A Logic Programming Approach to the Conservation of Buildings Based on an Extension of the Eindhoven Classification Model

Guida Gomes, Henrique Vicente, Joaquim Macedo, Victor Alves, and José Neves

***Abstract*—The identification, classification and recording of events that may lead to the deterioration of buildings are crucial for the development of appropriate repair strategies. This work presents an extension of the *Eindhoven Classification Model* to sort adverse events root causes for *Building Conservation*. *Logic Programming* was used for knowledge representation and reasoning, letting the modelling of the universe of discourse in terms of defective data, information and knowledge. Indeed, a systematization of the evolution process of the body of knowledge in terms of a new factor, the *Quality of Information* one, embedded in the *Root Cause Analysis* was accomplished, i.e., the system proposed led to a process of *Quality of Information* quantification that permit the study of the event's root causes, on time.**

***Index Terms*—Building conservation, Eindhoven classification model, knowledge representation and reasoning, logic programming, quality of information.**

I. Introduction

THE use of information systems as a tool for acquisition, storage and manipulation of data represents the minimum level that may be required from the information technology. In fact, presently more than the automation of processes and the increase of the data repositories are required. The focus is placed on the ability of the information systems to be an autonomous process of evaluation, decision and learning. This configures a transversal dimension that encompasses various scientific areas.

The application of methodologies emanating from the Scientific Area of Artificial Intelligence to solve problems in the field of Civil Engineering is not new, dating from the early 90s of XX century. Since then several studies have been published where techniques like Artificial Neural Networks and Genetic Algorithms have been applied to solve some specific problems within the Civil Engineering portfolio [1]. Recently Lu et al. [2] presented an overview of the application of new methodologies developed in the field of Artificial Intelligence to Civil Engineering. Among them some should be highlighted, like Evolutionary Computation, Swarm Intelligence, Fuzzy Systems, Reasoning Based Systems and Chaos Theory.

Dukić et al. [3] present a model to facilitate the planning of maintenance activities, in order to rationalize costs through preventive interventions. The system can store the information obtained in the regular inspections and based on them, infer about possible failures and/or loss of the buildings' functional characteristics. Furthermore the database allows monitoring the behavior of the various elements of construction. Motawa and Almarshad [4] developed an integrated system for archiving information and knowledge regarding the maintenance of buildings. The proposed system aims at the understanding of the causes of building deterioration, but also acts as a decision support system regarding preventive or corrective maintenance actions. This system comprises a registration module, a database and a knowledge extraction module for the construction of a knowledge base.

However, the machinery mentioned above does not work with incomplete, unknown and/or forbidden information. In fact, for many situations that occur daily in building conservation complete information does not exist at all. Instead, the information available is insufficient or incomplete.

Undeniably the building conservation area is complex and multifaceted and various types of adverse events may occur. An adverse event may be defined as the failure of a planned action to be completed as intended or the use of a wrong plan to achieve an aim, and includes problems in practice, relationships, procedures and systems. The most effective way to prevent adverse events is to attack directly their causes. Preventing the adverse events' root causes improves significantly the conservation/maintenance of buildings. Thus, the proposed model will focus primarily on preventing the adverse events' root causes. The model planned serves as the formal foundation to an adverse event reporting and learning computational system.

II. The Computational Model

An extended version of the *Eindhoven Classification Model* (*ECM*), with the extensions and adaptations for the area of

Manuscript received on August 6, 2013; accepted for publication on September 30, 2013.

Guida Gomes is with the Department of Informatics, University of Minho, Braga, Portugal (e-mail: mguida.mgomes@gmail.com).

Henrique Vicente is with the Department of Chemistry & Évora Chemistry Centre, University of Évora, Évora, Portugal (e-mail: hvicente@uevora.pt).

Joaquim Macedo, Victor Alves, and José Neves are with the Department of Informatics, University of Minho, Braga, Portugal (e-mail: {macedo, valves, jneves}@di.uminho.pt).

conservation and maintenance of buildings and its causal tree, used to classify the adverse events' root causes in conservation / maintenance of buildings, is presented. The theoretical foundation is based on an extension to *Logic Programming*, in terms of a revision of its knowledge representation and reasoning mechanisms. The introduction of explicit negation in this universe endorsed the development of a process of quantification of the above mentioned *Quality of Information* (*QoI*) factor, embedded in the predicates extensions that make one´s system, making possible to study the event's root causes and to generate alerts and recommendations in order to improve the state of building conservation and maintenance.

A. The Eindhoven Classification Model

The *ECM* was originally developed in order to manage human error in the chemical industry [5], being then applied to other industrial arenas, such as energy production, steel industry and healthcare. The *Eindhoven Classification Model – Medical Version* consists of 20 (twenty) codes, divided into four categories frequently used in a medical environment to classify the underlying causes of the adverse events [6], and recently was extended and adapted for the specific area of imaging [7]. This approach assumes that humans are fallible and that errors are to be expected in every organization, so it is necessary to concentrate efforts on the conditions under which individuals work and try to build defenses to avert errors or to mitigate their effects. Assigning codes to the causes of each adverse event, it is a practice that is useful for tracking and trending.

The first stage to use the *ECM* based classification system is to identify the root causes that result in a specific adverse event. These root causes are subsequently classified according to the classification model. Indeed, a causal tree is built and techniques of *Root Cause Analysis* (*RCA*) are applied. Once the root causes are identified, they may be used to provide a more realistic view of how the system really works, as well as to contribute to the creation of effective and lasting solutions.

B. The Extended Eindhoven Classification Model

The *Extended Eindhoven Classification Model* (*EECM*) was adapted from the *ECM*, presented in the previous section. To apply this model to the area of conservation and building maintenance, the authors developed extensions for each category of the original model. These extensions allow fitting each category into the area of conservation and maintenance of buildings and provide a broader view of the events that may occur and the degree of complexity of this field. Thus, the classification process becomes easier and more efficient. Table I shows the five categories that make up the model, a brief description of each one of them and the respective codes, while in Table II a subset of the *EECM* codes and some examples of adverse events are present. Figure 1, in turn, depicts the *EECM* flow chart.

For instance, in the original model, the adverse events classified as "*Human behaviour – Knowledge-based errors*" (*HKK*) occur due to "*the inability of an individual to apply existing knowledge to a new situation*". In the *EECM*, this definition was extended by saying that the events classified under this category are due to "*difficulties in execution, interpretation or reporting procedures*". Some of the adverse events falling into this category are "*poorly executed procedures*, *incomplete procedures* and *procedures poorly validated*".

The causal trees taken on by the original *ECM*, set that the recognition of the event's root causes and its mental picture, is done under a hierarchical structure. On the other hand, once one has to deal with incomplete and even contradictory information, an *Extension of Logic Programming* (*ELP*) was used for knowledge representation and reasoning, in order to get a truth value in the interval [0, 1] as a measure of confidence in any qualification process susceptible to be handled by the system. Since an event may only occur due to the combination of more than one cause, and a different event may come about due to two or more causes, taken separately, in the original model *AND-gates* and *OR-gates* are used to embody these two possibilities in the causal tree.

The usual situations may also include the case where only one cause leads to the occurrence of a certain event. In any case the adverse events' origins are known, i.e., there is certainty about the events' grounds. Beyond these situations, it may happen that the causes of an event, action or decision are unknown; it may be known that certain views are the source of a given event, but it may not be sure what are the event grounds; or it is not allowed to know the origin of a given event (e.g. due to internal policies of the organization in charge of maintaining the building).

Therefore, it is proposed the use of "*unknown*" and "*forbidden*" operators, to allow for a representation of unknown values of an infinite set of values, unknown values of a given set of values, and values not allowed or forbidden. The information contained in each causal tree is then represented in *ELP* by the extensions of a predicates set, being also used as a formalism to quantify the causal tree´s *QoI* (see Section 2.4). The *QoI* allows the identification of the causes that should be taken into account, in first place, and how this hampers all the classification process.

The information obtained in this way to the *RCA* enables automatic report generation with improvements in the recommendations. Figure 2 presents the application of the *EECM* to the adverse event "*study not available*". In the source of this event there is a great diversity of reasons. It is possible that only one situation might be enough for the event to occur or, perhaps, it may be necessary a combination of several factors. The causal trees should include all possible causes and aim to be a generic representation of the problem. For a particular occurrence of the event, its causes will fall on a branch of the tree.

TABLE I.
CATEGORIES OF THE EXTENDED EINDHOVEN CLASSIFICATION MODEL FOR CONSERVATION AND MAINTENANCE OF BUILDINGS AND RESPECTIVE CODES

Category	Description	Code
Technical		
External	Technical failures beyond the control and responsibility of the organization.	TEX
Design	Failures due to the poor design of the building project.	TD
Construction	Construction faults despite a well drawn up building project.	TC
Materials	Failures due to the materials used.	TSR
Structural Response	Failures due to the structural response of the buildings.	TM
…	…	…
Environmental		
Climate	Faults relating with the climate factors that the buildings are subjected to.	EC
Geotechnical	Failures related to geotechnical aspects of the place where the buildings are implanted (soil mechanics).	EG
…	…	…
Organizational		
External	Failures at an organizational level beyond the control of the organization, such as in another department or area.	OEX
Transfer of Knowledge	Failures resulting from inadequate options that do not ensure that the knowledge is transmitted to inexperienced staff.	OK
Protocols	Failures related to the quality/availability of the internal protocols (too complex/simple, unclear, or nonexistent).	OP
Management Priorities	Internal decisions in which safety is relegated to an inferior position reflecting a conflict between productivity and safety.	OM
Culture	Failures resulting from the collective approach and/or risk behaviors.	OC
…	…	…
Human behaviour		
External	Human failures originating beyond the control of the organization, such as in another department or area.	HEX
Knowledge-Based Behavior		
Knowledge-Based Errors	The inability of an individual to apply existing knowledge to a new situation.	HKK
…	…	…
Rule-Based Behavior		
Qualifications	Incorrect fit between an individual's qualifications, training, or education and a particular task.	HRQ
Coordination	Lack of task coordination within a team in an organization (e.g., an essential task not performed because everyone thought that someone else had completed the task).	HRC
Verification	Failures in the correct and complete assessment of a situation before starting the intervention. Includes the relevant conditions of buildings and materials to be used.	HRV
Intervention	Failures that result from faulty planning of task and/or poor execution.	HRI
Monitoring	Failures during monitoring of a activity/process during or after a rehabilitation intervention.	HRM
…	…	…
Skill-Based Behavior		
Slips	Failures in the performance of a task due to the lack of fine motor skills of the technician.	HSS
…	…	…
Other		
Technicians Related Factor	Failures related to physical and/or psychic conditions of the technician that influence the task performance and are beyond the control of the organization.	TRF
…	…	…
Unclassifiable	Failures that cannot be classified in any other category.	X

TABLE II.
A SUBSET OF CODES OF THE EXTENDED EINDHOVEN CLASSIFICATION MODEL FOR CONSERVATION AND MAINTENANCE OF BUILDINGS AND SOME EXAMPLES OF POSSIBLE ADVERSE EVENTS

Code	Extension to the conservation and maintenance of buildings	Examples
TD	Difficulties in the elaboration of projects. Failures sizing.	Lack of details. Overloads not provided. Specifications of recoating improper.
TC	Difficulties in interpreting projects. Lack of inspection.	Armature badly positioned. Lack of cure or cure poorly executed. Concrete with excess of water.
HKK	Difficulties in execution, interpretation or reporting procedures.	Poorly executed procedures. Incomplete procedures. Procedures poorly validated.

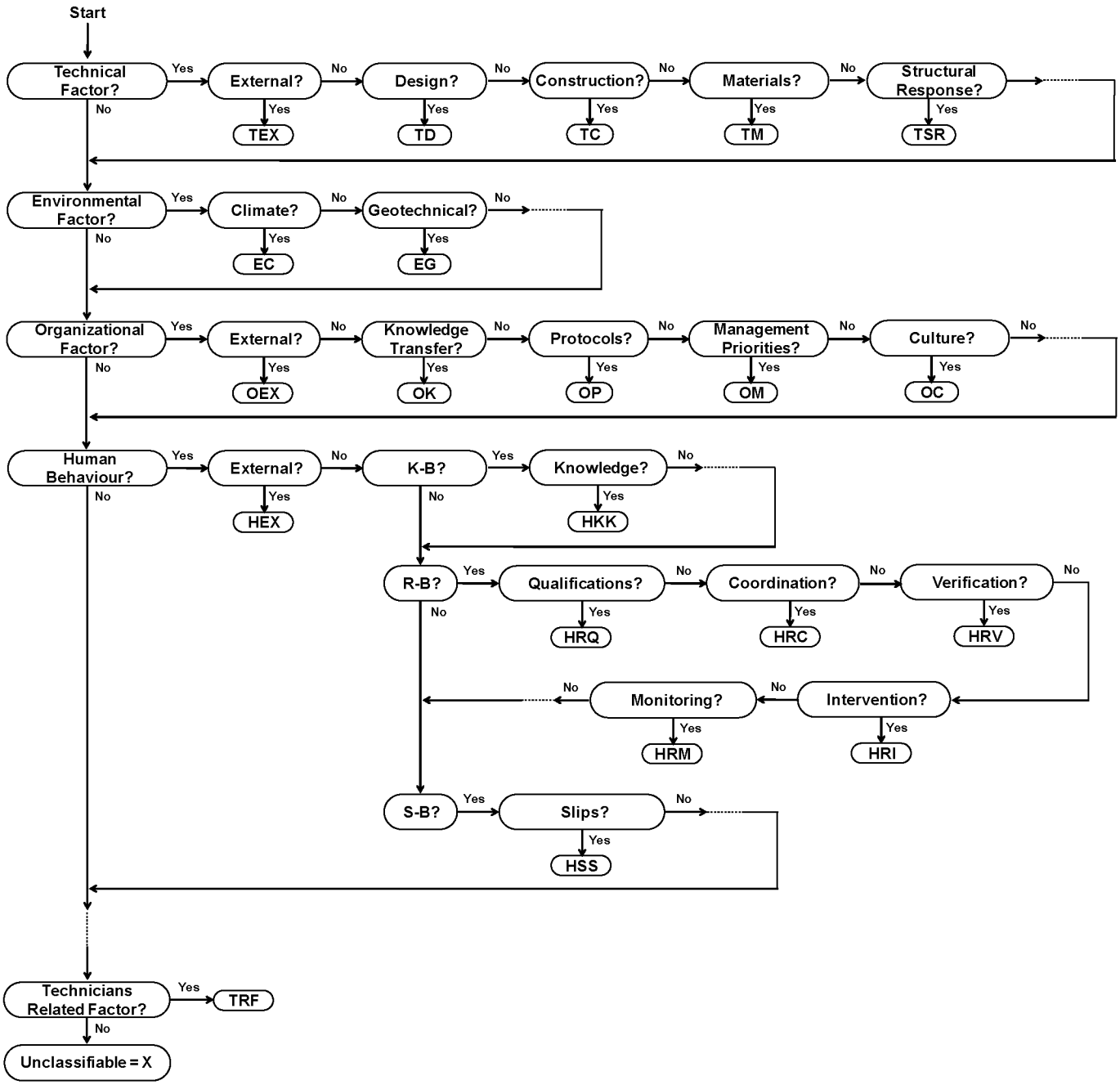

Fig. 1. The Extended Eindhoven Classification Model for Conservation and Maintenance of Buildings flow chart

C. Knowledge Representation and Reasoning

A few decades ago non-classical techniques for modelling the universe of discourse and the reasoning procedures of intelligent systems have been proposed, in addition to the classical ones [8]. Of particular interest to this work are the techniques to deal with incomplete, inconsistent, contradictory, default and forbidden information [9]. Intelligent systems require the ability to reason with incomplete information, since in the real world complete information is hard to obtain, even in the most controlled situations. The idea behind default information is the ability to make assumptions or to jump to a plausible conclusion, derived from a knowledge base in the absence of information to the contrary. The derived information is defeasible, because in light of new information the conclusion may need to be retracted, i.e., we are in the presence of non-monotonic reasoning [9], [10]. A suitable logic is needed, one that permits the representation of incomplete, inconsistent and default information and supports non-monotonic reasoning. In a classical logical theory or logic program, the proof of a theorem (here understood as a question submitted to the classification system) the outcome is a truth value, namely false (0) or true (1), i.e., {0, 1}.

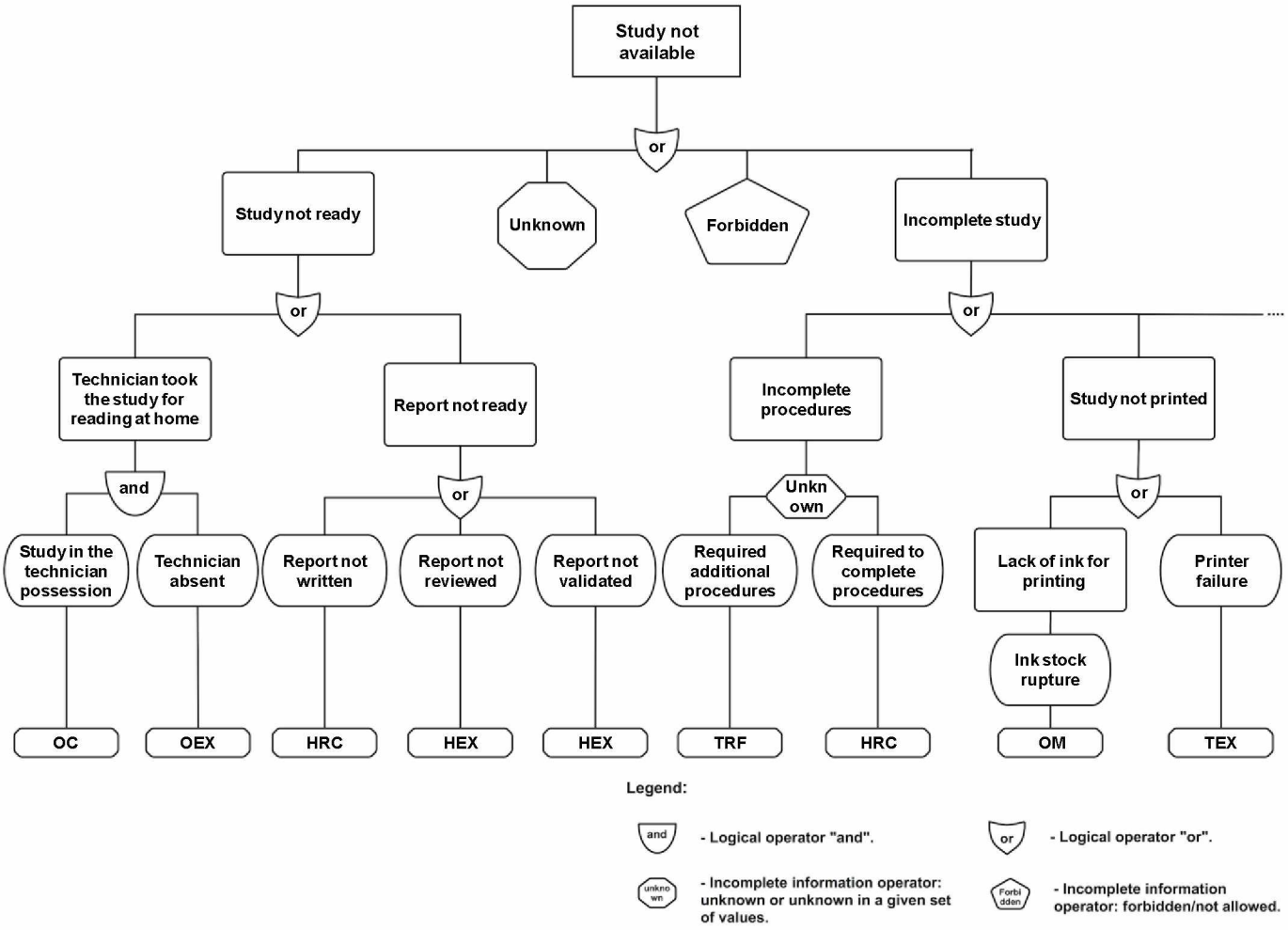

Fig. 2. Extended Causal Tree for the adverse event "*Study Not Available*"

ELP introduces another kind of negation, strong negation, represented by the classical negation sign ¬. In most situations, it is useful to represent ¬*p* as a literal, if it is possible to prove ¬*p*. In *ELP*, the expressions *p* and *not p*, being *p* a literal, are extended literals, while *p* or ¬*p* are simple literals. Intuitively, *not p* is true whenever there is no reason to believe *p*, whereas ¬*p* requires a proof of the negated literal [10].

Every program is associated with a set of *abducibles*, which may be seen as *hypotheses* that provide possible solutions or explanations of given queries, being given here in the form of *exceptions* to the extensions of the predicates that make the logical program or theory. The issue is providing expressive power for representing explicitly negative information, as well as to directly describe the closed world assumption for some predicates, also known as predicate circumscription [11].

Three types of answers to a given question are then possible, i.e., *true*, *false* and *unknown*. The representation of null values will be scoped by the *ELP*. It is possible to consider three types of null values: the former will allow for the representation of unknown values, not necessarily taken from a given set of values, the middle one will represent unknown values taken from a given set of possible values, and the latest will define values that are not allowed or forbidden. Taking the example of the adverse event "*study not available*" (Fig. 2) it might represent all the possible situations according to the following setting:

- It is known that the study was not available because it was in the technician's possession – *known value*;
- The professional that recorded the adverse event only informed that the study report was not ready. It is not possible to be constructive, concerning the action or truth- -value to consider. However, it is false that the action or decision could be different. This situation suggests that the lack of knowledge may be associated to a set of possible known values – *unknown value in a finite set of values* (in this case there are three possibilities, i.e., *report not written*, *report not reviewed* or *report not validated*);

- It is only known that the study was not available. In this case who reported the adverse event did not know which actions or decisions led to the event occurrence – *unknown value*;
- And finally, namely due to internal policies of the organization, it is not permitted to know the causes of a given event – *forbidden or not allowed values*.

Considering the extensions of the predicates that represent the information expressed in a generic causal tree when the *EECM* is applied, where the first predicate denotes the adverse event that was reported (*adverse_event (study not available)*), the second represents an action or decision that led to the adverse event occurrence and the third concerns the root cause that was the primary factor that contribute to the actions and decisions taken and, consequently, to the event occurrence:

- *adverse_event: X*
- *action_or_decision: Y*
- *root_cause: Z*

The knowledge representation in terms of the extension of predicate *action_or_decision*, concerning possible action or decision that leads to the adverse event in the situations presented above, may be depicted by the following programs.

Program 1. Extension of predicate *action_or_decision*, concerning a possible action or decision that leads to the adverse event "*study not available*", with a *known value*.

```
¬ action_or_decision(Y) ←
                    not action_or_decision(Y),
                    not exception(action_or_decision(Y)).
action_or_decision("study in the technician possession").
```

Program 2. Extension of predicate *action_or_decision*, concerning a possible action or decision that leads to the adverse event "*study not available*", with an *unknown value in a finite set of values*.

```
¬ action_or_decision(Y) ←
                    not action_or_decision(Y),
                    not exception(action_or_decision(Y)).
exception(action_or_decision("report not written")).
exception(action_or_decision("report not reviewed")).
exception(action_or_decision("report not validated")).
```

Program 3. Extension of predicate *action_or_decision*, concerning a possible action or decision that leads to the adverse event "*study not available*", with an *unknown value*, were ⊥ stand for a null value of an undefined type.

```
action_or_decision(⊥).
¬ action_or_decision(Y) ←
                    not action_or_decision(Y),
                    not exception(action_or_decision(Y)).
exception(action_or_decision(Y)) ←
                    action_or_decision(⊥).
```

Program 4. Extension of predicate *action_or_decision*, concerning a possible action or decision that leads to the adverse event "*study not available*", with a *value forbidden* or *not allowed*.

```
action_or_decision(forbidden).
¬ action_or_decision(Y) ←
                    not action_or_decision(Y),
                    not exception(action_or_decision(Y)).
exception(action_or_decision(Y)) ←
                    action_or_decision(forbidden)
null(forbidden).
```

Using *ELP*, as the logic programming language, it is now possible to set a procedure given in terms of the extension of a predicate called *demo: question, answer* → [0, 1]. Given a question *(Q)*, it returns a solution based on a set of assumptions, where question indicates a theorem to be proved and answer denotes a truth value (see Program 5; *True* (*1*), *False* (*0*), being *Unknown (U)* in the range of the truth values in the interval]0, 1[).

Program 5. Extension of meta-predicate demo.

```
demo(Q,T) ← Q
demo(Q,F) ← ¬Q
demo(Q,U) ← not Q ∧ not ¬Q
```

D. Quality of Information

The *Quality of Information* (*QoI*) factor with respect to the extension of a generic predicate *p* may be analysed in different contexts and measured in the interval [0, 1]. When the information is known; when the information is unknown; when the information is unknown but can be taken from a set of values. If the information is *known* the QoI_p for the extension of predicate *p* is 1. For situations where the value is *unknown* the QoI_p is given by:

$$QoI_p = \lim_{N \to \infty} \frac{1}{N} = 0 (N >> 0) \quad (1)$$

Finally, if the information is *unknown* but can be derived from a set of values, the QoI_p is set in terms of *1/Card*, where *Card* denotes the cardinality of the *abducibles* set for *p*.

The next element of the model to be considered is the relative importance that a predicate assigns to each of its attributes under observation, i.e., w_{ij} stands for the relevance of attribute *j* for predicate *i*. Assuming that the weights of all predicates are normalized, it is now possible to define a predicate's *scoring function* ($V_i(x)$), i.e., for a value $x = (x_1, ..., x_n)$ in the multi-dimensional space defined by the attributes domains, which is given in the form:

$$V_i(x) = \sum_{j=1}^{n} w_{ij} * V_{ij}(x_j) \quad (2)$$

It is viable to measure the *QoI* that occurs as a result of invoking a logic program to prove a theorem, by posting the

$V_i(x)$ values into a multi-dimensional space and projecting it onto a two dimensional one. Using this procedure, a circle with dashed n-slices can be defined denoting the *QoI* that is associated with each one of the predicate extensions that make the logic program.

As an example the *QoI* associated with the information about the *RCA* of the adverse event "*study not available*", for the first three cases present in the previous section, is given in the form:

- $V_{action\_or\_decision}$ *(former case)* = 1
- $V_{action\_or\_decision}$ *(middle term case)* = 0.33
- $V_{action\_or\_decision}$ *(latest case)* = 0

Thereby it is possible to measure the *QoI* associated to the question put in context, in terms of a logic program that endorses procedures of *action_or_decision*, which may be given in the form *Which are the actions or decisions that led to the adverse event occurrence?*. The shaded n-slices (here n is equal to 3 (three)) of the circle depicted in Figure 3 denote the *QoI*.

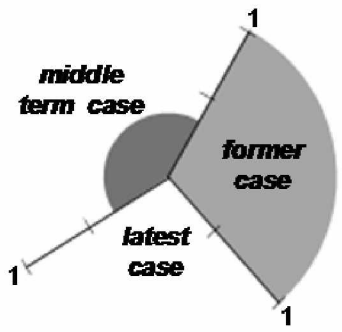

Fig 3. The embedded QoI with respect to the question Which are the actions or decisions that led to the adverse event occurrence?

III. Discussion

Based on the formal approach referred to above, an adverse event reporting and learning system was introduced. Indeed, to the professionals of conservation and maintenance of buildings and mostly to the organizations of the sector, this approach may bring some advantages. After the adverse events have been registered, similar to what happens in other reporting systems, the analysis process becomes easier, more expedite and reliable.

Undoubtedly, with the recourse to *ELP*, leading to an on the fly measurement of the *QoI* of the logic terms used in the process of judgement (in terms of a theorem to be proved), the human intervention in the analyse process is only necessary to approve the recommendations, causes and events that may need attention. It also caters for the credibility and the measurement of the efficacy of the implemented strategies and actions.

Although the causal classification of events is sometimes time-consuming and difficult, with the development of a generic causal tree for each possible event, the increase in time consuming is on the initial phase of the model enforcement.

The *QoI* allows the ordering of causes, identifying the ones that should be taken into account in the first place. In the generic tree it is necessary to consider all possible causes, rather than the most probable or usual ones. The information obtained is useful in identifying possible trends and areas requiring further investigation.

The conceptualized logic model offers the means for knowledge extraction, providing the identification of the most significant causes and suggestions of changes in the organization policies and maintenance procedures, subject to formal proof. Indeed, the creation of an inference system in support of the logical model enables the generation of reports with strategies for quality improvement on time, where a quality measure of the system is on one´s confidence on the results, in terms of the *QoI*.

IV. Conclusion

The main contribution of this work is to be understood in terms of the evaluation of the *QoI* in the RCA and the possibility to address the issue of incomplete information, through the use of an *Extension to Logic Programming* (*ELP*) in the construction of causal trees. *ELP* was used for knowledge representation and reasoning with defective information, catering for the modelling of the universe of discourse in terms of incomplete, inconsistent, forbidden and default data, information and knowledge.

A systematisation of the body of knowledge's evolution about *QoI* embedded in the *RCA* was accomplished. A way to solve the representation problem of defective information was presented, adequate for evaluating the *QoI* in such situations. It was also presented a computationally feasible formal tool to measure the value of *QoI*. With this approach to *RCA* and classification it was possible to identify the causes, actions and decisions that may lead to the adverse events and define the strategies to prevent them.

V. Future Work

In the future an *Adverse Event reporting and learning System applied to the Conservation and maintenance of Buildings* (*AESCB*) will be developed. The *AESCB* will comprise 3 (three) core modules, making it not only a system for adverse event registration, but also a learning system. The *Adverse Event Reporting Forms in Conservation and maintenance of Buildings* (*AERFCB*) module will provide a *Web* interface for adverse event registration.

The effort on this interface will be focused in its usability. The event registration will be made by professionals of the sector of conservation and maintenance of buildings and by those who use the buildings, through predefined forms adapted to each user profile.

The *Adverse Events Manager Reports in Conservation and maintenance of Buildings* (*AEMRCB*) module will be also *Web* based and aims to enable the analysis of the adverse events recorded by *AERFCB*, based on the *Extension of the Eindhoven Classification Model* (*EECM*). The system will provide an individual report for each adverse event recorded, which will include all its details and the extended causal tree obtained using the *EECM*.

The *AEMRCB* module will also provide charts with statistical information about the impact, place of occurrence, type of form and type of event recorded. Finally, the *Adverse Events Knowledge Manager in Conservation and maintenance of Buildings* (*AEKMCB*) module will use the data from the system database to create a *Knowledge Base* (*KB*), which although had been given in terms of *ELP*, will be rewritten to productions in the logic programming language *PROLOG* [12], based on the *EECM*.

From the *KB* other reports relevant to the improvement of the repair strategies may be generated, always with the assurance of data reliability and credibility, by taking into account its *QoI*.

ACKNOWLEDGMENTS

This work is funded by ERDF—European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT—Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project FCOMP-01-0124-FEDER-028980.

REFERENCES

[1] H. Adeli, "Neural Networks in Civil Engineering: 1989–2000," *Computer--Aided Civil and Infrastructure Engineering*, vol. 16, pp. 126-142, 2001.

[2] P. Lu, S. Chen and Y. Zheng "Artificial Intelligence in Civil Engineering," *Mathematical Problems in Engineering*, vol. 2012. [Online]. Available: http://www.hindawi.com/journals/mpe/2012/145974/

[3] D. Dukić, M. Trivunić and A. Starčev-Ćurčin, "Computer-Aided Building Maintenance with "BASE-FM" Program," *Automation in Construction*, vol. 30, 57-69, 2013.

[4] I. Motawa and A. Almarshad, "A Knowledge-Based BIM System for Building Maintenance," *Automation in Construction*, vol. 29, 173-182, 2013.

[5] T. W. van der Schaaf, "Near Miss Reporting in the Chemical Process Industry: An Overview," *Microelectronics Reliability*, vol. 35, 1233-1243, 1995.

[6] T. W.van der Schaaf and M. Habraken, "PRISMA-Medical: A Brief Description," Eindhoven University of Technology, Faculty of Technology Management, Patient Safety Systems, 2005. [Online]. Available: http://www.who.int/patientsafety/taxonomy/PRISMA_Medical.pdf

[7] S. Rodrigues, P. Brandão, L. Nelas, J. Neves, and V. Alves, "A Logic Programming Approach to Medical Errors in Imaging," *International Journal of Medical Informatics*, vol. 80, 669-679, 2011.

[8] F. Sheridan, "A Survey of Techniques for Inference under Uncertainty," *Artificial Intelligent Review*, vol. 5, 89-119, 1991.

[9] M. L. Ginsberg, *Readings in Nonmonotonic Reasoning*. San Francisco: Morgan Kauffman Publishers Inc, 1987.

[10] J. Neves, "A logic interpreter to handle time and negation in logic data bases," in Proceedings of the 1984 annual conference of the ACM on the fifth generation challenge, R. L. Muller and J. J. Pottmyer, Eds. New York: Association for Computing Machinery, 1984, pp. 50-54.

[11] U. Hustadt, "Do we need the Closed-World Assumption in Knowledge Representation?," in Working Notes of the KI'94 Workshop – Reasoning about Structured Objects: Knowledge Representation meets Databases (KRDB'94), F. Baader, M. Buchheit, M. A. Jeusfeld and W. Nutt, Eds. Saarbrüken: German Research Center for Artificial Intelligence, 1994, pp. 24-26.

[12] I. Bratko, *PROLOG Programming for Artificial Intelligence*. Toronto: Pearson Education Canada, 2011.