How can Knowledge Acquisition benefit from Terminology ?

Nathalie Aussenac-Gilles, Didier Bourigault, Anne Condamines, Cécile Gros

How Can Knowledge Acquisition Benefit from Terminology ?

NATHALIE AUSSENAC-GILLES*, DIDIER BOURIGAULT**, ANNE CONDAMINES***, CECILE GROS**

* Institut de Recherches en Informatique de Toulouse (IRIT), URA 1399 du CNRS - UPS - 118, route de Narbonne, 31062 Toulouse Cedex, FRANCE

** Electricité De France (EDF), Direction des Etudes et Recherches, 1, avenue du Général de Gaulle, 92141 Clamart Cedex, FRANCE

*** Equipe de Recherches en Syntaxe et Sémantique (ERSS), URA 1033 du CNRS, UTM, Maison de la recherche 5, Allées Antonio Machado, 31058 Toulouse Cedex, FRANCE

Topic : Knowledge acquisition from natural language

Key words : Knowledge acquisition and modelling, terminology, hypertexts

Abstract :

Our aim is to identify and suggest solutions to some methodological problems raised by the exploitation of terminology in knowledge acquisition. through a cross-disciplinary study involving AI, terminology and linguistic engineering. We mention three possible kinds of contributions. of terminology to knowledge acquisition : the terminologies and terminological knowledge bases ; the tools designed for text analysis ; the methodologies applied by terminologists. Then, we report on an experiment where we used a terminological tool, LEXTER, for knowledge acquisition with the MACAO methodology. This experiment stressed some limitations in MACAO and enabled us to test a possible methodological integration. We propose new definitions in the knowledge modelling environment and expose our perspectives to validate them and our hypotheses.

0. INTRODUCTION

For some time now, some terminologists have been claiming that knowledge acquisition should take better account of terminology ; more rarely, knowledge engineers have encouraged their colleagues to pay attention to results of research in terminology. For Skuce and Meyer, a true "symbiotic relationship" (Skuce and Meyer, 1991) must be established between terminology and knowledge acquisition. As they say : "At the heart of the relationship between terminology and knowledge engineering is the fact that practitioners of both disciplines function as intermediaries in a knowledge communication context involving experts on the one hand and a knowledge processing technology on the other" (Meyer *et al.*, 1992) p.957.

Whereas D. Skuce and I. Meyer are mainly interested in the contribution of knowledge acquisition to terminology (they have built a knowledge acquisition tool (CODE) for the construction of Terminological Knowledge Bases), we are mainly interested in the contribution of terminology to knowledge acquisition.

A part of Artificial Intelligence is concerned with the constitution of Knowledge Based System (KBS). A knowledge base is built upon the definition of a structured set of concepts, derived, for a large part, by the knowledge engineer from text analyses (transcriptions of discussions with an expert or technical documentation). As for terminology, the focus of knowledge acquisition is the conceptual organisation of knowledge fields from the study of terms representing concepts in texts.

Our aim in this paper, is not to propose results on the way of exploiting terminology in knowledge acquisition, but rather to try and define some methodological problems. From our competence and experiences in AI, terminology and linguistic engineering, we set up a cross-disciplinary study in order to evaluate the relevance of terminology within knowledge acquisition. We present here the results of this confrontation. First, we identified three possible kinds of contributions of terminology to knowledge acquisition (section 1) : the terminologies and terminological knowledge bases ; the methodologies applied by terminologists. Each of these hypotheses would require to be validated ; the tools designed for text analysis.

Then, we report on an experiment where we used a terminological tool, LEXTER, for knowledge acquisition with the MACAO methodology. This experiment contributes to the test of the third kind of contribution. In section 2, we present the tools and methodology we used, LEXTER and MACAO. In section 3, the SADE project is described. This experiment made us test our ideas on the contribution of terminology to knowledge acquisition. Then, we expose its major conclusions. On a methodological point of view, it enabled us to test a possible methodological integration. Other results stressed some limitations in MACAO. In order to overcome them, we propose new definitions in the knowledge modelling environment, concerning terms, conceptual structures and their relations with texts. As a conclusion (section 4), we expose our perspectives to validate our hypotheses and these modifications. We also discuss the concept of Knowledge Bank as a kind of application that could consistently gather terminological and conceptual analyses.

1. DISCUSSION ON THE CONTRIBUTION OF TERMINOLOGY TO KNOWLEDGE ACQUISITION

The contribution of terminology to knowledge acquisition may be organised along three distinct but not independent dimensions, taking into account the contribution of products of terminology (section 1.1), methodologies for terminological analysis, (section 1.2) and terminological software (section 1.3).

1.1 Terminological Knowledge Base and Knowledge Acquisition

1.1.1 From terminological databases to Terminological Knowledge Bases

The terminologists' work in a specific subject-field more often materialises by the building of terminological databases, gathering a list of terms, with which some linguistic information is associated, as well as a natural language definition. The main opening of these productions is today manual translation. As Meyer *et al* (1992) very well asserted it, terminological databases, as they are more usually conceived, are not suitable for an exploitation in Artificial Intelligence, especially for the building of KBS. This is mainly due to the poverty of the conceptual

description one can find in these databases, which is more often limited to natural language definitions of the concepts. The integration of an explicit part of conceptual modelling into terminological databases (TDB) produced by terminologists is then a necessary condition for a possible exploitation of these bases in knowledge acquisition. From this acknowledgement, the meeting between researchers from AI and terminology can be situated around the definition of the concept of Terminological Knowledge Base (TKB), proposed in (Meyer *et al,* 1992). Within the terminologists' community, more and more researchers are convinced of the necessity to integrate into the databases they build, not only linguistic data, but also the conceptual data on the knowledge conveyed by the terms as concepts designators.

1.1.2 The relevance of Terminological Knowledge Bases to Knowledge Acquisition

TKBs would accumulate the work of conceptual analysis of a domain performed by terminologists to an adequate format, so that they could be sources of information of great interest for knowledge engineers having to build a KBS on this domain. Theoretically speaking, the TKB concept can serve as a place for meeting between the disciplines of linguistics, terminology and artificial intelligence. It can be an object of a fruitful collaboration between terminologists and computer scientists, as it is in the TAI research group in France[1].

As far as knowledge engineering is concerned, one can see here an opportunity to provide some strong theoretical and practical references to current debates and proposals concerning ontologies. After favouring researches about problem-solving methods, an emerging key direction in that domain focus on the use of ontologies for the acquisition of the domain knowledge. Practically speaking, a TKB on a specific domain could serve as a source of information for building a KBS on the same domain or even a connected one. In a TKB, knowledge engineers will first find explicit conceptual information, which will assist them in developing the conceptual model of the system. But, they will also find the linguistic information, which the knowledge bases that are today developed in AI often lack. In the latter, the denominations of concepts by terms effectively used by experts in their discourses more often are lost or absent, which disrupts the communication between experts, knowledge engineers and end-users.

Only the names that knowledge engineers have given to the entities (variables) that will be manipulated by the system subsist. This problem is not a secondary one : the labels by which a system presents its concepts to a user have an influence on the interpretation the user will give on them. A wrong interpretation may have unhappy consequences on the accuracy of the data the user will enter in the system, and on the judgements of validity he will pass on its results. In a TKB, knowledge engineers will find some information, which will help them to adequately choose the names they will give to the concepts manipulated by the system, and set links between these names and the terms of the domain. Furthermore, the fact that these links exist will make easier the maintenance and the evolution of the KBS.

---

[1]D. Bourigault and A. Condamines lead a working group that is supported by the MRES and entitled “Terminology and Artificial Intelligence”. This group is composed of linguists, terminologists and computer scientists of various French research teams. The idea that are presented in this section owe a lot to the discussions carried on in this group. We thank : B. Bachimont, B. Biebow, J. Bouaud, J. Charlet, B. Habert, C. Jacquemin, G. Otman, F. Rousselot, J. Royauté, S. Szulman, Y. Toussaint, P. Zweigenbaum.

1.1.3. Important issues

One can expect that the knowledge engineers, having specific applications in mind, will find relevant information in a TKB. Nevertheless, they will also have to perform a complementary conceptual analysis in order to build the corresponding knowledge bases. This additional interpretation work results from the clear distinction to be made between a TKB and a knowledge base, and from the differences between domain and terminological analyses.

In a usual situation of knowledge acquisition, a domain model is built according to the type of problem-solving procedures to be implemented. The corpus of textual documents is a source of information, in which knowledge engineers draw pieces of knowledge in order to abstract domain concepts and relations, and, in a complementary way, to identify the domain items playing the roles expected by the problem solving method.

In a usual situation of terminological analysis, terminologists are not guided by a specific application. For them, the textual corpus is not merely a source of information, it must be considered as a reference : the terminologists must remain as faithful as possible to a reference corpus, reflecting the consensus knowledge of a group of specialists. Terminologist building a TKB will aim at developing a static description of the domain, regardless of any expert's problem-solving knowledge.

So, the rapid confrontation of these two approaches points out that their own objectives are different enough to prevent their integration from being straightforward. When terminologists try to remain neutral and adopt an application-independent point-of-view, knowledge engineers willingly select the domain knowledge in keeping with the requirements of the task model. Finally, the TKB provides mainly terminological information, with little connection to problem solving, whereas the knowledge engineer looks for problem-solving as much as domain conceptual data. In order to anticipate its use to design KBS, the TKB has to be as compatible as possible with the views corresponding to the various applications.

1.2 Methodological confrontation

1.2.1 Terminological analysis methodology

From the point of view of terminology, the linguistic analysis of texts allows terminologists to identify terms and semantic relations between them and, extrapolating from these results, to reach the conceptual system used by the experts of the domain. The two levels, linguistic and conceptual, are kept separate.

This work is mostly done intuitively and it succeeds because terminologists (coming very often from translation practice) have extensive experience in it. Unfortunately, in such an approach, computational tools are rarely used. From a theoretical point of view, the linguistic characterisation of terms is not sufficiently advanced. In the scope of a collaboration with disciplines such as knowledge acquisition, the terminologists are led to make their approach more systematic and better structured if they want to communicate it.

Nevertheless, the new needs in natural language processing and artificial intelligence entail new perspectives for both practice and theory ; research has now begun in two directions :

- the development or adaptation of tools to help terminologists in their task,

- theoretical studies concerning terms and semantic relations.

Four main tasks in which tools may be used can be identified within terminological processing :

- identifying terms (or selecting terms among candidate terms given by tools),
- identifying semantic relations (or selecting semantic relations among candidate semantic relations given by tools),
- identifying concepts denominated by several terms,
- identifying terms denominating several concepts.

Beside their role in identifying data in texts, the results of these tools could be exploited in a "request mode". For instance, they could help to check whether a certain concept or relation exists in a text or not ; however, such facilities are hardly available.

1.2.2 Knowledge acquisition methodology for text analysis

Together with human experts, texts are the major knowledge sources for knowledge acquisition and modelling. Most of the helps provided to facilitate their analysis consist of software such as tools for an automatic text interpretation or hypertext environments that help structure texts (Bourigault, 1994). These tools are discussed in section 1.3.

Even though words and texts are indicators of knowledge, texts are not explicitly analysed through linguistic criteria. Very few approaches promote linguistic or terminological guidance to extract knowledge from texts. For instance, very few knowledge acquisition methodologies underline a purposeful and explicit distinction between terms and concepts or are interested in the complete domain terminology. When used (like in the early versions of KADS), the word 'term' and 'terminology' refer to the structure labels and their list rather than to the terms used in the domain. As knowledge modelling is interested in a knowledge level description of a system behavior, the labels used to refer to structures will become the system variables, regardless of their precise meaning for people in the domain.

However, many approaches also consider the conceptual model as a means to discuss with domain experts. Then, the model is all the more readable as domain terms are used. So the structure labels have two possible interpretations, one in the scope of the system design and another to be given by the expert. In general, a compromise is found to define words that are acceptable in both contexts. But this frequently leads to confusion between terms and conceptual structures, and about the meaning of the concepts.

The links between conceptual structures and texts often correspond to hyperlinks (Khün *et al.,* 1991). A structure is linked to one or several paragraphs where it is defined, explained or illustrated by an example (Woodward, 1990). For instance, in knowledge acquisition tools such as K-station[2] for the KOD methodology (Vogel, 1988) or KADS-tool[2] , knowledge engineers can select parts of texts. These parts of texts are highlighted or coloured when they are attached a structure in the conceptual model (a concept, a task or an inference step in KADS-tool for example). This link has nothing to do with the 'occurrence' link used in terminology because the names of the structures may not appear in the text selection. In a similar way, knowledge engineers decide from their own interpretation of the text to link concepts together or to connect

[2] Product of Ilog, France

them to other structures. In that purpose, they may consider their meaning in the domain or their role in the problem solving process.

Finally, it could seem that knowledge acquisition has first poorly integrated results and methods from terminology or linguistics. If we look at more recent works, we can see new interests from KA researchers for these disciplines. On the one hand, some projects directly deal with reciprocal contributions of terminology and KA (Skuce & Meyer, 91) (Gomez, 94). On the other hand, a recent direction of investigation in knowledge acquisition, the definition of reusable domain ontologies, has brought a new insight towards terminologies. Because ontologies should facilitate the use of concept definitions when building any application in a domain, they should propose a consensual view on the domain. For this reason, a part of their role is similar to the one of TKB. But major differences exist, such as the way that they are obtained (terminologies gather terms identified in a corpus of texts whereas concepts in an ontology are the result of a domain analysis) or the nature of the entities they contain (words with linguistic information in TDB against concepts and their relations in a problem solving context in ontologies).

1.2.3 Synthesis : definitions and comparison

As a synthesis, we propose to recall definitions in terminology (T) and knowledge acquisition (KA), and to compare text analysis in both of the two domains along eight dimensions :

- Definitions :

term : For T, a term can be defined as a linguistic sign made up of one or several words which designates, in general without ambiguity, a concept within a particular domain. In KA, when used, terms are labels, often confused with the concepts they denote.

concept : For T, it is a mental representation of an entity. In KA, it is a knowledge representation structure built up to describe a domain entity

relations between concepts : In most TDB, the only relations mentioned are *is-a* and *part-of*. Domain models in KA also propose relations depending on the problem solving, such as expressions or causal relations.

context : In T, a context is a part of a text located on the right and left sides of a term. In KA, it refers to a situation or the status of the environment in which events may occur.

- Each discipline refers to a specific corpus : In KA, texts may be technical documents (historical reports, presentations the domain theory, guide-books, training supports, etc.) or the knowledge engineer's work documents (reports of interviews with the expert, descriptions of how he/she solved cases). Texts are considered as knowledge sources. In T, the corpus is mainly composed of the first kind of documents. Texts have the status of references.

- The objectives of the text analysis vary from one discipline to the other : In KA, text analysis is supposed to help to understand the domain and it contributes to the domain analysis. For T, a first objective is to identify domains terms among other words in the corpus, and a second one is to reach concepts via these terms.

- The methodologies applied are quite different : In KA, pieces of text are selected and linked to conceptual structures (abstraction stage). Texts also provide specific domain knowledge in order to complete the conceptual model (instanciation stage). In T, the analysis is carried out according

linguistic principles. For instance, the terminologist looks for deviancy, relies on linguistic properties of their use to identify terms (Condamines, 95).

- The expected output differ slightly : Text analysis for KA aims at identifying structures that will form the conceptual model, whereas terminologists look for terms, information concerning their usage and their related concepts.

- The criteria for selecting terms are not of the same kind : Knowledge engineers are free to define new terms which may not occur as such in documents whereas terminologists generally restrict candidate terms to those appearing in the corpus.

- Criteria for defining concepts : Knowledge engineers select paragraphs or parts of text according to their own understanding, and to the relevance they attribute them for the problem-solving description. For T, concepts are underlying the text. They must be brought to light thanks to the interpretation of the results of the linguistic analysis.

- Interviews of the expert are carried out with different purposes : in KA, the first interviews are transcribed and used as knowledge sources. Then, the expert validates and defines the identified concepts. In a similar way, the expert helps the terminologist to validate the candidate terms selected according linguistic criteria as well as their related concepts with their definitions.

- The importance of human interpretation and its role : In KA, the last two remarks point out the major role of human interpretation. In T, the human interpretation takes place after a linguistic analysis, to link linguistic facts with concepts or to study the usage of terms.

1.3 Terminological software for knowledge acquisition

In the field of terminology, text processing tools are mainly statistical tools, which give numerical information about words, and concordancers, such as SATO (Daoust et al., 1989), which give all the contexts containing occurrences of a term. Research progress in natural language processing make it available a new kind of tools : linguistic-based terminology extraction tool, such as (for French) TERMINO (David & Plante, 1991) and LEXTER, presented in the next section. These tools perform a morpho-syntactic analysis of a corpus and yield a list of candidate terms, with a good recall rate. This list has to be validated by a terminologist or an expert.

These tools are specifically dedicated to terminological processing, but they could be used by knowledge engineers. Most of knowledge acquisition methodologies recommend to begin with the building of the subject-field terminology. Nevertheless, these methodologies usually give very little information to guide the knowledge engineers in their work of spotting the terms and concepts of the expertise domain. This task is not a self-evident one. One can then judge in advance that a terminology extraction software will be of use to knowledge engineers. In the community of knowledge acquisition, Reimer developed a system (WIT) which automatically acquires terminological knowledge about a domain of discourse (Reimer, 1990).

We made a detailed bibliographical study on the theme of text analysis software tools for knowledge acquisition (Bourigault, 1994). A synthesis of this study is presented in (Bourigault, 1995). We have identified two main classes of tools : *transfer tools*, whose aims are the direct translation of a text into (parts of) a knowledge base, and *hypertext editors*, which do not perform any text analysis but which help the knowledge engineer to structure his

documentation. Between these two types of tools, we have proposed a third way : *scanning tools*. These tools perform a low-level mass processing in order to assist the knowledge engineer in the scanning of a large-sized technical documentation. Terminology extraction software well represent this family of tools. They could join the ranges of knowledge acquisition tools that are proposed by different knowledge acquisition methodologies.

1.4 Conclusion

In this section, we considered three kinds of contributions of terminology to knowledge acquisition : it may benefit from (1) TKBs, (2) methods used by terminologists, (3) tools used by terminologists. Even though parts of this belief were confirmed by our past experiences, we consider them as hypotheses to be validated. Three types of studies are required to evaluate how each contribution practically takes place. At the moment, we are mainly working on the third hypothesis. We are testing the use of a terminology extraction software (LEXTER) in knowledge acquisition. Knowledge acquisition for knowledge-based systems is a complex process, which involves a lot of skill and know-how. A tool likely to assist knowledge engineers in this process has to be tested in the framework of a global knowledge acquisition methodology, within which texts analysis is only one of many components. This is necessary to clearly specify the function of the tool and the way to use it within the overall acquisition process. That is the reason why we decided to test our terminology extraction software in association with the MACAO knowledge acquisition methodology. In section 3, we will report on an experiment of using LEXTER in a knowledge acquisition application, within the framework of the MACAO methodology. In the next section, we first present both LEXTER and MACAO, the tools taking part in the experiment.

2. LEXTER AND MACAO

2.1 LEXTER

LEXTER is a Terminology Extraction Software (Bourigault, 1992, 1994, 1995), (Gros & al., 1994). Starting from a corpus of texts on any technical subject, LEXTER extracts noun phrases which are likely to be terms. LEXTER has been developed in an industrial context, which is the Research and Development Division of *Electricité de France*, the French Electricity Board. The development of a noun phrases extractor was a very delicate task. We were subject to two antinomical constraints : robustness and accuracy. In order to satisfy these constraints, we were led to develop some original techniques of Natural Language Processing (Bourigault, 1994) : the system uses syntactic parsing techniques and endogenous corpus-based learning procedures. Two versions of LEXTER are available, one for French texts, the other for English texts. Both of them are organised in the same way, with four successive modules : (1) Tagging, (2) Splitting, (3) Parsing, (4) Structuring. One can find a brief description of these modules in (Bourigault, 1995). In this section, we only present the user interface.

In an entire automatic way, LEXTER analyses a whole corpus in a batch mode. It is only at the end of this process that the user consults and validates candidate terms. Each complex candidate

term is analysed into two constituents : a constituent in head-position (H-position), representing more often a super-ordinate concept, e.g. *analysis* in the term *syntactic analysis*, and a constituent in expansion-position (E-position), mentioning a specific attribute, e.g. *syntactic* in the term *syntactic analysis*. LEXTER links each analysed complex term to both of the terms which constitute its head and its expansion, as well as to the terms of which it is either the head or the expansion. Thus it yields a very dense network of candidate terms connected to one another by two types of oriented links : H-links and E-links. At the end of the automatic processing of the corpus, LEXTER organises the network of candidate terms, together with the corpus it has been extracted form, as a hypertext-like structure.

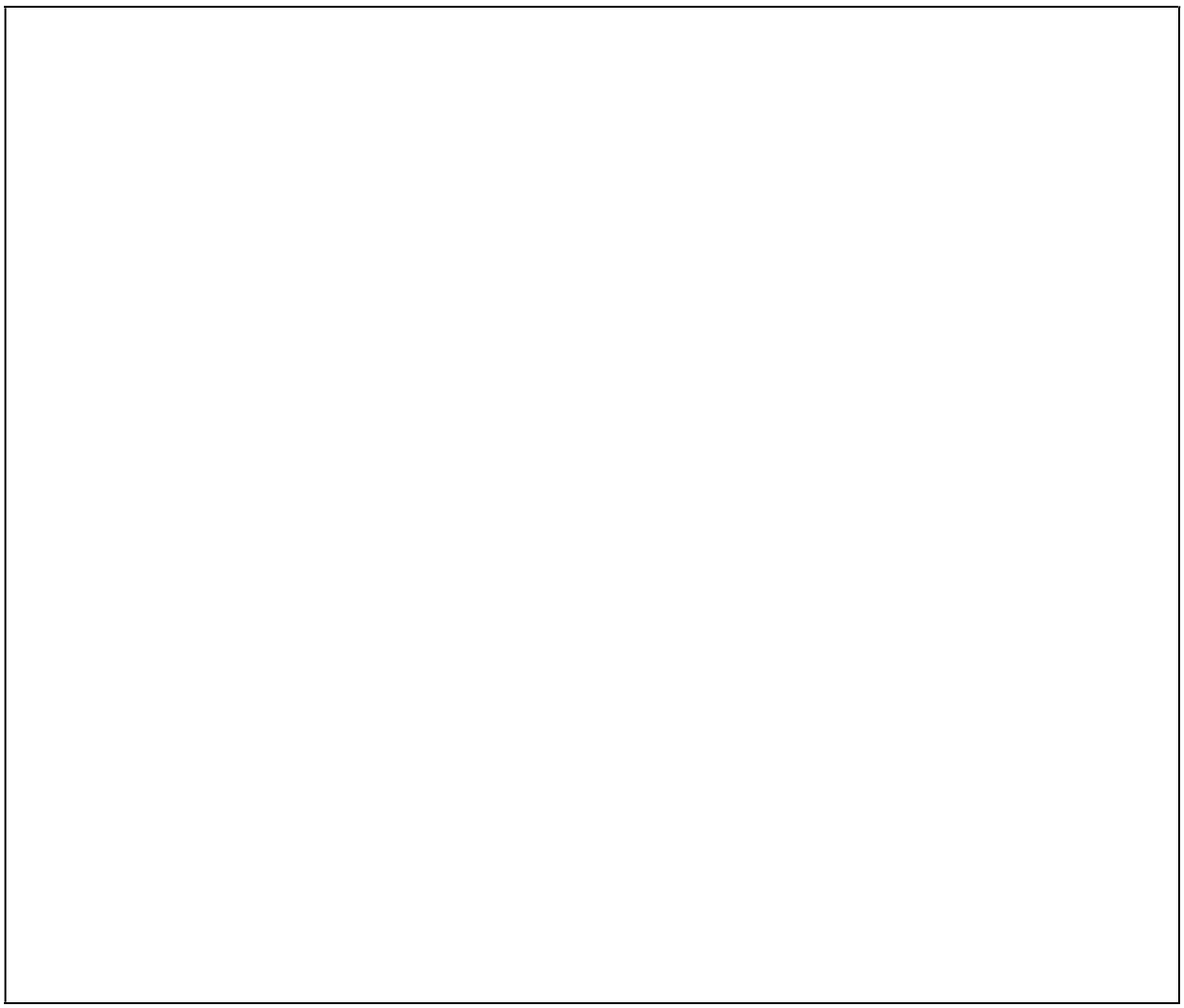

Figure 1. A simplified view of some term-nodes and text-nodes in the Terminological Hypertext Web
with some hypertext links

The interface, that allows to consult the results provided by LEXTER, is a hypertext web, known as terminological hypertext web. It is composed of nodes and links. The two main types of nodes are term-nodes and text-nodes.

Each term-node is associated with a candidate term extracted by LEXTER from the original corpus. There are as many term-nodes as candidate terms. To each term-node corresponds a display which provides a large set of information concerning the term, especially two

paradigmatic lists : the list of complex terms in which the term is in H-position and the list of complex terms in which it is in E-position.

Each text-node is associated with a textual unit of the original corpus. There are as many text-nodes as textual units. To each text-node corresponds a display in which both the textual unit and all the terms LEXTER has detected in it are displayed.

The terminological hypertext web is more fully described in (Bourigault, 1995). Figure 1 gives some simplified views of term-nodes and text-nodes displays, with some hypertext links.

2.2 MACAO

MACAO is a general purpose Knowledge Acquisition methodology (Aussenac *et al.*, 1988), which provides techniques and guidelines for expert knowledge elicitation, analysis and modelling. A software is available to build up the conceptual model with a specific knowledge representation language. The method and tool have been used for validation to design KBSs. One of the most recent and complete experiments is the SADE project mentioned in section 2 (Lépine & Bourigault, 1993).

2.2.1 Main steps of the method

Knowledge acquisition with MACAO is organised into four steps (Aussenac & Matta, 1994)

(1) expertise characterisation

(2) abstraction of a framework of the conceptual model

(3) refinement of this framework and description of the complete conceptual model

(4) model operationalisation and validation.

Recommendations and the method decomposition are gathered in a methodological hand-book. Some of the characteristics of the method are to pay special attention to the expert's work analysis, to promote the analysis and modelling of problem solving protocols and to propose to identify categories among all the possible problems solved by the expert. The way problems of a category are solved is also modelled. So, unlike many modelling approaches that directly consider the system specifications and the characterisation of its task, MACAO suggests to analyse and model partly how the expert proceeds to solve problems before carrying out a system oriented analysis. As a consequence, most of the time is spent on data abstraction in the first two steps, whereas the next two steps are model driven. Recent evolutions in MACAO tend to promote the combination of abstraction and the reuse of generic models for the design of the framework of the conceptual model (Aussenac, 1994).

2.2.2 Knowledge representation in MACAO

MONA is the knowledge representation language defined for MACAO. Each conceptual structure corresponds to an object defined with slots. It is made of three parts (Fig. 2) :

- a natural language description presents the whole structure and, when detailed knowledge is available, each of its slots ;

- a graphic description helps define and show the connexions associating this structure or its slots with other structures in the model ;

- an operational description leads to a formal representation ; we chose not to decribe it explicitely in the structure itself, and preferred to propose to connect the structure to another one written with an operationnal language, LISA (Delouis, 93).

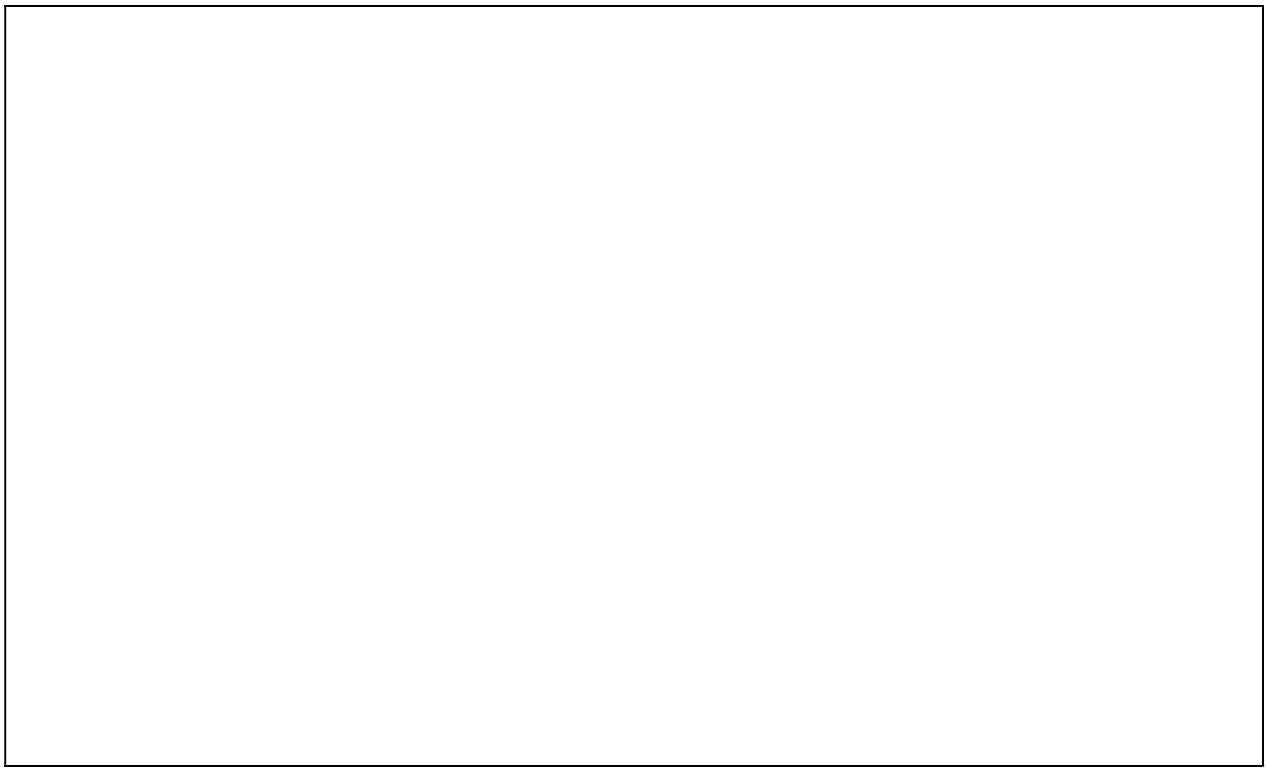

Figure 2 : Examples of editors of conceptual strucutres in MONA : the first editor is used to show a concept of the domain model, the second one for a task of the problem-solving model.

So the structures of the model can be first described with comments in natural language, and progressively be made formal before being connected to an operational LISA structure. As a consequence, different views exist on the model: an unstructured one in texts and terms, a conceptual one in the models and a design one in the operational model.

MONA offers structures to model the domain knowledge on the one hand, the problem solving knowledge on the other hand. *Concepts, expressions* and *links* are used to describe domain entities and their relations, which form the domain model. *Tasks*, *methods* and *roles* represent problem solving knowledge, and form the problem solving model. Modelling how a problem is solved consists in decomposing the main task into subtasks and in listing and defining the appliable methods for each of the tasks. So, the knowledge representation separates two inter-dependent networks. One can edit, create and modify parts of the each network through graphic editors. Moreover, views combining problem solving and domain knowledge (such as the concepts playing one role) are under development to make the model description easier.

2.2.3 Terms, lexicon and texts in MACAO

Besides the conceptual model, MACAO manages texts in *files* and *terms* to facilitate the understanding of domain knowledge and the dialogue with the expert and end-users.

During the abstraction tasks of the method, texts play an important role as knowledge sources. For instance, they often contain general presentations that help the knowledge engineer understand better the domain he discovers. They also provide definitions and knowledge for the domain characterisation. All along the acquisition process, the knowledge engineer is asked to write down his interviews of the expert, to transcribe the expert's protocols, to report progress meeting notes, modelling choices and design analyses. All these text *files* are gathered in the *dossier* in the software. They form a rich and useful data set to be used several times during knowledge acquisition. Unfortunaltely, not all the analysed texts are in a computer readable form : many remain available only on paper.

In order to keep track of the domain understanding, MACAO proposes to "identify the vocabulary of the expertise" or *lexicon*: the *lexicon* contains all the words that, according to the knowledge engineer, characterise the domain or help understand it. This list is refined and improved with the help of the expert. Each word in the list is called a *term* and the list of terms forms the lexicon. A definition, to be validated by the expert, a type, a list of possible values and references (names of *files* where knowledge concerning this term can be found), are associated to each *term*. During the knowledge modelling, the *lexicon* is supposed to be used as a provider of names to label the conceptual structures. The name of every new structure in the conceptual model must be selected in the *lexicon* or, in a reverse way, if a new name is used, this creates a new *term* in the *lexicon*. The denomination of the structures in the conceptual model with domain specific terms is promoted in order to facilitate the model understanding and validation by the expert.

3. INTEGRATION OF LEXTER AND MACAO : AN EXPERIMENT

3.1 Using LEXTER in a knowledge acquisition experiment

In this section, we report on an experiment where LEXTER has been used in a real application of knowledge acquisition, the SADE project, within the framework of the MACAO methodology (Bourigault & Lépine, 1994) (Lépine & Aussenac-Gilles, 1994).

The SADE project aimed at designing a system to assist the management and recovery of loans contracted for real estate acquisition by employees of the EDF-GDF companies. The KBS capitalizes the skill of those who deal with repudiated contracts, to help them to better analyze loan redemption files and to perform more adequate procedures for debt recovery. Evaluating a file not only requires the application of written law rules such as the loan contract clauses and the company regulation. It also relies on the expert's experience. Moreover, the final decision often is influenced by social, political and economical considerations about debt recovery.

The knowledge engineer first gathered textual knowledge sources (technical documents and transcripts of expert interviews). The use of a terminological scanning tool like LEXTER is justified only if the documentary corpus is large enough to make it difficult an exhaustive analysis of it directly by the knowledge engineer. Once the knowledge engineer has built an electronic documentary corpus, he split it into textual units, giving each of them an identifier. These textual units are the textual contexts of the candidate terms displayed in the terminological hypertext web. The corpus was around 70 pages (30 000 words). This was not a very big corpus. In documentation applications, LEXTER usually treats corpus of several hundred thousand

pages. The knowledge engineer had then the corpus processed by LEXTER, which yielded a terminological hypertext web.

The knowledge engineer used the terminological hypertext web in the framework of the MACAO methodology. As we already explained it (section 2.2), this methodology distinguishes (at least) two main phases within the acquisition process : (1) a bottom-up phase (data-driven knowledge acquisition), which leads to the design of a conceptual model, (2) a top-down phase (model-driven knowledge acquisition), which aims at the instanciation of the model. The terminological hypertext web was used during the first phase in a scanning mode, to build the vocabulary of the domain or lexicon. It was used during the second phase in a searching mode, as an efficient access to the documentation.

(a) *Scanning mode*. During the bottom-up knowledge acquisition phase, the knowledge engineer's aim was to discover and master the expertise domain. In this perspective, the terminological hypertext web was used as a terminological scanning tool in order to built the vocabulary of the domain. The knowledge engineer skimmed in a systematic and exhaustive way through the network of candidate terms. For each candidate term, he had an access to its different contexts : *textual contexts*, the textual units in which it has been detected, and *terminological contexts*, paradigmatic lists of candidate terms having the same term in common (see section 2.1). While progressing towards a better mastery of the domain, the knowledge engineer has performed three main tasks : (1) eliminating non-relevant candidate terms, (2) finding synonymies between terms, (3) picking out some generic terms that might well represent central concepts within the model which would later be built. To perform this last task, the knowledge engineer relied on coefficients of productivity and of relevance given by the system. He also added some terms which have not been detected by LEXTER. At last, he took the initiative for the building of hypertext links between nodes of his choice. During this phase, the knowledge engineer has reduced the list of candidate terms from 2 800 (900 candidate terms occurring at least two times) to 400 terms.

The result of this scanning stage was the terminological hypertext web validated and enriched by the knowledge engineer. It was similar to what Motta, Rajan & Eisenstadt (1990) refer to as digested data, that is a weakly structured glossary of relevant terms, together with their textual contexts. There remained to connect these terms with a conceptual model of expertise.

(b) *Searching mode*. During the top-down acquisition phase, the knowledge engineer's aim was to instanciate the model of expertise he built. In this perspective, the terminological hypertext web was used as a searching tool. The knowledge engineer chose the terms that he knows to be relevant at that point of the knowledge acquisition process. He made an investigation around the terms, by examining their terminological and textual concepts. He performed a contextual analysis of these terms in order to precisely define the corresponding concepts. In this phase, he used the terminological hypertext web mainly as an efficient and non-linear access to the documentation.

3.2 Problems raised by the integration

The SADE project put to light strengths but also some limitations of MACAO, mainly during the early steps. The knowledge engineer is given very few indications about how to identify the vocabulary of the expertise : Which words must he select or reject ? How to fix the boundaries of

the domain vocabulary ? Selecting and defining terms without any precise modelling goal in mind results time consuming. Moreover, after the lexicon is built up, even though it is automatically updated, it is hardly used when organising knowledge in the conceptual model. More generally, terms and structures in the conceptual model are not efficiently managed.

As a first solution, the knowledge engineer used LEXTER to help him list the most relevant terms in the domain from a corpus of texts (technical documents and interview transcriptions). Although promising, this experiment could not enable the test of all benefits or difficulties of integrating LEXTER with MACAO because neither the concepts nor the tools were compatible enough. Some of these differences would be observable with other knowledge acquisition approaches (see section 4):

- the words *term* and *text* obviously have different meanings ;
- links between texts and terms are of a different nature ;
- the criteria for selecting a term are slightly different.

Other incoherences and inadequacies with basic linguistic principles are more specific to MACAO. We would have met these problems with any terminological software :

- a misleading confusion exists between *terms* and the associated conceptual structure: as long as *terms* have a definition, they denote much more than linguistic information and contain conceptual knowledge. They overlap the conceptual structures ; so, knowledge engineers either have to repeat twice some definitions or they ignore *terms* because they give priority to structures in the model.
- the lexicon contains two kinds of words with different status : words found in texts and selected for their importance in the domain understanding ; labels of conceptual structures, defined by the knowledge engineer for modelling purposes.

Finally, the new attention paid to words and their use modifies the point of view one may have during the knowledge acquisition. We improved MACAO to take into account these conclusions.

3.3 MACAO-II

MACAO-II is an evolution of MACAO where the knowledge representation (MONA), the software and the methodology have been modified. The motivations of this re-design include an integration of results from the state of the art, the willingness to overcome previous limitations and also the evolutions required by the integration of the use of a terminological tool. This adaptation led to define more rigorously notions and structures related to terms and texts in the conceptual model. From now on, we do not refer any longer to the definitions given in section 2.2.3.

3.3.1 Terms and labels

A new knowledge representation structure was defined to represent terms with MONA. Knowledge representation in TKB influenced this choice. A term contains linguistic information provided by LEXTER and connections towards the conceptual model or other terms added by the knowledge engineer (Fig. 3).

```
Term :   name                          ; the signifier
occurs-in Text000          ; connections towards texts or paragraphs in which it is used
occurs-in ...              ; and, for each text, the number of times it appears in it
productivity               ; number of times this term appears in the corpus
has-head                   ; term in head position
has-expansion              ; term in expansion position
is-head-of                 ; terms in which the current term appears as the head
is-expansion-of            ; terms in which the current term appears as the expansion
usage-1          ; conceptual structure denoted by this term according to 'usage-...'
usage-...
synonymous terms           ; terms denoting the same structure in different contexts
```

Figure 3 : The new term structure defined in MONA in order to integrate better LEXTER and MACAO. Most of the attributes of a term are automatically valued by LEXTER. The remaining one (in italics) must be valued by the knowledge engineer.

Beside terms, the definition of all the other knowledge representation structures has been modified. The example of the new concept structure is shown on figure 4. Each structure has a label or *name,* given by the knowledge engineer with the expert's agreement. This label may correspond to a term or not. If it does, the structure also is connected to this term (and its synonymous terms if any) through a *usage* link. So the difference between terms and labels is now more explicit. A label not corresponding to a term will not give birth to a new term. On the one hand, labels are gathered in several lists, one for each type of structure : the libraries that enable to edit the structures. On the other hand, the list of terms forms the expertise *vocabulary.*

```
Concept :      name                          ; its label
        description                ; definition and description of concept and attributes in natural language
        possible values            ; unformal indication of possible or allowed values
        attributes                 ; list of names of attributes, and, for each of them, possible values
        type                       ; concept type according to the problem solving : case concept, variable
                          ; or domain knowledge
        links                      ; links with other concepts : for each link, link label and concept name
                is-a      concept-2        ; example
                is-by     concept-3        ; example
        references                 ; labelled links towards sections of texts
                is-defined-in     section1          ; example
                s-explained-in    section44         ; example
        usage1...          term1   ; term denoting this concept according to 'usage-...'
        usage2.. term2     ; term1 and term2 are synonymous
```

Figure 4: The new concept structure defined in MONA. In order to integrate better LEXTER and MACAO, new attributes (in italics) were defined.

Figure 5 shows the possible connections between terms and other conceptual structures in the model : a conceptual structure can be linked to several terms, corresponding to the various words that can be used to designate it in various situations or *usages*. An ambiguous term can denote several structures according to different *usages*. However, a conceptual structure has a unique name (or label) which may correspond to a term or not.

The relation between terms reflect linguistic properties (such as 'a term is head of another term'), whereas the relations between structures represent conceptual properties (such as 'a concept is-a sub-class of another'). The first one are automatically set by LEXTER, whereas the second one are manually set by the knowledge engineer. However, the existence of a relation between terms may reveal a possible relation between the corresponding structures.

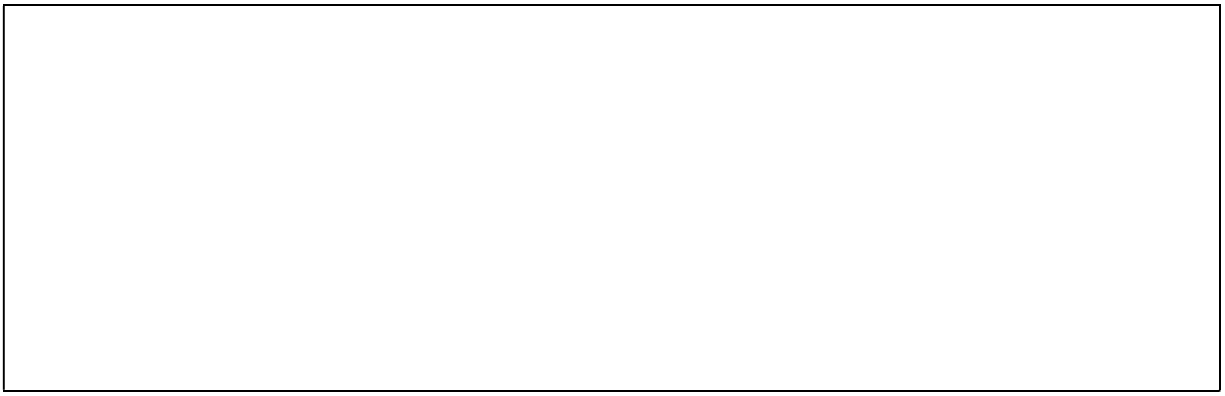

Figure 5 : Connexions between terms and conceptual structures

3.3.2 Links between terms and texts versus labels and texts

The possible links between texts and terms on the one hand, between conceptual structures and terms on the other hand, have been differentiated. These links are represented with labelled hyper-links that reflect the meaning of the connection, such as shown on figure 6. Just as the links between structures, the first one are automatically set by LEXTER, whereas the second one are manually set by the knowledge engineer. In a similar way, the existence of a relation between term and a text may reveal a possible relation between the corresponding structures and this text.

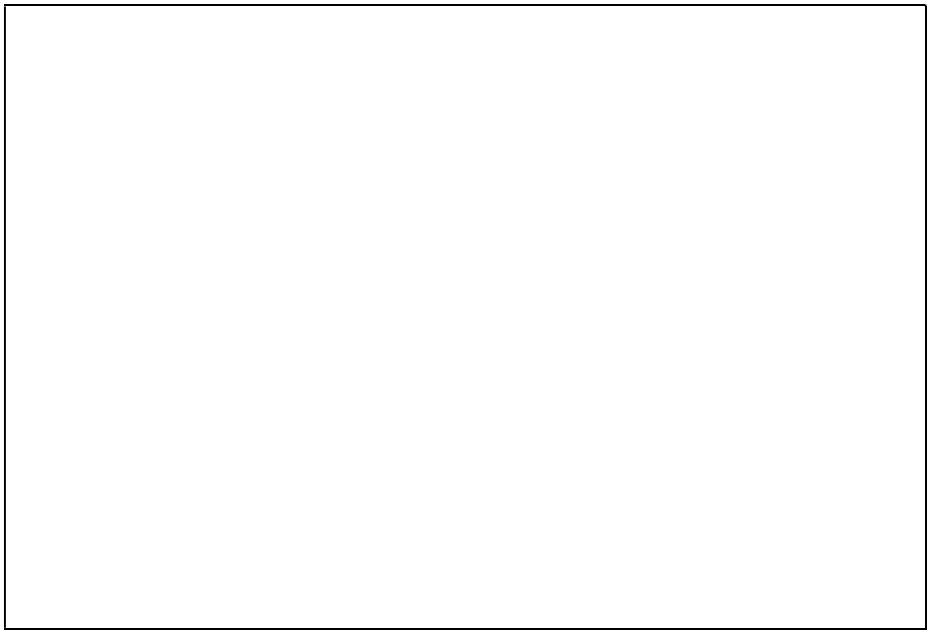

Figure 6 : The connection between texts and terms versus the connection between conceptual structures and terms. A term must appear in a text, whereas the names of structures can be created by the knowledge engineer. Structures are associated with texts where they are defined or explained.

In order to experiment this integration, terminological hypertext webs are now available in the same platform as MACAO. We developed with Aïda[3] a hypertextual interface for the results produced by LEXTER, similar to the existing one on Macintosh. We also modified the term structure in MONA as well as the connections between terms, texts and conceptual structures.

[3] Product of Ilog, France

The management of direct links from raw data (in texts) to structures (in the conceptual and then in the operational models) is under development.

Now, the knowledge engineers can select the relevant terms for the expertise among all the candidate terms proposed by LEXTER. Then, they can use this list to identify conceptual structures such as domain concepts. Or, in a reverse way, they can pick up names in this list to label the structures they want to define.

4. CONCLUSION

4.1 Other validation experiments

At the moment, two problem-solving knowledge are being modelled with MACAO : one in finance (economical diagnosis of a company[4]) and one in agriculture (assessment of strategies for breeding sheep flocks all in the south of France[5]). In both of these cases, knowledge elicitation has started without any terminological analysis. These two projects aim at obtaining an explicit model of the expert's know-how rather than at designing a system. These situations are adequate for our experiment because they stress the need for a detailed analysis of expert's knowledge through documents concerning their work. We will report here how we plan to carry out the second experiment, the first one being very similar.

Two experts, specialists in different sub-domains (the grass versus animals), are involved in this project. One of the experts (the grass specialist) specialises in botanics and the growth of mediterranean vegetals. The other expertise (animal specialist) covers zoology and sheep farming. The experts produced documents reporting on the way they analyse the breeding strategy of a sheep-breeder. But most of their knowledge is not explicit yet. It is elicited during interviews and transcribed. The various texts totalize about 100 pages : technical documents, scientific papers, interview transcriptions, notes and reports of working meetings.

Terminological differences appear clearly between the two specialists. The use of LEXTER could help confirm these differences and precise the definition of terms according to each of them. Using LEXTER should also help index all the available documents and characterise their contents (*scanning mode*). Another expected gain is to ease the access to documents and the search for specific information in them (*search mode*).

We plan to take into account the nature of the documents and the point of view of their authors (grass or animal specialists) in the interpretation of the results. In fact, LEXTER proves efficient when analysing a technical corpus. But we have to evaluate its interest when the corpus is made mainly of meeting reports or transcripts of interviews.

As the results obtained with LEXTER will directly be available from MACAO, the use of the terminology in the *scanning mode* will be facilitated. Direct links between terms and conceptual

---

[4] This project is carried out by F. Tort, from LRI - Univ. d'Orsay, since 1993 She has started using MACAO from the beginning of the project.

[5] This project is being carried out by N. Girard, from INRA-Avignon, Unité d'écodéveloppement, since 1992. She has started using MACAO in 1993.

structures will enable the consultation of terms while building the model. All these facilities should enhance the use of the terminological results.

In parallel to the current investigations, a collaboration between N. Aussenac-Gilles and A. Condamines has started to specify where, in the knowledge acquisition process, a terminological analysis could prove helpful and how to perform it. Together with the on-going experiments, it should provide more precise guidelines on how to combine the use of terminological tools and methods with knowledge modelling.

4.2 Towards Knowledge Banks ?

The perspectives of our work are twofold :

- we have to experiment the use of an existing TKB for the modelling of an expertise. Such a TKB is available for applications in Space from A. Condamines (Condamines, 1993) It was developed at ARAMIIHS[6] as one of the assistants to be integrated in an authoring support tool-box. We are currently looking for a knowledge-based application in Space.

- we plan to evaluate the costs and gains of this terminological analysis. We must determine more precisely in which conditions such an analysis is worth being carried on. We feel it will be costly. So it is important that the data gathered during a terminological and then a conceptual analysis could be reusable for other purposes.

As a consequence, we consider that a natural application of our work could be in the gathering of corporate knowledge, including technical documents on the one hand, and human skill and know-how on the other hand. We prefer to call such a network of informal but structured knowledge a *Knowledge Bank* (KBk). It would contain terminological and conceptual knowledge elicited not only in texts (like for the TKB) but also from domain practitioners. It would also contain problem-solving knowledge.

[6]Action de Recherches et Applications Matra-IRIT en Interaction Homme Système, joint laboratory involving engineers from MMS-France and researchers from the IRIT laboratory

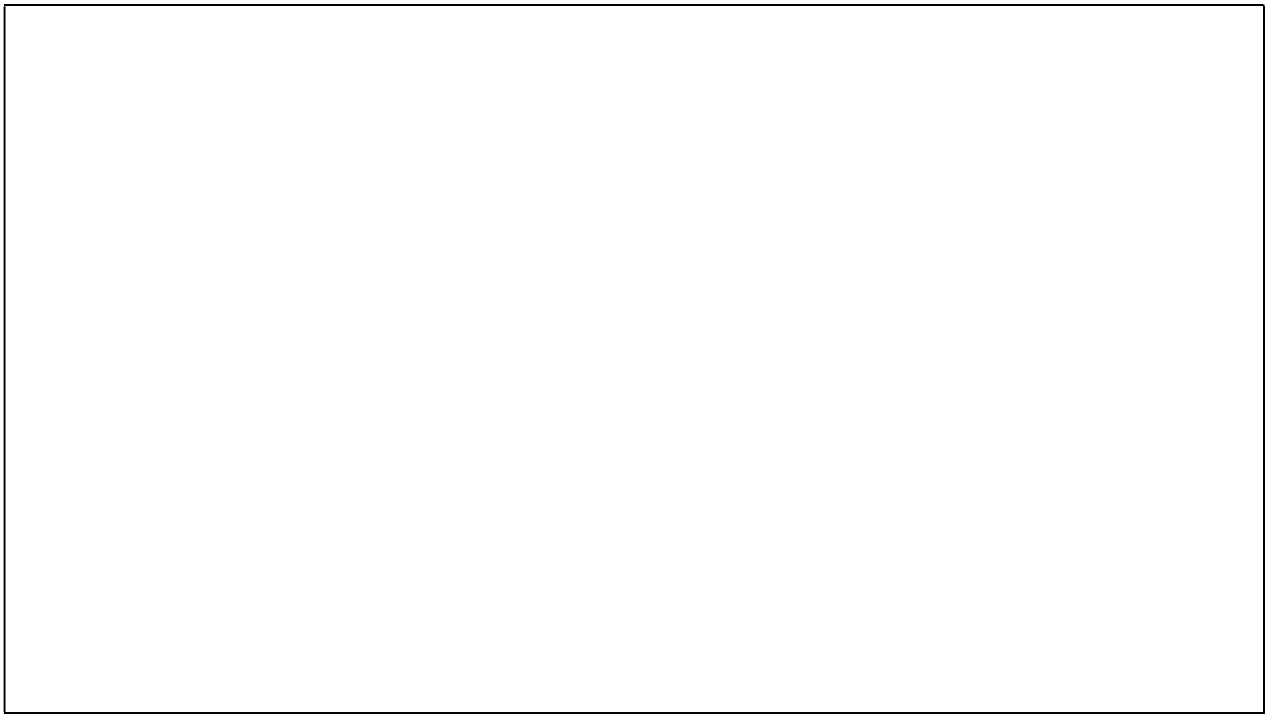

Figure 7 : The various ways to exploit a domain knowledge and terminology : Two knowledge sources, the expert and texts, may provide knowledge that feeds intermediary models to be exploited for various purposes according to the users' needs, the degree of formalisation and the knowledge required. Such models (ellipses on the picture) are TKB, Expert's Conceptual Model (Expert's CM), System Conceptual Models (System CM) and Knowledge Bank (KBk). These models can be used to develop (boxes on the picture) intelligent document browsers, terminology consultation systems, decision support or problem solving systems, etc.

Figure 7 shows how the Knowledge Bank could be built from a TKB and conceptual models of experts or users. In this perspective, the Knowledge Bank should allow the design of a wide range of systems for its browsing, for problem-solving assistance or training. Some of the numerous problems to be solved before are the definition of a knowledge representation that could facilitate reuse, and methodological guidelines that would help specify the content of these bases. Such problems are very close the one dealt with for the definition of ontologies.

ACKNOWLEDGEMENTS

The idea of writing this paper comes in part from the work carried out by Pascal LÉPINE for the SADE project. The integration of MACAO and LEXTER benefited a lot from the implementation performed by Patrick SÉGUÉLA. We would like to thank also Marie-Paule PÉRY WOODLEY for her helpful comments on drafts and the reviewers for their precise reading of the paper.

REFERENCES

AUGER P., DROUIN P., L'HOMME M.C., (1991) Automatisation des procédures de travail en terminographie, *Meta: Journal des traducteurs*, volume 36 (1).

AUSSENAC N., SOUBIE J-L., FRONTIN J., (1988) A knowledge Acquisition Tool for Expertise Transfer, In *Proc. of EKAW'88*, GMD Studien Nr 143, pp 8.1-8.12.

AUSSENAC N., (1994) How to combine data abstraction and model refinement: a methodological contribution in MACAO. *A future for Knowledge Acquisition,* Proc. of EKAW'94, 8th

European Knowledge Acquisition Workshop. Berlin: Springer Verlag. Series Lecture Notes in AI, N°867. pp 262-282.

AUSSENAC N., MATTA N., (1994) Making a method of problem solving explicit with MACAO. *International Journal of Human Computer Studies.* **40**, 193-219.

BOURIGAULT, D. (1992). Surface Grammatical Analysis for the Extraction of Terminological Noun Phrases. In *Proc. 14th International Conference on Computational Linguistics*. Nantes. pp. 977-981.

BOURIGAULT, D. (1994). *LEXTER, a Terminology Extraction Software. Application to Knowledge Acquisition from Texts*. Ph.D. Thesis, Ecole des Hautes Etudes en Sciences Sociales. Paris.

BOURIGAULT, D. (1995). LEXTER, a Terminology Extraction Software for Knowledge Acquisition from TEXTS. (Submitted to) *9th Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff.

BOURIGAULT, D. & LEPINE, P. (1994). Une méthode d'utilisation de LEXTER en acquisition des connaissances. In *Proc. 5ème Journées d'Acquisition des Connaissances*, Strasbourg. pp. F1-F13.

CONDAMINES A., AMSILI P., (1993) Terminology between language and knowledge : an example of a terminological knowledge base. In *Proc. of TKE'93 (Terminology and Knowledge Engineering)*. Francfurt: Springer Verlag.

CONDAMINES A., (1995). Terminology : new needs, new perspectives. To appear in *Terminology*.

DAOUST F., DUPUY L., PAQUIN L.C., (1989) ACTE: Workbench for Knowledge Engineering and textual data Analysis in the social sciences, *Proc. 4th international Conference on Symbolic and Logic Computing*, Dakota State University.

DAVID, S. & PLANTE, P. (1991). Le progiciel TERMINO: de la nécessité d'une anlyse morpho-syntaxique pour le dépouillement terminologique de textes. In *Proc. Colloque sur les industries de la langue*, Québec: Office de la langue française. pp. 71-88.

DELOUIS I., Krivine J.P., (1993) Opérationalisation du modèle conceptuel: vers une architecture permettant une meilleure coopération système/utilisateur, In *Actes des 12èmes journées internationales "Systèmes experts et leurs applications"*, Avignon (F).

GOMEZ F., SEGAMI C., Knowledge acquisition from natural language for expert systems based on classification problem-solving methods. *Knowledge Acquisition*. **2**. 1990.

GROS, C., BOURIGAULT, D. & VULDY J.-L. (1994). Linguistic-Based Toolbox for Hypertext Automatic Linking on Large Technical Documentation. In *Proc. 3rd International Conference on Information and Knowledge Management*, Gaithersburg.

JOUIS C., (1994) Contextual Approach: SEEK, a linguistic and computational tool for use in Knowledge Acquisition, *Proc. of First European conference Cognitive Science in Industry*, 28th - 30th Sept. 1994.

KHÜN O., LINSTER M., SCHMIDT G., (1991) Clamping, COKAM, KADS and OMOS. In *Porceedings of EKAW91.* Smeed, Linster, Boose and Gaines Editors. Univ. od Strathclyde (Scotland).

LEPINE, P. & AUSSENAC-GILLES, N. (1994). Modélisation de la résolution de problèmes : comparaison expérimentale de KADS et MACAO. In *Proc. 5ème Journées d'Acquisition des Connaissances*, Strasbourg. pp. H-1/H-14.

LERAT P. (1988) Terminologie et sémantique descriptive ; in: *La banque des mots*, numéro spécial 1988.

LINSTER M., (1992) *Knowledge acquisition based on explicit methods of problem solving*, Ph.D. dissertation, Univ. Kaiserslautern. Feb. 1992.

MEYER, I., SKUCE, D., BOWKER, L. & ECK, K. (1992). Toward a new generation of terminological resources: an experiment in building a terminological knowledge base. In *Proc. 14th International Conference on Computational Linguistics*. Nantes. pp. 956-960.

MOTTA, E., RAJAN, T. & EISENSTADT, M. (1990). Knowledge acquisition as a process of model refinement. *Knowledge Acquisition*, 2. pp. 21-49.

REIMER, U. (1989). Automatic Knowledge Acquisition from Texts : Learning Terminological Knowledge via Text Understanding and Inductive Generalization. In *Proc. 5th Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff. pp. 27/1-27/16.

SKUCE, D. & MEYER, I. (1991). Terminology and knowledge acquisition: exploring a symbiotic relationship. In *Proc. 6th Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff. pp. 29/1-29/21.

VOGEL C., (1988) *Génie cognitif*, Paris : Masson.

WOODWARD B., (1990), Knowledge acquisition at the front end: defining the domain, *Knowledge Acquisition*, **2**, N° 1 pp 73-94