A UMLS-based knowledge acquisition tool for rule-based clinical decision support system development.

Soumeya Achour, Michel Dojat, Claire Rieux, Philippe Bierling, Eric Lepage

A UMLS-Based Knowledge Acquisition Tool for Rule-Based Clinical Decision Support Systems Construction

Soumeya L. ACHOUR[1], MS, DVM, Michel DOJAT[2], Eng, PhD, Claire RIEUX[1], MD, Philippe BIERLING[1], MD, PhD, Eric LEPAGE[1], MD, PhD

[1] Département d'Information Hospitalier, C.H.U. Henri Mondor, AP-HP, Créteil, France.
[2] Institut National de la Santé et de la Recherche Médicale INSERM U438, Grenoble, France.

To appear in Journal of American Medical Informatics Association
Note: This is a preprint and may differ in minor details from the published version

Mailing address: Soumeya Achour

Département d'Information Hospitalier
C.H.U. Henri Mondor
51, avenue du Maréchal de Lattre de Tassigny
F-94019 Créteil, France
Phone: 33 1 49 81 23 82 Fax: 33 1 49 81 40 50
e-mail: soumeya.achour@hmn.ap-hop-paris.fr

Decision support systems in the medical field have to be easily modified by medical experts themselves. We have designed a knowledge acquisition tool to facilitate the creation and the maintenance of a knowledge base by the domain expert and its sharing and reuse by other institutions. The Unified Medical Language System (UMLS) constitutes the domain entities and relations repository from which the expert builds, through a specific browser, the explicit domain ontology. The expert is then guided in creating the knowledge base according to the pre-established domain ontology and condition-action rule templates that are well adapted to several clinical decision making processes. Corresponding Medical Logic Modules (MLMs) are eventually generated. The application of this knowledge acquisition tool to the construction of a decision support system in blood transfusion demonstrates the value of such a pragmatic methodology for the design of rule-based clinical systems that rely on highly progressive knowledge embedded in hospital information systems.

Key Words: Knowledge Engineering, Domain Ontology, Arden Syntax, and Blood Transfusion.

I. Introduction

Clinical decision support systems (CDSSs) have been shown to be very helpful to medical practitioners. Knowledge acquisition (KA) and modeling play leading roles in the development of such knowledge based systems. Unfortunately, two key factors limit the development of CDSSs and their integration in Hospital Information Systems (HIS): 1) the time and effort required by medical experts to create and maintain knowledge bases; and 2) the extreme difficulty of sharing and reusing the validated knowledge bases because of their idiosyncrasies and lack of clarity for users outside the originating institution. The latter factor is considered to be the main limitation to daily use of CDSSs[1]. The goal of our work is the design of a KA tool to facilitate the creation and maintenance of a knowledge base by the medical expert and its subsequent sharing and reuse by other medical institutions. There are several approaches to achieving reuse, notably

- through generic abstractions: Lexicons, ontologies and problem-solving methods are built up and then adapted to fit the specific needs of a typical medical domain application[2-5].
- through the standardization of the knowledge representation: several standards have been proposed, such as KIF[6], Ontolingua[7] and Arden Syntax[8].
- through specific models, such as GLIF[9] or PRO*forma*[10], to represent certain kinds of clinical guidelines and support their dissemination.

Our project relies on the first two approaches.

Some KA environments have been proposed taking into consideration the specifics of the medical reasoning process and medical expertise. The research project GAMES[11], views the KA process as the construction of two models: the epistemological model, entailing the knowledge required to perform a particular task, and the computational model, containing data structures and algorithms designed to allow the computerized execution of that task. The PROTÉGÉ[12] environment allows developers to configure available problem-solving methods that are mapped to domain ontologies and generates task-specific knowledge acquisition tools. In parallel to architectures for KA, there is considerable work under way to create medical ontologies, a key component for building and reusing knowledge-based systems. For instance, the representation and integration language GRAIL, from the GALEN project, is especially designed to support models of medical terminology and achieve reuse of taxonomies[4, 13]. We consider that the Unified Medical Language Sources (UMLS) constitutes a knowledge corpus useful for building knowledge bases. Our approach, less general than that of PROTÉGÉ, is designed for medical experts without specific training in medical informatics. It allows them to configure rule-based templates that are mapped to the UMLS-based domain specific lexicon they have formerly constructed. Medical logic modules (MLMs) written in Arden Syntax are eventually generated in order to facilitate reuse and sharing of this knowledge base.

MLMs have been used to generate clinical alerts, interpretations or diagnoses[14-18]. For several years there have been numerous attempts to facilitate the creation of MLMs with respect to the Arden syntax. Interesting clinical experiences with MLM editors have been reported[19-21]. Our project differs from previous work in that it integrates the generation of MLMs into the general process of KA (ontology and knowledge base). In this article we report on our KA methodology; on the use of this methodology to build a CDSS applied to blood transfusion; and on a preliminary test of our system. In blood transfusion, the evolution of both medical knowledge and government regulations imposes a continuous adaptation of the CDSS that wholly justifies the use of our approach. Several CDSS have been proposed in this field[22-23]. Similar to all classical rule-based systems, these systems were difficult to maintain and extend. No specific knowledge acquisition technique was used for their development. One application embedded in the HELP system, concerns a computerized monitor blood ordering system based on a critiquing approach[24-26]. A set of knowledge tools is proposed to the expert for the definition and the maintenance of the knowledge base and a local data dictionary can be used. All these CDSS in blood transfusion turn essentially around the question of whether to transfuse or not. In addition to this issue, we studied, the choice of the qualifier (e.g. phenotyped or irradiated) of the product to be transfused. Our CDSS is currently being tested at the Henri Mondor Hospital (Créteil, France).

II. Methodology

Our methodology for knowledge acquisition and representation relies on two steps: first a domain ontology is developed: the expert selects the entities and relations present in UMLS that are relevant to constructing the domain ontology; and if needed, s/he introduces new terms and relations. Secondly, the domain knowledge is constructed: according to the pre-established domain ontology, the expert is guided in creating the knowledge base via condition-action rule templates. Corresponding MLMs are eventually generated automatically. Figure 1 represents our methodology.

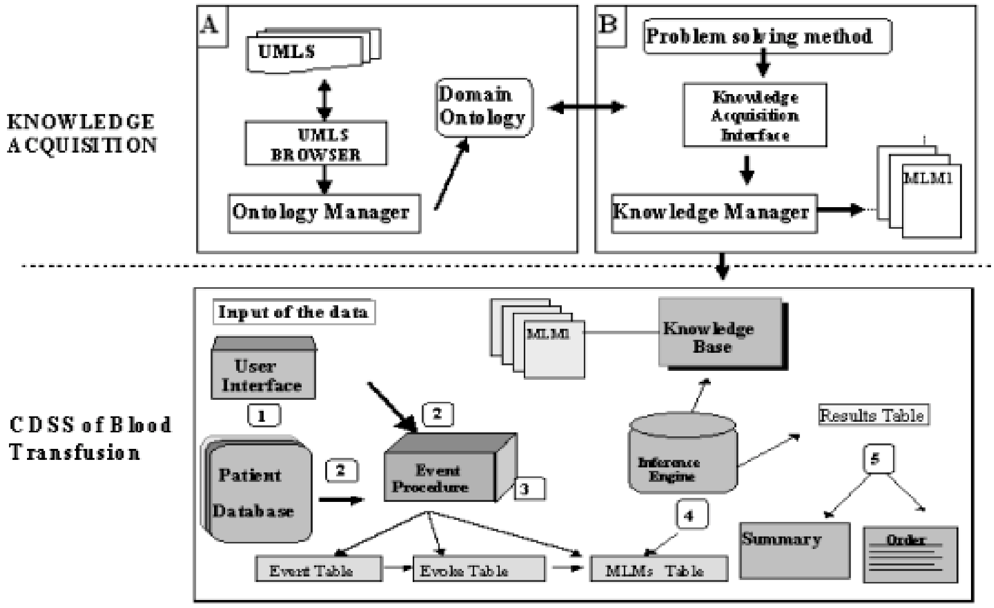

Figure 1. Knowledge acquisition process and decision support system design.
Block A: The domain ontology is built via the UMLS browser.
Block B: The knowledge acquisition interface allows the instantiation of a general problem solving method using the domain ontology. MLMs are automatically created and organized thanks to the knowledge manager.
Block C: An example of a decision support system architecture whose knowledge base is generated using our methodology. The arrival of new data (1) triggers the event procedure (2), and then the relevant MLMs (3), which represent the knowledge base, are executed by the inference engine (4). A results table containing a summary of the diagnosis and the order for blood products is provided (5). Initial data are sent automatically by the hospital information network or entered manually by a physician.

A. Construction of the Domain Ontology using UMLS

To facilitate sharing and reuse, we have chosen UMLS, elaborated to facilitate the integration of information from multiple biomedical sources, to constitute the repository of entities and relations from which the expert builds the domain ontology. Three components constitute the UMLS. The Metathesaurus contains a collection of biomedical concepts and their inter-relations. The Semantic Network, through its high level semantic types, provides a consistent categorization of all concepts represented in the Metathesaurus. The Specialist Lexicon is an

English language lexicon with biomedical terms. The lexicon entries for each word or term records syntactic, morphological, and orthographic information. UMLS provides a shared source of terminology widely used in clinical applications[27-30]. In order to navigate between concepts through the concept hierarchy and through the semantic network, we have designed a specific navigator[31].

1. The UMLS Navigator

The conceptual model (Figure 2) of our navigator is based on two components of the UMLS: the Metathesaurus and the Semantic Network. We have adopted an object-oriented representation. *Concept* is the main class of our Metathesaurus model. It includes several attributes (for example *ConceptDefinition*, *ConceptLinks*,etc.). It reflects properties present in UMLS; each concept has a unique identifier. Different terms with the same meaning are linked to the same concept identifier. We have reified the link (*LinkConceptToVariantes*) between a concept and these terms (preferred term, synonyms, etc). The relation (*ConceptLink*) between two concepts has three parameters represented as classes: a *ConceptRelation* (parent, child, general, specific...), a *ConceptRelationAttribut* (isa, inverse_isa...) and an *InformationSource* (Mesh98...) that provides it. Our representation of the UMLS semantic network is then a reified graph whose edges are the semantic relations (*SemanticLinks*) such as affected_by, occurs_in., and vertices are the semantic types (*SemanticType)* or semantic relations (*SemanticRelation*).

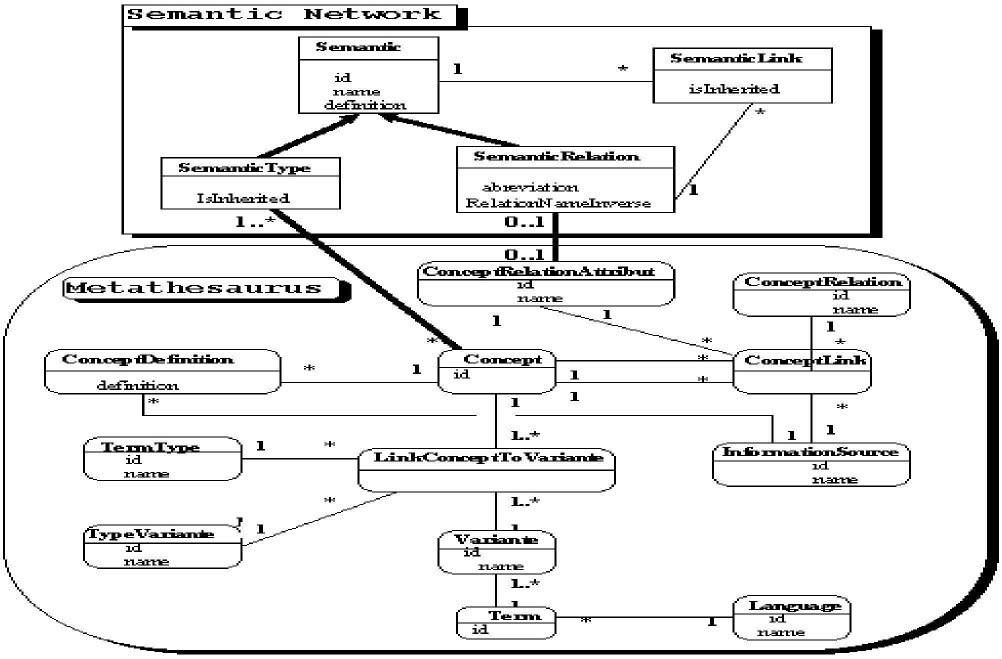

Figure. 2. Conceptual model of UMLS including Metathesaurus and Semantic Network (OMT-diagram[34]).
The lines indicate relations between classes.

2. Domain Ontology Construction

The construction of the domain ontology requires four steps. First, the expert identifies the medical terms used in the specific domain. Using our specific browser (Figure 3), s/he looks in the UMLS metathesaurus for the corresponding concepts, selects any relevant concepts and adds them to create a list. To facilitate comprehension and to enrich her/his list s/he then identifies for each concept its ascending hierarchy (i.e. its parents). Depending on the source vocabulary within the Metathesaurus, a concept can be found at several levels within a specific ascending hierarchy. Figure 4 shows the different possible ascending hierarchies for *hemolytic autoimmune anemia* term. The expert selects the hierarchy that corresponds to the semantic meaning s/he associates with a specific concept, or constructs her/his hierarchy by choosing the concepts from the different sources.

Figure. 3. Our UMLS Navigator GUI.
The figure shows the exploration of the Metathesaurus for the anemia concept.

Because strong standardization is not always possible in medicine, a medical expert can insert new terms or extend the initial hierarchy corpus to take into account particular considerations[32]. For term insertion, two constraints must be respected: the new concept must be linked to a semantic type from the UMLS semantic network and must have a parent. If a specific classification is constructed, it is saved in the UMLS format (MRCXT table) with a specific tag (PERSO) in the raw source (SAB).
All the concepts extracted from the UMLS or created have a local identifier, and are stored in a local table. This local table allows the link with the clinical database of the hospital. When introducing new terms and classification, the expert has the responsibility, for checking their adequacy and their consistency with the set of information contained in PERSO. As semantic types are general, an intermediate layer, named categories, has been added to introduce more specific types. The categories correspond to the concepts located at the top of the hierarchy of the contextual list selected from those present in the Metathesaurus or created by the expert. For instance, *Anemia Sickle Cell* is initially linked to the *Diseases* or *Syndrome* semantic types. We have introduced the category *Hematological Diseases* as a subclass of *Disease* or *Syndrome* semantic type, to which *Anemia Sickle Cell is linked* (see Figure 5).

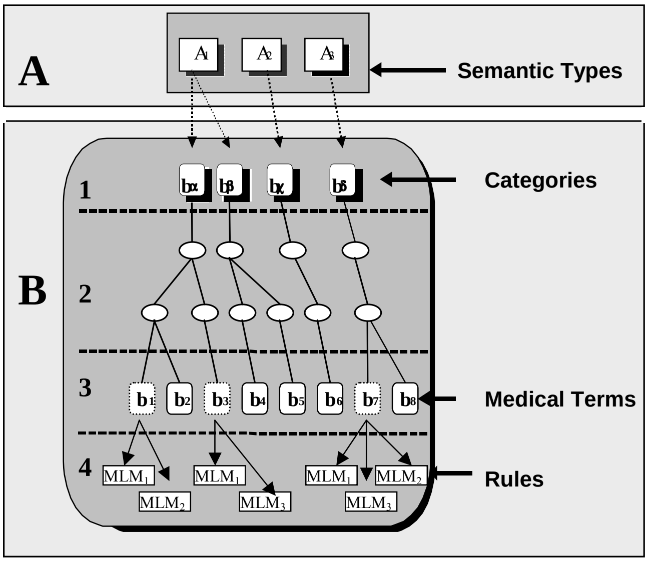

Figure. 4. General outline of the ontology.
A: The semantic types in relation with categories.
The expert also assigns the semantic types from the semantic network for any new concept added.
B: 1) Categories that subsume the medical terms used, 2) Concepts in relation with medical terms according to a hierarchic classification, 3) Medical terms used, 4) some MLMs where b_1, b_3 and b_7 are present in condition or action parts.

In the fourth and final step, following the classification of the terms according to an ascending hierarchy, all possible relations among these terms are automatically determined. For this purpose, a query is performed on the UMLS semantic network, to extract relationships between semantic types and concepts useful to represent the expert's knowledge. Then, the expert selects the relevant relationships between concepts among those present in the Metathesaurus. Once again the expert may add new relationships to this contextual list (identified with the label «PERSO »). During the ontology design, a local identifier (CLI) is automatically created for each concept acquired. This CLI is stored in the slot data of the MLM during the knowledge acquisition process. A local table is then designed to link and translate these local identifiers to the clinical database identifiers (CDBI).

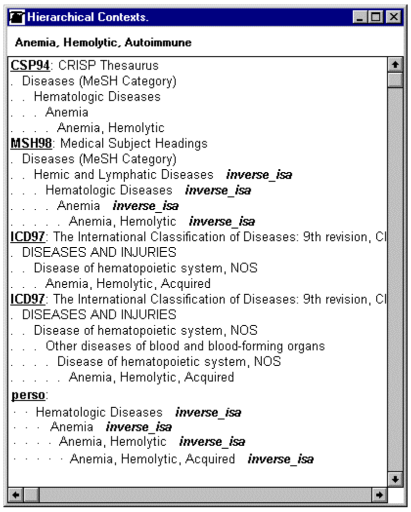

Figure 5. Selection of the relevant hierarchy from different sources. A specific popup menu (not shown here) allows for the construction of our personal hierarchy (namely perso).

From the development of the domain ontology, three different structures are obtained: 1) the hierarchy of terms and concepts used in the domain. These terms have an ascending hierarchical context, connected by the relation “inverse_isa”; 2) the concepts located at the top of the hierarchy that have been selected will compose the list of categories that subsume the medical terms used; and 3) the UMLS semantic network excerpt covering the concepts selected for the domain, such that each of the terms has a semantic type and a category.
For the collection of terms present (n=120), in the blood transfusion domain, the medical expert found corresponding UMLS Metathesarus conceptsfor 88 (i.e., 73%). The blood transfusion domain used highly specialized terms such as "transfusion of red blood cell plasma depleted", " transfusion of red blood cells phenotyped ", " patient erythrocyte alloimmunised ", etc. which were not present in the version of the Metathesaurus used. The coverage of disease terms qualified by such adjectives as minor, severe, or chronic was uneven.

B. Domain Knowledge Construction

Starting from the predefined domain ontology, the expert creates the domain knowledge needed by the application. We have modeled the knowledge with production rule templates. Our knowledge acquisition tool guides the user through three steps of rule description: 1) documentation following the Arden syntax definitions; 2) specification of the condition part of each rule; and 3) specification of the action part of each rule. The entities that compose the conditions and action parts of the rules come from the domain ontology and are linked to the semantic types. To illustrate the process of rule specification, let us start with the rule template (see Figure 6) $(A_1) \longrightarrow (A_3)$. After selecting, A1 for the condition part, the expert can choose between b_α and b_β. The selection of b_α leads to three possible choices: b_1, b_2 and b_3, the expert chooses the concepts $\boldsymbol{b_1}$ and $\boldsymbol{b_3}$ for the condition part. A_3 then b_δ and then b_7 is selected for the action part leading to rule 1:

Rule1

IF condition **(b_1)** present **and**
condition **(b_3)** present
THEN action **(b_7)** activated

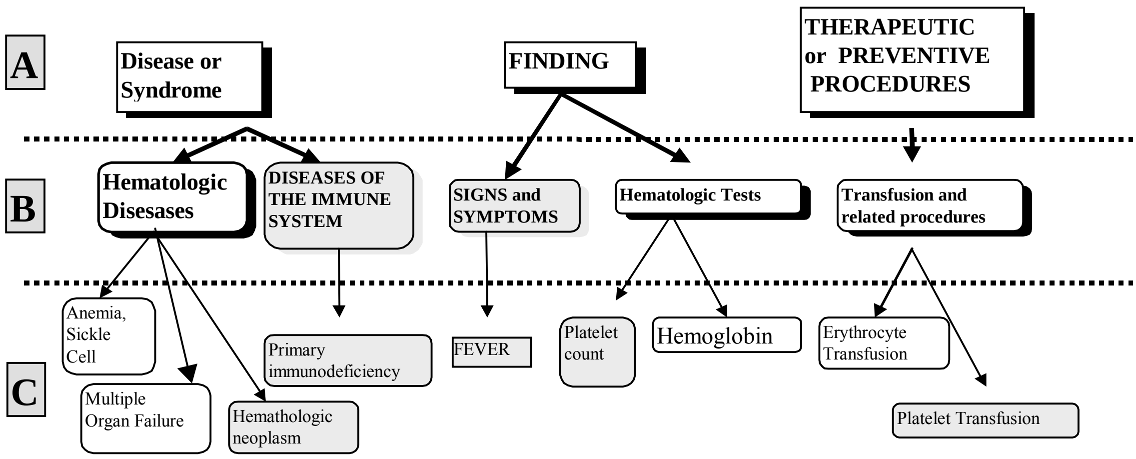

Figure. 6. An excerpt of our domain ontology based on the UMLS.
A: Semantic types, B: Categories, C: Terms used in the medical expertise. Rectangles with white background indicate entities used in rule 2 (see text).

1. Generation of Medical Logic Modules

A MLM, equivalent to a single rule in a rule-based expert system, contains enough medical knowledge and data to make a single clinical decision[8]. MLMs can be independent of one another but also can be linked by CALL statement. A MLM is an ASCII file composed of slots grouped into three categories: *Maintenance*, *Library* and *Knowledge*. The categories parts of the MLM are acquired via the KA-tool. The Figure 7 shows the interface for the construction of Rule 2.

Rule 2
IF (Sickle Cell Anemia)
AND (Multiple Organ Failure)
AND (Hemoglobin<10)
THEN (ErythrociteTransfusion)

In order to facilitate maintenance and avoid redundancy or inconsistency, MLMs are organized depending on the concepts they manipulate. The name of the MLM is stored in the slot *filename* of the category *Maintenance* and the concepts used in this MLM (e.g b_1, b_3, b_7 for Rule 1) are stored in the slot *keywords* of the category *library*. The concepts appearing in the condition part are linked, to the corresponding MLMs, by the relation "has as rule" to facilitate the retrieval of specific MLMs. The goal of each MLM, represented by its action, is stored in the slot *purpose* of the category *library*. The concepts appearing in the action part are linked to the corresponding

MLMs with the relation: "the action of ". This allows the expert to retrieve easily all MLMs that trigger a specific action. Finally, this knowledge is formalized according to the Arden syntax with MLMs linked to the domain ontology.
One hundred and ten rules (translated in MLMs) entail the knowledge base. The indexing and the referencing of the terms that compose each MLM are used to identify possible redundancies and inconsistencies during the creation of a new MLM. In the current prototype, the checking for redundancy and inconsistency is performed manually. Inheritance property is not used during MLM construction.

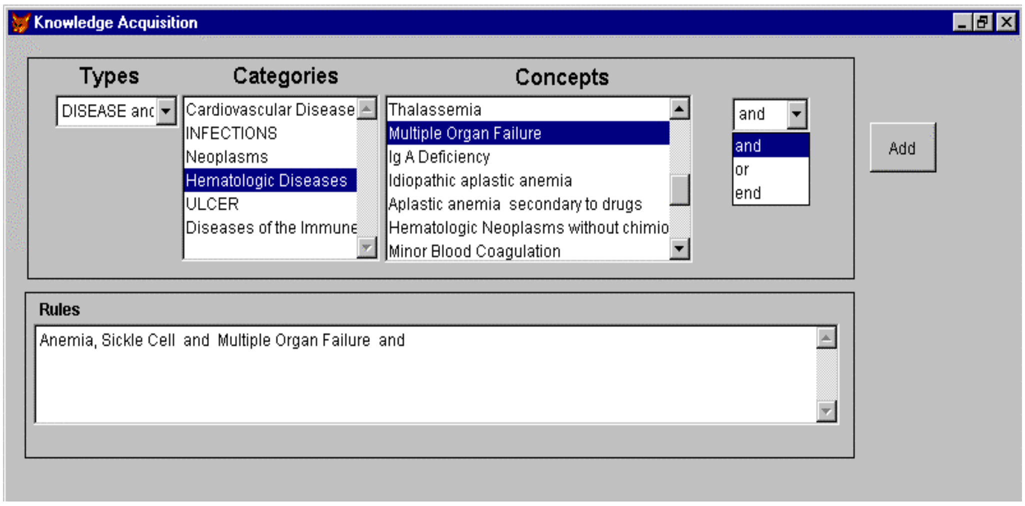

Figure. 7. Knowledge acquisition interface.
Three list boxes allow the selection of the corresponding entities that appear in condition or action parts of the rule. The rule under construction is shown at the bottom of the figure.

III. Experience with a CDSS for Blood Transfusion

The transfusion of blood products is an inescapable therapy for certain pathologies that has the potential to induce undesirable immunological and infectious effects, like HIV or Hepatitis. The best guarantee for the patient's safety is strict adherence to transfusion guidelines. In addition, the transfusion of blood products is an expensive therapy that, in terms of public health, must be prescribed advisedly. A CDSS that assists the prescription of blood products seems to be an appropriate tool for enhancing patient safety and reducing heath care costs. However the expansion of medical knowledge about blood transfusion and changes in government regulations impose a requirement for frequent adaptations of decision rules for blood transfusion.

At the Henri Mondor Hospital (Créteil, France), approximately 30,000 blood products per year are distributed to 5,000 patients. Such volume justifies the development of a CDSS integrated into the hospital information network to assist the clinician in blood product prescriptions. We constructed a CDSS for blood transfusion using the methodology previously described.

A. Architecture of the CDSS

Starting from blood data (essentially hemoglobin and platelet concentrations) and the patient's state (disease, current therapy, ect…), the system indicates whether transfusion is required and if so, which type and quantity of product to transfuse. It works according to the data-driven and application-driven principles of the MLM-controller[15, 16] that schedules the MLMs execution. The CDSS is composed of three modules (see Figure 1): 1) The user interface to retrieve information from the patient database, 2) The knowledge base that contains expertise for blood transfusion in MLMs, 3)The inference engine that executes the relevant MLMs. Two modes are available: alert (data-driven) and consultation (application-driven).

Alert: When biological data are automatically sent into the patient database (1), the event procedure is triggered (2). This activates (3) several MLMs executed by the inference engine (4). Actions are performed and a table of results is generated (5). Depending on the seriousness of the diagnosis an alarm signal could be set on. These steps are shown in Figure 1.

Consultation: The physician keys-in data about the patient as well as the prescription specifying the type and quantity of products to be transfused. Then, the system determines whether the transfusion is appropriate and the type, qualifier and quantity of the product to be transfused. Details of the system's decisions are displayed to the physician. If s/he disagrees, s/he can order other products and justify her/his choices in a brief report. A complete report including the system's and the physician’s recommendations is sent to the blood bank.

B. Preliminary results

A first evaluation was performed in order to validate the coherence and the correctness of the CDSS. For that, thirty orders were studied, which demonstrated the reliability of transfusion order (98% for platelet and 97% for red cells transfusion). A discrepancy was observed between the system and the physicians in 18 of 30 cases (60%) for the qualifiers of the product to transfuse such as phenotyped and/or compatibilized.

Analysis of these discrepancies revealed a need to add more information about the patient's status and to modify rules in the knowledge base. Twenty new orders were then evaluated using the modified CDSS. Some disagreements persisted between physician's order and CDSS's conclusions (40 %), but there was 100% agreement between the CDSS and the experts who reviewed the orders. This result was explained by the fact that physicians did not follow the current transfusion guidelines. A working committee was appointed to modify current practice, to adhere to new transfusion guidelines. The prototype is presently being evaluated at the Henri Mondor's Hospital.

IV. Discussion

We argue that our approach is a practical way to build CDSSs: it 1) avoids constructing the domain specific ontology from scratch, 2) limits the introduction of idiosyncratic terms and relations and 3) combines reuse and sharing with domain-specific lexicons familiar to clinicians. In our experiment, the UMLS provides a useful corpus of medical knowledge for designing a domain specific lexicon. Parallel to efforts to define a specific knowledge representation language for clinical concepts[4], UMLS constitutes a pragmatic way to facilitate knowledge sharing and reuse. There have been many projects to quantify the content coverage of the UMLS in several fields: clinical radiology[27], laboratory terminology[28], surgical procedures[30] and in hypertension notes[33]. These projects showed limitations of Metathesaurus terms in specific areas, demonstrating the necessity to add new terms and relations for a particular use of UMLS. Nevertheless, like other investigations[34] we consider that UMLS is a useful formal framework, easily available to medical institutions and continuously updated by NLM. The introduction of new terms besides those found in UMLS, reflects the specific needs of the expert and respects the philosophy of the UMLS, that concatenates several databases to gather the particularities of each of them.

We have designed a domain specific interface to allow the domain expert to input directly or modify existing knowledge in a CDSS using domain ontology. Our KA-tool uses rule templates appear well adapted to the decision-making process in the clinical applications we envisage. Our tool is designed for medical experts without specific training in medical informatics and allows their active participation during the process of CDSS construction. This can facilitate future integration of the CDSS into the hospital system. Domain knowledge is automatically transformed into MLMs in Arden syntax so that it can be reused and shared by other institutions. Our KA-tool is not only a MLM editor. It allows the acquisition of knowledge through the predefined domain ontology. It is in the same vein that the work of Thurin et al.[35] who proposed a tool for building MLMs from a GALEN based terminology. Arden syntax is mature enough to be used successfully in large working projects in clinical institutions. However, it does have drawbacks such as a style of rule representation that is not necessarily adapted to all clinical applications, low expressiveness of temporal primitives, and possible interactions between MLMs that can pose long-term maintenance problems.

In our system, the introduction of new terms, during domain ontology acquisition, can generate redundancies and conflicts with terms already present in the database. During domain knowledge acquisition, some conflicts may appear between rules. In the current implementation, the domain expert receives relatively little help from the system in solving redundancies and conflicts. Temporal aspects of the domain are currently not taken into account. Clearly, our tool should be extended to incorporate specific mechanisms to deal with these drawbacks.

To date we have used the prototype only in collaboration with two medical experts involved in this project in our hospital. Their involvement since the start of the project clearly biases their fully positive opinion. However, we believe that our tool will be accepted by end-users in those medical fields where decision support systems have to be easily modified by medical experts themselves. Creating domain ontologies that are reusable in new medical applications is a challenging task[36]. Our experiment provides a practical example of how large resources such as UMLS, can be used to design a domain ontology and how that ontology can be put to use.

Results reached in the area of blood transfusion demonstrate that UMLS provides a useful and manageable corpus of medical knowledge for domain ontology construction.

References

1. Musen MA. Dimensions of knowledge sharing and reuse. *Comput Biomed Res*. 1992;25:435-67.
2. Chandrasekaran B. Generic tasks in knowledge-based reasoning: High-level building blocks for expert system design. *IEEE Expert*. 1986;1:23-30.
3. Steels L. Components of expertise. *AI Magazine*. 1990;11:28-49.
4. Rector AL, Bechhofer S, Goble CA, Horrocks I, Nowlan WA, Solomon WD. The GRAIL concept modelling language for medical terminology. *Artif Intell Med*. 1997;9:139-71.
5. Shahar Y. A framework for knowledge-based temporal abstraction. *Artif Intell*. 1997; 90:79-133.
6. Genesereth MR, Fikes RE. Knowledge Interchange Format , Version 3.0. : Stanford University; 1992.
7. Farquhar A, Fikes R, Rice J. The Ontolingua Server: a Tool for Collaborative Ontology Construction. *Proc of Knowledge Acquisition Workshop*. Canada; 1996:1-19.
8. Hripcsak G, Ludemann P, Pryor TA, Wigertz O, Clayton PD. Rationale for the Arden Syntax. *Comput Biomed Res*. 1994;27:291-324.
9. Ohno-Machado L, Gennari JH, Murphy SN, et al. The guideline interchange format: a model for representing guidelines. *J. Am. Med. Inform. Assoc.* 1998;5:357-72.
10. Fox J, Johns N, Rahmanzadeh A. Disseminating medical knowledge: the PROforma approach. *Artif Intell Med*. 1998;14:157-81.
11. van Heijst G, Lanzola G, Schreiber T, Stefanelli M. Foundations for a methodology for medical KBS development. *Knowledge Acquisition*. 1994;6:395-433.
12. Musen MA. Domain ontologies in software engineering: use of Protege with the EON architecture. *Methods Inf Med*. 1998;37:540-50.
13. Rector AL, Nowlan WA. The GALEN project. *Comput Methods Programs Biomed.* 1994;45:75-8.
14. Hripcsak G, Clayton PD, Pryor TA, Haug P, Wigertz OB, Van der lei J. The Arden Syntax for medical logic modules. *Proc Annu Symp Comput Appl Med Care*; 1990:200-204.
15. Hripcsak G, Clayton PD, Jenders RA, Cimino JJ, Jhonson SB. Design of a clinical event monitor. *Comput Biomed Res*. 1996;29:194-221.
16. Gao X, Johansson B, Shahsavar N, Arkad K, Ahlfeldt H, Wigertz O. Pre-compiling medical logic modules into C++ in building medical decision support systems. *Comput Methods Programs Biomed*. 1993;41:107-19.
17. Pryor TA. The use of medical logic modules at LDS hospital. *Comput Biol Med*. 1994;24:391-5.
18. Jenders RA, Hripcsak G, Sideli RV, et al. Medical decision support: experience with implementing the Arden Syntax at the Columbia-Presbyterian Medical Center. *Proc Annu Symp Comput Appl Med Care*; 1995:169-73.

19. Gao X, Shahsavar N, Arkad K, Ahlfeldt H, Hripcsak G, Wigertz O. Design and fuctions of medical knowledge editors for the Arden Syntax. . *Medinfo*; 1992:472-477.
20. Jenders RA, Dasgupta B. Assessment of a knowledge-acquisition tool for writing Medical Logic Modules in the Arden Syntax. *Proc Annu Symp Comput Appl Med Care*; 1996:567-71.
21. Sailors RM. MLM Builder: an integrated suite for development and maintenance of Arden Syntax medical logic modules. *Proc Annu Symp Comput Appl Med Care*; 1997:996.
22. Spackman KA, Beck JR. A knowledge-based system for transfusion advice. *Am J Clin Pathol*. 1990;94:S25-9.
23. Marconi M, Almini D, Pizzi MN, et al. Quality assurance of clinical transfusion practice by implementation of the privilege of blood prescription and computerized prospective audit of blood requests. *Transfus Med*. 1996;6:11-9.
24. Gardner RM, Golubjatnikov OK, Laub RM, Jacobson JT, Evans RS. Computer-critiqued blood ordering using the HELP system. *Comput Biomed Res*. 1990;23:514-28.
25. Lepage E, Traineau R, Marchetti P, Benbunan M, Gardner RM. Development of a computerized knowledge based system integrated to a medical workstation: Application to blood transfusion. *Medinfo*; 1992:585-590.
26. Lepage E, Gardner RM, Laub RM, Golubjatnikov OK. Improving blood transfusion practice: role of a computerized hospital information system. *Transfusion*. 1992;32:253-9.
27. Friedman C. The UMLS coverage of clinical radiology. *Proc Annu Symp Comput Appl Med Care*; 1992:309-13.
28. Cimino JJ. Representation of clinical laboratory terminology in the Unified Medical Language System. *Proc Annu Symp Comput Appl Med Care*; 1991:199-203.
29. Campbell KE, Oliver DE, Spackman KA, Shortliffe EH. Representing Thoughts Words, and Things in the UMLS. *J. Am. Med. Inform. Assoc.* 1998;5:421-31.
30. Burgun A, Bodenreider O, Denier P, et al. Knowledge acquisition from the UMLS sources: application to the description of surgical procedures. *Medinfo*. 1995;8 Pt 1:75-9.
31. Achour SL, Dojat M, Brethon J-M, Blain G, Lepage E. The use of the UMLS Knowledge Sources for the Design of a Domain Specific Ontology : a Practical Experience in Blood Transfusion. *Proc of Conference on Artificial Intelligence in Medicine and Medical Decision Making*; 1999:249-253.
32. Dojat M, Pachet F. Effective domain-dependent reuse in medical knowledge bases. *Comput Biomed Res*. 1995;28:403-432.
33. Campbell JR, Kallenberg GA, Sherrick RC. The clinical utility of META: an analysis for hypertension. *Proc Annu Symp Comput Appl Med Care*; 1992:397-401.
34. Campbell KE, Oliver DE, Shortliffe EH. The Unified Medical Language System: toward a collaborative approach for solving terminologic problems. *J. Am. Med. Inform. Assoc.* 1998;5:12-6.
35. Thurin A, Carlsson M, Gill H, Wigertz O. Arden syntax and GALEN terminology support: a powerful combination to represent medical knowledge. *Medinfo*. 1995;8 Pt 1:110.
36. van Heijst G, Falasconi S, Abu-Hanna A, Schreiber G, Stefanelli M. A case study in ontology library construction. *Artif Intell Med*. 1995;7:227-55.

37. Booch G. Object-Oriented Design with Applications. . Menlo Park (Ca): Benjamin/Cummings; 1990.