Knowledge Organisation in Medical KBS Construction*

A. Th. Schreiber ‡ G. van Heijst ‡ G. Lanzola § M. Stefanelli §

‡ University of Amsterdam, Social Science Informatics
Roetersstraat 15, NL-1018 WB Amsterdam, The Netherlands
E-mail: {schreiber,gertjan}@swi.psy.uva.nl

§ University of Pavia, Medical Informatics
Via Abbiategrasso 209, I-27100, Pavia, Italy
E-mail: giordano@ipaimed1.unipv.it, stefa@ipvccn.bitnet

Paper proposal for AIME'93

Abstract

In this paper we present an analysis of the structure of medical knowledge for the purpose of the construction of knowledge based systems. We discern three different views on knowledge that each have a different but complementary function with regard to reusability. We show that each of these views suggests distinct activities in the knowledge engineering process. The reported knowledge organisation is a key element of the the GAMES methodology for the construction of medical knowledge based systems.

1 Introduction

In this paper, we discuss structuring principles of medical domain knowledge in knowledge-based applications. We take a knowledge-level [Newell, 1982] viewpoint, abstracting from potential symbolic representations of the knowledge. We distinguish three views on medical knowledge, namely (i) the *vocabulary* of basic medical terms, (ii) the knowledge *types* distinguished in an application, comprising the *ontology* of the application, and (iii) the knowledge *roles*, pointing to the way in which knowledge is used during problem solving.

Multiple views on medical knowledge are useful for a number of purposes. In this paper we focus on two aspects:

*The research reported here was carried out in the course of the GAMES-II project. This project is partially funded by the AIM Programme of the Commission of the European Communities as project number A2034. The partners in this project are SAGO (Florence, Italy), Foundation of Research and Technology (Crete, Greece), Geneva University Hospital (Switzerland), the University of Amsterdam (The Netherlands), University College of London (UK), the University of Pavia (Italy) and the University of Ulm (Germany).

This paper reflects the opinions of the authors and not necessarily those of the consortium.

- The identification of generic, reusable knowledge elements.
- The rationale for specifying activities in knowledge-based system (KBS) construction.

The organisation of medical knowledge as discussed in this paper in one of the corner stones of the AIM project GAMES-II. The goal of this project is to develop a methodology, including theory and practical tools, for structured medical KBS construction (see [van Heijst *et al.*, 1993] for an overview of the GAMES methodology). This paper reports on ongoing work and results within GAMES in the area of knowledge organisation and ontologies. We do not claim that all this is novel. In fact, our aim is to integrate results from knowledge engineering research and specialise these for use in medical applications. In the area of knowledge reuse our ideas are very much in line with [Musen, 1991].

In Sec. 2 we discuss in more detail the notion of knowledge-level modelling and the multiple views of medical knowledge used in GAMES. Subsequent sections describe how this view can be exploited for defining generic libraries of knowledge elements (Sec. 3) and for describing knowledge engineering activities (Sec. 4). Sec. 5 discusses related work and sketches a research agenda that follows from this work.

2 Knowledge Organisation

2.1 Background: knowledge-level modelling The first generation knowledge-based systems employed one relatively simple *inference engine* working on a knowledge base in a particular representational format, usually production rules. Clancey showed in his analysis of the prototypical system of this generation, MYCIN that such a knowledge base hides various important properties of the reasoning process, and of the structure of the knowledge in the application domain [Clancey, 1983]. Certain rules, or parts of rules, fulfill particular roles in the reasoning process which remain implicit in such a KBS organization. This implicitness of underlying structures impairs the acquisition and refinement of knowledge for the KBS as well as the reuse of the system, its explanatory power and the assessment of its relation with other systems.

It is fair to say that this problem was not specific for the field of knowledge engineering. Similar problems were being identified in the broader area of knowledge representation. Clear evidence of this was brought forward by Brachman and Smith through the results of their SIGART questionnaire [Brachman & Smith, 1980].The aim of this questionnaire was to get data on various knowledge representation approaches in order to perform a comparative study. About the results of their analysis of the huge amount of data received, they remark:

> "Perhaps more than anything else, it has emerged as a testament to an astounding range and variety of opinions held by many different people in many different places." [Brachman & Smith, 1980, p. 1]

Everyone seemed to be speaking a different language: a true Babel.

In response to this confusion, Newell coined, in his presidential address to AAAI-80, the "knowledge-level hypothesis". The key point underlying his argument was that the confusion arose because AI research was too much focused on detailed representational issues. What was missing was a description of the rationale behind the use of AI techniques. He pleaded for a shift of emphasis in AI research from the "how" questions to the "why" questions. The knowledge-level was his proposal for realizing a description of an AI system in terms of its rational behaviour: why does the system (the "agent") perform this "action", independent of its symbolic representation in rules, frames or logic (the "symbol level").

2.2 Multiple views on medical domain knowledge During the last decade, a number of proposals have been put forward in knowledge engineering research for describing knowledge-level models [Clancey, 1985, Neches *et al.*, 1985, Wielinga & Breuker, 1986, Chandrasekaran, 1988, Marcus, 1988, Musen, 1989, Steels, 1990, Schreiber *et al.*, 1993, Wielinga *et al.*, 1993, Ramoni *et al.*, 1992]. A common distinction that is being made is between (i) domain knowledge, defining a declarative theory of the application domain, and (ii) control knowledge which specifies how to use domain knowledge to solve a problem.

In this paper we focus on the description of domain knowledge and its relation with control knowledge. Although terminology varies, there appear to be at least three descriptive levels of domain knowledge that can be found in most approaches:

Application ontology

The application ontology, in short "ontology" [1], specifies the structure of the domain knowledge in terms of a number of (domain) knowledge *types*. It characterises the types of objects and expressions that one finds in the domain knowledge.

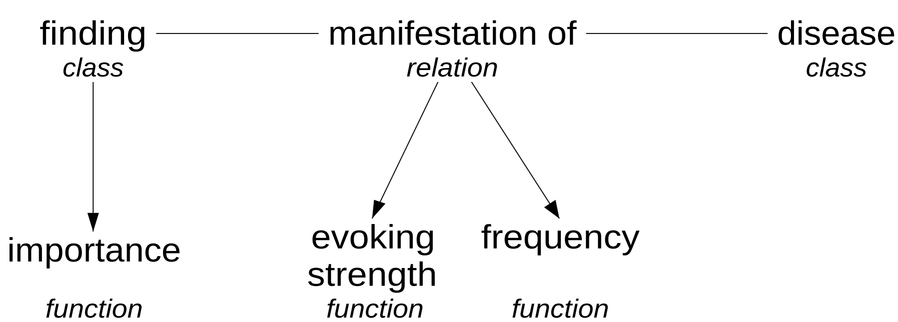

FIGURE 1: Example domain knowledge types, derived from the ontology underlying INTERNIST-1 [Miller *et al.*, 1982]

Fig. 1 shows a partial example ontology, derived from the way in which domain knowledge is described in INTERNIST-1 [Miller *et al.*, 1982]. As [Davis *et al.*, 1993] point out, there is an additional underlying ontological commitment in such an ontology, namely the representational primitives that are used for describing the ontology. In GAMES we have chosen the primitives used by Gruber in Ontolingua [Gruber, 1992]: classes, relations and functions.

Application knowledge

The application knowledge constitutes the "actual" domain knowledge. Roughly, it consists of two parts:

[1] In a previous publication [Wielinga *et al.*, 1992] the first author favoured the term "schema" above "ontology" to stress that it is product of engineering, and does not necessarily imply a faithful description of the real world. The use of the second term has, however, become predominant in the literature.

1. *Domain vocabulary* The set of terms that the application employs to describe the application domain. Example medical terms are "liver size", "enlarged" and "veno occlusive disease".
2. *Domain models* Domain models are sets of expressions about how domain terms are related. An example expression is the statement that the finding that the "liver size" is "enlarged" is a manifestation of "veno occlusive disease". Domain models contain relation tuples and function expressions of which the underlying structure is defined in the ontology.

From a logicist stance the domain vocabulary and the domain models can respectively be termed the universe of discourse and sets of axioms about elements of this universe.

Loosely speaking, one could view the application knowledge as an instantiation of the ontology. However, one should realise that medical terms are objects in their own right. The fact that some term, say "headache" is viewed as being of type "observable" is dependent on the application domain. In some other application, it might well be a disease! Thus, it is probably more correct to view the relation between the application knowledge and ontology as a mapping relation.

Knowledge roles

Knowledge roles are names for domain knowledge elements that characterise their role during problem solving. Example role names are "hypothesis" and "expected datum". Knowledge roles are linked to the ontology to specify which elements can play this role. For example, in a diagnostic application the role "hypothesis" will usually be linked to an ontological type "disease", which in turn can be mapped on a set of medical terms representing the diseases in the domain.

Knowledge roles specify how domain knowledge is manipulated by the control knowledge. In GAMES this control knowledge consists of an *inference model* that specifies the basic inference steps, and a set of task descriptions specifying how inferences can be ordered to reach a problem solving goal (see for more detailed descriptions [Ramoni *et al.*, 1992]). Knowledge roles are part of the inference model. Fig. 2 shows the generic version of the inference model used in GAMES.

Fig. 3 summarises the three views by organising them in three levels and indicating typical mappings between these levels. The levels can be seen as *attributing various types of semantics* to domain knowledge elements. This is in contrast with the the traditional logicists view of model-theoretic semantics, which implies a description of semantics at one level.

1. The *vocabulary* of the application knowledge defines medical terms that are used in the application. Although the particular set of terms in a vocabulary is geared towards the application, the terms *themselves* are expected to belong to the field of medicine in general. The semantics of terms like "lateral side of the femur" is determined by general medical knowledge, in this case anatomical knowledge. We will argue in the next section that this effects the way in which we should view reusability of medical terms.

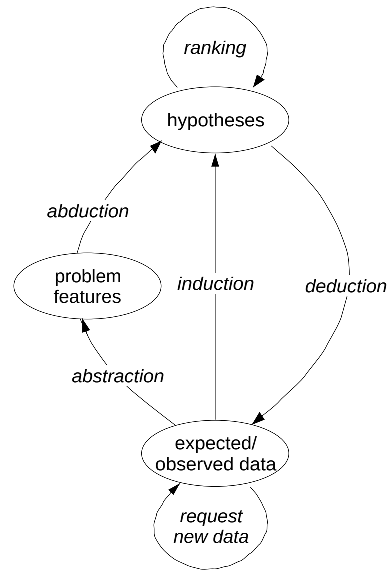

FIGURE 2: General "Select and Test" (ST) model of problem solving used in GAMES. The names within the ellipses represent knowledge roles. Italic names denote inference steps. See for a detailed description [Ramoni *et al.*, 1992]

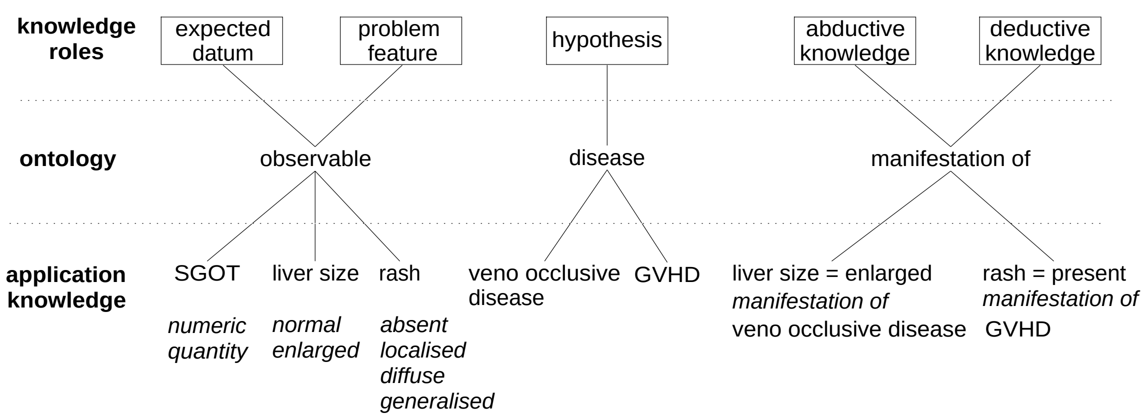

FIGURE 3: Example knowledge elements illustrating the three-level organisation of medical domain knowledge

2. The *ontology* defines additional semantics of application knowledge. We say that some term like "liver size" is interpreted as an "observable" in our application domain. Unlike the vocabulary, the conceptualisations that we chose for an application ontology are dependent both on the medical sub-domain, e.g. liver diseases, and on the problem solving task, e.g. diagnosis.
3. The *knowledge roles* specify the semantic interpretations that we attach to knowledge elements during problem solving. For example, the finding "liver size = enlarged" can play the role of expected datum for the hypothesis "veno occlusive disease", and be used to disconfirm or strengthen the belief in this hypothesis.

In the following sections we elaborate on how these different levels of knowledge description can be employed to aid the process of knowledge engineering.

3 Reusability

KBS construction is a time-consuming activity. Even building simple systems requires a considerable effort. Reusing parts of previously built KBSs is potentially a powerful approach to overcome these problems. Reuse changes the nature of knowledge engineering from a pure construction activity to a selection and refinement process of existing components. It helps turning knowledge engineering from an art into a real engineering discipline. In fact, it is probably fair to say that reuse is a necessary condition for building the complex systems we are ultimately aiming at.

It has been recognised that the knowledge-level stance is particularly useful for reusability purposes [Gruber, 1992, Musen, 1991, Wielinga *et al.*, 1992]. For reusability of knowledge, it is necessary to abstract from the symbol-level peculiarities of knowledge representation. Knowledge-level descriptions are intended to provide precisely this abstraction. In this section we consider how the knowledge organisation as sketched in the previous section can be employed to support reusability.

Library of problem solving methods Knowledge roles are part of an abstract model of how problem solving should be performed. In the literature such a model is called a *problem solving method* (PSM) [Musen *et al.*, 1987, McDermott, 1988, Steels, 1990]. The Select and Test model in Fig. 2 is the general PSM supported by GAMES. In addition, three specialised versions of this PSM were developed that are specific for prototypical medical tasks: diagnosis, therapy planning, and monitoring. These PSMs are used in GAMES as a *library* of PSMs from which the knowledge engineer can choose a PSM to realise a medical task in the application domain. Each PSM contains a set of specific knowledge roles for this task.

Library of ontologies In GAMES we have taken the position that it is not possible to fix in advance the ontology for applications. As pointed out in the previous section, the ontology is usually application specific and is dependent on the application task and the medical sub-domain. Fixing the ontology would constrain the task of building an adequate knowledge base containing adequate epistemological distinctions too much. However, each application ontology contains definitions which reappear in other applications. The approach we have taken in GAMES is to supply the knowledge engineer with a library of predefined partial ontologies. Through a process of selection and refinement, this library can be used to support the construction of the application ontology. We distinguish four kinds of reusable ontologies:

Ontologies for general medical entities With this we mean the terminology that is related to the medical practice. Here objects such as disease, observable, finding and therapy, and relations between these objects are defined. The ontology in Fig. 1 is ax example of such an ontology.

Generic medical and physiological ontologies These are ontologies which are domain dependent in the sense that they are related to medical domains, but which are generic in the sense that the same ontology is required in many different application areas of medicine.

Generic basic ontologies With these we mean ontologies which are reusable across a large number of domains. Examples of these are number-theory, scales and scale conversions, and time ontologies.

Meta-ontologies These ontologies are general systems for organising the world. An example of such an ontologies is the "frame ontology" underlying Ontolingua [Gruber, 1992] which defines primitive notions such as classes, objects, functions and relations. For practical purposes meta ontologies are usually fixed, but making them explicit facilitates extension and modification when required.

Fig. 5 in the next section shows a snapshot of part of a tool that was built in GAMES to support the selection and refinement process of partial ontologies.

Library of standard medical terms Currently, most knowledge-acquisition front ends for building medical KBSs simply ask a domain expert to type in a name when eliciting, for example, possible diseases in the domain. They do not check the correctness of the name (other than syntax or internal name clashes). They rely on the expertise of the user to enter correct medical terms. This situation seriously hampers the reuse and sharing of knowledge bases. Medical terminology contains many slight variations, such as the language used (Latin, native language), the position of words in composite terms such as "lateral side of the femur", etcetera.

There is a large amount of research in the field of standardising medical vocabulary, e.g. [Coté, 1982, Tuttle *et al.*, 1990]. It is outside the scope of GAMES to do any work in this area. Instead, our aim is to provide the necessary anchor points that enable the construction of bridges with terminological knowledge bases to ensure that terms in the application knowledge are consistent and potentially sharable. This is also one of the reasons why a clear distinction between application vocabulary and ontology is important.

4 Knowledge Acquisition Activities and Tools

It will be clear that in the GAMES view "knowledge acquisition" comprises more that just entering medical domain-specific knowledge such as the set of diseases and findings. A major part of knowledge acquisition concerns the knowledge-level description of the structure and the use of this domain knowledge during problem-solving. In GAMES the full knowledge-level model (comprising the three types of descriptions of domain knowledge plus the instantiated problem solving methods) is called the *epistemological model* [Ramoni *et al.*, 1992].

The multi-level knowledge organisation and their associated generic libraries provide also a rationale for identifying knowledge acquisition activities that need to be performed when building an application. Fig. 4 gives a schematic overview of these activities. We discuss each of these briefly, indicating also support tools for activities.

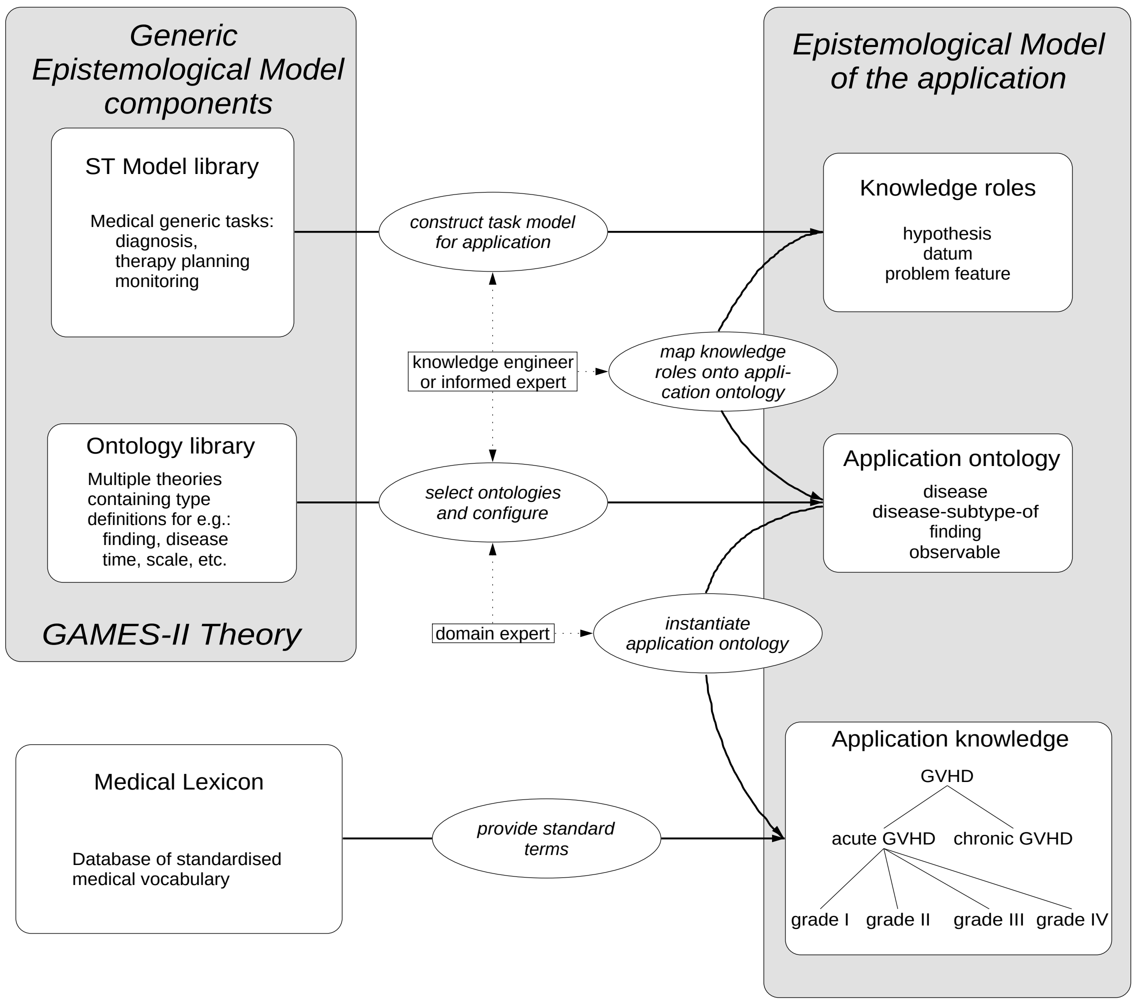

FIGURE 4: Schematic overview of the domain knowledge organisation in the GAMES epistemological model. The right-hand side represents the model of an application; the left-hand side denotes corresponding generic knowledge. GAMES supports the upper two types of generic knowledge (libraries of PSMs and ontologies). The solid arrows represent knowledge acquisition activities that need to be performed when building an application. The dotted arrows indicate the type of agent that is expected to carry out some activity

Task model construction The phrase "task model" is used in GAMES to refer to the control-knowledge description of some application. This activity uses the library of PSM specifications in order to build an inference model and a set of task definitions for the application. At the University of Pavia, a prototype system has been built that supports the construction of a task model using a metarule language with an associated metarule editor [Lanzola & Stefanelli, 1992, Lanzola & Stefanelli, 1993]. The system provides standard sets of metarules, representing the different library tasks. These metarules can be edited to meet specific demands of applications.

It should be noted that the task model of an application often contains a combination of PSMs. Many real-life medical tasks are combinations of the generic medical tasks represented in the library. For example, the task of managing graft-versus-host disease (GVHD) after bone marrow transplantation involves both a diagnostic task (establishing the presence of GVHD; grading of GVHD) as well as therapy planning activities. Typically, this involves multiple mappings between the ontology and the knowledge roles. For example, GVHD takes the role of "hypothesis" during diagnosis, but is considered a "problem feature" during therapy planning. Such multiple mappings constitute an additional rationale for the explicit distinction we make between ontological elements and knowledge roles.

Ontology specification The goal of this activity is to define a schematic description of the medical knowledge in some domain: the application ontology. This needs to be done by the knowledge engineer or an informed expert that takes the role of knowledge engineer. It is supported by a predefined library of ontology descriptions. The idea is that the knowledge engineer does not build the ontology from scratch, but can use existing ontologies from the library, and, if necessary, tune these to meet the demands of the application. For example, several ontologies of time exist, each with varying degrees of complexity. Monitoring applications usually require a much more explicit representation of time than diagnostic ones. Typically, the knowledge engineer will select the simplest time ontology required by the application.

In GAMES we use the Ontolingua language developed in the context of the knowledge sharing initiative in the US [Neches *et al.*, 1991, Gruber, 1992] to represent ontologies. We have built a tool that acts as a front-end to Ontolingua representations and allows both the construction of ontological theories as well as the selection and fine-tuning for a particular application. Fig. 5 shows the part of the tool that supports viewing and refining a theory (partial ontology) in the library.

Role mapping This activity defines how the elements of the medical ontology map onto roles in the inference model. This is largely predefined by the GAMES theory [Ramoni *et al.*, 1992], but can be modified by the knowledge engineer, This activity requires detailed knowledge of the GAMES theory and should thus be carried out by the knowledge engineer. The mapping is represented in the task model (see above) and is achieved in the prototype system through metarules.

Entering application knowledge The ontology defines the *format* for the application knowledge. The actual specification of the application knowledge in this format is a separate knowledge acquisition activity. This activity can and probably should be carried out by the domain expert. This activity has often been viewed as the only real "knowledge acquisition" activity. This was mainly due to the fact that the formats that were used by medical KBSs were fixed by the symbol-level representation of knowledge and their associated inference regimes.

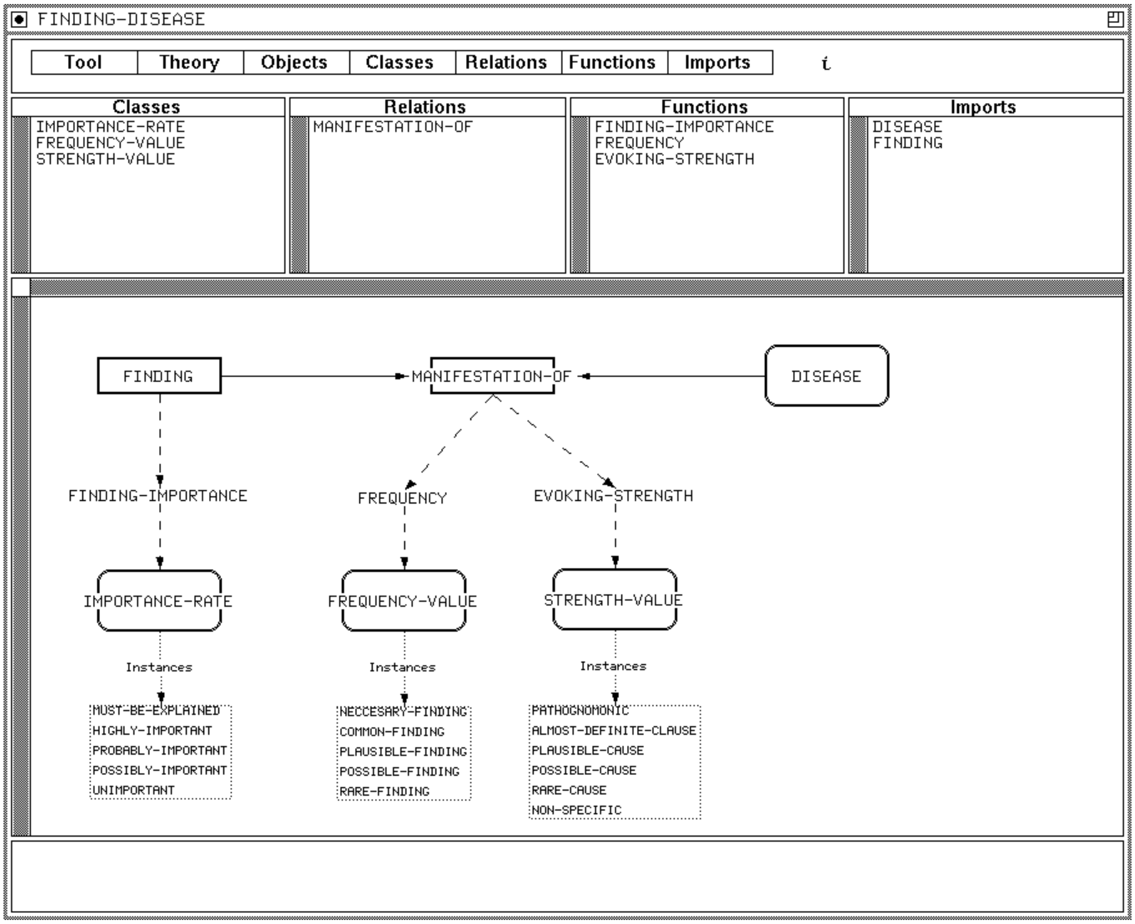

FIGURE 5: Screen dump of the part of the GAMES ontology editing tool that supports viewing and refining a theory (partial ontology) in the library. The ontology shown is derived from INTERNIST-1 (see also Fig. 1)

The aim of the GAMES approach is to allow experts to interact with a knowledge acquisition tool in a vocabulary that is meaningful to them. Similarly to OPAL [Musen *et al.*, 1987], we aim for a dialogue with the expert in terms of the ontology ("What are diseases in this domain?"), rather that in terms of knowledge roles ("What are hypotheses in this domain?") as in systems like MOLE and SALT [Marcus, 1988].

One point that arises from the discussion on standardised medical vocabulary in the previous section, is that one would like to support the entering by the expert of medical terms with tools that browse and/or search through terminological databases. This would ensure the presence of correct and potentially sharable terms in the knowledge base of the application. One of the longer term research goals of GAMES is to make such additional facilities available to the GAMES user.

5 Discussion

We have described the framework employed by the GAMES project for organisation of medical knowledge. This framework is intended to support maximisation of reusability and to provide a rationale for identification of KBS-construction activities.

In this paper we concentrated on knowledge-level modeling, focusing on the contents of the knowledge instead of the form (the symbol level). However, some words are in place on the computational aspects. In order to build a running KBS the selection of representation formalisms and reasoning techniques needs to be addressed as well. In GAMES we account for these decisions through the *computational model*: the symbol level counterpart of the epistemological model. The structure of the computational model is loosely based on the notion of a control blackboard architecture [Hayes-Roth, 1985], where the knowledge sources implement the inferences in the ST model. In principle, the knowledge sources can be realised with different problem solvers, thus allowing the use of multiple representations and multiple reasoning techniques in one KBS. We are currently experimenting with qualitative simulation techniques, causal-probabilistic networks and rule and frame based reasoners. Construction of the computational model involves two steps, namely (i) selecting appropriate problem solver for the inferences, and (ii) translating the ontology and the domain knowledge into the particular representation formalism of that problem solver. Note that we select the problem solvers after we have constructed a knowledge level model. This allows us to select problem solvers that are both epistemologically and computationally adequate for the knowledge and the task at hand. This form of integration, which we call *knowledge-level integration*, should be contrasted with the approach in hybrid systems because integration is realised on the basis of the contents of the knowledge instead of the form. The use of Ontolingua allows us to use the associated translation architecture for automatic translation of the application ontology and the application knowledge into the specific formalisms.

As pointed out in Sec. 1, the framework for knowledge organisation as described in this paper is derived from recent work in knowledge engineering. A relevant development in this field is the Sisyphus initiative [Linster, 1993] of which the aim is to come up with a common framework for describing problem solving methods and the ontologies these methods employ, which allows sharing and reusing PSMs across research groups. Our aim is to extend and specialise this research in a medical context. We view the following items as major research topics for further work in this area:

- Construction, validation and refinement of a library of PSMs that are specific for the medical domain.

- Construction, validation and refinement of a library of partial ontologies that are relevant in the medical domain. Example elements of such a library include partial ontologies of medical data, causal networks, disease taxonomies, time, etc.

- Experiment with links between knowledge acquisition tools and terminological databases.

- Establish the relationship between medical knowledge bases and medical information systems.

  In practice, KBSs are not used in isolation but have to be integrated with more conventional information technology such as hospital information systems. The framework outlined in this paper provides clear anchor points for establishing bridges between medical knowledge bases and databases, e.g. through shared ontologies.

Ultimately, this should pave the way the way for building a new generation of KBSs that meets the demands of medical practice.

Acknowledgements We are grateful to the other members of the GAMES-II consortium for fruitful discussions on the ideas presented in this paper.

References

[Brachman & Smith, 1980] BRACHMAN, R. J. & SMITH, B. C. (1980). Special issue on knowledge representation. *SIGART Newsletter*, 70:1–138.

[Chandrasekaran, 1988] CHANDRASEKARAN, B. (1988). Generic tasks as building blocks for knowledge-based systems: The diagnosis and routine design examples. *The Knowledge Engineering Review*, 3(3):183–210.

[Clancey, 1983] CLANCEY, W. J. (1983). The epistemology of a rule based system -a framework for explanation. *Artificial Intelligence*, 20:215–251.

[Clancey, 1985] CLANCEY, W. J. (1985). Heuristic classification. *Artificial Intelligence*, 27:289–350.

[Coté, 1982] COTÉ, R. A., editor (1982). *Systematized Nomenclature of Medicine*. Skokie, Illinois, College of American Pathologists.

[Davis *et al.*, 1993] DAVIS, R., SHROBE, H., & SZOLOVITS, P. (1993). What is a knowledge representation? *AI Magazine*, Spring:17–33.

[Gruber, 1992] GRUBER, T. (1992). Ontolingua: A mechanism to support portable ontologies. version 3.0. Technical report, Knowledge Systems Laboratory, Stanford University, California.

[Hayes-Roth, 1985] HAYES-ROTH, B. (1985). A blackboard architecture for control. *Artificial Intelligence*, 26(3):251–321.

[Lanzola & Stefanelli, 1992] LANZOLA, G. & STEFANELLI, M. (1992). A specialized framework for medical knowledge based systems. *Computers and Biomedical Research*, 25:351–365.

[Lanzola & Stefanelli, 1993] LANZOLA, G. & STEFANELLI, M. (1993). Inferential knowledge acquisition. *Artificial Intelligence in Medicine.* to appear.

[Linster, 1993] LINSTER, M. (1993). Special issue on sisyphus 91-92 "models of problem solving". *Knowledge Acquisition.* forthcoming.

[Marcus, 1988] MARCUS, S., editor (1988). *Automatic knowledge acquisition for expert systems.* Boston, Kluwer.

[McDermott, 1988] MCDERMOTT, J. (1988). Preliminary steps towards a taxonomy of problem-solving methods. In Marcus, S., editor, *Automating Knowledge Acquisition for Expert Systems*, pages 225–255. Boston, Kluwer.

[Miller *et al.*, 1982] MILLER, R. A., POPLE, H. E., & MYERS, J. D. (1982). INTERNIST-I: A general computer-based diagnostic consultant for general internal medicine. *New England Journal of Medicine*, 307:468–476. Also in: *Readings in Medical Artificial Intelligence: The First Decade*, W. J. Clancey and E. H. Shortliffe (eds.), Addison Wesley, Reading, MA, 1984.

[Musen, 1991] MUSEN, M. (1991). Dimensions of knowledge sharing and reuse. Report KSL-91-65, Knowledge Systems Laboratory, Stanford University, California. To appear in *Computers and Biomedical Research.*

[Musen, 1989] MUSEN, M. A. (1989). *Automated Generation of Model-Based Knowledge-Acquisition Tools.* London, Pitman. Research Notes in Artificial Intelligence.

[Musen *et al.*, 1987] MUSEN, M. A., FAGAN, L. M., COMBS, D. M., & SHORTLIFFE, E. H. (1987). Use of a domain-model to drive an interactive knowledge-editing tool. *Int. J. of Man-Machine Studies*, 26:105–121.

[Neches *et al.*, 1991] NECHES, R., FIKES, R., FININ, T., GRUBER, T., PATIL, R., SENATOR, T., & SWARTOUT, W. R. (1991). Enabling technology for knowledge sharing. *AI Magazine*, pages 36–56. Fall.

[Neches *et al.*, 1985] NECHES, R., SWARTOUT, W. R., & MOORE, J. D. (1985). Enhanced maintenance and explanation of expert systems through explicit models of their development. *IEEE Transactions on Software Engineering*, 11:1337–1351.

[Newell, 1982] NEWELL, A. (1982). The knowledge level. *Artificial Intelligence*, 18:87–127.

[Ramoni *et al.*, 1992] RAMONI, M., STEFANELLI, M., BAROSI, G., & MAGNANI, L. (1992). An epistemological framework for medical knowledge based systems. *IEEE Transactions on Systems, Man and Cybernetics*, 22:1361–1375.

[Schreiber *et al.*, 1993] SCHREIBER, A. T., WIELINGA, B. J., & BREUKER, J. A., editors (1993). *KADS: A Principled Approach to Knowledge-Based System Development.* Knowledge-Based Systems Book Series. London, Academic Press. ISBN 0-12-629040-7.

[Steels, 1990] STEELS, L. (1990). Components of expertise. *AI Magazine.* Also as: AI Memo 88-16, AI Lab, Free University of Brussels.

[Tuttle *et al.*, 1990] TUTTLE, M., SHERERTZ, D., OLSON, N., ERLBAUM, M., SPERZEL, D., FULLER, L., & NELSON, D. (1990). Using Meta-1 – the first version of the ULMS metathesaurus. In Miller, R. A., editor, *Proc. of the 14th Annual Symposium on Computer Applications in Medical Care*, pages 131–135, Washington DC. IEEE Computing Society Press.

[van Heijst *et al.*, 1993] VAN HEIJST, G., LANZOLA, G., SCHREIBER, A. T., & STEFANELLI, M. (1993). Methodological foundations of medical KBS construction. Technical Report GAMES-II/T1.1/UvA/PP/009/1.0, University of Amsterdam and University of Pavia. Available from: University of Amsterdam, Social Science Informatics, Roetersstraat 15, NL-1018 WB, Amsterdam, The Netherlands.

[Wielinga & Breuker, 1986] WIELINGA, B. J. & BREUKER, J. A. (1986). Models of expertise. In *Proceedings ECAI–86*, pages 306–318.

[Wielinga *et al.*, 1992] WIELINGA, B. J., SCHREIBER, A. T., & BREUKER, J. A. (1992). KADS: A modelling approach to knowledge engineering. *Knowledge Acquisition*, 4(1):5–53. Special issue 'The KADS approach to knowledge engineering'. Reprinted in: Buchanan, B. and Wilkins, D. editors (1992), *Readings in Knowledge Acquisition and Learning*, San Mateo, California, Morgan Kaufmann, pp. 92-116.

[Wielinga *et al.*, 1993] WIELINGA, B. J., VAN DE VELDE, W., SCHREIBER, A. T., & AKKERMANS, J. M. (1993). Towards a unification of knowledge modelling approaches. In David, J.-M., Krivine, J.-P., & Simmons, R., editors, *Second Generation Expert Systems.* Springer-Verlag.