TOWARDS AN INTEGRATION OF ENGINEERING KNOWLEDGE MANAGEMENT AND KNOWLEDGE BASED ENGINEERING

R□diger Klein
DaimlerChrysler Research and Technology
Knowledge Based Engineering Group
Alt-Moabit 96a
D-10559 Berlin
Germany

Abstract: Knowledge is one of the key issues in every business today. Computer based support is getting increasing importance here. This has currently mainly two main objectives: knowledge management and knowledge based systems. The complexity of engineering knowledge results in high requirements to the knowledge structuring and retrieval techniques to be used here. It is also valuable as a first phase in the life cycle of knowledge based engineering systems. An Informal Model is introduced here as a representation of knowledge suitable for engineering knowledge management as well as to support the development of knowledge based systems.

Keywords: engineering knowledge management, knowledge based engineering.

1 INTRODUCTION

Knowledge is a key issue in every business today. Though humans continue to be the main knowledge ”sources” and ”processors”, computers have a growing importance in the business of knowledge. Engineering knowledge management (EKM) and knowledge based engineering (KBE) are the two main knowledge oriented approaches today.

Knowledge management is mainly human oriented. Thus, social, cultural, organizational, and other aspects are as important in knowledge management as sophisticated information technologies. The new quality of *knowledge* management compared with traditional information systems is directly related to the complexity of knowledge. A traditional information system ”knows” exactly what it contains explicitly. It has no notion of and can not deal with implicit knowledge, tacit knowledge, background knowledge, or underlying assumptions. It does not know anything about interrelationships between ”pieces” of knowledge or about consistency.

It will remain a challenge for the foreseeable future to find well-founded and satisfying solutions for knowledge management in its full extend – like, for instance, natural language understanding. Nevertheless, the requirements from daily practice result *already today* in a strong pressure to introduce techniques which provide at least some intelligent support to knowledge management – using currently available information technologies.

The other main knowledge related issue in engineering is the application of knowledge based technologies, i.e., the automatic, computer based processing of knowledge in knowledge based engineering (KBE) systems. Again, the complexity of knowledge is challenging here. First generation systems (rule based or object oriented approaches) were useful to some extend but proved to be not powerful enough in general. They did neither enable the representation of the many interrelated engineering knowledge issues, nor was their reasoning power sufficient. Today the general belief is that *model based* systems are needed in engineering applications as well as in many other complex domains. Model based means that all aspects of knowledge are modeled explicitly – reflecting the large diversity of knowledge aspects. Expressive models of *domain knowledge* and of *problem solving knowledge* are required in order to represent the many knowledge issues in engineering (and elsewhere) in their diversity and interdependencies.

Unfortunately, there is a large gap between human-oriented knowledge like text documents, interview protocols, etc., and such expressive, formal knowledge models as used in model based systems. Currently, highly skilled knowledge engineers transform human knowledge into formal knowledge – sometimes with some support from knowledge acquisition tools. In any case, this is a very complex, laborious, expensive, and error-prone process.

A way out of this situation can be the *reuse* of knowledge. For this purpose, knowledge should be formulated as application-independent as possible: the domain knowledge as well as the problem solving knowledge. *Ontologies* can be used as conceptual backbone of such generic knowledge modules [1, 2]: they provide generic definitions of all notions and their interrelationships in a knowledge base. But ontology construction and ontology-based knowledge modeling are still an evolving technology [3].

In this paper we describe a *pragmatic* approach to knowledge modeling which tries to fulfill as many requirements as possible using currently available techniques. This approach has two main objectives:

- to support knowledge modeling for human oriented engineering knowledge management, and
- to reduce the gap between human-oriented and formal knowledge in model based engineering systems.

The key element of our approach is an *Informal Model* (IM). It provides a platform for human-oriented knowledge in EKM, and at the same time it can be used as an *intermediate* representation for engineering knowledge on the way from human to formal knowledge.

The paper is organized as follows: in Chapter 2 we analyze the requirements to an informal model as needed in knowledge management and in knowledge processing. In Chapter 3 the Informal Model will be introduced. Chapter 4 contains an outline of how this Informal Model can be build up incrementally. The transformation of an Informal Model into fully formal knowledge as used in knowledge based systems will be discussed in Chapter 5. Finally, in Chapter 6 a discussion and outlook will be given.

2 REQUIREMENTS TO AN INFORMAL MODEL

Engineering knowledge tends to be very complex, diverse, and interrelated in many ways. Consequently, knowledge modeling in engineering must be based on a rich and structured representation of this knowledge, and an adequate way of user interaction for modeling and using this knowledge. The requirements to these issues will be summarized here:

Modeling requirements:

Engineering knowledge is on one side very complex (many different aspects interact), but on the other side it is also relatively well-structured. Consequently, an informal model for engineering knowledge should support structuring of large and diverse bodies of knowledge with many interdependencies. Typical structures in engineering knowledge (and elsewhere) are:

- classes with attributes, class hierarchies (taxonomies), and inheritance;
- relations and constraints between entities;
- different abstraction levels (for instance, part-of hierarchies in technical systems with assemblies and parts); and
- consistency conditions between various parts of knowledge.

Depending on the concrete application domain, there are typical *main knowledge categories* like, for instance, observations and failure modes in technical diagnosis or functions, structures, and behaviors in model based design and engineering. These main categories tend to be quite stable.

Product and process knowledge:

Engineering knowledge does not only comprise product descriptions (though also this provides considerable difficulties), it also contains knowledge about *how* to solve a given problem, i.e., process knowledge. Both sides of engineering knowledge are closely related to each other.

Requirements related to the modeling process:

Knowledge modeling is typically an iterative process. On one side, building-up a model of a certain engineering domain may need more than one loop of modeling (with improvements and refinements of previous models). On the other side, engineering knowledge is not static. Some aspects may change frequently (like concrete product and process descriptions), are quite stable (like main knowledge categories), but there is always some change in engineering knowledge bases.

User Interactions:

Due to the complexity of engineering knowledge, knowledge modeling in engineering is a complex task. Many relations and interdependencies have to be taken into account in order to come up with a model which is as differentiated, precise, generic, consistent and concise as possible. So, each new piece of knowledge which should be inserted into an existing knowledge model has to be related in many ways to the already contained knowledge. Thus, during modeling a maximum of information about the already existing model has to be available and easily accessible by the knowledge engineer.

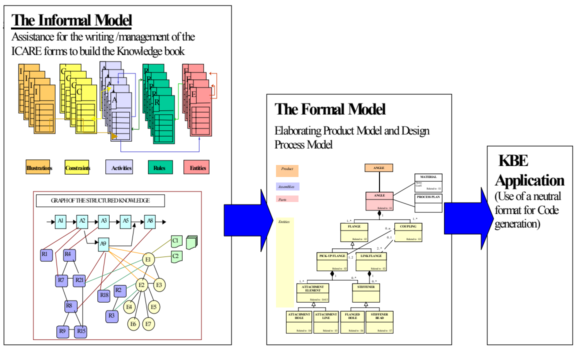

Figure 1 : The knowledge processing chain from informal to formal models (and finally KBE applications)

3 THE INFORMAL MODEL

The informal model has to fulfil two main purposes:

- It provides the basic representation of engineering knowledge for knowledge management, and
- it is a first step in the ”knowledge processing chain” (Fig. 1) from human knowledge to fully formal knowledge which then can be processed in a model based KBE system.

Following the requirements discussed in the previous chapter, the main aspects of this Informal Model can be outlined as follows:

An Informal Model is a collection of knowledge

elements, called ICARE forms[1] [4].
These elements can be related on one side to human knowledge in text documents, interview protocols, etc., and on the other side to formal knowledge in a formal model (see Fig. 1 and Chapter 5).

Entity form: is-a:	water pump assembly	
description:	The water pump system as a whole	
source:	<<link to text in Document>>	
subclasses:	directly-driven-water-pump indirectly-driven-water-pump one-stream-pump two-stream-pump	
attributes:	pump-flow rotational-speed weight shape	cm3/s 1/s kg geometric-shape
constraints: part-relations:	has-driving-syst. has-transmission has-housing has-pump-syst.	driving-syst. transmission housing pump-syst.
relations:	Cooled-motor location	Motor geometric-space

Figure 2a: Example of an entity ICARE form
The attributes and constraints are related to their potential (generic) value types

Constraint form: is-a:	has-driving-system Structural constraint Part relation
description:	The part relation from the water pump to its driving system
source:	<<link: text in Document>>
domain: range:	water-pump driving-system
conditions: cardinality constraints	[1,1]; [1,1] water-pump.rot-speed = driving-system.rot-speed

Fig. 2b: Example of a constraint ICARE form

ICARE elements can have five different types (which together give the abbreviation ICARE) and which allow us to represent the main kinds of engineering knowledge in a structured way:

- Illustrations for comments and complex explanations (to be used only for guidance and comments - not to be transformed directly into a formal model);
- Entities are the knowledge elements which describe the objects (tangible and intangible; generic and concrete) in the domain (Fig. 2a);
- Constraints represent all kinds of relationships between entities (Fig. 2b)[2];
- Activity forms are used to model the problem solving steps; and finally
- Rules represent the control knowledge.

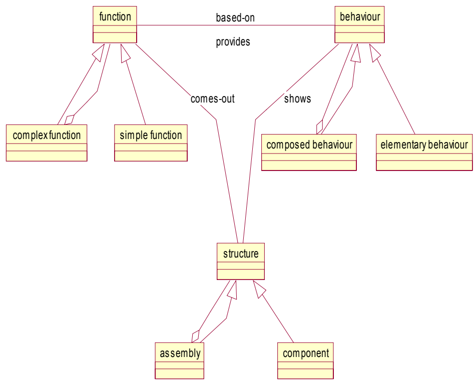

Figure 3: an example of an upper structure with main knowledge categories ‘function’, ‘structure’, and ‘behavior’, their sub-categories and some relations between them (in UML-like notation)

ICARE elements can be related to each other in various ways. This allows us to fulfil the requirements of engineering knowledge mentioned in the previous chapter:

- The main knowledge categories and their relations in a domain (like observation and failure mode in diagnosis or structure, function, and behavior in design) form an *upper structure* (Fig. 3). The entities and constraints representing these main categories provide a backbone for the whole knowledge model. All the other entity and constraint forms can be related to these upper structure elements as subclasses or instances. In this way, an expressive and flexible structuring of Informal Models can be achieved which reflects the main knowledge categories in a domain.
- Each entity can be characterized by a set of attributes in conjunction with potential values or value types (Fig. 2b).
- Based on the assignments of entities and constraints to upper structure elements we can define different *views*: a structural view, a functional or behavioral view, for instance, in

[1] The ICARE scheme described in this paper has some extensions and modifications compared with the scheme originally developed within the MOKA project [4].

[2] Entities and constraints together provide a kind of entity relationship model for the product knowledge.

design applications.

- Entities can be ordered in a taxonomy (class hierarchy – Fig. 4) and in part-of hierarchies.

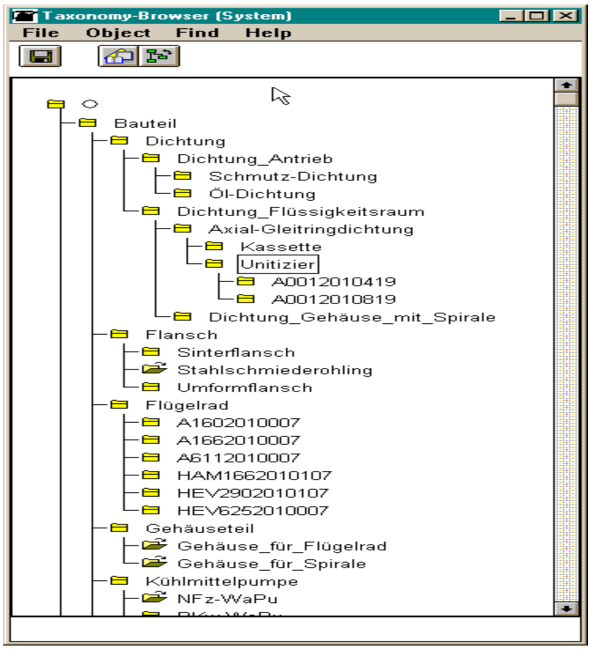

Figure 4: Example of a class hierarchy of entity forms

With these modeling capabilities the main aspects of typical engineering *product* knowledge can be represented in an adequate way. The same is true for the *process knowledge* which is represented as a set of activity and rule ICARE forms:

- Activities can be related to each other by *sequence* or precedence relations. Entities and constraints can be assigned to activities by *roles* which define in which way the entities and constraints will be used within activities.
- Activities can be hierarchically composed, i.e., a compound activity has an *internal structure* consisting of a system of more elementary activities and their relations.

The activation of an activity can be controlled by a set of rules which are associated with this activity. These rules contain control knowledge. It can be related to entities and constraints used by this activity.

4 HOW TO USE AN INFORMAL MODEL

Informal models are sets of interrelated ICARE forms (Fig. 5). The information in the forms and the relations between them together form the content of an Informal Model. Due to the complexity of engineering knowledge the knowledge modeling process is typically incremental and quite often iterative. The generation of an Informal Model is an *information gathering* and *structuring* process. Both aspects - information gathering and structuring - have to be considered when the user interactions with an Informal Model are discussed.

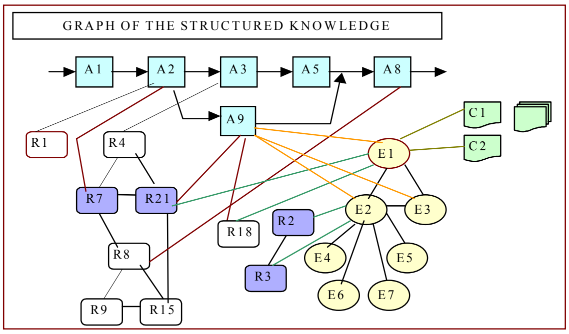

Figure 5: A graphical representation of different ICARE forms and their interrelations. The Ai are activity forms, related to each other by precedence relations; the Ei forms are entities, the Ci constraints, and the Ri rule forms.

4.1 ICARE form modeling

The set of ICARE forms an Informal Model consists of can be generated incrementally. New forms can be created during modeling, and existing forms can be modified by adding new attributes, constraints, or subclasses. The new attributes, constraints, or subclasses may still be undefined when mentioned for the first time in an ICARE form. An ICARE form management system should be aware of such situations (and generate, for instance, a corresponding warning).

Due to this incremental information gathering the Informal Model will be more and more elaborated. Starting with a very rough description of the product and process knowledge, more detailed and more precise knowledge will be filled in into the Informal Model. At the end of this process the Informal Model contains all information which is necessary to transform it into a formal model ready for use in a knowledge based engineering system.

4.2 Dependencies and hierarchies in an Informal Model

The knowledge stored in a set of ICARE forms is not independent – it is related to each other in many ways. For instance, constraint forms which represent part-of relations are typically associated with relations between attributes of the involved components (see Fig. 2b). Subclasses inherit knowledge from their superclasses which may be additionally restricted within the subclasses.

All these interdependencies have to be taken into account during modeling. For this purpose, these dependencies can be represented in a graphical form which corresponds to the types of dependencies and to the complexity of information to be dealt with:

- Class hierarchies are used to represent

superclass-subclass relations and inheritance (Fig. 4);
- Part hierarchies represent compositions (not only structural ones but also in composed functions, activities, etc.);
- Relational networks describe the ICARE forms and the various relations they can have: entity-constraint nets (Fig. 3) show the dependencies in the product model, activity charts (Fig. 5) show relations between activities, etc.

4.3 User Interactions building up an Informal Model

The main point in building an Informal model is the typically huge amount of knowledge to be dealt with in engineering knowledge modeling. Thus, it is essential that the information the user is just interested in can carefully be selected and filtered. Graphical representations are especially useful to get overviews over large amounts of related knowledge. It is necessary that graphical representations can flexibly be generated - with a maximum of control by the user. The explicit representation of the many kinds of knowledge in ICARE forms allows an efficient filtering of the information contained in an Informal Model. Especially the *upper structure*, i.e., the most general entity and constraint types, can be used for this purpose: *views* can be associated with structures, functions, behaviors, or more special kinds of entities – depending on the concrete domain to be modeled.

A close coupling between these graphical representations and the ICARE form editing capabilities has to be provided: Easy switching from a graphical representation to information directly collected in ICARE forms and vice versa.

4.4 How to build an Informal Model

Knowledge Modeling is an interaction between analyzing domain and problem solving knowledge (comprehension of the domain) and describing it within the chosen approach (here the ICARE forms of Informal Models). Typically, the first type of activities is more related to domain experts whereas the second needs some special knowledge engineering skills.

Both aspects play a different role during model generation:

The first phase is mainly related to the acquisition of the main domain knowledge categories. The upper structure of the corresponding domain has to be analyzed and modeled. This might be a complicated process with more than one iteration. Typically, a domain expert and a knowledge engineer have to interact here. The advantage of knowledge in the upper structure is its stability: once modeled it can be used and reused in large classes of applications.

Next, the knowledge of the corresponding domain has to be filled into ICARE forms. The flexibility of this representation allows the domain expert to do this job – guided by the basic knowledge categories in the upper structure. For this purpose, the domain expert may concentrate on the description of the knowledge. She fills in her knowledge into the ICARE forms as far as she is able to do – leaving aside the more complicated parts of modeling as, for instance, the definition of the various interdependencies between ICARE forms.

This latter modeling activity needs more modeling skills. Now, the knowledge engineer again is involved into the modeling process: She has to complete the knowledge filled in by the domain expert and structure it according to the modeling principles described above. Sometimes, interviews with the domain expert may be needed in order to clarify ambiguous situations or to retrieve missing knowledge.

There may be various iterations of this knowledge modeling process – incrementally refining the knowledge model, extending it to greater parts, etc.

5 FROM INFORMAL TO FORMAL MODELS

The Informal Model provides a structured representation of knowledge: a set of ICARE forms and the various types of links between them. This is an expressive and flexible way of representing structured but still informal knowledge. *Formal models* have to fulfil much more rigid requirements than informal ones. Formal knowledge must be processed on a computer in such a way that the intended meaning of the knowledge can be inferred – taking the modeling principles, hidden assumptions, the many interacting aspects of knowledge, etc. into account. All the consistency requirements which apply to a domain and to the applied modeling must be fulfilled by a formal model. This has two important consequences:
- the formal model has to be represented in a unique and computer-understandable form (a formal knowledge representation language)
- which allows the representation of the large diversity of knowledge aspects in a differentiated way.

Formal knowledge representation is an established field today [5] – at least in the area of domain knowledge (the representation of problem solving knowledge is a different topic). There are well-understood principles and rules. Some approaches (KIF [6], Ontolingua [7], CYC [8]) have been developed which can serve as standard for formal knowledge representation today. They are all based on (modified and extended) versions of First Order Logic (FOL) with a precise declarative semantics. The extensions and modifications are useful in order to provide improved structuring facilities within a knowledge modeling scheme.

The Informal Model contains a rich repertoire of knowledge structuring facilities (see Chapter 3). This is essential for building up a Formal Model – but it is not sufficient. The knowledge modeling assumptions and principles must be incorporated into this *translation* from the Informal to the Formal Model:

- The underlying open world or a closed world semantics;
- the inheritance of knowledge according to the defined taxonomies must be guaranteed;
- the various consistency conditions and constraints.

6 DISCUSSION

The Informal Model introduced in this paper is a valuable facility to structure engineering knowledge. The two main application fields are engineering knowledge management and the transition from human to formal knowledge in model based KBE applications.

We introduced a rich informal knowledge modeling approach, called ICARE forms. ICARE forms allow the representation of knowledge in a structured, but still informal way which is also comprehensible (to a large extend) by human domain experts.

A main issue in our approach was to provide knowledge structuring capabilities which are rich enough for the diverse knowledge in typical engineering domains. Especially the upper structure and the relation of ICARE forms to elements therein proved to be valuable for structuring knowledge. The same is true for the other main representational facilities like classes, taxonomies, partonomies, generic and instantiated relations and constraints, etc.

The Informal Model has proven to be a well-suited means on the complicated way from human oriented knowledge to fully formalized knowledge. Informal Models help us to bridge the gap between these two forms of knowledge.

At the same time, Informal Models can be used as a representation of knowledge in Engineering Knowledge Management. With their structuring facilities they provide a realistic and pragmatic trade-off between requirements on one side and currently available technologies.

The main open issues can be summarized as follows:

The translation of the structured knowledge in the Informal Model into a full Formal Model is a complex process. Currently, this has to be done by highly qualified and experienced knowledge engineers. It is essential for the practicability of KBE to provide efficient and adequate computer support in this translation process.

In order to reduce the efforts in the cumbersome knowledge modeling process reusability of knowledge is important. The various aspects of a domain should be modeled within ”separate chunks” of knowledge which are related to each other [9]. Thus, the development of modular knowledge representations also suitable for Informal Models has to be achieved.

In this endeavor the structuring of knowledge using upper structures is necessary. Currently, upper structures are only a ”weak form” of ontologies, with restricted expressiveness and formal rigor. Fully formalized expressive ontologies could be helpful already in the knowledge structuring process when the Informal Model is built-up. The interplay between formal and informal knowledge in the knowledge modeling process needs further investigations.

Acknowledgement

The research presented in this paper was partially funded by the European Commission within the ESPRIT-IV framework in the MOKA project (No. 25 418). The ideas and results are related to research in the MOKA project, though some modifications and extensions were applied. Figures 1 and 5 are taken from MOKA project deliverables.

References

[1] Gruber, Th.: 1993, Towards principles for the design of ontologies, *Techn. report KSL 93?04*, Stanford Univ., 1993.

[2] Guarino, N. (ed.), 1998, Formal Ontology in information systems, IOS Press, Amsterdam.

[3] Swartout, W., Tate, A., (eds.), 1999, IEEE Intelligent Systems, 1-2/1999, Special Issue on Ontologies.

[4] Callot, M., Oldham, K., Kneebone, S., Murton, A., and Brimble, R., 1998, MOKA, *Proceedings of the Conference on Integration in Manufacturing*, Goteborg, Sweden, IOS Press, Amsterdam, 1998, pp 198-207. (see also http://www.kbe.coventry.ac.uk/MOKA).

[5] Lloyd, J.W. (ed.), 1990, Computational Logic, Springer, Berlin.

[6] Genesereth, M., and Fikes, R., 1992, Knowledge Interchange Format Version 3.0 Reference Manual, *Technical Report KSL 91-1*, Stanford University Logic Group, 1992.

[7] Gruber, Th., 1993, Ontolingua: a mechanism to support portable ontologies, *Technical Report KSL 91-66*, Stanford University, Knowledge Systems Lab.

[8] Lenat, D.B., and Guha, R.V., 1990, Building large knowledge based systems - the CYC project, Addison Wesley, 1990.

[9] Borst, P., 1997, Construction of Engineering Ontologies, Ph.D. thesis, CTIT, Enschede, The Netherlands.