

Using Electronic Documents for Knowledge Acquisition and Model Maintenance

Martin Molina and Gemma Blasco

Department of Artificial Intelligence, Universidad Politécnica de Madrid,
Campus de Montegancedo S/N 28660 Boadilla del Monte, Madrid, Spain
mmolina@fi.upm.es, gblasco@isys.dia.fi.upm.es

Abstract. Knowledge acquisition and model maintenance are key problems in knowledge engineering to improve the productivity in the development of intelligent systems. Although historically a number of technical solutions have been proposed in this area, the recent experience shows that there is still an important gap between the way end-users describe their expertise and the way intelligent systems represent knowledge. In this paper we propose an original way to cope with this problem based on electronic documents. We propose the concept of intelligent document processor as a tool that allows the end-user to read/write a document explaining how an intelligent system operates in such a way that, if the user changes the content of the document, the intelligent system will react to these changes. The paper presents the structure of such a document based on knowledge categories derived from the modern knowledge modeling methodologies together with a number of requirements to be understandable by end-users and problem solvers.

1 Introduction

The recent experience in the creation and maintenance of large and complex models of intelligent systems reveals a set of problems that should be solved to improve productivity and efficiency in knowledge engineering. In the field of Artificial Intelligence, a particular concept of software tools were proposed with the name of *knowledge acquisition tools* (KA tools) specialized in establishing a understandable dialogue with an expert user to help her/him in the formulation of a knowledge model. Examples of these tools are SALT [1], MOLE [2] and EXPECT [3] that show different levels of assistance based on the tool's expertise about the structure of the model.

However, when the complexity and dimension of the knowledge model is important the specialized tools present significant difficulties mainly derived from the fact that there is still an important gap between the way end-users describe their expertise and the language used by KA tools. More recently, AI researchers in the context of web-based applications and semantic web have paid more attention to this problem, in order to facilitate the creation and maintenance of knowledge bases by non-experts in computers [4].

Problem	Explanation
<i>Dimension and complexity</i>	The knowledge model presents a high level of complexity with different interrelated types of knowledge for different purposes (e.g., 143 KBs for a particular case).
<i>Heterogeneity of symbolic representation</i>	Each type of knowledge base has its own symbolic representation (frames, rules, uncertainty, temporal and spatial dimensions, etc.). Despite they are based on natural and declarative representation, this factor increases the difficulty of understanding the complete model.
<i>Low level of certain representations</i>	For certain types of knowledge, persons use certain common sense usual in their professional area, but the corresponding knowledge base may use low level representations to represent such a knowledge with excessive detail about implicit terms that makes the model more artificial and difficult to understand.
<i>Abstract computer-oriented terminology</i>	The KBs model follows a general knowledge engineering methodology. This usually introduces an additional terminology closer to process information, different from hydrology, that sometimes is too abstract for end-users and increases the difficulty to understand the complete model.
<i>Consistency between modules</i>	The model presents a distributed organization of knowledge (e.g. a multiagent architecture) and, for each module there is a set of knowledge bases with different inference procedures. Sometimes it is difficult anticipate the dependencies between modules in order to keep the global consistency.
<i>False idea of procedural representation</i>	The edition of the content of knowledge bases uses text processors with the corresponding language for each case. We found that this may give the false idea that the user writes a kind of procedure (following a conventional programming style) instead of a set of expertise criteria with a declarative approach.
<i>Different procedures for knowledge acquisition</i>	Another problem is that the user must combine different non-integrated software tools to cope with different sources of knowledge. Thus, for example, certain knowledge can be manually represented using symbolic formalisms, but another type of knowledge can be learned with machine learning procedures.
<i>Low level of guidance</i>	The user has the possibility of editing and modifying any part of the model with certain freedom. However, this freedom should be complemented with certain guidance in the model development, to suggest to the developer about what are the next steps to be done.

Fig. 1. Problems reported during the maintenance of a knowledge model.

As an example of this, figure 1 shows a summary of problems reported by end users corresponding to an intelligent system, called SAIDA [5] that was developed as a tool for decision making in the field of hydrologic emergency situations following a knowledge-based multiagent approach with a variety of symbolic representations (with more than 140 domain knowledge bases for a particular realization). These problems show that, although the final system provides certain freedom for knowledge model maintenance based on the use of symbolic representations, still, the languages managed by the system use certain structuring principles and formalisms that are artificial for users that are not familiarized with computer oriented terminology.

2 Knowledge Acquisition Tools as Intelligent Document Processors

In order to facilitate the process of model creation and maintenance, our approach changes the type of communication with the end user. Thus, instead of considering the user interface of the knowledge acquisition tool as a set of canned windows representing forms that must be filled by the user, together with a set of warning and

help windows that try to supervise the process and guide the user, we propose an alternative communication mechanism based on the concept of an electronic document. The basic idea is that when the user creates/maintains a knowledge model, she/he reads and writes a document in the same way that reads and writes a conventional document using standard text processors. A document is an entity very familiar to persons non specialized in computer science, so this a natural media to formulate the content of a model and it has been assumed and extended in the context of web-based applications with electronic features (hyper-links, multimedia, etc.).

Section / subsection		Content	
Introduction		static	Goals of the document, context, motivation
Terminology	General Terms	static	System concepts, symbolic representation, example, concept requirements
		user	User general concepts, general concept requirements
		automatic	Roles in inferences, attribute dependencies, types of attributes (static/dynamic, internal/external, etc), class hierarchy, requirements satisfaction
		user	User domain concepts, domain concept requirements
		automatic	Roles in inferences, attributes dependencies, types of attributes (static/dynamic, internal/external, etc.), class hierarchy, statistics (N. of concepts, etc.), requirements satisfaction
Model overview		static	Goals, task decomposition, domain knowledge organization, general model requirements
		automatic	Summary of tasks, domain knowledge statistics (N. of relations, etc.), requirements satisfaction
Task 1	Procedural view	static	Goal, subtasks, i/o roles, task structure, algorithm, example
	Domain knowledge	static	Set of knowledge bases used by the task
		automatic	Statistics (N. of relations, etc.)
...
Task N
Inference 1	Procedural view	static	Goal, i/o roles, algorithm, example
	Domain knowledge	static	content description, symbolic representation (deductive tables, functions, etc.), system concepts, domain knowledge requirements, example
		user	User domain concepts, occurrences of relations, domain model requirements
		automatic	Precedence relations, attribute dependencies, quality measures (consistency degree, etc.), statistics (N. of relations, etc), requirements satisfaction
...
Inference M
Summary		automatic	Summary of requirements satisfaction

Fig. 2: General view of the format of an electronic document for knowledge acquisition.

In order to use electronic documents in knowledge acquisition, we understand that the document is like a manual that describes problem solving methods for decision making procedures based on certain expertise. The document must accomplish the following three basic requirements:

- *Human understandability.* The content of the document must be totally comprehensible by users non-experts in computer science. A way to guarantee this assumption is that the complete set of procedures described by the document should be able to be performed manually by a user specialized in the domain of the problem.

- *Computer understandability.* The content of the document must be also comprehensible by computer programs that, based on this content, automatically perform the decision making procedures for which the knowledge based system has been designed.
- *Dynamic content.* The content of the document must be able to be modified by the user. For this purpose, in a document we consider three types of parts: (1) *static*, that describe prefixed areas assumed by computer programs that manipulate the document, (2) *user*, dynamic areas that can be edited and modified by users, and (3) *automatic*, dynamic areas that are automatically generated by processes applied to information provided by the user.

According to this, the knowledge acquisition tool is considered as an *intelligent document processor*, i.e. a text processor specialized in the domain of the knowledge based system that allows the user to write the knowledge base as an electronic document. The intelligent text processor provides the following services: (1) *user interaction* to present the document to the user (with text, graphics and other multimedia resources) with facilities to modify the content, (2) *consistency checking* to guarantee the consistency of the model applying both syntax and semantic validation (the semantic validation is based on the processor specialization in the domain knowledge), (3) *changes propagation*, to produce complementary views of the model, (4) *document translation*, to generate operational versions of the document using symbolic languages (rules, frames, bayesian networks, logic clauses, etc.) to be processed by intelligent systems, (5) *development assistance*, to suggest what are the next steps to be done during the model construction .

In order to accomplish the requirement for human understandability, it is important that the document presents a structure with the different issues about the problem that need to be understood by persons. For this purpose, we consider the categories of knowledge established by knowledge engineering methodologies and tools (for instance, CommonKADS [6], KSM [7], etc.). In addition to that, we distribute the domain knowledge and we add complementary views at different levels of abstraction to facilitate global views of the model. Figure 2 shows the preliminary format of an electronic document for knowledge acquisition based on this idea.

During the creation/modification of the knowledge model is useful that the acquisition tool provides guidance to the user in order to know what is the next step (or remaining steps) to be done. For this purpose, we use what we call *model requirements*, i.e. affirmations about the form of the knowledge model that can be evaluated to check to what extend the model is complete. This is a kind of meta-knowledge and can be either formulated by the user or prefixed by the knowledge based system.

Another important factor to make the document understandable by humans is that the type of symbolic formalism used to represent domain knowledge must be familiar to persons non-expert in computers. Thus, instead of including complex algorithms and/or low level symbolic representations, the document must follow natural representations such as tables, mathematical functions, etc. typically used in the professional domain of the problem. In general, there is an open library of graphical resources to formulate certain types of relations such as: deductive tables, probabilistic causal relations, patterns, functional representation, etc.

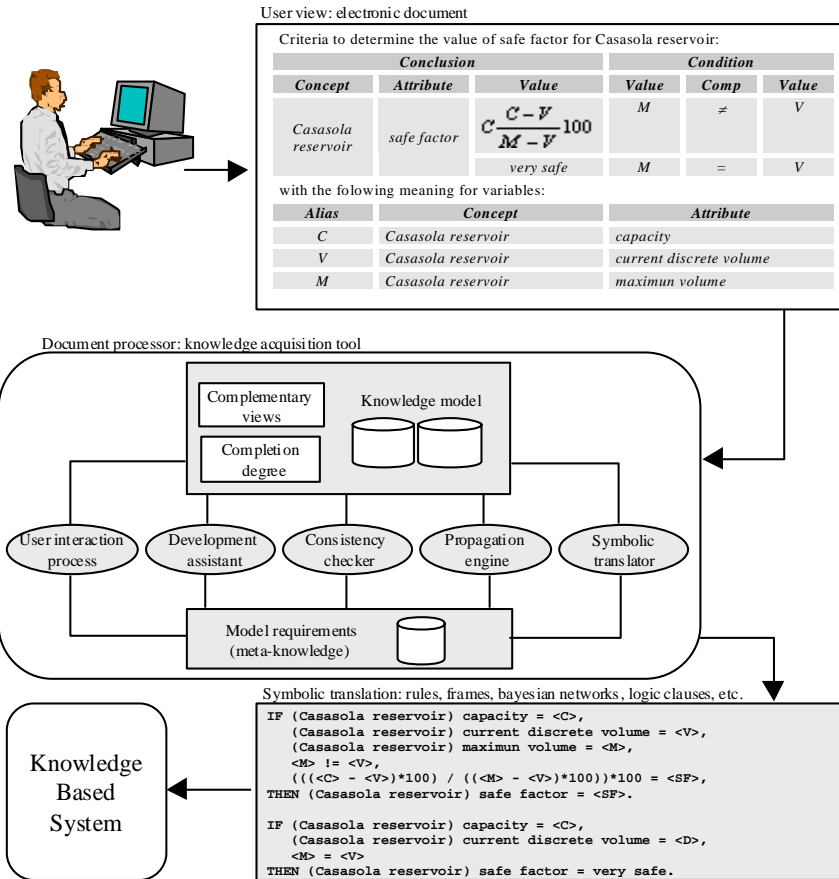


Fig. 3: Global view of the operation with the knowledge acquisition tool.

Finally, in order to be operational (understandable by programs) the content of the document must be translated to conventional symbolic structures to be manipulated by problem solvers, such a rules, frames, logic clauses, etc. Figure 3 shows the correspondence of the user-oriented representation and the operational version. In this case, a table with formulas is translated to rules.

3 Discussion

The idea of assisting end-users, non-experts in computers, in the creation and maintenance of knowledge bases has been treated since the initial shells until the modern knowledge acquisition tools. However, in most of these cases, the user receives a limited perception of the knowledge base, that is presented as a kind of data base that must be written with the help of prefixed windows. This approach does

not guarantee that the user has a global view and appropriate guidance to successfully create and maintain large and complex models. In addition to that, the KA tool normally assumes certain expertise of the end user, that make more difficult to be used by new users, and also, the tool sometimes uses certain structuring principles that are not familiar for end-users. An alternative way for knowledge acquisition is to use quasi-natural language sentences in a directed dialogue. This approach could be useful for knowledge bases with uniform representation and limited size but it is difficult to be followed in a complex knowledge based system. Our approach differs from the previous ones in an alternative communication media, the electronic document, which is familiar with persons non experts in computer science. The document includes certain format requirements oriented to make comprehensible by end-users together with automatic sections to provide guidance.

Recently, within the web-based applications has emerged the concept of *semantic web* [8]. This approach is oriented to have electronic web documents that are not only understandable by persons by also by programs using standard *ontologies* [9] that establish certain common meaning about the potential content of documents. This idea is similar to our approach in the sense that both use the concept of document that should be understandable by humans and machines. However, the idea in semantic web establishes that programs are able to understand the *kind* of information in the document but not the information itself, for any kind of document. On the contrary, our approach is limited to documents that represent knowledge models for intelligent systems, and it establishes a particular format with areas that correspond to knowledge bases that are understood by problem solvers.

Our approach extends the idea of *text processor* to be used as a tool for knowledge acquisition, so it is influenced by standard text processors that can be found in conventional software. We conceive a more advanced text processor that, besides the conventional user interaction for document edition with syntax validation, it is specialized in a particular format and content of document. Thus, the processor can guide the user and verify the content using certain semantic features corresponding to the domain of the knowledge based system. In addition to that, we also have found certain similarities of our approach with the idea of *intelligent product manuals* [10]. This approach uses the concept of electronic document to communicate knowledge about a system or device, as a manual. The idea is that the document serves as user interface to explain how to manipulate the device and uses a knowledge based system to answer certain questions to the user. In contrast to our approach, the document is not the knowledge base and it cannot be changed by the user to modify the behavior of the knowledge based system.

4 Conclusions

In summary, this paper presents an original contribution to improve knowledge acquisition and model maintenance of intelligent systems. In contrast to the traditional knowledge acquisition tools based on conventional user interfaces, we propose the idea of electronic document as basic media for knowledge acquisition. Under this approach, the knowledge acquisition tool is like an intelligent document

processor that allows the end user to read/write large and complex knowledge bases with automatic assistance (consistency checking, guidance, etc.).

This concept of tool has been specifically treated in our group with the case of a complex knowledge based system (SAIDA) that needs to be maintained by end-users. This system follows a knowledge-based multiagent approach with several symbolic representations and with, for example, 143 knowledge bases in a particular case. For this system we have developed the CAM-Hidro tool within the RIADA project, which was used as preliminary KA tool to establish the approach presented in this paper.

Acknowledgements. The work presented in this paper has been supported by the Ministry of Science and Technology of Spain within the RIADA project, and with the help of the Ministry of Environment of Spain (*Dirección General de Obras Hidráulicas y Calidad de las Aguas*) together with local public organizations of river basins (*Conf. Hidrográfica del Júcar* and *Conf. Hidrográfica del Sur de España*).

References

1. Marcus S., McDermott J.: "SALT: A knowledge acquisition language for propose-and-revise systems". *Artificial Intelligence*, 39(1) 1-38, 1989.
2. Eshelman L. "MOLE: A knowledge acquisition tool for Cover-and-Differentiate systems" in Marcus S. (ed.): "Automating knowledge acquisition for expert systems" Chap. 3, Kluwer Academic Publishers, 1988.
3. Blythe J., Kim J., Ramachandran S., Gil Y.: An Integrated environment for knowledge acquisition. International Conference on Intelligent User Interface. Proceedings of the 2001 International Conference on Intelligent User Interfaces (UI-2001), Santa Fe, New Mexico, January 2001.
4. Hendler J., Feigenbaum E.A.: "Knowledge Is Power: The Semantic Web Vision" in *Web Intelligence: Research and Development*, N.Zhong, Y.Yao, J.Liu, S.Ohsuga (eds.), Lecture Notes in Artificial Intelligence 2198, Springer, 2001.
5. Molina M., Blasco G: "A Multi-agent System for Emergency Decision Support". Proceedings of the Fourth International Conference Intelligent data Engineering and automated Learning IDEAL 2003. LNCS, Springer.
6. Schreiber G., Akkermans H., Anjewierden A., De Hoog R., Shadbolt N., Van de Velde W., Wielinga B.: "Knowledge engineering and management. The CommonKADS methodology" MIT Press, 2000.
7. Cuenca J., Molina M.: "The role of knowledge modelling techniques in software development: a general approach based on a knowledge management tool" *International Journal of Human-Computer Studies*. No. 52. pp 385-421. Academic Press, 2000.
8. Barnes-Lee T, Hendler J. and Lassila O. "The Semantic Web." *Scientific American*, May 2001.
9. T. R. Gruber. "A translation approach to portable ontology specifications". *Knowledge Acquisition*, 6(2): 199-221, 1993.
10. Pham, D.T., Dimov, S.S. and Peat, B.J.: "Intelligent Product Manuals". Proceedings of the Institution of Mechanical Engineers, Part B, *Journal of Engineering Manufacture*, Vol. 214, No. B5, pp. 411-419. 2000.