# A Generic Architecture for Knowledge Acquisition Tools in Cardiology

**Káthia Maral de Oliveira**[1], **Antônio A. Ximenes**[2], **Stan Matwin**[3], **Guilherme Travassos**[1] and **Ana Regina Rocha**[1]

**Abstract.** Knowledge-acquisition is well known to be a bottleneck activity in the development of knowledge-based systems. Several techniques and tools were proposed to support this process. However, knowledge engineers still have difficulties to understand the problem domain, to apply these techniques and to interact with the experts. Considering that domain-specific tools can be useful for knowledge acquisition, we defined a generic architecture of knowledge acquisition tools and, based on that, we built KED, a Knowledge Editor for Diagnosis in the cardiology domain. KED is part of a general environment that aims at supporting the software development in cardiology domain.

## 1 INTRODUCTION

Knowledge acquisition is one of the longest and most difficult activities in the development of knowledge-based systems [8]. We could verify this while developing SEC [7], an expert system for diagnosis of acute myocardial infarction. At the beginning of this project the computer science team had a one-day course on basic cardiology concepts to make it possible to start the knowledge acquisition process. Although we used various techniques, we had several problems like the difficulty to schedule a meeting with the experts, or the difficulty to interact with them using their own jargon. One way to assist in this process is to use knowledge acquisition tools. In this context, domain-specific tools [3] are very useful because they can interact directly with the experts using the terminology of the domain and provide basic knowledge for the knowledge engineers in the knowledge acquisition process.

With the goal of developing new expert systems and believing in the importance of assisting in the knowledge acquisition process by using domain-specific tools we defined a generic architecture for knowledge acquisition tools for the cardiology domain. The basic feature of this architecture is to organize independently the domain knowledge and the tasks. Using this architecture we built KED, a knowledge editor for cardiology specialized in diagnosis.

KED is one of the tools of an environment that provides support for the construction, management and maintenance of software products for Cardiology (see [10]). This environment uses embedded domain-knowledge to guide the software developers across the several phases of the software process. It is made of a set of domain-specific tools like KED. These tools differ from more classical ones because they use the vocabulary of the domain (specified in the domain ontology) for all their interactions.

In the following sections, we will first describe how the cardiology knowledge is organized in our environment and how the domain-specific tools use it (section 2). In section 3, we describe the generic architecture of the knowledge acquisition tools exemplified with KED. Finally, in section 4, we present our conclusions and future works.

## 2 THE DOMAIN-KNOWLEDGE

Knowledge acquisition involves knowledge elicitation and representation. Ontology [1, 2] can be used during this process to facilitate the communication between the experts and the knowledge engineers by establishing a common vocabulary and a semantic interpretation of terms. Ontology is defined as a coherent set of representation terms, together with textual and formal definitions, that embodies a set of representation design choices. It can be used in knowledge-acquisition tools that directly interact with domain experts and effectively avoid errors in the acquired knowledge by constantly verifying constraints on the form and content of the knowledge. Ontology can also provide sharability of knowledge bases and knowledge organization.

To define the ontology for cardiology we used a methodology defined in [5]. To simplify this activity and better organize the domain model, we divided the cardiology domain in sub-domains. Each sub-domain has a group of concepts and relations among them sharing the same semantic context. The sub-domains also have relations between themselves to compose the whole domain. We have identified the following sub-domains: (i) heart anatomy (concepts about the heart structure and the physiology); (ii) findings (concepts used in the physician's investigation process); (iii) therapy (kinds of therapies and their features); (iv) diagnosis (concepts and characteristics that identify syndrome and etiology diagnoses); and (v) pathologies (representing different situations of heart components whose classification and features are important for the purpose of the domain theory of cardiology in the environment). Figure 1 shows some of the concepts of these sub-domains. The ontology was validated by cardiologists and formalized using first order logic and Prolog.

We also identified potential tasks of the domain and mapped them with the domain knowledge. These tasks represent activities such as diagnosis or interpretation that happen in the domain, but are domain-independent (e.g. diagnosis of diseases and diagnosis of machine failures). They are important to specify tools. For cardiology

[1] COPPE / UFRJ - Graduate School of Engineering - Federal University of Rio de Janeiro. Cx Postal 685111 Cep 22945-970. Rio de Janeiro - RJ - Brazil - email: kathia, ght, darocha@cos.ufrj.br

[2] UCCV/FBC- Unit of Cardiology and Cardiovascular Surgery - Fundao Bahiana de Cardiologia - Federal University of Bahia - Rua Augusto Viana s/n Cep 40140-060 Brazil

[3] SITE - School of Information Technology and Engineering University of Ottawa - Canada - email: stan@csi.uottawa.ca

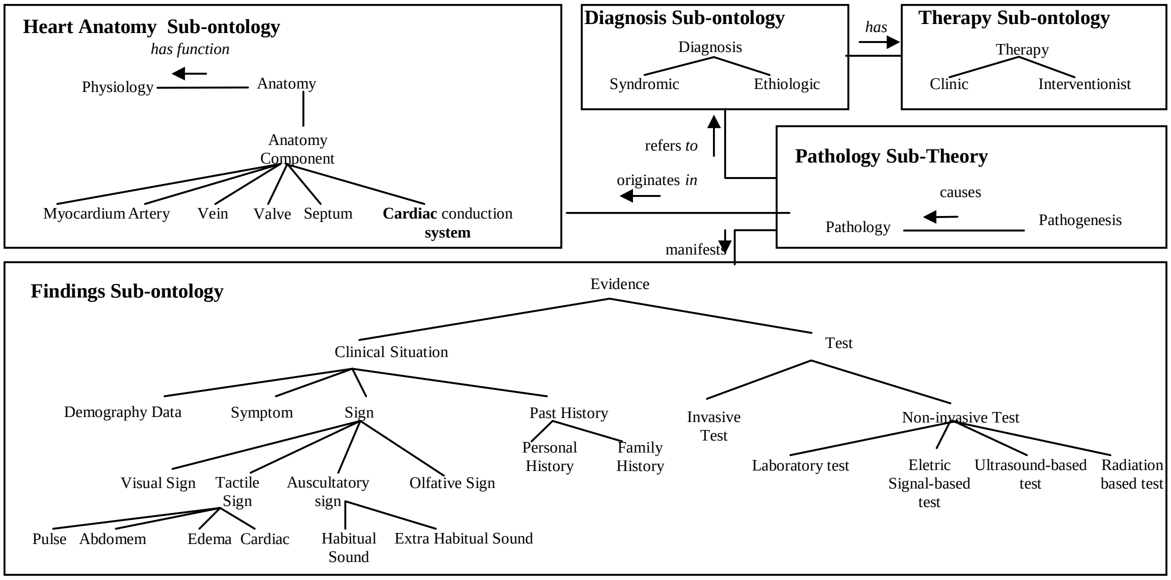


**Figure 1.** Domain Knowledge for Cardiology

we consider the following main tasks: diagnosis, therapeutic planning, simulation and monitoring. The mapping between the task and the ontology gives an idea of the concepts more closely related to each task. For the diagnosis task, for instance, it is important to consider the findings, pathology and diagnosis sub-ontologies. Similarly, for therapeutic planning, the important sub-ontologies are therapy and pathology; for monitoring, the heart anatomy and findings sub-ontologies; and for simulation, the heart anatomy sub-ontology.

## 3 KED: A KNOWLEDGE EDITOR FOR DIAGNOSIS IN CARDIOLOGY

Using this domain ontology, we defined a generic architecture to build knowledge acquisition tools. This architecture is based on interactions with the experts (asking them to enter cases) and is composed of three levels (Figure 2): the Knowledge Representation Level, the Generic Process Level and the Operational Level. The *Knowledge Representation Level* is the ontology for cardiology defining the structure and content of the knowledge domain. The *Generic Process Level* is a model of the task specific to each tool (e.g. for KED it will be diagnosis). The *Operational Level* records a set of cases, used to define the generic process model and to find out the knowledge. The tools following this generic architecture can ask information about cases to the expert using the domain language (concepts from the ontology). These cases are stored in a database and can be used later for testing. They are also used to instantiate the domain ontology and, if possible, to define general rules for the specific task.

Using this architecture we built KED, the knowledge editor for cardiology, specialized in diagnosis task. Figure 3 shows KED's conceptual model for the three-level architecture. We can see that the first level consists in the domain sub-ontologies related to the diagnosis task (findings, pathology and diagnosis). The Generic Process Level and the Operational Level are based on analysis patterns defined by Fowler [4]. We also consider the generic task model of systematic diagnosis by causal tracing proposed by KADS [6] to better understand and define the types in the Generic Process Level. In this model, concepts are caused by other concepts and these in turn by others, etc., in a causal hierarchy. This hierarchy is represented in the Generic Process Level by defining general rules (*Norms*) for diagnosis using the cases (set of *qualitative* and *quantitative observations* from the patient) entered in the Operational Level. These rules are formed by justifications of the diagnoses entered by the cardiologists, they are simple diagnostic associations. While entering new justifications the tool generalizes them with the previous ones, so that, in the end, some general rules for the diagnosis of the pathologies are available for future use in interviews during the elicitation process.

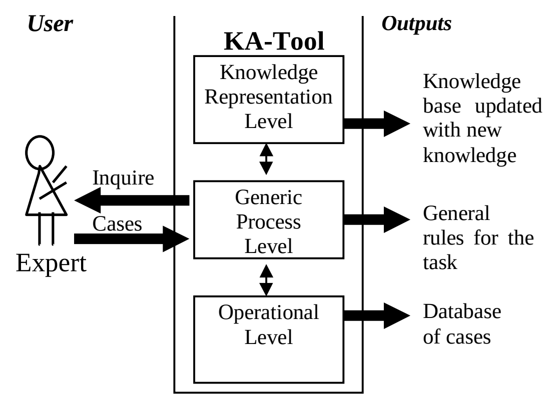


**Figure 2.** Three -level architecture of the Knowledge Acquisition Tool

KED supports knowledge elicitation in five steps. In the first one, it inquires *observations* from one patient case using the information from the ontology. In the second step, the expert enters a diagnostic for that patient specifying which *observations* are significant for his decision. The third and fourth steps are performed starting from the second case. In this situation, KED analyzes the important observations (chosen in the second step) and generalizes the previous associations (defined for others patient cases for the same pathology) using simple machine learning rules [9]. Then, KED shows this generalization and the cardiologist validates it (fourth step). If he/she disagree,

**Table 1.** Example of Knowledge Elicitation with KED

| | Case 1 | Case 2 |
|---|---|---|
| 1st step (expert enters the required information) | Sex =Male, Age=45, Weigh=70, Symptom=chest pain | Sex=Female, Age=60, Weigh=55, Symptom=chest pain |
| 2nd step (expert gives the diagnosis and chooses the significant observations) | Diagnosis Acute myocardial infarction, Sex=Male, Age=45, Symptom=chest pain | Diagnosis Acute myocardial infarction, Sex=Female, Age=60, Symptom=chest pain |
| 3rd step (Ked generalizes the cases entered) | | (Sex=female or Sex=male) and Age≥45 and Symptom=chest pain |
| 4th step (expert evaluates the generalization) | | Disagree |
| 5rd step (expert provides justifications) | Sex=Male and Age≥45 and Symptom=chest pain | Sex=female and Age≥55 and Symptom=chest pain |

KED learns that this generalization should never be proposed again. Finally, the expert gives justifications for that case (that will be generalized in the next cases) and new justifications when he/she did not agree with the generalization in the previous step. Table 1 shows a simple example of this process considering the information sex, age, weigh and symptom. The generalization in the third step was done with the justification provided in case 1. This is just one kind of generalization that the KED uses. In the fifth step the expert provides an association to correct the one gave by the system.

The associations can be translated to Prolog to be used as a first draft knowledge base. They can also be used, by the knowledge engineers, to continue the knowledge acquisition.

KED was validated with several cases collected at UCCV/FBC and considering associations provided by cardiologists.

## 4 CONCLUSION

Knowledge acquisition tools are an essential support for the development of knowledge-based systems. They can be used directly by the experts and can provide a valuable starting point for the knowledge engineers in the knowledge elicitation process. This paper presents a generic architecture for building knowledge acquisition tools in the cardiology domain along with a specific example: KED, the Knowledge Editor for Diagnosis in Cardiology. The generic architecture relies on a single domain ontology for cardiology and a model of the desired task for each knowledge acquisition tool. In this paper we present how we organize the knowledge and how these tools interact by describing KED.

We are working on this tool to manage conflicting associations defined by the experts. Besides this, we will define tools for other tasks to help in the development of expert systems to be used in intelligent tutors under development at UCCV/FBC.


## ACKNOWLEDGEMENTS

We thank Dr. Alvaro Rabelo for his valuable contribution and CNPq, the Brazilian Government financial support for this project.


## References


[1] A. Abu-Hanna and W. Jansweijer, 'Modelling domain knowledge using explicit conceptualization', *IEEE Expert*, 53–63, (oct 1994).

[2] B. Chandrasekaran, R. Johsephson, and V. R. Bejamins, '"what are ontologies, and why do we need them?"', *IEEE Software*, 20–27, (Jan/Feb 1999).

[3] H. Eriksson and M. Musen, 'Metatools for knowledge acquisition', *IEEE Software*, 23–29, (may 1993).

[4] M. Fowler, *Analysis Patterns: Reusable Object Models*, Addison-Wesley, 1997.

[5] A. Gómez-Pérez, M. Fernandez, and A. J. Vicente, '"toward a method to conceptualize domain ontologies"', in *Proceedings of Workshop on Ontological Engineering / ECAI96*, (Aug. 1996).

[6] F. Hickman, J. Killin, L. Land, T. Mulhall, D. Porte, and R. M. Taylor, *Analysis for Knowledge-Based Systems: A practical Guide for KADS, Methodology*, Ellis Horwood, 1992.

[7] Rabelo Jr, A. R. Rocha, and K. M. *et al* Oliveira, 'An expert system for diagnosis of acute myocardial infarction with ecg analysis', *Artificial Intelligence in Medicine*, **10**, 75–92, (1997).

[8] K. McGraw and K. Harbison-Briggs, *Knowledge Acquisition: Principles and Guidelines*, Prentice Hall, 1989.

[9] R. S. Michalski, *Machine Learning - An Artificial Intelligence Approach*, volume I, chapter A Theory and Methodology of Inductive Learning, Morgan Kaufmann Publishers, 1986.

[10] K. M. Oliveira, A. R. Rocha, and G. H. Travassos, 'A domain-oriented software development environment for cardiology', in *Proceedings of America Medical Informatics Association conference - AMIA*, Washington, D.C., USA, (Nov. 1999). AMIA.


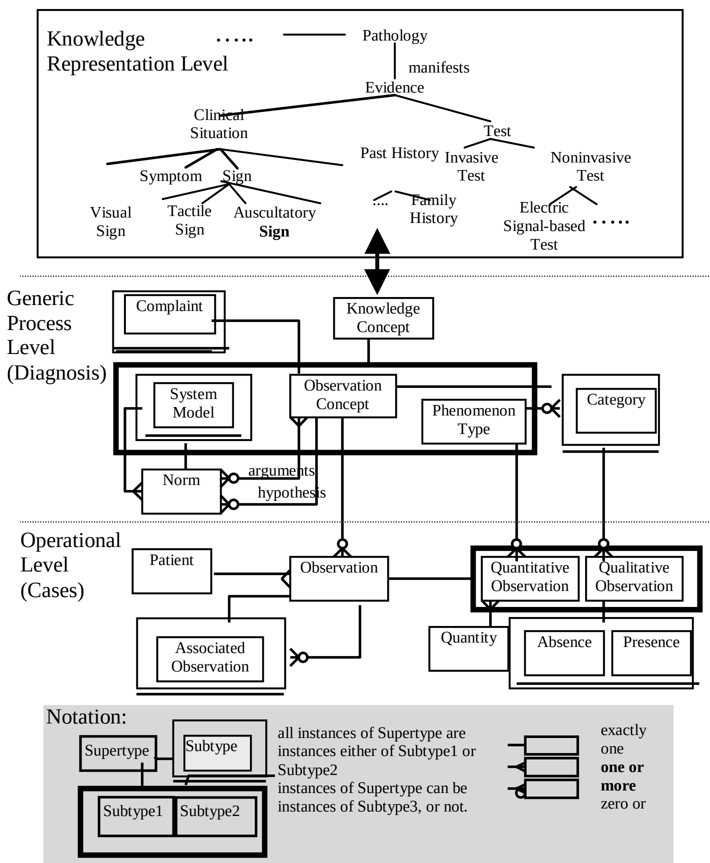


**Figure 3.** KED Conceptual Model