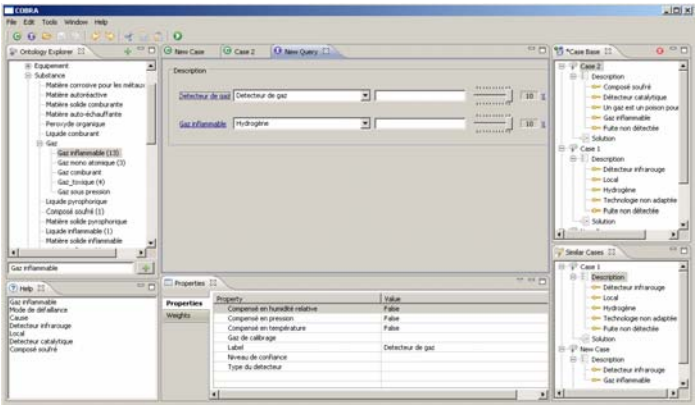

**Fig. 3.** COBRA platform

To evaluate the proposed approach, cases have been added to the case base. They correspond to real situations observed in industry or to tests realized in the INERIS during campaigns for qualification of sensors. These cases allowed us to do a preliminary validation of the case retrieval phase, which gave very satisfying results. The validation has been done to show the quality of results when adopting each of the improvements proposed; *i.e.,* Formula 5, similarity regions, and attributes' roles. We have also developed the "diagnose" phase of the CBR processes (out of the paper's scope). Then, more complex cases have to be added to the case base to allow validating our system. This validation will be done by experts at two levels:

– the first level concerns the CBR architecture: to what end the case structure and the reasoning processes are close to the real experts' activity? What other concepts should be added to the ontology? *etc.*

– the second level concerns the quality of diagnosis returned by the system for some given queries.

**Acknowledgments.** The present work has received the financial support of the French Ministry of Ecology and sustainable development. The authors also thank Mr. Sébastien Bouchet (INERIS) for his active participation to the development of the gas sensor ontology and the case base.

## References

1. Riesbeck, C., Schank, R.: Inside Case-Based Reasoning. Lawrence Erlbaum Associates, Mahwah (1989)
2. Aamodt, A., Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. AI Communications 7(1), 39–59 (1994)
3. Aamodt, A.: Explanation-Driven Case-Based Reasoning. LNCS, pp. 274–274 (1994)
4. Fuchs, B., Mille, A.: Une modélisation au niveau connaissance du raisonnement à partir de cas. In: L'Harmattan (ed.) Ingénierie des connaissances (2005)
5. Dìaz-Agudo, B., Gonzàlez-Calero, P.: An architecture for knowledge intensive CBR systems. In: Blanzieri, E., Portinale, L. (eds.) EWCBR 2000. LNCS (LNAI), vol. 1898, pp. 37–48. Springer, Heidelberg (2000)
6. Dìaz-Agudo, B., Gonzàlez-Calero, P.: CBROnto: a task/method ontology for CBR. In: Procs. of the 15th International FLAIRS, vol. 2, pp. 101–106 (2002)
7. Recio-Garcìa, J., Dìaz-Agudo, B., Gonzàlez-Calero, P., Sanchez, A.: Ontology based CBR with jCOLIBRI. Applications and Innovations in Intelligent Systems 14, 149–162 (2006)
8. Lamontagne, L., Lapalme, G.: Raisonnement à base de cas textuels: Etat de l'art et perspectives. Revue d'intelligence artificielle 16(3), 339–366 (2002)
9. Abou Assali, A., Lenne, D., Debray, B., Bouchet, S.: COBRA: Une plateforme de RàPC basée sur des ontologies. In: Actes des 20es Journées Francophones d'Ingénierie des Connaissances (IC 2009), Hammamet, Tunisie, May 2009, pp. 277–288 (2009)
10. Oberle, D.: Ontologies. In: Semantic Management of Middleware. Semantic Web and Beyond, vol. 1, pp. 33–53. Springer, Heidelberg (2006)
11. Abou Assali, A., Lenne, D., Debray, B.: Ontology development for industrial risk analysis. In: IEEE International Conference on Information & Communication Technologies: from Theory to Applications (ICTTA 2008), Damascus, Syria (April 2008)
12. Bergmann, R., Stahl, A.: Similarity measures for object-oriented case representations. In: Smyth, B., Cunningham, P. (eds.) EWCBR 1998. LNCS (LNAI), vol. 1488, p. 25. Springer, Heidelberg (1998)
13. Richter, M.: Similarity. In: Case-Based Reasoning on Images and Signals. Studies in Computational Intelligence, vol. 73, pp. 25–90. Springer, Berlin (2008)
14. Zhang, K., Tang, J., Hong, M., Li, J., Wei, W.: Weighted Ontology-Based Search Exploiting Semantic Similarity. In: Zhou, X., Li, J., Shen, H.T., Kitsuregawa, M., Zhang, Y. (eds.) APWeb 2006. LNCS, vol. 3841, pp. 498–510. Springer, Heidelberg (2006)